Biology 458 *Biostatistics* Prototypes

Week 01

2007 Biostatistics 01 - Introduction
Getting Started with the R interpreter
2007 Biostatistics 02 - Descriptive Statistics
2007 Biostatistics 03 - Graphic Display of Data

Week 02

Frequentist versus Baysian views
2007 Biostatistics 04 – Probability Distributions
2007 Biostatistics 05 – Standard & Conditional Probability
2007 Biostatistics 06 – Examples using Bayes' Rule
2007 Biostatistics 07 – Determining Risk for Families using Pedigree Analysis
2007 Biostatistics 08 – Receiver Operating Characteristic (ROC) Curves

Week 03

2007 Biostatistics 09 – Working with the Binomial Probability Distribution
2007 Biostatistics 10 – Binomial Distribution Prototyping Examples
2007 Biostatistics 11 – The Poisson Distribution

Week 04

Interpreting Probability Distribution tables & functions in R
2007 Biostatistics 12 – The Normal Distribution
2007 Biostatistics 13 – Linear Combinations of Variables
2007 Biostatistics 14 – Assessing Covariance and Correlation of Variables
2007 Biostatistics 15 – Normal Approximations for Discrete Distributions
2007 Biostatistics 16 – Normal Distribution – Prototyping Examples

Week 05

Assessing and Using the Normal Distribution on real data & QQ plots
2007 Biostatistics 17 – Point and Interval Estimation for the Normal Distribution
2007 Biostatistics 18 – Confidence Intervals for Means and Variances of a Normal Distribution
2007 Biostatistics 19 – Point and Interval Estimation of Discrete Distributions Parameters
2007 Biostatistics 20 – General Strategies for Sampling a Population

Week 06

2007 Biostatistics 21 – One Sample t-Test
2007 Biostatistics 22 – One Sample χ^2 Test of Variance for a Normal Distribution
2007 Biostatistics 23 – One Sample Tests of Discrete Distribution Parameters
2007 Biostatistics 24 – Estimating Power and Sample Size for a One Sample t-Test
2007 Biostatistics 25 – Constructing Q-Q Plots

Week 07

One and Two Sample Tests using the R Interpreter
2007 Biostatistics 26 – Paired t-Test
2007 Biostatistics 27 – Two Sample t-Test with Equal Variances
2007 Biostatistics 28 – Two Sample t-Test with Unequal Variances

2007 Biostatistics 29 – F-Test for Equal Variances in Two Samples
2007 Biostatistics 30 – POWER & Sample Size in t-Tests for Two Samples

Week 08

Nonparametric Statistics in R
2007 Biostatistics 31 – Sign Test
2007 Biostatistics 32 – Wilcoxon Signed-Rank Test
2007 Biostatistics 33 – Wilcoxon Rank-Sum Test – Mann-Whitney Test

Week 09

Tests on Contingency Tables in R
2007 Biostatistics 34 – 2X2 Contingency Tests
2007 Biostatistics 35 – McNemar's Test for Paired Data
2007 Biostatistics 36 - χ^2 Test for Association in RXC Contingency Tables
2007 Biostatistics 37 - χ^2 Test for Goodness of Fit
2007 Biostatistics 38 – Fisher's Exact Test

Week 10

2007 Biostatistics 39 – Simple Linear Regression
2007 Biostatistics 40 – ANOVA for Linear Regression
2007 Biostatistics 41 – The t-Test Approach for Interval Estimation for Simple Linear Regression
2007 Biostatistics 42 – Association and Correlation in Simple Linear Regression

Week 11

Least Squares Fitting, Linear Models, & ANOVA in R
2007 Biostatistics 43 – Multiple Regression
2007 Biostatistics 44 – Inference in Multiple Regression
2007 Biostatistics 45 – Interpreting Regression Results from Statistical Packages
2007 Biostatistics 46 – Regression and General Linear Models

Week 12

Factors & ANOVA tables for Linear Fit in R
2007 Biostatistics 47 – One-Way ANOVA with Fixed Effects Model
2007 Biostatistics 48 – F-Test for H_0: all $\alpha_i = 0$ in One-Way ANOVA Fixed Effects Model
2007 Biostatistics 49 – t-Test for H_0: $\alpha_i = \alpha_j$ versus H H_1: $\alpha_i \neq \alpha_j$ in One-Way ANOVA Fixed Effects
2007 Biostatistics 50 – t-Test for Linear Contrasts H_0:L=0 versus H_1: L≠0 ANOVA Fixed Effects

Week 13

Running ANOVA problems
2007 Biostatistics 51 – Two-Way ANOVA with Equal Sample Sizes, Fixed Effects
2007 Biostatistics 52 – Kruskal-Wallis Test
2007 Biostatistics 53 – Single and Multiple Simultaneous Confidence Intervals in ANOVA Tests
2007 Biostatistics 54 – Repeated Measures One-Way ANOVA with Fixed Effects Model
2007 Biostatistics 55 – Friedman Two-Way ANOVA by Ranks Test

Welcome to *Biostatistics!*

Please fill out a card with:

- Name
- Major & Class (i.e., year)
- A way to contact you if necessary
  such as email or telephone #
- Brief reason for taking this class...

Also, please fill out the survey...

Class Syllabus & Organization:

- This course is above all "hands on"!
- Attendance is key to success...
- Textbook will be the prime narrative.
- Read assignment each day/week before **lecture:**
- Reading assignments can be found in the **Tentative Schedule** on Blackboard
- Worksheets will be posted on Blackboard. print them out and bring to class...
- Weekly Projects due on Tuesday in class.

Grading:

- See syllabus for breakdown.
- Note: be prepared for a quiz at any time!
- Grad students: We'll talk about this later...

Building a portfolio:

- a good strategy for using statistical materials.
- importance of "prototyping"
- beware the "black box"!

Mathophobia:

"Mathophobia is an irrational and impeditive dread of mathematics. For any of a variety of reasons a student can develop this emotional and intellectual block, making further progress in mathematics and closely related fields very difficult." Mitchell Lazarus (ERIC)

In my opinion:

- important to realize the fears are ***irrational***
- everyone has them
- failure ***normally happens*** to everyone.
- math is a tool, ***not*** an IQ test.
- math represents **power** in academics & life
- the pathetic role of the 'phobe' in ceding initive & a role in decision making...
- math is **fun** to both ***fail*** and ***succeed*** in doing

Definition of Statistics:

"a branch of applied mathematics concerned with the collection and interpretation of quantitative data and the use of probability theory to estimate population parameters" (wordnet.princeton.edu)

"Statistics is the science and practice of developing knowledge through the use of empirical data expressed in quantitative form. It is based on statistical theory which is a branch of applied mathematics. Within statistical theory, randomness and uncertainty are modelled by probability theory. Because one aim of statistics is to produce the "best" information from available data, some authors consider statistics a branch of decision theory." (wikipedia.org)

Lies, Damned Lies, and Statistics

(e.g., http://www.bklein.de/statistics/)

Read Rossner Chapter 1 for a general motivation...

- Statistics have always been important in fields filled with lots of data requiring summary but having exceptions.

- Varying usefulness in fields such as Physics versus Biology.

- Many traditional uses in Psychology, Evolution & Ecology

- Growing importance in Molecular Biology & Bioinformatics

Read Rossner Chapter 2 ASAP

- We will begin addressing these topics on Thursday.

- Check Blackboard and download available Lecture worksheets and Thursday computer pod assignment.

- Follow instructions ...

- Assignments will be due in class on the following Tuesday.

Handling Data:

- Computer-based data Manipulation is key to working with modern forms of statistics.

- We will begin using:

Microsoft Excel - a spreadsheet - check for tutorial on the cd disk accompaning your text.

SPSS - In the pod. If you have reason to use another program such as **SAS, Systat, Minitab** ***please feel free***.

R - a free web-resource (**S** & **S-plus** are similar but not free!). This is rapidly gaining a major following among many different workers including theoretical mathematicians, biological researchers working in bioinformatics and many other fields.

#GETTING STARTED WITH THE R INTERPRETER

Useful functions for summarizing statistical data in R:
Note that anything prefaced by # is ignored by the R interpreter.

Examples of dataset come already installed with R that may be consulted right away.
For instance, the famous iris dataset of Anderson.
Type or cut and past the following line into the R interpreter and see what happens:

```
iris
```

Note the structure of this data table with rows (each flower often called statistical "objects" or "individuals") and columns (variables). One column includes the species name for each individual.

This kind of data is typical in statistics. In R, the structure is given a special name. Try:

```
class(iris)
```

the class "data.frame" is R's way of specifying flexible kinds of data including both numbers and character information (as in the species column) along with labels for rows and columns.
Now, for summary information on the iris dataset, try:

```
summary(iris)
```

#Now, each variable (column) of the iris dataset is summarized with minimum and maximum values, means and medians, quantiles – all good statistical information. Note also that the Species column contains counts for each of the three species names in the iris dataset.

For pairwise plots of all variables, try:

```
plot(iris)
```

#Now, you get pairwise plots of all columns. Note that some plots don't make much sense! Why? In all statistical analysis, your job will be to interpret reports such as this and decide which are meaningful and which are not.

It is often useful to be able to extract particular pieces of data from larger data tables.
In R, you can extract the columns using the symbol$. Type:

```
iris$sepal.length
```

What you get is an error message "**NULL**" meaning that R reports nothing! It is important to compare this line with the column variable label "Sepal.Length" reported above. Note the difference? Now try:

```
iris$Sepal.Length
```

The R statistical language requires that you be specific about the case for all names. It turns out we were lucky in the first place that "iris" was all in lower case letters. However, "Sepal.Length" has both upper and lower case letters, and we must type things correctly. "Sepal.Length" is different from "Sepal.length" and so on. Irritating, perhaps, but not a big problem now that you have been warned!

To avoid much typing, it is possible to simplify names for different data columns by “attaching” a datafile to the current environment. Type:

Sepal.Length

#The interpreter returns the complaint: “**`Error: object "Sepal.Length" not found`**”. But if you:

```
attach(iris)
```

#and then:

```
Sepal.Length
```

you can now view each data column by name directly. In R, the opposite of “attach” is “detach”. Try it an see what happens.

Now, let's do something useful with a single column of the iris dataset. After attaching the iris datafile to the environment, try:

```
hist(Petal.Length)
```

A histogram like this is useful for investigating the distribution of the individual measurements of this variable (called "values" in this "sample" of measurements) in order to make a guess at the distribution of all possible values (called the "population" of measurements). This distinction is very important in statistics.

#To make you histogram more useful, you can specify the number of bins using “nclass” and colors to the bars as follows:

```
hist(Petal.Length,nclass=25,col=”gray”,border=”red”)
```

#Now, let's make a scatter plot of two variables. We will place Sepal.Length on the x axis and Sepal.Width on the y axis:

```
plot(Sepal.Length,Sepal.Width,col="red",pch=21,bg="green")
```

Note that in R, as with many statistical programs, a single column of data such as Sepal.Length is called a "vector". A vector is simply an ordered list of numbers (sometimes other things) with the order indicated by an "index" indicating placement within the list.

To access individual items within a vector in R, we use []. For instance, try:

```
Sepal.Length[7]
```

#What does this number mean? Compare this with the entire data frame, and find the 7th item in vector Sepal.Length.
An entire list of data numbers, consisting of vectors side by side is called a data "matrix". The data frame "iris" consists of a data matrix of four variable vectors plus a vector of species names.
To access any piece of this information, [] may be used as well. Here, however, you must specify both row and column indices:

```
iris[3,4]
```

Can you find this number in the data matrix?
Now try:

```
iris[3,5]
```

Here the R interpreter tell you that the word "setosa" sits in this spot and is one of three possible alternatives (called "levels") including "setosa", "versicolor" and "virginica".
For a little more fun, multiple values in the data frame can be extracted by using a vector we make on the fly, using the c (concatenation function):

```
iris[c(3,4,5,6,7,),2]
```

Compare with the entire iris data set to see how this works. Here's a powerful (and cool) way to make a vector by specifying start and end points of series of numbers incrementing by one using ":". Try this and see what it does:

```
iris[c(1:6),3]
```

One of the powerful features of R, like many programming languages, is the ability to name new variables and load them with new values. This is done by use of an "assignment" operator. In R, the assignment operator "<-" or "=" (two different ways to say the same thing) place values you give it into a variable you name. Let's name a new vector variable called "NewVar" and assign it the values "1,3,5,7":

```
NewVar <- c(1,3,5,7)
```

#Now "evaluate" the variable you have just made:

```
NewVar
```

#The evaluation shows that you have placed the values in the concatenation function c()inside NewVar. Now let's make another:

```
NewVar2 = c(5,6,7,8)
```

and evaluate:

```
NewVar2
```

Easy. We can now use "functions" to do many important things. For instance, to calculate the "mean" (average) of a vector, use the built-in R function "mean()" placing whatever variable you want within the parentheses:

```
mean(NewVar)
```

How about the median, with "median()":

```
median(NewVar2)
```

Can you find the median of Sepal Length in iris:

```
median(Sepal.Length)?
```

Or, how about the mean of the first 50 rows in of iris for Petal Width?

```
mean(iris$Petal.Width[1:50])
```

There are many other useful functions in R, such as:

```
min(NewVar)                  # minimum value in vector
max(NewVar2)                 # maximum value in vector
sum(NewVar)                  # adding the elements of the vector.
length(NewVar2)              # finding how many numbers occur in the vector.
var(Petal.Length[1:50])      # for the variance of Petal Length for iris setosa.
```

Many more functions may be found by typing:

```
help.start()
```

As you have probably noticed, learning about and remembering the syntax of a programming language such as R is a major challenge and fundamental to using it effectively. To find more information about any built in function in R, type a "?" followed by function name, eg:

```
?var
```

#It is often very useful to look at some variables according to values exhibited by another. For instance, looking at the iris dataset:

iris

one can see that data for iris species “setosa” are found in the first 50 lines, data for “versicolor” in the next 50 lines, and for “virginica” in the last 50 lines. We can calculate the number of lines, minimum value, maximum value, mean value, standard deviation, and variance for one variable by applying the above functions. For Sepal.Length in species “versicolor”, try:

```
length(Sepal.Length[51:100])
min(Sepal.Length[51:100])
max(Sepal.Length[Species==”versicolor”])
mean(Sepal.Length[Species==”versicolor”])
sd(Sepal.Length[Species==”versicolor”])
var(Sepal.Length[Species==”versicolor”])
```

As you can see, this is somewhat tedious, and requires manually checking rows in the iris dataset to determine which belong to the species “versicolor”. Alternatively, one can use “==” (double equal sign indicating logical evaluation rather than assignment) and allow R to do the counting for you. An easier way to combine such functions is to use the function called “tapply” creating variables like this:

```
xbar=tapply(Sepal.Length,Species,mean)
n=tapply(Sepal.Length,Species,length)
mn=tapply(Sepal.Length,Species,min)
mx=tapply(Sepal.Length,Species,max)
s=tapply(Sepal.Length,Species,sd)
v=tapply(Sepal.Length,Species,var)
```

and then using the “column combine: function:

```
cbind(“NUMBER”=n,
“MINIMUM”=mn,
“MAXIMUM”=mx,
“MEAN”=xbar,
“STD DEV”=s,
“VARIANCE”=s)  #Note use of multiple lines only to make this more readable! R doesn’t care.
```

The result is a tabulation of these variables for each species in turn. Note in the command above, that words in “ ” are used to specify labels; the symbol ‘ ‘ work also, but should not be intermixed.

We can histogram each now by making the following Sepal.Length variables:

```
SL.setosa=Sepal.Length[Species=="setosa"]
SL.versicolor=Sepal.Length[Species=="versicolor"]
SL.virginica=Sepal.Length[Species=="virginica"]
```

Then formatting using the function"mfcol" for making 3 rows and 1 column:

```
 par(mfcol=c(3,1))
```

followed by making the histograms:

```
hist(SL.setosa,nclass=15,col="red")
hist(SL.versicolor,nclass=15,col="blue")
hist(SL.virginica,nclass=15,col="green")
```

After that, it is a good practice to reset the plotter back to a single plot:

```
par(mfcol=c(1,1))
```

unless you intend to continue plotting graphs in groups of three indefinitely. A similar function "mfrow" allows graphing in rows instead of columns:

```
par(mfrow=c(1,3))
hist(SL.setosa,nclass=15,col="red")
hist(SL.versicolor,nclass=15,col="blue")
hist(SL.virginica,nclass=15,col="green")
par(mfrow=c(1,1))

hist(SL.setosa,nclass=15,col="red")
hist(SL.versicolor,nclass=15,col="blue")
hist(SL.virginica,nclass=15,col="green")
```

To allow comparison between histograms, limits based on maximum and minimum values (observed on the graphs or calculated above) can be applied to the x and y axes:

```
par(mfcol=c(3,1))
hist(SL.setosa,nclass=15,col="red",
xlim=c(4,8),ylim=c(0,10))
hist(SL.versicolor,nclass=15,col="blue",
xlim=c(4,8), ylim=c(0,10))
hist(SL.virginica,nclass=15,col="green",
xlim=c(4,8), ylim=c(0,10))
par(mfcol=c(1,1))
```

Now for making scatter plots with multiple coded points, we make variables by extracting Sepal.Width for each Species:

```
SW.setosa=Sepal.Width[Species=="setosa"]
```

```
SW.versicolor=Sepal.Width[Species=="versicolor"]
SW.virginica=Sepal.Width[Species=="virginica"]
```

Now for we make a plot using function “plot”. Limits xlim and ylim are specified to allow plotting of all points in the graph. We then add points for the others using function “points”:

```
plot(SL.setosa,SW.setosa,pch=19,col="red",
     xlim=c(4,8),ylim=c(2,4.5))
points(SL.versicolor,SW.versicolor,pch='v',col="blue",
     xlim=c(4,8),ylim=c(2,4.5))
points(SL.virginica,SW.virginica,pch=22,col="green",
     xlim=c(4,8),ylim=c(2,4.5))
```

Of course, points are color coded using “col” and different symbols are used using “pch”. To find available options, enter:

SAVING PLOTS:

To save your histograms or plots, it is a simple matter of cutting and pasting them into your favorite word processor such as MS Word. They can then be printed out in the normal way.

```
?points
```

READING AND WRITING DATA:

Writing and Reading data from external files is an important aspect of any statistical analysis. Simple text files are the most general way to exchange data between formats and programs as nearly all have ability to do this in one way or another. To write the “iris” data table to a text file, the easiest way is to cut and paste. Open a text file editor, and then cut and paste normally. Be sure to include the first line containing names of the variables. Use your text editor to make a simple text file named “iris.txt” and place this within R’s working directory. You can find out where the working directory is located by looking under “File/Change dir” on the R console.

After writing the file, let’s see how to read it back into R. For this, we will make a new variable called “newIris”. Use the function “read.table” and then list the file.

```
newIris=read.table(“iris.txt”)
newIris
```

As you can see, read.table in R has correctly interpreted you “iris.txt” file and read all the data points into the appropriate columns. From this, you can obtain summary information like before:

```
summary(newIris)
```

To convert import iris.txt in to MS Excel, open the program and then under “File/Open” choose R’s working directory, and “All Files” in the “Files of type” box. Open “iris.txt” and follow Excel’s formatting instructions. Click the “Delimited” radio button with “Start import at row 1”

then, click “Next”. Check the “Space” box and now fields delimitation is shown by vertical lines. The lines should correctly separate each data point is in its own field. Now click “Next”. Now you can change data format or just accept the defaults, click “Finish”. At this point, everything should look like the original and you can save the file as a normal Excel worksheet. To reverse the process and import a data file into R from Excel, it’s best to have Excel write a simple text file. Open your data in Excel, and choose “File/Save As…” Make a new name for your file such as: “IrisFromExcel” and “Tab delimited Text” in the “Save as type” box. Excel then complains that changing to text format may loose formatting information, but say “Yes” anyway. Now exit Excel WITHOUT SAVING (this preserves your original file in Excel). Now, on the R console, make a new variable and use function “read.table” again:

```
Iris2=read.table(“IrisFromExcel.txt”)
```

And to verify all went well:

```
summary(Iris2)
Iris2
```

Descriptive Statistics

Interpreting MathCad Worksheets:

For classes such as this, where it is useful to make documents with math symbols, graphs, etc, I find the program MathCad to be quite useful. This program makes available an extensive library of mathematics functions allowing import, export, and manipulation of data in real-time. It also allows me to document what I have done using familiar mathematics symbols directly comparable to that seen in the text, and lots of words in the worksheet itself. For the purpose of prototyping statistical procedures and tests, I find the combination ideal.

Note, however, that I do not require that you buy MathCad as I will make these sheets available to you in both MathCad (*.mcd) and in Adobe Acrobat (*.pdf) formats.

To get started, this worksheet is designed to provide an overview of what you might expect to see in lecture worksheets from now on.

$ORIGIN \equiv 0$ **<- I normally put this in all my worksheets to standardize use of index variables across all my worksheets. It is an example of "global assignment" (using the symbol ~ on the keyboard). Don't worry about what it means at this point.**

^ OK, so now you see how I normally label things...

Calculations:

$2 + 2 = 4$ **<- Calculations are done in the normal way using familiar symbols.**

$\frac{35}{5} = 7$

$6 \cdot 5 = 30$

$\pi = 3.142$ **<- Some common mathematical values are built in the program...**

Assignment versus Evaluation:

<- Variables may be named at any time. Note, however, that there are two distinct meanings here for what we normally term "equals"!

$Var := 78$ $Var = 78$

^ assignment **^ evaluation**

Assignment (indicated by use of : and shown on the worksheet as :=) means "put the numerical value 79 into a variable I now name Var".

$var2 := 6 \cdot \frac{7}{9}$ $var2 = 4.667$

Evaluation (indicated by use of =) means "tell me what value is placed in the already named variable Var".

$var3 := \frac{(5 + 3)}{\sqrt{\pi}}$ $var3 = 4.514$

This distinction is important and common to most programming languages...

Data Input:

iris := READPRN("c:/2007BiostatsData/iris.txt")

^ This is easy using the built-in READPRN() function for simple text format data. When prototyping, using an existing worksheet, I can read in different data files and calculate things in exactly the same way. A worksheet showing how to do a specific statistical test, for instance, is critical for evaluating output from canned programs that might otherwise appear to be a "BLACK BOX".

iris =

	0	1	2	3	4
0	1	5.1	3.5	1.4	0.2
1	2	4.9	3	1.4	0.2
2	3	4.7	3.2	1.3	0.2
3	4	4.6	3.1	1.5	0.2
4	5	5	3.6	1.4	0.2
5	6	5.4	3.9	1.7	0.4
6	7	4.6	3.4	1.4	0.3
7	8	5	3.4	1.5	0.2
8	9	4.4	2.9	1.4	0.2
9	10	4.9	3.1	1.5	0.1
10	11	5.4	3.7	1.5	0.2
11	12	4.8	3.4	1.6	0.2
12	13	4.8	3	1.4	0.1
13	14	4.3	3	1.1	0.1
14	15	5.8	4	1.2	0.2
15	16	5.7	4.4	1.5	0.4

<- Evaluation of variable iris.

Note that the display is often a *partial* list that may be scrolled in the normal manner like a spreadsheet. The variable might also be displayed in matrix form...

SL := $iris^{\langle 1 \rangle}$
SW := $iris^{\langle 2 \rangle}$
PL := $iris^{\langle 3 \rangle}$
PW := $iris^{\langle 4 \rangle}$

<- New variables are now named and assigned to the values in different columns of the dataset iris. Note that columns start their numbering with '0'. This is the result of my ORIGIN assignment above. The first column of numbers is merely the row number. Scrolling down one can see that there are 150 rows. Species names in column 5 didn't import here as MathCad interpreted the data to be numeric... Statistics programs such as R will do a better job with this. However, we won't worry about it for our purposes here.

length(SL) = 150

<- using built-in function length() to evaluate the number of rows (i.e., objects = flowers) inside our variable SL. This is useful.

length(SW) = 150
length(PL) = 150
length(PW) = 150

<- Evaluating the other variables. The result is hardly a surprise, but a useful check anyway in case something went wrong in using function READPRN() above.

Descriptive Statistics:

mean:

$n := length(SL)$ $n = 150$

$i := 0 .. n - 1$ **<- sets up a list of numbers from 0 to 149.**

$SL_2 = 4.7$ **Scroll on the evaluation here -> to see them all!**

i =

0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15

^ Note: this is the value in row 2 of variable SL, called by using the index ('[' left bracket on the keyboard).

$X_{bar} := \frac{1}{n} \cdot \sum_i \left(SL_i\right)$ **<- X.bar is the name of the variable. The bar part is shown as a subscript...**

^ Sum values of SL over all rows and divide by n.

prototype for mean:

$X_{bar} = 5.843$ **<- Evaluation of X.bar**

$mean(SL) = 5.843$ **<- compare with MathCad's built-in function mean(). Our method for calculating a mean is confirmed.**

median:

$SL_{sort} := sort(SL)$ **<- using MathCad's sort() function to rearrange the values of SL in order.**

$midpoint := \frac{n}{2}$ $midpoint = 75$ **<- figuring the midpoint (i.e., halfway) index**

$medianSL := \frac{1}{2}\left(SL_{sort_{midpoint}} + SL_{sort_{midpoint+1}}\right)$ **<- variable SL.sort indexed by midpoints**

^ When the number of values in a variable are even, the definition of median requires that we average the two closest points.

prototype for median:

$medianSL = 5.8$

$median(SL) = 5.8$ **<- our explicit calculation matches MathCad's built-in function median().**

$mean(SL) = 5.843$	$median(SL) = 5.8$
$mean(SW) = 3.057$	$median(SW) = 3$
$mean(PL) = 3.758$	$median(PL) = 4.35$
$mean(PW)$	$median(PW) = 1.3$

<- Having prototyped one, we now have confidence that we know how to calculate ALL of these!

Descriptive Statistics:

sample variance and sample standard deviation:

$$var_{SL} := \frac{1}{n-1} \cdot \left[\sum_i \left[\left(SL_i - mean(SL) \right)^2 \right] \right]$$

<- applying formula for sample variance. Variable SL is indexed by previously defined index i. with mean(SL) as prototyped above for mean.

$$standdev_{SL} := \sqrt{var_{SL}}$$

<- Standard deviation is the square root of variance

prototype:

$var_{SL} = 0.686$ $var(SL) = 0.681$

<- Note the difference. MathCad's built-in function must be calculating population variance!

$$\frac{n}{n-1} \cdot var(SL) = 0.686$$

<- This converts population variance into sample variance. Matches our calculation and confirms what MathCad is doing.

$standdev_{SL} = 0.828$ $stdev(SL) = 0.825$

<- Again, doesn't match for same reason.

$$\sqrt{\frac{n}{n-1} \cdot var(SL)} = 0.828$$

<- Again converting to population standard deviation.

This section displays the value and power of making prototypes! In statistical analysis, it is *very important* to understand *exactly* what you are doing using a computer-based statistical program. For minor reasons like here, a program may be doing something subtly different different than you expect. Without making a prototype the first time you use a procedure, you might end up reporting, and perhaps trying to publish, an ERROR... THIS CAN BE VERY EMBARRASSING!

Properties of Mean and Variance:

Often, one has a choice in the units employed in measuring or counting a property. For instance, one might decide to measure temperature in either degrees Celsius or Fahrenheit. Conversion from one measurement to the other typically involves translation (adding or subtracting a constant) and scaling (multiplying a measurement by a constant). Translation and scaling together may be summarized by the following fomula, where x is the original measurment and y is a measurement "transformed" by translation and scaling.

$$y := a \cdot x + b$$

where c is the multiplication constant in scaling, and b is the translation constant.

You might recognize this formula as the equation for a line. As a result, transformations of this kind are often called linear transformations.

In statistics, we are interested in what happens to means and variance when original measurements are modified in this way.

translation:

$b := 5$

$translated_{SL} := SL + b$

^ Let b = 5 in the linear transformation above ->

SL =

	0
0	5.1
1	4.9
2	4.7
3	4.6
4	5
5	5.4
6	4.6
7	5
8	4.4
9	4.9
10	5.4
11	4.8
12	4.8
13	4.3
14	5.8
15	5.7

$translated_{SL}$ =

	0
0	10.1
1	9.9
2	9.7
3	9.6
4	10
5	10.4
6	9.6
7	10
8	9.4
9	9.9
10	10.4
11	9.8
12	9.8
13	9.3
14	10.8
15	10.7

$mean(SL) = 5.843$

$mean(translated_{SL}) = 10.843$

$mean(SL) + b = 10.843$

^ translation shifts the mean value by b

$var_{SL} = 0.686$

$var_{SLt} := \frac{n}{(n-1)} \cdot var(translated_{SL})$

$var_{SLt} = 0.686$

^ translation does nothing to variance. Since standard deviation is the square root of variance, translation does nothing to standard deviation as well.

scaling:

$c := 5$

$scaled_{SL} := c \cdot SL$

^ Let c = 5 in the linear transformation above ->

SL =

	0
0	5.1
1	4.9
2	4.7
3	4.6
4	5
5	5.4
6	4.6
7	5
8	4.4
9	4.9
10	5.4
11	4.8
12	4.8
13	4.3
14	5.8
15	5.7

$scaled_{SL}$ =

	0
0	25.5
1	24.5
2	23.5
3	23
4	25
5	27
6	23
7	25
8	22
9	24.5
10	27
11	24
12	24
13	21.5
14	29
15	28.5

$mean(SL) = 5.843$

$mean(scaled_{SL}) = 29.217$

$mean(SL) \cdot c = 29.217$

^ scaling multiplies the mean by the same factor c as each of the values in SL

$var_{SL} = 0.686$

$var_{SLs} := \frac{n}{(n-1)} \cdot var(scaled_{SL})$

$var_{SLs} = 17.142$ $var_{SL} \cdot c^2 = 17.142$ **<- scaling multiplies observed variance by factor c^2.**

$standdev_{SLs} := \sqrt{var_{SLs}}$ $standdev_{SLs} = 4.14$

<- scaling multiplies observed standard deviation by factor c.

$standdev_{SL} \cdot c = 4.14$

linear transformation:

$c := 1.8$ $b := 32$ **<- note that these values would convert a degree measurement in Celsius in to the equivalent value on the Fahrenheit scale.**

$transformed_{SL} := c \cdot SL + b$

^ Let c = 1.8 & b = 32 in the linear transformation above ->

$mean(SL) = 5.843$

$mean(transformed_{SL}) = 42.518$

$c \cdot mean(SL) + b = 42.518$

^ scaling multiplies the mean by the same factor c as each of the values in SL and adds factor b

$var_{SL} = 0.686$

$var_{SLtrans} := \frac{n}{(n-1)} \cdot var(transformed_{SL})$

$SL =$

	0
0	5.1
1	4.9
2	4.7
3	4.6
4	5
5	5.4
6	4.6
7	5
8	4.4
9	4.9
10	5.4
11	4.8
12	4.8
13	4.3
14	5.8
15	5.7

$transformed_{SL} =$

	0
0	41.18
1	40.82
2	40.46
3	40.28
4	41
5	41.72
6	40.28
7	41
8	39.92
9	40.82
10	41.72
11	40.64
12	40.64
13	39.74
14	42.44
15	42.26

$var_{SLtrans} = 2.222$ $var_{SL} \cdot c^2 = 2.222$ **<- scaling multiplies observed variance by factor c^2 and translation in b has no effect.**

$standdev_{SLtrans} := \sqrt{var_{SLtrans}}$

$standdev_{SLtrans} = 1.491$ **<- scaling multiplies observed standard deviation by factor c and translation in b has no effect.**

$standdev_{SL} \cdot c = 1.491$

Graphic Display of Data

ORIGIN ≡ 0

iris := READPRN("c:/2007BiostatsData/iris.txt") **<- Input iris same dataset as before**

SL := $iris^{\langle 1 \rangle}$
SW := $iris^{\langle 2 \rangle}$
PL := $iris^{\langle 3 \rangle}$
PW := $iris^{\langle 4 \rangle}$ **<- Column variables as before**

n := length(SL) n = 150

i := 0 .. n − 1

Evaluation of variable iris. ->

iris =

	0	1	2	3	4
0	1	5.1	3.5	1.4	0.2
1	2	4.9	3	1.4	0.2
2	3	4.7	3.2	1.3	0.2
3	4	4.6	3.1	1.5	0.2
4	5	5	3.6	1.4	0.2
5	6	5.4	3.9	1.7	0.4
6	7	4.6	3.4	1.4	0.3
7	8	5	3.4	1.5	0.2
8	9	4.4	2.9	1.4	0.2
9	10	4.9	3.1	1.5	0.1
10	11	5.4	3.7	1.5	0.2
11	12	4.8	3.4	1.6	0.2
12	13	4.8	3	1.4	0.1
13	14	4.3	3	1.1	0.1
14	15	5.8	4	1.2	0.2
15	16	5.7	4.4	1.5	0.4

Descriptive Statistics:

mean and median:

mean(SL) = 5.843 median(SL) = 5.8

mean(SW) = 3.057 median(SW) = 3

mean(PL) = 3.758 median(PL) = 4.35

mean(PW) median(PW) = 1.3

sample variance and sample standard deviation:

$var_{SL} := \frac{n}{n-1} \cdot var(SL)$ $var_{SL} = 0.686$ $std_{SL} := \sqrt{var_{SL}}$ $std_{SL} = 0.828$

$var_{SW} := \frac{n}{n-1} \cdot var(SW)$ $var_{SW} = 0.19$ $std_{SW} := \sqrt{var_{SW}}$ $std_{SW} = 0.436$

$var_{PL} := \frac{n}{n-1} \cdot var(PL)$ $var_{PL} = 3.116$ $std_{PL} := \sqrt{var_{PL}}$ $std_{PL} = 1.765$

$var_{PW} := \frac{n}{n-1} \cdot var(PW)$ $var_{PW} = 0.581$ $std_{PW} := \sqrt{var_{PW}}$ $std_{PW} = 0.762$

range:

min(SL) = 4.3 max(SL) = 7.9

min(SW) = 2 max(SW) = 4.4

min(PL) = 1 max(PL) = 6.9

min(PW) = 0.1 max(PW) = 2.5

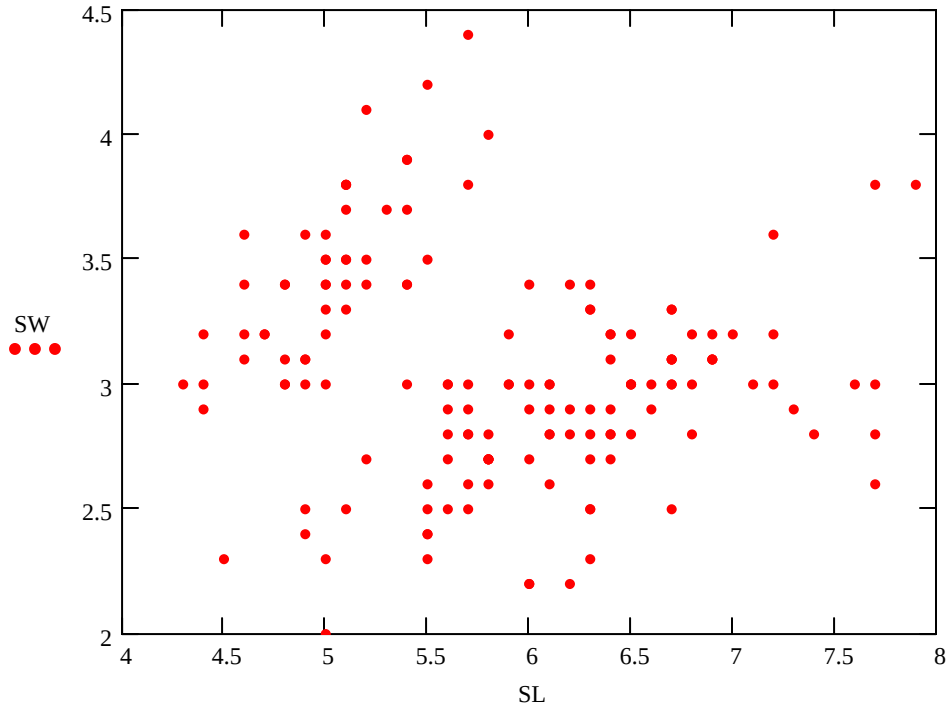
<- using built-in minimum and maximum functions

coefficient of variation:

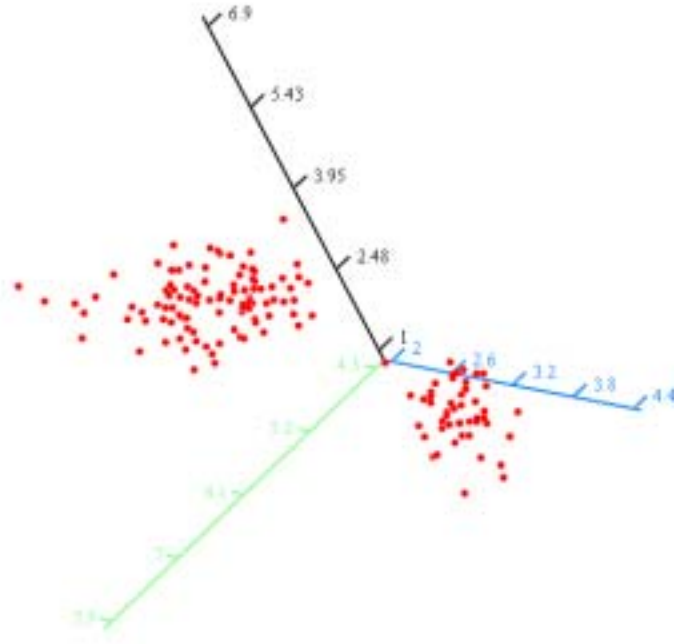
$cv_{SL} := \frac{std_{SL}}{mean(SL)}$ $cv_{SL} = 0.142$ $cv_{PL} := \frac{std_{PL}}{mean(SL)}$ $cv_{PL} = 0.302$

$cv_{SW} := \frac{std_{SW}}{mean(SL)}$ $cv_{SW} = 0.075$ $cv_{PW} := \frac{std_{PW}}{mean(SL)}$ $cv_{PW} = 0.13$

Scatter Plots:



^ Any pair of variables can be plotted to look for patterns...



(SL, SW, PL)

^ Same idea looking a three variables at a time...

Histograms:

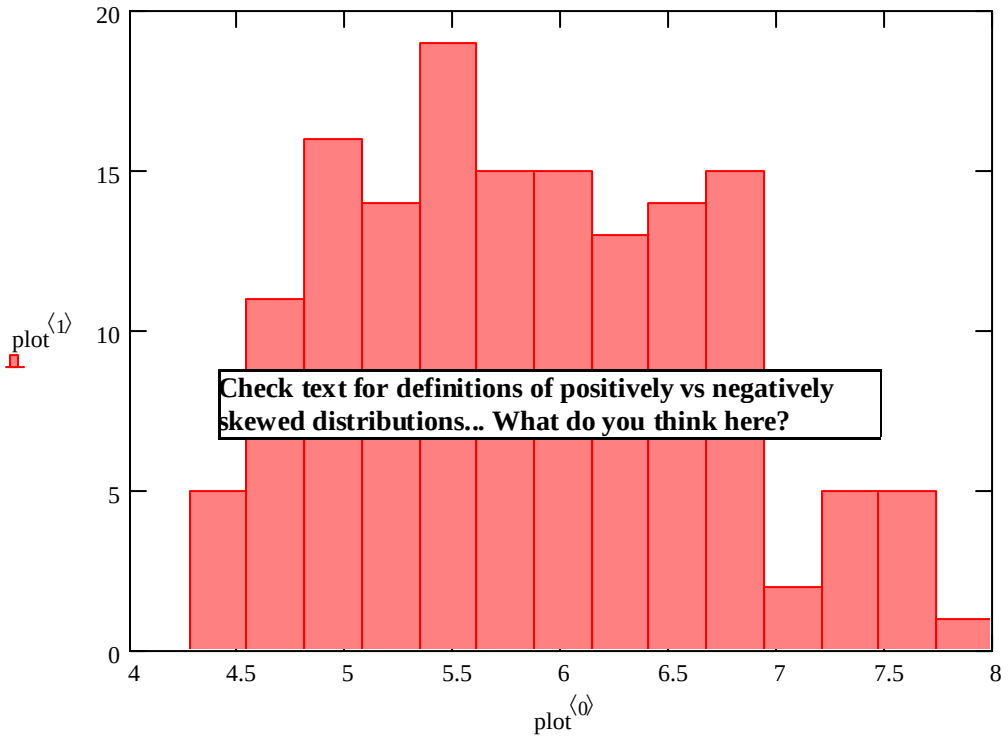
plot := histogram(15, SL)

^ **variable plot contains two columns:**

column 0: x axis = number of bins
column 1: y axis = count in each bin

plot =

	0	1
0	4.133	0
1	4.4	5
2	4.667	11
3	4.933	16
4	5.2	14
5	5.467	19
6	5.733	15
7	6	15
8	6.267	13
9	6.533	14
10	6.8	15
11	7.067	2
12	7.333	5
13	7.6	5
14	7.867	1



Stem & Leaf Plots and Box plots:

Plots of this kind generally require a more sophisticated system more directly related to statistical analysis than MathCad worksheets.

Go to SPSS, Systat, or R...

Assignment for Week 2

This week's reading assignment is admittedly a difficult one. The chapter goes well beyond what's really necessary for an introduction to biostatistics, ***but please read it anyway!*** At this point, I do not expect that you will be able to remember all of it, or that you will be able to work many of the more difficult problems at the end of the chapter. This material is waiting for you in the future as your experience with statistics increases, or when needed. Personally, I do this often - usually learning something new (or forgotten) each time I return to a subject – and it's nice to have somewhere familiar to start.

The purpose in reading this section now is to become familiar with some of the basic terminology associated with probability, and to get a feel for how probability is used in real-life clinical studies and other situations. For our purposes, please familiarize yourself with the ***basic logic of probability*** seen in the first part of the chapter and in Lecture Worksheet 05. As you can see, calculations of ***multiplied***, ***added***, or ***conditional*** **probabilities** are central to many of these endeavors, as are concepts of ***mutually exclusive***, ***potentially co-occurring*** **events**, and ***dependent*** versus ***independent*** **variables**. Care in framing probabilities with regard to the above concepts, and in setting up appropriate study questions, are key to obtaining important results in each case. The text provides a wealth of examples about how to compute derived calculations for different real-life situations. The examples and problems thus serve as templates waiting for you as the need may arise. In conducting your own statistical analysis, there may be a problem that has a similar basic structure to one or more of these. At that point, the examples become critically relevant. You should work through the problems to master the appropriate calculations in a prototype sheet. After that, the techniques you have learned can be applied to the problem at hand with confidence.

In reading this chapter for the first time, it is also interesting to see how ***debate about the use of statistics*** is framed, such as between the "**Frequentist**" versus "**Bayesian**" views of **probability** and **statistical inference**. I found this interesting as I am increasingly asked about these topics (usually missing from introductory texts) motivated by recent developments in different biological fields. I spent some time working through this material, so Lecture Worksheets 06-08 are intended as beginning prototypes for those who may wish to delve into the topic further. Some of you, especially graduate students, may have already encountered Bayesian risk assessments, already. If interested, I will be happy to assist in these areas. We'll have to learn together!

For our project this week, I would like us to turn our attention to some practical aspects of data simulation and analysis. Consult Lecture Worksheet 04 for a beginning discussion of probability distributions. Next week and the following, we will look at particular distributions such as the **Binomial Distribution**, **Poisson Distribution**, **Normal Distribution** and **Chi-Square Distribution** in much greater detail. At this point, I think it would be useful for us to become familiar with their basic properties by constructing **simulated populations**, graphing them, and calculating a few descriptive statistics. We can then compare our simulated samples with the **theoretical properties** of a perfectly distributed **population** for each distribution. The R statistical system is ideal for this, as it has built-in a wide range of functions, so this is an opportunity to become more familiar with this powerful tool also. Excel will do some of what we have in mind here, but a lot more would have to be done by hand.

For this week, divide into groups of two or three. Make sure at least someone has a working version of R. If not, let's spend some time trying to get R going on your machines. Please bring your computer to class if possible. Don't worry if that's not possible, since you will be working in groups… However, since everyone said they had access to their own computers at home, now is the time to get R installed and running. I can help you with this in lab if necessary.

The project this week is a simple one! Consult the html Help section of the R console for definition and syntax of the statistical functions you will use. For each distribution below, use the appropriate function (prefaced with 'r') to generate 1000 data random data points and assign the vector created to a variable name. Now using whatever program(s) you wish, histogram the data and investigate each set of data points using appropriate descriptive statistics. Try different parameter values for these distributions to see what they do. At this point, don't worry very much about what they mean. We will look into that shortly.

Now use the appropriate function (prefaced with 'd') to construct a population distribution using the parameter values you have used before. For this function you will have to construct a vector containing a series of X variables for which the function returns P(X). Plot this function and compare with your histograms.

For the **Binomial Distrubution**: parameters you will have to specify **n** = number of trials, **k** = number of "successes" & **p** = probability of successes. Try different values of each in turn (keeping the others constant) and see what you get!

For the **Poisson Distribution**: you will have to specify parameter $\boldsymbol{\lambda}$ (lambda) the expected number of events over unit time. Vary $\boldsymbol{\lambda}$ to see its effect on the distribution of your points.

For the **Normal Distribution:** you will have to specify $\boldsymbol{\mu}$ = mean, and $\boldsymbol{\sigma^2}$ = variance of the distribution. Try different values of each holding the other constant to see the effect.

For the **Chi-Square Distribution**: you will have to specify **df** = degrees of freedom. Vary this to see the distribution change.

Due next Tuesday in Class.

Probability Distributions

ORIGIN := 1

Statistics is based upon comparisons of measurements collected from one or more limited samples, with what might be the expected values characterizing the underlying population from which the samples have been drawn. In fact, these expected values are sometimes/always not easily determined. Important assumptions are always involved linking samples with populations and these assumptions underlie the usefulness of descriptive statistics, such as mean and variance.

Eductive Inference:

eductive: "Tending to draw out; extractive."
http://www.thefreedictionary.com/Eductive

Statistics is typically based on a pair of quantities:

x <- observed sample values
P(x) <- probability of the sampled values under some model of probability.

Models of probability differ depending on what's being analyzed and are generally of two types:

Discrete <- Here only a limited number of values are expected such as "heads" versus "tails" in a coin toss, or "1", "2", "3", "4", "5", or "6" in a roll of a single die.

Continuous <- Here an infinite (or nearly so) number of observations are possible as in measuring temperature, length, weight, etc. of some animal.

In either case, specific observations (x) are associated with probability P(x) using Probability Density functions where the area under the curve gives the probabilty for each value of x.

Example Discrete Probability Density functions:

coin toss:

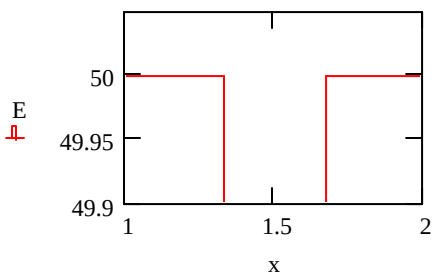
There are two possible observations: H <- "heads" = 1
T <- "tails" = 2

$x_1 := 1$
$x_2 := 2$ $\quad x = \begin{pmatrix} 1 \\ 2 \end{pmatrix}$

For a fair coin: P(H) = 1/2
P(T) = 1/2

For 100 coin tosses, expected number of H = 100(1/2) = 50
expected number of T = 100(1/2) = 50

$E_1 := 50$
$E_2 := 50$ $\quad E = \begin{pmatrix} 50 \\ 50 \end{pmatrix}$



<- Two classes in x have equal numbers

single die:

There are six possible observations: "1" = 1
"2" = 2
"3" = 3
"4" = 4
"5" = 5
"6" = 6

$i := 1 .. 6$

$x_i := i$

$$x = \begin{pmatrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

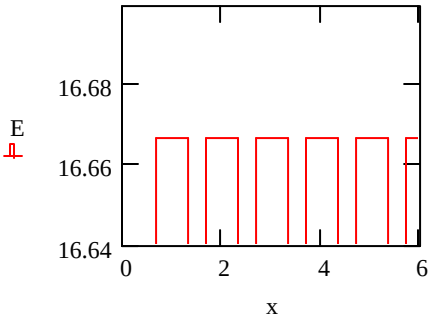
For a fair die, all probabilities are 1/6 for obtaining one of the numbers on any throw:

$P := \frac{1}{6}$ $P = 0.167$

For 100 die tosses, expected number for each:

$E_i := P \cdot 100$

$$E = \begin{pmatrix} 16.667 \\ 16.667 \\ 16.667 \\ 16.667 \\ 16.667 \\ 16.667 \end{pmatrix}$$



<- Six classes in x with equal values that need not be whole numbers

Binomial distribution:

ORIGIN := 0

If one conducts multiple trials with two possible outcomes, such as tosing a coin resulting in either a "heads" or "tails", the expected number of "heads" in a set of trials follows the binomial distribution.

total number of trials (n):

$n := 20$

probability of obtaining a heads (p) (more genererally termed "success")

$p := \frac{1}{2}$

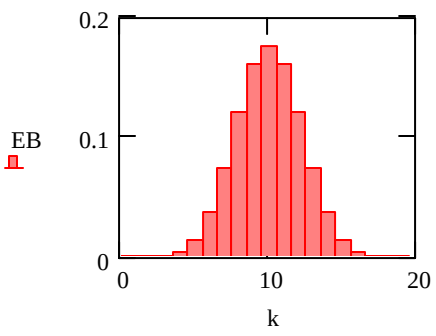
number of times one obtains a "heads" Note that this is a range of discrete possibilities (ranging from 0 to n)

$i := 0 .. n - 1$

$k_i := i$

Expected probability for each k: (E_k):

$EB := dbinom(k, n, p)$



<- It is unlikely to find 0 or 20 heads as outcome of 20 coin tosses. An intermediate number is much more likely.

EB =

	0
0	$9.537 \cdot 10^{-7}$
1	$1.907 \cdot 10^{-5}$
2	$1.812 \cdot 10^{-4}$
3	$1.087 \cdot 10^{-3}$
4	$4.621 \cdot 10^{-3}$
5	0.015
6	0.037
7	0.074
8	0.12
9	0.16
10	0.176
11	0.16
12	0.12
13	0.074
14	0.037
15	0.015

Example Continuous Probability Density functions:

Normal Distribution:

Many forms of data are continuous, so the probability function is continuous and the area under the curve represents probability (often called "probability density"). Normal distributions are common, and underly many statistical methods.

$n := 50$ **<- Out of all possible values, we will arbitrarily look at a set of n points. At the scale we plot things here, this might as well be continuous...**

$i := 0 .. n$

$b := -\left(\frac{1}{2} \cdot n\right)$ $c := 0.1$ **<- Arbitrary scaling factors so we can see things in the plot.**

$x_i := c \cdot (i + b)$ **<- Individual scaled values we plot on our x axis below.**

$\mu := 0$ $\sigma_{sq} := 1$ **<- parameters of the standard normal curve where μ is the mean of the distribution and $\sigma^2 = \sigma^{sq}$ is the variance**

$EN_A := dnorm(x, \mu, \sigma_{sq})$

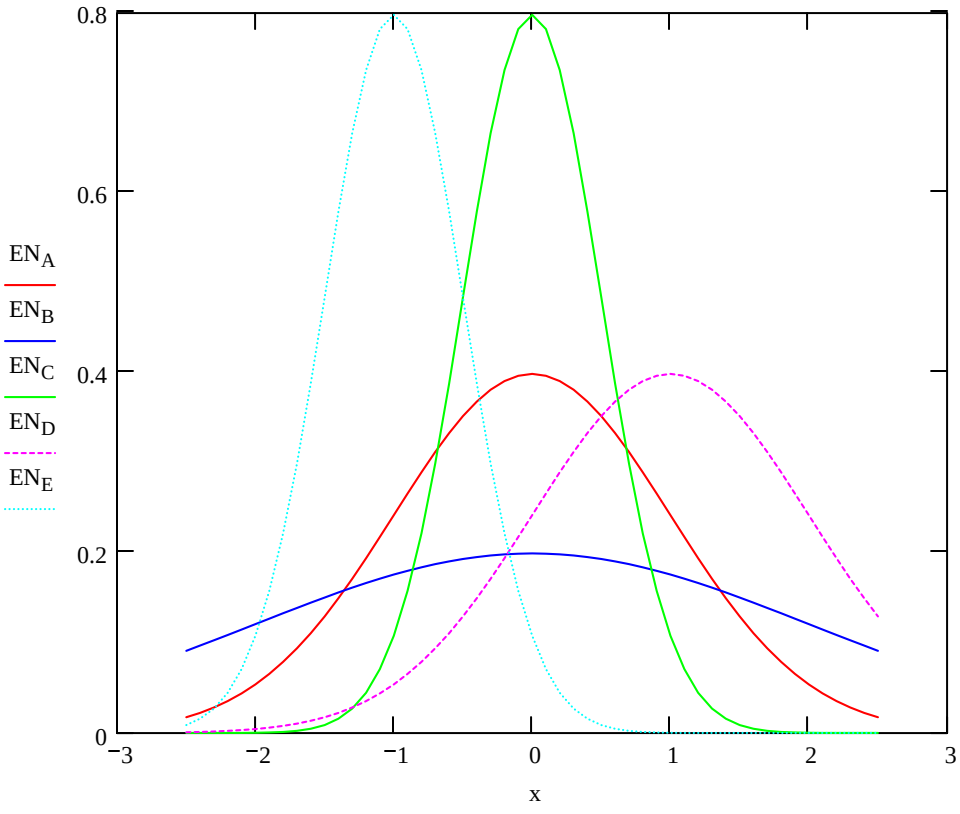
$EN_B := dnorm[x, \mu, (2\sigma_{sq})]$

$EN_C := dnorm[x, \mu, (0.5\sigma_{sq})]$

$EN_D := dnorm[x, (\mu + 1), \sigma_{sq}]$

$EN_E := dnorm[x, (\mu - 1), (0.5 \cdot \sigma_{sq})]$

<- The Normal distribution is family of curves defined by different values of μ and σ^2.



Example Continuous Probability Density functions:

Chi-Square (χ^2) Distribution:

This distribution is commonly encountered in statistics, especially in what is known as "Goodness of Fit" tests. We will work with it later, but it is interesting here to see that χ^2 density distributions exhibit a different shape.

$n := 50$ **<- Out of all possible values, we will arbitrarily look at a set of n point. At the scale we plot things here, this might as well be continuous...**

$i := 0 .. n$

$b := 1.4$ $c := 0.3$ **<- Arbitrary scaling factors so we can see things in the plot.**

$x_i := c \cdot (i + b)$ **<- Individual scaled values we plot on our x axis below.**

$d := 1$ **<- d is a parameter for the χ^2 distribution called "degrees of freedom". Thus χ^2 is also a family of curves.**

$EC_A := dchisq(x, d)$

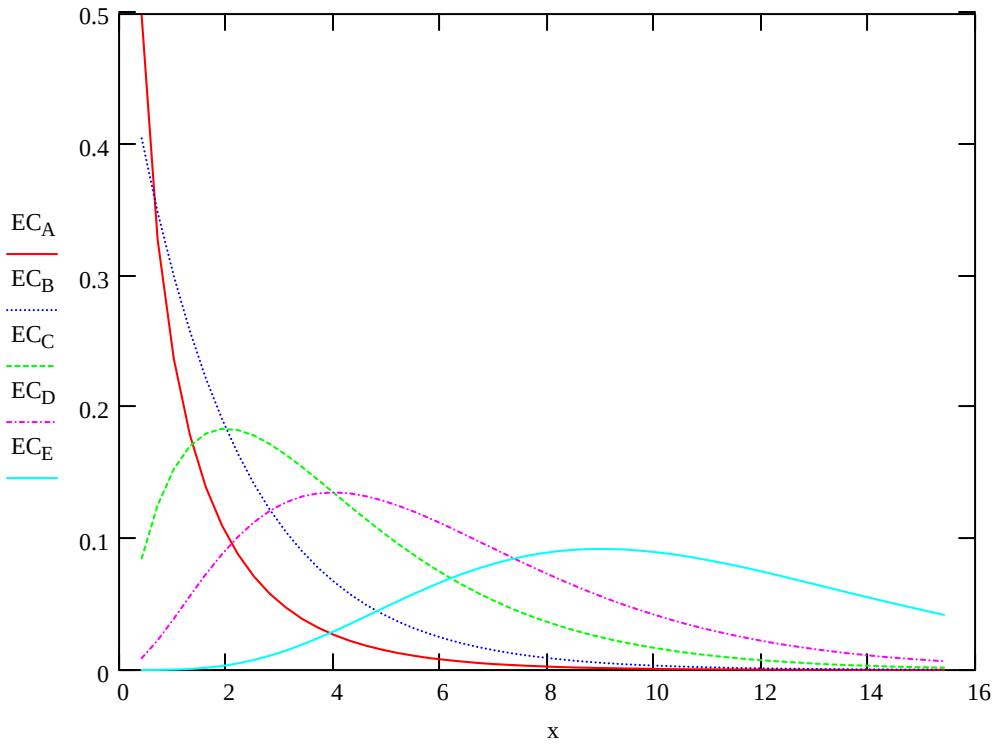
$EC_B := dchisq[x, (d + 1)]$

$EC_C := dchisq[x, (d + 3)]$

$EC_D := dchisq[x, (d + 5)]$

$EC_E := dchisq[x, (d + 10)]$

χ^2 family plotted below. As above, probability density the area under each curve.



Standard & Conditional Probability

ORIGIN := 0

Statistics is typically based on a pair of quantities:

x <- observed sample values
P(x) <- probability of the sampled values under some model of probability.

In fact, associating these two quantities is not at all straightforward and is often a point of controversy as both a theoretical and practical matter. There are two important perspectives:

- **Frequentist (or Standard) Statistical Methods - mostly what we will do in this course.**
- **Bayesian Inference - increasingly prominent in several biological & biomedical fields.**

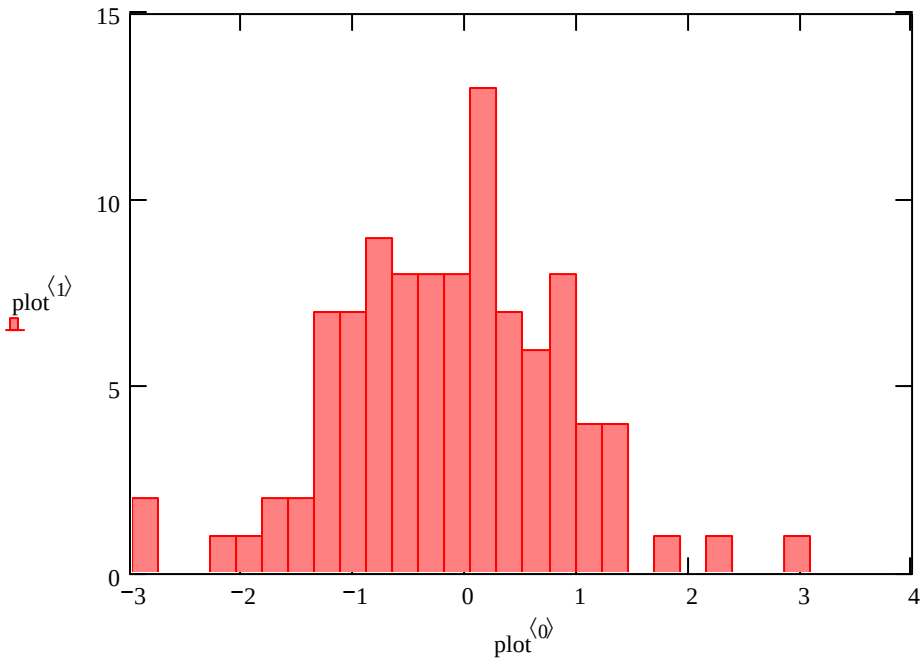
Frequentist Method:

" The probability of an event is the relative frequency of a set of outcomes over an indefinitely (or infinite) large number of trials." Rosner p. 44 Definition 3.1

Sometimes, for theoretical reasons, aspects of the probablity distributions are known or are assumed. More commonly in practice, however, one takes a reasonably large empirical sample and compares it with known theoretical distributions, such as the Normal Distribution.

x := rnorm(100, 0, 1) **<- For example drawing 100 values values from a Normal population distribution by a random number generator gives the following histogram...**

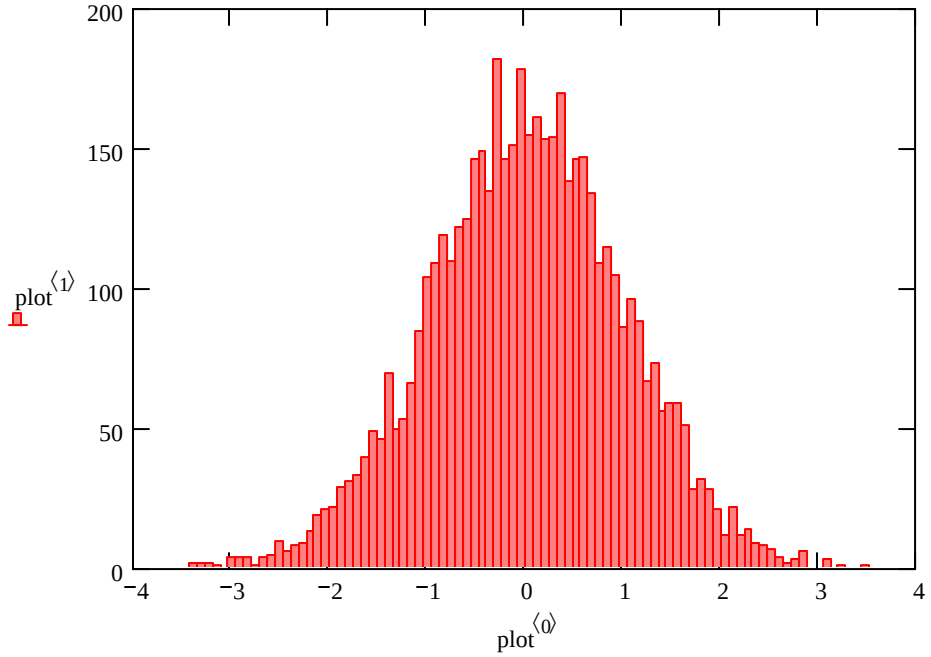
plot := histogram(30, x)



^ From this limited sample, one might conclude that the population from which it was drawn has a Normal distribution...

```
x := rnorm(5000, 0, 1)

plot := histogram(100, x)
```

<- But what if we draw a bigger sample, say 5000 values, and plot it with 100 bins instead of 30?



<- Conclusion: a bigger sample is usually better...
But other factors usually come into play including cost/time in conducting the study, and small scale bias of one sort or another.

Bayesian Inference:

Here two kinds of probability are distinguished:

"The prior probability of an event is the best guess by the observer of an event's probability in the absence of data. This prior probability may be a single number, or it may be a range of likely values for the probability, perhaps with weights attached to each possible value." Rosner p. 63, Definition 3.16.

"The posterior probability of an event is the probability of an event after collecting some empirical data. It is obtained by integrating information from the prior probability with additional data related to the event in question." Rosner p. 64, Definition 3.17.

We'll look at aspects of Bayesian Inference shortly...

The general logic of probability:

Under either of the above views, probability (both as a concept and a property) obeys fundamental logical (or mathematical) rules. These rules are very important to all aspects of statistical inference and in direct prediciton of outcomes.

Terminology:

sample space	**= the set of all possible outcomes**
an event	**= any specific set of outcomes**
P(x)	**= probability of event x, where** $0 \leq P(x) \leq 1$
compliment of x	**= (1-P(x)) = P(~x). Compliment is the probability of x not happening.**

Mutually exclusive events:

Two events, A & B are mutually exclusive if they can not both happen simultaneously.

Intersection of events is the empty set:

$P(A \wedge B) = \phi$ **<- for two events (the smallest number where intersection has a meaning)**

$P(A_1 \wedge A_2 \wedge ....A_i) = \phi$ **< for two or more events i**

Union of events - The Law of Addition of probabilities applies:

$P(A \vee B) = P(A) + P(B)$ **<- Probability of either A or B happening are their separate probabilities added together...**

$$P(A_1 \vee A_2 \vee A_3 \vee ....A_i) = (P(A_1) + P(A_2) + P(A_3) + ... + P(A_i))$$

^ for multiple exclusive events, add them all.

Potentially co-ocurring events:

Two events may occur simultaneously. There are two kinds:

1. Independent events:

The probilities of two events, i.e., P(A) & P(B) have no bearing on each other.

Intersection of events - the Law of Multiplied probabilities applies:

$P(A \wedge B) = P(A)\,P(B)$ **< multiply the separate probabilities to find the probability of both events occuring simultaneously.**

$$P(A_1 \wedge A_2 \wedge A_3 \wedge ....A_i) = P(A_1) \cdot P(A_2) \cdot P(A_3) .... P(A_i)$$

^ multiply all of them for multiple events

Union of events - Expanded Law of Addition applies::

$P(A \vee B) = P(A) + P(B) - P(A \wedge B)$

< The probability of A or B happening is the separate probabilities added together minus the probability that both A & B occur together

Alternate equivalent forms for Independent events only:

$P(A \vee B) = P(A) + P(B) \cdot (1 - P(A))$

$P(A \vee B) = P(B) + P(A) \cdot (1 - P(B))$

<- The probability of A or B happening is the probability of one plus the simultaneous occurrence of the other but not the first!

Note: the compliment here ^ = P(~A) or P(~B)

Check the Venn diagrams in the text to puzzle this out!

Note that Union for multiple independent events greater than two is not given...

2. Dependent events:

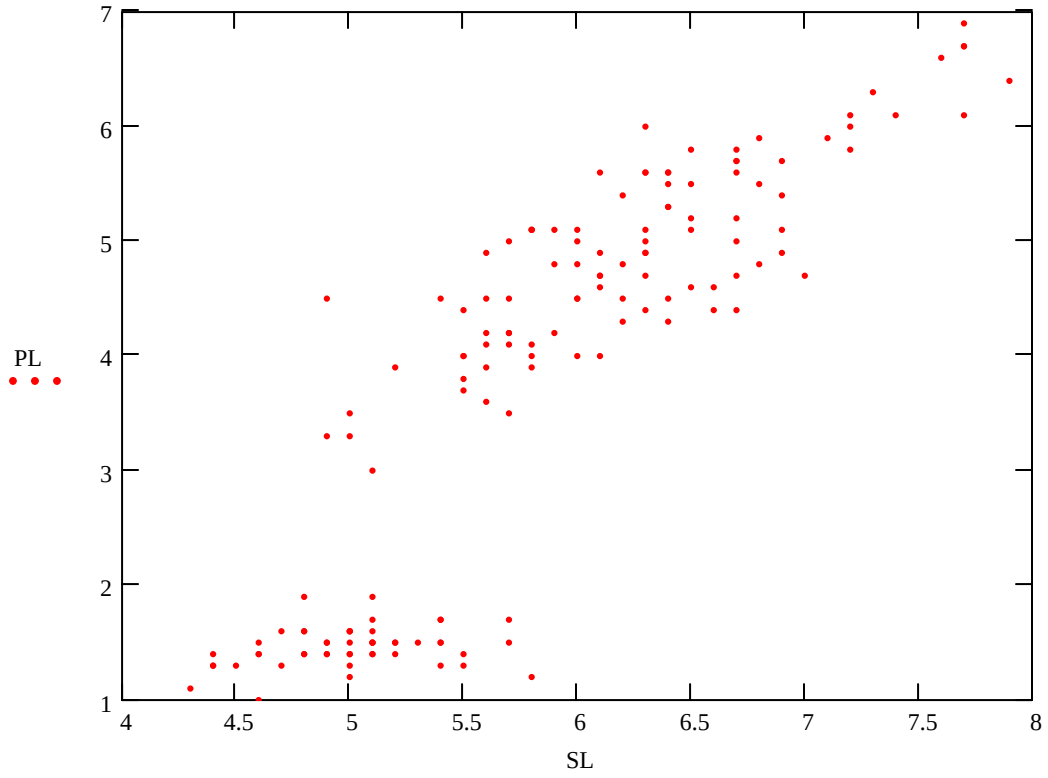
The probabilities of two events are related such that knowing the outcome of one event influences the probability of the other.

iris := READPRN("c:/2007BiostatsData/iris.txt")

SL := $iris^{\langle 1 \rangle}$ SW := $iris^{\langle 2 \rangle}$

PL := $iris^{\langle 3 \rangle}$ PW := $iris^{\langle 4 \rangle}$

<- Reading the Famous Iris data again...



^ A plot of Sepal Length (SL) and Petal Length (PL) shows dependence. Measuring one variable gives important information about the probable values of the other.

2. Dependent events:

Intersection of events - the Law of Multiplied probabilities fails:

$P(A \wedge B) \neq P(A)\,P(B)$ **<- This is a more formal statement of what dependence actually means. In practical terms, one often assesses the separate probabilities for A and B, and then compares their product with a separately estimated probability of both events occuring simultaneously to see if they match.**

To proceed at this point, one needs a concept of conditional probability...

$P(A \wedge B) = \text{"}P(B|A)\text{"} \cdot P(A)$

$P(B \wedge A) = \text{"}P(A|B)\text{"} \cdot P(B)$

<- intersection in terms of conditional probability. Note that you can switch the roles of A & B depending on which is prior probability = known versus posterior probability = unknown.

*** See below for more than two events!**

Conditional Probability:

The concept of conditional probability can be applied to both the independent and dependent cases of potentially simultaneous events above, so I'll give both here..

Definition of Conditional Probability:

Rearranging the Law of Multipied Probabilities to solve for one of the individual probabilities (i.e., P(B)), gives the definition for conditional probability:

$$P(B) = \frac{P(A \wedge B)}{P(A)}$$ **also written P(B|A) with no difference in meaning.**

^ This is the conditional probability for B given prior knowledge of A...

Calculating Intersection with Conditional Probability:

1. Independent case:

P(B|A) = P(B) = P(~A)

$P(A \wedge B) = P(A) \cdot P(B)$ **< Equalities here makes the Law of Multiplied probabilities a special case of the more general one below...**

2. More general Dependent case:

$$P(A \wedge B) = P\left(\frac{B}{A}\right) \cdot P(A)$$ **<- For two events....**

$$P(A_1 \wedge A_2 \wedge A_3) = P(A_1) \cdot P\left[\frac{A_2}{(A_1)}\right] \cdot P\left[\frac{A_3}{(A_2 \wedge A_1)}\right]$$ **<- For three events...**

$$P(A_1 \wedge A_2 \wedge A_3 \wedge ....A_i) = P(A_1) \cdot P\left[\frac{A_2}{(A_1)}\right] \cdot P\left[\frac{A_3}{(A_2 \wedge A_1)}\right] \cdot .. \cdot P\left(\frac{A_i}{A(i-1) \wedge A_3 \wedge A_2 \wedge A_1}\right)$$

^ For more than three events...

Calculating Total Probability from Conditional Probability:

This formulation is commonn to both the Independent and Dependent cases:

$P(A) = P(A|B)*P(B) + P(A|\sim B)*P(\sim B)$ **<- For two possibilities A & B... Note that the roles of A & B are interchangeable**

$P(B) = P(B|A)*P(A) + P(B|\sim A)*P(\sim A)$

$P(A) = \Sigma_i P(A|B_i)*P(B_i)$ **<- For A given multiple prior probabilities B_i**

1. Independent case:

The formulas simply reduces to multiplying P(B) or $P(B_i)$ depending on number of B's

2. Dependent case:

The formula doesn't reduce and conditional probabilties must be used as stated above.

Bayes' Rule:

The point of this procedure for two events A & B is to estimate one conditional probability P(B|A) from the other conditional probability P(A|B) and one total probablity P(B).

$$P\left(\frac{B}{A}\right) = \frac{P\left(\frac{A}{B}\right) \cdot P(B)}{P\left(\frac{A}{B}\right) \cdot P(B) + P\left(\frac{A}{notB}\right) \cdot P(notB)}$$

<- Of course, as above, the defined roles of A & B here can be reversed.

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

<- General form of Bayes' Rule giving multiple conditional probabilities for the B's given knowledge of multiple conditional probabilites $P(A|B_i)$ and multiple total probabilites $P(B_i)$.

In clinical situations, A represents symptoms or results of a test, and the B's represent patient condition(s) such as a disease. Known conditional probabilites $P(A|B_i)$ can be estimated from the portion of patients with a known condition(s) B_i showing positive test results. Total probablility $P(B_i)$ for the condition(s) can be estimated from the population at large. Bayes' Rule allows the researcher to estimate the conditional probability that the symptoms or test results indicate any particular condition or disease. Powerful stuff!

Clinical Terminology often used with Bayes' Rule:

$P(B|A) / P(B|\sim A)$	**Relative Risk**
$P(B_i|A)$	**Predictive value positive of the test (PV^+)**
$P(\sim B_i|\sim A)$	**Predictive value negative of the test (PV^-)**
$P(A|B_i)$	***Sensitivity*** **of the symptoms or test**
$P(\sim A|\sim B_i)$	***Specificity*** **of the symptoms or test**
$(\sim A|B_i)$	**false negative for the symptoms or test**
$(A|\sim B_i)$	**false positive for the symptoms or test**

Examples using Bayes' Rule:

ORIGIN := 1

Bayesian Inference:

Here two kinds of probability are distinguished:

"The prior probability of an event is the best guess by the observer of an event's probability in the absence of data. This prior probability may be a single number, or it may be a range of likely values for the probability, perhaps with weights attached to each possible value." Rosner p. 63, Definition 3.16.

"The posterior probability of an event is the probability of an event after collecting some empirical data. It is obtained by integrating information from the prior probability with additional data related to the event in question." Rosner p. 64, Definition 3.17.

Bayes' Rule:

The point of this procedure for two events A & B is to estimate one conditional probability P(B|A) from the other conditional probability P(A|B) and one total probablity P(B).

$$P\left(\frac{B}{A}\right) = \frac{P\left(\frac{A}{B}\right) \cdot P(B)}{P\left(\frac{A}{B}\right) \cdot P(B) + P\left(\frac{A}{notB}\right) \cdot P(notB)}$$

<- Of course, as above, the defined roles of A & B here can be reversed.

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

<- General form of Bayes' Rule giving multiple conditional probabilities for the B's given knowledge of multiple conditional probabilites $P(A|B_i)$ and multiple total probabilites $P(B_i)$.

In clinical situations, A represents symptoms or results of a test, and the B's represent patient condition(s) such as a disease. Known conditional probabilites $P(A|B_i)$ can be estimated from the portion of patients with a known condition(s) B_i showing positive test results. Total probablility $P(B_i)$ for the condition(s) can be estimated from the population at large. Bayes' Rule allows the researcher to estimate the conditional probability that the symptoms or test results indicate any particular condition or disease. Powerful stuff!

Clinical Terminology often used with Bayes' Rule:

$P(B\|A) / P(B\|\sim A)$	**Relative Risk**
$P(B_i\|A)$	**Predictive value positive of the test (PV^+)**
$P(\sim B_i\|\sim A)$	**Predictive value negative of the test (PV^-)**
$P(A\|B_i)$	***Sensitivity*** **of the symptoms or test**
$P(\sim A\|\sim B_i)$	***Specificity*** **of the symptoms or test**
$(\sim A\|B_i)$	**false negative for the symptoms or test**
$(A\|\sim B_i)$	**false positive for the symptoms or test**

Example 3.26 Rosner p. 61:

HYPERTENSION

The Data in Matrix Form:

$$M := \begin{pmatrix} 1 & 1 & .84 \\ 1 & 0 & .23 \\ 0 & 1 & 1 - .84 \\ 0 & 0 & 1 - .23 \end{pmatrix} \qquad M = \begin{pmatrix} 1 & 1 & 0.84 \\ 1 & 0 & 0.23 \\ 0 & 1 & 0.16 \\ 0 & 0 & 0.77 \end{pmatrix}$$

A B #

Terminology:

< Sensitivity: P(A|B)

< (1- Specificity)

< (1 - Sensitivity)

< Specificity P(~A|~B$_i$)

Given unconditional (prior) probabilities:

$P_B := .2 \qquad P_B = 0.2$ **< Probability that an adult in the population generally is hypertensive**

$P_{nB} := 1 - P_B \qquad P_{nB} = 0.8$ **< Probability NOT hypertensive (1-P$_B$)**

Conditional probabilities:

$CP_{AB} := .84$ **< Sensitivity: P(A|B) - Probability that hypertensives (B) are classed hypertensive by the machine (A)**

$CP_{AnB} := .23$ **< (1 - Specificity): P(A|~B) - Probability that NON hypertensives are classed hypertensive by the machine**

Bayes' Rule:

$$CP_{BA} := \frac{CP_{AB} \cdot P_B}{CP_{AB} \cdot P_B + CP_{AnB} \cdot P_{nB}} \qquad CP_{BA} = 0.477$$

^ Note that this corresponds to calculating PV$^+$ above and in the text.

The conditional (posterior) probability that the machine properly classes hypertensives as hypertensives is 0.477

Bayes' Rule using the above terminology:

$sensitivity := .84$

$specificity := .77$

$$CV_{plus} := \frac{sensitivity \cdot P_B}{sensitivity \cdot P_B + (1 - specificity) \cdot (1 - P_B)} \qquad CV_{plus} = 0.477$$

^ Different variable names, same result...

Bayes' Rule for Predictive Value Negative (PV$^-$):

$$CV_{minus} := \frac{specificity \cdot (1 - P_B)}{specificity \cdot (1 - P_B) + (1 - sensitivity) \cdot P_B} \qquad CV_{minus} = 0.951$$

^ It is important to note that PV$^-$ serves to ask the same question as PV$^+$ except in the opposite sense for the meaning of condition B. The 0's or 1's can be reversed above, or the interpretation of PV$^+$ vs PV$^-$ reversed, giving the same result.

So if: **Terminology:**

$$M := \begin{pmatrix} 0 & 0 & .84 \\ 0 & 1 & .23 \\ 1 & 0 & 1 - .84 \\ 1 & 1 & 1 - .23 \end{pmatrix} \qquad M = \begin{pmatrix} 0 & 0 & 0.84 \\ 0 & 1 & 0.23 \\ 1 & 0 & 0.16 \\ 1 & 1 & 0.77 \end{pmatrix}$$

< Specificity P(~A|~B$_i$)
< (1 - Sensitivity)
< (1- Specificity)
< Sensitivity: P(A|B)

$sensitivity := .77$
$specificity := .84$ **< Meanings are now turned around.**

$P_B := 0.8$ **< This is turned around also...**

$$CV_{plus} := \frac{sensitivity \cdot P_B}{sensitivity \cdot P_B + (1 - specificity) \cdot (1 - P_B)} \qquad CV_{plus} = 0.951$$

^ Same result as for PV$^-$ above now that everything is turned around.

Applying Bayes' Rule in its general form:

Bayes' Rule in general form as above:

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

For this problem i = 2

Unconditional (prior) probabilities:

$P_{B1} := .2$ **<- P(B$_1$): probability of hypertensives**

$P_{B2} := .8$ **<- P(B$_2$): probability of NOT hypertensives**

Conditional probabilities:

$P_{AB1} := .84$ **<- P(A|B$_1$): Probability of positive test for hypertensives**

$P_{AB2} := 0.23$ **<- P(A|B$_2$): Probability of positive test for NON hypertensives**

Applying Bayes' Rule in its general form:

For P(B$_1$|A):

$$PB1A := \frac{P_{AB1} \cdot P_{B1}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2}} \qquad PB1A = 0.477$$

^ Same result as the first PV$^+$ test above.

For P(B$_2$|A):

$$PB2A := \frac{P_{AB2} \cdot P_{B2}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2}} \qquad PB2A = 0.523$$

^ Note that this is NOT the same probability as for PV$^-$ above as the conditional probability is dependent on A here (whereas above it was ~A)!

Example 3.27 Rosner p. 62:

Bayes' Rule in general form as above:

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

Unconditional (prior) probabilities:	**Conditional probabilities:**
$P_{B1} := .99$	$P_{AB1} := .001$
$P_{B2} := .001$	$P_{AB2} := .9$
$P_{B3} := .009$	$P_{AB3} := .9$

$$P_{B1A} := \frac{P_{AB1} \cdot P_{B1}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B1A} = 0.099 \qquad < \mathbf{P(B_1|A)}$$

$$P_{B2A} := \frac{P_{AB2} \cdot P_{B2}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B2A} = 0.09 \qquad < \mathbf{P(B_2|A)}$$

$$P_{B3A} := \frac{P_{AB3} \cdot P_{B3}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B3A} = 0.811 \qquad < \mathbf{P(B_3|A)}$$

Example 3.28 Rosner p. 63:

Unconditional (prior) probabilities:	**Conditional probabilities:**
$P_{B1} := .98$	$P_{AB1} := .001$
$P_{B2} := .015$	$P_{AB2} := .9$
$P_{B3} := .005$	$P_{AB3} := .9$

$$P_{B1A} := \frac{P_{AB1} \cdot P_{B1}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B1A} = 0.052 \qquad < \mathbf{P(B_1|A)}$$

$$P_{B2A} := \frac{P_{AB2} \cdot P_{B2}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B2A} = 0.711 \qquad < \mathbf{P(B_2|A)}$$

$$P_{B3A} := \frac{P_{AB3} \cdot P_{B3}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2} + P_{AB3} \cdot P_{B3}} \qquad P_{B3A} = 0.237 \qquad < \mathbf{P(B_3|A)}$$

Determining Risk for Families using Pedigree Analysis:

ORIGIN := 1

Bayes' Rule:

The point of this procedure for two events A & B is to estimate one conditional probability P(B|A) from the other conditional probability P(A|B) and one total probablity P(B).

$$P\left(\frac{B}{A}\right) = \frac{P\left(\frac{A}{B}\right) \cdot P(B)}{P\left(\frac{A}{B}\right) \cdot P(B) + P\left(\frac{A}{notB}\right) \cdot P(notB)}$$

<- Of course, as above, the defined roles of A & B here can be reversed.

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

<- General form of Bayes' Rule giving multiple conditional probabilities for the B's given knowledge of multiple conditional probabilites $P(A|B_i)$ and multiple total probabilites $P(B_i)$.

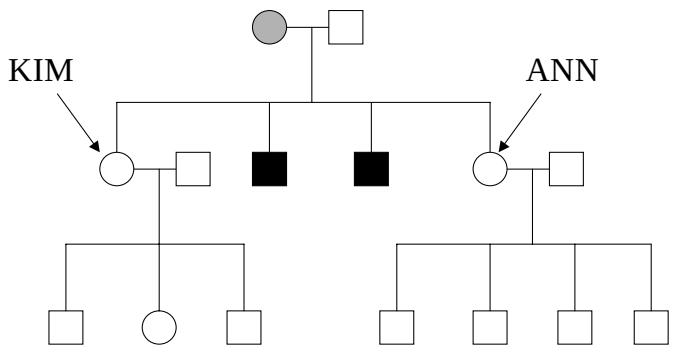
In clinical situations, A represents symptoms or results of a test, and the B's represent patient condition(s) such as a disease. Known conditional probabilites $P(A|B_i)$ can be estimated from the portion of patients with a known condition(s) B_i showing positive test results. Total probablility $P(B_i)$ for the condition(s) can be estimated from the population at large. Bayes' Rule allows the researcher to estimate the conditional probability that the symptoms or test results indicate any particular condition or disease. Powerful stuff!

Pedigree Analysis:

In genetic counseling, potential parents in a family with history of a genetic disease often ask about the risk they face in deciding whether to have additional children or not. Use of Bayes' Rule (also called Bayes theorem) is standard practice in providing them with this information.

Example:

Two sisters, Kim and Ann, are in a family with a history of Hemophilia A as shown in the following pedigree. Hemophilia A is a sex-linked recessive trait (gene located on the X chromosome). Of course, given their family history, each woman wants to know her risk of being a carrier for this genetic trait.



Because brothers of both women exhibit the trait, they must have received it from their mother (sex-linked). Since their mother doesn't exhibit symptoms she must be a carrier - that is, one of her X chromosomes carries the allele for Hemophilia A but it is masked by a normal allele on the other chromosome. So mother is indicated as a carrier by gray on the pedigree above.

From simple Mendelian inheritance, we know that both Kim and Ann have a 50% chance of receiving the Hemophilia A allele from their mother. We call this their common or unconditional probability of being a carrier for the trait.

However, each woman has already had children whose traits we can assess, so we know something more that is specific for each. We call this their conditional probability of being a carrier given knowledge about their children.

So we have all the information we need to perform a Bayesian analysis.

Using Bayes' Rule:

KIM

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

unconditional probability ("prior"):

$P_{B1} := .5$ **< probability she's a carrier**

$P_{B2} := .5$ **< probability she's not a carrier**

conditional probability

$P_{AB1} := .25$ **< probability her sons are normal given that she's a carrier (her condition is B_1) and Sons are event A (a test)**

$P_{AB2} := 1.0$ **< probability her sons are normal given she's NOT a carrier - her condition is B_2**

$$PB1A := \frac{P_{AB1} \cdot P_{B1}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2}} \qquad PB1A = 0.2 \qquad \textbf{< } P(B_1|A)$$

ANNA

$$P\left(\frac{B_i}{A}\right) = \frac{P\left(\frac{A}{B_i}\right) \cdot P(B_i)}{\sum_i P\left(\frac{A}{B_i}\right) \cdot P(B_i)}$$

unconditional probability ("prior"):

$P_{B1} := .5$ **< probability she's a carrier**

$P_{B2} := .5$ **< probability she's not a carrier**

conditional probability

$P_{AB1} := .5^4$ **< probability her sons are normal given that she's a carrier (her condition is B_1) and Sons are event A (a test)**

$P_{AB2} := 1.0$ **< probability her sons are normal given she's NOT a carrier - her condition is B_2**

$$PB1A := \frac{P_{AB1} \cdot P_{B1}}{P_{AB1} \cdot P_{B1} + P_{AB2} \cdot P_{B2}} \qquad PB1A = 0.058824 \qquad \textbf{< } P(B_1|A)$$

So even given their common genetic history from their mother, knowledge about the children each woman has borne substantially modifies our interpretion of her risk of being a carrier!

Bayesian Analysis in Tabular Form:

The above analyses have been set up in exactly the same way as our other examples. From what I understand, geneticists often present their analysis in a slightly different tabular form. Same results, but it looks a little different:

Hemophilia A KIM:

Probability:	**Kim is a carrier**	**Kim is NOT a carrier**		***Compare with above:***	
prior:	**0.5**	**0.5**	<-	P_{B1}	P_{B2}
conditional: (two normal sons)	**0.25 =** 0.5^2	**1.0**	<-	P_{AB1}	P_{AB2}
joint:	**0.125**	**0.5**	<-	$P_{AB1} \cdot P_{B1}$	$P_{AB2} \cdot P_{B2}$
posterior:	**0.125/(0.125+0.5) = 0.2 or 20%**		<-	PB1A	

For Hemophilia A: ANN

Probability:	**Ann is a carrier**	**Ann is NOT a carrier**		***Compare with above:***	
prior:	**0.5**	**0.5**	<-	P_{B1}	P_{B2}
conditional: (four normal sons)	**0.0625** = 0.5^4	**1.0**	<-	P_{AB1}	P_{AB2}
joint:	**0.03125**	**0.5**	<-	$P_{AB1} \cdot P_{B1}$	$P_{AB2} \cdot P_{B2}$
posterior:	**0.03125/(0.03125+0.5) = 0.058824 or 5.8%**		<-	PB1A	

Receiver Operating Characteristic (ROC) Curves:

ORIGIN := 1

In previous examples, we treated the tests (column A) as strictly binary, that is, + versus -, 0 versus 1, or "yes" versus "no". In real life, of course the results of a test may involve a "grey area" such as numerical results in which a cut-off for "test-positive" versus "test-negative" must be established.

Example 3.26 Rosner p. 61:

HYPERTENSION

The Data in Matrix Form:

$$M := \begin{pmatrix} 1 & 1 & .84 \\ 1 & 0 & .23 \\ 0 & 1 & 1-.84 \\ 0 & 0 & 1-.23 \end{pmatrix} \qquad M = \begin{pmatrix} 1 & 1 & 0.84 \\ 1 & 0 & 0.23 \\ 0 & 1 & 0.16 \\ 0 & 0 & 0.77 \end{pmatrix}$$

A B #

Terminology:

< Sensitivity: P(A|B)

< (1- Specificity)

< (1 - Sensitivity)

< Specificity P(~A|~B$_i$)

^ "grey area" in deciding 0 vs 1 in column A???

Example 3.32-3.34 Rosner p.64-66:

RADIOLOGY

Roc curves are a graphic display of the peformance of a test given that the test allows different criteria for deciding "test-positive" versus "test-negative". In this example, five different dividing points between "test-positive" versus "test-negative" were proposed. For each criterion, sensitivity and specificity (as defined above) were determined:

Table 3.3 p. 65:

true status	**-**	**test**	**+**			
	1	2	3	4	5	6 **< criteria (1-6)**
- Normal >	33	6	6	11	2	58
+ Abnormal >	3	2	2	11	33	51
	36	8	8	22	35	109 **< Column Totals**

^ Row Totals

Interpreting the results of the test under the different criteria:

Criterion (1): "definitely normal"

$$one_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 51 & 0 & 51 \\ 58 & 0 & 58 \\ 109 & 0 & 109 \end{pmatrix} \end{array} \begin{matrix} \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

$$M := \begin{pmatrix} 1 & 1 & \frac{51}{51} \\ 1 & 0 & \frac{58}{58} \\ 0 & 1 & \frac{0}{51} \\ 0 & 0 & \frac{0}{58} \end{pmatrix} \qquad M = \begin{pmatrix} 1 & 1 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

$$sensitivity_1 := 1.0$$

$$specificity_1 := 0.0$$

Criterion (2): "probably normal"

$$\text{two}_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 48 & 3 & 51 \\ 25 & 33 & 58 \\ 73 & 36 & 109 \end{pmatrix} \end{array} \begin{matrix} \\ \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

$$M := \begin{pmatrix} 1 & 1 & \frac{48}{51} \\ 1 & 0 & \frac{25}{58} \\ 0 & 1 & \frac{3}{51} \\ 0 & 0 & \frac{33}{58} \end{pmatrix} \quad M = \begin{pmatrix} 1 & 1 & 0.941 \\ 1 & 0 & 0.431 \\ 0 & 1 & 0.059 \\ 0 & 0 & 0.569 \end{pmatrix}$$

$$\text{sensitivity}_2 := 0.941 \qquad \text{specificity}_2 := 0.569$$

Criterion (3): "questionable"

$$\text{three}_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 46 & 5 & 51 \\ 19 & 39 & 58 \\ 65 & 44 & 109 \end{pmatrix} \end{array} \begin{matrix} \\ \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

$$M := \begin{pmatrix} 1 & 1 & \frac{46}{51} \\ 1 & 0 & \frac{19}{58} \\ 0 & 1 & \frac{5}{51} \\ 0 & 0 & \frac{39}{58} \end{pmatrix} \quad M = \begin{pmatrix} 1 & 1 & 0.902 \\ 1 & 0 & 0.328 \\ 0 & 1 & 0.098 \\ 0 & 0 & 0.672 \end{pmatrix}$$

$$\text{sensitivity}_3 := 0.902 \qquad \text{specificity}_3 := 0.672$$

Criterion (4): "probably abnormal"

$$\text{four}_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 44 & 7 & 51 \\ 13 & 45 & 58 \\ 57 & 52 & 109 \end{pmatrix} \end{array} \begin{matrix} \\ \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

$$M := \begin{pmatrix} 1 & 1 & \frac{44}{51} \\ 1 & 0 & \frac{13}{58} \\ 0 & 1 & \frac{7}{51} \\ 0 & 0 & \frac{45}{58} \end{pmatrix} \quad M = \begin{pmatrix} 1 & 1 & 0.863 \\ 1 & 0 & 0.224 \\ 0 & 1 & 0.137 \\ 0 & 0 & 0.776 \end{pmatrix}$$

$$\text{sensitivity}_4 := 0.863 \qquad \text{specificity}_4 := 0.776$$

Criterion (5): "definitely abnormal"

$$\text{five}_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 33 & 18 & 51 \\ 2 & 56 & 58 \\ 35 & 74 & 109 \end{pmatrix} \end{array} \begin{matrix} \\ \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

$$M := \begin{pmatrix} 1 & 1 & \frac{33}{51} \\ 1 & 0 & \frac{2}{58} \\ 0 & 1 & \frac{18}{51} \\ 0 & 0 & \frac{56}{58} \end{pmatrix} \quad M = \begin{pmatrix} 1 & 1 & 0.647 \\ 1 & 0 & 0.034 \\ 0 & 1 & 0.353 \\ 0 & 0 & 0.966 \end{pmatrix}$$

$$\text{sensitivity}_5 := 0.647 \qquad \text{specificity}_5 := 0.966$$

Criterion (6): "everyone abnormal"

$$\text{six}_{plus} := \begin{array}{c} \begin{matrix} \textbf{test} & & \\ \textbf{+} & \textbf{-} & \textbf{totals} \end{matrix} \\ \begin{pmatrix} 0 & 51 & 51 \\ 0 & 58 & 58 \\ 0 & 109 & 109 \end{pmatrix} \end{array} \begin{matrix} \\ \textbf{actual} \\ \textbf{+} \\ \textbf{-} \\ \textbf{totals} \end{matrix}$$

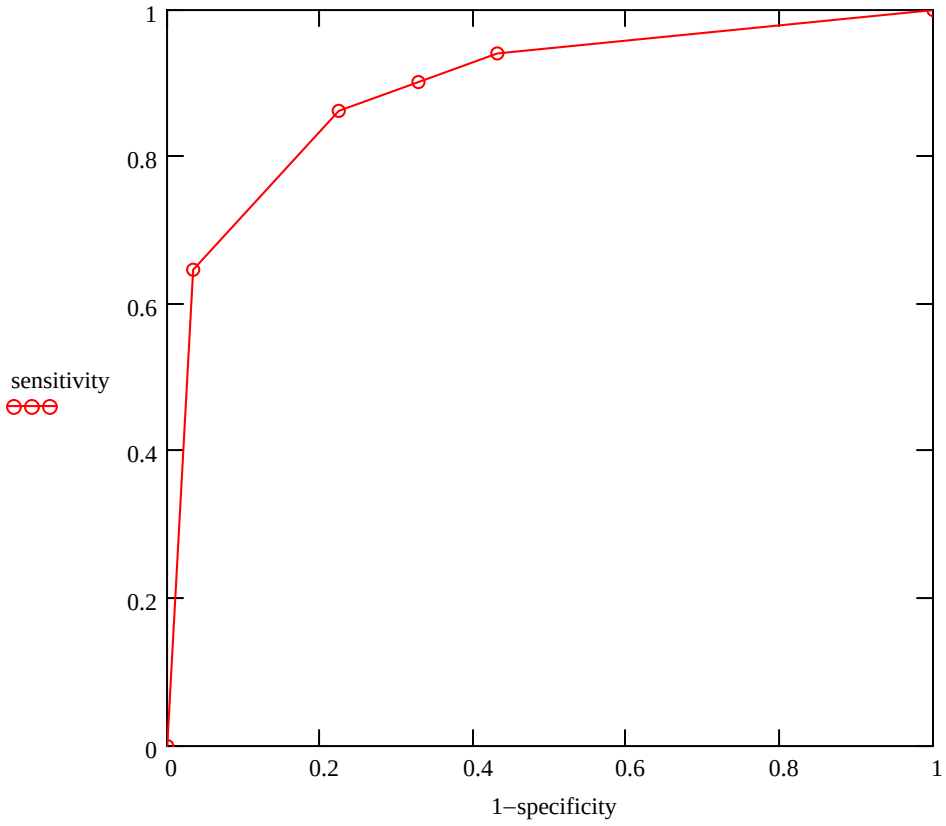
$$M := \begin{pmatrix} 1 & 1 & \frac{0}{51} \\ 1 & 0 & \frac{0}{58} \\ 0 & 1 & \frac{51}{51} \\ 0 & 0 & \frac{58}{58} \end{pmatrix} \quad M = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\text{sensitivity}_6 := 0.0 \qquad \text{specificity}_6 := 1.0$$

Collecting Sensitivity & Specificity:

$$\text{sensitivity} = \begin{pmatrix} 1 \\ 0.941 \\ 0.902 \\ 0.863 \\ 0.647 \\ 0 \end{pmatrix} \quad \text{specificity} = \begin{pmatrix} 0 \\ 0.569 \\ 0.672 \\ 0.776 \\ 0.966 \\ 1 \end{pmatrix} \quad 1 - \text{specificity} = \begin{pmatrix} 1 \\ 0.431 \\ 0.328 \\ 0.224 \\ 0.034 \\ 0 \end{pmatrix}$$

ROC plot:



In comparing different test methods, the area under this curve may be estimated using the Trapezoidal Method or compared visually... The greater the area under the curve the better.

For references on how to employ the Trapezoidal Method for determining areas under curves, search Google or see:

http://metric.ma.ic.ac.uk/integration/techniques/definite/numerical-methods/trapezoidal-rule/

http://www.geocities.com/rsrirang2001/Mathematics/NumericalMethods/trape/trape.htm

http://www.kent.k12.wa.us/staff/DavidWright/calculus/book/46/index.html

Working with the Binomial Probability Distribution

ORIGIN ≡ 0

The Binomial probability distribution, also called 'Binomial probability-mass' function is a commonly employed theoretical distribution for data taking on discrete values. It is derived from considerations of permutations and combinations:

Permutations: "The number of permutations of n things taken k at a time ... represents the number of ways of selecting k items out of n *where the order of selection is important*." Rosner Definition 4.8, p. 91.

$n := 3$ **< n things...**

$k := 2$ **< taken k at a time**

$$nP_k := \frac{n!}{(n-k)!} \qquad nP_k = 6$$

meaning of factorials (!)

$n! = 6 \qquad 3 \cdot 2 \cdot 1 = 6$

$k! = 2 \qquad 2 \cdot 1 = 2$

$(n-k)! = 1$

^ number of Permutations of n things take k at at time

For example, let the n things be the letters: A, B, C. How many pairs of letters can we make where the order of letters is important?

AB	AC	BC
BA	CA	CB

< fortunately the number n is relatively small, six!

What happens if:

$n := 20$ **< n things...**

$k := 7$ **< taken k at a time**

$$nP_k := \frac{n!}{(n-k)!} \qquad nP_k = 3.907 \times 10^8$$

$n! = 2.4329 \times 10^{18}$

$k! = 5040$

$(n-k)! = 6.227 \times 10^9$

^ number of Permutations of n things take k at at time

Fortunately we have this formula, because listing all of the possilities and counting them up would take some time...

Combinations: "The number of combinations of n things taken k at a time ... represents the number of ways of selecting k objects out of n *where the order of selection does NOT matter*." Rosner Definition 4.11, p. 93.

For example with same n & k as the first one above:

$n := 3$ **< n things...**

$k := 2$ **< taken k at a time**

$$nC_k := \frac{n!}{k! \cdot (n-k)!} \qquad nC_k = 3$$

$n! = 6$

$k! = 2$

$(n-k)! = 1$

^ number of Combination of n things take k at at time

Note that this is half the number of Permutations with n=3, k=2.

What about the larger example above?

$n := 20$ **< n things...**

$k := 7$ **< taken k at a time**

$$nC_k := \frac{n!}{k! \cdot (n-k)!} \qquad nC_k = 77520$$

$n! = 2.4329 \times 10^{18}$

$k! = 5040$

$(n-k)! = 6.227 \times 10^{9}$

^ number of Combination of n things take k at at time

$$\frac{nP_k}{nC_k} = 5040$$ **< a somewhat larger difference here!**

Most software packages contain built-in functions for Permutations and Combinations:

$n := 20$

$k := 7$

$permut(n,k) = 3.907 \times 10^{8}$ $nP_k = 3.907 \times 10^{8}$ **< and match our calculations above so serve as prototype...**

$combin(n,k) = 77520$ $(nC_k) = 77520$

Important symmetry in calculation of Combinations:

$combin(n,k) = 77520$ $combin[n,(n-k)] = 77520$ **< k or (n-k) give the same result for combination but NOT permutation.**

$permut(n,k) = 3.907 \times 10^{8}$ $permut[n,(n-k)] = 4.8272 \times 10^{14}$

The Binomial Distribution:

Statistics is typically based on a pair of quantities (Note greater precision here in statement):

X <- A "random variable" some of whose values may be observed in a dataset.
P(X) <- probability of all values of X under some model of probability.

The binomial distribution is an example of a probability function linking specific values X with a probability P(X) where X takes on only discrete values, such as 1,2,3, ...

"The distribution of the number of successes in statistically independent trials, where the probability of success on each trial is p, is known as the binomial distribution..." Rosner Equation 4.5 p. 96.

n & k take on the same meaning as above for Combinations:

$n := 20$ **< total number of things - usually called "trials" in this context.**

$k := 7$ **< our X above = number of "successes" - where "success" versus "failure" take on two arbitrary states such as "heads" vs "tails", or "present" vs "absent" etc.**

one additional consideration:

$p := 0.5$ **< the probability of "success" for any one time. In a coin flip, p = 0.5, but this is one of several possibilities one might want to investigate for instance if the coin were thought to be 'not fair'...**

$q := 1 - p$ **< probability p for "success" implies probability q for "failure"...**

So, having specified a value for the random variable X as k:

We employ the binomial probability function - let's call it $P_B(X = k)$:

$$P_B := \text{combin}(n,k) \cdot p^k \cdot q^{n-k} \qquad P_B = 0.0739$$

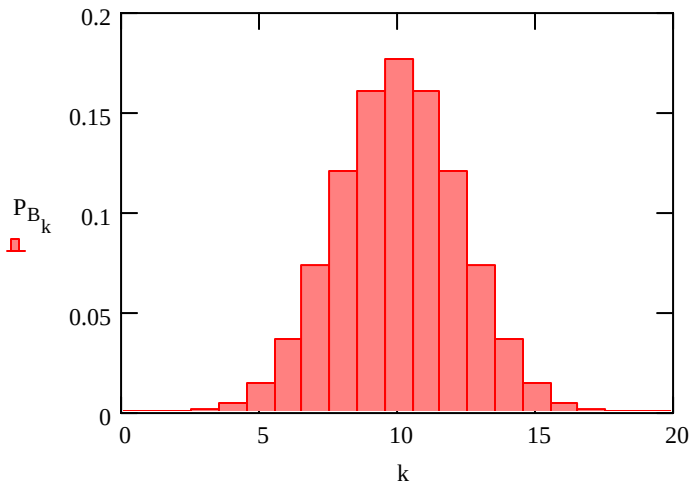
^ This is the probability that k "successes" will be found in n trials.

We can look at $P_B(X)$ for other values of k if we like. Since n = 20 is not too large, let's look at all values of {k = 0.1.2.3 ... 20} here:

$$k := 0 .. 20$$

$$P_{B_k} := \text{combin}(n,k) \cdot p^k \cdot q^{n-k}$$

And Plot P_B:



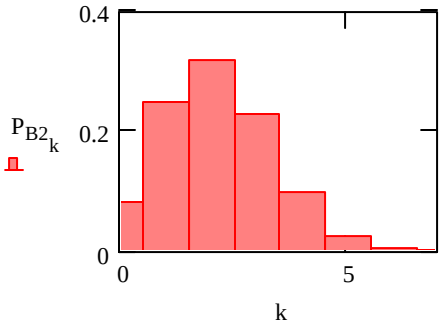
$$P_B = \begin{pmatrix} 9.5367 \times 10^{-7} \\ 1.9073 \times 10^{-5} \\ 0.0002 \\ 0.0011 \\ 0.0046 \\ 0.0148 \\ 0.037 \\ 0.0739 \\ 0.1201 \\ 0.1602 \\ 0.1762 \\ 0.1602 \\ 0.1201 \\ 0.0739 \\ 0.037 \\ 0.0148 \\ 0.0046 \\ 0.0011 \\ 0.0002 \\ 1.9073 \times 10^{-5} \\ 9.5367 \times 10^{-7} \end{pmatrix}$$

^ Remember here, k represents the values that random variable X can have, and P_B represents the associated probability P(X=k). This is the Binomial "probability distribution" or "probability-mass function" named above.

Of course, different values of n, k & p give different results:

$$n := 7 \quad k := 0 .. n \quad p := .3 \quad q := 1 - p$$

$$P_{B2_k} := \text{combin}(n,k) \cdot p^k \cdot q^{n-k}$$



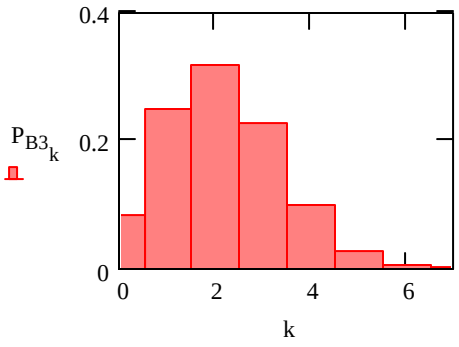
$$P_{B2} = \begin{pmatrix} 0.0824 \\ 0.2471 \\ 0.3177 \\ 0.2269 \\ 0.0972 \\ 0.025 \\ 0.0036 \\ 0.0002 \end{pmatrix}$$

Software calculation of the "exact" Binomial Distribution:

Most computer packages have built-in functions for calculating the Binomial distribution:

$n := 7 \quad k := 0..n \quad p := .3 \quad q := 1 - p$

$P_{B3_k} := \text{dbinom}(k, n, p)$



$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{pmatrix}$ **X**

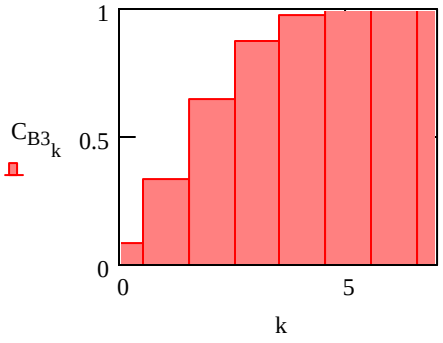
$P_{B3} = \begin{pmatrix} 0.0824 \\ 0.2471 \\ 0.3177 \\ 0.2269 \\ 0.0972 \\ 0.025 \\ 0.0036 \\ 0.0002 \end{pmatrix}$ **P(X=k)**

^ This protype gives us confidence in the meaning of the built-in function dbinom(k,n,p) ...

Software calculation of the Cumulative Binomial Distribution:

They also have functions calculating the cumulative distribution also:

$C_{B3_k} := \text{pbinom}(k, n, p) \qquad n = 7 \qquad p = 0.3$



$C_{B3} = \begin{pmatrix} 0.0824 \\ 0.3294 \\ 0.6471 \\ 0.874 \\ 0.9712 \\ 0.9962 \\ 0.9998 \\ 1 \end{pmatrix}$

^ Compare PB3 with CB3 above. The cumulative distribution simply adds the probabilities P(X=k) as k goes from 0 to n.

Note that these functions make obsolete traditional standard tables, such as Table 1 in the Rosner's Appendix. However, it is important to know what these functions are actually doing, so consulting the this table serves as an important prototype.

Use the table to verify whether we obtained the correct values for:

n=7, p = 0.3 and k = {0,1,2, ... n}.

Software calculation of the Inverse Cumulative Binomial Distribution:

Most computer packages also include functions for calculating the inverse of standard probability distribution functions. In other workds, they are designed to allow us to go backwards and recover X from the cumulative distribution of P(X).

$$P_{B3} = \begin{pmatrix} 0.0824 \\ 0.2471 \\ 0.3177 \\ 0.2269 \\ 0.0972 \\ 0.025 \\ 0.0036 \\ 0.0002 \end{pmatrix} \quad C_{B3} = \begin{pmatrix} 0.0824 \\ 0.3294 \\ 0.6471 \\ 0.874 \\ 0.9712 \\ 0.9962 \\ 0.9998 \\ 1 \end{pmatrix}$$

$n = 7 \quad p = 0.3$

$Q_{B3_k} := qbinom\left(p, n, C_{B3_k}\right)$

^ inverse function

$$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{pmatrix} \quad Q_{B3} = \begin{pmatrix} 0 \\ 2 \\ 4 \\ 6 \\ 7 \\ 7 \\ 7 \\ 7 \end{pmatrix}$$

^ cumulative probability distribution

^ "exact" probability distribution

Here QB3 are the values ^ of X recovered from f(X=k) for different values of k as above.

As you can see, this inverse function works, but not all that well... One is attempting to convert probabilities P(X) which MIGHT be viewed as continuous into discrete values X. This invariably involves deciding on boundaries in P(X) to assign to each X. Still, one might have hoped for a better implimentation - so I wil be very careful in using this function in the future - thus the value of prototyping! Perhaps another software package does a better job...

Calculating Mean and Variance of a Binomial Distribution:

Mean of the binomial population also known as "Expected Value":

$n := 7 \quad p := .3 \quad q := 1 - p \quad q = 0.7$ **< parameters of the binomial distribution**

$k := 0 .. n$ **< values of X = k**

$\mu := \sum_k k \cdot dbinom(k, n, p) \quad \mu = 2.1 \quad n \cdot p = 2.1$ **< μ = n*p verified!**

^ general formula for calculating the mean of a discrete distribution

Variance of the binomial population:

$$Var := \sum_k (k - n \cdot p)^2 \cdot combin(n, k) \cdot p^k \cdot q^{n-k} \quad Var = 1.47 \quad n \cdot p \cdot q = 1.47$$

^ variance = n*p*q verified!

^ general formula for calculating variance of a discrete distribution

Generating Pseudo-Random Samples of a Binomial Distribution:

Most computer packages provide a function for generating a "random" sample of data using a built-in random number generator. These samples are very useful in comparing "real" data and prototyping procedures. It must be noted, however, that no "random" number generator implemented by instructions in a computer can be truly random. So, we call them "pseudo-random". In the better programs, however, pseudo-random data nevertheless can be very realistic.

$n := 7 \quad p := .3$ **< We must specify these parameters for our intended binomial distribution.**

$m := 100$ **< We must also tell the pseudo-random number generator how many datapoints we want.**

$R1 := \text{rbinom}(m, n, p)$ **< Our random sample is placed in variable R1, so let's evaluate it!**

	Sample R:		**Binomial Distribution (population)**	
mean >	$X_{barR1} := \text{mean}(R1)$	$X_{barR1} = 1.95$	$\mu := n \cdot p$	$\mu = 2.1$
variance >	$Ssq_{R1} := \frac{m}{m-1} \cdot \text{var}(R1)$	$Ssq_{R1} = 1.3207$	$varR := n \cdot p \cdot q$	$varR = 1.47$

If we collect a larger sample, then we might expect the sample and population mean and variance to be closer - assuming the random number generator is up to the task!

$n := 7 \quad p := .3$ **< We must specify these parameters for our intended binomial distribution.**

$m := 5000$ **< We must also tell the pseudo-random number generator how many datapoints we want.**

$R2 := \text{rbinom}(m, n, p)$ **< Our random sample is placed in variable R, so let's evaluate it!**

	Sample R:		**Binomial Distribution (population)**	
mean >	$X_{barR2} := \text{mean}(R2)$	$X_{barR2} = 2.1206$	$\mu := n \cdot p$	$\mu = 2.1$
variance >	$Ssq_{R2} := \frac{m}{m-1} \cdot \text{var}(R2)$	$Ssq_{R2} = 1.518359$	$varR := n \cdot p \cdot q$	$varR = 1.47$

^ close, but certainly not stellar...

Binomial Distribution - prototyping examples from Rosner text

ORIGIN ≡ 0

Example 4.26, p. 96:

INFECTIOUS DISEASE

$n := 10 \quad p := 0.2$ **< binomial distribution parameters**

$k := 2$ **< value of X**

$dbinom(k, n, p) = 0.302$ **< "exact" value of P(X)**

Example 4.27, p. 97:

PULMONARY DISEASE

$n := 20 \quad p := 0.05$ **< binomial distribution parameters**

$k := 0 .. 2$ **< value of X**

$P_{B4_k} := dbinom(k, n, p) \qquad P_{B4} = \begin{pmatrix} 0.3585 \\ 0.3774 \\ 0.1887 \end{pmatrix}$ **< "exact" values of P(X)**

$\sum_k P_{B4_k} = 0.9245$ **< summing probabilites gives P(X < 3)**

$1 - \sum_k P_{B4_k} = 0.0755$ **< since we know that 1-P(X<3) = P(** $X \geq 3$**)**

Note that the cumulative function will calculate the sums for us automatically:

$n := 20 \quad p := 0.05$

$k := 0 .. 2$

$C_{B4_k} := pbinom(k, n, p) \qquad C_{B4} = \begin{pmatrix} 0.3585 \\ 0.7358 \\ 0.9245 \end{pmatrix}$

or even more directly:

$k := 2$

$pbinom(k, n, p) = 0.9245$ **< summing probabilites gives P(X < 3)**

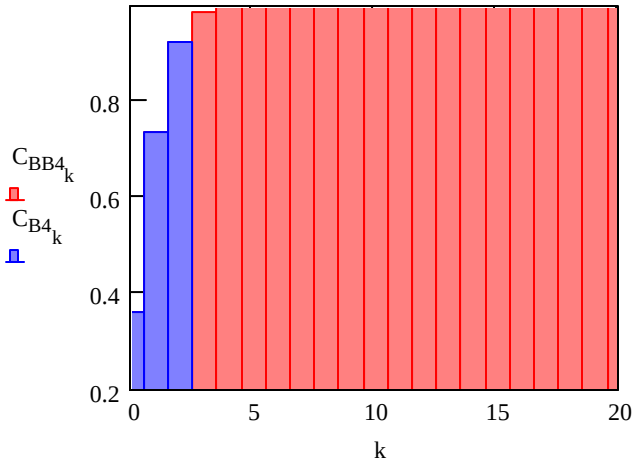
$1 - pbinom(k, n, p) = 0.0755$ **< since we know that 1-P(X<3) = P(** $X \geq 3$**)**

Visualizing the cumulative probabilities as areas under the curve:

$n := 20 \quad p := 0.05$

$k := 0 .. n$

$C_{BB4_k} := pbinom(k, n, p)$ **< Entire probability curve for k = {0,1,2 ... 20}**



< Red : P(X < 3)

< Blue : P($X \geq 3$**)**

Example 4.28, p. 98: INFECTIOUS DISEASE

Direct calculation:

$n := 5 \quad p := 0.6$ **< binomial distribution parameters**

$k := 0 .. n$ **< value of X**

$P_{B5_k} := dbinom(k, n, p)$ $P_{B5} = \begin{pmatrix} 0.0102 \\ 0.0768 \\ 0.2304 \\ 0.3456 \\ 0.2592 \\ 0.0778 \end{pmatrix}$ **<- P(X=0)** **< "exact" values of P(X)** **<- P(X=5)**

But also:

$n := 5 \quad q := 1 - p \quad q = 0.4$ **< binomial distribution parameters using q**

$k := 0 .. n$ **< value of X**

$P_{B5Q_k} := dbinom(k, n, q)$ $P_{B5Q} = \begin{pmatrix} 0.0778 \\ 0.2592 \\ 0.3456 \\ 0.2304 \\ 0.0768 \\ 0.0102 \end{pmatrix}$ **<- P(Y=0)** **< "exact" values of P(Y)** **<- P(Y=5)**

^ Same values as above, but reversed in order...

Example 4.29, p. 98: PULMONARY DISEASE

$n := 1500 \quad p := 0.05$ **< binomial distribution parameters**

$k := 75$ **< value of X**

$dbinom(k, n, p) = 0.0472$ **< Exact value for X = k = 75 cases in 1500 trials**

$k := 74$

$pbinom(k, n, p) = 0.4835$ **< This is the cumulative probability of obtaining 74 or fewer cases**

$1 - pbinom(k, n, p) = 0.5165$ **< This is the cumulative probability of obtaining 75 or greater cases... The only tricky thing is placing the cut off in the distribution...**

Example 4.30, p. 99: INFECTIOUS DISEASE

$n := 100 \quad p := 0.020$ **< binomial distribution parameters**

$k := 4$ **< value of X**

$pbinom(k, n, p) = 0.9492$ **< This is the cumulative probability of 1-4 deaths**

$1 - pbinom(k, n, p) = 0.0508$ **< This is the cumulative probability of >4 deaths**

but if we evaluate ten deaths instead of five:

$n := 100 \quad p := 0.020$

$k := 9$

$pbinom(k, n, p) = 0.999966$ **< This is the cumulative probability of 1-9 deaths**

$1 - pbinom(k, n, p) = 0.000034$ **< This is the cumulative probability of >9 deaths**

The Poisson Distribution

ORIGIN ≡ 0

The Poisson Probability Distribution (also called probability-mass function) is a discrete distribution designed to simulate very rare events in time or highly spacially separated occurrences in space.

The idea of "rare events" here depends on the following assumptions (see Rosner p. 103):

- The probability of observing 1 event is directly proportional to the length of time interval (or space) Δt so that the probability of the event P(X) is approximately $\lambda\Delta t$ for some constant λ.

- The probability of observing 0 events over Δt is approximately $1-\lambda\Delta t$.

- The probability of observing more than one event over $\lambda\Delta t$ is approximately 0.

The Poisson Distribution is also based on these assumptions:

- **Stationarity - The average or total number of events over time stays constant.**

- **Independence - The occurrence of an event in one time interval has no bearing on the occurrence of an event in a subsequent time (or space) interval.**

As with other Probability Distributions, the Poisson Distributions associates "events" X with the probability of occurrence P(X=k) - in this case over intervals of time (or space) Δt. It has a single parameter that must be specified that occurs in one of two forms λ or μ with:

μ = the expected number of events over interval (of time or space) t.

λ = the the expected number of events over unit (of time or space) t.

$$\mu = \lambda t$$

Example 4.33, Rosner p. 104:

For 6 month time interval:

INFECTIOUS DISEASE

$\lambda := 4.6$ **< 4.6 deaths per year expected rate**

$t := 0.5$ **< analyzed time interval 6 months = 0.5 year**

$\mu := \lambda \cdot t \qquad \mu = 2.3$ **< total expected number of events over 0.5 year.**

$k := 0 .. 6$ **< looking at deaths over monthly intervals 0-6.**

$$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

$P_{P1_k} := e^{-\mu} \cdot \frac{\mu^k}{k!} \qquad 0! = 1$

$$P_{P1} = \begin{pmatrix} 0.1 \\ 0.231 \\ 0.265 \\ 0.203 \\ 0.117 \\ 0.054 \\ 0.021 \end{pmatrix}$$

< "exact" Poisson probabilities P(X=k) for time intervals k

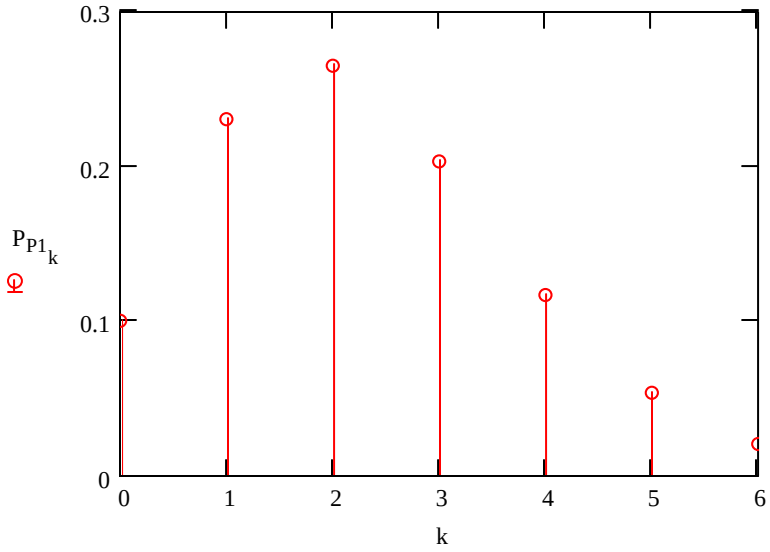
< "exact" Poisson probability P(X=6) for 7th time interval

$i := 0 .. 5$

$\sum_i P_{P1_i} = 0.97$ **< summing intervals 0-6**

$1 - \sum_i P_{P1_i} = 0.03$ **< remainder P(X ≥ 6)**

Plot:



For 3 month time interval:

$\lambda := 4.6$ **< 4.6 deaths per year expected rate**

$t := 0.25$ **< analyzed time interval 3 months = 0.25 year**

$\mu := \lambda \cdot t \qquad \mu = 1.15$ **< total expected number of events over 0.25 year**

$k := 0 .. 4 \qquad k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \end{pmatrix}$ **< looking at deaths over monthly intervals 0-4.**

$P_{P2_k} := e^{-\mu} \cdot \frac{\mu^k}{k!} \qquad 0! = 1$

$i := 0 .. 3$

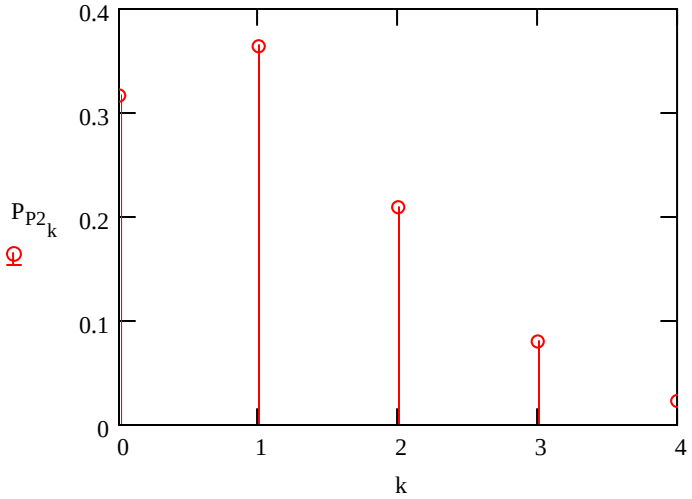
$\sum_i P_{P2_i} = 0.97$ **< summing intervals 0**

$1 - \sum_i P_{P2_i} = 0.03$ **< remainder P(X ≥ 4)**

$P_{P2} = \begin{pmatrix} 0.317 \\ 0.364 \\ 0.209 \\ 0.08 \\ 0.023 \end{pmatrix}$ **< "exact" Poisson probabilities P(X=k) for time intervals k**

< "exact" Poisson probability P(X=4) for 5th time interval

Plot:



Built-in Software functions: Exact Probabilities:

Equivalent functions for the Poisson Distribution appear in most software packages:

$\lambda := 4.6 \quad t := 0.5 \quad \mu := \lambda \cdot t$

$k := 0 .. 6$

$P_{P3_k} := dpois(k, \mu)$

Prototype confirmed although terminology in the help section of the program confuses λ and μ.

$$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix} \quad P_{P1} = \begin{pmatrix} 0.1003 \\ 0.2306 \\ 0.2652 \\ 0.2033 \\ 0.1169 \\ 0.0538 \\ 0.0206 \end{pmatrix} \quad P_{P3} = \begin{pmatrix} 0.1003 \\ 0.2306 \\ 0.2652 \\ 0.2033 \\ 0.1169 \\ 0.0538 \\ 0.0206 \end{pmatrix}$$

result of built-in function ^

^ explicitly calculated above

Built-in Software functions: Cumulative Probabilities:

Equivalent functions for the Poisson Distribution appear in most software packages:

$\lambda := 4.6 \quad t := 0.5 \quad \mu := \lambda \cdot t$

$k := 0 .. 6$

$C_{P3_k} := ppois(k, \mu)$

Prototype confirmed although terminology in the help section of the program confuses λ and μ. This function sums the "exact" probabilities as one might expect.

$$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix} \quad P_{P1} = \begin{pmatrix} 0.1003 \\ 0.2306 \\ 0.2652 \\ 0.2033 \\ 0.1169 \\ 0.0538 \\ 0.0206 \end{pmatrix} \quad C_{P3} = \begin{pmatrix} 0.1003 \\ 0.3309 \\ 0.596 \\ 0.7993 \\ 0.9162 \\ 0.97 \\ 0.9906 \end{pmatrix}$$

result of built-in function ^

^ explicitly calculated above

Built-in Software functions: Inverse Cumulative Probabilities:

Equivalent functions for the Poisson Distribution appear in most software packages:

$\lambda := 4.6 \quad t := 0.5 \quad \mu := \lambda \cdot t$

$k := 0 .. 6$

$P_{P3_k} := qpois(C_{P3_k}, \mu)$

Prototype confirmed although terminology in the help section of the program confuses λ and μ. This function recovers the k categories quite well...

$$k = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix} \quad C_{P3} = \begin{pmatrix} 0.1003 \\ 0.3309 \\ 0.596 \\ 0.7993 \\ 0.9162 \\ 0.97 \\ 0.9906 \end{pmatrix} \quad P_{P3} = \begin{pmatrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \end{pmatrix}$$

result of built-in function ^

^ explicitly calculated above

Mean (Expected Value) and Variance of the Poisson Distribution:

Both mean and variance of a Poisson Distribution = μ **Rosner p. 107**

$\mu = 2.3$ **< from the previous examples...**

Generating a Pseudo-random Poisson Distribution:

$\lambda := 4.6 \quad t := 0.5 \quad \mu := \lambda \cdot t$ **< parameter of the poisson distribution**

$m := 100$ **< number of points**

$R3 := rpois(m, \mu)$

Sample R3: **Poisson Distribution (population)**

mean > $X_{barR3} := mean(R3)$ $X_{barR3} = 2.04$ $\mu = 2.3$

variance > $Ssq_{R3} := \frac{m}{m-1} \cdot var(R3)$ $Ssq_{R3} = 1.917576$ $\mu = 2.3$

^ OK

Generating a LARGER Pseudo-random Poisson Distribution:

$\lambda := 4.6 \quad t := 0.5 \quad \mu := \lambda \cdot t$ **< parameter of the poisson distribution**

$m := 5000$ **< number of points**

$R4 := rpois(m, \mu)$

Sample R4: **Poisson Distribution (population)**

mean > $X_{barR4} := mean(R4)$ $X_{barR4} = 2.3332$ $\mu = 2.3$

variance > $Ssq_{R4} := \frac{m}{m-1} \cdot var(R4)$ $Ssq_{R4} = 2.369452$ $\mu = 2.3$

^ better but still just OK

More Poisson Examples for Prototype from Rosner:

Example 4.35, p. 106: COMPARE WITH TABLE 2 IN APPENDIX

$\mu := 3$ **< expected number over total interval t**

$k := 0 .. 4$ **< events (0 up to 4)**

$$dpois(k,\mu) = \begin{pmatrix} 0.0498 \\ 0.1494 \\ 0.224 \\ 0.224 \\ 0.168 \end{pmatrix}$$ **< P(X=0)** **< P(X=4)**

exact

$$ppois(k,\mu) = \begin{pmatrix} 0.0498 \\ 0.1991 \\ 0.4232 \\ 0.6472 \\ 0.8153 \end{pmatrix}$$ **< cumulative P(** $X \le 4$**)**

cumulative

$1 - ppois(4,\mu) = 0.1847$ **< cumulative P(X>4)**

Example 4.36, p. 106: INFECTIOUS DISEASE

$\lambda := 4.6$ **< rate per year**

$t := 1.0$ **< interval analyzed**

$\mu := \lambda \cdot t$ $\mu = 4.6$ **< expected number over total interval t**

$k := 0 .. 8$ **< events (0 up to 8)**

$$dpois(k,\mu) = \begin{pmatrix} 0.0101 \\ 0.0462 \\ 0.1063 \\ 0.1631 \\ 0.1875 \\ 0.1725 \\ 0.1323 \\ 0.0869 \\ 0.05 \end{pmatrix}$$ **< P(X=0)** **< P(X=8)**

exact

$$ppois(k,\mu) = \begin{pmatrix} 0.0101 \\ 0.0563 \\ 0.1626 \\ 0.3257 \\ 0.5132 \\ 0.6858 \\ 0.818 \\ 0.9049 \\ 0.9549 \end{pmatrix}$$ **< cumulative P(** $X \le 8$**)**

cumulative

$1 - ppois(8,\mu) = 0.0451$ **< cumulative P(X>8)**

Example 4.38, p. 108: OCCUPATIONAL HEALTH

$\lambda := 5.8$ **< rate per unit time or space**

$t := 1.0$ **< interval analyzed**

$\mu := \lambda \cdot t \quad \mu = 5.8$ **< expected number over total interval t**

$k := 0 .. 6$ **< events (0 up to 6)**

$$\text{dpois}(k, \mu) = \begin{pmatrix} 0.003 \\ 0.0176 \\ 0.0509 \\ 0.0985 \\ 0.1428 \\ 0.1656 \\ 0.1601 \end{pmatrix} \quad \begin{matrix} \text{< P(X=0)} \\ \\ \\ \\ \\ \\ \text{< P(X=6)} \end{matrix}$$

exact

$$\text{ppois}(k, \mu) = \begin{pmatrix} 0.003 \\ 0.0206 \\ 0.0715 \\ 0.17 \\ 0.3127 \\ 0.4783 \\ 0.6384 \end{pmatrix} \quad \text{< cumulative P(} X \leq 6\text{)}$$

cumulative

$1 - \text{ppois}(6, \mu) = 0.3616$ **< cumulative P(X>6)**

Example 4.39, p. 109: CANCER GENETICS

$\lambda := 1.0$ **< rate per unit time or space**

$t := 1.0$ **< interval analyzed**

$\mu := \lambda \cdot t \quad \mu = 1$ **< expected number over total interval t**

$k := 0 .. 3$ **< events (0 up to 4)**

$$\text{dpois}(k, \mu) = \begin{pmatrix} 0.3679 \\ 0.3679 \\ 0.1839 \\ 0.0613 \end{pmatrix} \quad \begin{matrix} \text{< P(X=0)} \\ \\ \\ \text{< P(X=3)} \end{matrix}$$

exact

$$\text{ppois}(k, \mu) = \begin{pmatrix} 0.3679 \\ 0.7358 \\ 0.9197 \\ 0.981 \end{pmatrix} \quad \text{< cumulative P(} X \leq 3\text{)}$$

cumulative

$1 - \text{ppois}(3, \mu) = 0.019$ **< cumulative P(X>3)**

Example 4.40, p. 110: INFECTIOUS DISEASE

$\lambda := 3.67$ **< rate per unit time or space - average rate per month over 18 months**

$t := 1.0$ **< interval analyzed - looking at an unusual one month**

$\mu := \lambda \cdot t \quad \mu = 3.67$ **< expected number over total interval t - expected rate in that unusual month**

$1 - \text{ppois}(13, \mu) = 3.0924 \times 10^{-5}$ **< cumulative P(** $X \geq 14$**)**

Assignment for Week 4

The readings in our text this week and last, involve the fundamental relationship between data we might collect (generally termed X) and the probability different values of data might have (termed 'P(X)'). In real world situations, of course, we don't usually know what the probability of X might be. In general, one usually consults one or several 'exact probability functions' for discrete variables or 'probability density functions' for continuous variables that have proven over the years to be very useful. For each of these probability functions, it is important to understand the basic rationale underlying the use of the distributions and the parameters that define specific P(X) given X from a family of similar curves. Deciding the suitability of fit between real data with theoretical distributions often involves comparing histograms of real data with what might be theoretically expected of the distributions or simulated, such as through R's 'r' statistical functions. If the fit seems good, one then proceeds to use the standard probability distributions to estimate **probability of particular values of X**, **probability cutoffs**, **and probability intervals**. In essence, this is all that statistics does in the design of confidence intervals and statistical tests.

Because associating values of X with P(X) is so important, all statistics texts include tables like those in Rosner's Appendix designed to simplify calculations of otherwise complex formulae. Standard software packages, such as R, include explicit 'd', 'p' & 'q' functions to do the same thing, often with greater precision. ***To proceed with statistics, it is essential that you understand how these tables and functions work***. It is also important to be able to use this theoretical apparatus to work boundaries in either X or P(X).

So, this week your assignment is to complete your prototype of five important probability distributions: **Binomial**, **Poisson**, **Normal**, **Student's t**, and **Chi-square**.

1. Set up a range of X's and use the 'd' function to calculate P(X). To do this, you will have to pick 'reasonable' values of each distribution's parameters.
2. Plot P(X) vs X to visualize each distribution. Compare this 'exact' curve with a histogram of simulated data generated by each distribution's corresponding 'r' function. Notice the fit of simulated data with the theoretical P(X) vs X function – or lack thereof.
3. Calculate the cumulative function $\Phi(X)$ for each X using a corresponding 'p' function. and plot $\Phi(X)$ vs X. Show the relationship between P(X) and $\Phi(X)$ for each X.
4. Now show how to retrieve X from $\Phi(X)$ using the inverse cumulative probability 'q' function. Interpret in words what this function allows you to do.
5. For each plot of P(X) vs X, characterize the distribution's shape. Note whether the distribution is symmetrical or non symmetrical. Note its central tendency or mode versus tail(s).
6. For each distribution, find the values of X below or above which $P(X) < 5\%$. Annotate your graph of P(X) vs X to show what this means.
7. For each distribution, find lower X_{lower} and upper X_{upper} bound values of X such that P(X) is at least 95%

Welcome to the world of Confidence Intervals – Chapter 6!

Here are the R Documentation Pages for the distributions we are trying to prototype:

Normal {stats} R Documentation

The Normal Distribution

Description

Density, distribution function, quantile function and random generation for the normal distribution with mean equal to `mean` and standard deviation equal to `sd`.

Usage

```
dnorm(x, mean=0, sd=1, log = FALSE)
pnorm(q, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
qnorm(p, mean=0, sd=1, lower.tail = TRUE, log.p = FALSE)
rnorm(n, mean=0, sd=1)
```

Arguments

`x,q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations. If `length(n) > 1`, the length is taken to be the number required.
`mean`	vector of means.
`sd`	vector of standard deviations.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$, otherwise, $P[X > x]$.

Details

If `mean` or `sd` are not specified they assume the default values of `0` and `1`, respectively.

The normal distribution has density

$$f(x) = 1/(\sqrt{2\pi}\sigma) e^{-((x-\mu)^2/(2\sigma^2))}$$

where μ is the mean of the distribution and σ the standard deviation.

`qnorm` is based on Wichura's algorithm AS 241 which provides precise results up to about 16 digits.

Value

`dnorm` gives the density, `pnorm` gives the distribution function, `qnorm` gives the quantile function, and `rnorm` generates random deviates.

Source

For `pnorm`, based on

Cody, W. D. (1993) Algorithm 715: SPECFUN – A portable FORTRAN package of special function routines and test drivers. *ACM Transactions on Mathematical Software* **19**, 22–32.

For `qnorm`, the code is a C translation of

Wichura, M. J. (1988) Algorithm AS 241: The Percentage Points of the Normal Distribution. *Applied Statistics*, **37**, 477–484.

For `rnorm`, see RNG for how to select the algorithm and for references to the supplied methods.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

Johnson, N. L., Kotz, S. and Balakrishnan, N. (1995) *Continuous Univariate Distributions*, volume 1, chapter 13. Wiley, New York.

See Also

`runif` and `.Random.seed` about random number generation, and `dlnorm` for the *Log*normal distribution.

Examples

```
dnorm(0) == 1/ sqrt(2*pi)
dnorm(1) == exp(-1/2)/ sqrt(2*pi)
dnorm(1) == 1/ sqrt(2*pi*exp(1))

## Using "log = TRUE" for an extended range :
par(mfrow=c(2,1))
plot(function(x) dnorm(x, log=TRUE), -60, 50,
     main = "log { Normal density }")
curve(log(dnorm(x)), add=TRUE, col="red",lwd=2)
mtext("dnorm(x, log=TRUE)", adj=0)
mtext("log(dnorm(x))", col="red", adj=1)
```

```
plot(function(x) pnorm(x, log=TRUE), -50, 10,
     main = "log { Normal Cumulative }")
curve(log(pnorm(x)), add=TRUE, col="red",lwd=2)
mtext("pnorm(x, log=TRUE)", adj=0)
mtext("log(pnorm(x))", col="red", adj=1)

## if you want the so-called 'error function'
erf <- function(x) 2 * pnorm(x * sqrt(2)) - 1
## (see Abrahamowitz and Stegun 29.2.29)
## and the so-called 'complementary error function'
erfc <- function(x) 2 * pnorm(x * sqrt(2), lower = FALSE)
```

The Student t Distribution

Description

Density, distribution function, quantile function and random generation for the t distribution with `df` degrees of freedom (and optional noncentrality parameter `ncp`).

Usage

```
dt(x, df, ncp = 0, log = FALSE)
pt(q, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)
qt(p, df, ncp = 0, lower.tail = TRUE, log.p = FALSE)
rt(n, df, ncp = 0)
```

Arguments

`x, q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations. If `length(n) > 1`, the length is taken to be the number required.
`df`	degrees of freedom (> *0*, maybe non-integer). `df = Inf` is allowed. For `qt` only values of at least one are currently supported.
`ncp`	non-centrality parameter *delta*; currently for `pt()` and `dt()`, only for `abs(ncp) <= 37.62`.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*, otherwise, *P[X > x]*.

Details

The *t* distribution with `df` = *n* degrees of freedom has density

$$f(x) = \Gamma((n+1)/2) / (\sqrt{n\pi}\,\Gamma(n/2))\,(1 + x^2/n)^{-((n+1)/2)}$$

for all real *x*. It has mean *0* (for *n > 1*) and variance *n/(n-2)* (for *n > 2*).

The general *non-central t* with parameters *(df,Del)* `= (df, ncp)` is defined as the distribution of *T(df, Del) := (U + Del) / (Chi(df) / sqrt(df))* where *U* and *Chi(df)* are independent random variables, *U ~ N(0,1)*, and *Chi(df)^2* is chi-squared, see Chisquare.

The most used applications are power calculations for *t*-tests:
Let *T= (mX - m0) / (S/sqrt(n))* where *mX* is the `mean` and *S* the sample standard deviation (`sd`) of *X_1,X_2,...,X_n* which are i.i.d. *N(mu,sigma^2)*. Then *T* is distributed as non-centrally *t* with `df`= *n-1* degrees of freedom and **n**on-**c**entrality **p**arameter `ncp`= *(mu - m0) * sqrt(n)/sigma*.

Value

`dt` gives the density, `pt` gives the distribution function, `qt` gives the quantile function, and `rt` generates random deviates.
Invalid arguments will result in return value `NaN`, with a warning.

Source

The central `dt` is computed via an accurate formula provided by Catherine Loader (see the reference in `dbinom`).

For the non-central case of `dt`, contributed by Claus Ekstrøm based on the relationship (for *x != 0*) to the cumulative distribution.

For the central case of `pt`, a normal approximation in the tails, otherwise via `pbeta`.

For the non-central case of `pt` based on a C translation of

Lenth, R. V. (1989). *Algorithm AS 243* — Cumulative distribution function of the non-central *t* distribution, *Applied Statistics* **38**, 185–189.

For central `qt`, a C translation of

Hill, G. W. (1970) Algorithm 396: Student's t-quantiles. *Communications of the ACM*, **13(10)**, 619–620.

altered to take account of

Hill, G. W. (1981) Remark on Algorithm 396, *ACM Transactions on Mathematical Software*, **7**, 250–1.

The non-central case is done by inversion.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole. (Except non-central versions.)

Johnson, N. L., Kotz, S. and Balakrishnan, N. (1995) *Continuous Univariate Distributions*, volume 2, chapters 28 and 31. Wiley, New York.

See Also

`df` for the F distribution.

Examples

```
1 - pt(1:5, df = 1)
qt(.975, df = c(1:10,20,50,100,1000))

tt <- seq(0,10, len=21)
ncp <- seq(0,6, len=31)
ptn <- outer(tt,ncp, function(t,d) pt(t, df = 3, ncp=d))
image(tt,ncp,ptn, zlim=c(0,1),main=t.tit <- "Non-central t -
Probabilities")
persp(tt,ncp,ptn, zlim=0:1, r=2, phi=20, theta=200, main=t.tit,
      xlab = "t", ylab = "noncentrality parameter", zlab = "Pr(T <=
t)")

plot(function(x) dt(x, df = 3, ncp = 2), -3, 11, ylim = c(0, 0.32),
     main="Non-central t - Density", yaxs="i")
```

The (non-central) Chi-Squared Distribution

Description

Density, distribution function, quantile function and random generation for the chi-squared (*chi^2*) distribution with `df` degrees of freedom and optional non-centrality parameter `ncp`.

Usage

```
dchisq(x, df, ncp=0, log = FALSE)
pchisq(q, df, ncp=0, lower.tail = TRUE, log.p = FALSE)
qchisq(p, df, ncp=0, lower.tail = TRUE, log.p = FALSE)
rchisq(n, df, ncp=0)
```

Arguments

`x, q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations. If `length(n) > 1`, the length is taken to be the number required.
`df`	degrees of freedom (non-negative, but can be non-integer).
`ncp`	non-centrality parameter (non-negative).
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are *P[X <= x]*, otherwise, *P[X > x]*.

Details

The chi-squared distribution with `df`= *n > 0* degrees of freedom has density

$$f_n(x) = 1 / (2^{n/2} \Gamma(n/2)) x^{n/2-1} e^{-x/2}$$

for *x > 0*. The mean and variance are *n* and *2n*.

The non-central chi-squared distribution with `df`= *n* degrees of freedom and non-centrality parameter `ncp` = λ has density

$$f(x) = \exp(-\lambda/2) \sum_{r=0}^{\infty} ((\lambda/2)^r / r!) \, dchisq(x, df + 2r)$$

for $x >= 0$. For integer n, this is the distribution of the sum of squares of n normals each with variance one, λ being the sum of squares of the normal means; further, $E(X) = n + \lambda$, $Var(X) = 2(n + 2*\lambda)$, and $E((X - E(X))^3) = 8(n + 3*\lambda)$.

Note that the degrees of freedom `df` $= n$, can be non-integer, and for non-centrality $\lambda > 0$, even $n = 0$; see Johnson et al. (1995, chapter 29).

Note that `ncp` values larger than about 1e5 may give inaccurate results with many warnings for `pchisq` and `qchisq`.

Value

`dchisq` gives the density, `pchisq` gives the distribution function, `qchisq` gives the quantile function, and `rchisq` generates random deviates.
Invalid arguments will result in return value `NaN`, with a warning.

Source

The central cases are computed via the gamma distribution.

The non-central `dchisq` and `rchisq` are computed as a Poisson mixture central of chi-squares (Johnson et al, 1995, p.436).

The non-central `pchisq` is for `ncp < 80` computed from the Poisson mixture of central chi-squares and for larger `ncp` based on a C translation of

Ding, C. G. (1992) Algorithm AS275: Computing the non-central chi-squared distribution function. *Appl.Statist.*, **41** 478–482.

which computes the lower tail only (so the upper tail suffers from cancellation).

The non-central `qchisq` is based on inversion of `pchisq`.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

Johnson, N. L., Kotz, S. and Balakrishnan, N. (1995) *Continuous Univariate Distributions*, chapters 18 (volume 1) and 29 (volume 2). Wiley, New York.

See Also

A central chi-squared distribution with *n* degrees of freedom is the same as a Gamma distribution with `shape` $a = n/2$ and `scale` $s = 2$. Hence, see `dgamma` for the Gamma distribution.

Examples

```
dchisq(1, df=1:3)
pchisq(1, df= 3)
pchisq(1, df= 3, ncp = 0:4)# includes the above

x <- 1:10
## Chi-squared(df = 2) is a special exponential distribution
all.equal(dchisq(x, df=2), dexp(x, 1/2))
all.equal(pchisq(x, df=2), pexp(x, 1/2))

## non-central RNG -- df=0 is ok for ncp > 0:  Z0 has point mass at 0!
Z0 <- rchisq(100, df = 0, ncp = 2.)
graphics::stem(Z0)

## Not run:
## visual testing
## do P-P plots for 1000 points at various degrees of freedom
L <- 1.2; n <- 1000; pp <- ppoints(n)
op <- par(mfrow = c(3,3), mar= c(3,3,1,1)+.1, mgp= c(1.5,.6,0),
          oma = c(0,0,3,0))
for(df in 2^(4*rnorm(9))) {
  plot(pp, sort(pchisq(rr <- rchisq(n,df=df, ncp=L), df=df, ncp=L)),
       ylab="pchisq(rchisq(.),.)", pch=".")
  mtext(paste("df = ",formatC(df, digits = 4)), line= -2, adj=0.05)
  abline(0,1,col=2)
}
mtext(expression("P-P plots : Noncentral  "*
                 chi^2 *"(n=1000, df=X, ncp= 1.2)"),
      cex = 1.5, font = 2, outer=TRUE)
par(op)
## End(Not run)
```

The Binomial Distribution

Description

Density, distribution function, quantile function and random generation for the binomial distribution with parameters `size` and `prob`.

Usage

```
dbinom(x, size, prob, log = FALSE)
pbinom(q, size, prob, lower.tail = TRUE, log.p = FALSE)
qbinom(p, size, prob, lower.tail = TRUE, log.p = FALSE)
rbinom(n, size, prob)
```

Arguments

`x, q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of observations. If `length(n) > 1`, the length is taken to be the number required.
`size`	number of trials (zero or more).
`prob`	probability of success on each trial.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$, otherwise, $P[X > x]$.

Details

The binomial distribution with `size` = n and `prob` = p has density

$$p(x) = choose(n,x)\ p^x\ (1-p)^{(n-x)}$$

for $x = 0, ..., n$.

If an element of `x` is not integer, the result of `dbinom` is zero, with a warning. $p(x)$ is computed using Loader's algorithm, see the reference below.

The quantile is defined as the smallest value x such that $F(x) >= p$, where F is the distribution function.

Value

`dbinom` gives the density, `pbinom` gives the distribution function, `qbinom` gives the quantile function and `rbinom` generates random deviates.
If `size` is not an integer, `NaN` is returned.

Source

For `dbinom` a saddle-point expansion is used: see

Catherine Loader (2000). *Fast and Accurate Computation of Binomial Probabilities*; available from http://www.herine.net/stat/software/dbinom.html.

`pbinom` uses `pbeta`.

`qbinom` uses the Cornish–Fisher Expansion to include a skewness correction to a normal approximation, followed by a search.

`rbinom` is based on

Kachitvichyanukul, V. and Schmeiser, B. W. (1988) Binomial random variate generation. *Communications of the ACM*, **31**, 216–222.

See Also

`dnbinom` for the negative binomial, and `dpois` for the Poisson distribution.

Examples

```
# Compute P(45 < X < 55) for X Binomial(100,0.5)
sum(dbinom(46:54, 100, 0.5))

## Using "log = TRUE" for an extended range :
n <- 2000
k <- seq(0, n, by = 20)
plot (k, dbinom(k, n, pi/10, log=TRUE), type='l', ylab="log density",
      main = "dbinom(*, log=TRUE) is better than  log(dbinom(*))")
lines(k, log(dbinom(k, n, pi/10)), col='red', lwd=2)
## extreme points are omitted since dbinom gives 0.
mtext("dbinom(k, log=TRUE)", adj=0)
mtext("extended range", adj=0, line = -1, font=4)
mtext("log(dbinom(k))", col="red", adj=1)
```

The Poisson Distribution

Description

Density, distribution function, quantile function and random generation for the Poisson distribution with parameter `lambda`.

Usage

```
dpois(x, lambda, log = FALSE)
ppois(q, lambda, lower.tail = TRUE, log.p = FALSE)
qpois(p, lambda, lower.tail = TRUE, log.p = FALSE)
rpois(n, lambda)
```

Arguments

`x`	vector of (non-negative integer) quantiles.
`q`	vector of quantiles.
`p`	vector of probabilities.
`n`	number of random values to return.
`lambda`	vector of (non-negative) means.
`log, log.p`	logical; if TRUE, probabilities p are given as log(p).
`lower.tail`	logical; if TRUE (default), probabilities are $P[X <= x]$, otherwise, $P[X > x]$.

Details

The Poisson distribution has density

$$p(x) = \lambda^x \exp(-\lambda)/x!$$

for $x = 0, 1, 2, \ldots$. The mean and variance are $E(X) = Var(X) = \lambda$.

If an element of `x` is not integer, the result of `dpois` is zero, with a warning. $p(x)$ is computed using Loader's algorithm, see the reference in `dbinom`.

The quantile is left continuous: `qgeom(q, prob)` is the largest integer x such that $P(X <= x) < q$.

Setting `lower.tail = FALSE` allows to get much more precise results when the default, `lower.tail = TRUE` would return 1, see the example below.

Value

`dpois` gives the (log) density, `ppois` gives the (log) distribution function, `qpois` gives the quantile function, and `rpois` generates random deviates.
Invalid `lambda` will result in return value `NaN`, with a warning.

Source

`dpois` uses C code contributed by Catherine Loader (see `dbinom`).

`ppois` uses `pgamma`.

`qpois` uses the Cornish–Fisher Expansion to include a skewness correction to a normal approximation, followed by a search.

`rpois` uses

Ahrens, J. H. and Dieter, U. (1982). Computer generation of Poisson deviates from modified normal distributions. *ACM Transactions on Mathematical Software*, **8**, 163–179.

See Also

`dbinom` for the binomial and `dnbinom` for the negative binomial distribution.

Examples

```
-log(dpois(0:7, lambda=1) * gamma(1+ 0:7)) # == 1
Ni <- rpois(50, lam= 4); table(factor(Ni, 0:max(Ni)))

1 - ppois(10*(15:25), lambda=100)               # becomes 0
(cancellation)
    ppois(10*(15:25), lambda=100, lower=FALSE)  # no cancellation

par(mfrow = c(2, 1))
x <- seq(-0.01, 5, 0.01)
plot(x, ppois(x, 1), type="s", ylab="F(x)", main="Poisson(1) CDF")
plot(x, pbinom(x, 100, 0.01),type="s", ylab="F(x)",
     main="Binomial(100, 0.01) CDF")
```

The Normal Distribution

ORIGIN ≡ 0

The Normal Distribution, also known as the "Gaussian Distribution" or "bell-curve", is the most widely employed function relating observations X with probabilty P(X) in statistics. Many natural populations are approximately normally distributed, as are several important derived quantitities even when the original population is not normally distributed.

Properly speaking, the Normal Distribution is a continuous "probability density function" meaning that values of a random variable X may take on any numerical value, not just discrete values. In addition, because the values of X are infinite the "exact" probabiliy P(X) for any X is zero. Thus, in order to determine probabilities one typically looks at invervals of X such as X >2.3 or 1< X < 2 and so forth. It is interesting to note that because the probability P(X) = 0, we don't have to worry about correctly interpreting pesky boundaries, as seen in discrete distributions, since X > 2 means the same thing as $X \geq 2$ and X < 2 is the same as $X \leq 2$.

As described previously, the Normal distribution consists of a family of curves that are specified by supplying values for two parameters:

μ = the mean of the Normal population, and

σ^2 = the variance of the same population.

Prototyping the Normal Function using the Gaussian formula:

Making the plot of N(50,100) in Rosner Fig. 5.5 p. 127:

$\mu := 50$ **< specifying mean (μ)**

$\sigma := \sqrt{100}$ $\sigma^2 = 100$ **< specifying variance (σ^2)**

$i := 0 .. 100$

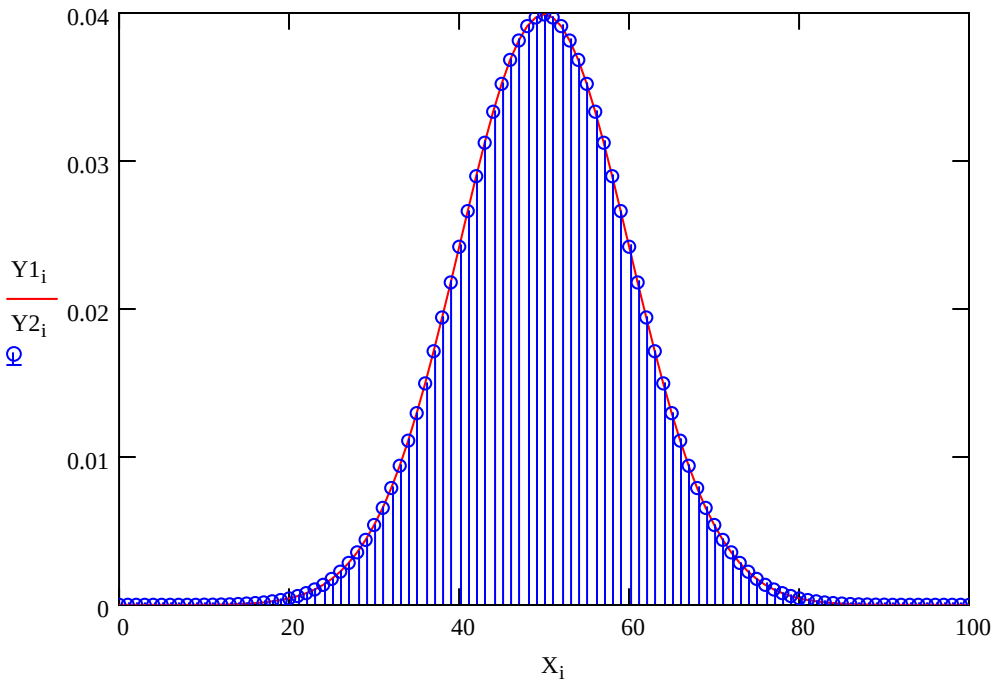
$X_i := i$ **< Defining a bunch of X's ranging in value from 0 to 100. Remember that the range of X is infinite, but we'll plot 101 point here. That should give us enough points to give us an idea of the Gaussian function shape!**

$$Y1_i := \frac{1}{\sigma \cdot \sqrt{2 \cdot \pi}} \cdot e^{\left[\frac{-1}{2 \cdot \sigma^2} \cdot (X_i - \mu)^2\right]}$$ **< Formula for Normal distribution. Here we have computed P(X) for each of our X's. Careful reading Definition 5.5 p. 126....**

Now, let's compare with Mathcad's built-in function:

$Y2_i := dnorm(X_i, \mu, \sigma)$ $\sigma^2 = 100$ **< MathCad's function asks us provide standard deviation rather than variance...**

Plotting the two sets of Y's:



^ The two approaches give the same probability function P(X) for X, so this prototype confirms the built-in function.

What happens when μ or σ^2 is changed:

Location of mode changes (translation of μ) and width of hump changes showing greater or lesser variance - see Biostatistics Lecture Worksheet 04.

Cumulative Normal Distribution N(0,1):

$i := 0 .. 100$

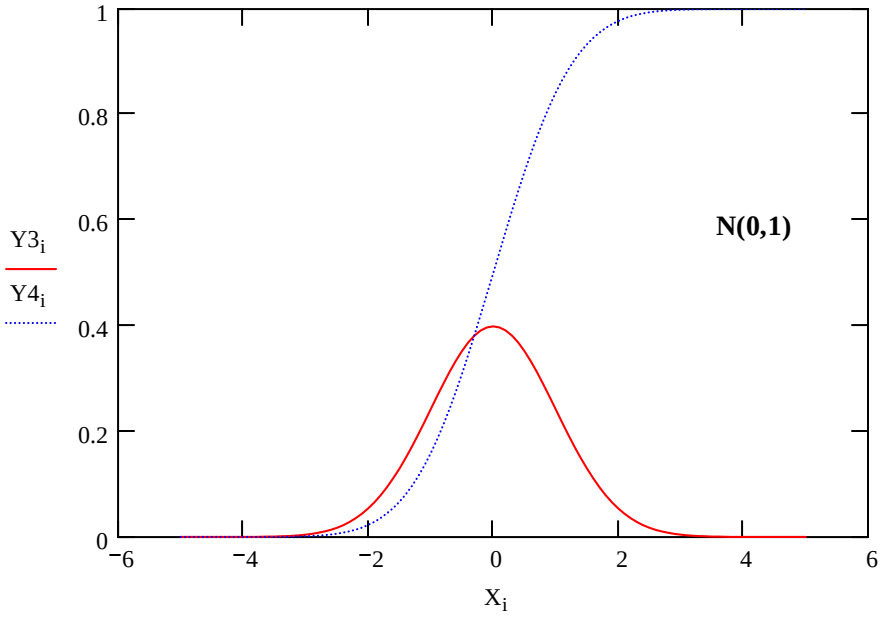
$X_i := \frac{i - 50}{10}$ **< scaling 101 X's to a reasonable scale...**

$\mu := 0 \quad \sigma := 1 \quad \sigma^2 = 1$ **< parameters of the Normal N(0,1) distribution...**

$Y3_i := dnorm(X_i, \mu, \sigma)$ **< P(X) for each X**

$Y4_i := pnorm(X_i, \mu, \sigma)$ **< Cumulative probability Φ(X) for each X**

Plots of Normal Distribution and Cumulative Normal Distributions



Calculating Intervals of the Cumulative Distribution Rosner, p. 129-130:

$\mu = 0$ $\sigma = 1$ **< Normal distribution parameters (change these if needed)**

Probability that X ranges between -1 and 1:

$dnorm(-1,\mu,\sigma) = 0.242$ $dnorm(1,\mu,\sigma) = 0.242$ **< P(X)**

$pnorm(-1,\mu,\sigma) = 0.1587$ $pnorm(1,\mu,\sigma) = 0.8413$ **< Φ(X)**

$pnorm(1,\mu,\sigma) - pnorm(-1,\mu,\sigma) = 0.6827$ **< Calculating MAX cut-off - MIN cut-off**

^ cumulative value at MIN of interval

^ cumulative value at MAX of interval

68.27%

Probability that X ranges between -2.576 and 2.576:

$dnorm(-2.576,\mu,\sigma) = 0.0145$ $dnorm(2.576,\mu,\sigma) = 0.0145$ **: P(X)**

$pnorm(-2.576,\mu,\sigma) = 0.005$ $pnorm(2.576,\mu,\sigma) = 0.995$ **< Φ(X)**

$pnorm(2.576,\mu,\sigma) - pnorm(-2.576,\mu,\sigma) = 0.99$ **< Calculating MAX cut-off - MIN cut-off**

^ cumulative value at MIN of interval

^ cumulative value at MAX of interval

99%

Probability that X ranges between -1.96 and 1.96

$dnorm(-1.96,\mu,\sigma) = 0.0584$ $dnorm(1.96,\mu,\sigma) = 0.0584$ **< P(X)**

$pnorm(-1.96,\mu,\sigma) = 0.025$ $pnorm(1.96,\mu,\sigma) = 0.975$ **< Φ(X)**

$pnorm(1.96,\mu,\sigma) - pnorm(-1.96,\mu,\sigma) = 0.95$ **< Calculating MAX cut-off - MIN cut-off**

^ cumulative value at MIN of interval

^ cumulative value at MAX of interval

95%

Standardizing the Normal Distribution:

In many instances, we will have a sample that we may compare to a Normal Distribution, normally indicated like this: ~N(μ,σ^2). Using computer-based functions as above, one has little difficulty calculating probabilities P(X) and cumulative probabilities Φ(X). However, in comparing variables it is often useful to compare probabilities for each to those expected of the Standard Normal Distribution ~N(0,1).

This is done by Standardizing the Data:

Given your X's ~N(μ,σ^2) you create a new variable Z ~N(0,1) by means of a Linear Transformation:

$\mu := 50$ $\sigma := \sqrt{100}$ $\sigma^2 = 100$ **< original distribution ~N(50,100)**

$i := 0 .. 100$

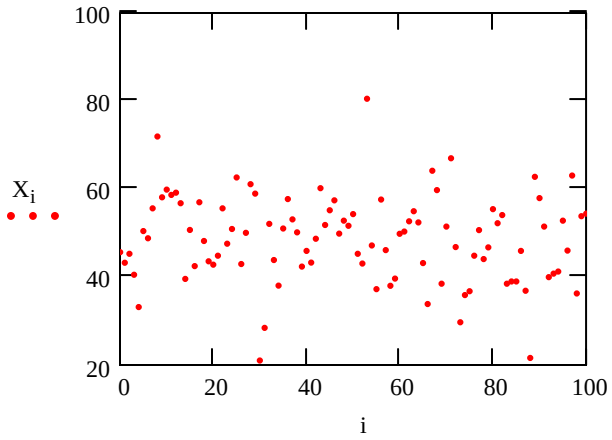
$X_i := i$

$Z_i := \frac{(X_i - \mu)}{\sigma}$ **< Z's are now Standardized ~N(0,1)**

Simulation of Normally Distributed Data:

$\mu := 50$ $\sigma := \sqrt{100}$ $\sigma^2 = 100$

$X := rnorm(1000, \mu, \sigma)$



Descriptive Statistics for X:

$n := length(X)$ $n = 1000$

$mean(X) = 49.4025$

$\frac{n}{n-1} \cdot var(X) = 97.0097$ **< both sample means - here as calculated in previous worksheets**

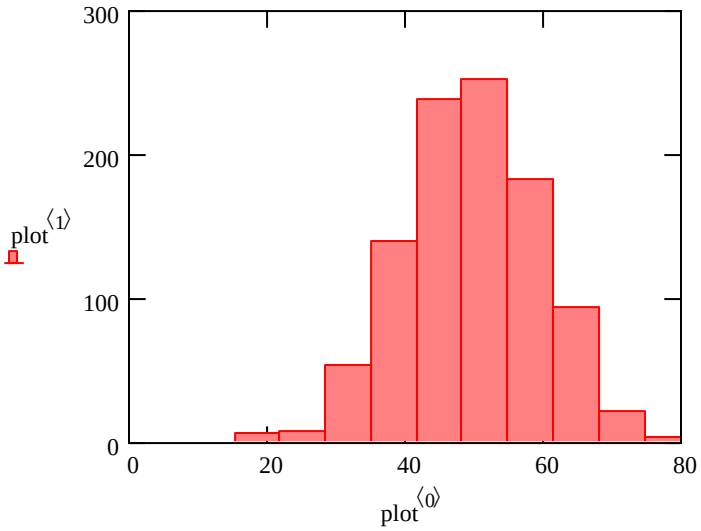
< Note: mathcad has two functions: var(X) = population variance

$Var(X) = 97.0097$ **Var(X) = sample mean**

Most computer programs have functions showing this distinction...

Histogram of X:

plot := histogram(10, X)



$$plot = \begin{pmatrix} 18.3 & 6 \\ 24.9 & 8 \\ 31.5 & 54 \\ 38.1 & 140 \\ 44.7 & 238 \\ 51.3 & 252 \\ 57.9 & 182 \\ 64.5 & 94 \\ 71.1 & 22 \\ 77.7 & 4 \end{pmatrix}$$

Standardizing our Sample Data:

$$Z := \frac{X - \mu}{\sigma}$$

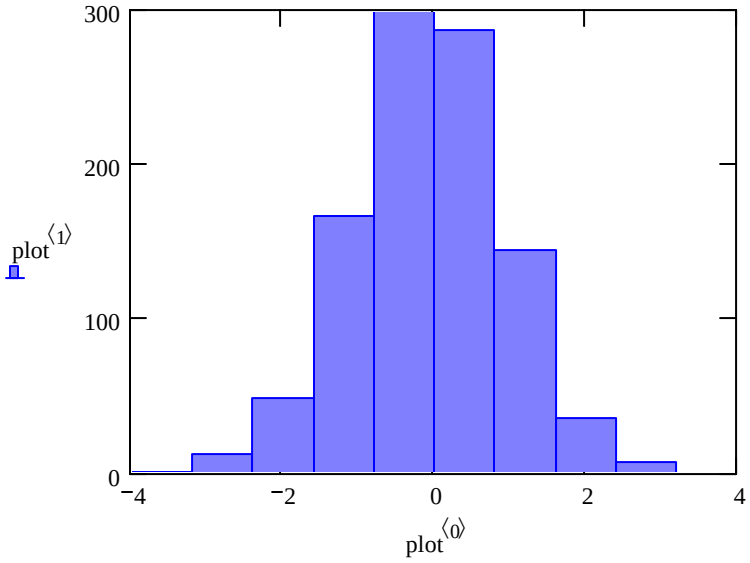
Descriptive Statistics for Z:

n := length(Z) n = 1000

mean(Z) = −0.0598

Var(Z) = 0.9701

plot := histogram(10, Z)



Linear Combinations of Variables

ORIGIN ≡ 0

In data analysis of real-life situations, it is often the case that data on multiple variables are collected. The task of the statistical researcher is then to construct important questions that might be asked of the data, and then choose an appropriate statistical technique. One common approach is to construct new variables that combine the original collected variables in a meaningful way that summarizes, and hopefully simplifies, the issues involved. This approach is often performed by constructing Linear Combinations (also known as "linear contrasts") of the original variables.

So let's grab some familiar data:

iris := READPRN("c:/2007BiostatsData/iris.txt")

SL := $iris^{\langle 1 \rangle}$

SW := $iris^{\langle 2 \rangle}$

PL := $iris^{\langle 3 \rangle}$

PW := $iris^{\langle 4 \rangle}$

n := length(SL) n = 150

A linear combination is any NEW variable we make that consists of a constant c_j (for each original variable) times each original variable all added together. We do this for each of the values i representing instances of the original variables:

j := 0 .. 3 **< index (j) of the constants: c_0, c_1, c_2 & c_3 because ORIGIN=0**

i := 0 .. 149 **< index (i) of the values in each variable SL, SW, PL,PW above**

c_0 := 1
c_1 := 1
c_2 := 1
c_3 := 1

< constants called "linear coeffieients"

$$c = \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$

< the values of linear coefficients c_j in vector form, often called the "contrast matrix"

$$LC1_i := c_0 \cdot SL_i + c_1 \cdot SW_i + c_2 \cdot PL_i + c_3 \cdot PW_i$$

$$c := \begin{pmatrix} 0.25 \\ 0.25 \\ -0.25 \\ -0.25 \end{pmatrix}$$

^ This linear combination (LC1) is made by adding each of the original variables together.

$$LC2_i := c_0 \cdot SL_i + c_1 \cdot SW_i + c_2 \cdot PL_i + c_3 \cdot PW_i$$

^ This linear combination (LC2) "contrasts" sepals (SL+SW) versus petals (PL+PW). As you can see many such "contrasts" are possible by specifying different values for the contrast matrix.

Mean and Variance of Linear Combinations:

$$c := \begin{pmatrix} 1 \\ 1 \\ 1 \\ 1 \end{pmatrix}$$ **< contrast matrix**

mean(SL) = 5.8433 Var(SL) = 0.6857

mean(SW) = 3.0573 Var(SW) = 0.19

mean(PL) = 3.758 Var(PL) = 3.1163

mean(PW) = 1.1993 Var(PW) = 0.581

< means and sample variances for each of the original variables in iris...

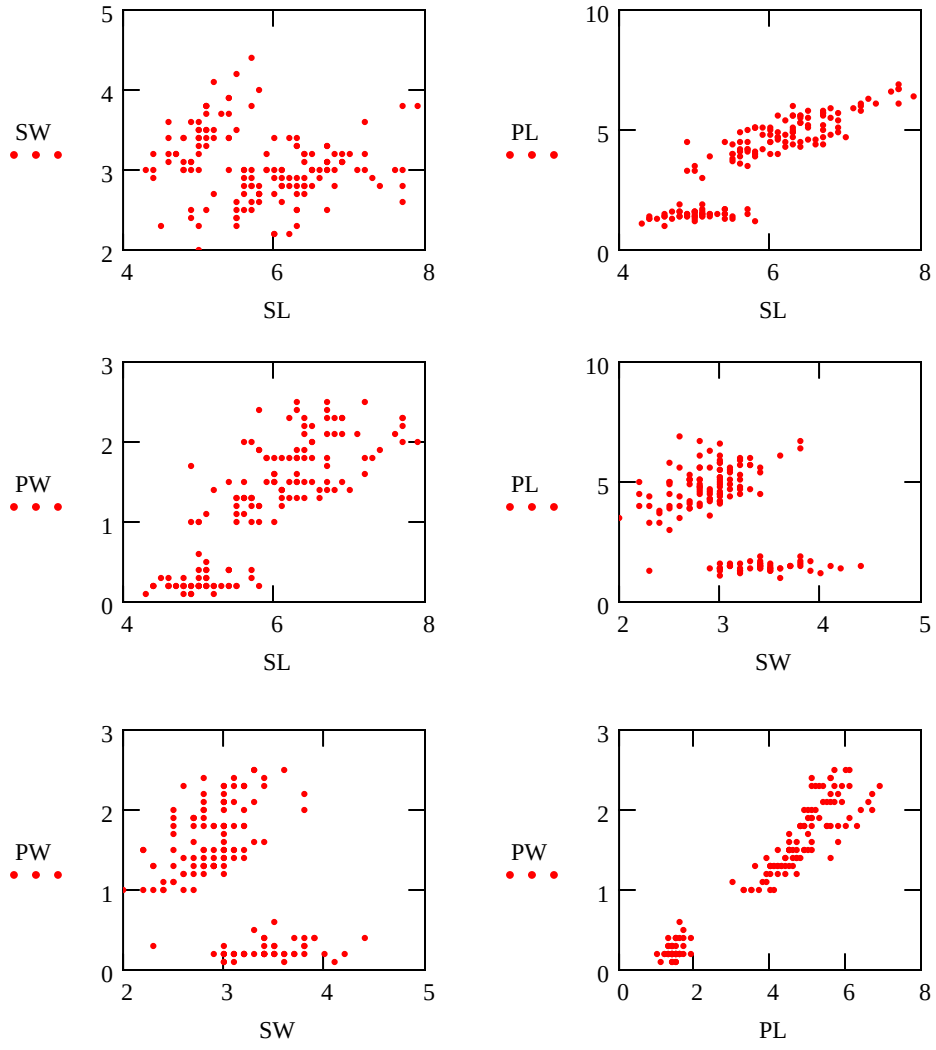
mean(LC1) = 13.858 $c_0 \cdot \text{mean(SL)} + c_1 \cdot \text{mean(SW)} + c_2 \cdot \text{mean(PL)} + c_3 \cdot \text{mean(PW)} = 13.858$

^ mean of the linear combination is the sum of each mean times its linear coefficient.

Var(LC1) = 9.7579 $(c_0)^2 \cdot \text{Var(SL)} + (c_1)^2 \cdot \text{Var(SW)} + (c_2)^2 \cdot \text{Var(PL)} + (c_3)^2 \cdot \text{Var(PW)} = 4.573$

^ theory for INDEPENDENT variables says that these variances should be the same...

combin(4,2) = 6 **< figuring the number of pairwise graphs 4 variables two at a time**



^ important COVARIATION is noticed between some pairs of variables here!

So this accounts for why the variance of the Linear Combination does not match the sum of the individual variances...

Using Simulated Random Data:

$i := 0 .. 999$ **< index of values in each variable**

$j := 0 .. 2$ **< index for three variables**

$c := \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix}$ **< contrast matrix**

$X^{\langle 0 \rangle} := rnorm(1000, 5, 10)$

$X^{\langle 1 \rangle} := rnorm(1000, 2, 5)$

$X^{\langle 2 \rangle} := rnorm(1000, 3, 7)$

$LC_i := \sum_j c_j \cdot \left(X^{\langle j \rangle}\right)_i$

^ the linear contrast

$Var\left(X^{\langle 0 \rangle}\right) = 97.0097$

$Var\left(X^{\langle 1 \rangle}\right) = 23.8932$

$Var\left(X^{\langle 2 \rangle}\right) = 51.3127$

$Var(LC) = 236.3934$

X =

	0	1	2
0	0.6103	5.1717	4.2218
1	-1.7941	3.4605	6.9156
2	0.2671	2.7073	-3.8088
3	-4.5147	3.7385	7.9945
4	-11.8568	0.0829	8.4299
5	5.4353	1.6503	4.8077
6	3.7937	-2.819	6.8711
7	10.5643	-0.7961	-1.4344
8	26.9179	-8.2845	13.065
9	13.0873	1.1709	1.4187
10	14.8514	2.521	9.9579
11	13.6223	5.5577	5.9813
12	14.1557	-4.118	0.5645
13	11.73	9.679	4.251
14	-5.4431	1.5269	7.7743
15	5.6908	-2.2593	4.2523

LC =

	0
0	6.7319
1	-1.7888
2	9.4905
3	-5.0321
4	-20.1211
5	3.9282
6	-8.7155
7	10.4065
8	-2.7163
9	14.0104
10	9.9356
11	18.7565
12	5.3552
13	26.837
14	-10.1637
15	-3.0801

$Var(LC) = 236.3934$

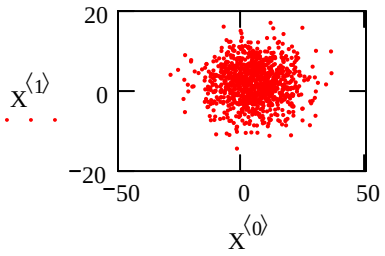
^ variance calculated from this SAMPLE of linear contrasts directly

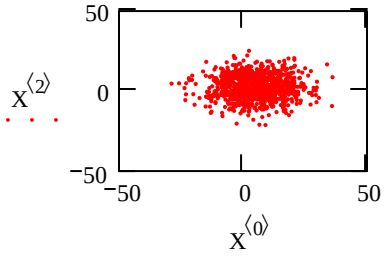
$\sum_j \left(c_j\right)^2 \cdot Var\left(X^{\langle j \rangle}\right) = 243.8951$

^ theoretical variance for INDEPENDENT POPULATIONS calculated by Rosner Eq 5.9 p. 141

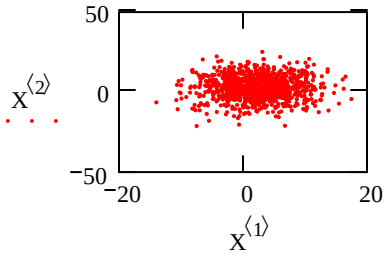
^ These are closer, although you can still see a difference. The SAMPLE of 1000 random points for each variable X_j still has *some* unintentional variable dependence.

$combin(3, 2) = 3$ **< figuring the number of pairwise graphs 3 variables two at a time**





These graphs *look* random ... >



Assessing Covariance & Correlation of Variables

ORIGIN ≡ 0

When bivariate plots or other diagnostic techniques indicate dependence between variables, it is useful to have quantities describing this dependence. Covariance and Correlation are two such quantities that have an important relationship.

Again, let's grab some familiar data:

iris := READPRN("c:/2007BiostatsData/iris.txt")

SL := $iris^{\langle 1 \rangle}$

SW := $iris^{\langle 2 \rangle}$

PL := $iris^{\langle 3 \rangle}$

PW := $iris^{\langle 4 \rangle}$

n := length(SL) n = 150

We have already seen in pairwise graphs that some variable pairs show dependence...

Covariance:

As with mean and variance, covariance may be determined in terms of the population or a specific sample. In practical terms, we are almost always calculating values for samples, so that's what we will do here...

i := 0 .. n − 1 **< index for values in the variables**

$$CV_{SLPL} := \sum_i \frac{\left(SL_i - mean(SL)\right) \cdot \left(PL_i - mean(PL)\right)}{n - 1} \qquad CV_{SLPL} = 1.2743$$

^ sample covariance is the sum of "cross-products" divided by (n-1). The reason for using (n-1) instead of (n) for the SAMPLE is the same reason used for variance.

Prototype for MathCad's built-in covariance function:

cvar(SL, PL) = 1.2658 $\frac{n}{n-1} \cdot cvar(SL, PL) = 1.2743$

^ must correct built-in function for SAMPLE using (n-1)

^ built-in function calculates covariance for POPULATION using (n)

Unfortunately, there is no corresponding built-in function for SAMPLE in MathCad. We'll just have to do it ourselves ...

Correlation:

i := 0 .. n − 1 **< index for values in the variables**

$s_{SL} := \sqrt{Var(SL)}$

$s_{PL} := \sqrt{Var(PL)}$ **< SAMPLE standard deviations**

$$COR_{SLPL} := \frac{CV_{SLPL}}{s_{SL} \cdot s_{PL}} \qquad COR_{SLPL} = 0.8718$$

^ Correlation is calculated as the covariance between two variables divided by the product of their individual standard deviations.

Prototype for MathCad's built-in correlation function:

$\text{corr}(SL, PL) = 0.8718$

$\sigma_{SL} := \sqrt{\text{var}(SL)}$
$\sigma_{PL} := \sqrt{\text{var}(PL)}$ **< POPULATION standard deviations using (n)**

$\frac{\text{cvar}(SL, PL)}{\sigma_{SL} \cdot \sigma_{PL}} = 0.8718$ **< POPULATION correlation using (n)**

$\frac{\text{cvar}(SL, PL)}{s_{SL} \cdot s_{PL}} = 0.8659$ **< WRONG calculation using (n) for covariance but using (n-1) for individual standard deviations**

$\frac{\left(\frac{n}{n-1}\right) \cdot \text{cvar}(SL, PL)}{\sigma_{SL} \cdot \sigma_{PL}} = 0.8776$ **< WRONG calculation using (n-1) for covariance but using (n) for individual standard deviations**

$\frac{\left(\frac{n}{n-1}\right) \cdot \text{cvar}(SL, PL)}{s_{SL} \cdot s_{PL}} = 0.8718$ **< SAMPLE correlation using (n-1).**

Note that when correctly calculated, the SAMPLE and POPULATION correlations are the same!

Effect of Standardizing Data:

Sample variables are often standardized creating a new variables for the POPULATION~N(0,1):

$Z_{SL_i} := \frac{SL_i - \text{mean}(SL)}{s_{SL}}$ $\text{mean}(Z_{SL}) = 0$ $\text{Var}(SL) = 0.6857$

$Z_{PL_i} := \frac{PL_i - \text{mean}(PL)}{s_{PL}}$ $\text{mean}(Z_{PL}) = 0$ $\text{Var}(PL) = 3.1163$

$sZ_{SL} := \sqrt{\text{Var}(SL)}$
$sZ_{PL} := \sqrt{\text{Var}(PL)}$ **< SAMPLE standard deviations for the Standardized variables**

Covariance:

$CVZ_{SLPL} := \sum_i \frac{\left(Z_{SL_i} - \text{mean}(Z_{SL})\right) \cdot \left(Z_{PL_i} - \text{mean}(Z_{PL})\right)}{n - 1}$ $CVZ_{SLPL} = 0.8718$

^ same as Correlation of the *unstandardized* variables!

Effect of Variable Dependence on variance of Linear Combinations:

We won't spend a lot of effort on this here, but for two variables (Rosner Eq 5.11. p. 144):

$$c := \begin{pmatrix} .75 \\ -1.5 \end{pmatrix}$$ **< contrast matrix**

$L_i := c_0 \cdot SL_i + c_1 \cdot PL_i$ **< making the linear combination**

$Var(SL) = 0.6857$

$Var(PL) = 3.1163$

$\frac{n}{n-1} \cdot cvar(SL, PL) = 1.2743$

$Var(L) = 4.5301 \qquad (c_0)^2 \cdot Var(SL) + (c_1)^2 \cdot Var(PL) + 2 \cdot c_0 \cdot c_1 \cdot \left(\frac{n}{n-1} \cdot cvar(SL, PL)\right) = 4.5301$

^ variance calculated using Rossner Eq 5.11, p. 144.

^ variance of the linear combination calculated directly

Here the dependence between variables SL and PL are taken into account, so the calculations based on original variables and Linear Combination now match

Normal Approximations for Discrete Distributions

ORIGIN ≡ 0

The Normal Distribution is very commonly used to approximate discrete Binomial and Poisson distributions when calculation of the latter become problematic. To use these approximations, it is important to see that general boundary conditions involving size of the distribution are met. Also, the approximations are best when (1/2) modifiers to specific cut-offs are used.

Approximating the Binomial Distribution:

Parameters of the Binomial Distribution:
n = total number of things or trials
k = number of "successes"
p = probability of "success"
q = probability of "failure" = (1-p)

Boundary condition:

To be valid, sample variance = $npq \geq 5$

Rosner Example 5.33, p. 147:

$n := 25$

$p := 0.4$ **< Parameters of the binomal distribution**

$q := 1 - p \quad q = 0.6$

$n \cdot p \cdot q = 6$ **< variance boundary condition is met**

$n \cdot p = 10$ **< mean**

Problem: find the Probability P(X=k) for $7 \leq k \leq 12$

Binomial calculation using cumulative binomial function:

$pbinom(12, n, p) = 0.8462$

$pbinom(6, n, p) = 0.0736$ **< cumulative probabilities for each cut-off**

$pbinom(12, n, p) - pbinom(6, n, p) = 0.7727$ **< subtracting the cumulative probabilities remembering that we want to include k=7**

Normal approximation using the cumulative Normal function:

$pnorm\left(12.5, n \cdot p, \sqrt{n \cdot p \cdot q}\right) = 0.8463$

$pnorm\left(6.5, n \cdot p, \sqrt{n \cdot p \cdot q}\right) = 0.0765$ **< cumulative probabilities for each cut-off**

$pnorm\left(12.5, n \cdot p, \sqrt{n \cdot p \cdot q}\right) - pnorm\left(6.5, n \cdot p, \sqrt{n \cdot p \cdot q}\right) = 0.7698$

^ subtracting the cumulative probabilities

Approximating the Poisson Distribution:

Parameters of the Poission distribution:

μ = the expected number of events over an interval of time t
λ = the expected number of events over unit time
$\mu = \lambda t$

Boundary condition:

To be valid, sample variance = $\mu \geq 10$

Rosner Example 5.36, p. 147:

$\lambda := 0.1$

$t := 100$ **< t = A here...**

$\mu := \lambda \cdot t$ $\mu = 10$ **< mean = variance boundary condition is met**

Problem: find the Probability P(X=k) for $k \geq 20$

Poisson calculation using cumulative Poisson function:

$\text{ppois}(19, \mu) = 0.9965$ **< cumulative probabiliy for k < 20**

$1 - \text{ppois}(19, \mu) = 0.0035$ **< cumulative probability for what's left i.e.,** $k \geq 20$

Normal approximation using cumulative Normal function:

$\text{pnorm}(19.5, \mu, \sqrt{\mu}) = 0.9987$ **< cumulative probabiliy for k < 20**

$1 - \text{pnorm}(19.5, \mu, \sqrt{\mu}) = 0.0013$ **< cumulative probability for what's left i.e.,** $k \geq 20$

Normal Distribution - Prototyping Examples from Rosner text

ORIGIN ≡ 0

Example 5.11, p. 131:

$\mu := 0$ **< mean**

$\sigma := 1 \qquad \sigma^2 = 1$ **< standard deviation & variance**

$X := 1.96$ **< critical value to look up in table or use in cumulative function**

$pnorm(X, \mu, \sigma) = 0.975$ **< Note that MathCad requires input of Standard Deviation σ here. Other electronic functions may require Variance σ^2...**

$X := 1$

$pnorm(X, \mu, \sigma) = 0.8413$

BE SURE YOU CAN DO THIS FROM TABLE 3 IN THE APPENDIX ALSO!

Example 5.12, p. 131:

$X := -1.96$

$pnorm(X, \mu, \sigma) = 0.025$

$X := 1.96$ **< This shows that Φ(-X) = 1 - Φ(X)**

$pnorm(X, \mu, \sigma) = 0.975$

$1 - pnorm(X, \mu, \sigma) = 0.025$

Example 5.13, p. 132:

$\mu := 0 \qquad \sigma := 1 \qquad \sigma^2 = 1$

Problem: Compute P(-1<X<1.5):

$pnorm(1.5, \mu, \sigma) = 0.9332$

$pnorm(-1, \mu, \sigma) = 0.1587$

$pnorm(1.5, \mu, \sigma) - pnorm(-1, \mu, \sigma) = 0.7745$ **< Remember these are cumulative probabilites**

Example 5.14, p. 132:

$\mu := 0 \qquad \sigma := 1$

Problem: Compute P(X<-1.5):

$pnorm(-1.5, \mu, \sigma) = 0.0668$

$1 - pnorm(1.5, \mu, \sigma) = 0.0668$ **< Note that the N(0,1) distribution is symmetric but the cumulative distribution is not.**

Example 5.15, p. 133:

$\mu := 0 \qquad \sigma := 1 \qquad \sigma^2 = 1$

Problem: Compute P(-1.5<X<1.5):

$pnorm(1.5, \mu, \sigma) - pnorm(-1.5, \mu, \sigma) = 0.8664$

Example 5.16, p. 133:

$\mu := 0 \qquad \sigma := 1 \qquad \sigma^2 = 1$

Problem: Compute P(0<X<1.45):

$pnorm(1.45, \mu, \sigma) - pnorm(0, \mu, \sigma) = 0.4265$

Example 5.17, p. 133:

$\mu := 0 \quad \sigma := 1 \quad \sigma^2 = 1$

Problem: Compute P(X<2.824):

$\text{pnorm}(2.824, \mu, \sigma) = 0.9976$

Example 5.18, p. 134:

$\mu := 0 \quad \sigma := 1 \quad \sigma^2 = 1$

Problem: Compute Z where P(Z) = 0.975, P(Z) = 0.95, P(Z) = .5 & P(Z) = 0.025:

$\text{qnorm}(0.975, \mu, \sigma) = 1.96$

$\text{qnorm}(0.95, \mu, \sigma) = 1.6449$

$\text{qnorm}(0.5, \mu, \sigma) = 0$

$\text{qnorm}(0.025, \mu, \sigma) = -1.96$

< the qnorm() function is the inverse of the cumulative probability function pnorm(). Most software packages have these functions built in.
However, BE SURE YOU CAN READ TABLE 3 BACKWARDS WHEN NECESSARY

Example 5.19, p. 133:

$\mu := 0 \quad \sigma := 1 \quad \sigma^2 = 1$

Problem: Compute Z where P(Z) < 0.85:

$\text{qnorm}(0.85, \mu, \sigma) = 1.0364$

Example 5.20, p. 135-137:

$\mu := 80 \quad \sigma := \sqrt{144} \quad \sigma = 12 \quad \sigma^2 = 144$

Problem: Compute P(90<X<100) for ~N(μ,σ^2):

Evaluated directly:

$\text{pnorm}(90, \mu, \sigma) = 0.7977$

$\text{pnorm}(100, \mu, \sigma) = 0.9522$

$\text{pnorm}(100, \mu, \sigma) - \text{pnorm}(90, \mu, \sigma) = 0.1545$

Evaluated ~N(0,1) following standardization:

$\text{pnorm}\left(\frac{90 - \mu}{\sigma}, 0, 1\right) = 0.7977$

$\text{pnorm}\left(\frac{100 - \mu}{\sigma}, 0, 1\right) = 0.9522$

$\text{pnorm}\left(\frac{100 - \mu}{\sigma}, 0, 1\right) - \text{pnorm}\left(\frac{90 - \mu}{\sigma}, 0, 1\right) = 0.1545$

< Note that standardization allows use of Table 3 whereas direct computation must be done with a computer-based function...

Example 5.21, p. 137:

$\mu := 8 \quad \sigma := 2 \quad \sigma = 2 \quad \sigma^2 = 4$

Problem: Compute P(12<X) for ~N(8,4):

$\text{pnorm}\left(\frac{12 - \mu}{\sigma}, 0, 1\right) = 0.9772$ **< for the cumulative probabily after standardization**

$1 - \text{pnorm}\left(\frac{12 - \mu}{\sigma}, 0, 1\right) = 0.0228$ **< for the remainder X > 12**

Also directly:

$1 - \text{pnorm}(12, \mu, \sigma) = 0.0228$

Example 5.22, p. 137:

$\mu := 75 \quad \sigma := 17 \qquad \sigma = 17 \qquad \sigma^2 = 289$

Problem: Compute P(X<40) for ~N(75,289):

$$\text{pnorm}\left(\frac{40-\mu}{\sigma}, 0, 1\right) = 0.0198$$ **< for the cumulative probabily after standardization**

$$1 - \text{pnorm}\left(\frac{\mu-40}{\sigma}, 0, 1\right) = 0.0198$$ **< Looking at the other tail of the distribution, i.e., P(-X) = 1-P(X)**

Also directly:

$\text{pnorm}(40, \mu, \sigma) = 0.0198$

Example 5.23, p. 138:

$\mu := 16 \quad \sigma := 3 \qquad \sigma = 3 \qquad \sigma^2 = 9$

Problem: Compute P(12<X<20) for ~N(16,9):

$$\text{pnorm}\left(\frac{20-\mu}{\sigma}, 0, 1\right) - \text{pnorm}\left(\frac{12-\mu}{\sigma}, 0, 1\right) = 0.8176$$ **< straight calculation**

$$\text{pnorm}\left(\frac{20.5-\mu}{\sigma}, 0, 1\right) - \text{pnorm}\left(\frac{11.5-\mu}{\sigma}, 0, 1\right) = 0.8664$$ **< with modification to incorporate "continuity correction" indicating uncertainty in measuring...**

Also directly:

$\text{pnorm}(20, \mu, \sigma) - \text{pnorm}(12, \mu, \sigma) = 0.8176$

$\text{pnorm}(20.5, \mu, \sigma) - \text{pnorm}(11.5, \mu, \sigma) = 0.8664$

Example 5.24, p. 139:

$\mu := 80 \quad \sigma := \sqrt{144} \quad \sigma = 12 \qquad \sigma^2 = 144$

Problem: Compute X where P(X) = 0.05, and X where P(X) = 0.95:

$Z_{05} := \text{qnorm}(0.05, 0, 1)$

$Z_{95} := \text{qnorm}(0.95, 0, 1)$

< calculating percentiles based on ~N(0,1)

$X_{05} := \sigma \cdot Z_{05} + \mu \qquad X_{05} = 60.2618$

$X_{95} := \sigma \cdot Z_{95} + \mu \qquad X_{95} = 99.7382$

< Calculating X from standardized Z: $X_i = \sigma Z_i + \mu$

Also directly:

$\text{qnorm}(0.05, \mu, \sigma) = 60.2618$

$\text{qnorm}(0.95, \mu, \sigma) = 99.7382$

< letting the built-in function do all the work. Note, however, that this must be done on the computer as Table 3 doesn't apply...

Assignment for Week 5

This week we can begin statistical data analysis more-or-less for real. In our reading, we have seen how to construct confidence intervals for the parameters of populations assuming, of course, that our data sample comes from the distribution characterizing that population. In lab, let's concentrate on how to do this with the **iris** dataset.

The famous data set on the genus *Iris* involves four measurements (columns) for 150 individuals that the author (Anderson) originally thought to belong to three species (last column). We can use these measurements to assess whether the species he identified can be distinguished morphometrically (i.e., by differences in the mean of their measurements). Of course, individuals in a population such as a species naturally show variance, so mean values of each variable for each species must be judged accordingly. Constructing confidence intervals allows us to circumscribe the location of the population mean for each of the four variables and to see if the species differ in some way or completely overlap.

So, this week, fire up R and try the following tasks. Note also that I have posted R documentation for you and some helpful hints on our website.

1. Find the **iris** data set in R and print out a copy for reference as you work on this problem.
2. Construct X, Y plots of the variables to see how they are distributed. Look for breaks in the data and interpret what you see.
3. For each species, construct a histogram of each variable to assess normality of the data. Again, interpret what you see.
4. Now, for each species, construct Q-Q plots and compare. Are the data Normally distributed? How can you tell?
5. For Sepal.Length of Species *Iris setosa*, construct a 95% confidence interval of the mean. Compare your results with *2007 Biostatistics 18* and confirm your prototype.
6. Now construct a 99% confidence interval for the same data using your calculations. How does this change in α affect the width of the confidence interval?
7. Finally, use R's built-in t.test() function to calculate 95% and 99% confidence intervals for each species over all four variables.
8. Given these confidence intervals, what evidence can you cite supporting or rejecting the presence of multiple species?

Q-Q Plot with Theoretical Distribution

Description

Quantile-Quantile plot of a sample and a theoretical distribution

Usage

```
qqmath(x, data, ...)

## S3 method for class 'formula':
qqmath(x,
       data,
       allow.multiple = is.null(groups) || outer,
       outer = !is.null(groups),
       distribution = qnorm,
       f.value = NULL,
       auto.key = FALSE,
       aspect = "fill",
       panel = "panel.qqmath",
       prepanel = NULL,
       scales, strip, groups,
       xlab, xlim, ylab, ylim,
       drop.unused.levels = lattice.getOption("drop.unused.levels"),
       ...,
       default.scales = list(),
       subscripts,
       subset)
## S3 method for class 'numeric':
qqmath(x, data, ylab, ...)
```

Arguments

`x`	The object on which method dispatch is carried out. For the `"formula"` method, a formula of the form `~ x | g1 * g2 * ...`, where `x` must be a numeric. For the `"numeric"` method, a numeric vector.
`data`	For the `formula` method, an optional data frame in which variables in the formula (as well as `groups` and `subset`, if any) are to be evaluated. Usualll ignored with a warning in other methods.
`distribution`	a quantile function that takes a vector of probabilities as argument and produces the corresponding quantiles. Possible values are `qnorm`, `qunif` etc. Distributions with other required arguments need to be passed in as user defined functions.

`f.value`	optional numeric vector of probabilities, quantiles corresponding to which should be plotted. Can also be a function of a single integer (representing sample size) that returns such a numeric vector. The typical value for this argument is the function `ppoints`, which is also the S-PLUS default. If specified, the probabilities generated by this function is used for the plotted quantiles, using the `quantile` function for the sample, and the function specified as the `distribution` argument for the theoretical distribution. `f.value` defaults to `NULL`, which has the effect of using `ppoints` for the quantiles of the theoretical distribution, but the exact data values for the sample. This is similar to what happens for `qqnorm`, but different from the S-PLUS default of `f.value=ppoints`. For large `x`, this argument can be useful in plotting a smaller set of quantiles, which is usually enough to capture the pattern.
`panel`	The panel function to be used. Unlike in older versions, the default panel function does most of the actual computations and has support for grouping. See `panel.qqmath` for details.
`allow.multiple, outer, auto.key, aspect, prepanel, scales, strip, groups, xlab, xlim, ylab, ylim, drop.unused.levels, default.scales, subscripts, subset`	See `xyplot`
`...`	Further arguments. See corresponding entry in `xyplot` for non-trivial details.

Details

`qqmath` produces a Q-Q plot of the given sample and a theoretical distribution. The default behaviour of `qqmath` is different from the corresponding S-PLUS function, but is similar to `qqnorm`. See the entry for `f.value` for specifics.

The implementation details are also different from S-PLUS. In particular, all the important calculations are done by the panel (and prepanel function) and not `qqmath` itself. In fact, both the arguments `distribution` and `f.value` are passed unchanged to the panel and prepanel function. This allows, among other things, display of grouped Q-Q plots, which are often useful. See the help page for `panel.qqmath` for further details.

This and all other high level Trellis functions have several arguments in common. These are extensively documented only in the help page for `xyplot`, which should be consulted to learn more detailed usage.

Value

An object of class `"trellis"`. The `update` method can be used to update components of the object and the `print` method (usually called by default) will plot it on an appropriate plotting device.

Author(s)

Deepayan Sarkar Deepayan.Sarkar@R-project.org

See Also

`xyplot`, `panel.qqmath`, `panel.qqmathline`, `prepanel.qqmathline`, `Lattice`, `quantile`

Examples

```
qqmath(~ rnorm(100), distribution = function(p) qt(p, df = 10))
qqmath(~ height | voice.part, aspect = "xy", data = singer,
       prepanel = prepanel.qqmathline,
       panel = function(x, ...) {
          panel.qqmathline(x, ...)
          panel.qqmath(x, ...)
       })
vp.comb <-
    factor(sapply(strsplit(as.character(singer$voice.part), split = " "),
                  "[", 1),
           levels = c("Bass", "Tenor", "Alto", "Soprano"))
vp.group <-
    factor(sapply(strsplit(as.character(singer$voice.part), split = " "),
                  "[", 2))
qqmath(~ height | vp.comb, data = singer,
       groups = vp.group, auto.key = list(space = "right"),
       aspect = "xy",
       prepanel = prepanel.qqmathline,
       panel = function(x, ...) {
          panel.qqmathline(x, ...)
          panel.qqmath(x, ...)
       })
```

Object Summaries

Description

`summary` is a generic function used to produce result summaries of the results of various model fitting functions. The function invokes particular methods which depend on the class of the first argument.

Usage

```
summary(object, ...)

## Default S3 method:
summary(object, ..., digits = max(3, getOption("digits")-3))
## S3 method for class 'data.frame':
summary(object, maxsum = 7,
       digits = max(3, getOption("digits")-3), ...)

## S3 method for class 'factor':
summary(object, maxsum = 100, ...)

## S3 method for class 'matrix':
summary(object, ...)
```

Arguments

`object`	an object for which a summary is desired.
`maxsum`	integer, indicating how many levels should be shown for factors.
`digits`	integer, used for number formatting with signif() (for `summary.default`) or format() (for `summary.data.frame`).
`...`	additional arguments affecting the summary produced.

Details

For factors, the frequency of the first `maxsum - 1` most frequent levels is shown, where the less frequent levels are summarized in `"(Others)"` (resulting in `maxsum` frequencies).

The functions `summary.lm` and `summary.glm` are examples of particular methods which summarise the results produced by lm and glm.

Value

The form of the value returned by `summary` depends on the class of its argument. See the documentation of the particular methods for details of what is produced by that method.

References

Chambers, J. M. and Hastie, T. J. (1992) *Statistical Models in S*. Wadsworth & Brooks/Cole.

See Also

anova, summary.glm, summary.lm.

Examples

```
summary(attenu, digits = 4) #-> summary.data.frame(...), default precision
summary(attenu $ station, maxsum = 20) #-> summary.factor(...)

lst <- unclass(attenu$station) > 20 # logical with NAs
## summary.default() for logicals -- different from *.factor:
summary(lst)
summary(as.factor(lst))
```

Quantile-Quantile Plots

Description

`qqnorm` is a generic function the default method of which produces a normal QQ plot of the values in `y`. `qqline` adds a line to a normal quantile-quantile plot which passes through the first and third quartiles.

`qqplot` produces a QQ plot of two datasets.

Graphical parameters may be given as arguments to `qqnorm`, `qqplot` and `qqline`.

Usage

```
qqnorm(y, ...)
## Default S3 method:
qqnorm(y, ylim, main = "Normal Q-Q Plot",
       xlab = "Theoretical Quantiles", ylab = "Sample Quantiles",
       plot.it = TRUE, datax = FALSE, ...)

qqline(y, datax = FALSE, ...)

qqplot(x, y, plot.it = TRUE, xlab = deparse(substitute(x)),
       ylab = deparse(substitute(y)), ...)
```

Arguments

`x`	The first sample for `qqplot`.
`y`	The second or only data sample.
`xlab, ylab, main`	plot labels. The `xlab` and `ylab` refer to the y and x axes respectively if `datax = TRUE`.
`plot.it`	logical. Should the result be plotted?
`datax`	logical. Should data values be on the x-axis?
`ylim, ...`	graphical parameters.

Value

For `qqnorm` and `qqplot`, a list with components

`x` The x coordinates of the points that were/would be plotted

`y` The original y vector, i.e., the corresponding y coordinates *including* NAs.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

See Also

`ppoints`, used by `qqnorm` to generate approximations to expected order statistics for a normal distribution.

Examples

```
y <- rt(200, df = 5)
qqnorm(y); qqline(y, col = 2)
qqplot(y, rt(300, df = 5))

qqnorm(precip, ylab = "Precipitation [in/yr] for 70 US cities")
```

Student's t-Test

Description

Performs one and two sample t-tests on vectors of data.

Usage

```
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula':
t.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	a (non-empty) numeric vector of data values.
`y`	an optional (non-empty) numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`. You can specify just the initial letter.
`mu`	a number indicating the true value of the mean (or difference in means if you are performing a two sample test).
`paired`	a logical indicating whether you want a paired t-test.
`var.equal`	a logical variable indicating whether to treat the two variances as being equal. If `TRUE` then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
`conf.level`	confidence level of the interval.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is a numeric variable giving the data values and `rhs` a factor with two levels giving the corresponding groups.
`data`	an optional matrix or data frame (or similar: see model.frame) containing the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
`subset`	an optional vector specifying a subset of observations to be used.
`na.action`	a function which indicates what should happen when the data contain `NA`s.

	Defaults to `getOption("na.action")`.
`...`	further arguments to be passed to or from methods.

Details

The formula interface is only applicable for the 2-sample tests.

`alternative = "greater"` is the alternative that `x` has a larger mean than `y`.

If `paired` is `TRUE` then both `x` and `y` must be specified and they must be the same length. Missing values are removed (in pairs if `paired` is `TRUE`). If `var.equal` is `TRUE` then the pooled estimate of the variance is used. By default, if `var.equal` is `FALSE` then the variance is estimated separately for both groups and the Welch modification to the degrees of freedom is used.

If the input data are effectively constant (compared to the larger of the two means) an error is generated.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value of the t-statistic.
`parameter`	the degrees of freedom for the t-statistic.
`p.value`	the p-value for the test.
`conf.int`	a confidence interval for the mean appropriate to the specified alternative hypothesis.
`estimate`	the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.
`null.value`	the specified hypothesized value of the mean or mean difference depending on whether it was a one-sample test or a two-sample test.
`alternative`	a character string describing the alternative hypothesis.
`method`	a character string indicating what type of t-test was performed.
`data.name`	a character string giving the name(s) of the data.

See Also

prop.test

Examples

```
t.test(1:10,y=c(7:20))      # P = .00001855
t.test(1:10,y=c(7:20, 200)) # P = .1245    -- NOT significant anymore
```

```
## Classical example: Student's sleep data
plot(extra ~ group, data = sleep)
## Traditional interface
with(sleep, t.test(extra[group == 1], extra[group == 2]))
## Formula interface
t.test(extra ~ group, data = sleep)
```

Point and Interval Estimation for the Normal Distribution

ORIGIN ≡ 0

Given the general setup in statistics between random variable X and the probability P(X) governed by a Probability Density Function such as the Normal Distribution, Binomial Distribution, etc., one typically uses specific random samples to estimate the population parameters. Estimation of this sort takes on additional error over direct knowledge of the population parameters. However, one rarely knows them.

For the Normal Distribution, the *population parameters* are:

μ = **population mean**

σ^2 = **population variance**

From our *sample,* we have the analogous calculations termed *point estimates*:

X_{bar} = **sample mean**

s^2 = **sample variance**

Different kinds of statistical theory underlies these estimates generally allowing them to be categorized in one of two ways:

- "minimum variance", also known as "least squares minimum" "unbiased" or "Normal theory" estimators, and

- "maximum liklihood" estimators.

How to calculate estimators of these two types is generally beyond the scope of introductory statistics courses, although Rosner can't resist showing you an example of one derivation using "maximum liklihood" estimators for p in the binomal distribution on p. 203. It is nice to see it, but don't worry too much about details at this point.

The important thing to remember is that the two methods of estimation sometimes but not always yield the same point estimators. The point estimators, then feed into specific statistical techniques. Thus, it is sometimes important to know which estimator is associated with a particular technique so as not mix approaches. Generally, maximum liklihood estimators, based on newer theory, are specifically indicated as such (often using 'hat' notation).

In the case estimating parameters for the Normal Distribution, X_{bar} is the point estimate for μ under both estimation theories. However s^2 sum of squares with (n-1) as divisor is the point estimate using "unbiased" theory whereas σ^2_{hat} with same sum of sqares but using (n) as divisor is the point estimate using "maximum liklihood" theory. Confusing, yes, but now that you know the difference not all that bad...

Estimating error on point estimates of the mean:

Although X_{bar} is our Normal theory estimate of population parameter μ based on a single sample, one might readily expect X_{bar} to differ from sample to sample, and it does. We thus need to estimate how far X_{bar} will vary from sample to sample. Multiply collected *means* differ from each other much less than individual sample values X will. The relationship is called the *"standard variance of the mean"*:

Standard Variance of the Mean = sample variance/n

or

Standard Error of the Mean = sample standard deviation / $\sqrt{n}$

Central Limit Theorem:

This result is one of the reasons why Normal theory, and the Normal Distribution underlie much of "parametric" statistics. See Rosner Eq. 6.3 p. 184 for a formal definition. It says that although the populations from which random variable X are drawn may not necessarily be normally distributed, the population of means derived by replicate sampling will be normally distributed. This result allows us to use the Normal Distribution with parameters μ, σ^2 estimated respectively by X_{bar} and s^2 (or occasionally σ^2_{hat}) to estimate probabilities of *means* P(X) for various values of X.

Statistics evaluating location of the mean:

Rosner Eq. 6.4 p. 187 gives the usual approach to estimating the difference between X_{bar} of a sample and μ of a population. It involves standardizing the random variable $X_{bar} - \mu$ which measures the difference between sample and population means:

$$Z := \frac{X_{bar} - \mu}{\frac{\sigma}{\sqrt{n}}}$$

< Note use of standardization of the variable by $\sigma/\sqrt{n}$

If somehow we know the population parameter σ then we can resort directly to the standardized Normal Distribution ~N(0,1) to calculate probabilities P(Z). However, in real life situations, σ is not known and we must estimate σ by s. When we do this, the analogous variable t:

$$t := \frac{X_{bar} - \mu}{\frac{s}{\sqrt{n}}}$$

< Same standardizing approach but using s instead of σ

is no longer Normally distributed. Instead, we resort to a new probability density function, known as "Student's t" to calculate P(t) given t. Student's t is a commonly employed statistical function ranking high in importance along with the chi-square distribution (χ^2) and the F distribution.

Confidence Intervals:

Confidence intervals are are statements of ranges of X around X_{bar} within which μ is expected to reside over a certain fraction of samples. This fraction is set by specifying a confidence limit α.

Let's calculate this from a pseudo-random example:

$X := rnorm(100, 50, \sqrt{100})$ **< here in fact *we know* μ=50 and σ^2 = 100**

$n := length(X)$ $n = 100$

$\mu := 50$ $\sigma := \sqrt{100}$

$X_{bar} := mean(X)$ $X_{bar} = 48.4955$

$s := \sqrt{Var(X)}$ $s^2 = 96.4487$

< we can also *pretend* that we *don't know* the population parameters and must use sample mean and variance instead as one usually would with real data.

Calculation of Confidence Intervals:

$\alpha := 0.05$

$1 - \alpha = 0.95$

< We choose a limit probability allowing μ to reside outside the range of X around X_{bar} (1-α) X 100 percent of the time...

^ since the Normal Distribution and the t distribution are both symmetrical, there are equal- sized tails for each distribution above or below which μ will fall half of the time. Each tail therefore has $\alpha/2$ probability.

This is commonly known as the Two-Tail case...

To calculate probabilities, we employ the commonly implemented cumulative 'p' and 'q' functions for the Normal distribution seen in statistical software:

$\alpha := 0.05$ **< confidence limit that we must set explicitly each time**

If μ and σ are known:

$\mu = 50$ $\sigma = 10$

$L := qnorm\left(\frac{\alpha}{2}, 0, 1\right)$ $L = -1.96$ $\frac{\alpha}{2} = 0.025$ **< lower limit of N(0,1) for $\alpha/2$**

$U := qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right)$ $U = 1.96$ $1 - \frac{\alpha}{2} = 0.975$ **< upper limit of N(0,1) for $\alpha/2$**

$CI := \left(\mu + \frac{\sigma \cdot L}{\sqrt{n}} \quad \mu + \frac{\sigma \cdot U}{\sqrt{n}}\right)$ **< calculating Confidence Interval using population μ and σ see Rosner Eq. 6.4. p. 187 Note here that I calculated each tail explicitly so I *added* both L and U to determine the CI.**

$CI = (48.04 \quad 51.96)$

If μ and σ must be estimated by sample X_{bar} and s:

$X_{bar} = 48.4955$ $s = 9.8208$

$df := n - 1$ $df = 99$ **< single parameter of Student's t distribution called "degrees of freedom"**

$L := qt\left(\frac{\alpha}{2}, df\right)$ $L = -1.9842$ $\frac{\alpha}{2} = 0.025$

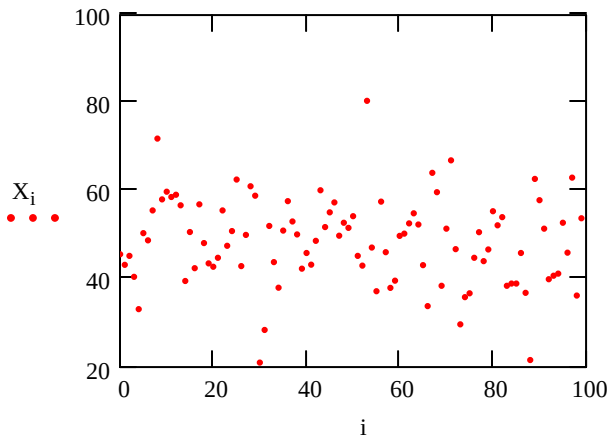
$U := qt\left(1 - \frac{\alpha}{2}, df\right)$ $U = 1.9842$ $1 - \frac{\alpha}{2} = 0.975$

$CI := \left(X_{bar} + L \cdot \frac{s}{\sqrt{n}} \quad X_{bar} + U \cdot \frac{s}{\sqrt{n}}\right)$ **< calculating Confidence Interval. See Rosner Eq. 6.6, p. 190. Note here that I calculated each tail explicitly so I *added* both L and U to determine the CI. Also note SE of mean measured by the sample quantity $\frac{s}{\sqrt{n}}$**

$CI = (46.5468 \quad 50.4441)$

^ Occasionally we may be unlucky here when our pseudo-random number generator gives us a deviant sample with confidence interval that doesn't include $\mu = 50$ in a sample of 100 X's, but that's the breaks!

$i := 0 .. n$



^ note outlier points influencing X_{bar} here!

$X =$

	0
0	45.6103
1	43.2059
2	45.2671
3	40.4853
4	33.1432
5	50.4353
6	48.7937
7	55.5643
8	71.9179
9	58.0873
10	59.8514
11	58.6223
12	59.1557
13	56.73
14	39.5569
15	50.6908

Confidence Intervals for Mean and Variance of a Normal Distribution

ORIGIN ≡ 0

Calculating confidence intervals on the sample mean and sample variance are important statistical functions. This worksheet shows the calculation of both using our familiar Iris data:

iris := READPRN("c:/2007BiostatsData/iris.txt")

SL := $iris^{\langle 1 \rangle}$ PL := $iris^{\langle 3 \rangle}$

SW := $iris^{\langle 2 \rangle}$ PW := $iris^{\langle 4 \rangle}$

n := length(SL) n = 150

$Xbar_{SL}$:= mean(SL) $Xbar_{SL}$ = 5.8433

$Xbar_{SW}$:= mean(SW) $Xbar_{SW}$ = 3.0573

$Xbar_{PL}$:= mean(PL) $Xbar_{PL}$ = 3.758

$Xbar_{PW}$:= mean(PW) $Xbar_{PW}$ = 1.1993

< calculating sample means

$SD_{SL} := \sqrt{Var(SL)}$ SD_{SL} = 0.8281

$SD_{SW} := \sqrt{Var(SW)}$ SD_{SW} = 0.4359

$SD_{PL} := \sqrt{Var(PL)}$ SD_{PL} = 1.7653

$SD_{PW} := \sqrt{Var(PW)}$ SD_{PW} = 0.7622

< calculating sample standard deviations

$SE_{SL} := \frac{SD_{SL}}{\sqrt{n}}$ SE_{SL} = 0.0676

$SE_{SW} := \frac{SD_{SW}}{\sqrt{n}}$ SE_{SW} = 0.0356

$SE_{PL} := \frac{SD_{PL}}{\sqrt{n}}$ SE_{PL} = 0.1441

$SE_{PW} := \frac{SD_{PW}}{\sqrt{n}}$ SE_{PW} = 0.0622

< standard error of the mean based on sample standard deviation

NOTE: CI's assume underlying Normal distribution for each variable, but Central Limit Theorem provides robust outcome anyway...

Confidence Intervals for mean:

α := 0.05 **< We choose a limit probability...**

$1 - \frac{\alpha}{2} = 0.975$ **< upper limit for tail of the symmetrical t distribution**

df := n − 1 df = 149 **< single parameter of the t distribution called "degrees of freedom"**

$CIm_{SL} := \left(Xbar_{SL} - qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{SL} \quad Xbar_{SL} + qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{SL} \right)$

$CIm_{SW} := \left(Xbar_{SW} - qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{SW} \quad Xbar_{SW} + qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{SW} \right)$

$CIm_{PL} := \left(Xbar_{PL} - qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{PL} \quad Xbar_{PL} + qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{PL} \right)$

$CIm_{PW} := \left(Xbar_{PW} - qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{PW} \quad Xbar_{PW} + qt\left(1 - \frac{\alpha}{2}, df\right) \cdot SE_{PW} \right)$

^ confidence intervals calculated using upper tail of the t distribution only.

Prototype for Confidence Interval of the mean:

$CIm_{SL} = (5.7097 \quad 5.9769)$

$CIm_{SW} = (2.987 \quad 3.1277)$ **< Evaluation of above CI calculations**

$CIm_{PL} = (3.4732 \quad 4.0428)$

$CIm_{PW} = (1.0764 \quad 1.3223)$

SYSTAT Output confirms calculations:

	SEPALLEN	SEPALWID	PETALLEN	PETALWID
N of cases	**150**	**150**	**150**	**150**
Minimum	**4.3000000**	**2.0000000**	**1.0000000**	**0.1000000**
Maximum	**7.9000000**	**4.4000000**	**6.9000000**	**2.5000000**
Mean	**5.8433333**	**3.0573333**	**3.7580000**	**1.1993333**
95% CI Upper	**5.9769342**	**3.1276563**	**4.0428146**	**1.3223134**
95% CI Lower	**5.7097325**	**2.9870103**	**3.4731854**	**1.0763533**
Std. Error	**0.0676113**	**0.0355883**	**0.1441360**	**0.0622364**
Standard Dev	**0.8280661**	**0.4358663**	**1.7652982**	**0.7622377**

Confidence Interval for Variance:

NOTE: CI's assume underlying Normal distribution for each variable...

$Var(SL) = 0.6857$

$Var(SW) = 0.19$ **< sample variances**

$Var(PL) = 3.1163$

$Var(PW) = 0.581$

For variance, this assumption is crucial & sensitive

$\alpha := 0.05$ **< We choose a limit probability...**

$1 - \frac{\alpha}{2} = 0.975$ $\frac{\alpha}{2} = 0.025$ **< upper and lower limits of asymmetrical $\chi 2$ distribution**

$$CIv_{SL} := \left[\frac{(n-1)\cdot Var(SL)}{qchisq\left(1-\frac{\alpha}{2}, df\right)} \quad \frac{(n-1)\cdot Var(SL)}{qchisq\left(\frac{\alpha}{2}, df\right)} \right] \qquad CIv_{SL} = (0.5532 \quad 0.8725)$$

$$CIv_{SW} := \left[\frac{(n-1)\cdot Var(SW)}{qchisq\left(1-\frac{\alpha}{2}, df\right)} \quad \frac{(n-1)\cdot Var(SW)}{qchisq\left(\frac{\alpha}{2}, df\right)} \right] \qquad CIv_{SW} = (0.1533 \quad 0.2417)$$

$$CIv_{PL} := \left[\frac{(n-1)\cdot Var(PL)}{qchisq\left(1-\frac{\alpha}{2}, df\right)} \quad \frac{(n-1)\cdot Var(PL)}{qchisq\left(\frac{\alpha}{2}, df\right)} \right] \qquad CIv_{PL} = (2.5141 \quad 3.9653)$$

$$CIv_{PW} := \left[\frac{(n-1)\cdot Var(PW)}{qchisq\left(1-\frac{\alpha}{2}, df\right)} \quad \frac{(n-1)\cdot Var(PW)}{qchisq\left(\frac{\alpha}{2}, df\right)} \right] \qquad CIv_{PW} = (0.4687 \quad 0.7393)$$

^ I haven't yet found an automated procedure in Systat or another canned statistical package for direct comparison as Prototype. The R program will allow hand calculation in the same way.

Point and Interval Estimation of Discrete Distributions Parameters

ORIGIN ≡ 0

Estimates of expected values and confidence intervals for parameter p of the Binomial Distribution and μ of Poisson Distribution can be made in a way analogous to that seen for μ and σ^2 in the Normal Distribution.

Binomial Distribution:

Making some a sample derived from the Binomial Distribution:

$n := 30$ $p := 0.3$ $m := 100$

$R_B := rbinom(m, n, p)$ $length(R_B) = 100$

	0
0	12
1	10
2	10
3	8
4	13
5	10
6	8
7	7
8	12
9	10
10	9
11	6
12	10
13	11
14	9
15	6

$R_B =$

Point estimate for p:

Remember p is defined as the probability of "success" at each trial, as in the probability of "heads" in the coin flip problem...

$i := 0 .. 99$

$$phat := \frac{\frac{1}{m} \cdot \sum_i R_{B_i}}{n}$$ $phat = 0.3067$ **< point estimate of fraction p is the mean number of heads over sample of size m divided total possible number of heads**

$$SE_p := \sqrt{\frac{phat \cdot (1 - phat)}{n}}$$ $SE_p = 0.0842$ **< Standard error of p remember q=(1-p)**

^ Compare this calculation with Rosner p. 202. On following pages, Rosner shows how the p_{hat} utilized here is the Maximum Liklihood point estimate of p.

Note that p_{hat} is dependent on m = the number of replicates.
If m=1, then p_{hat} becomes the single observation X that you have.

Interval estimate for p:

Confidence Interval using Normal Theory Methods:

Rossner p. 205 gives a rational for the use of this method as long as $n \cdot phat \cdot qhat \geq 5$

$n = 30$ $phat = 0.3067$ $qhat := 1 - phat$ $qhat = 0.6933$

$n \cdot phat \cdot qhat = 6.3787$ **< OK to proceed!**

$\alpha := 0.05$ **< Specify confidence limit**

$1 - \frac{\alpha}{2} = 0.975$ **< upper limit on symmetric Standardized Normal Distribution**

$$CI_{Np} := \left(phat - qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right) \cdot \sqrt{\frac{phat \cdot qhat}{n}} \quad phat + qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right) \sqrt{\frac{phat \cdot qhat}{n}} \right)$$

$CI_{Np} = (0.1417 \quad 0.4717)$ **< Confidence Interval for p based on Normal Theory Methods.**

^ Compare this interval with p_{hat} - the point estimate for p above.

Confidence Interval using Exact Methods:

See Rosner p. 208-209 for his example of calculating Confidence Intervals indirectly with the Cumulative Binomial Probability function. The key to using this method correctly is to first find p_{hat} = the mean value of the sample. Then set cutoffs in the cumulative probability distribution Φ_B (i.e., the 'p' Binomial function in Mathcad & R) that surround the observed mean. Then choose a range of X values that bracket the probability $\alpha/2$ above and below. Values (X) are read from the same position in matrices X1 & X2 as the $\Phi 1(X)$ and $\Phi 2(X)$ at $\alpha/2$ cutoff - i.e., values of X are recovered at appropriate cumulative cutoff probabilities $\Phi_B(X)$.

$n = 30$ $p_{hat} = 0.3067$ $q_{hat} := 1 - p_{hat}$ $q_{hat} = 0.6933$

$\alpha := 0.05$ **< Specify confidence limit**

$\frac{\alpha}{2} = 0.025$ **< upper limit on symmetric Standardized Normal Distribution**

$CI_{Ep} := (0.15 \quad 0.51)$ **< using Table 7 α=0.05 for p_{hat} = 0.3123**

$n = 30$ $mean(R_B) = 9.2$ **< mean of Binomial distribution R_B**

$i := 0 .. 9$

$X1_i := (i + 1) \cdot 0.05$

$X2_i := (i + 1) \cdot 0.02 + 0.4$

< I picked a range of values around where I expected the CI limits in X to fall from table above.

$\Phi 1_i := 1 - pbinom(9, n, X1_i)$

$\Phi 2_i := pbinom(10, n, X2_i)$

< calculating cumulative probabilities with cutoff around mean(R_B)

values (X):

$$X1 = \begin{pmatrix} 0.05 \\ 0.1 \\ 0.15 \\ 0.2 \\ 0.25 \\ 0.3 \\ 0.35 \\ 0.4 \\ 0.45 \\ 0.5 \end{pmatrix} \quad X2 = \begin{pmatrix} 0.42 \\ 0.44 \\ 0.46 \\ 0.48 \\ 0.5 \\ 0.52 \\ 0.54 \\ 0.56 \\ 0.58 \\ 0.6 \end{pmatrix}$$

> lower limit ^ upper limit ^ <

Cumulative Probabilities $\Phi_B(X)$:

$$\Phi 1 = \begin{pmatrix} 1.1615 \times 10^{-6} \\ 0.0005 \\ 0.0097 \\ 0.0611 \\ 0.1966 \\ 0.4112 \\ 0.6425 \\ 0.8237 \\ 0.9306 \\ 0.9786 \end{pmatrix} \quad \Phi 2 = \begin{pmatrix} 0.2201 \\ 0.1604 \\ 0.1126 \\ 0.0761 \\ 0.0494 \\ 0.0307 \\ 0.0183 \\ 0.0104 \\ 0.0056 \\ 0.0029 \end{pmatrix}$$

< lower probability cutoff for CI Find corresponding value in X1

upper probability cutoff for CI > Find corresponding value in X2

^ It would be useful to compare this prototype with confidence intervals for p provided by a canned statistical procedure. So far, I haven't found a program that does this... No doubt, R would allow me to make similar calculations, but that would not suffice as a check on procedure.

Prototype using Exact Binomial Test Function in R:

>RB=c(12,10,10,8,13,10,8,7,12,10,9,6,10,11,9,6,11,8,10,8,6,9,10,12,11,4,13,12,9,11,9,9,5,9,9,8,9,8,-13,8,11,9,9,10,10,9,8,9,8,12,10,9,13,8,12,4,5,6,9,7,12,11,12,8,8,9,5,11,10,10,10,9,10,7,9,7,6,9,6,12,-13,12,9,10,13,10,9,10,11,6,8,9,8,10,14,6,12,7,9,3)

^ values of R_B above cut and pasted into variable RB in R

> binom.test(9,30,p=(mean(RB)/30),alternative="two.sided",conf.level=0.95)

^ binom.test function used...

Exact binomial test

data: 9 and 30
number of successes = 9, number of trials = 30, p-value = 1
alternative hypothesis: true probability of success is not equal to 0.3066667
95 percent confidence interval:
0.1473452 0.4939590 < compare with cutoffs in Φ1 & Φ2 above...
sample estimates:
probability of success
0.3

binom.test {stats} R Documentation
Exact Binomial Test
Description

Performs an exact test of a simple null hypothesis about the probability of success in a Bernoulli experiment.
Usage

```
binom.test(x, n, p = 0.5,
       alternative = c("two.sided", "less", "greater"),
       conf.level = 0.95)
```

Arguments

x	**number of successes, or a vector of length 2 giving the numbers of successes and failures, respectively.**
n	**number of trials; ignored if x has length 2.**
p	**hypothesized probability of success.**
alternative	**indicates the alternative hypothesis and must be one of "two.sided", "greater" or "less". You can specify just the initial letter.**
conf.level	**confidence level for the returned confidence interval.**

Details

Confidence intervals are obtained by a procedure first given in Clopper and Pearson (1934). This guarantees that the confidence level is at least conf.level, but in general does not give the

Poisson Distribution:

Making some a sample derived from the Poisson Distribution:

$\lambda := 4$ $m := 100$

$R_P := rpois(m, \lambda)$

$R_P =$

	0
0	5
1	4
2	6
3	3
4	4
5	8
6	5
7	4
8	5
9	4
10	6
11	6
12	2
13	3
14	2
15	8

Point Estimate for λ:

$\lambda_{bar} := mean(R_P)$ $\lambda_{bar} = 4.24$ **< mean occurrence**

Exact Interval Estimate for λ:

$\alpha := 0.05$ **< Specify confidence limit**

$1 - \alpha = 0.95$ **< α defines the confidence interval**

As with the Binomial Distribution, the Confidence Interval is difficult to calculate explicitly. See Rosner Table 8 in Appendix p. 835. The table requires specifying (1-α) level and observed X for a single trial. One then reads upper and lower bounds directly.

So here using 95% CI and an observed X of 4 nearest λ_{bar} of 4.15

$CI_P := (1.09 \quad 10.24)$ **< this is the 95% CI for $\mu = \lambda$ because time T = 1**

^ In our example, we specified λ. If time interval T other than unity (i.e., 1), then $\lambda = \mu/T$ and CI for $\lambda = (\mu/T, \mu/T)$

It is interesting to note that had we been dealing with real data, we might have observed an X value different than close to the mean of $\lambda_{bar} = 4.15$. For instance, in a replicate of R_P above we might have observed something different, for example X = 10. The CI now changes a little:

$CI_P := (4.80 \quad 18.39)$ **< 95% CI for $\mu = \lambda$ given that we observed X=10 in our data.**

Note that indirect calculation bypassing Table 8 can be done in a way similar to use of pbinom function shown above. See Rosner p. 213.

General Strategies for Sampling a Population

ORIGIN ≡ 0

Rosner pp. 169-177 provides an excellent survey of the research designs employed in obtaining "unbiased" and "representative" samples that may be viewed as fairly representing the population from which they come. Although statistical calculations are generally silent about issues of sampling, of course, the believability of statistical results obtained are often critically dependent on them. Rosner highlights the use of pseudo-random number tables in setting up different kinds of studies, and provides some important terminology, summarized here...

Random Selection - Use of random numbers to select uniquely identified individuals from a population, usually without replacement.

Random Assignment - Use of random numbers to a assign *fixed numbers* of individuals to each treatment or analysis category, usually without replacement.

Randomized Trial - In comparing the effect of different levels of "treatment" (clinical or otherwise), individuals from a population are assigned *at random* to specific treatment classes (or categories). This hopefully guards against some other factor biasing the sample and being responsible for observed difference in outcome between the classes, rather than the treatment themselves.

Block Randomization - Random selection placing individuals into treatment classes often involves *replicate blocks* - each essentially a randomized trial.

Stratified Design - Treatment classes are set up explicitly regarding values observed in individuals for one or more *"accessory"* or *"covariate" variables.* The different classes defined by these variables are called *strata* (singl. *stratum).* Within strata, random selection, random assignment, or block ransomization may be employed.

Blind Designs - When *knowledge* on the part of researcher, subject, or both ("double blind') might influence behavior within strata or blocks, care is taken insulate the study from this knowledge.

Standard Statistical packages, such as SYSTAT, SAS, or SPSS offer the ability to partition data into strata and sub-blocks with ease. Thus, once prototyped, they can offer a significant time advantage in analysis of large data sets having a complex design.

ORIGIN ≡ 0

One Sample t-Test

This test with associated descriptivive statistics is designed to test hypotheses about the mean of a population with unknown variance.

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

H_0: $\mu = \mu_0$ **< μ_0 is a specified value for μ**

H_1: $\mu < \mu_0$ **< One sided test**

Test Statistic:

$$t := \frac{X_{bar} - \mu_0}{\frac{s}{\sqrt{n}}}$$

< t is the normalized distance between means Xbar and μ_0

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := inverse\Phi_t(\alpha)$ $C := qt(\alpha, n - 1)$ **< α implies C is the 'Critical Value' (in X) specified by cumulative probability $\Phi_N(X) = \alpha$ found by the 'q' function of the t Distribution.**

Sampling Distribution:

If Assumptions hold and H_0 is true, then t ~$t_{(n-1)}$

Decision Rule:

IF t < C, THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

P = Φ_t(t) **< probability of finding normalized distance t given the assumptions.**

Common attributions for P:

IF	**0.001**	**< P**		**then the results are *statistically very highly significant.***
IF	**0.001**	**< P <**	**0.01**	**then the results are *statistically highly significant.***
IF	**0.01**	**< P <**	**0.05**	**then the results are *statistically signifcant.***
IF	**0.05**	**< P**		**then the resulst are *NOT statistically signifcant.***

Example:

iris := READPRN("c:/2007BiostatsData/iris.txt")

i := 0 .. 49

$SL_i := \left(iris^{\langle 1 \rangle}\right)_i$ < **Assembling Sepal Length data for the first species only**

n := length(SL) n = 50 < **n = number of observations X**

$Xbar_{SL}$:= mean(SL) $Xbar_{SL}$ = 5.006 < **mean of X**

$SD_{SL} := \sqrt{Var(SL)}$ SD_{SL} = 0.3525 < **sample standard deviation of X**

$SE_{SL} := \frac{SD_{SL}}{\sqrt{n}}$ SE_{SL} = 0.0498 < **standard error of the sample *mean* of X**

Assumptions:

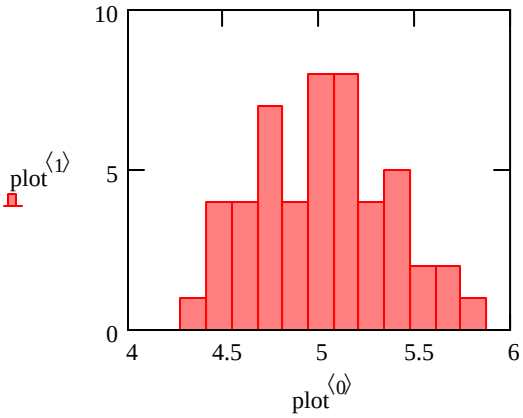
- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,$σ^2$).

- Variance $σ^2$ of the popopulation σ2 is *unknown.*

Var(SL) = 0.1242

^ our sample estimate of population variance

plot := histogram(15, SL)



Hypotheses:

$μ_0$:= 5.1 < **Set $μ_0$ for the test:**

H_0: μ = $μ_0$ < **$μ_0$ is a specified value for μ**

H_1: μ < $μ_0$ < **One sided test**

Test Statistic:

$t := \frac{Xbar_{SL} - μ_0}{SE_{SL}}$ t = –1.8857

Critical Value of the Test and Distribution of t:

α := 0.05 < **Probability of Type I error must be explicitly set**

C := qt(α, n – 1) C = –1.6766

Decision Rule:

IF t < C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

t = –1.8857 C = –1.6766

Probability Value:

pt(t, n – 1) = 0.0326 **P = $Φ_t$(t)**

Protoype using R:

Commands:

```
>attach iris
>SL=Sepal_Length[Species==setosa]
>t.test(SL,alternative="less",mu=5.1,
conf.level = 0.95)
```

Results: One Sample t-test

data: SL
t = -1.8857, df = 49, p-value = 0.03264
alternative hypothesis: true mean is less than 5.1
95 percent confidence interval:
-Inf 5.089575
sample estimates:
mean of x
5.006

One Sample t-Test

Other One Way case:

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_n$ are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

H_0: $\mu = \mu_0$ < μ_0 is a specified value for μ

H_1: $\mu > \mu_0$ < Other One sided test

Test Statistic:

$$t := \frac{X_{bar} - \mu_0}{\frac{s}{\sqrt{n}}}$$

Critical Value of the Test:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$C := \text{inverse}\Phi_t(1 - \alpha)$ $C := qt(1 - \alpha, n - 1)$

^ α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi(X) = 1-\alpha$ for the other tail of the 'q' function of the t Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then $t \sim t_{(n-1)}$

Decision Rule:

IF t > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = $\Phi_t(X)$ at 1-α < Rosner p 237

Example Other One Way case:

$Xbar_{SL} = 5.006$ $SD_{SL} = 0.3525$ $SE_{SL} = 0.0498$ < same descriptive statistics as above

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_n$ are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

$\mu_0 := 4.9$ < Set μ_0 for the test:

H_0: $\mu = \mu_0$ < μ_0 is a specified value for μ

H_1: $\mu > \mu_0$ < Other One sided test

Test Statistic:

$$t := \frac{Xbar_{SL} - \mu_0}{SE_{SL}} \qquad t = 2.1264$$

Critical Value of the Test and Distribution of t:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$C := qt(1 - \alpha, n - 1)$ $C = 1.6766$ < Critical One way test on the other tail of the t Distribution

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$|t| = 2.1264$ $C = 1.6766$

Probability Value:

$1 - pt(t, n - 1) = 0.0193$

Prototype with R:

Command:
t.test(SL,alternative="greater",mu=4.9,conf.level = 0.95)

One Sample t-test data: SL
t = 2.1264, df = 49, p-value = 0.01927
alternative hypothesis: true mean is greater than 4.9
95 percent confidence interval: 4.922425 Inf
sample estimates:
mean of x
5.006

Two Way Case:

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_n$ are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation $\sigma 2$ is *unknown.*

Hypotheses:

$H_0: \mu = \mu_0$ < μ_0 is a specified value for μ

$H_1: \mu \neq \mu_0$ < **TWO sided test**

Test Statistic:

$$t := \frac{X_{bar} - \mu_0}{\frac{s}{\sqrt{n}}}$$

Critical Value of the Test:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right) \qquad C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$$

$$C_1 := qt\left(\frac{\alpha}{2}, n-1\right) \qquad C_2 := qt\left(1 - \frac{\alpha}{2}, n-1\right)$$

< α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi_t(X) = \alpha/2$ for each tail of the 'q' function of the t Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then $t \sim t_{(n-1)}$

Decision Rule:

IF $|t| > C$, THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 $\Phi_t(t)$,1-2 $\Phi_t(t)$ < Rosner Eq 7.11 p. 241

$$P := \min[\, 2 \cdot pt(t, n-1), 2 \cdot (1 - pt(t, n-1))\,]$$

Confidence Interval for the mean:

$$\left(X_{bar} + C_1 \cdot \frac{s}{\sqrt{n}} \quad X_{bar} + C_2 \cdot \frac{s}{\sqrt{n}}\right)$$

< Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find the Lower Bound of the CI.

Example Two Way case:

$Xbar_{SL} = 5.006$ $SD_{SL} = 0.3525$ $SE_{SL} = 0.0498$ **< same descriptive statistics as above**

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown.*

Test Statistic:

$$t := \frac{Xbar_{SL} - \mu_0}{SE_{SL}} \qquad t = 2.1263975$$

Hypotheses:

$\mu_0 := 4.9$ **< Set μ_0 for the test:**

H_0: $\mu = \mu_0$ **< μ_0 is a specified value for μ**

H_1: $\mu \neq \mu_0$ **< TWO sided test**

Critical Value of the Test and Distribution of t:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n-1\right)$ $C_1 = -2.0096$ $C_2 := qt\left(1-\frac{\alpha}{2}, n-1\right)$ $C_2 = 2.0096$

Decision Rule: IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 2.1264$ $C = 1.6766$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n-1), 2 \cdot (1 - pt(t, n-1))\,]$ $P = 0.0385322$

Confidence Interval for the mean:

$CI := \left(Xbar_{SL} + C_1 \cdot SE_{SL} \quad Xbar_{SL} + C_2 \cdot SE_{SL}\right)$ $CI = (4.9058235 \quad 5.1061765)$

Prototype for Two Way Case:

Prototype with Systat:

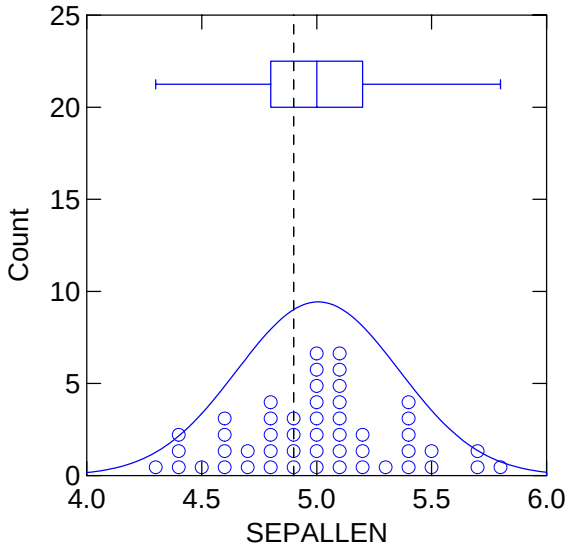
SYSTAT Rectangular file C:\Program Files\SYSTAT 9\Data\Iris.syd,
created Wed May 20, 1987 at 12:41:12,
contains variables:

SPECIES SEPALLEN SEPALWID PETALLEN PETALWID

The following results are for:
SPECIES = 1.0000000

One-sample t test of SEPALLEN with 50 cases; Ho: Mean = 4.9000000

Mean = 5.0060000 95.00% CI = 4.9058235 to 5.1061765
SD = 0.3524897 t = 2.1263975
df = 49 Prob = 0.0385322



Systat's t-test is only of the Two Way variety!

Prototype with R:

Command:
>t.test(SL,alternative="two.sided",mu=4.9,conf.level = 0.95)

One Sample t-test

data: SL
t = 2.1264, df = 49, p-value = 0.03853
alternative hypothesis: true mean is not equal to 4.9
95 percent confidence interval:
4.905824 5.106176
sample estimates:
mean of x
5.006

ORIGIN ≡ 0

One Sample χ^2 Test of Variance for a Normal Distribution

This test is designed to test hypotheses about whether the variance of a population is statistically equivalent to specified values.

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

Note that this requirement is critical and not robust, thus limiting this test's usefulness.

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

H_0: $\sigma^2 = \sigma_0^2$ < σ_0^2 is a specified value for σ^2

H_1: $\sigma^2 <> \sigma_0^2$ < Two Sided Case

Test Statistic:

$$Xsq := \frac{(n-1)\cdot s^2}{\sigma_0^2}$$ **< population corrected ratio of observed sample variance and hypothesized variance**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := \text{inverse}\Phi\chi 2\left(\frac{\alpha}{2}\right) \qquad C_2 := \text{inverse}\Phi\chi 2\left(1-\frac{\alpha}{2}\right)$$

$$C_1 := \text{qchisq}\left(\frac{\alpha}{2}, n-1\right) \qquad C_2 := \text{qchisq}\left(1-\frac{\alpha}{2}, n-1\right)$$

α implies lower limit C_1 and upper limit C_2 'Critical Values' (in X) specified by cumulative probability $\Phi\chi^2(X) = \alpha$ found by the 'q' function of the χ^2 Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then Xsq ~ $\chi^2_{(n-1)}$

Decision Rule:

IF $\chi^2 < C_1$ or $\chi^2 < C_1$, THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

P = $2\Phi\chi^2_{n-1,\alpha}$ or $2\Phi\chi^2_{n-1,1-\alpha/2}$

Common attributions for P:

IF	**0.001**	**< P**		**then the results are *statistically very highly significant.***
IF	**0.001**	**< P <**	**0.01**	**then the results are *statistically highly significant.***
IF	**0.01**	**< P <**	**0.05**	**then the results are *statistically signifcant.***
IF	**0.05**	**< P**		**then the resulst are *NOT statistically signifcant.***

Confidence Interval for σ^2:

$$CI := \left[\frac{(n-1)\cdot s^2}{C_2} \quad \frac{(n-1)\cdot s^2}{C_1}\right]$$

Example:

iris := READPRN("c:/2007BiostatsData/iris.txt")

i := 0 .. 49

$SL_i := \left(iris^{\langle 1 \rangle}\right)_i$ < **Assembling Sepal Length data for the first species only**

n := length(SL) n = 50 < **n = number of observations X**

$Xbar_{SL} := mean(SL)$ $Xbar_{SL} = 5.006$ < **mean of X**

$SD_{SL} := \sqrt{Var(SL)}$ $SD_{SL} = 0.3525$ < **sample standard deviation of X**

$SE_{SL} := \frac{SD_{SL}}{\sqrt{n}}$ $SE_{SL} = 0.0498$ < **standard error of the sample *mean* of X**

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

Note that this requirement is critical and not robust, thus limiting this test's usefulness.

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

$\sigma_0 := 0.4$ $\sigma_0^2 = 0.16$ < **variance to be tested**

H_0: $\sigma^2 = \sigma_0^2$ < **σ_0^2 is a specified value for σ^2**

H_1: $\sigma^2 <> \sigma_0^2$ < **Two Sided Case**

Test Statistic:

$Xsq := \frac{(n-1) \cdot SD_{SL}^2}{\sigma_0^2}$ $Xsq = 38.0512$ < **ratio of variances**

Critical Value of the Test and Distribution of t:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$C_1 := qchisq\left(\frac{\alpha}{2}, n-1\right)$ $C_1 = 31.5549$ $C_2 := qchisq\left(1-\frac{\alpha}{2}, n-1\right)$ $C_2 = 70.2224$

Decision Rule:

IF $\chi^2 < C_1$ or $\chi^2 < C_1$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$Xsq = 38.0512$ $C_1 = 31.5549$ $C_2 = 70.2224$

Probability Value:

$2 \cdot pchisq(Xsq, n-1) = 0.2575$

Confidence Interval for σ^2:

$CI := \left[\frac{(n-1) \cdot SD_{SL}^2}{C_2} \quad \frac{(n-1) \cdot SD_{SL}^2}{C_1}\right]$ $CI = (0.0867 \quad 0.1929)$

Prototype for χ^2 Test of Variance:

Rosner Example 7.46 p. 268

Rosner Table 6.6 Differences:

$d := \begin{pmatrix} -6 \\ 3 \\ 2 \\ -3 \\ 1 \\ 0 \\ -1 \\ 1 \\ 3 \\ -2 \end{pmatrix}$

$d_{bar} := mean(d)$ $d_{bar} = -0.2$

$s := \sqrt{Var(d)}$ $s^2 = 8.1778$

$n := length(d)$ $n = 10$

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown.*

Hypotheses:

$\sigma_0 := \sqrt{35}$ $\sigma_0^2 = 35$ **< variance to be tested, Rosner p. 267**

H_0: $\sigma^2 = \sigma_0^2$ **< σ_0^2 is a specified value for σ^2**

H_1: $\sigma^2 <> \sigma_0^2$ **< Two Sided Case**

Test Statistic:

$Xsq := \frac{(n-1)\cdot s^2}{\sigma_0^2}$ $Xsq = 2.1029$ **< confirmed Rosner p. 268**

Critical Value of the Test and Distribution of t:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qchisq\left(\frac{\alpha}{2}, n-1\right)$ $C_1 = 2.7004$ $C_2 := qchisq\left(1-\frac{\alpha}{2}, n-1\right)$ $C_2 = 19.0228$

^ values confirmed Rosner p. 268

Decision Rule:

IF $\chi^2 < C_1$ or $\chi^2 < C_1$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$Xsq = 2.1029$ $C_1 = 2.7004$ $C_2 = 19.0228$

Probability Value:

$2 \cdot pchisq(Xsq, n-1) = 0.0205$ **< confirmed Rosner p. 269**

Confidence Interval for σ^2:

$CI := \left[\frac{(n-1)\cdot s^2}{C_2} \quad \frac{(n-1)\cdot s^2}{C_1}\right]$

$CI = (3.869 \quad 27.2553)$ **< interval confirmed Rosner p. 201**

ORIGIN ≡ 0

One Sample Tests of Discrete distribution Parameters

Hypothesis tests for parameters of Binomial and Poisson Distibution are handled in ways corresponding to construction of confidence limits shown in Biostatistics Worksheet 19.

Binomial Distribution test for p - the probability of "success" in each trial:

Assumptions:

- Let $X_1, X_2, X_3, \ldots, X_m$ be a random sample from a population ~ Binomial(n,p) i.e., with Binomial Distribution with parameters n=number of trials and p=probability of success.

Given sample parameters n, p_{hat}, $q_{hat} = (1- p_{hat})$

- IF $n \cdot p_{hat} \cdot q_{hat} \geq 5$ **then use Normal Theory Approximation OTHERWISE use Exact Methods.**

Hypotheses:

H_0: $p = p_0$ **< must specify hypothesized value p_0**

H_1: $p <> p_0$ **< Two Sided Test**

Normal Theory Approximation:

It is asssumed that: $p_{hat} \sim N(p_0, p_0 q_0/n)$

Normal Theory Test Statistic:

$$z := \frac{p_{hat} - p_0}{\sqrt{\frac{p_0 \cdot (q_0)}{n}}}$$

< Standardized distance between p_{hat} and p_0

Critical Values of the Test:

$\alpha := 0.05$ **< probability of Type I error must be explicitly set**

α implies C_1 & C_2 - upper and lower 'Critical Values' (in p)

$C_1 := \text{inverse}\Phi_N\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}\right)$

$C_1 := \text{qnorm}\left(\frac{\alpha}{2}, 0, 1\right)$ $C_2 := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$ **< the results of 'q' functions of N(0,1)**

Sampling Distribution:

If Assumptions hold and H_0 is true, then z ~N(0,1)

Decision Rule:

IF $z < C_1$ or $z > C_2$ THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value for z:

P = minimum($2\Phi_N(z)$,$2(1-\Phi_N(z))$)

$P := \min[\, 2 \cdot (1 - \text{pnorm}(z, 0, 1)), 2 \cdot (\text{pnorm}(z, 0, 1))\,]$

Confidence Interval:

$$CI := \left(p_{hat} + C_1 \cdot \sqrt{\frac{p_{hat} \cdot q_{hat}}{n}} \quad p_{hat} + C_2 \cdot \sqrt{\frac{p_{hat} \cdot q_{hat}}{n}}\right)$$

< Note that C_1 & C_2 are explicitly calculated above so added to p_{hat} here

Prototype of Normal Theory Approximation of Binomial Distribution:

using Rosner Examples 6.48-649 p. 205-206 & 7.47-7.48 p. 268-270.

$n := 10000$ **< sample size 10,000 women assessed for cancer**

$p_{hat} := 0.040$ $q_{hat} := 1 - p_{hat}$ **< estimated incidence of cancer in sample**

$n \cdot p_{hat} \cdot q_{hat} = 384$ **< problem qualifies for Normal Theory Approximation**

$p_0 := 0.020$ $q_0 := 1 - p_0$ **< p_0 is the hypothesis to be tested**

Normal Theory Approximation:

It is asssumed that: $p_{hat} \sim N(p_0, p_0 q_0/n)$

Hypotheses:

H_0: $p = p_0$

H_1: $p <> p_0$ **< Two Sided Test**

Normal Theory Test Statistic:

$$z := \frac{p_{hat} - p_0}{\sqrt{\frac{p_0 \cdot (q_0)}{n}}} \qquad z = 14.2857$$

Critical Values of the Test:

$\alpha := 0.05$ **< probability of Type I error must be explicitly set**

$C_1 := qnorm\left(\frac{\alpha}{2}, 0, 1\right)$ $C_1 = -1.96$ $\frac{\alpha}{2} = 0.025$

$C_2 := qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right)$ $C_2 = 1.96$ $1 - \frac{\alpha}{2} = 0.975$

Sampling Distribution:

If Assumptions hold and H_0 is true, then z ~N(0,1)

Decision Rule:

IF $z < C_1$ or $z > C_2$ THEN REJECT H_0 OTHERWISE ACCEPT H_0

$z = 14.2857$ $C_1 = -1.96$ $C_2 = 1.96$

Probability Value for z:

$P := 2 \cdot \Phi(z)$ **if $p_{hat} < p_0$** **OR** $P := 2 \cdot (1 - \Phi(z))$ **if $p_{hat} > p_0$**

$p_{hat} = 0.04$ $p_0 = 0.02$ $pnorm(z, 0, 1) = 1$

$P := 2 \cdot (1 - pnorm(z, 0, 1))$ $P = 0$

$P := 2 \cdot (pnorm(z, 0, 1))$ $P = 2$ **< confirmed p.272**

Confidence Interval:

$$CI := \left(p_{hat} + C_1 \cdot \sqrt{\frac{p_{hat} \cdot q_{hat}}{n}} \quad p_{hat} + C_2 \cdot \sqrt{\frac{p_{hat} \cdot q_{hat}}{n}}\right) \qquad CI = (0.0362 \quad 0.0438)$$

^ confirmed p. 206

Binomial Distribution test for p - the probability of "success" in each trial:

Assumptions:

- Let $X_1, X_2, X_3, \ldots, X_m$ be a random sample from a population ~ Binomial(n,p) i.e., with Binomial Distribution with parameters n=number of trials and p=probability of success.

Given sample parameters n_{hat}, p_{hat}, $q_{hat} = (1- p_{hat})$

- IF $n_{hat} \cdot p_{hat} \cdot q_{hat} \geq 5$ **then use Normal Theory Approximation**
OTHERWISE use Exact Methods.

Hypotheses:

H_0: $p = p_0$

H_1: $p <> p_0$ **< Two Sided Test**

Exact Methods Probabilities:

$P := 2 \cdot \Phi(X \leq k)$ **or** $P := 2 \cdot \Phi(k \geq X)$

Critical Values of the Test:

$\alpha := 0.05$ **< probability of Type I error must be explicitly set**

Decision Rule:

IF $P < \alpha$ THEN REJECT H_0
OTHERWISE ACCEPT H_0

Prototype of Exact Methods:

Rosner Example 7.49 p. 274

$n := 13$ $p_0 := 0.20$ $q_0 := 1 - p_0$ $q_0 = 0.8$ **< hypothesized value**

$n \cdot p_0 \cdot q_0 = 2.08$ **< fails criterion for Normal Theory Approximation**

$p_{hat} := \frac{5}{13}$ $p_{hat} = 0.3846$ **< sample point estimate of p**

Hypotheses:

H_0: $p = p_0$

H_1: $p <> p_0$ **< Two Sided Test**

Exact Methods Probabilities:

$i := 0 .. 4$

$D_i := dbinom(i, n, p_0)$

$\Phi_i := pbinom(i, n, p_0)$

$$D = \begin{pmatrix} 0.055 \\ 0.1787 \\ 0.268 \\ 0.2457 \\ 0.1535 \end{pmatrix} \qquad \Phi = \begin{pmatrix} 0.055 \\ 0.2336 \\ 0.5017 \\ 0.7473 \\ 0.9009 \end{pmatrix}$$

$P := 2 \cdot \Phi_4$ $P = 1.8017$

$P := 2 \cdot (1 - \Phi_4)$ $P = 0.1983$ **< P must be less than one. Is it less than α?**

Prototype of Above Binomial examples using R's Exact Binomial test:

Rossner Example 7.49 p. 274:

```
>n=13
> p=0.2
> x=5
> binom.test(x,n,p=0.2,alternative="two.sided",conf.level=0.95)


        Exact binomial test

data:  x and n
number of successes = 5, number of trials = 13, p-value = 0.1541
alternative hypothesis: true probability of success is not equal to 0.2
95 percent confidence interval:
 0.1385793 0.6842224
sample estimates:
probability of success
            0.3846154
```

Results are match for p_{hat} and are close for p-value, but not exactly the same. Thus, the methods for calculating P must be subtly different..

Rossner Examples 6.48-649 p. 205-206 & 7.47-7.48 p. 268-270.

```
> n=10000
> p=0.020
> x=0.040
> binom.test(400,10000,p=0.2,alternative="two.sided",conf.level=0.95)

        Exact binomial test

data:  400 and 10000
number of successes = 400, number of trials = 10000, p-value < 2.2e-16
alternative hypothesis: true probability of success is not equal to 0.2
95 percent confidence interval:
 0.03624378 0.04402702
sample estimates:
probability of success
                 0.04
```

Again, similar results, but not the same...

Poisson Distribution test for μ:

Assumptions:

- Let $X_1, X_2, X_3, \ldots, X_m$ be a random sample from a population ~ Poisson(μ) i.e., with Poisson Distribution with parameter μ events per time interval.

Hypotheses:

H_0: $\mu = \mu_0$

H_1: $\mu <> \mu_0$ < **Two Sided Test**

Remainder of this section not worked out at this time... See Rosner p. 277

ORIGIN ≡ 0

Estimating Power and Sample Size for a One Sample t-Test

Pilot studies are often run in advance of collecting data for major statistical analyses. These studies are used to determine the POWER of an analysis - i.e., the ability of the analysis to satisfactorily lead to rejection of the Null Hypothesis and determining sufficient Sample Size to support sufficient power.

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_n$ are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ2 is *unknown*.

Hypotheses:

H_0: $\mu = \mu_0$ < **μ_0 is a specified value for μ**

H_1: $\mu = \mu_1 < \mu_0$ < **One sided test - here a specific alternative μ_1 must be chosen**

OR

H_1: $\mu = \mu_1 <> \mu_0$ < **Two sided test - here a specific alternative μ_1 must be chosen**

Hypothesis Distance:

$$D := \frac{|\mu_0 - \mu_1|}{\frac{s}{\sqrt{n}}}$$ < **t is the normalized distance between alternates μ_0 and μ_1**

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

Approximating POWER of the Test:

Note that Rosner's entire presentation of this topic is predicated on *known* population variance σ^2, allowing him to estimate probabilities using the standardized normal distribution N(0,1). In general, however, σ^2 must be estimated by sample variance s^2. Thus,the calculations must be considered only approximate...

$z := \text{inverse}\Phi_z(\alpha)$ $z := \text{qnorm}(\alpha, 0, 1)$ < **ONE SIDED approximately!**

$z := \text{inverse}\Phi_z(\alpha)$ $z := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$ < **TWO SIDED approximately!**

Power of the Test: < POWER = (1-β) the inverse probability of Type II error, Rosner p. 229

$POWER := \Phi_z(z + D)$ $POWER := \text{pnorm}(z + D, 0, 1)$

Estimated Sample Size Needed:

$\beta := 0.1$ $1 - \beta = 0.9$ < **Type II error rate (β) or POWER (1-β) must be explicitly set**

$\alpha := 0.05$ $1 - \alpha = 0.95$ < **Type I error rate (α) must be explicitly set**

ONE WAY:

$$N := \frac{\sigma^2 \cdot \left(\text{inverse}\Phi_z(1-\beta) + \text{inverse}\Phi_z(1-\alpha)\right)^2}{(\mu_0 - \mu_1)^2} \qquad N := \frac{s^2 \cdot \left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}(1-\alpha,0,1)\right)^2}{(\mu_0 - \mu_1)^2}$$

TWO WAY:

$$N := \frac{\sigma^2 \cdot \left(\text{inverse}\Phi_z(1-\beta) + \text{inverse}\Phi_z\left(1-\frac{\alpha}{2}\right)\right)^2}{(\mu_0 - \mu_1)^2} \qquad N := \frac{s^2 \cdot \left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2}{(\mu_0 - \mu_1)^2}$$

Example:

Rosner Eample 7.27-7.28 p. 248-249

Assumptions:

- Observed values X_1, X_2, X_3, ... X_n are a random sample from ~N(μ,σ^2).

- Variance σ^2 of the popopulation σ^2 is *known*.

$s := 50 \qquad n := 10$

Hypotheses:

H_0: $\mu = \mu_0$ $\qquad \mu_0 := 175$

H_1: $\mu = \mu_1 < \mu_0$ $\qquad \mu_1 := 190$

Hypothesis Distance:

$$D := \frac{|\mu_0 - \mu_1|}{\frac{s}{\sqrt{n}}} \qquad D = 0.9487$$

Critical Value of the Test:

$\alpha := 0.01$ **< Probability of Type I error must be explicitly set**

Approximating POWER of the Test:

$z := qnorm(\alpha, 0, 1) \quad z = -2.3263$ **< approximately!**

Power of the Test:

$POWER := pnorm(z + D, 0, 1) \qquad POWER = 0.0842 \qquad 1 - 0.9158 = 0.0842$

^ calculation confirmed p. 249

Example:

Rosner Eample 7.35 p. 255

$s := 50 \quad \mu_0 := 175 \quad \mu_1 := 190$

Estimated Sample Size Needed for One Way Analysis:

$\beta := 0.1 \quad 1 - \beta = 0.9$ **< Type II error rate (β) or POWER (1-β) must be explicitly set**

$\alpha := 0.05 \quad 1 - \alpha = 0.95$ **< Type I error rate (α) must be explicitly set**

$$N := \frac{s^2 \cdot (qnorm(1 - \beta, 0, 1) + qnorm(1 - \alpha, 0, 1))^2}{(\mu_0 - \mu_1)^2}$$

$s^2 = 2500$

$qnorm(1 - \beta, 0, 1) = 1.2816$

$qnorm(1 - \alpha, 0, 1) = 1.6449$

$(1.28 + 1.645)^2 = 8.5556$

$(\mu_0 - \mu_1)^2 = 225$

$N = 95.1539$

^ calculation confirmed p. 255

Example:

Rosner Eample 7.37 p. 257-258

$s := 10$ $|\mu_0 - \mu_1| := 5$

Estimated Sample Size Needed for One Way Analysis:

$\beta := 0.2$ $1 - \beta = 0.8$ **< Type II error rate (β) or POWER (1-β) must be explicitly set**

$\alpha := 0.05$ $1 - \alpha = 0.95$ **< Type I error rate (α) must be explicitly set**

$$N := \frac{s^2 \cdot (\text{qnorm}(1-\beta,0,1) + \text{qnorm}(1-\alpha,0,1))^2}{(5)^2}$$

$s^2 = 100$

$\text{qnorm}(1-\beta,0,1) = 0.8416$

$\text{qnorm}(1-\alpha,0,1) = 1.6449$

$N = 24.7302$ **< calculation confirmed p. 258**

$(5)^2 = 25$

Constructing Q-Q Plots

ORIGIN ≡ 1

Assessing Normality of sample data is an essential part of statistical analysis. Q-Q Plots are one way easy to do this. They are also interesting at this point in our course since the demonstrate the use of the inverse cumulative probability function for the Normal Distribution.

So loading some familiar data to assess:

iris := READPRN("c:/2007BiostatsData/iris.txt")

i := 1 .. 50

$SL_i := \left(iris^{\langle 2 \rangle}\right)_i$ **< Assembling Sepal Length data for the first species only**

n := length(SL) n = 50 **< n = number of observations X**

$Xbar_{SL} := mean(SL)$ $Xbar_{SL} = 5.006$ **< mean of X**

$SD_{SL} := \sqrt{Var(SL)}$ $SD_{SL} = 0.352$ **< sample standard deviation of X**

$SE_{SL} := \frac{SD_{SL}}{\sqrt{n}}$ $SE_{SL} = 0.05$ **< standard error of the sample *mean* of X**

Calculating Cumultive Probability levels $\Phi_N(X)$:

We will look at variable SL here:

SL =

	1
1	5.1
2	4.9
3	4.7
4	4.6
5	5
6	5.4
7	4.6
8	5
9	4.4
10	4.9
11	5.4
12	4.8
13	4.8
14	4.3
15	5.8
16	5.7

First we sort SL:

$SL_{sort} := sort(SL)$

SL_{sort} =

	1
1	4.3
2	4.4
3	4.4
4	4.4
5	4.5
6	4.6
7	4.6
8	4.6
9	4.6
10	4.7
11	4.7
12	4.8
13	4.8
14	4.8
15	4.8
16	4.8

Now we treat each index of SL as a quantile:

$P_i := \frac{\left(i - \frac{1}{2}\right)}{n}$

i := 1 .. n

^ the 1/2 here is a correction factor

P =

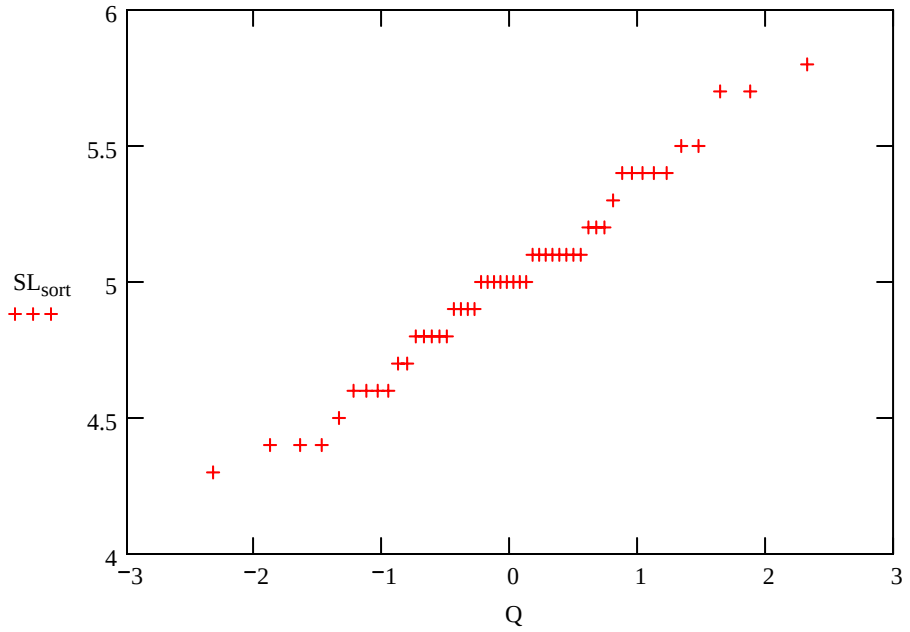
	1
1	0.01
2	0.03
3	0.05
4	0.07
5	0.09
6	0.11
7	0.13
8	0.15
9	0.17
10	0.19
11	0.21
12	0.23
13	0.25
14	0.27
15	0.29
16	0.31

From the values of P = $\Phi_N(X)$, we now convert back to X

$Q_i := qnorm(P_i, 0, 1)$

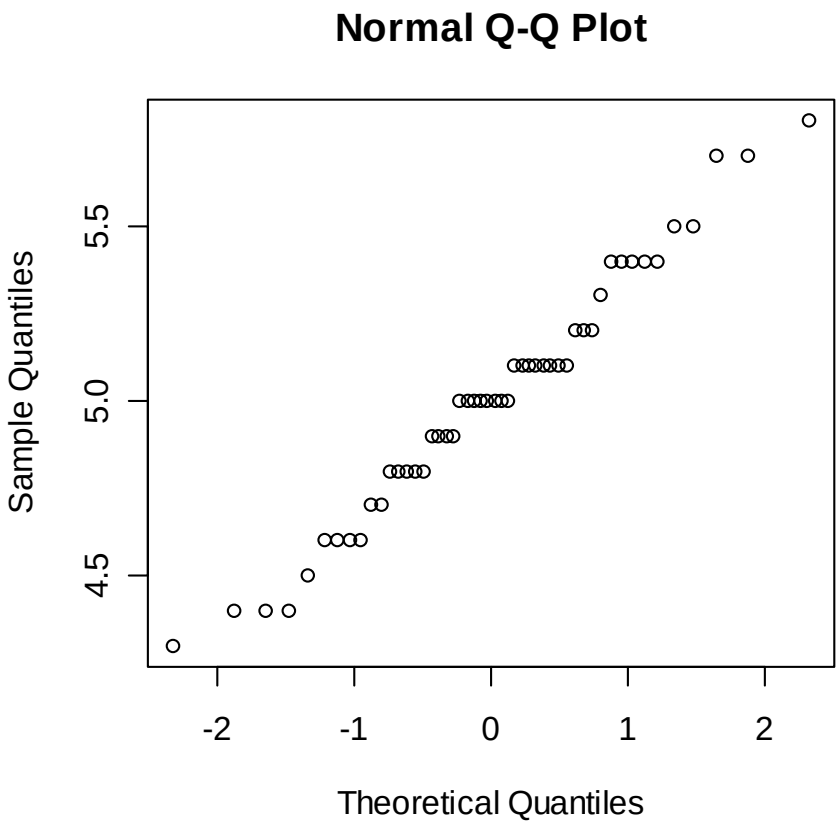
Q =

	1
1	-2.326
2	-1.881
3	-1.645
4	-1.476
5	-1.341
6	-1.227
7	-1.126
8	-1.036
9	-0.954
10	-0.878
11	-0.806
12	-0.739
13	-0.674
14	-0.613
15	-0.553
16	-0.496



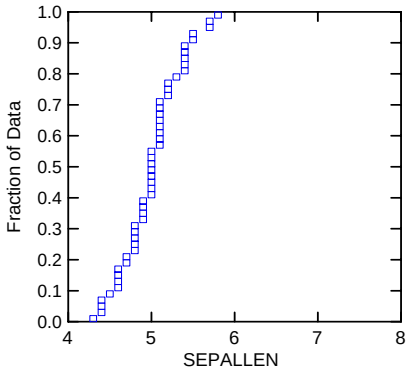
If the sample data are distributed close to the Normal distribution, the Q-Q plot should be mostly a straight line in the center with an overall S-shaped curve towards each end.

Output from R:

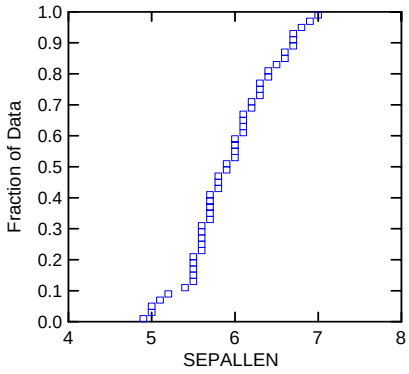


Systat output showing graphs for species 1,2 & 3:

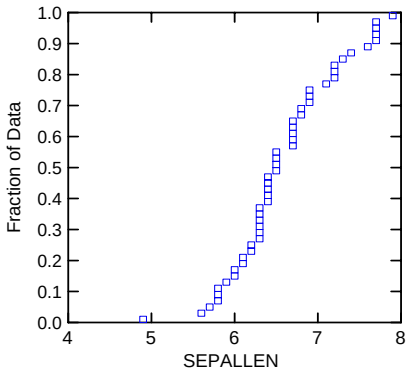
1



2



3



Assignment for Week 7

This week we begin the task of prototyping some of the most important standard statistical tests. Our object is to not only to understand how calculations are done by hand as exemplified, for example, by the various Biostatistics worksheets. We also need to be able to identify appropriate data for each test, and to conduct analyses on a routine basis.

So, this week ***use both R and SPSS*** and try the following tasks. Note also that I have posted R documentation for you on our website.

1. ***Single population t-test***. Devise a small dataset of your own consisting of only a few objects (say around five). State your assumptions, as well as null and alternative hypotheses. Then calculate the t statistic, critical values (for a given α) and probability. State the decision rule and results. Finally calculate the associated $(1-\alpha)$ confidence interval for μ.

   Now find a realistic set of data and perform the single population t-test using both R and SPSS and compare the results. Example datasets are posted on our website and others may be found in online files associated with each program. You may have to 'prep' the data using Word or Excel, before inputting into each program, but that's a normal part of the process.

2. ***Paired t-test***. Devise a small dataset of your own consisting of only a few object pairs (say around five). State your assumptions, as well as null and alternative hypotheses. Then calculate the t statistic, critical values (for a given α) and probability. State the decision rule and results. Finally calculate the associated $(1-\alpha)$ confidence interval for μ_d.

   Now find a realistic set of data and perform a paired t-test using both R and SPSS and compare the results.

3. ***Two population t-tests with equal and unequal variances***. Devise a small dataset of your own consisting of only a few objects (say around five) for each group. State your assumptions, as well as null and alternative hypotheses. Then calculate the t statistic, critical values (for a given α) and probability. State the decision rule and results. Finally calculate the associated $(1-\alpha)$ confidence interval for $\mu_1 - \mu_2$. ***Note that here you will be working with two different tests, so it will be useful to compare these results.***

   Now find a realistic set of data and perform ***both*** two population t-tests using both R and SPSS and compare.

4. ***F-test for equality of variance between two populations.*** Now use the small dataset in 3 for this test. State your assumptions, as well as null and alternative hypotheses. Then calculate the F statistic, critical values (for a given α) and probability. State the decision rule and results.

   Using your realistic data from 3, perform this test and interpret the results. Based on your F-test, which two-population t-test should be performed?

Student's t-Test

Description

Performs one and two sample t-tests on vectors of data.

Usage

```
t.test(x, ...)

## Default S3 method:
t.test(x, y = NULL,
       alternative = c("two.sided", "less", "greater"),
       mu = 0, paired = FALSE, var.equal = FALSE,
       conf.level = 0.95, ...)

## S3 method for class 'formula':
t.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	a (non-empty) numeric vector of data values.
`y`	an optional (non-empty) numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`. You can specify just the initial letter.
`mu`	a number indicating the true value of the mean (or difference in means if you are performing a two sample test).
`paired`	a logical indicating whether you want a paired t-test.
`var.equal`	a logical variable indicating whether to treat the two variances as being equal. If `TRUE` then the pooled variance is used to estimate the variance otherwise the Welch (or Satterthwaite) approximation to the degrees of freedom is used.
`conf.level`	confidence level of the interval.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is a numeric variable giving the data values and `rhs` a factor with two levels giving the corresponding groups.
`data`	an optional matrix or data frame (or similar: see `model.frame`) containing the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
`subset`	an optional vector specifying a subset of observations to be used.

na.action	a function which indicates what should happen when the data contain NAs. Defaults to getOption("na.action").
...	further arguments to be passed to or from methods.

Details

The formula interface is only applicable for the 2-sample tests.

alternative = "greater" is the alternative that x has a larger mean than y.

If paired is TRUE then both x and y must be specified and they must be the same length. Missing values are removed (in pairs if paired is TRUE). If var.equal is TRUE then the pooled estimate of the variance is used. By default, if var.equal is FALSE then the variance is estimated separately for both groups and the Welch modification to the degrees of freedom is used.

If the input data are effectively constant (compared to the larger of the two means) an error is generated.

Value

A list with class "htest" containing the following components:

statistic	the value of the t-statistic.
parameter	the degrees of freedom for the t-statistic.
p.value	the p-value for the test.
conf.int	a confidence interval for the mean appropriate to the specified alternative hypothesis.
estimate	the estimated mean or difference in means depending on whether it was a one-sample test or a two-sample test.
null.value	the specified hypothesized value of the mean or mean difference depending on whether it was a one-sample test or a two-sample test.
alternative	a character string describing the alternative hypothesis.
method	a character string indicating what type of t-test was performed.
data.name	a character string giving the name(s) of the data.

See Also

prop.test

Examples

```
t.test(1:10,y=c(7:20))      # P = .00001855
t.test(1:10,y=c(7:20, 200)) # P = .1245    -- NOT significant anymore

## Classical example: Student's sleep data
plot(extra ~ group, data = sleep)
## Traditional interface
with(sleep, t.test(extra[group == 1], extra[group == 2]))
## Formula interface
t.test(extra ~ group, data = sleep)
```

F Test to Compare Two Variances

Description

Performs an F test to compare the variances of two samples from normal populations.

Usage

```
var.test(x, ...)

## Default S3 method:
var.test(x, y, ratio = 1,
         alternative = c("two.sided", "less", "greater"),
         conf.level = 0.95, ...)

## S3 method for class 'formula':
var.test(formula, data, subset, na.action, ...)
```

Arguments

`x, y`	numeric vectors of data values, or fitted linear model objects (inheriting from class `"lm"`).
`ratio`	the hypothesized ratio of the population variances of `x` and `y`.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`. You can specify just the initial letter.
`conf.level`	confidence level for the returned confidence interval.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is a numeric variable giving the data values and `rhs` a factor with two levels giving the corresponding groups.
`data`	an optional matrix or data frame (or similar: see `model.frame`) containing the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
`subset`	an optional vector specifying a subset of observations to be used.
`na.action`	a function which indicates what should happen when the data contain `NA`s. Defaults to `getOption("na.action")`.
`...`	further arguments to be passed to or from methods.

Details

The null hypothesis is that the ratio of the variances of the populations from which `x` and `y` were drawn, or in the data to which the linear models `x` and `y` were fitted, is equal to `ratio`.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value of the F test statistic.
`parameter`	the degrees of the freedom of the F distribtion of the test statistic.
`p.value`	the p-value of the test.
`conf.int`	a confidence interval for the ratio of the population variances.
`estimate`	the ratio of the sample variances of `x` and `y`.
`null.value`	the ratio of population variances under the null.
`alternative`	a character string describing the alternative hypothesis.
`method`	the character string `"F test to compare two variances"`.
`data.name`	a character string giving the names of the data.

See Also

`bartlett.test` for testing homogeneity of variances in more than two samples from normal distributions; `ansari.test` and `mood.test` for two rank based (nonparametric) two-sample tests for difference in scale.

Examples

```
x <- rnorm(50, mean = 0, sd = 2)
y <- rnorm(30, mean = 1, sd = 1)
var.test(x, y)                  # Do x and y have the same variance?
var.test(lm(x ~ 1), lm(y ~ 1))  # The same.
```

Power calculations for one and two sample t tests

Description

Compute power of test, or determine parameters to obtain target power.

Usage

```
power.t.test(n = NULL, delta = NULL, sd = 1, sig.level = 0.05,
             power = NULL,
             type = c("two.sample", "one.sample", "paired"),
             alternative = c("two.sided", "one.sided"),
             strict = FALSE)
```

Arguments

`n`	Number of observations (per group)
`delta`	True difference in means
`sd`	Standard deviation
`sig.level`	Significance level (Type I error probability)
`power`	Power of test (1 minus Type II error probability)
`type`	Type of t test
`alternative`	One- or two-sided test
`strict`	Use strict interpretation in two-sided case

Details

Exactly one of the parameters `n`, `delta`, `power`, `sd`, and `sig.level` must be passed as NULL, and that parameter is determined from the others. Notice that the last two have non-NULL defaults so NULL must be explicitly passed if you want to compute them.

If `strict = TRUE` is used, the power will include the probability of rejection in the opposite direction of the true effect, in the two-sided case. Without this the power will be half the significance level if the true difference is zero.

Value

Object of class `"power.htest"`, a list of the arguments (including the computed one) augmented with `method` and `note` elements.

Note

`uniroot` is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given.

Author(s)

Peter Dalgaard. Based on previous work by Claus Ekstrøm

See Also

`t.test`, `uniroot`

Examples

```
power.t.test(n = 20, delta = 1)
power.t.test(power = .90, delta = 1)
power.t.test(power = .90, delta = 1, alt = "one.sided")
```

ORIGIN ≡ 0

Paired t-Test

The Paired t-test is employed in cases, such as a longitudinal study, where two sets of measurements are exactly matched for each individual of a population.

Assumptions:

- Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.

- Let $d_i = X_{2,i} - X_{1,i}$ for each individual i are a random sample from ~$N(\mu_d, \sigma_d^2)$.

- Variance σ_d^2 of the popopulation σ^2 is *unknown*.

Hypotheses:

H_0: $\mu_d = 0$ **< No difference in mean between populations X_1 & X_2.**

H_1: $\mu_d <> 0$ **< Two sided test**

Test Statistic:

$$t := \frac{d_{bar}}{\frac{s_d}{\sqrt{n}}}$$

< t is the normalized mean X_2bar - X_2bar

< s_d is the sample standard deviation of d_i

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right) \qquad C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$$

$$C_1 := qt\left(\frac{\alpha}{2}, n - 1\right) \qquad C_2 := qt\left(1 - \frac{\alpha}{2}, n - 1\right)$$

< α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi_t(X) = \alpha/2$ for each tail of the 'q' function of the t Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then t ~$t_{(n-1)}$

Decision Rule:

IF |t| > C, THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t) **< Rosner Eq 7.11 p. 241**

$P := \min[\, 2 \cdot pt(t, n - 1), 2 \cdot (1 - pt(t, n - 1))\,]$

Confidence Interval for the mean:

$$\left(d_{bar} + C_1 \cdot \frac{s}{\sqrt{n}} \quad d_{bar} + C_2 \cdot \frac{s}{\sqrt{n}}\right)$$

< Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find the Lower Bound of the CI.

Example:

Blood Pressure data Rosner Table 8.2 p. 301:

$$X := \begin{pmatrix} 115 & 128 \\ 112 & 115 \\ 107 & 106 \\ 119 & 128 \\ 115 & 122 \\ 138 & 145 \\ 126 & 132 \\ 105 & 109 \\ 104 & 102 \\ 115 & 117 \end{pmatrix}$$

Subtracting values in second column from values in first column:

$$d := X^{\langle 1 \rangle} - X^{\langle 0 \rangle} \qquad d = \begin{pmatrix} 13 \\ 3 \\ -1 \\ 9 \\ 7 \\ 7 \\ 6 \\ 4 \\ -2 \\ 2 \end{pmatrix}$$

Descriptive statistics for d:

$n := \text{length}(d)$ $n = 10$

$d_{bar} := \text{mean}(d)$ $d_{bar} = 4.8$

$s_d := \sqrt{\text{Var}(d)}$ $s_d = 4.5656$

^ values confirmed p. 301

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.**
- **Let $d_i = X_{2,i} - X_{1,i}$ for each individual i are a random sample from ~$N(\mu_d, \sigma_d^2)$.**
- **Variance σ_d^2 of the popopulation σ^2 is *unknown*.**

Test Statistic:

$$t := \frac{d_{bar}}{\frac{s_d}{\sqrt{n}}}$$

< t is the normalized mean X_2bar - X_2bar

< s_d is the sample standard deviation of d_i

Hypotheses:

H_0: $\mu_d = 0$ **< No difference in mean between populations X_1. & X_2.**

H_1: $\mu_d <> 0$ **< Two sided test**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := qt\left(\frac{\alpha}{2}, n-1\right) \qquad C_2 := qt\left(1-\frac{\alpha}{2}, n-1\right)$$

$(C_1 \;\; C_2) = (-2.2622 \;\; 2.2622)$ **< confirmed p. 301**

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Decision Rule:

$(C_1 \;\; C_2) = (-2.2622 \;\; 2.2622)$ $t = 3.3247$ **< confirmed p. 301**

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[\, 2 \cdot pt(t, n-1), 2 \cdot (1 - pt(t, n-1))\,]$ $P = 0.0088743369$ **< confirmed p. 301**

Confidence Interval for the mean:

$$CI := \left(d_{bar} + C_1 \cdot \frac{s_d}{\sqrt{n}} \quad d_{bar} + C_2 \cdot \frac{s_d}{\sqrt{n}}\right)$$

$CI = (1.534 \;\; 8.066)$ **< confirmed p. 303**

Prototype of Example in Systat:

Paired samples t test on X1 vs X2 with 10 cases

Mean X1 = 115.6000000
Mean X2 = 120.4000000
Mean Difference = -4.8000000 95.00% CI = -8.0660132 to -1.5339868
SD Difference = 4.5655716 t = -3.3246511
df = 9 Prob = 0.0088743

calculations from above:

$\text{mean}\left(X^{\langle 0 \rangle}\right) = 115.6$

$\text{mean}\left(X^{\langle 1 \rangle}\right) = 120.4$

$\text{mean}(d) = 4.8$

$s_d = 4.5656$

$t = 3.3247$
$n - 1 = 9$

$P = 0.0089$

$CI = (1.534 \quad 8.066)$

^ Note that the difference (d) in Systat involved subtracting X1 from X2, thus all numbers are reversed but the results are the same. SD Difference is slightly off from MathCad's calculation. This is the result, I guess of rounding in taking the square root of variance.

Prototype of Example in R:

COMMANDS:

```
> X1=c(115,112,107,119,115,138,126,105,104,115)
> X2=c(128,115,106,128,122,145,132,109,102,117)
> t.test(X1,X2,paired=TRUE,alternative="two.sided")
```

Paired t-test

data: X1 and X2
t = -3.3247, df = 9, p-value = 0.008874
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-8.066013 -1.533987
sample estimates:
mean of the differences
-4.8

^ Same results as SYSTAT

Example:

$X := \text{READPRN}(\text{"C:/2007BiostatsData/AnorexiaALL.txt"})$

$d := X^{\langle 0 \rangle} - X^{\langle 1 \rangle}$

$n := \text{length}\left(X^{\langle 0 \rangle}\right) \qquad n = 72$

$\text{mean}\left(X^{\langle 0 \rangle}\right) = 82.4083 \qquad \text{mean}\left(X^{\langle 1 \rangle}\right) = 85.1722$

Descriptive statistics for d:

$n := \text{length}(d) \qquad n = 72$

$d_{bar} := \text{mean}(d) \qquad d_{bar} = -2.7639$

$s_d := \sqrt{\text{Var}(d)} \qquad s_d = 7.9836$

$X =$

	0	1
0	80.7	80.2
1	89.4	80.1
2	91.8	86.4
3	74	86.3
4	78.1	76.1
5	88.3	78.1
6	87.3	75.1
7	75.1	86.7
8	80.6	73.5
9	78.4	84.6
10	77.6	77.4
11	88.7	79.5
12	81.3	89.6
13	78.1	81.4
14	70.5	81.8
15	77.3	77.3

$d =$

	0
0	0.5
1	9.3
2	5.4
3	-12.3
4	2
5	10.2
6	12.2
7	-11.6
8	7.1
9	-6.2
10	0.2
11	9.2
12	-8.3
13	-3.3
14	-11.3
15	0

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.**
- **Let $d_i = X_{2,i} - X_{1,i}$ for each individual i are a random sample from ~$N(\mu_d, \sigma_d^2)$.**
- **Variance σ_d^2 of the popopulation σ^2 is *unknown*.**

Test Statistic:

$$t := \frac{d_{bar}}{\frac{s_d}{\sqrt{n}}}$$

< t is the normalized mean X_2bar - X_2bar

< s_d is the sample standard deviation of d_i

$t = -2.9376$

Hypotheses:

H_0: $\mu_d = 0$ **< No difference in mean between populations $X_{1.}$ & $X_{2.}$**

H_1: $\mu_d <> 0$ **< Two sided test**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := qt\left(\frac{\alpha}{2}, n-1\right) \qquad C_2 := qt\left(1-\frac{\alpha}{2}, n-1\right)$$

$(C_1 \;\; C_2) = (-1.9939 \;\; 1.9939)$

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Decision Rule:

$(C_1 \;\; C_2) = (-1.9939 \;\; 1.9939)$ $t = -2.9376$

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[\, 2 \cdot pt(t, n-1), 2 \cdot (1 - pt(t, n-1))\,]$ $P = 0.0044577181$

Confidence Interval for the mean:

$$CI := \left(d_{bar} + C_1 \cdot \frac{s_d}{\sqrt{n}} \;\; d_{bar} + C_2 \cdot \frac{s_d}{\sqrt{n}} \right)$$

$CI = (-4.6399 \;\; -0.8878)$

Prototype with SYSTAT:

Paired samples t test on BEFORE vs AFTER with 72 cases

Mean BEFORE = 82.4083333
Mean AFTER = 85.1722222
Mean Difference = -2.7638889 95.00% CI = -4.6399424 to -0.8878353
SD Difference = 7.9835977 t = -2.9375697
df = 71 Prob = 0.0044577

Compare this result with that those using nonparametric Sign and Signed-Rank Tests. See 2007 Biostatistics Worksheets 30 & 31.

ORIGIN ≡ 0

Two Sample t-Test with Equal Variances

This test is employed where two sets of measurements are derived from samples with approximately equal varainces.

Assumptions:

- Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)

- Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)

- Variances σ_1^2 & σ_2^2 are approximately equal but *unknown.*

- Samples $X_{1,n1}$ and $X_{2,n2}$ are *independent.*

Hypotheses:

H_0: $\mu_1 = \mu_2$ **< No difference in mean between populations X_1 & X_2.**

H_1: $\mu_1 <> \mu_2$ **< Two sided test**

Pooled Sample Variance:

$$s_p := \frac{(n_1 - 1)\cdot s_1^{\,2} + (n_2 - 1)\cdot s_2^{\,2}}{n_1 + n_2 - 2}$$

< variance is pooled from the two samples and adjusted for each sample's size n_1 & n_2.

Test Statistic:

$$t := \frac{X_{1bar} - X_{2bar}}{\sqrt{s_p^{\,2}\cdot\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

< t is the normalized mean X_2bar - X_2bar

< s_p^2 is the pooled sample variance defined above

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n_1 + n_2 - 2\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n_1 + n_2 - 2\right)$

< α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi_t(X) = \alpha/2$ for each tail of the 'q' function of the t Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then t ~$t_{(n1+n2-2)}$

Decision Rule:

IF |t| > C, THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min\left[2\cdot pt\left(t, n_1 + n_2 - 2\right), 2\cdot\left(1 - pt\left(t, n_1 + n_2 - 2\right)\right)\right]$

Confidence Interval for the mean:

$$\left[X_{1bar} - X_{2bar} + C_1\cdot\sqrt{s_p^{\,2}\cdot\left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \quad X_{1bar} - X_{2bar} + C_2\cdot\sqrt{s_p^{\,2}\cdot\left(\frac{1}{n_1} + \frac{1}{n_2}\right)}\right]$$

^ Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{1bar} - X_{2bar} here to find the Lower Bound of the CI.

Example (BWT difference between Females and Males):

cats := READPRN("c:/2007BiostatsData/cats.txt")

Females:

i := 0 .. 46

$F_{BWT_i} := (cats^{\langle 1 \rangle})_i$

$F_{HWT_i} := (cats^{\langle 2 \rangle})_i$

$n_1 := length(F_{BWT})$ $n_1 = 47$

$X_{1bar} := mean(F_{BWT})$ $X_{1bar} = 2.3596$

$s_1 := \sqrt{Var(F_{BWT})}$ $s_1^2 = 0.0751$

Males:

j := 47 .. 143

$M_{BWT_{j-47}} := (cats^{\langle 1 \rangle})_j$

$M_{HWT_j} := (cats^{\langle 2 \rangle})_j$

$n_2 := length(M_{BWT})$ $n_2 = 97$

$X_{2bar} := mean(M_{BWT})$ $X_{2bar} = 2.9$

$s_2 := \sqrt{Var(M_{BWT})}$ $s_2^2 = 0.2185$

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)**
- **Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)**
- **Variances σ_1^2 & σ_2^2 are approximately equal but *unknown.***
- **Samples $X_{1,n1}$ and $X_{2,n2}$ are *independent.***

Hypotheses:

H_0: $\mu_1 = \mu_2$ < **No difference in mean between populations $X_{1.}$ & $X_{2.}$**

H_1: $\mu_1 <> \mu_2$ < **Two sided test**

Pooled Sample Variance:

$$s_p := \sqrt{\frac{(n_1 - 1) \cdot s_1^2 + (n_2 - 1) \cdot s_2^2}{n_1 + n_2 - 2}} \qquad s_p^2 = 0.1721$$

Test Statistic:

$$t := \frac{X_{1bar} - X_{2bar}}{\sqrt{s_p^2 \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad t = -7.3307$$

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n_1 + n_2 - 2\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n_1 + n_2 - 2\right)$ $(C_1 \quad C_2) = (-1.9768 \quad 1.9768)$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$(C_1 \quad C_2) = (-1.9768 \quad 1.9768)$ $t = -7.3307$

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[2 \cdot pt(t, n_1 + n_2 - 2), 2 \cdot (1 - pt(t, n_1 + n_2 - 2))]$ $P = 1.5904499939 \times 10^{-11}$

Confidence Interval for the mean:

$$CI := \left[X_{1bar} - X_{2bar} + C_1 \cdot \sqrt{s_p^2 \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \quad X_{1bar} - X_{2bar} + C_2 \cdot \sqrt{s_p^2 \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \right]$$

CI = (–0.6861584 –0.39469266)

Prototype of Example in Systat:

Females:	**Males:**	
$n_1 = 47$	$n_2 = 97$	
$X_{1bar} = 2.3596$	$X_{2bar} = 2.9$	**< Values from above**
$s_1 = 0.274$	$s_2 = 0.4675$	
$s_p = 0.4148$	$s_p^2 = 0.1721$	
$df := n_1 + n_2 - 2$	$df = 142$	
$t = -7.33066683$		
$X_{1bar} - X_{2bar} = -0.5404255$		$CI = (-0.6861584 \quad -0.39469266)$

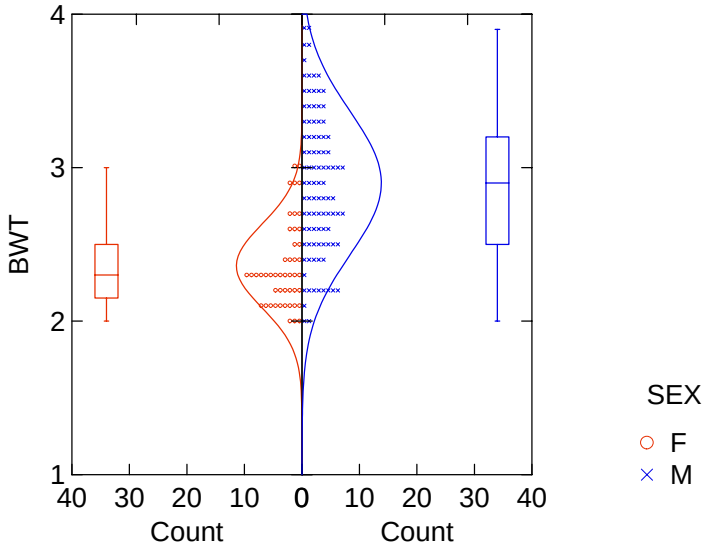
Two-sample t test on BWT grouped by SEX$

Group	**N**	**Mean**	**SD**
F	**47**	**2.3595745**	**0.2739879**
M	**97**	**2.9000000**	**0.4674844**

Separate Variance t = -8.7094885 df = 136.8 Prob = 0.000000
Difference in Means = -0.5404255 95.00% CI = -0.6631268 to -0.4177242

Pooled Variance t = -7.3306668 df = 142 Prob = 0.0000000
Difference in Means = -0.5404255 95.00% CI = -0.6861584 to -0.3946

^ Same as "Pooled Variance t" results in SYSTAT



Prototype of Example in R:

COMMANDS:

```
> cats=read.table("c:/2007BiostatsData/cats.txt")
> attach(cats)
> X1=Bwt[Sex=="F"]
> X2=Bwt[Sex=="M"]
> t.test(X1,X2,alternative="two.sided",var.equal=TRUE)
```

^ note specification of equal variances here

Two Sample t-test

```
data:  X1 and X2
t = -7.3307, df = 142, p-value = 1.590e-11
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.6861584 -0.3946927
sample estimates:
mean of x mean of y
 2.359574  2.900000
```

^ Results confirmed.

ORIGIN ≡ 0

Two Sample t-Test with Unequal Variances

This test is employed where two sets of measurements are derived from samples failing the F test for equal varainces.

Assumptions:

- Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)

- Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)

- Variances σ_1^2 & σ_2^2 are *unequal* and *unknown*.

- Samples $X_{1,n1}$ and $X_{2,n2}$ are *independent*.

Hypotheses:

H_0: $\mu_1 = \mu_2$ **< No difference in mean between populations X_1. & X_2.**

H_1: $\mu_1 <> \mu_2$ **< Two sided test**

Test Statistic:

$$t := \frac{X_{1bar} - X_{2bar}}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$

< t is the normalized mean X_1bar - X_2bar

Satterthwaite's Method Degrees of Freedom:

$$d_p := \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}\right)^2}{\frac{\left(\frac{s_1^2}{n_1}\right)^2}{(n_1 - 1)} + \frac{\left(\frac{s_2^2}{n_2}\right)^2}{(n_2 - 1)}}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, d_p\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, d_p\right)$

< α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi_t(X) = \alpha/2$ for each tail of the 'q' function of the t Distribution.

Sampling Distribution:

If Assumptions hold and H_0 is true, then t ~$t_{(dp)}$

Decision Rule:

IF |t| > C, THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min\left[2 \cdot pt(t, d_p), 2 \cdot \left(1 - pt(t, d_p)\right)\right]$

Confidence Interval for the mean:

$$\left(X_{1bar} - X_{2bar} + C_1 \cdot \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \quad X_{1bar} - X_{2bar} + C_2 \cdot \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \right)$$

^ Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{1bar} - X_{2bar} here to find the Lower Bound of the CI.

Example (BWT difference between Females and Males):

cats := READPRN("c:/2007BiostatsData/cats.txt")

Females:

i := 0 .. 46

$F_{BWT_i} := (cats^{\langle 1 \rangle})_i$

$F_{HWT_i} := (cats^{\langle 2 \rangle})_i$

$n_1 := length(F_{BWT})$ $n_1 = 47$

$X_{1bar} := mean(F_{BWT})$ $X_{1bar} = 2.3596$

$s_1 := \sqrt{Var(F_{BWT})}$ $s_1^2 = 0.0751$

Males:

j := 47 .. 143

$M_{BWT_{j-47}} := (cats^{\langle 1 \rangle})_j$

$M_{HWT_j} := (cats^{\langle 2 \rangle})_j$

$n_2 := length(M_{BWT})$ $n_2 = 97$

$X_{2bar} := mean(M_{BWT})$ $X_{2bar} = 2.9$

$s_2 := \sqrt{Var(M_{BWT})}$ $s_2^2 = 0.2185$

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)**
- **Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)**
- **Variances σ_1^2 & σ_2^2 are *unequal* and *unknown*.**
- **Samples $X_{1,n1}$ and $X_{2,n2}$ are *independent*.**

Hypotheses:

H_0: $\mu_1 = \mu_2$ < **No difference in mean between populations $X_{1.}$ & $X_{2.}$**

H_1: $\mu_1 <> \mu_2$ < **Two sided test**

Test Statistic:

$$t := \frac{X_{1bar} - X_{2bar}}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$$ $t = -8.7095$

Satterthwaite's Method Degrees of Freedom:

$$d_p := \frac{\left(\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2} \right)^2}{\frac{\left(\frac{s_1^2}{n_1} \right)^2}{(n_1 - 1)} + \frac{\left(\frac{s_2^2}{n_2} \right)^2}{(n_2 - 1)}}$$ $d_p = 136.8379$

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, d_p\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, d_p\right)$ $(C_1 \quad C_2) = (-1.9775 \quad 1.9775)$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$(C_1 \quad C_2) = (-1.9775 \quad 1.9775)$ $t = -8.7095$

Probability Value:

P = minimum(2 $\Phi_t(t)$,1-2 $\Phi_t(t)$

$P := \min\left[2 \cdot pt(t, d_p), 2 \cdot \left(1 - pt(t, d_p)\right)\right]$ $P = 8.8818 \times 10^{-15}$

Confidence Interval for the mean:

$$CI := \left(X_{1bar} - X_{2bar} + C_1 \cdot \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \quad X_{1bar} - X_{2bar} + C_2 \cdot \sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}} \right)$$

$CI = (-0.66312684 \quad -0.41772423)$

Prototype of Example in Systat:

Females:	**Males:**	
$n_1 = 47$	$n_2 = 97$	
$X_{1bar} = 2.3596$	$X_{2bar} = 2.9$	**< Values from above**
$s_1 = 0.274$	$s_2 = 0.4675$	
$d_p = 136.8379$		
$t = -8.7094885$		
$X_{1bar} - X_{2bar} = -0.5404255$		$CI = (-0.66312684 \quad -0.41772423)$

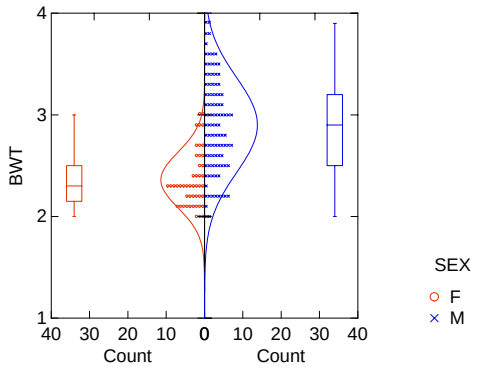
Two-sample t test on BWT grouped by SEX$

Group	**N**	**Mean**	**SD**
F	**47**	**2.3595745**	**0.2739879**
M	**97**	**2.9000000**	**0.4674844**

Separate Variance t = -8.7094885 df = 136.8 Prob = 0.0000000
Difference in Means = -0.5404255 95.00% CI = -0.6631268 to -0.4177242

Pooled Variance t = -7.3306668 df = 142 Prob = 0.0000000
Difference in Means = -0.5404255 95.00% CI = -0.6861584 to -0.3946927

^ Same result as "Separate Variance" report above.



Prototype of Example in R:

COMMANDS:

```
> cats=read.table("c:/2007BiostatsData/cats.txt")
> attach(cats)
> X1=Bwt[Sex=="F"]
> X2=Bwt[Sex=="M"]
> t.test(X1,X2,alternative="two.sided",var.equal=FALSE)
                    OR
> t.test(X1,X2,alternative="two.sided")   ^ note specification of unequal variances here.
                                            This is the default setting in R
```

Welch Two Sample t-test

```
data:  X1 and X2
t = -8.7095, df = 136.838, p-value = 8.831e-15
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.6631268 -0.4177242
sample estimates:
mean of x mean of y
 2.359574  2.900000
```

^ **Results confirmed.**

ORIGIN ≡ 0

F-Test for Equal Variances in Two Samples

This test tests for equal variances between two samples as a way of deciding which t-test to use.

Assumptions:

- Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n1}$ are a random sample from ~$N(\mu_1,\sigma_1^2)$

- Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n2}$ are a random sample from ~$N(\mu_2,\sigma_2^2)$

- Samples from the two samples are *independent.*

Hypotheses:

H_0: $\sigma_1^2 = \sigma_2^2$ < No difference in variance between populations X_1. & X_2.

H_1:$\sigma_1^2 <> \sigma_2^2$ < Two sided test

Test Statistic:

$$F := \frac{s_1^2}{s_2^2}$$

< F is the ratio of sample variances

Sampling Distribution:

If Assumptions hold and H_0 is true, then F ~$F_{(n1-1)/(n2-1)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := \text{inverse}\Phi_F\left(\frac{\alpha}{2}\right) \qquad C_2 := \text{inverse}\Phi_F\left(1-\frac{\alpha}{2}\right)$$

$$C_1 := qF\left(\frac{\alpha}{2}, n_1 - 1, n_2 - 1\right) \qquad C_2 := qF\left(1-\frac{\alpha}{2}, n_1 - 1, n_2 - 1\right)$$

< α implies C the 'Critical Value' (in X) specified by cumulative probability $\Phi_F(X) = \alpha/2$ for each tail of the 'q' function of the F Distribution with $(n_1-1)/(n_2-1)$ degrees of freesom.

Decision Rule:

IF |F| > C, THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_F(F),1-2 Φ_F(F)

$$P := \min\left[2 \cdot pF\left(F, n_1 - 1, n_2 - 1\right), 2 \cdot \left(1 - pF\left(F, n_1 - 1, n_2 - 1\right)\right)\right]$$

Example (BWT difference between Females and Males):

cats := READPRN("c:/2007BiostatsData/cats.txt")

Females:

$i := 0 .. 46$

$F_{BWT_i} := \left(cats^{\langle 1 \rangle}\right)_i$

$F_{HWT_i} := \left(cats^{\langle 2 \rangle}\right)_i$

$n_1 := length(F_{BWT})$ $n_1 = 47$

$X_{1bar} := mean(F_{BWT})$ $X_{1bar} = 2.3596$

$s_1 := \sqrt{Var(F_{BWT})}$ $s_1^2 = 0.0751$

Males:

$j := 47 .. 143$

$M_{BWT_{j-47}} := \left(cats^{\langle 1 \rangle}\right)_j$

$M_{HWT_j} := \left(cats^{\langle 2 \rangle}\right)_j$

$n_2 := length(M_{BWT})$ $n_2 = 97$

$X_{2bar} := mean(M_{BWT})$ $X_{2bar} = 2.9$

$s_2 := \sqrt{Var(M_{BWT})}$ $s_2^2 = 0.2185$

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)**
- **Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)**
- **Samples from the two samples are *independent.***

Hypotheses:

H_0: $\sigma_1^2 = \sigma_2^2$ < No difference in variance between populations $X_{1.}$ & $X_{2.}$

H_1:$\sigma_1^2 <> \sigma_2^2$ < Two sided test

Test Statistic:

$$F := \frac{s_1^2}{s_2^2} \qquad F = 0.3435$$

Sampling Distribution:

If Assumptions hold and H_0 is true, then $F \sim F_{(n1-1)/(n2-1)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qF\left(\frac{\alpha}{2}, n_1 - 1, n_2 - 1\right)$ $C_2 := qF\left(1 - \frac{\alpha}{2}, n_1 - 1, n_2 - 1\right)$

$(C_1 \quad C_2) = (0.5919 \quad 1.6155)$

Decision Rule:

IF |F| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$(C_1 \quad C_2) = (0.5919 \quad 1.6155)$ $F = 0.3435$

Probability Value:

P = minimum(2 Φ_F(F),1-2 Φ_F(F)

$P := \min\left[2 \cdot pF(F, n_1 - 1, n_2 - 1), 2 \cdot \left(1 - pF(F, n_1 - 1, n_2 - 1)\right)\right]$ $P = 0.0001$

Prototype of Example in R:

Females:	**Males:**	
$n_1 = 47$	$n_2 = 97$	
$X_{1bar} = 2.3596$	$X_{2bar} = 2.9$	**< Values from above**
$s_1^2 = 0.0751$	$s_2^2 = 0.2185$	
$F = 0.3435$	$df_1 := n_1 - 1$	$df_1 = 46$
$P = 0.0001$	$df_2 := n_2 - 1$	$df_2 = 96$

COMMANDS:

```
> cats=read.table("c:/2007BiostatsData/cats.txt")
> attach(cats)
> X1=Bwt[Sex=="F"]
> X2=Bwt[Sex=="M"]
> var.test(X1,X2,alternative="two.sided",conf.level=0.95)
```

```
        F test to compare two variances

data:  X1 and X2
F = 0.3435, num df = 46, denom df = 96, p-value = 0.0001157
alternative hypothesis: true ratio of variances is not equal to 1
95 percent confidence interval:
 0.2126277 0.5803475
sample estimates:
ratio of variances
         0.3435015
```

^ calculation confirmed. Note that we do not have a formula for calculating the Confidence Interval of the ratio reported here by R.

ORIGIN ≡ 0

POWER & Sample Size in t-Tests for Two Samples

Estimates for sample size (N) and power (1-β) on this page are similar to that seen in Biostatistics Worksheet 24, but here for comparing two samples. The methods assume knowledge of σ_1^2 and σ_2^2 which, of course we do not know, and the Normal distribution N(0,1). Thus, values obtained can only be considered approximate.

Two Samples of Equal Size:

Assumptions:

- **Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n}$ are a random sample from ~N(μ_1,σ_1^2)**
- **Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n}$ are a random sample from ~N(μ_2,σ_2^2)**
- **Samples from the two samples are *independent***
- **Population variances $\sigma_1^2 = \sigma_2^2$**

Estimated Sample Size:

$\alpha := 0.05$ **< Type I error rate must be explicitly set**

$\beta := 0.10$ **< Type II error rate must be explicitly set**

$\Delta = \mu_1 - \mu_2$ **< desired distance between means must be explicitly set**

$$n := \frac{\left(\sigma_1^2 + \sigma_2^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2}$$

Estimated POWER:

$$z := \text{qnorm}\left(\frac{\alpha}{2},0,1\right)$$

$$D := \frac{\Delta}{\sqrt{\frac{\sigma_1^2}{n_1} + \frac{\sigma_2^2}{n_2}}}$$

< here $n_1 = n_2 = n$, but this Power calculation also applies to samples of unequal size. See below.

$\text{POWER} := \text{pnorm}(z + D, 0, 1)$

Example: Cardiovascular Disease - Rosner Ex. 8.29 p. 332:

Sample Size:

$X_{1bar} := 132.86$ $s_1 := 15.34$

$X_{2bar} := 127.44$ $s_2 := 18.23$

< we will use X_{bar} & s from the samples as point estimates for σ & μ of the populations

$\Delta := X_{1bar} - X_{2bar}$ $\Delta = 5.42$

$\alpha := 0.05$ $\beta := 0.2$ **< parameters must be explicitly set to estimate N**

$$n := \frac{\left(s_1^2 + s_2^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2} \qquad n = 151.6661$$

^ sample size confirmed p. 332

Power:

$n_1 := 100$ $n_2 := 100$ $\Delta := 5$

$z := qnorm\left(\frac{\alpha}{2}, 0, 1\right)$ $z = -1.96$

$D := \frac{\Delta}{\sqrt{\frac{s_1^2}{n_1} + \frac{s_2^2}{n_2}}}$ $D = 2.0986$

$POWER := pnorm(z + D, 0, 1)$ $POWER = 0.5551$ **< confirmed p. 334**

Prototype in R:

Sample Size:

COMMANDS:

```
> X1bar=132.86
> X2bar=127.44
> del=X1bar-X2bar
> s1=15.34
> s2=18.23
> SD=mean(c(s1,s2))
> power.t.test(n=NULL,delta=del,sd=SD,sig.level=0.05,power=0.8,
  type="two.sample",alternative="two.sided")
```

Note from R documentation:

Exactly one of the parameters n, delta, power, sd, and sig.level must be passed as NULL, and that parameter is determined from the others.

Two-sample t test power calculation

Sample size calculation approximately confirmed >

n = 151.5167
delta = 5.42
sd = 16.785
sig.level = 0.05
power = 0.8
alternative = two.sided

Power:

NOTE: n is number in *each* group

COMMANDS:

```
> s1=15.34
> s2=18.23
> SD=mean(c(s1,s2))
> power.t.test(n=100,delta=5,sd=16.785,sig.level=0.05,power=NULL,
  type="two.sample",alternative="two.sided")
```

Two-sample t test power calculation

power calculation approximately confirmed >

n = 100
delta = 5
sd = 16.785
sig.level = 0.05
power = 0.5541596
alternative = two.sided

NOTE: n is number in *each* group

Two Samples of Unequal Size:

Assumptions:

- **Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n1}$ are a random sample from ~N(μ_1,σ_1^2)**
- **Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n2}$ are a random sample from ~N(μ_2,σ_2^2)**
- **Samples from the two samples are *independent***
- **Population variances $\sigma_1^2 = \sigma_2^2$**

Estimated Sample Size:

$\alpha := 0.05$ **< Type I error rate must be explicitly set**

$\beta := 0.10$ **< Type II error rate must be explicitly set**

$\Delta = \mu_1 - \mu_2$ **< desired distance between means must be set**

$n_1 := \frac{n_2}{k}$ **< k must be set to determine relative size of n_1 & n_2**

$$n_1 := \frac{\left(\sigma_1^2 + \frac{\sigma_2^2}{k}\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2}$$

$$n_2 := \frac{\left(k \cdot \sigma_1^2 + \sigma_2^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2}$$

Estimated POWER:

Same as above...

Example: Rosner Ex 8.30 p. 333

$X_{1bar} := 132.86$ $s_1 := 15.34$

$X_{2bar} := 127.44$ $s_2 := 18.23$

$\Delta := X_{1bar} - X_{2bar}$ $\Delta = 5.42$

$\alpha := 0.05$ $\beta := 0.2$

$k := 2$

< we will use X_{bar} & s from the samples as point estimates for σ & μ of the populations

$\text{qnorm}(1-\beta,0,1) = 0.8416$

$\text{qnorm}\left(1-\frac{\alpha}{2},0,1\right) = 1.96$

$\Delta^2 = 29.3764$

$$n_1 := \frac{\left(s_1^2 + \frac{s_2^2}{k}\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2} \qquad n_1 = 107.2692$$

$$n_2 := \frac{\left(k \cdot s_1^2 + s_2^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\Delta^2} \qquad n_2 = 214.5385$$

values approximately confirmed p. 333 ^

Two Samples with Paired Design:

Assumptions:

- Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n}$ across individuals 1,2,3, ... ,n.
- Let $d_i = X_{2,i} - X_{1,i}$ for each individual i are a random sample from ~$N(\mu_d, \sigma_d^2)$.
- Variance σ_d^2 of the popopulation σ^2 is *unknown*.

Variance of the differences (d_i) given correlation of the observations:

$\rho := \text{corr}(X_{1.}, X_{2.})$

$\sigma_d^2 := \sigma_1^2 + \sigma_2^2 - 2 \cdot \rho \cdot \sigma_1 \cdot \sigma_2$ < variance of di in terms of variance for each population and correlation coefficient ρ

Estimated Sample Size:

$\alpha := 0.05$ < Type I error rate must be explicitly set

$\beta := 0.10$ < Type II error rate must be explicitly set

$\delta = \mu_1 - \mu_2$ < desired distance between means must be set

$$n := \frac{\left(2 \cdot \sigma_d^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\delta^2}$$

Estimated POWER:

$z := \text{qnorm}\left(\frac{\alpha}{2},0,1\right)$

$D := \frac{\sqrt{n} \cdot \delta}{\sigma_d \cdot \sqrt{2}}$

$\text{POWER} := \text{pnorm}(z + D, 0, 1)$

Example: Hypertension Rosner Ex 8.33 p. 334-336

$s_1 := 15 \quad s_2 := 15 \quad \rho := .7 \quad \delta := 5$

$s_d := \sqrt{s_1^2 + s_2^2 - 2 \cdot \rho \cdot s_1 \cdot s_2}$ $\quad s_d^2 = 135$

$\alpha := 0.05 \quad \beta := 0.20$

Estimated Sample Size:

$$n := \frac{\left(2 \cdot s_d^2\right) \cdot \left[\left(\text{qnorm}(1-\beta,0,1) + \text{qnorm}\left(1-\frac{\alpha}{2},0,1\right)\right)^2\right]}{\delta^2}$$ $\quad n = 84.7679$

^ **confirmed p. 336**

Estimated POWER:

$n := 75 \quad \alpha := 0.05 \quad \beta := 0.20 \quad \delta := 5 \quad s_d^2 = 135$

$z := \text{qnorm}\left(\frac{\alpha}{2},0,1\right)$ $\quad z = -1.96$

$D := \frac{\sqrt{n} \cdot \delta}{s_d \cdot \sqrt{2}}$ $\quad D = 2.6352$

$\text{POWER} := \text{pnorm}(z + D, 0, 1)$ $\quad \text{POWER} = 0.7502$

^ **confirmed p. 336**

Prototype in R:

Sample Size:

COMMANDS:

```
> s1=15
> s2=15
> del=5
> rho=0.7
> SD=sqrt(s1^2+s2^2-2*rho*s1*s2)
> power.t.test(n=NULL,delta=del,sd=SD,sig.level=0.05,power=0.8,
  type="paired",alternative="two.sided")

                Paired t test power calculation

half of what I expected >          n = 44.34303
                               delta = 5
                                  sd = 11.61895
                           sig.level = 0.05
                               power = 0.8
                         alternative = two.sided

                NOTE: n is number of *pairs*, sd is std.dev. of *differences* within
                pairs

> power.t.test(n=NULL,delta=del,sd=SD,sig.level=0.05,power=0.8,
  type="two.sample",alternative="two.sided")
                                        Two-sample t test power calculation

approximately what I expected >          n = 85.73891
                                     delta = 5
                                        sd = 11.61895
                                 sig.level = 0.05
                                     power = 0.8
                               alternative = two.sided

                                        NOTE: n is number in *each* group
```

Note from R documentation:

Exactly one of the parameters n, delta, power, sd, and sig.level must be passed as NULL, and that parameter is determined from the others.

Power:

COMMANDS"

```
> power.t.test(n=75,delta=del,sd=SD,sig.level=0.05,power=NULL,
  type="two.sample",alternative="two.sided")

                                        Two-sample t test power calculation

                                              n = 75
                                          delta = 5
                                             sd = 11.61895
                                      sig.level = 0.05
approximately what I expected >           power = 0.7447735
                                    alternative = two.sided

                                        NOTE: n is number in *each* group
```

```
> power.t.test(n=75,delta=del,sd=SD,sig.level=0.05,power=NULL,
  \ type="paired",alternative="two.sided")

     Paired t test power calculation

              n = 75
          delta = 5
             sd = 11.61895
      sig.level = 0.05
          power = 0.9571042
    alternative = two.sided

NOTE: n is number of *pairs*, sd is std.dev. of *differences* within pairs
```

not what I expected! >

Assignment for Week 8

Today we extend our survey of the standard two-population tests into the realm of nonparametric statistics. Given the number of tests in this course (and there are many in addition we will not cover), it is important to begin placing all statistical tests within an analytic framework. In general, for each strategy of data collection and analysis represented by paired and separate population t-test designs, there are corresponding nonparametric tests that cover much the same ground. Although less powerful because they consult less information derived from the data, they are often employed when parametric or other nonparametric tests "fail" due to violation of one or more underlying assumptions. The question of failure of tests due to lack of normality or sample size is often not a clear-cut decision, but a judgment call where degree, amount or importance of failure relative to the conclusion reached by the test should also be considered. Generally, I advise analyzing problems using multiple tests to compare results. If all the tests say the same thing, then it hardly matters which test to use. In publishing, I generally report the most conservative test, or occasionally all of them, to avoid complications with reviewers who may have a personal preference for one over another. When the different tests give importantly different probability levels then analysis becomes much more interesting and, in my opinion, stops being a strictly statistical problem. At issue is whether the extra information embedded in a "flawed" parametric test versus "correct" nonparametric test has meaning and value. If it does, then every effort must be made to utilize it. Sometimes a "variance stabilizing" or other kind of non-linear transformation corrects a flaw sufficiently to allow the parametric test. The results can then be back-transformed (using the transformation inverse) to obtain results in the "space" of the original variables. Other times, another test having "weird poisson" (or whatever) distribution and the exact design parameters that you need is already sitting in literature just waiting for you. When in doubt, I consult a "real" statistician. Occasionally it helps.

So, this week ***use both R and SPSS*** and try the following tasks. Note also that I have posted R documentation for you on our website. For each below, check the course website, R, or SPSS, for a suitable dataset. You may have to manipulate your data in Excel or Notepad or Word to get it in a form you can use. This is definitely part of the game of analyzing statistics using a computer, so work on your skills here. For each item in the list below, run the same dataset using the parametric and non-parametric analogs, and compare your results.

1. ***Paired t-test and non-parametric analogs***.
2. ***Two population t-tests with equal and unequal variances and non-parametric analog***.

Exact Binomial Test

Description

Performs an exact test of a simple null hypothesis about the probability of success in a Bernoulli experiment.

Usage

```
binom.test(x, n, p = 0.5,
           alternative = c("two.sided", "less", "greater"),
           conf.level = 0.95)
```

Arguments

`x`	number of successes, or a vector of length 2 giving the numbers of successes and failures, respectively.
`n`	number of trials; ignored if `x` has length 2.
`p`	hypothesized probability of success.
`alternative`	indicates the alternative hypothesis and must be one of `"two.sided"`, `"greater"` or `"less"`. You can specify just the initial letter.
`conf.level`	confidence level for the returned confidence interval.

Details

Confidence intervals are obtained by a procedure first given in Clopper and Pearson (1934). This guarantees that the confidence level is at least `conf.level`, but in general does not give the shortest-length confidence intervals.

Value

A list with class `"htest"` containing the following components:

`statistic`	the number of successes.
`parameter`	the number of trials.
`p.value`	the p-value of the test.
`conf.int`	a confidence interval for the probability of success.
`estimate`	the estimated probability of success.
`null.value`	the probability of success under the null, `p`.
`alternative`	a character string describing the alternative hypothesis.
`method`	the character string `"Exact binomial test"`.

`data.name` a character string giving the names of the data.

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, **26**, 404–413.

William J. Conover (1971), *Practical nonparametric statistics*. New York: John Wiley & Sons. Pages 97–104.

Myles Hollander & Douglas A. Wolfe (1973), *Nonparametric statistical inference*. New York: John Wiley & Sons. Pages 15–22.

See Also

`prop.test` for a general (approximate) test for equal or given proportions.

Examples

```
## Conover (1971), p. 97f.
## Under (the assumption of) simple Mendelian inheritance, a cross
##  between plants of two particular genotypes produces progeny 1/4 of
##  which are "dwarf" and 3/4 of which are "giant", respectively.
##  In an experiment to determine if this assumption is reasonable, a
##  cross results in progeny having 243 dwarf and 682 giant plants.
##  If "giant" is taken as success, the null hypothesis is that p =
##  3/4 and the alternative that p != 3/4.
binom.test(c(682, 243), p = 3/4)
binom.test(682, 682 + 243, p = 3/4)   # The same.
## => Data are in agreement with the null hypothesis.
```

Pairwise Wilcoxon rank sum tests

Description

Calculate pairwise comparisons between group levels with corrections for multiple testing.

Usage

```
pairwise.wilcox.test(x, g, p.adjust.method = p.adjust.methods, ...)
```

Arguments

`x`	Response vector
`g`	Grouping vector or factor
`p.adjust.method`	Method for adjusting p values (see `p.adjust`)
`...`	Additional arguments to pass to `wilcox.test`.

Value

Object of class `"pairwise.htest"`

See Also

`wilcox.test`, `p.adjust`

Examples

```
attach(airquality)
Month <- factor(Month, labels = month.abb[5:9])
## These give warnings because of ties :
pairwise.wilcox.test(Ozone, Month)
pairwise.wilcox.test(Ozone, Month, p.adj = "bonf")
detach()
```

Power calculations two sample test for proportions

Description

Compute power of test, or determine parameters to obtain target power.

Usage

```
power.prop.test(n = NULL, p1 = NULL, p2 = NULL, sig.level = 0.05,
                power = NULL,
                alternative = c("two.sided", "one.sided"),
                strict = FALSE)
```

Arguments

`n`	Number of observations (per group)
`p1`	probability in one group
`p2`	probability in other group
`sig.level`	Significance level (Type I error probability)
`power`	Power of test (1 minus Type II error probability)
`alternative`	One- or two-sided test
`strict`	Use strict interpretation in two-sided case

Details

Exactly one of the parameters `n`, `p1`, `p2`, `power`, and `sig.level` must be passed as NULL, and that parameter is determined from the others. Notice that `sig.level` has a non-NULL default so NULL must be explicitly passed if you want it computed.

If `strict = TRUE` is used, the power will include the probability of rejection in the opposite direction of the true effect, in the two-sided case. Without this the power will be half the significance level if the true difference is zero.

Value

Object of class `"power.htest"`, a list of the arguments (including the computed one) augmented with `method` and `note` elements.

Note

`uniroot` is used to solve power equation for unknowns, so you may see errors from it, notably about inability to bracket the root when invalid arguments are given. If one of them is computed `p1 < p2` will hold, although this is not enforced when both are specified.

Author(s)

Peter Dalgaard. Based on previous work by Claus Ekstrøm

See Also

`prop.test`, `uniroot`

Examples

```
power.prop.test(n = 50, p1 = .50, p2 = .75)
power.prop.test(p1 = .50, p2 = .75, power = .90)
power.prop.test(n = 50, p1 = .5, power = .90)
```

Wilcoxon Rank Sum and Signed Rank Tests

Description

Performs one and two sample Wilcoxon tests on vectors of data; the latter is also known as 'Mann-Whitney' test.

Usage

```
wilcox.test(x, ...)

## Default S3 method:
wilcox.test(x, y = NULL,
            alternative = c("two.sided", "less", "greater"),
            mu = 0, paired = FALSE, exact = NULL, correct = TRUE,
            conf.int = FALSE, conf.level = 0.95, ...)

## S3 method for class 'formula':
wilcox.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	numeric vector of data values. Non-finite (e.g. infinite or missing) values will be omitted.
`y`	an optional numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`. You can specify just the initial letter.
`mu`	a number specifying an optional parameter used to form the null hypothesis. See Details.
`paired`	a logical indicating whether you want a paired test.
`exact`	a logical indicating whether an exact *p*-value should be computed.
`correct`	a logical indicating whether to apply continuity correction in the normal approximation for the *p*-value.
`conf.int`	a logical indicating whether a confidence interval should be computed.
`conf.level`	confidence level of the interval.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is a numeric variable giving the data values and `rhs` a factor with two levels giving the corresponding groups.
`data`	an optional matrix or data frame (or similar: see `model.frame`) containing

	the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
`subset`	an optional vector specifying a subset of observations to be used.
`na.action`	a function which indicates what should happen when the data contain `NA`s. Defaults to `getOption("na.action")`.
`...`	further arguments to be passed to or from methods.

Details

The formula interface is only applicable for the 2-sample tests.

If only `x` is given, or if both `x` and `y` are given and `paired` is `TRUE`, a Wilcoxon signed rank test of the null that the distribution of `x` (in the one sample case) or of `x - y` (in the paired two sample case) is symmetric about `mu` is performed.

Otherwise, if both `x` and `y` are given and `paired` is `FALSE`, a Wilcoxon rank sum test (equivalent to the Mann-Whitney test: see the Note) is carried out. In this case, the null hypothesis is that the distributions of `x` and `y` differ by a location shift of `mu` and the alternative is that they differ by some other location shift (and the one-sided alternative `"greater"` is that `x` is shifted to the right of `y`).

By default (if `exact` is not specified), an exact *p*-value is computed if the samples contain less than 50 finite values and there are no ties. Otherwise, a normal approximation is used.

Optionally (if argument `conf.int` is true), a nonparametric confidence interval and an estimator for the pseudomedian (one-sample case) or for the difference of the location parameters `x-y` is computed. (The pseudomedian of a distribution *F* is the median of the distribution of *(u+v)/2*, where *u* and *v* are independent, each with distribution *F*. If *F* is symmetric, then the pseudomedian and median coincide. See Hollander & Wolfe (1973), page 34.) If exact *p*-values are available, an exact confidence interval is obtained by the algorithm described in Bauer (1972), and the Hodges-Lehmann estimator is employed. Otherwise, the returned confidence interval and point estimate are based on normal approximations.

With small samples it may not be possible to achieve very high confidence interval coverages. If this happens a warning will be given and an interval with lower coverage will be substituted.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value of the test statistic with a name describing it.

`parameter`	the parameter(s) for the exact distribution of the test statistic.
`p.value`	the *p*-value for the test.
`null.value`	the location parameter `mu`.
`alternative`	a character string describing the alternative hypothesis.
`method`	the type of test applied.
`data.name`	a character string giving the names of the data.
`conf.int`	a confidence interval for the location parameter. (Only present if argument `conf.int = TRUE`.)
`estimate`	an estimate of the location parameter. (Only present if argument `conf.int = TRUE`.)

Warning

This function can use large amounts of memory and stack (and even crash **R** if the stack limit is exceeded) if `exact = TRUE` and one sample is large (several thousands or more).

Note

The literature is not unanimous about the definitions of the Wilcoxon rank sum and Mann-Whitney tests. The two most common definitions correspond to the sum of the ranks of the first sample with the minimum value subtracted or not: **R** subtracts and S-PLUS does not, giving a value which is larger by *m(m+1)/2* for a first sample of size *m*. (It seems Wilcoxon's original paper used the unadjusted sum of the ranks but subsequent tables subtracted the minimum.)

R's value can also be computed as the number of all pairs `(x[i], y[j])` for which `y[j]` is not greater than `x[i]`, the most common definition of the Mann-Whitney test.

References

David F. Bauer (1972), Constructing confidence sets using rank statistics. *Journal of the American Statistical Association* **67**, 687–690.

Myles Hollander & Douglas A. Wolfe (1973), *Nonparametric statistical inference*. New York: John Wiley & Sons. Pages 27–33 (one-sample), 68–75 (two-sample).
Or second edition (1999).

See Also

`psignrank`, `pwilcox`.

`wilcox.exact` in **exactRankTests** covers much of the same ground, but also produces exact *p*-values in the presence of ties.

`wilcox_test` in package **coin** for exact and approximate *conditional p*-values for the Wilcoxon tests.

`kruskal.test` for testing homogeneity in location parameters in the case of two or more samples; `t.test` for an alternative under normality assumptions [or large samples]

Examples

```
## One-sample test.
## Hollander & Wolfe (1973), 29f.
## Hamilton depression scale factor measurements in 9 patients with
##  mixed anxiety and depression, taken at the first (x) and second
##  (y) visit after initiation of a therapy (administration of a
##  tranquilizer).
x <- c(1.83,  0.50,  1.62,  2.48, 1.68, 1.88, 1.55, 3.06, 1.30)
y <- c(0.878, 0.647, 0.598, 2.05, 1.06, 1.29, 1.06, 3.14, 1.29)
wilcox.test(x, y, paired = TRUE, alternative = "greater")
wilcox.test(y - x, alternative = "less")    # The same.
wilcox.test(y - x, alternative = "less",
            exact = FALSE, correct = FALSE) # H&W large sample
                                            # approximation

## Two-sample test.
## Hollander & Wolfe (1973), 69f.
## Permeability constants of the human chorioamnion (a placental
##  membrane) at term (x) and between 12 to 26 weeks gestational
##  age (y).  The alternative of interest is greater permeability
##  of the human chorioamnion for the term pregnancy.
x <- c(0.80, 0.83, 1.89, 1.04, 1.45, 1.38, 1.91, 1.64, 0.73, 1.46)
y <- c(1.15, 0.88, 0.90, 0.74, 1.21)
wilcox.test(x, y, alternative = "g")        # greater
wilcox.test(x, y, alternative = "greater",
            exact = FALSE, correct = FALSE) # H&W large sample
                                            # approximation

wilcox.test(rnorm(10), rnorm(10, 2), conf.int = TRUE)

## Formula interface.
boxplot(Ozone ~ Month, data = airquality)
wilcox.test(Ozone ~ Month, data = airquality,
            subset = Month %in% c(5, 8))
```

ORIGIN ≡ 0

Sign Test

The Sign Test is a nonparametric analog to the paired t Test. It requires the use of ordinal data - data that can be ordered but has no specific numerical values. Of course, ordinal data can be constructed from cardinal data - metric data to which standard arithmetic and measuring distances apply. Conversion is typically done when a parametric test violates the underlying assumption of normality. However, doing so involves loss of information and, as a result, lessens power of the test.

Assumptions:

- Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n}$ across individuals 1,2,3, ... ,n.
- Let the value $d_i = X_{2,i} - X_{1,i}$ for each individual i be assessed as $|d_i|$ = rank order of single observations or discrete classes of observations with observed frequency.
- The d_i's are independent.
- The underlying distribution of the d_i's is continuous & symmetric but not necessarily a Normal Distribution.
- All d_i's have the same median

Hypotheses:

$H_0: \Delta = 0$ < No population ordinal difference in median

$H_1: \Delta <> 0$ < Two sided test

Criterion for Normal Approximation:

- IF number of non-zero d_i < 20 THEN use Exact Method
  OTHERWISE Normal Approximation may be used

Normal Approximation:

Test Statistic:

C = number of d_i's where d_i is +

D = number of d_i's where d_i is - < used only for simplified calculation of Probability below

Critical Value of the Test:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$$c_1 := \frac{n}{2} + \frac{1}{2} + \sqrt{\frac{n}{4}} \cdot \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}, 0, 1\right) \qquad c_2 := \left(\frac{n}{2} - \frac{1}{2}\right) - \sqrt{\frac{n}{4}} \cdot \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}, 0, 1\right)$$

$$c_1 := \frac{n}{2} + \frac{1}{2} + \sqrt{\frac{n}{4}} \cdot \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right) \qquad c_2 := \left(\frac{n}{2} - \frac{1}{2}\right) - \sqrt{\frac{n}{4}} \cdot \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$$

Decision Rule:

IF C outside interval CV = (c_1, c_2) THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

$$P := 2 \cdot \left(1 - \Phi_N\left(\frac{|C - D| - 1}{\sqrt{n}}\right)\right)$$ **< for C <> D OTHERWISE P = 1.0**

$$P := 2 \cdot \left(1 - pnorm\left(\frac{|C - D| - 1}{\sqrt{n}}, 0, 1\right)\right)$$

Example:

Dermatology Example Rosner Ex. 9.8 p. 364:
Arm A > B = 22
Arm B < A = 18
Arm A = B = 5

$n := 22 + 18$ $n = 40$

Paired SampleAssumptions:

- Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.
- Let the value $d_i = X_{2,i} - X_{1,i}$ for each individual i only be assessed as +, -, or =.

Hypotheses:

H_0: Δ = 0 **< No population ordinal difference**
H_1: Δ <> 0 **< Two sided test**

Criterion for Normal Approximation:

- IF number of non-zero d_i < 20 THEN use Exact Method OTHERWISE Normal Approximation may be used

$n = 40$ **< qualifies!**

Normal Approximation:

Test Statistic:

$C := 18$ $D := 22$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$c_1 := \frac{n}{2} + \frac{1}{2} + \sqrt{\frac{n}{4}} \cdot qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right) \qquad c_2 := \left(\frac{n}{2} - \frac{1}{2}\right) - \sqrt{\frac{n}{4}} \cdot qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right)$$

$c_1 = 26.698$ $c_2 = 13.302$ $CV := (c_1 \quad c_2)$

Decision Rule:

IF C outside interval CV = (c_1, c_2) THEN REJECT H_0 OTHERWISE ACCEPT H_0

$C = 18$ $CV = (26.698 \quad 13.302)$ **< confirmed p. 364**

Probability Value:

$$P := 2 \cdot \left(1 - pnorm\left(\frac{|C - D| - 1}{\sqrt{n}}, 0, 1\right)\right)$$ $P = 0.6353$

^ confirmed p. 364

Exact Method:

Test Statistic:

C = number of d_i's where d_i is +

Probability Value:

IF C > n/2

$$P := 2 \cdot \sum_{k=C}^{n} \text{combin}(n,k) \cdot \left(\frac{1}{2}\right)^n$$

IF C < n/2

$$P := 2 \cdot \sum_{k=0}^{C} \text{combin}(n,k) \cdot \left(\frac{1}{2}\right)^n$$

OTHERWISE C = n/2 and P = 1.0

Example:

Ophthalmology Example Rosner Ex. 9.9 p. 365-366:
Drug A better than B = 8
Drug B better than A = 2
Drugs A & B equal = 5

$n := 8 + 2$ $n = 10$

Paired SampleAssumptions:

- Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.
- Let the value $d_i = X_{2,i} - X_{1,i}$ for each individual i only be assessed as +, -, or =.

Hypotheses:

H_0: $\Delta = 0$ **< No population ordinal difference**
H_1: $\Delta <> 0$ **< Two sided test**

Criterion for Normal Approximation:

- IF number of non-zero d_i < 20 THEN use Exact Method
OTHERWISE Normal Approximation may be used $n = 10$ **< NOT qualified!**

Binomial Exact Calculation:

Test Statistic:

$C := 8$

Probability Value:

$$P := 2 \cdot \sum_{k=C}^{n} \text{combin}(n,k) \cdot \left(\frac{1}{2}\right)^n$$ **< for C > n/2** $P = 0.1094$

confirmed p. 366 ^

Note that if $\alpha = 0.05$ then P > 0.05 and we do not reject H_0

Be sure you can use Table 1 in the Appendix to do this!

Prototype in Systat & R:

X := READPRN("C:/2007BiostatsData/AnorexiaALL.txt")

$n := \text{length}\left(X^{\langle 0 \rangle}\right)$ n = 72

$d := \text{sort}\left(X^{\langle 0 \rangle} - X^{\langle 1 \rangle}\right)$

C := 42 n := 71 D := n − C D = 29

X =

	0	1
0	80.7	80.2
1	89.4	80.1
2	91.8	86.4
3	74	86.3
4	78.1	76.1
5	88.3	78.1
6	87.3	75.1
7	75.1	86.7
8	80.6	73.5
9	78.4	84.6
10	77.6	77.4
11	88.7	79.5
12	81.3	89.6
13	78.1	81.4
14	70.5	81.8
15	77.3	77.3

d =

	0
0	-21.5
1	-20.9
2	-17.1
3	-15.9
4	-15.4
5	-14.9
6	-13.6
7	-13.4
8	-13.1
9	-12.6
10	-12.3
11	-11.7
12	-11.6
13	-11.4
14	-11.3
15	-11

Paired SampleAssumptions:

- Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n}$ across individuals 1,2,3, ... ,n.

- Let the value $d_i = X_{2,i} - X_{1,i}$ for each individual i only be assessed as +, -, or =.

Hypotheses:

H_0: Δ = 0 **< No population ordinal difference**

H_1: Δ <> 0 **< Two sided test**

Criterion for Normal Approximation:

- IF number of non-zero d_i < 20 THEN use Exact Method

OTHERWISE Normal Approximation may be used n = 71 **< qualified!**

Normal Approximation:

Test Statistic:

C = 42 D = 29

Critical Value of the Test:

α := 0.05 **< Probability of Type I error must be explicitly set**

$c_1 := \frac{n}{2} + \frac{1}{2} + \sqrt{\frac{n}{4}} \cdot \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$ $c_2 := \left(\frac{n}{2} - \frac{1}{2}\right) - \sqrt{\frac{n}{4}} \cdot \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$

$c_1 = 44.2575$ $c_2 = 26.7425$ $CV := \left(c_1 \;\; c_2\right)$

Decision Rule:

IF C outside interval CV = (c_1, c_2) THEN REJECT H_0 OTHERWISE ACCEPT H_0

C = 42 CV = (44.2575 26.7425)

Probability Value:

$$P := 2 \cdot \left(1 - \text{pnorm}\left(\frac{|C - D| - 1}{\sqrt{n}}, 0, 1\right)\right) \qquad P = 0.1544$$

Binomial Exact Calculation:

Test Statistic:

C = 42

Probability Value:

$$P := 2 \cdot \sum_{k=C}^{n} \text{combin}(n, k) \cdot \left(\frac{1}{2}\right)^n$$ **< for C > n/2** P = 0.1539

Systat Results:

$X^{\langle 0 \rangle}$ **< before**

$X^{\langle 1 \rangle}$ **< after**

Sign test results

Counts of differences (row variable greater than column)

	BEFORE	AFTER
BEFORE	**0**	**29**
AFTER	**42**	**0**

Two-sided probabilities for each pair of variables

	BEFORE	AFTER
BEFORE	**1.0000000**	
AFTER	**0.1544065**	**1.0000000**

^ My guess is the SYSTAT uses the Normal Approximation here!

R Results:

COMMANDS:

```
> X=read.table("C:/2007BiostatsData/anorexia.txt")
> X
> binom.test(42,71,p=0.5,alternative="two.sided",conf.level=0.95)
```

Exact binomial test

data: 42 and 71
number of successes = 42, number of trials = 71, p-value = 0.1539
alternative hypothesis: true probability of success is not equal to 0.5
95 percent confidence interval:
0.4684018 0.7068122
sample estimates:
probability of success
0.5915493

^ Here I had to count by hand C = number of + or "successes" out of n trials without tie.

The results are a straight-forward binomial test for binomial parameter p = 0.

ORIGIN ≡ 0

Wilcoxon Signed-Rank Test

The Signed-Rank Test is a nonparametric analog to the paired t Test utilizing more information than available in the Sign Test. The Signed-Rank test requires use of ordinal data that can be ordered, or ranked, according to amount of effect. However, amount of effect need not have meaning beyond order of classes of data.

Assumptions:

- Observed values $X_{1,1}, X_{1,2}, X_{1,3}, \ldots X_{1,n}$ are a random sample exactly matched with Observed values $X_{2,1}, X_{2,2}, X_{2,3}, \ldots X_{2,n}$ across individuals 1,2,3, ... ,n.
- Let the value $d_i = X_{2,i} - X_{1,i}$ for each individual i be assessed as $|d_i|$ = rank order of single observations or discrete classes of observations with observed frequency.
- The d_i's are independent.
- The underlying distribution of the d_i's is continuous & symmetric but not necessarily a Normal Distribution.
- All d_i's have the same median

Hypotheses:

$H_0: \Delta = 0$ < No population ordinal difference in median

$H_1: \Delta <> 0$ < Two sided test

Criterion for Normal Approximation:

- IF number of non-zero d_i < 16 THEN use Special Tables e.g., Rosner Table 11 in Appendix OTHERWISE Normal Approximation may be used

Normal Approximation:

Rank Data and Sum:

- Ignore all d_i's = 0 - Don't include them in the rankings.
- The $|d_i|$'s are ranked (R_i = rank($|d_i|$) according to their absolute value with smallest $|d_i| = 1$ and largest $|d_i| = n$.
- Give all d_i's with same absolute value the same average rank.
- Count number of ties (t_j) for each group (g) of ties for the d_i's
- Compute the Rank Sum [RS_{pos}] of positive di's.

Test Statistic:

IF RS_{pos} <> n(n+1)/4 AND there are NO ties THEN:

$$T := \frac{\left[\left|RS_{pos} - \frac{n \cdot (n+1)}{4}\right| - \frac{1}{2}\right]}{\sqrt{\frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{24}}}$$

IF RS_{pos} <> n(n+1)/4 AND there ARE ties THEN:

$$T := \frac{\left[\left|RS_{pos} - \frac{n \cdot (n+1)}{4}\right| - \frac{1}{2}\right]}{\sqrt{\frac{n \cdot (n+1) \cdot (2 \cdot n+1)}{24} - \frac{\sum\left(t^3 - t\right)}{48}}}$$

where t is the number (count) of members of each class

IF $RS_{pos} = n(n+1)/4$ THEN:

$T := 0$

GENERAL ALTERNATIVE TO THE ABOVE:

$j := 1 .. g$

$$T := \frac{\left[\left|RS_{pos} - \frac{n \cdot (n+1)}{4}\right| - \frac{1}{2}\right]}{\sqrt{\sum_j \frac{\left[t_j \cdot \left(AR_j\right)^2\right]}{4}}}$$

where AR_j represents the average rank of each class j

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}\right)$ $C := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$

Decision Rule:

IF T > C THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

$P := 2 \cdot \left(1 - \Phi_N(T)\right)$ $P := 2 \cdot (1 - \text{pnorm}(T, 0, 1))$

Example:

Dermatology Example
Rosner Ex 9.12, p. 370

$d_{neg} := \begin{pmatrix} -8 \\ -7 \\ -6 \\ -5 \\ -4 \\ -3 \\ -2 \\ -1 \end{pmatrix}$ $count_{neg} := \begin{pmatrix} 1 \\ 3 \\ 2 \\ 2 \\ 1 \\ 5 \\ 4 \\ 4 \end{pmatrix}$ $d_{pos} := \begin{pmatrix} 8 \\ 7 \\ 6 \\ 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix}$ $count_{pos} := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 2 \\ 6 \\ 10 \end{pmatrix}$

$R := d_{pos}$

$count := count_{neg} + count_{pos}$

$i := 0 .. \text{length}(R) - 1$

$n := \sum_i count_i$ $n = 40$

$R = \begin{pmatrix} 8 \\ 7 \\ 6 \\ 5 \\ 4 \\ 3 \\ 2 \\ 1 \end{pmatrix}$ $count = \begin{pmatrix} 1 \\ 3 \\ 2 \\ 2 \\ 1 \\ 7 \\ 10 \\ 14 \end{pmatrix}$ $AR := \begin{pmatrix} 40 \\ 38 \\ 35.5 \\ 33.5 \\ 32 \\ 28 \\ 19.5 \\ 7.5 \end{pmatrix}$

< average rank: (40+40)/2
< average rank: (37-39)/2
< average rank: (35+36)/2
< average rank: (33-34)/2
< average rank:(32+32)/2
< average rank:(25+31)/2
< average rank:(15+24)/2
< average rank:(1+14)/2

$RS_{pos} := 10 \cdot 7.5 + 6 \cdot 19.5 + 2 \cdot 28$ $RS_{pos} = 248$ **< sum of the ranks for positive d_i's**

^ verified p. 370

$\frac{n \cdot (n+1)}{4} = 410$ **< criterion for test statistic T** **< expected rank sum verified p. 370**

Criterion for Normal Approximation:

- IF number of non-zero d_i < 16 THEN use Special Tables e.g., Rosner Table 11 in Appendix
OTHERWISE Normal Approximation may be used

$n = 40$ **< qualifies for Normal Approximation**

$t := \text{count}$

$St := \sum\left(t^3 - t\right)$ $St = 4092$

^ note use of vector sum function here that adds all elements of a vector together...

$$t^3 = \begin{pmatrix} 1 \\ 27 \\ 8 \\ 8 \\ 1 \\ 343 \\ 1000 \\ 2744 \end{pmatrix} \qquad t = \begin{pmatrix} 1 \\ 3 \\ 2 \\ 2 \\ 1 \\ 7 \\ 10 \\ 14 \end{pmatrix}$$

Test Statistic:

IF RS_{pos} <> n(n+1)/4 AND there ARE ties THEN:

$$T := \frac{\left[\left|RS_{pos} - \frac{n\cdot(n+1)}{4}\right| - \frac{1}{2}\right]}{\sqrt{\frac{n\cdot(n+1)\cdot(2\cdot n+1)}{24} - \frac{\sum\left(t^3 - t\right)}{48}}} \qquad T = 2.1877$$

$$\frac{n\cdot(n+1)\cdot(2\cdot n+1)}{24} = 5535$$

^ verified p. 370

$$\frac{n\cdot(n+1)\cdot(2\cdot n+1)}{24} - \frac{\sum\left(t^3 - t\right)}{48} = 5449.75$$

^ verified p. 370

GENERAL ALTERNATIVE TO THE ABOVE:

$j := 0 \,..\, \text{length}(AR) - 1$

$$\left[\left|RS_{pos} - \frac{n\cdot(n+1)}{4}\right| - \frac{1}{2}\right] = 161.5$$

^ verified p. 370

$$T := \frac{\left[\left|RS_{pos} - \frac{n\cdot(n+1)}{4}\right| - \frac{1}{2}\right]}{\sqrt{\sum_j \frac{\left[t_j\cdot\left(AR_j\right)^2\right]}{4}}} \qquad T = 2.1877$$

^ verified p. 370

$$\sum_j \frac{\left[t_j\cdot\left(AR_j\right)^2\right]}{4} = 5449.75$$

^ verified p. 370

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$ $C = 1.96$

Decision Rule:

IF T > C THEN REJECT H_0 OTHERWISE ACCEPT H_0

$T = 2.1877$ $C = 1.96$

Probability Value:

$P := 2\cdot(1 - \text{pnorm}(T, 0, 1))$ $P = 0.0287$ **< verified p. 371**

Example:

Dermatology Example
Rosner Ex 9.13, p. 371
Using R_{neg} instead of R_{pos}

$$d_{neg} := \begin{pmatrix} -8 \\ -7 \\ -6 \\ -5 \\ -4 \\ -3 \\ -2 \\ -1 \end{pmatrix} \quad count_{neg} := \begin{pmatrix} 1 \\ 3 \\ 2 \\ 2 \\ 1 \\ 5 \\ 4 \\ 4 \end{pmatrix} \quad AR := \begin{pmatrix} 40 \\ 38 \\ 35.5 \\ 33.5 \\ 32 \\ 28 \\ 19.5 \\ 7.5 \end{pmatrix}$$

$i := 0 .. length(d_{neg}) - 1$

$R_{neg_i} := AR_i \cdot count_{neg_i}$

$RS_{neg} := \left| \sum R_{neg} \right| \qquad RS_{neg} = 572$

^ verified p. 371

Criterion for Normal Approximation:

- IF number of non-zero d_i < 16 THEN use Special Tables e.g., Rosner Table 11 in Appendix
OTHERWISE Normal Approximation may be used

$n = 40$ **< qualifies for Normal Approximation**

Test Statistic:

IF RS_{pos} <> n(n+1)/4 AND there ARE ties THEN:

$$T := \frac{\left[\left| RS_{neg} - \frac{n \cdot (n+1)}{4} \right| - \frac{1}{2} \right]}{\sqrt{\frac{n \cdot (n+1) \cdot (2 \cdot n + 1)}{24} - \frac{\sum \left(t^3 - t \right)}{48}}} \qquad T = 2.1877$$

< verified p. 371

GENERAL ALTERNATIVE TO THE ABOVE:

Note Same values for T as above...

$j := 0 .. length(AR) - 1$

$$T := \frac{\left[\left| RS_{neg} - \frac{n \cdot (n+1)}{4} \right| - \frac{1}{2} \right]}{\sqrt{\sum_j \frac{\left[t_j \cdot \left(AR_j \right)^2 \right]}{4}}} \qquad T = 2.1877$$

< verified p. 371

SYSTAT Prototype:

From above:

```
Wilcoxon Signed Ranks Test Results

  Counts of differences (row variable greater
than column)

          BEFORE  AFTER
BEFORE     0            18
AFTER      22           0

  Z = (Sum of signed ranks)/square root(sum of
squared ranks)

           BEFORE AFTER
BEFORE     0.0000000
AFTER      2.1944551          0.0000000

  Two-sided probabilities using normal
approximation

            BEFORE      AFTER
BEFORE     1.0000000
AFTER       0.0282027         1.0000000
```

$\sum \text{count}_{pos} = 18$

$\sum \text{count}_{neg} = 22$

< GENERAL ALTERNATIVE T value here

$P = 0.0287$

^ Probability approximately matches

R Prototype:

COMMANDS:

```
> Cooked=read.table("c:\2007BiostatsData\Rosner Ex 9.12 Cooked.txt")
> Cooked
> attach(cooked)
> wilcox.test(Before,After,alternative="two.sided",paired=T)
```

From above:

Wilcoxon signed rank test with continuity correction

data: Before and After
V = 248, p-value = 0.02869
alternative hypothesis: true location shift is not equal to 0

$RS_{pos} = 248$

$P = 0.0287$

Warning message:
cannot compute exact p-value with ties in:
wilcox.test.default(Before, After, alternative = "two.sided",

Example:

X := READPRN("C:/2007BiostatsData/AnorexiaALL.txt")

$n := \text{length}\left(X^{\langle 0 \rangle}\right)$ n = 72

$d := \text{sort}\left(X^{\langle 0 \rangle} - X^{\langle 1 \rangle}\right)$

X =

	0	1
0	80.7	80.2
1	89.4	80.1
2	91.8	86.4
3	74	86.3
4	78.1	76.1
5	88.3	78.1
6	87.3	75.1
7	75.1	86.7
8	80.6	73.5
9	78.4	84.6
10	77.6	77.4
11	88.7	79.5
12	81.3	89.6
13	78.1	81.4
14	70.5	81.8
15	77.3	77.3

d =

	0
0	-21.5
1	-20.9
2	-17.1
3	-15.9
4	-15.4
5	-14.9
6	-13.6
7	-13.4
8	-13.1
9	-12.6
10	-12.3
11	-11.7
12	-11.6
13	-11.4
14	-11.3
15	-11

Criterion for Normal Approximation:

- IF number of non-zero d_i < 16 THEN use Special Tables e.g., Rosner Table 11 in Appendix OTHERWISE Normal Approximation may be used

n = 72 **< qualifies for Normal Approximation**

i := 43 .. 71

$RS_{pos} := (71 - 42)$ $RS_{pos} = 29$

$RS_{neg} := (42)$ $RS_{neg} = 42$

SYSTAT Results:

Wilcoxon Signed Ranks Test Results

Counts of differences (row variable greater than column)

	BEFORE	AFTER
BEFORE	**0**	**29**
AFTER	**42**	**0**

Z = (Sum of signed ranks)/square root(sum of squared ranks)

	BEFORE	AFTER
BEFORE	**0.0000000**	
AFTER	**2.5612838**	**0.0000000**

Two-sided probabilities using normal approximation

	BEFORE	AFTER
BEFORE	**1.0000000**	
AFTER	**0.0104286**	**1.0000000**

ORIGIN ≡ 0

Wilcoxon Rank-Sum Test
Mann-Whitney Test

These fully equivalent procedures are the nonparametric analog to the two-sample t Test. They are applied when analyzing independent samples from two populations without assuming an underlying Normal distribution for each. Thus they may be applied to most/all situations one might normally apply a parametric solution, but with fewer assumptions and less power.

Assumptions:

- **Observed values $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... $X_{1,n1}$ are a random sample**
  Observed values $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... $X_{2,n2}$ are a random sample.
- **Variables X_1's and X_2 are independent.**
- **Underlying distributions are continuous.**
- **Measurement scale is at least ordinal - i.e, data can be ranked.**

Hypotheses:

H_0: $\Delta = 0$ < **No population ordinal difference in median**

H_1: $\Delta <> 0$ < **Two sided test**

Criterion for Normal Approximation:

- **IF** $(n_1 \geq 10) \wedge (n_2 \geq 10)$ **THEN Normal Approximation may be used**
  OTHERWISE use Special Tables e.g., Rosner Table 11 in Appendix

Normal Approximation:

Rank Data and Sum:

- **Pool Data and Rank observations.**
- **Compute Rank Sum (RS_1 or RS_2) of one population (doesn't matter which).**

Test Statistic:

IF $RS_1 <> n_1(n_1+n_2+1)/2$ AND there are NO ties THEN:

$$T := \frac{\left[\left|RS_1 - \frac{n_1 \cdot (n_1 + n_2 + 1)}{2}\right| - \frac{1}{2}\right]}{\sqrt{\left(\frac{n_1 \cdot n_2}{12}\right) \cdot (n_1 + n_2 + 1)}}$$

IF $RS_1 <> n_1(n_1+n_2+1)/2$ AND there ARE ties THEN:

$$T := \frac{\left[\left|RS_1 - \frac{n_1 \cdot (n_1 + n_2 + 1)}{2}\right| - \frac{1}{2}\right]}{\sqrt{\left(\frac{n_1 \cdot n_2}{12}\right) \cdot \left[n_1 + n_2 + 1 - \sum_i \frac{t_i \cdot \left[(t_i)^2 - 1\right]}{(n_1 + n_2) \cdot (n_1 + n_2 - 1)}\right]}}$$

where:

t = number of tied individuals in each class or group.

i = is used to sum across all classes or groups.

IF $RS_1 = n_1(n_1+n_2+1)/2$ THEN:

$$T := 0$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}\right)$ $C := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$

Decision Rule:

IF T > C THEN REJECT H_0

OTHERWISE ACCEPT H_0

Probability Value:

$P := 2 \cdot \left(1 - \Phi_N(T)\right)$ $P := 2 \cdot (1 - \text{pnorm}(T, 0, 1))$

Example:

Ophthalmology Example
Rosner Ex 9.17, p. 375

$$X := \begin{pmatrix} 5 & 1 \\ 9 & 5 \\ 6 & 4 \\ 3 & 4 \\ 2 & 8 \\ 0 & 5 \\ 0 & 2 \\ 0 & 1 \end{pmatrix}$$

$n_1 := \sum X^{\langle 0 \rangle}$ $n_1 = 25$

$n_2 := \sum X^{\langle 1 \rangle}$ $n_2 = 30$

$t := X^{\langle 0 \rangle} + X^{\langle 1 \rangle}$

$$t = \begin{pmatrix} 6 \\ 14 \\ 10 \\ 7 \\ 10 \\ 5 \\ 2 \\ 1 \end{pmatrix}$$

$i := 0 .. \text{length}(t) - 1$

$$AR := \begin{pmatrix} 3.5 \\ 13.5 \\ 25.5 \\ 34.0 \\ 42.5 \\ 50.0 \\ 53.5 \\ 55.0 \end{pmatrix}$$

$\frac{1 + 2 + 3 + 4 + 5 + 6}{6} = 3.5$

$j := 7 .. 20$

$\frac{\sum j}{j} = 13.5$

$R_{1_i} := \left(X^{\langle 0 \rangle}\right)_i \cdot AR_i$

$R_{2_i} := \left(X^{\langle 1 \rangle}\right)_i \cdot AR_i$

$$R_1 = \begin{pmatrix} 17.5 \\ 121.5 \\ 153 \\ 102 \\ 85 \\ 0 \\ 0 \\ 0 \end{pmatrix} \quad R_2 = \begin{pmatrix} 3.5 \\ 67.5 \\ 102 \\ 136 \\ 340 \\ 250 \\ 107 \\ 55 \end{pmatrix}$$

$RS_1 := \sum R_1$ $RS_1 = 479$

$RS_2 := \sum R_2$ $RS_2 = 1061$

$n_1 \cdot \left(\frac{n_1 + n_2 + 1}{2}\right) = 700$

$n_2 \cdot \left(\frac{n_1 + n_2 + 1}{2}\right) = 840$

Criterion for Normal Approximation:

- IF $(n_1 \geq 10) \wedge (n_2 \geq 10)$ THEN Normal Approximation may be used
OTHERWISE use Special Tables e.g., Rosner Table 12 in Appendix

$n_1 = 25$ $n_2 = 30$ **< qualifies for Normal Approximation**

Test Statistic:

IF RS_1 <> $n_1(n_1+n_2+1)/2$ AND there ARE ties THEN:

$$T := \frac{\left[\left|RS_1 - \frac{n_1 \cdot (n_1 + n_2 + 1)}{2}\right| - \frac{1}{2}\right]}{\sqrt{\left(\frac{n_1 \cdot n_2}{12}\right) \cdot \left[n_1 + n_2 + 1 - \sum_i \frac{t_i \cdot \left[(t_i)^2 - 1\right]}{(n_1 + n_2) \cdot (n_1 + n_2 - 1)}\right]}}$$

$$t = \begin{pmatrix} 6 \\ 14 \\ 10 \\ 7 \\ 10 \\ 5 \\ 2 \\ 1 \end{pmatrix}$$

T = 3.7889 **< verified p. 375**

IF RS_2 <> $n_2(n_1+n_2+1)/2$ AND there ARE ties THEN:

$$T := \frac{\left[\left|RS_2 - \frac{n_2 \cdot (n_1 + n_2 + 1)}{2}\right| - \frac{1}{2}\right]}{\sqrt{\left(\frac{n_1 \cdot n_2}{12}\right) \cdot \left[n_1 + n_2 + 1 - \sum_i \frac{t_i \cdot \left[(t_i)^2 - 1\right]}{(n_1 + n_2) \cdot (n_1 + n_2 - 1)}\right]}}$$

T = 3.7889 **< same**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right)$ C = 1.96

Decision Rule:

IF T > C THEN REJECT H_0 OTHERWISE ACCEPT H_0

T = 3.7889 C = 1.96

Probability Value:

$P := 2 \cdot (1 - pnorm(T, 0, 1))$ P = 0.0002 **< verified p. 376**

SYSTAT Prototype:

Mann-Whitney Test is found as a subset under the Kruskal-Wallis nonparametricANOVA analog.

$n_1 = 25$ $RS_1 = 479$

$n_2 = 30$ $RS_2 = 1061$

P = 0.0002 **< approximately the same >**

Note that Mann-Whitney U is not explicitly calculated here...

Categorical values encountered during processing are:
GROUP (2 levels)
1, 2

Kruskal-Wallis One-Way Analysis of Variance for 55 cases
Dependent variable is CLASS
Grouping variable is GROUP

Group	Count	Rank Sum
1	25	479.0000000
2	30	1.06100E+03

Mann-Whitney U test statistic = 154.0000000
Probability is 0.0001461
Chi-square approximation = 14.4212324 with 1 df

R Prototype:

COMMANDS:

```
> X=read.table("c:/2007BiostatsData/Rosner Ex 9.17 Cooked.txt")
> X
> attach(X)
> wilcox.test(Dominant,SexLinked,paired=F,mu=0,alternative="two.sided")
```

Wilcoxon rank sum test with continuity correction

data: Dominant and SexLinked
W = 154, p-value = 9.62e-06
alternative hypothesis: true location shift is not equal to 0

Warning message:
cannot compute exact p-value with ties in: wilcox.test.default(Dominant, SexLinked, paired = F, mu = 0,

^ according to the documentation for wilcox.test() explicit calculation of the test statistic W is made if the samples contain less than 50 values and there are no ties.

Results show a small (but not the same) P value as expected, and statistic W doesn't match! See R's documentation about this... and below

$W := 154$ **< W from R & SYSTAT's Mann-Whitney U above**

$cf := \frac{n_1 \cdot (n_1 + 1)}{2}$ $cf = 325$ **< correction factor indicated in documentation**

$W + cf = 479$ $RS_1 = 479$ **< W + cf is the same as our RS_1**

Assignment for Week 9

Today there will be no formal assignment. Enjoy your week off!

On Tuesday after the break, however, there will be an

unannounced-pop-*take-home* quiz

covering all material you might expect to see on the second exam the following week. So, if you have a little time, take a look at the parametric and non-parametric tests. A good way to be sure you can work exam problems is to set yourself the task of performing an analysis by hand. Given the data in Quiz 4, or anything similar, you should be able to distinguish the tests and perform the following:

- one sample t-test of mean
- paired t-test of mean
- two-sample t-test of mean on populations with equal variances
- two-sample t-test of mean on populations with unequal variances
- F test for equal variances in two populations
- estimate power and sample size in *both* single population and two population tests
- one sample test of parameter p (probability of "heads") in a binomial population using the Normal Approximation
- Sign test for paired non-Normal populations design

For the following tests, devise a simple contingency chart and see if you can perform:

- Wilcoxon signed-rank test for paired non-Normal populations design
- Wilcoxon Rank-sum = Mann-Wittney Test for mean of two populations
- 2X2 Contingency test
- McNemar's Test for Paired data
- Chi-square Test for Association in RXC Contingency Tables
- Chi-square Goodness of Fit test

For all of the above tests, be sure you can **state clearly *all*** of the ***formal structure*** of each test such as **Assumptions**, **Model**, **Hypotheses**, **Criterion for Normal Approximation**, **Decision Rule**, and **Result**.

Exact Binomial Test

Description

Performs an exact test of a simple null hypothesis about the probability of success in a Bernoulli experiment.

Usage

```
binom.test(x, n, p = 0.5,
           alternative = c("two.sided", "less", "greater"),
           conf.level = 0.95)
```

Arguments

`x`	number of successes, or a vector of length 2 giving the numbers of successes and failures, respectively.
`n`	number of trials; ignored if `x` has length 2.
`p`	hypothesized probability of success.
`alternative`	indicates the alternative hypothesis and must be one of `"two.sided"`, `"greater"` or `"less"`. You can specify just the initial letter.
`conf.level`	confidence level for the returned confidence interval.

Details

Confidence intervals are obtained by a procedure first given in Clopper and Pearson (1934). This guarantees that the confidence level is at least `conf.level`, but in general does not give the shortest-length confidence intervals.

Value

A list with class `"htest"` containing the following components:

`statistic`	the number of successes.
`parameter`	the number of trials.
`p.value`	the p-value of the test.
`conf.int`	a confidence interval for the probability of success.
`estimate`	the estimated probability of success.
`null.value`	the probability of success under the null, `p`.

`alternative` a character string describing the alternative hypothesis.

`method` the character string `"Exact binomial test"`.

`data.name` a character string giving the names of the data.

References

Clopper, C. J. & Pearson, E. S. (1934). The use of confidence or fiducial limits illustrated in the case of the binomial. *Biometrika*, **26**, 404–413.

William J. Conover (1971), *Practical nonparametric statistics*. New York: John Wiley & Sons. Pages 97–104.

Myles Hollander & Douglas A. Wolfe (1973), *Nonparametric statistical inference*. New York: John Wiley & Sons. Pages 15–22.

See Also

`prop.test` for a general (approximate) test for equal or given proportions.

Examples

```
## Conover (1971), p. 97f.
## Under (the assumption of) simple Mendelian inheritance, a cross
##  between plants of two particular genotypes produces progeny 1/4 of
##  which are "dwarf" and 3/4 of which are "giant", respectively.
##  In an experiment to determine if this assumption is reasonable, a
##  cross results in progeny having 243 dwarf and 682 giant plants.
##  If "giant" is taken as success, the null hypothesis is that p =
##  3/4 and the alternative that p != 3/4.
binom.test(c(682, 243), p = 3/4)
binom.test(682, 682 + 243, p = 3/4)   # The same.
## => Data are in agreement with the null hypothesis.
```

Wilcoxon Rank Sum and Signed Rank Tests

Description

Performs one and two sample Wilcoxon tests on vectors of data; the latter is also known as 'Mann-Whitney' test.

Usage

```
wilcox.test(x, ...)

## Default S3 method:
wilcox.test(x, y = NULL,
            alternative = c("two.sided", "less", "greater"),
            mu = 0, paired = FALSE, exact = NULL, correct = TRUE,
            conf.int = FALSE, conf.level = 0.95, ...)

## S3 method for class 'formula':
wilcox.test(formula, data, subset, na.action, ...)
```

Arguments

`x`	numeric vector of data values. Non-finite (e.g. infinite or missing) values will be omitted.
`y`	an optional numeric vector of data values.
`alternative`	a character string specifying the alternative hypothesis, must be one of `"two.sided"` (default), `"greater"` or `"less"`. You can specify just the initial letter.
`mu`	a number specifying an optional parameter used to form the null hypothesis. See Details.
`paired`	a logical indicating whether you want a paired test.
`exact`	a logical indicating whether an exact *p*-value should be computed.
`correct`	a logical indicating whether to apply continuity correction in the normal approximation for the *p*-value.
`conf.int`	a logical indicating whether a confidence interval should be computed.
`conf.level`	confidence level of the interval.
`formula`	a formula of the form `lhs ~ rhs` where `lhs` is a numeric variable giving the data values and `rhs` a factor with two levels giving the corresponding groups.
`data`	an optional matrix or data frame (or similar: see `model.frame`) containing

	the variables in the formula `formula`. By default the variables are taken from `environment(formula)`.
`subset`	an optional vector specifying a subset of observations to be used.
`na.action`	a function which indicates what should happen when the data contain `NA`s. Defaults to `getOption("na.action")`.
`...`	further arguments to be passed to or from methods.

Details

The formula interface is only applicable for the 2-sample tests.

If only `x` is given, or if both `x` and `y` are given and `paired` is `TRUE`, a Wilcoxon signed rank test of the null that the distribution of `x` (in the one sample case) or of `x - y` (in the paired two sample case) is symmetric about `mu` is performed.

Otherwise, if both `x` and `y` are given and `paired` is `FALSE`, a Wilcoxon rank sum test (equivalent to the Mann-Whitney test: see the Note) is carried out. In this case, the null hypothesis is that the distributions of `x` and `y` differ by a location shift of `mu` and the alternative is that they differ by some other location shift (and the one-sided alternative `"greater"` is that `x` is shifted to the right of `y`).

By default (if `exact` is not specified), an exact *p*-value is computed if the samples contain less than 50 finite values and there are no ties. Otherwise, a normal approximation is used.

Optionally (if argument `conf.int` is true), a nonparametric confidence interval and an estimator for the pseudomedian (one-sample case) or for the difference of the location parameters `x-y` is computed. (The pseudomedian of a distribution *F* is the median of the distribution of *(u+v)/2*, where *u* and *v* are independent, each with distribution *F*. If *F* is symmetric, then the pseudomedian and median coincide. See Hollander & Wolfe (1973), page 34.) If exact *p*-values are available, an exact confidence interval is obtained by the algorithm described in Bauer (1972), and the Hodges-Lehmann estimator is employed. Otherwise, the returned confidence interval and point estimate are based on normal approximations.

With small samples it may not be possible to achieve very high confidence interval coverages. If this happens a warning will be given and an interval with lower coverage will be substituted.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value of the test statistic with a name describing it.

`parameter`	the parameter(s) for the exact distribution of the test statistic.
`p.value`	the *p*-value for the test.
`null.value`	the location parameter `mu`.
`alternative`	a character string describing the alternative hypothesis.
`method`	the type of test applied.
`data.name`	a character string giving the names of the data.
`conf.int`	a confidence interval for the location parameter. (Only present if argument `conf.int = TRUE`.)
`estimate`	an estimate of the location parameter. (Only present if argument `conf.int = TRUE`.)

Warning

This function can use large amounts of memory and stack (and even crash **R** if the stack limit is exceeded) if `exact = TRUE` and one sample is large (several thousands or more).

Note

The literature is not unanimous about the definitions of the Wilcoxon rank sum and Mann-Whitney tests. The two most common definitions correspond to the sum of the ranks of the first sample with the minimum value subtracted or not: **R** subtracts and S-PLUS does not, giving a value which is larger by *m(m+1)/2* for a first sample of size *m*. (It seems Wilcoxon's original paper used the unadjusted sum of the ranks but subsequent tables subtracted the minimum.)

R's value can also be computed as the number of all pairs `(x[i], y[j])` for which `y[j]` is not greater than `x[i]`, the most common definition of the Mann-Whitney test.

References

David F. Bauer (1972), Constructing confidence sets using rank statistics. *Journal of the American Statistical Association* **67**, 687–690.

Myles Hollander & Douglas A. Wolfe (1973), *Nonparametric statistical inference*. New York: John Wiley & Sons. Pages 27–33 (one-sample), 68–75 (two-sample).
Or second edition (1999).

See Also

`psignrank`, `pwilcox`.

`wilcox.exact` in **exactRankTests** covers much of the same ground, but also produces exact *p*-values in the presence of ties.

`wilcox_test` in package **coin** for exact and approximate *conditional p*-values for the Wilcoxon tests.

`kruskal.test` for testing homogeneity in location parameters in the case of two or more samples; `t.test` for an alternative under normality assumptions [or large samples]

Examples

```
## One-sample test.
## Hollander & Wolfe (1973), 29f.
## Hamilton depression scale factor measurements in 9 patients with
##  mixed anxiety and depression, taken at the first (x) and second
##  (y) visit after initiation of a therapy (administration of a
##  tranquilizer).
x <- c(1.83,  0.50,  1.62,  2.48, 1.68, 1.88, 1.55, 3.06, 1.30)
y <- c(0.878, 0.647, 0.598, 2.05, 1.06, 1.29, 1.06, 3.14, 1.29)
wilcox.test(x, y, paired = TRUE, alternative = "greater")
wilcox.test(y - x, alternative = "less")    # The same.
wilcox.test(y - x, alternative = "less",
            exact = FALSE, correct = FALSE) # H&W large sample
                                            # approximation

## Two-sample test.
## Hollander & Wolfe (1973), 69f.
## Permeability constants of the human chorioamnion (a placental
##  membrane) at term (x) and between 12 to 26 weeks gestational
##  age (y).  The alternative of interest is greater permeability
##  of the human chorioamnion for the term pregnancy.
x <- c(0.80, 0.83, 1.89, 1.04, 1.45, 1.38, 1.91, 1.64, 0.73, 1.46)
y <- c(1.15, 0.88, 0.90, 0.74, 1.21)
wilcox.test(x, y, alternative = "g")        # greater
wilcox.test(x, y, alternative = "greater",
            exact = FALSE, correct = FALSE) # H&W large sample
                                            # approximation

wilcox.test(rnorm(10), rnorm(10, 2), conf.int = TRUE)

## Formula interface.
boxplot(Ozone ~ Month, data = airquality)
wilcox.test(Ozone ~ Month, data = airquality,
            subset = Month %in% c(5, 8))
```

Pearson's Chi-squared Test for Count Data

Description

`chisq.test` performs chi-squared contingency table tests and goodness-of-fit tests.

Usage

```
chisq.test(x, y = NULL, correct = TRUE,
           p = rep(1/length(x), length(x)), rescale.p = FALSE,
           simulate.p.value = FALSE, B = 2000)
```

Arguments

`x`	a vector or matrix.
`y`	a vector; ignored if `x` is a matrix.
`correct`	a logical indicating whether to apply continuity correction when computing the test statistic for 2x2 tables: one half is subtracted from all $\|O-E\|$ differences. No correction is done if `simulate.p.value = TRUE`.
`p`	a vector of probabilities of the same length of `x`. An error is given if any entry of `p` is negative.
`rescale.p`	a logical scalar; if TRUE then `p` is rescaled (if necessary) to sum to 1. If `rescale.p` is FALSE, and `p` does not sum to 1, an error is given.
`simulate.p.value`	a logical indicating whether to compute p-values by Monte Carlo simulation.
`B`	an integer specifying the number of replicates used in the Monte Carlo test.

Details

If `x` is a matrix with one row or column, or if `x` is a vector and `y` is not given, then a "goodness-of-fit test" is performed ("`x` is treated as a one-dimensional contingency table"). The entries of `x` must be non-negative integers. In this case, the hypothesis tested is whether the population probabilities equal those in `p`, or are all equal if `p` is not given.

If `x` is a matrix with at least two rows and columns, it is taken as a two-dimensional contingency table. Again, the entries of `x` must be non-negative integers. Otherwise, `x` and `y` must be vectors or factors of the same length; incomplete cases are removed, the

objects are coerced into factor objects, and the contingency table is computed from these. Then, Pearson's chi-squared test of the null hypothesis that the joint distribution of the cell counts in a 2-dimensional contingency table is the product of the row and column marginals is performed.

If `simulate.p.value` is `FALSE`, the p-value is computed from the asymptotic chi-squared distribution of the test statistic; continuity correction is only used in the 2-by-2 case (if `correct` is `TRUE`, the default). Otherwise the p-value is computed for a Monte Carlo test (Hope, 1968) with `B` replicates.

In the contingency table case simulation is done by random sampling from the set of all contingency tables with given marginals, and works only if the marginals are strictly positive. (A C translation of the algorithm of Patefield (1981) is used.) Continuity correction is never used, and the statistic is quoted without it. Note that this is not the usual sampling situation for the chi-squared test but rather that for Fisher's exact test.

In the goodness-of-fit case simulation is done by random sampling from the discrete distribution specified by `p`, each sample being of size `n = sum(x)`. This simulation is done in `R` and may be slow.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value the chi-squared test statistic.
`parameter`	the degrees of freedom of the approximate chi-squared distribution of the test statistic, `NA` if the p-value is computed by Monte Carlo simulation.
`p.value`	the p-value for the test.
`method`	a character string indicating the type of test performed, and whether Monte Carlo simulation or continuity correction was used.
`data.name`	a character string giving the name(s) of the data.
`observed`	the observed counts.
`expected`	the expected counts under the null hypothesis.
`residuals`	the Pearson residuals, `(observed - expected) / sqrt(expected)`.

References

Hope, A. C. A. (1968) A simplified Monte Carlo significance test procedure. *J. Roy, Statist. Soc. B* **30**, 582–598.

Patefield, W. M. (1981) Algorithm AS159. An efficient method of generating r x c tables with given row and column totals. *Applied Statistics* **30**, 91–97.

Examples

```
## Not really a good example
chisq.test(InsectSprays$count > 7, InsectSprays$spray)
                                # Prints test summary
chisq.test(InsectSprays$count > 7, InsectSprays$spray)$obs
                                # Counts observed
chisq.test(InsectSprays$count > 7, InsectSprays$spray)$exp
                                # Counts expected under the null

## Effect of simulating p-values
x <- matrix(c(12, 5, 7, 7), nc = 2)
chisq.test(x)$p.value           # 0.4233
chisq.test(x, simulate.p.value = TRUE, B = 10000)$p.value
                                # around 0.29!

## Testing for population probabilities
## Case A. Tabulated data
x <- c(A = 20, B = 15, C = 25)
chisq.test(x)
chisq.test(as.table(x))         # the same
x <- c(89,37,30,28,2)
p <- c(40,20,20,15,5)
try(
chisq.test(x, p = p)            # gives an error
)
chisq.test(x, p = p, rescale.p = TRUE)
                                # works
p <- c(0.40,0.20,0.20,0.19,0.01)
                                # Expected count in category 5
                                # is 1.86 < 5 ==> chi square approx.
chisq.test(x, p = p)            #               maybe doubtful, but is
ok!
chisq.test(x, p = p,simulate.p.value = TRUE)

## Case B. Raw data
x <- trunc(5 * runif(100))
chisq.test(table(x))            # NOT 'chisq.test(x)'!
```

McNemar's Chi-squared Test for Count Data

Description

Performs McNemar's chi-squared test for symmetry of rows and columns in a two-dimensional contingency table.

Usage

```
mcnemar.test(x, y = NULL, correct = TRUE)
```

Arguments

`x`	either a two-dimensional contingency table in matrix form, or a factor object.
`y`	a factor object; ignored if `x` is a matrix.
`correct`	a logical indicating whether to apply continuity correction when computing the test statistic.

Details

The null is that the probabilities of being classified into cells `[i,j]` and `[j,i]` are the same.

If `x` is a matrix, it is taken as a two-dimensional contingency table, and hence its entries should be nonnegative integers. Otherwise, both `x` and `y` must be vectors of the same length. Incomplete cases are removed, the vectors are coerced into factor objects, and the contingency table is computed from these.

Continuity correction is only used in the 2-by-2 case if `correct` is `TRUE`.

Value

A list with class `"htest"` containing the following components:

`statistic`	the value of McNemar's statistic.
`parameter`	the degrees of freedom of the approximate chi-squared distribution of the test statistic.
`p.value`	the p-value of the test.
`method`	a character string indicating the type of test performed, and whether

continuity correction was used.

`data.name` a character string giving the name(s) of the data.

References

Alan Agresti (1990). *Categorical data analysis*. New York: Wiley. Pages 350–354.

Examples

```
## Agresti (1990), p. 350.
## Presidential Approval Ratings.
##  Approval of the President's performance in office in two surveys,
##  one month apart, for a random sample of 1600 voting-age Americans.
Performance <-
matrix(c(794, 86, 150, 570),
       nr = 2,
       dimnames = list("1st Survey" = c("Approve", "Disapprove"),
                       "2nd Survey" = c("Approve", "Disapprove")))
Performance
mcnemar.test(Performance)
## => significant change (in fact, drop) in approval ratings
```

Fisher's Exact Test for Count Data

Description

Performs Fisher's exact test for testing the null of independence of rows and columns in a contingency table with fixed marginals.

Usage

```
fisher.test(x, y = NULL, workspace = 200000, hybrid = FALSE,
            control = list(), or = 1, alternative = "two.sided",
            conf.int = TRUE, conf.level = 0.95,
            simulate.p.value = FALSE, B = 2000)
```

Arguments

`x`	either a two-dimensional contingency table in matrix form, or a factor object.
`y`	a factor object; ignored if `x` is a matrix.
`workspace`	an integer specifying the size of the workspace used in the network algorithm. In units of 4 bytes. Only used for non-simulated p-values larger than *2 by 2* tables.
`hybrid`	a logical. Only used for larger than *2 by 2* tables, in which cases it indicated whether the exact probabilities (default) or a hybrid approximation thereof should be computed. See Details.
`control`	a list with named components for low level algorithm control. At present the only one used is `"mult"`, a positive integer >= *2* with default 30 used only for larger than *2 by 2* tables. This says how many times as much space should be allocated to paths as to keys: see file ‘`fexact.c`’ in the sources of this package.
`or`	the hypothesized odds ratio. Only used in the *2 by 2* case.
`alternative`	indicates the alternative hypothesis and must be one of `"two.sided"`, `"greater"` or `"less"`. You can specify just the initial letter. Only used in the *2 by 2* case.
`conf.int`	logical indicating if a confidence interval should be computed (and returned).
`conf.level`	confidence level for the returned confidence interval. Only used in the *2 by 2* case if `conf.int = TRUE`.
`simulate.p.value`	a logical indicating whether to compute p-values by Monte Carlo

simulation, in larger than *2 by 2* tables.

`B` an integer specifying the number of replicates used in the Monte Carlo test.

Details

If `x` is a matrix, it is taken as a two-dimensional contingency table, and hence its entries should be nonnegative integers. Otherwise, both `x` and `y` must be vectors of the same length. Incomplete cases are removed, the vectors are coerced into factor objects, and the contingency table is computed from these.

For *2 by 2* cases, p-values are obtained directly using the (central or non-central) hypergeometric distribution. Otherwise, computations are based on a C version of the FORTRAN subroutine FEXACT which implements the network developed by Mehta and Patel (1986) and improved by Clarkson, Fan and Joe (1993). The FORTRAN code can be obtained from http://www.netlib.org/toms/643. Note this fails (with an error message) when the entries of the table are too large. (It transposes the table if necessary so it has no more rows than columns. One constraint is that the product of the row marginals be less than *2^31 - 1*.)

For *2 by 2* tables, the null of conditional independence is equivalent to the hypothesis that the odds ratio equals one. 'Exact' inference can be based on observing that in general, given all marginal totals fixed, the first element of the contingency table has a non-central hypergeometric distribution with non-centrality parameter given by the odds ratio (Fisher, 1935). The alternative for a one-sided test is based on the odds ratio, so `alternative = "greater"` is a test of the odds ratio being bigger than `or`.

Two-sided tests are based on the probabilities of the tables, and take as 'more extreme' all tables with probabilities less than or equal to that of the observed table, the p-value being the sum of such probabilities.

For larger than *2 by 2* tables and `hybrid = TRUE`, asymptotic chi-squared probabilities are only used if the "Cochran conditions" are satisfied, that is if no cell has count zero, and more than 80% of the cells have counts at least 5.

Simulation is done conditional on the row and column marginals, and works only if the marginals are strictly positive. (A C translation of the algorithm of Patefield (1981) is used.)

Value

A list with class `"htest"` containing the following components:

`p.value` the p-value of the test.

`conf.int` a confidence interval for the odds ratio. Only present in the *2 by 2* case if

	argument `conf.int = TRUE`.
`estimate`	an estimate of the odds ratio. Note that the *conditional* Maximum Likelihood Estimate (MLE) rather than the unconditional MLE (the sample odds ratio) is used. Only present in the *2 by 2* case.
`null.value`	the odds ratio under the null, `or`. Only present in the *2 by 2* case.
`alternative`	a character string describing the alternative hypothesis.
`method`	the character string `"Fisher's Exact Test for Count Data"`.
`data.name`	a character string giving the names of the data.

References

Agresti, A. (1990) *Categorical data analysis*. New York: Wiley. Pages 59–66.

Fisher, R. A. (1935) The logic of inductive inference. *Journal of the Royal Statistical Society Series A* **98**, 39–54.

Fisher, R. A. (1962) Confidence limits for a cross-product ratio. *Australian Journal of Statistics* **4**, 41.

Fisher, R. A. (1970) *Statistical Methods for Research Workers.* Oliver & Boyd.

Mehta, C. R. and Patel, N. R. (1986) Algorithm 643. FEXACT: A Fortran subroutine for Fisher's exact test on unordered *r*c* contingency tables. *ACM Transactions on Mathematical Software*, **12**, 154–161.

Clarkson, D. B., Fan, Y. and Joe, H. (1993) A Remark on Algorithm 643: FEXACT: An Algorithm for Performing Fisher's Exact Test in *r x c* Contingency Tables. *ACM Transactions on Mathematical Software*, **19**, 484–488.

Patefield, W. M. (1981) Algorithm AS159. An efficient method of generating r x c tables with given row and column totals. *Applied Statistics* **30**, 91–97.

See Also

`chisq.test`

Examples

```
## Agresti (1990), p. 61f, Fisher's Tea Drinker
## A British woman claimed to be able to distinguish whether milk or
##  tea was added to the cup first.  To test, she was given 8 cups of
##  tea, in four of which milk was added first.  The null hypothesis
##  is that there is no association between the true order of pouring
##  and the woman's guess, the alternative that there is a positive
##  association (that the odds ratio is greater than 1).
```

```
TeaTasting <-
matrix(c(3, 1, 1, 3),
       nr = 2,
       dimnames = list(Guess = c("Milk", "Tea"),
                       Truth = c("Milk", "Tea")))
fisher.test(TeaTasting, alternative = "greater")
## => p=0.2429, association could not be established

## Fisher (1962, 1970), Criminal convictions of like-sex twins
Convictions <-
matrix(c(2, 10, 15, 3),
       nr = 2,
       dimnames =
       list(c("Dizygotic", "Monozygotic"),
            c("Convicted", "Not convicted")))
Convictions
fisher.test(Convictions, alternative = "less")
fisher.test(Convictions, conf.int = FALSE)
fisher.test(Convictions, conf.level = 0.95)$conf.int
fisher.test(Convictions, conf.level = 0.99)$conf.int

## A r x c table  Agresti (2002, p. 57) Job Satisfaction
Job <- matrix(c(1,2,1,0, 3,3,6,1, 10,10,14,9, 6,7,12,11), 4, 4,
dimnames = list(income=c("< 15k", "15-25k", "25-40k", "> 40k"),
                satisfaction=c("VeryD", "LittleD", "ModerateS",
"VeryS")))
fisher.test(Job)
fisher.test(Job, simulate=TRUE, B=1e5)
```

ORIGIN ≡ 0

2 X 2 Contingency Tests

Contingency tests consider data from categorical (also called nominal) variables - variables in which observations may be placed in classes, but the classes themselves need not have numerical or ordinal significance. When comparing two categorical variables it is customary to construct a contingency table showing which observations may be simultaneously classified according to the classes. From the contingency table, tests of association (or alternatively tests of independence) may be performed. Here we look at the 2X2 case in which there are only 2 classes for each of two variables.

Assumptions:

- **Observed values $X_1, X_2, X_3, \ldots X_{n1}$ are a random sample**
  Observed values $Y_1, Y_2, Y_3, \ldots Y_{n2}$ .are a random sample.

Contingency Table		
1,1	1,2	X=1
2,1	2,2	X=2
Y=1	Y=2	Grand Total

Model:

Let Probabilities:
- **$P_i = P(X=i)$**
- **$P_j = P(Y=j)$**
- **$P_{ij} = (X=i, Y=j)$**

Hypotheses:

H_0: $P_{ij} = (P_i)(P_j)$ < **That is, variables X & Y are independent!**

H_1: $P_{ij} <> (P_i)(P_j)$ < **Two sided test**

Criterion for Normal Approximation:

- **IF expected values in each cell $E_{ij} \geq 5$ THEN Normal Approximation may be used**
  OTHERWISE use Exact Test e.g., Fisher's Exact Test

Normal Approximation:

Construct Contingency Tables of Observed and Expected in each cell:

- **Tabulate O_{ij} for each cell**
- **Calculate Observed Row and Column Totals**
- **Calculate Expected for each cell**

$$E_{ij} := \frac{R_i \cdot C_j}{GT}$$

Contingency Table				Row Totals
$O_{1,1}$	$E_{1,2}$	$O_{1,1}$	$E_{1,2}$	$R_1 = \Sigma_{X=1}$
$O_{2,1}$	$E_{2,2}$	$O_{2,1}$	$E_{2,2}$	$R_2 = \Sigma_{X=2}$
$C_1 = \Sigma_{Y=1}$		$C_2 = \Sigma_{Y=2}$		Σ_R or Σ_C
Column Totals				Grand Total

Test Statistic (Yates Corrected):

$$X_{sq} := \sum_{(i,j)} \frac{\left(\left|O_{i,j} - E_{i,j}\right| - \frac{1}{2}\right)^2}{E_{i,j}}$$

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$CV := \text{inverse}\chi\text{sq}(1-\alpha)$ $\quad df := 1$ $\quad CV := \text{qchisq}(1-\alpha, df)$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi sq}(X_{sq})\right)$ $P := \left(1 - \text{pchisq}(X_{sq}, df)\right)$

Example:

Breast Cancer Example Rosner Ex 10.13, p. 397

Observed: $i := 0..1$ $j := 0..1$ $O := \begin{pmatrix} 683 & 2537 \\ 1498 & 8747 \end{pmatrix}$

$R_0 := O_{0,0} + O_{0,1}$ $R_0 = 3220$

$R_1 := O_{1,0} + O_{1,1}$ $R_1 = 10245$

$GT := R_0 + R_1$ $GT = 13465$

$C_0 := O_{0,0} + O_{1,0}$ $C_0 = 2181$

$C_1 := O_{0,1} + O_{1,1}$ $C_1 = 11284$

$C_0 + C_1 = 13465$

Observed Contingency Table:

$O = \begin{pmatrix} 683 & 2537 \\ 1498 & 8747 \end{pmatrix}$ $R = \begin{pmatrix} 3220 \\ 10245 \end{pmatrix}$

$C^T = (2181 \quad 11284)$ $GT = 13465$

^ Vector/Matrix Transpose function

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_{n1}$ are a random sample
Observed values $Y_1, Y_2, Y_3, \ldots Y_{n2}$ .are a random sample.

Model:

Let Probabilities:
- P_i = P(X=i)
- P_j = P(X=i)
- P_{ij} = (X=i, Y=j)

Expected:

$E_{0,0} := \frac{R_0 \cdot C_0}{GT}$ $E_{0,1} := \frac{R_0 \cdot C_1}{GT}$

$E_{1,0} := \frac{R_1 \cdot C_0}{GT}$ $E_{1,1} := \frac{R_1 \cdot C_1}{GT}$

Expected Contingency Table:

$E = \begin{pmatrix} 521.5611 & 2698.4389 \\ 1659.4389 & 8585.5611 \end{pmatrix}$

^ confirmed p. 395

Hypotheses:

H_0: $P_{ij} = (P_i)(P_j)$ **< That is, variables X & Y are independent!**

H_1: $P_{ij} <> (P_i)(P_j)$ **< Two sided test**

Criterion for Normal Approximation:

- IF expected values in each cell $E_{ij} \geq 5$ THEN Normal Approximation may be used
OTHERWISE use Exact Test e.g., Fisher's Exact Test

All $E_{i,j} \geq 5$ thus data qualifies for this approximation...

Test Statistic (Yates Corrected):

$$XsqBLOCK_{i,j} := \frac{\left(\left|O_{i,j} - E_{i,j}\right| - \frac{1}{2}\right)^2}{E_{i,j}}$$

Calculation for Each Cell:

$XsqBLOCK = \begin{pmatrix} 49.6612 & 9.5986 \\ 15.6085 & 3.0168 \end{pmatrix}$

Sum:

$X_{sq} := \sum XsqBLOCK^{\langle 0 \rangle} + \sum XsqBLOCK^{\langle 1 \rangle}$ $X_{sq} = 77.8851$ **< confirmed p. 398**

Critical Value of the Test:

$\alpha := 0.01$ **< Probability of Type I error must be explicitly set**

$df := 1$ $CV := qchisq(1 - \alpha, df)$ $CV = 6.6349$ **< confirmed p. 398**

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

$CV = 6.6349$ $X_{sq} = 77.8851$

Probability Value:

$P := (1 - pchisq(X_{sq}, df))$ $P = 0$ **< Rosner's very small value more-or-less confirmed**

Alternate Calculation of Test Statistic using Determinant:

See Rosner p. 399:

$n := GT$

$a := O_{0,0}$ $c := O_{1,0}$

$b := O_{0,1}$ $d := O_{1,1}$

$$Xs := n \cdot \frac{\left(|O| - \frac{GT}{2}\right)^2}{[(a + b) \cdot (c + d) \cdot (a + c) \cdot (b + d)]}$$

$|O| = 2.1738 \times 10^6$ **< determinant of matrix O**

$X_{sq} = 77.8851$ **< from above**

$Xs = 77.8851$ **< here**

R Prototype:

COMMANDS:
> X=matrix(c(683,2537,1498,8747),nrow=2,byrow=T) < note here how to construct
> X a simple contingency table of
> chisq.test(X,correct=T) observations...

^ turns Yates correction 'on'

Pearson's Chi-squared test with Yates' continuity correction

data: X
X-squared = 77.8851, df = 1, p-value < 2.2e-16

^ Test Statistic, df & Probability confirmed!

Same Example worked as a Binomial Test of two populations:

Rosner p 387-388.

Observed Contingency Table:

$$O = \begin{pmatrix} 683 & 2537 \\ 1498 & 8747 \end{pmatrix} \qquad R = \begin{pmatrix} 3220 \\ 10245 \end{pmatrix}$$

$C^T = (2181 \quad 11284)$ GT = 13465

^ Vector/Matrix Transpose function

Model:

Two Binomial populations with:

p_1 = P(X=0, Y=0)

p_2 = P(X=1, Y=0)

Hypotheses:

H_0: $p_1 = p_2$ **< parameter p is the same in the two populaitons**

H_1: $p_1 <> p_2$ **< Two sided test**

Point Estimate of p for each population:

$p1_{hat} := \frac{O_{0,0}}{R_0}$ $p1_{hat} = 0.2121$ $p2_{hat} := \frac{O_{1,0}}{R_1}$ $p2_{hat} = 0.1462$

Pooled estimate of p & q:

$p_{hat} := \frac{\sum O^{\langle 0 \rangle}}{\sum R}$ $p_{hat} = 0.162$

$\sum O^{\langle 0 \rangle} = 2181$ **< sum of 1st column of O**

$\sum R = 13465$ **< sum of R**

$q_{hat} := 1 - p_{hat}$ $q_{hat} = 0.838$

Normal Theory Approximation:

Test is valid if $n_1 \cdot p1_{hat} \cdot q1_{hat} \geq 5$ **and** $n_2 \cdot p2_{hat} \cdot q2_{hat} \geq 5$

$n := R$

$n_0 \cdot p_{hat} \cdot q_{hat} = 437.081$ $n_1 \cdot p_{hat} \cdot q_{hat} = 1390.6505$ **< Normal approximation OK**

Test Statistic Z:

$$Z := \frac{\left|p1_{hat} - p2_{hat}\right| - \left(\frac{1}{2 \cdot n_0} + \frac{1}{2 \cdot n_1}\right)}{\sqrt{p_{hat} \cdot q_{hat} \cdot \left(\frac{1}{n_0} + \frac{1}{n_1}\right)}} \qquad Z = 8.8253$$

Critical Values of the Test:

$\alpha := 0.05$ **< probability of Type I error must be explicitly set**

$CV := qnorm\left(1 - \frac{\alpha}{2}, 0, 1\right)$ $CV = 1.96$

Decision Rule:

IF |Z| > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

CV = 1.96 Z = 8.8253

Probability Value for z:

$P := 2 \cdot (1 - pnorm(Z, 0, 1))$ $P = 0$

ORIGIN ≡ 0

McNemar's Test for Paired Data

This test employs a 2X2 contingency table in which pairs of observations such as in treatments or "before" versus "after" observations are exactly paired for individuals within a study. Analogy with the paired t-test situation is evident here, although here each variable involves categorical (nominal) data classes.

Assumptions:

- **Paired exactly matched observations are made.**
- **X & Y refer to paired dependent observations**

Paired Contingency Table

Concordant	**Discordant Type A**	X=1
Discordant Type B	**Concordant**	X=2
Y=1	Y=2	Grand Total

Model:

Interpret diagonal cells of paired observations as:

- **concordant - in agreement in result between X & Y**
- **discordant - not in agreement in result in two types:**
  - **Type A (+,-) and Type B (-,+) definition arbitrary**
- **Let p be the probability of the Type A discordant result**

Hypotheses:

H_0: p = 1/2 < **Discordant Type A and Type B results are equally probable**
There is no difference between treatments or between "before" and "after"

H_1: p <> 1/2 < **Two sided test**

Criterion for Normal Approximation:

- **IF number of discordant pairs** $n_D \geq 20$ **THEN Approximation may be used**

OTHERWISE use Exact Test

Construct Contingency Tables of Concordant and Discordant cells:

- **Tabulate paired O_{ij} for each cell**
- **Calculate n_D = total number of discordant pairs**
- **Calculate n_A = number of Type A discordant pairs**

Normal Approximation:

Test Statistic (Corrected):

$$X_{sq} := \frac{\left(\left|n_A - \frac{n_D}{2}\right| - \frac{1}{2}\right)^2}{\left(\frac{n_D}{4}\right)}$$

also calculated by:

$$X_{sq} := \frac{\left(\left|n_A - n_B\right| - 1\right)^2}{n_A + n_B}$$

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$CV := \text{inverse}\chi\text{sq}(1 - \alpha)$ $\quad df := 1$ $\quad CV := \text{qchisq}(1 - \alpha, df)$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi sq}(X_{sq})\right)$ $\quad P := \left(1 - \text{pchisq}(X_{sq}, df)\right)$

Example:

Cancer Example Rosner Ex 10.24, p. 411-412

Assumptions:

- Paired exactly matched observations are made.
- X & Y refer to paired dependent observations

Model:

Interpret diagonal cells of paired observations as:

- concordant - in agreement in result between X & Y
- discordant - not in agreement in result in two types:
- Type A (+,-) and Type B (-,+) definition arbitrary
- Let p be the probability of the Type A discordant result

Observed: $i := 0..1 \quad j := 0..1 \qquad O := \begin{pmatrix} 510 & 16 \\ 5 & 90 \end{pmatrix}$

$n_D := O_{1,0} + O_{0,1} \qquad n_D = 21$

$n_A := O_{1,0} \qquad n_A = 5$

$n_B := O_{0,1} \qquad n_B = 16 \qquad n_D - n_A = 16$

Hypotheses:

H_0: p = 1/2 **< Discordant Type A and Type B results are equally probable**
There is no difference between treatments or between "before" and "after"

H_1: p <> 1/2 **< Two sided test**

Criterion for Normal Approximation:

- IF number of discordant pairs $n_D \geq 20$ **THEN Approximation may be used**
OTHERWISE use Exact Test

$n_D = 21$ **< Normal Approximation is appropriate**

Normal Approximation:

Test Statistic (Corrected):

also calculated by:

$$X_{sq} := \frac{\left(\left|n_A - \frac{n_D}{2}\right| - \frac{1}{2}\right)^2}{\left(\frac{n_D}{4}\right)} \qquad X_{sq} = 4.7619 \qquad Xsq := \frac{\left(\left|n_A - n_B\right| - 1\right)^2}{n_A + n_B} \qquad Xsq = 4.7619$$

^ confirmed p. 412

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$df := 1 \qquad CV := qchisq(1 - \alpha, df) \qquad CV = 3.8415$ **< confirmed p. 412**

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

$X_{sq} = 4.7619 \qquad CV = 3.8415$

Probability Value:

$P := (1 - pchisq(X_{sq}, df)) \qquad P = 0.0291$ **< confirmed p. 412**

R Prototype:

COMMANDS:
> X=matrix(c(510,16,5,90),nrow=2,byrow=T)
> X
> mcnemar.test(X,correct=T)

McNemar's Chi-squared test with continuity correction

data: X
McNemar's chi-squared = 4.7619, df = 1, p-value = 0.02910

^ X_{sq}, df and P values confirmed

Exact Test:

Probability Values:

IF $n_A < n_D/2$:

$$P := 2 \cdot \sum_{k=0}^{n_A} \text{combin}(n_D, k) \cdot \left(\frac{1}{2}\right)^{n_D}$$

IF $n_A > n_D/2$:

$$P := 2 \cdot \sum_{k=n_A}^{n_D} \text{combin}(n_D, k) \cdot \left(\frac{1}{2}\right)^{n_D}$$

IF $n_A = n_D/2$:

$$P := 1$$

Example:

Rosner Ex. 10.25 p. 413-414

Observed: $i := 0..1 \quad j := 0..1 \qquad O := \begin{pmatrix} 3 & 7 \\ 1 & 9 \end{pmatrix}$

$n_D := O_{1,0} + O_{0,1} \qquad n_D = 8$

$n_A := O_{1,0} \qquad n_A = 1$

$n_B := O_{0,1} \qquad n_B = 7 \qquad n_D - n_A = 7$

Criterion for Normal Approximation:

- IF number of discordant pairs $n_D \geq 20$ THEN Approximation may be used
OTHERWISE use Exact Test

$n_D = 8$ **< Normal Approximation is NOT appropriate - Exact Method must be used**

Exact Test Probability: **$n_A < n_D/2$:**

$$P := 2 \cdot \sum_{k=0}^{n_A} \text{combin}(n_D, k) \cdot \left(\frac{1}{2}\right)^{n_D} \qquad P = 0.0703$$ **< confirmed p. 414**

R Prototype:

COMMANDS:

```
> X=matrix(c(3,7,1,9),nrow=2,byrow=T)
> X
> mcnemar.test(X)
```

```
        McNemar's Chi-squared test with continuity correction

data:  X
McNemar's chi-squared = 3.125, df = 1, p-value = 0.0771
```

^ Apparently not done the Exact way, but P result is close.

ORIGIN ≡ 0

χ^2 Test for Association in RXC Contingency Tables

This test employs a RXC contingency table consisting of R rows and C Columns and is thus an extension of the 2X2 case discussed previously.

Assumptions:

- **Observed values $X_{i,j}$ are a random sample**

- **Observed values for Rows and Columns are independent.**

RXC Contingency Table

$O_{1,1}$	$O_{1,2}$	$O_{1,3}$	…	$O_{1,j}$	
$O_{2,1}$	$O_{2,2}$	$O_{2,3}$	…	$O_{2,j}$	
$O_{3,1}$	$O_{3,2}$	$O_{3,3}$	…	$O_{3,j}$	**Row Totals**
…	…	…	…	…	
$O_{i,1}$	$O_{i,2}$	$O_{i,3}$	…	$O_{i,j}$	
	Column Totals				**Grand Total**

Model:

Let Probabilities:

- **P_i = P(R=i)**
- **P_j = P(C=j)**
- **P_{ij} = P(R=i, C=j)**

Hypotheses:

H_0: $P_{ij} = (P_i)(P_j)$ < **That is, variables R & C are independent!**

H_1: $P_{ij} <> (P_i)(P_j)$ < **Two sided test**

Criterion for Approximation:

- **IF no more than 1/5 of the cells have expected values in each block** $E_{ij} \le 5$

AND no cell has expected value $E_{ij} < 1$ **THEN Approximation may be used**

Construct Contingency Tables of Observed and Expected in each cell:

- **Tabulate O_{ij} for each cell**
- **Calculate Observed Row and Column Totals**
- **Calculate Expected for each cell**

$$E_{ij} := \frac{R_i \cdot C_j}{GT}$$

χ^2 Test Statistic:

$$X_{sq} := \sum_i \sum_j \frac{\left(O_{i,j} - E_{i,j}\right)^2}{E_{i,j}}$$

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$df := (R - 1) \cdot (C - 1)$ < **where R & C are the number of Row and Column cells respectively**

$CV := inverse\chi sq(1 - \alpha)$ $\qquad$ $CV := qchisq(1 - \alpha, df)$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi sq}(X_{sq})\right)$ $\qquad$ $P := \left(1 - pchisq(X_{sq}, df)\right)$

Example:

Cancer Rosner Example 10.35 p. 430

Assumptions:

- Observed values $X_{i,j}$ are a random sample

- Observed values for Rows and Columns are independent.

Model:

Let Probabilities:

- P_i = P(R=i)

- P_j = P(C=j)

- P_{ij} = P(R=i, C=j)

Hypotheses:

H_0: $P_{ij} = (P_i)(P_j)$

H_1: $P_{ij} <> (P_i)(P_j)$

Observed:

$R := 2 \quad C := 5$

$i := 0 .. R - 1 \quad j := 0 .. C - 1$

$$O := \begin{pmatrix} 320 & 1206 & 1011 & 463 & 220 \\ 1422 & 4432 & 2893 & 1092 & 406 \end{pmatrix}$$

Calculating Observed sums:

$RS_i := \sum \left(O^T\right)^{\langle i \rangle}$ **< sums for each row**

$CS_j := \sum O^{\langle j \rangle}$ **< sums for each column**

$GT := \sum RS^{\langle 0 \rangle} \quad GT = 13465 \quad \sum CS^{\langle 0 \rangle} = 13465$ **< Grand Total**

Observed Table with Sums:

$$O = \begin{pmatrix} 320 & 1206 & 1011 & 463 & 220 \\ 1422 & 4432 & 2893 & 1092 & 406 \end{pmatrix} \quad RS = \begin{pmatrix} 3220 \\ 10245 \end{pmatrix}$$

$$CS^T = (1742 \quad 5638 \quad 3904 \quad 1555 \quad 626) \quad GT = 13465$$

Construct Contingency Tables of Observed and Expected in each cell:

Calculating Expected table:

$$E_{i,j} := \frac{RS_i \cdot CS_j}{GT}$$

Calculating Expected sums as a check:

$ERS_i := \sum \left(E^T\right)^{\langle i \rangle}$ **< sums for each row**

$ECS_j := \sum E^{\langle j \rangle}$ **< sums for each column**

$EGT := \sum RS^{\langle 0 \rangle} \quad EGT = 13465 \quad \sum ECS^{\langle 0 \rangle} = 13465$ **< Grand Total**

Expected Table with Sums:

$$E = \begin{pmatrix} 416.5793 & 1348.2629 & 933.5967 & 371.8604 & 149.7007 \\ 1325.4207 & 4289.7371 & 2970.4033 & 1183.1396 & 476.2993 \end{pmatrix} \quad ERS = \begin{pmatrix} 3220 \\ 10245 \end{pmatrix}$$

$$ECS^T = (1742 \quad 5638 \quad 3904 \quad 1555 \quad 626) \qquad EGT = 13465$$

χ^2 Test Statistic:

$$X_{sq} := \sum_i \sum_j \frac{\left(O_{i,j} - E_{i,j}\right)^2}{E_{i,j}} \qquad X_{sq} = 130.338$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$df := (R - 1) \cdot (C - 1)$ $df = 4$ **< where R & C are the number of Row and Column cells respectively**

$CV := qchisq(1 - \alpha, df)$ $CV = 9.4877$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

$X_{sq} = 130.338$ $CV = 9.4877$

Probability Value:

$P := (1 - pchisq(X_{sq}, df))$ $P = 0$

Prototype in R:

COMMANDS:

```
> X=matrix(c(320, 1206, 1011, 463, 220, 1422, 4432, 2893, 1092,406),nrow=2,byrow=T)
> X
> chisq.test(X)
```

Pearson's Chi-squared test

data: X
X-squared = 130.338, df = 4, p-value < 2.2e-16

^ Xsq, df & P confirmed

Prototype in Systat:

To run, set the Count Column as a "frequency" under DATA
Run Cross tabs setting Row as ROW and Column as COL

Data format:

Count Row Column

$$\begin{pmatrix} 320 & 1 & 1 \\ 1206 & 1 & 2 \\ 1011 & 1 & 3 \\ 463 & 1 & 4 \\ 220 & 1 & 5 \\ 1422 & 2 & 1 \\ 4432 & 2 & 2 \\ 2893 & 2 & 3 \\ 1092 & 2 & 4 \\ 406 & 2 & 5 \end{pmatrix}$$

Case frequencies determined by value of variable COUNT.

Frequencies
ROW (rows) by COL (columns)

	1	2	3	4	5	Total
1	320	1206	1011	463	220	3220
2	1422	4432	2893	1092	406	10245
Total	1742	5638	3904	1555	626	13465

Test statistic	Value	df	Prob
Pearson Chi-square	130.33802	4.00000	0.00000

ORIGIN ≡ 0

χ^2 Test for Goodness of Fit

The RXC Contingency Table approach can be applied to many hypotheses in addition to independence of variables $P_{i,j} = (P_i)(P_j)$.

Assumptions:

- Observed values O_j are a random sample in g cells

Goodness of Fit Table

O_1	O_2	O_3	…	O_j	Total
E_1	E_2	E_3	…	E_j	

Model:

Let Expected Probabilities:
- P_j is specified:
- internally specified model with k parameters estimated from the sample.
OR
- externally specified model k=0

Hypotheses:

H_0: P_j are distributed according to the model
H_1: P_j differ from the model **< Two sided test**

Criterion for Approximation:

- IF no more than 1/5 of the cells have expected values in each cell $E_j \le 5$
AND no cell has expected value $E_j < 1$ **THEN Approximation may be used**

Construct Contingency Tables of Observed and Expected in each cell:

- Tabulate O_j for each cell
- Calculate Observed Row and Column Totals
- Calculate Expected for each cell:

$E_{ij} := GT \cdot pE_j$ **< where: pE_j are the expected probabilities of each cell**

χ^2 Test Statistic:

$$X_{sq} := \sum_j \frac{\left(O_j - E_j\right)^2}{E_j}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$df := g - k - 1$ **< where: g = the number of cells,**
k = number of parameters of the *internally* specified model

$CV := inverse\chi sq(1 - \alpha)$ $CV := qchisq(1 - \alpha, df)$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi sq}(X_{sq})\right)$ $P := \left(1 - pchisq(X_{sq}, df)\right)$

Example:

Testing for Normal Distribution Rosner Example 10.41 p. 441

Assumptions:

- Observed values O_j are
a random sample in g cells

$$O := \begin{pmatrix} 57 \\ 330 \\ 2132 \\ 4584 \\ 4604 \\ 2119 \\ 659 \\ 251 \end{pmatrix} \qquad GT := \sum O$$

$GT = 14736$

Model:

Let Expected Probabilities:
- P_j is specified:
- internally specified model with k parameters estimated from the sample. **< internally specified with parameters X_{bar} & s^2**
OR
- externally specified model k=0

$k := 2$

$g := 8$

Hypotheses:

H_0: P_j are distributed according to the model
H_1: P_j differ from the model **< Two sided test**

Constructing Expected Table:

$X_{bar} := 80.68$ **< mean & standard deviation given for sample whose frequencies are tabulated in X. Here we can not calculate them directly.**

$s := 12.00$

$j := 0 .. g - 1$

boundary values in X >

$$B := \begin{pmatrix} 0 & 50 \\ 50 & 60 \\ 60 & 70 \\ 70 & 80 \\ 80 & 90 \\ 90 & 100 \\ 100 & 110 \\ 110 & 999 \end{pmatrix}$$

$E_j := GT \cdot \left(pnorm\left(B_{j,1}, X_{bar}, s\right) - pnorm\left(B_{j,0}, X_{bar}, s\right) \right)$ **< calculating expected based on normal distribution: GT * prob of each cell**

observed	expected	totals
$O = \begin{pmatrix} 57 \\ 330 \\ 2132 \\ 4584 \\ 4604 \\ 2119 \\ 659 \\ 251 \end{pmatrix}$	$E = \begin{pmatrix} 77.8653 \\ 547.1493 \\ 2126.682 \\ 4283.3488 \\ 4478.5195 \\ 2431.1276 \\ 684.0861 \\ 107.2213 \end{pmatrix}$	$\sum O = 14736$ $\sum E = 14736$ **< E verified p.438**

χ^2 Test Statistic:

$$X_{sq} := \sum_j \frac{\left(O_j - E_j\right)^2}{E_j} \qquad X_{sq} = 350.198$$

^ verified p. 441

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$df := g - k - 1$ $df = 5$

$CV := qchisq\left(1 - \alpha, df\right)$ $CV = 11.0705$

< where: g = the number of cells, k = number of parameters of *internally* specified model

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0
$X_{sq} = 350.198$ $CV = 11.0705$

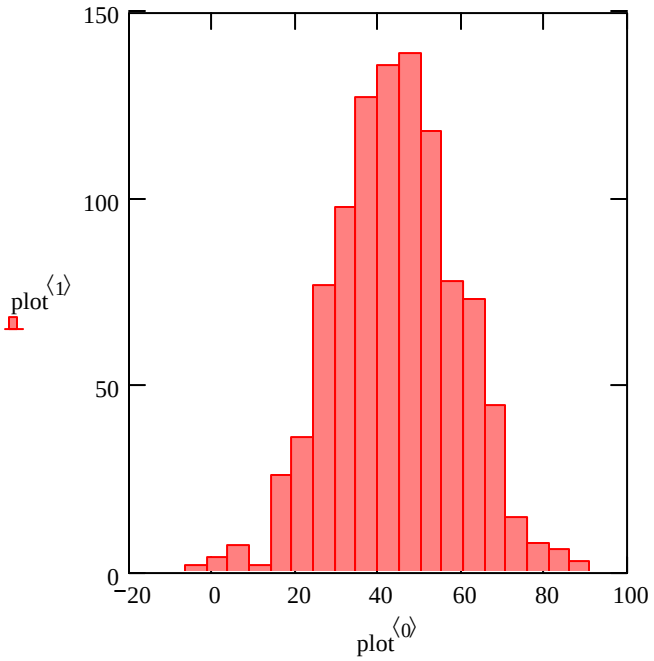
Probability Value:

$P := \left(1 - pchisq\left(X_{sq}, df\right)\right)$ $P = 0$ **< verified p. 441**

Example from Scratch:

Let's test the ability of MathCad's random number generator to make a Normal Distribution:

$\mu := 45 \quad \sigma := 15 \quad X := rnorm(1000, \mu, \sigma) \quad plot := histogram(19, X)$



$X =$

	0
0	38.4154
1	34.8089
2	37.9007
3	30.728
4	19.7147
5	45.653
6	43.1905
7	53.3464
8	77.8768
9	57.131
10	59.7771
11	57.9334
12	58.7335
13	55.095
14	29.3353
15	46.0362

$plot =$

−4.4211	2
0.7368	4
5.8947	7
11.0526	2
16.2105	26
21.3684	36
26.5263	77
31.6842	98
36.8421	127
42	136
47.1579	139
52.3158	118
57.4737	78
62.6316	73
67.7895	45
72.9474	15
78.1053	8
83.2632	6
88.4211	3

Here I had the histogram function make 19 cells with boundaries in X shown in the first column of variable plot. The second column in plot are the counts of Observed values in each cell.

$g := length(plot^{\langle 1 \rangle}) \quad g = 19 \quad j := 0 .. g - 2 \quad GT := \sum plot^{\langle 1 \rangle} \quad GT = 1000$

$B^{\langle 1 \rangle} := plot^{\langle 0 \rangle}$

$B_{j+1,0} := (plot^{\langle 0 \rangle})_j$ **< constructing boundary matrix B**

$B_{0,0} := -99999 \quad B_{g-1,1} := 99999$ **< tails**

$X_{bar} := mean(X)$

$s := \sqrt{Var(X)}$ **< internally estimated parameters**

$g := length(B^{\langle 0 \rangle}) \quad j := 0 .. g - 1$

$E_j := GT \cdot (pnorm(B_{j,1}, X_{bar}, s) - pnorm(B_{j,0}, X_{bar}, s))$

$E =$

0.5109
1.155
3.1858
7.788
16.8742
32.4045
55.1543
83.2052
111.255
131.8527
138.5033
128.9534
106.416
77.8361
50.4607
28.9948
14.7666
6.6654
4.0179

$B =$

−99999	−4.4211
−4.4211	0.7368
0.7368	5.8947
5.8947	11.0526
11.0526	16.2105
16.2105	21.3684
21.3684	26.5263
26.5263	31.6842
31.6842	36.8421
36.8421	42
42	47.1579
47.1579	52.3158
52.3158	57.4737
57.4737	62.6316
62.6316	67.7895
67.7895	72.9474
72.9474	78.1053
78.1053	83.2632
83.2632	99999

$\sum E = 1000$

< Expected matrix E >

$length(E) = 19$

Notice that I chose 19 cells in order to have no cell with Expected value less than 1...

Even so, the first cell violates that assumption...

So, lumping the first two cells (B_L):

$g := \text{length}\left(\text{plot}^{\langle 1 \rangle}\right)$ $j := 0 .. g - 2$ $m := 0 .. 1$

$BL_{j,m} := B_{j+1,m}$

$BL_{0,0} := -99999$

And recalculating Expected (E_L):

$g := \text{length}\left(BL^{\langle 0 \rangle}\right)$ $j := 0 .. g - 1$

$EL_j := GT \cdot \left(\text{pnorm}\left(BL_{j,1}, X_{bar}, s\right) - \text{pnorm}\left(BL_{j,0}, X_{bar}, s\right)\right)$

$EL =$

1.6659
3.1858
7.788
16.8742
32.4045
55.1543
83.2052
111.255
131.8527
138.5033
128.9534
106.416
77.8361
50.4607
28.9948
14.7666
6.6654
4.0179

$BL =$

–99999	0.7368
0.7368	5.8947
5.8947	11.0526
11.0526	16.2105
16.2105	21.3684
21.3684	26.5263
26.5263	31.6842
31.6842	36.8421
36.8421	42
42	47.1579
47.1579	52.3158
52.3158	57.4737
57.4737	62.6316
62.6316	67.7895
67.7895	72.9474
72.9474	78.1053
78.1053	83.2632
83.2632	99999

$\sum EL = 1000$

$\text{length}(EL) = 18$

Criterion for Approximation:

IF no more than 1/5 of the cells have expected values in each cell $E_j \le 5$

AND no cell has expected value $E_j < 1$

THEN Approximation may be used.

$\text{length}(EL) \cdot \frac{1}{5} = 3.6$

3 cells with expected less than 5 so qualifies for test...

Resizing the Observed:

$g = 18$ $j := 0 .. g - 1$

lumping the first two cells:

$O_j := \left(\text{plot}^{\langle 1 \rangle}\right)_{j+1}$

$O_0 := \left(\text{plot}^{\langle 1 \rangle}\right)_0 + \left(\text{plot}^{\langle 1 \rangle}\right)_1$

$\text{length}(O) = 18$

$O =$

6
7
2
26
36
77
98
127
136
139
118
78
73
45
15
8
6
3

χ^2 Test Statistic:

$g = 18$ $j := 0 .. g - 1$

$X_{sq} := \sum_j \frac{\left(O_j - EL_j\right)^2}{EL_j}$ $X_{sq} = 58.7119$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$g = 18$ $k = 2$ **< where: g = the number of cells, k = number of parameters of the *internally* specified model**

$df := g - k - 1$ $df = 15$

$CV := \text{qchisq}(1 - \alpha, df)$ $CV = 24.9958$

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

$X_{sq} = 58.7119$ $CV = 24.9958$

Probability Value:

$P := \left(1 - \text{pchisq}\left(X_{sq}, df\right)\right)$ $P = 4.1938 \times 10^{-7}$

Prototype in R:

COMMANDS:

```
> RN=read.table("c:/2007BiostatsData/
Random Normal Example.txt")
> RN
> attach RN
> X=observed
> P=probabilities
> chisq.test(X,p=P)
```

Chi-squared test for given probabilities

data: X
X-squared = 58.7106, df = 17, p-value = 1.713e-06

Warning message:
Chi-squared approximation may be incorrect in: chisq.test(X, p = P)

^ value of X_{sq} verified

Data transferred from MathCAD Random Normal Example.txt

	observed	probabilities
1	6	0.001666
2	7	0.003186
3	2	0.007788
4	26	0.016874
5	36	0.032405
6	77	0.055154
7	98	0.083205
8	127	0.111255
9	136	0.131853
10	139	0.138503
11	118	0.128953
12	78	0.106416
13	73	0.077836
14	45	0.050461
15	15	0.028995
16	8	0.014767
17	6	0.006665
18	3	0.004018

$1 - \text{pchisq}(X_{sq}, g - 1) = 1.7118 \times 10^{-6}$ **< Note that R considers the vector of probabilities P to be an externally specified model. Under these circumstances k=0 and df=17, and P is confirmed. BUT We can be more specific here...**

Using Externally Specified Model:

$\mu = 45$ $\sigma = 15$ **< externally specified parameters we gave the random number generator**

$E_{LE_j} := GT \cdot \left(\text{pnorm}\left(B_{L_{j,1}}, \mu, \sigma\right) - \text{pnorm}\left(B_{L_{j,0}}, \mu, \sigma\right)\right)$ **< new calculation of Expected**

χ^2 Test Statistic:

$g = 18$ $j := 0 .. g - 1$

$\sum E_{LE} = 1000$

$\text{length}(E_{LE}) = 18$

$X_{sqE} := \sum_j \frac{\left(O_j - E_{LE_j}\right)^2}{E_{LE_j}}$ $X_{sqE} = 82.3532$ **< new calculation of X_{sq}**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$g = 18$ $k := 0$ **< where: g = the number of cells, k = number of parameters of the *internally* specified model**

$df := g - k - 1$ $df = 17$

$CVE := \text{qchisq}(1 - \alpha, df)$ $CVE = 27.5871$

^new critical value

$E_{LE} =$

1.5844
2.9824
7.2462
15.6603
30.1043
51.4756
78.2927
105.9231
127.4713
136.4541
129.9313
110.051
82.9137
55.5659
33.1237
17.5636
8.2838
5.3726

Decision Rule:

IF X_{sq} > CV THEN REJECT H_0 OTHERWISE ACCEPT H_0

$X_{sq} = 58.7119$ $CVE = 27.5871$

Probability Value:

$P_E := \left(1 - \text{pchisq}(X_{sq}, df)\right)$ $P_E = 1.7118 \times 10^{-6}$ **< new probability**

ORIGIN ≡ 0

Fisher's Exact Test

Fisher's Exact Test may be used for 2 X 2 contingency tables that fail the criterion for use of the Normal Approximation.

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_{n1}$ are a random sample
Observed values $Y_1, Y_2, Y_3, \ldots Y_{n2}$ .are a random sample.

Contingency Table		
a	b	a+b
c	d	c+d
a+c	b+d	n

Model:

Let Probabilities:
- $P_i = P(X=i)$
- $P_j = P(Y=j)$
- $P_{ij} = (X=i, Y=j)$

Criterion for Normal Approximation:

- IF expected values in each cell $E_{ij} \geq 5$ THEN Approximation may be used
OTHERWISE use Exact Test e.g., Fisher's Exact Test

Fisher's Exact Test:

Enumerate all Possible Contingency Tables:

- Enumerate all possible 2X2 Contingency tables with identical row and column totals as the observed table.
- Calculate the exact probability of each table based on the Hypergeometric Distribution.

Hypergeometric Probability of a 2X2 contingency table:

$$P_T := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!}$$

< a,b,c,d,n are specified as in the table above.

Hypotheses:

H_0: p1 = $(P_i)(P_j)$ **< That is, variables X & Y are independent!**

H_1: P_{ij} <> $(P_i)(P_j)$ **< Two sided test**

Probability Value:

$$P := 2 \cdot \min\left[\left(P_{T_0}, P_{T_1}, .. , P_{T_a}\right), \left(P_{T_a}, P_{T_{a+1}}, .. , PT_k\right), 0.5\right]$$

< for all tables 0 to observed table a, and a + (a+1) up to max table k

^ this probability is interpreted as the probability of obtaining a table as extreme as the one observed.

One-sided probability Values:

Note that "one-sided" tests are possible here where P=min(P_T's) directly, but these must be formulated in terms of the binomial parameter p_1 & p_2 - see Biostatistics 34 for this.

$$P := \left(P_{T_0}, P_{T_1}, .. , P_{T_a}\right)$$

< for alternative hypothesis H_1: $p_1 < p_2$

$$P := \left(P_{T_a}, P_{T_{a+1}}, .. , PT_k\right)$$

< for alternative hypothesis H_1: $p_1 > p_2$

Point Estimates of Binomial Proportions:

$$p1hat := \frac{a}{a+b} \qquad p2hat := \frac{c}{c+d}$$

Critical Value of the Test & Decision Rule:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

IF P < α THEN REJECT H_0 OTHERWISE ACCEPT H_0

Example:

Rosner Example 10.20 p. 406

Contingency Table		
a	b	a+b
c	d	c+d
a+c	b+d	n

$a := 2 \quad b := 23 \qquad a! = 2 \qquad b! = 2.5852 \times 10^{22}$

$c := 5 \quad d := 30 \qquad c! = 120 \qquad d! = 2.6525 \times 10^{32}$

$n := a + b + c + d \qquad n = 60 \qquad n! = 8.321 \times 10^{81}$

Assumptions:

- Observed values $X_1, X_2, X_3, \ldots X_{n1}$ are a random sample
Observed values $Y_1, Y_2, Y_3, \ldots Y_{n2}$ .are a random sample.

$R := \begin{pmatrix} a+b \\ c+d \end{pmatrix} \qquad C := \begin{pmatrix} a+c \\ b+d \end{pmatrix}$

Model:

^ Row and Column totals

Let Probabilities:
- $P_i = P(X=i)$
- $P_j = P(Y=j)$
- $P_{ij} = (X=i, Y=j)$

Criterion for Normal Approximation:

- IF expected values in each cell $E_{ij} \geq 5$ **THEN Normal Approximation may be used**
OTHERWISE use Exact Test e.g., Fisher's Exact Test

$a = 2 \quad b = 23$
$c = 5 \quad d = 30$ **< Fisher's Exact Test must be used...**

Fisher's Exact Test:

Enumeration of all Possible Contingency Tables:

Observed Table:

$a = 2 \quad b = 23$
$c = 5 \quad d = 30$

$T_a := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad T_a = \begin{pmatrix} 2 & 23 \\ 5 & 30 \end{pmatrix} \quad R := \begin{pmatrix} a+b \\ c+d \end{pmatrix} \quad R = \begin{pmatrix} 25 \\ 35 \end{pmatrix} \quad C := \begin{pmatrix} a+c \\ b+d \end{pmatrix}$

Table 0:

$a := 0 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$C^T = (7 \quad 53)$

$T_0 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad T_0 = \begin{pmatrix} 0 & 25 \\ 7 & 28 \end{pmatrix}$

Hypergeometric Probability:

$P_{T_0} := \dfrac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_0} = 0.0174$

Table 1:

$a := 1 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_1 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad T_1 = \begin{pmatrix} 1 & 24 \\ 6 & 29 \end{pmatrix}$

Hypergeometric Probability:

$P_{T_1} := \dfrac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_1} = 0.1051$

Table 2 (The Observed Table):

$a := 2 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_2 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \quad T_2 = \begin{pmatrix} 2 & 23 \\ 5 & 30 \end{pmatrix}$

Hypergeometric Probability:

$P_{T_2} := \dfrac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_2} = 0.2522$

Table 3:

$a := 3 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_3 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad T_3 = \begin{pmatrix} 3 & 22 \\ 4 & 31 \end{pmatrix}$

Hypergeometric Probability:

$$P_{T_3} := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_3} = 0.3118$$

Table 4:

$a := 4 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_4 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad T_4 = \begin{pmatrix} 4 & 21 \\ 3 & 32 \end{pmatrix}$

Hypergeometric Probability:

$$P_{T_4} := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_4} = 0.2144$$

Table 5:

$a := 5 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_5 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad T_5 = \begin{pmatrix} 5 & 20 \\ 2 & 33 \end{pmatrix}$

Hypergeometric Probability:

$$P_{T_5} := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_5} = 0.0819$$

Table 6:

$a := 6 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_6 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad T_6 = \begin{pmatrix} 6 & 19 \\ 1 & 34 \end{pmatrix}$

Hypergeometric Probability:

$$P_{T_6} := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_6} = 0.016$$

Table 7:

$a := 7 \qquad b := R_0 - a$

$c := C_0 - a \qquad d := R_1 - c$

$T_7 := \begin{pmatrix} a & b \\ c & d \end{pmatrix} \qquad T_7 = \begin{pmatrix} 7 & 18 \\ 0 & 35 \end{pmatrix}$

Hypergeometric Probability:

$$P_{T_7} := \frac{(a+b)! \cdot (c+d)! \cdot (a+c)! \cdot (b+d)!}{n! \cdot a! \cdot b! \cdot c! \cdot d!} \qquad P_{T_7} = 0.0012$$

Hypotheses:

H_0: p1 = $(P_i)(P_j)$ **< That is, variables X & Y are independent!**

H_1: P_{ij} <> $(P_i)(P_j)$ **< Two sided test**

Probability Value:

$A := P_{T_0} + P_{T_1} + P_{T_2} \qquad A = 0.3747$ **< confirmed p. 407**

$B := P_{T_2} + P_{T_3} + P_{T_4} + P_{T_5} + P_{T_6} + P_{T_7} \qquad B = 0.8775$ **< confirmed p. 407**

$$P_T = \begin{pmatrix} 0.0174 \\ 0.1051 \\ 0.2522 \\ 0.3118 \\ 0.2144 \\ 0.0819 \\ 0.016 \\ 0.0012 \end{pmatrix}$$

Two sided Probability:

$P := 2 \cdot \min[(A), (B), 0.5] \qquad P = 0.7493$ **< confirmed p. 407**

Critical Value of the Test & Decision Rule:

values confirmed p. 406 ^

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

IF P < α THEN REJECT H_0 OTHERWISE ACCEPT H_0

$P = 0.7493 \qquad \alpha = 0.05$

Prototype in R:

COMMANDS:
> X=matrix(c(2,23,5,30),nrow=2,byrow=T)
> X
> fisher.test(X,alternative="two.sided",conf.level=0.95)

Fisher's Exact Test for Count Data

data: X
p-value = 0.6882
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
0.04625243 3.58478157
sample estimates:
odds ratio
0.527113

$$P_T = \begin{pmatrix} 0.0174 \\ 0.1051 \\ 0.2522 \\ 0.3118 \\ 0.2144 \\ 0.0819 \\ 0.016 \\ 0.0012 \end{pmatrix}$$

^ Here the calculations from Rosner & R's output are similar in result, but clearly do not match. Perhaps this represents rounding error.

Note that R also has a Hypergeometric Distribution function that may be used here:

COMMANDS:
> X=0:7
> X
> dhyper(X,7,53,25)

[1] 0.017411703 0.105070619 0.252169485 0.311822481 0.214377956 0.081853401
[7] 0.016049687 0.001244670

^ These numbers match calculation of vector P_T above

$$P_T = \begin{pmatrix} 0.0174 \\ 0.1051 \\ 0.2522 \\ 0.3118 \\ 0.2144 \\ 0.0819 \\ 0.016 \\ 0.0012 \end{pmatrix}$$

> phyper(X,7,53,25)

[1] 0.01741170 0.12248232 0.37465181 0.68647429 0.90085224 0.98270564 0.99875533
[8] 1.00000000
^ These numbers are the cumulative probabilities used in calcuation of A above.

$$A = 0.3747$$

This suggests that R's fisher.test() calculates probabilites somewhat differently...

ORIGIN ≡ 0

"Simple" Linear Regression

Linear Regression and the so-called "General Linear Model" represent a class of methods that seek to relate values of an observed "dependent" random variable (Y) that is Normally distributed to one or more "independent" (or predictor) variables (X) using a linear function analogous to a linear transformation - i.e., using only translation and change of scale. We typically employ "linear coeficients" (not to be confused with the probability of types I & II errors in statistical tests) to describe translation (α) and change of scale (β). Thus a function such as $Y = 5 + 23X$ qualifies as a linear function whereas $Y = X^2$ or $Y = 5 + X^3$ would not. Note, however, that with the use of an appropriate non-linear transformations of the data, many non-linear functions can be treated by general linear methods also. For instance, taking the square root allows one to model $Y = X^2$ as $Y = a\sqrt{X}$, and taking logs allows one to model the famous allometric equation: $Y = aX^b$ as $\ln(Y) = \ln(a) + b(\ln(X))$.

Assumptions:

- **Standard Linear Regression depends on specifying in advance which variable is to be considered 'dependent' and which 'independent'. This decision matters as changing roles for Y & X usually produces a different result.**
- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

where: α is the y intercept of the regression line (translation)
β is the slope of the regression line (scaling coefficient)
ε is the error factor in prediction of Y given that it is a random variable distributed as $N(0,\sigma^2)$.

Least Squares Estimation of the Regression Line:

Sums of Squares and Cross Products corrected for mean location:

$$L_{xx} := \sum_i \left(X_i - X_{bar}\right)^2$$ **< corrected Sum of squares of X**

$$L_{yy} := \sum_i \left(Y_i - Y_{bar}\right)^2$$ **< corrected Sum of squares of Y**

$$L_{xy} := \sum_i \left(X_i - X_{bar}\right) \cdot \left(Y_i - Y_{bar}\right)$$ **< corrected Sum of cross products**

Estimated Regression Coefficients for $Y = \alpha + \beta X$:

$$b := \frac{L_{xy}}{L_{xx}}$$ **< sample estimate of β**

$$a := Y_{bar} - b \cdot X_{bar}$$ **< sample estimate of α**

Estimted values of Y (Y_{hat}):

$$Y_{hat_i} := a + b \cdot X_i$$ **< using estimated coefficients and each value of the independent variable to estimate dependent value points on the Regression line.**

Residuals:

$$e_i := Y_{hat_i} - Y_i$$ **< deviation of each value Y_i from Regression line = $Yhat_i$**

Example:

Rosner Example 11.8 p. 471 K := READPRN("c:/2007BiostatsData/GreenTouchstone Study2.txt")

Assumptions:

- Let independent variable X be Estriol level in the first column of K
- Let dependent variable Y be Birthweight in the second column of K
- Y is a random sample ~N(μ,σ^2)

Model:

$Y = \alpha + \beta X + \varepsilon$

K =

	0	1
0	7	25
1	9	25
2	9	25
3	12	27
4	14	27
5	16	27
6	16	24
7	14	30
8	16	30
9	16	31
10	17	30
11	19	31
12	21	30
13	24	28
14	15	32
15	16	32

Least Squares Estimation of the Regression Line:

$X := K^{\langle 0 \rangle}$ $X_{bar} := mean(X)$ $X_{bar} = 17.2258$

$Y := K^{\langle 1 \rangle}$ $Y_{bar} := mean(Y)$ $Y_{bar} = 32.0323$

$n := length(Y)$ $i := 0 .. n - 1$ $n = 31$

Sums of Squares and Cross Products corrected for mean location:

$$L_{xx} := \sum_i \left(X_i - X_{bar}\right)^2 \qquad L_{xx} = 677.4194$$

^ verified p. 471

$$L_{yy} := \sum_i \left(Y_i - Y_{bar}\right)^2 \qquad L_{yy} = 680.9677$$

^ close but not the same as p. 478

$$L_{xy} := \sum_i \left(X_i - X_{bar}\right) \cdot \left(Y_i - Y_{bar}\right) \qquad L_{xy} = 410.7742$$

^ close but not the same as p. 471

My guesss here is that there's an error in his Table 11.1 or my GreenTouchStone Study.xls

Estimated Regression Coefficients for $Y = \alpha + \beta X$:

$b := \frac{L_{xy}}{L_{xx}}$ $b = 0.6064$ **< sample estimate of β**

$a := Y_{bar} - b \cdot X_{bar}$ $a = 21.5869$ **< sample estimate of α**

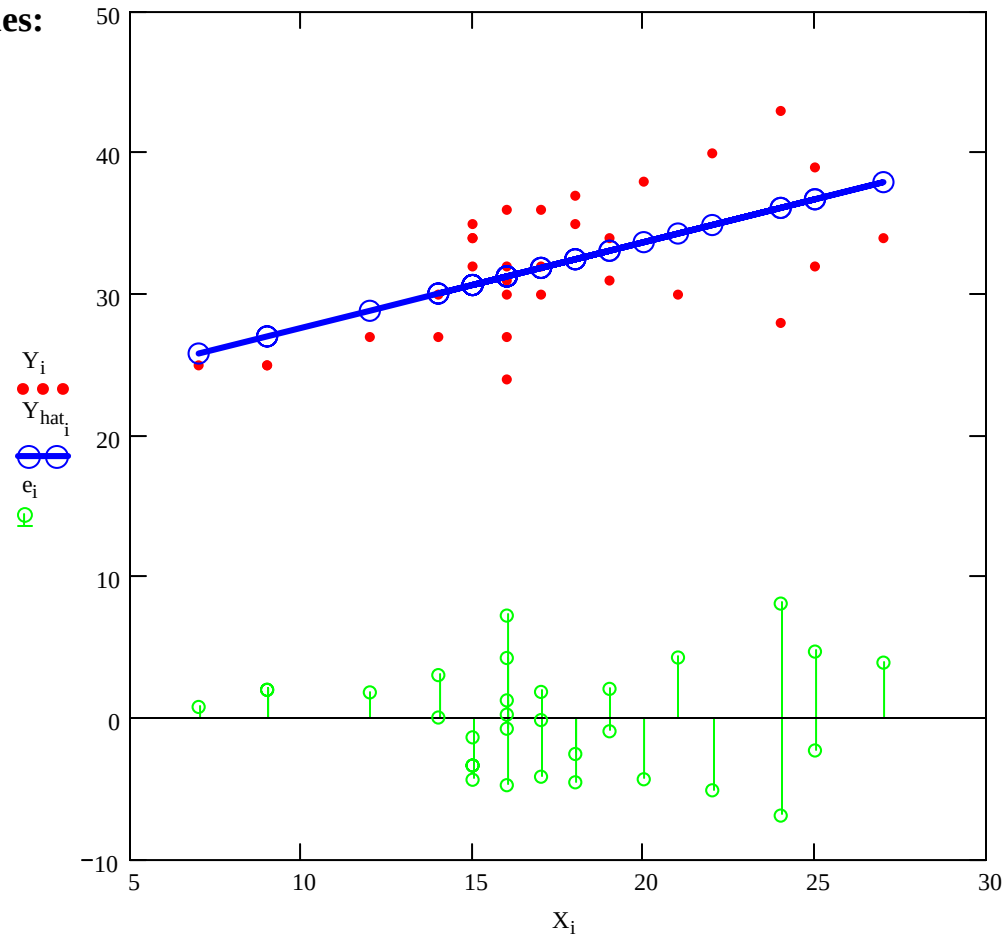
Estimted values of Y (Y_{hat}):

$Yhat_i := a + b \cdot X_i$ **< using estimated coefficients and each value of the independent variable to estimate dependent value points on the Regression line.**

Residuals: **< deviation of each value Y_i from Regression line = $Yhat_i$**

$e_i := Yhat_i - Y_i$

Plot of Values:



Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/GreenTouchstone.txt")
> K
> attach(K)
> X=Estriol
> Y=BirthWeight
> lsfit(X,Y)
```

coefficients confirmed

a = 21.5869

b = 0.6064

```
$coefficients
Intercept        X
21.586857  0.606381

$residuals
 [1] -0.83152381 -2.04428571 -2.04428571 -1.86342857 -3.07619048
 [6] -4.28895238 -7.28895238 -0.07619048 -1.28895238 -0.28895238
[11] -1.89533333 -2.10809524 -4.32085714 -8.14000000  1.31742857
[16]  0.71104762  0.10466667 -4.74638095 -3.95914286  3.31742857
[21]  3.31742857  4.31742857  4.71104762  0.89190476  2.49828571
[26]  4.10466667  4.49828571  4.28552381  5.07276190  2.25361905
[31]  6.86000000
```

And much more stuff...

e =

	0
0	0.8315
1	2.0443
2	2.0443
3	1.8634
4	3.0762
5	4.289
6	7.289
7	0.0762
8	1.289
9	0.289
10	1.8953
11	2.1081
12	4.3209
13	8.14
14	-1.3174
15	-0.711

Prototype in R:

COMMANDS:
> lm(Y~X)

Call:
lm(formula = Y ~ X)

Coefficients:

(Intercept)	X
21.5869	0.6064

< just the estimates of α and β

COMMANDS:
> predict(lm(Y~X))
> fitted(lm(Y~X))

1	2	3	4	5	6	7	8
25.83152	27.04429	27.04429	28.86343	30.07619	31.28895	31.28895	30.07619
9	10	11	12	13	14	15	16
31.28895	31.28895	31.89533	33.10810	34.32086	36.14000	30.68257	31.28895
17	18	19	20	21	22	23	24
31.89533	36.74638	37.95914	30.68257	30.68257	30.68257	31.28895	33.10810
25	26	27	28	29	30	31	
32.50171	31.89533	32.50171	33.71448	34.92724	36.74638	36.14000	

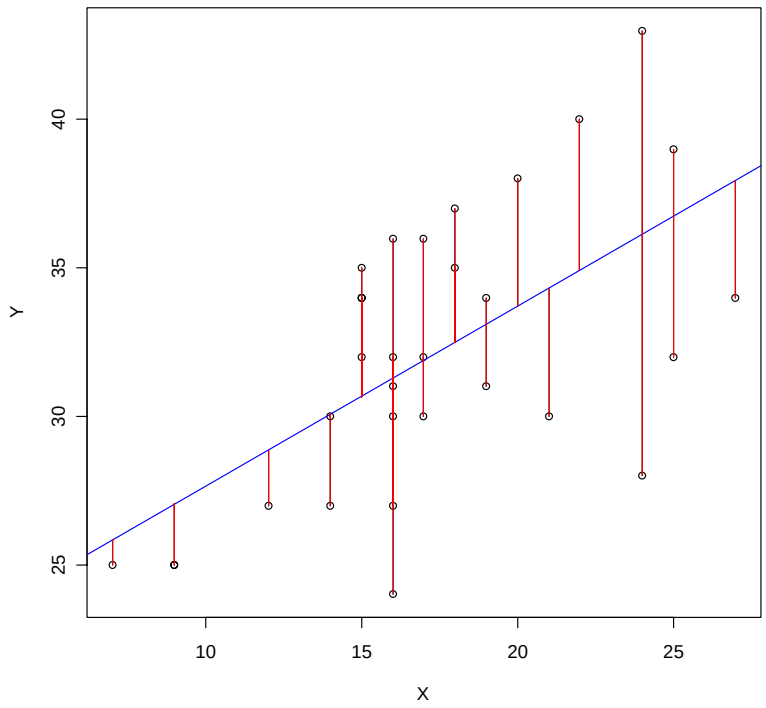
< Y_{hat} values calculated

COMMANDS:
> PRED=predict(lm(Y~X))
> RESIDUALS=resid(lm(Y~X))
> RESULTS=data.frame(Y,X,PRED,RESIDUALS)
> RESULTS
> plot(X,Y)
> abline(lm(Y~X),col="blue")
> segments(X,predict(lm(Y~X)),X,Y,col="red")

< In data frame format:

	Y	X	PRED	RESIDUALS
1	25	7	25.83152	-0.83152381
2	25	9	27.04429	-2.04428571
3	25	9	27.04429	-2.04428571
4	27	12	28.86343	-1.86342857
5	27	14	30.07619	-3.07619048
6	27	16	31.28895	-4.28895238
7	24	16	31.28895	-7.28895238
8	30	14	30.07619	-0.07619048
9	30	16	31.28895	-1.28895238
10	31	16	31.28895	-0.28895238
11	30	17	31.89533	-1.89533333
12	31	19	33.10810	-2.10809524
13	30	21	34.32086	-4.32085714
14	28	24	36.14000	-8.14000000
15	32	15	30.68257	1.31742857
16	32	16	31.28895	0.71104762
17	32	17	31.89533	0.10466667
18	32	25	36.74638	-4.74638095
19	34	27	37.95914	-3.95914286
20	34	15	30.68257	3.31742857
21	34	15	30.68257	3.31742857
22	35	15	30.68257	4.31742857
23	36	16	31.28895	4.71104762
24	34	19	33.10810	0.89190476
25	35	18	32.50171	2.49828571
26	36	17	31.89533	4.10466667
27	37	18	32.50171	4.49828571
28	38	20	33.71448	4.28552381
29	40	22	34.92724	5.07276190
30	39	25	36.74638	2.25361905
31	43	24	36.14000	6.86000000

Plot with Fitted values (blue) and Residuals (red)



Compare Values:

X	Y	Y_{hat}	e	$Y_{hat} - e$
7	25	25.8315	0.8315	25
9	25	27.0443	2.0443	25
9	25	27.0443	2.0443	25
12	27	28.8634	1.8634	27
14	27	30.0762	3.0762	27
16	27	31.289	4.289	27
16	24	31.289	7.289	24
14	30	30.0762	0.0762	30
16	30	31.289	1.289	30
16	31	31.289	0.289	31
17	30	31.8953	1.8953	30
19	31	33.1081	2.1081	31
21	30	34.3209	4.3209	30
24	28	36.14	8.14	28
15	32	30.6826	–1.3174	32
16	32	31.289	–0.711	32
17	32	31.8953	–0.1047	32
25	32	36.7464	4.7464	32
27	34	37.9591	3.9591	34
15	34	30.6826	–3.3174	34
15	34	30.6826	–3.3174	34
15	35	30.6826	–4.3174	35
16	36	31.289	–4.711	36
19	34	33.1081	–0.8919	34
18	35	32.5017	–2.4983	35
17	36	31.8953	–4.1047	36
18	37	32.5017	–4.4983	37
20	38	33.7145	–4.2855	38
22	40	34.9272	–5.0728	40
25	39	36.7464	–2.2536	39
24	43	36.14	–6.86	43

ORIGIN ≡ 0

ANOVA for "Simple" Linear Regression

Goodness of fit of a fitted regression line can be tested using the F-test for Regression (also known as the ANOVA for "Analysis of Variance" for Regression) or alternatively, and equivalently, the t-test of Regression. Here we consider the ANOVA approach.

Assumptions:

- **Standard Linear Regression depends on specifying in advance which variable is to be considered 'dependent' and which 'independent'. This decision matters as changing roles for Y & X usually produces a different result.**
- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

where: α is the y intercept of the regression line (translation)
β is the slope of the regression line (scaling coefficient)
ε is the error factor in prediction of Y given that it is a random variable with $N(0,\sigma^2)$

Variance (Sum of Squares) Decomposition of the Regression:

Once a regression model ($Y = \alpha + \beta X + \varepsilon$) is fitted with data, one still needs to determine how useful the regression might be, especially whether knowledge about the X_i provide insight into interpreting the Y_i as a random variable from a Normal distribution with error ε_i. This is done by considering a "partition" of total variance in the sample of Y_i.

Note that variance here is addressed in terms of "Sums of Squares" the numerator as this is the only important part of variance to consider at this point:

$$SS_T := \sum_i \left(Y_i - Y_{bar}\right)^2$$ < **Total Sum of Squares**

$$SS_R := \sum_i \left(Y_{hat_i} - Y_{bar}\right)^2$$ < **Regression Sum of Squares**

$$SS_E := \sum_i \left(Y_i - Y_{hat_i}\right)^2$$ < **Residual (also called "Error") Sum of Squares**

These Sums of Squares tally as follows:

$$SS_T := SS_R + SS_E$$

And the ratio of SS_R to SS_E can be used as a measure of "fit" of the data to the regression.

Hypotheses:

H_0: $\beta = 0$ < **Slope of the Regression is zero implying no relationship between X_i and Y_i**

H_1: $\beta <> \beta$ < **Two sided test**

ANOVA for Linear Regression:

Compute ANOVA Table:

ANOVA TABLE

	SS	df	MS
Regression:	SS_R	1	$MS_R := \frac{SS_R}{1}$
Residual:	SS_E	$(n - 2)$	$MS_E := \frac{SS_E}{(n - 2)}$
TOTAL:	SS_T	$(n - 1)$	$MS_T := \frac{SS_T}{(n - 1)}$

Test Statistic:

$F := \frac{MS_R}{MS_E}$ **< F is the ratio of sample variances**

Sampling Distribution:

If Assumptions hold and H_0 is true, then F ~$F_{(1)/(n-2)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := inverse\Phi_F(1 - \alpha)$ $CV := qF(1 - \alpha, 1, n - 2)$

Decision Rule:

IF F > CV, THEN REJECT H_0
OTHERWISE ACCEPT H_0

Probability Value:

P = 1- Φ_F(F) $P := 1 - pF(F, 1, n - 2)$

Example:

Rosner Example 11.12 p. 477 $K := READPRN("c:/2007BiostatsData/GreenTouchstone Study2.txt")$

Assumptions:

- **Let independent variable X be Estriol level in the first column of K**
- **Let dependent variable Y be Birthweight in the second column of K**
- **Y is a random sample ~N(μ,σ^2)**

Model:

$Y = \alpha + \beta X + \varepsilon$

Least Squares Estimation of the Regression Line:

$X := K^{\langle 0 \rangle}$ $X_{bar} := mean(X)$ $X_{bar} = 17.2258$

$Y := K^{\langle 1 \rangle}$ $Y_{bar} := mean(Y)$ $Y_{bar} = 32.0323$

$n := length(Y)$ $i := 0 .. n - 1$ $n = 31$

Sums of Squares and Cross Products corrected for mean location:

$$L_{xx} := \sum_i \left(X_i - X_{bar}\right)^2 \qquad L_{xx} = 677.4194$$

$$L_{yy} := \sum_i \left(Y_i - Y_{bar}\right)^2 \qquad L_{yy} = 680.9677$$

$$L_{xy} := \sum_i \left(X_i - X_{bar}\right) \cdot \left(Y_i - Y_{bar}\right) \qquad L_{xy} = 410.7742$$

Estimated Regression Coefficients for $Y = \alpha + \beta X$:

$b := \frac{L_{xy}}{L_{xx}}$ $b = 0.6064$ **< sample estimate of β**

$a := Y_{bar} - b \cdot X_{bar}$ $a = 21.5869$ **< sample estimate of α**

Estimted values of Y (Y_{hat}):

$Y_{hat_i} := a + b \cdot X_i$ **< using estimated coefficients and each value of the independent variable to estimate dependent value points on the Regression line.**

Residuals: **< deviation of each value Y_i from Regression line = $Yhat_i$**

$e_i := Y_{hat_i} - Y_i$

Sums of Squares:

$SS_T := \sum_i \left(Y_i - Y_{bar}\right)^2$ **< Total Sum of Squares - same as L_{yy} above**

$SS_R := \sum_i \left(Y_{hat_i} - Y_{bar}\right)^2$ **< Regression Sum of Squares**

$SS_E := \sum_i \left(Y_i - Y_{hat_i}\right)^2$ **< Residual (also called "Error") Sum of Squares**

Hypotheses:

H_0: $\beta = 0$ **< Slope of the Regression is zero implying no relationship between X_i and Y_i**

H_1: $\beta <> \beta$ **< Two sided test**

ANOVA for Linear Regression:

Compute ANOVA Table:

ANOVA TABLE

	SS	df	MS	
Regression:	$SS_R = 249.0856$	1	$MS_R := \frac{SS_R}{1}$	$MS_R = 249.0856$
Residual:	$SS_E = 431.8821$	$(n - 2)$	$MS_E := \frac{SS_E}{(n - 2)}$	$MS_E = 14.8925$
TOTAL:	$SS_T = 680.9677$	$(n - 1)$	$MS_T := \frac{SS_T}{(n - 1)}$	$MS_T = 22.6989$

Test Statistic:

$F := \frac{MS_R}{MS_E}$ **< F is the ratio of sample variances** $F = 16.7256$

Sampling Distribution:

If Assumptions hold and H_0 is true, then $F \sim F_{(1)/(n-2)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF(1 - \alpha, 1, n - 2)$ $CV = 4.183$

Decision Rule:

IF F > CV, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F = 16.7256$ $CV = 4.183$

Probability Value:

Powerful stuff!!

$P := 1 - pF(F, 1, n - 2)$ $P = 0.0003$

Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/GreenTouchstone.txt")
> K
> attach(K)
> X=Estriol
> Y=BirthWeight
> X
> lm(Y~X)
```

Call:
lm(formula = Y ~ X)

Coefficients:

(Intercept)	X
21.5869	0.6064

< a & b values same as above

COMMANDS:

```
> anova(lm(Y~X))
or
> anova.lm(lm(Y~X))
```

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	1	249.09	249.09	16.726	0.0003134 ***
Residuals	29	431.88	14.89		

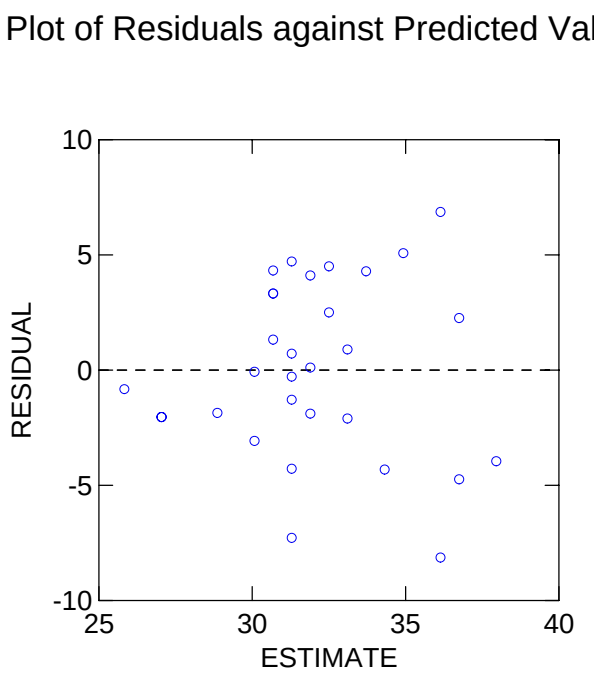
Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

^ ANOVA results match above

Prototype in SYSTAT:

```
     Dep Var: BIRTHWT   N: 31   Multiple R: 0.60480   Squared multiple R: 0.36578

      Adjusted squared multiple R: 0.34391   Standard error of estimate: 3.85908

   Effect         Coefficient    Std Error     Std Coef Tolerance     t   P(2 Tail)

   CONSTANT          21.58686      2.64645      0.00000    .       8.15690  0.00000
   ESTRIOL            0.60638      0.14827      0.60480   1.00000  4.08969  0.00031

                            Analysis of Variance

Source             Sum-of-Squares   df  Mean-Square     F-ratio       P

Regression             249.08565     1    249.08565    16.72559     0.00031
Residual               431.88210    29     14.89249
   ---------------------------------------------------------------------------


                        Durbin-Watson D Statistic     0.714
                        First Order Autocorrelation   0.588
```

^ values of tables match results above



ORIGIN ≡ 0

The t-Test Approach and Interval Estimation for "Simple" Linear Regression

Goodness of fit of a fitted regression line can be tested using a t-test approach. This method also provides for a direct estimation of confidence intervals for the slope parameter β. Interval estimates can also be derived for the Regression line itself as a mean, as well as for prediction of "new" observations.

Assumptions:

- **Standard Linear Regression depends on specifying in advance which variable is to be considered 'dependent' and which 'independent'. This decision matters as changing roles for Y & X usually produces a different result.**
- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

where: α is the y intercept of the regression line (translation)
β is the slope of the regression line (scaling coefficient)
ε is the error factor in prediction of Y given that it is a random variable with $N(0,\sigma^2)$

t-Test for Simple Linear Regression:

Hypotheses:

H_0: β = 0 < **Slope of the Regression is zero implying no relationship between X_i and Y_i**

H_1:β <> 0 < **Two sided test**

Test Statistic:

$$t := \frac{b}{\sqrt{\frac{MS_E}{L_{xx}}}}$$

< **b is unbiased point estimate of β**
< **MSE is Mean Square Error from ANOVA table also denoted $s^2_{x.y}$**
< **L_{xx} Corrected sums of squares of X as defined in Regression**

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n-2\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n-2\right)$

Note degrees of freedom = (n-2)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t) < **Rosner Eq 11.8, p. 481**

$P := \min[\, 2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2))\,]$

Confidence Interval for the Regression (β):

$$CI_R := \left(b + C_1 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \quad b + C_2 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \right)$$

Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find
< **the Lower Bound of the CI.**

Confidence Interval for Regression Estimates Y_{hat} and New Predictions of Y:

One or more values of X_n must be explicitly specified to obtain a prediction CI for Y_{hat}:

$X_{n_i} := X_i$ **< here using all original values of X, but any X values may be specified instead...**

Confidence Interval (CI):

$$CI_{RL_i} := Y_{hat_i} + C_1 \cdot \sqrt{MS_E \cdot \left[\frac{1}{n} + \frac{\left(X_{n_i} - X_{bar}\right)^2}{L_{xx}}\right]} \qquad CI_{RU_i} := Y_{hat_i} + C_2 \cdot \sqrt{MS_E \cdot \left[\frac{1}{n} + \frac{\left(X_{n_i} - X_{bar}\right)^2}{L_{xx}}\right]}$$

Prediction Interval (PI):

$$PI_{RL_i} := Y_{hat_i} + C_1 \cdot \sqrt{MS_E \cdot \left[1 + \frac{1}{n} + \frac{\left(X_{n_i} - X_{bar}\right)^2}{L_{xx}}\right]} \qquad PI_{RU_i} := Y_{hat_i} + C_2 \cdot \sqrt{MS_E \cdot \left[1 + \frac{1}{n} + \frac{\left(X_{n_i} - X_{bar}\right)^2}{L_{xx}}\right]}$$

Example:

Rosner Example 11.12 p. 477 K := READPRN("c:/2007BiostatsData/GreenTouchstone Study2.txt")

Assumptions:

- **Let independent variable X be Estriol level in the first column of K**
- **Let dependent variable Y be Birthweight in the second column of K**
- **Y is a random sample ~ N(μ,σ^2)**

Model:

$Y = \alpha + \beta X + \varepsilon$

Least Squares Estimation of the Regression Line:

$X := K^{\langle 0 \rangle}$ $X_{bar} := mean(X)$ $X_{bar} = 17.2258$

$Y := K^{\langle 1 \rangle}$ $Y_{bar} := mean(Y)$ $Y_{bar} = 32.0323$

$n := length(Y)$ $i := 0 .. n - 1$ $n = 31$

Sums of Squares and Cross Products corrected for mean location:

$$L_{xx} := \sum_i \left(X_i - X_{bar}\right)^2 \qquad L_{xx} = 677.4194$$

$$L_{yy} := \sum_i \left(Y_i - Y_{bar}\right)^2 \qquad L_{yy} = 680.9677$$

$$L_{xy} := \sum_i \left(X_i - X_{bar}\right) \cdot \left(Y_i - Y_{bar}\right) \qquad L_{xy} = 410.7742$$

Estimated Regression Coefficients for $Y = \alpha + \beta X$:

$b := \frac{L_{xy}}{L_{xx}}$ $b = 0.6064$ **< sample estimate of β**

$a := Y_{bar} - b \cdot X_{bar}$ $a = 21.5869$ **< sample estimate of α**

Estimted values of Y (Y_{hat}):

$Y_{hat_i} := a + b \cdot X_i$ **< using estimated coefficients and each value of the independent variable to estimate dependent value points on the Regression line.**

Residuals:

$e_i := Y_{hat_i} - Y_i$ **< deviation of each value Y_i from Regression line = $Yhat_i$**

Sums of Squares:

$SS_T := \sum_i \left(Y_i - Y_{bar}\right)^2$ **< Total Sum of Squares**

$SS_R := \sum_i \left(Y_{hat_i} - Y_{bar}\right)^2$ **< Regression Sum of Squares**

$SS_E := \sum_i \left(Y_i - Y_{hat_i}\right)^2$ **< Residual (also called "Error") Sum of Squares**

ANOVA for Linear Regression:

Compute ANOVA Table:

ANOVA TABLE

	SS	df	MS	
Regression:	$SS_R = 249.0856$	1	$MS_R := \frac{SS_R}{1}$	$MS_R = 249.0856$
Residual:	$SS_E = 431.8821$	$(n - 2)$	$MS_E := \frac{SS_E}{(n-2)}$	$MS_E = 14.8925$
TOTAL:	$SS_T = 680.9677$	$(n - 1)$	$MS_T := \frac{SS_T}{(n-1)}$	$MS_T = 22.6989$

Hypotheses:

H_0: $\beta = 0$ **< Slope of the Regression is zero implying no relationship between X_i and Y_i**

H_1:$\beta <> 0$ **< Two sided test**

Test Statistic:

$$t := \frac{b}{\sqrt{\frac{MS_E}{L_{xx}}}} \qquad t = 4.0897$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n - 2\right)$ $C_1 = -2.0452$

$C_2 := qt\left(1 - \frac{\alpha}{2}, n - 2\right)$ $C_2 = 2.0452$

Note degrees of freedom = (n-2)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 4.0897$ $C_1 = -2.0452$ $C_2 = 2.0452$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2)) \,]$ $P = 0.0003$

Confidence Interval for the Regression (β):

$$CI_R := \left(b + C_1 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \quad b + C_2 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \right)$$ $b = 0.6064$

$CI_R = (0.3031 \quad 0.9096)$

Confidence Interval for Regression Estimates Y_{hat} and New Predictions of Y:

One or more values of X_n must be explicitly specified to obtain a prediction CI for Y_{hat}:

$X_{n_i} := X_i$ **< here using all original values of X, but any X values may be specified instead...**

Confidence Interval (CI_R):

$$CI_{RL_i} := Y_{hat_i} + C_1 \cdot \sqrt{MS_E \cdot \left[\frac{1}{n} + \frac{(X_{n_i} - X_{bar})^2}{L_{xx}} \right]} \qquad CI_{RU_i} := Y_{hat_i} + C_2 \cdot \sqrt{MS_E \cdot \left[\frac{1}{n} + \frac{(X_{n_i} - X_{bar})^2}{L_{xx}} \right]}$$

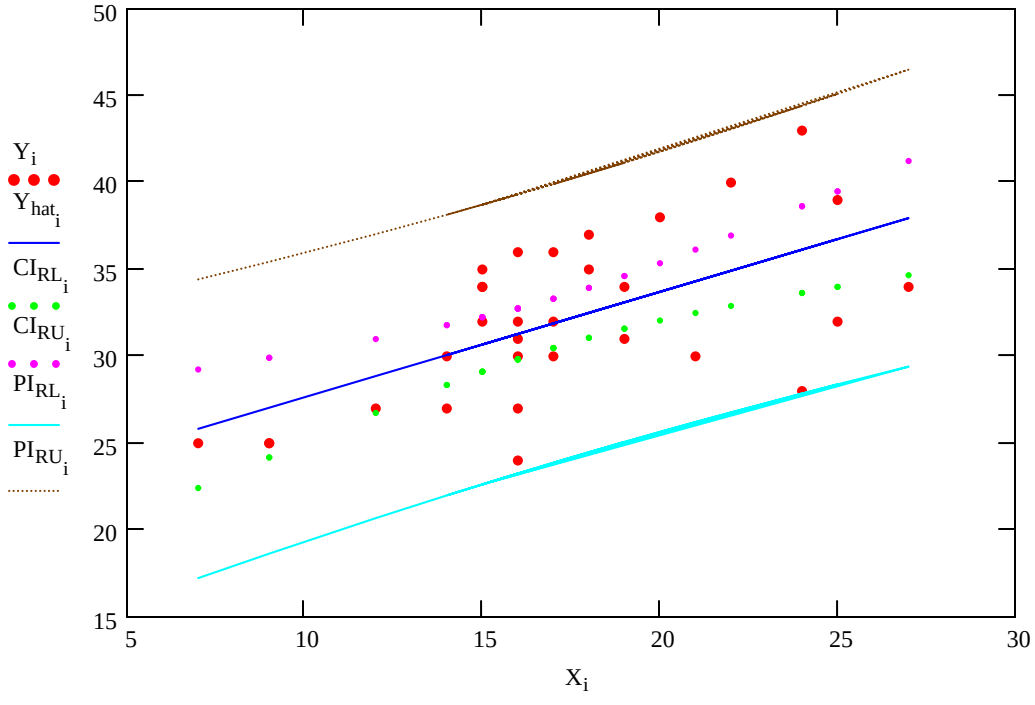
$(CI_{RL_1} \quad CI_{RU_1}) = (24.1752 \quad 29.9134)$ **for point: X=** $X_1 = 9$

Prediction Interval (PI_R):

$$PI_{RL_i} := Y_{hat_i} + C_1 \cdot \sqrt{MS_E \cdot \left[1 + \frac{1}{n} + \frac{(X_{n_i} - X_{bar})^2}{L_{xx}} \right]} \qquad PI_{RU_i} := Y_{hat_i} + C_2 \cdot \sqrt{MS_E \cdot \left[1 + \frac{1}{n} + \frac{(X_{n_i} - X_{bar})^2}{L_{xx}} \right]}$$

$(PI_{RL_1} \quad PI_{RU_1}) = (18.6463 \quad 35.4423)$ **for point: X=** $X_1 = 9$

Plot of Regression and Prediction Interval:



Prototype in SYSTAT:

```
     Dep Var: BIRTHWT   N: 31   Multiple R: 0.60480   Squared multiple R: 0.36578

      Adjusted squared multiple R: 0.34391   Standard error of estimate: 3.85908

   Effect         Coefficient    Std Error     Std Coef Tolerance     t   P(2 Tail)

   CONSTANT          21.58686      2.64645      0.00000    .       8.15690  0.00000
   ESTRIOL            0.60638      0.14827      0.60480   1.00000  4.08969  0.00031

                            Analysis of Variance

Source             Sum-of-Squares   df  Mean-Square     F-ratio       P

Regression             249.08565     1    249.08565    16.72559     0.00031
Residual               431.88210    29     14.89249
   ---------------------------------------------------------------------------


                        Durbin-Watson D Statistic     0.714
                        First Order Autocorrelation   0.588
```

Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/GreenTouchstone.txt")
>K
> attach(K)
> X=Estriol
> Y=BirthWeight
> summary(lm(Y~X))
```

Call:
lm(formula = Y ~ X)

Residuals:

Min	1Q	Median	3Q	Max
-8.14000	-2.07619	-0.07619	3.31743	6.86000

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.5869	2.6465	8.157	5.4e-09 ***
X	0.6064	0.1483	4.090	0.000313 ***

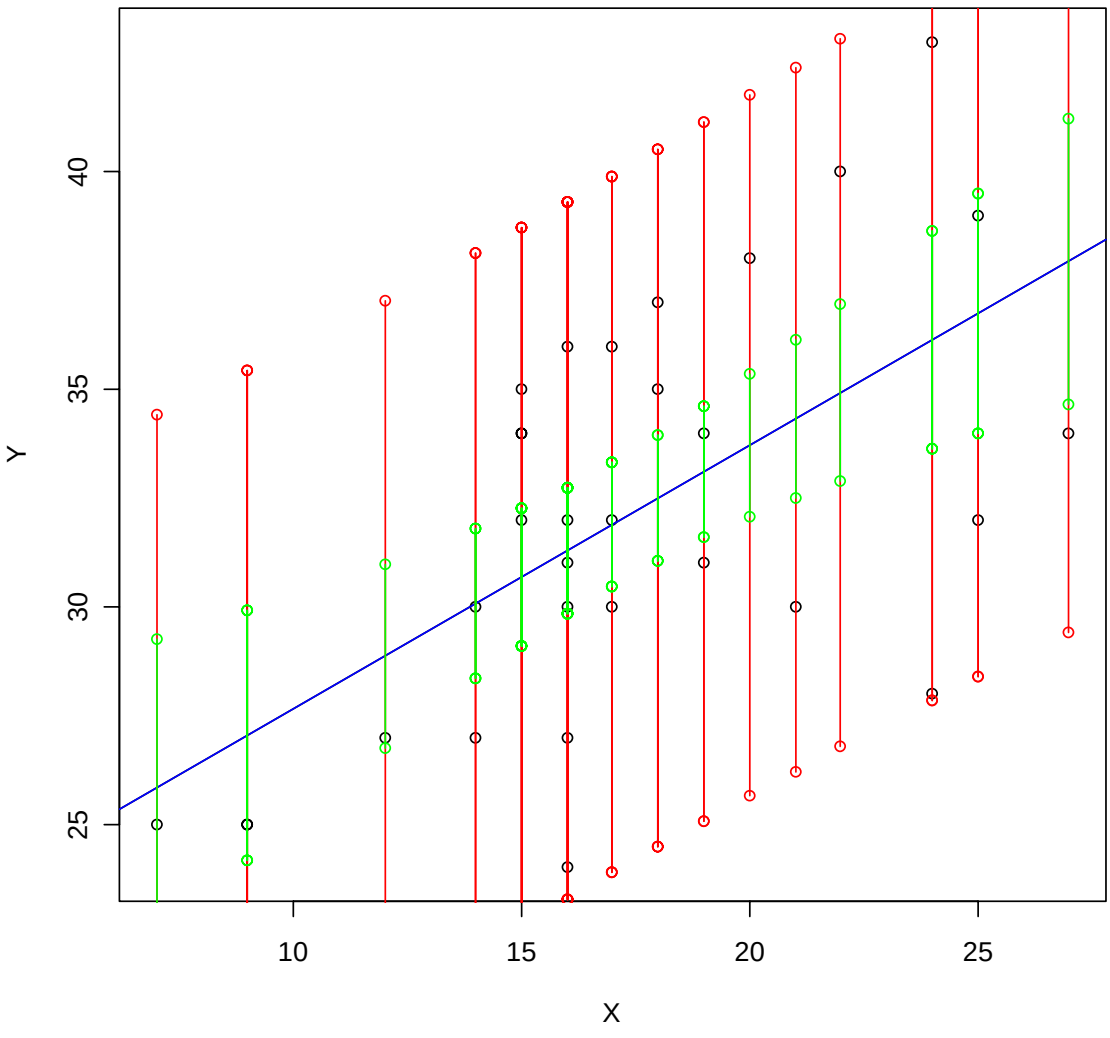
Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Residual standard error: 3.859 on 29 degrees of freedom
Multiple R-Squared: 0.3658, Adjusted R-squared: 0.3439
F-statistic: 16.73 on 1 and 29 DF, p-value: 0.0003134

Plotting Intervals in R:

COMMANDS:

```
> PRED=predict(lm(Y~X),interval="prediction",level=0.95)
> PR=data.frame(PRED)
> PR
> CONF=predict(lm(Y~X),interval="confidence",level=0.95)
>CN=data.frame(CONF)
> plot(X,Y)
> abline(lm(Y~X),col="blue")
> segments(X,PR$lwr,X,PR$upr,col="red")
> segments(X,CN$lwr,X,CN$upr,col="green")
> points(X,CN$lwr,col="green")
> points(X,CN$upr,col="green")
> points(X,PR$lwr,col="red")
> points(X,PR$upr,col="red")
```



Comparison of Confidence and Prediction Intervals:

In R:

COMMANDS:
> CN
> PR

CONFIDENCE INTERVAL (CN)

	fit	lwr	upr
1	25.83152	22.42192	29.24113
2	27.04429	24.17517	29.91340
3	27.04429	24.17517	29.91340
4	28.86343	26.73721	30.98965
5	30.07619	28.35386	31.79852
6	31.28895	29.82345	32.75445
7	31.28895	29.82345	32.75445
8	30.07619	28.35386	31.79852
9	31.28895	29.82345	32.75445
10	31.28895	29.82345	32.75445
11	31.89533	30.47611	33.31456
12	33.10810	31.59186	34.62433
13	34.32086	32.49893	36.14278
14	36.14000	33.64411	38.63589
15	30.68257	29.11251	32.25263
16	31.28895	29.82345	32.75445
17	31.89533	30.47611	33.31456
18	36.74638	33.99550	39.49726
19	37.95914	34.67360	41.24469
20	30.68257	29.11251	32.25263
21	30.68257	29.11251	32.25263
22	30.68257	29.11251	32.25263
23	31.28895	29.82345	32.75445
24	33.10810	31.59186	34.62433
25	32.50171	31.06483	33.93859
26	31.89533	30.47611	33.31456
27	32.50171	31.06483	33.93859
28	33.71448	32.06607	35.36288
29	34.92724	32.90103	36.95345
30	36.74638	33.99550	39.49726
31	36.14000	33.64411	38.63589

PREDICTION (PR)

	fit	lwr	upr
1	25.83152	17.23384	34.42920
2	27.04429	18.64628	35.44229
3	27.04429	18.64628	35.44229
4	28.86343	20.68935	37.03751
5	30.07619	21.99775	38.15463
6	31.28895	23.26135	39.31656
7	31.28895	23.26135	39.31656
8	30.07619	21.99775	38.15463
9	31.28895	23.26135	39.31656
10	31.28895	23.26135	39.31656
11	31.89533	23.87605	39.91462
12	33.10810	25.07107	41.14512
13	34.32086	26.22060	42.42111
14	36.14000	27.86206	44.41794
15	30.68257	22.63522	38.72992
16	31.28895	23.26135	39.31656
17	31.89533	23.87605	39.91462
18	36.74638	28.38803	45.10473
19	37.95914	29.40990	46.50838
20	30.68257	22.63522	38.72992
21	30.68257	22.63522	38.72992
22	30.68257	22.63522	38.72992
23	31.28895	23.26135	39.31656
24	33.10810	25.07107	41.14512
25	32.50171	24.47929	40.52414
26	31.89533	23.87605	39.91462
27	32.50171	24.47929	40.52414
28	33.71448	25.65148	41.77748
29	34.92724	26.77860	43.07587
30	36.74638	28.38803	45.10473
31	36.14000	27.86206	44.41794

As calculated above:

Y_{hat}:	CONFIDENCE INTERVAL:		PREDICTION INTERVAL:	
25.8315	22.4219	29.2411	17.2338	34.4292
27.0443	24.1752	29.9134	18.6463	35.4423
27.0443	24.1752	29.9134	18.6463	35.4423
28.8634	26.7372	30.9897	20.6894	37.0375
30.0762	28.3539	31.7985	21.9978	38.1546
31.289	29.8235	32.7545	23.2613	39.3166
31.289	29.8235	32.7545	23.2613	39.3166
30.0762	28.3539	31.7985	21.9978	38.1546
31.289	29.8235	32.7545	23.2613	39.3166
31.289	29.8235	32.7545	23.2613	39.3166
31.8953	30.4761	33.3146	23.876	39.9146
33.1081	31.5919	34.6243	25.0711	41.1451
34.3209	32.4989	36.1428	26.2206	42.4211
36.14	33.6441	38.6359	27.8621	44.4179
30.6826	29.1125	32.2526	22.6352	38.7299
Y_{hat} = 31.289	CI_{RL} = 29.8235	CI_{RU} = 32.7545	PI_{RL} = 23.2613	PI_{RU} = 39.3166
31.8953	30.4761	33.3146	23.876	39.9146
36.7464	33.9955	39.4973	28.388	45.1047
37.9591	34.6736	41.2447	29.4099	46.5084
30.6826	29.1125	32.2526	22.6352	38.7299
30.6826	29.1125	32.2526	22.6352	38.7299
30.6826	29.1125	32.2526	22.6352	38.7299
31.289	29.8235	32.7545	23.2613	39.3166
33.1081	31.5919	34.6243	25.0711	41.1451
32.5017	31.0648	33.9386	24.4793	40.5241
31.8953	30.4761	33.3146	23.876	39.9146
32.5017	31.0648	33.9386	24.4793	40.5241
33.7145	32.0661	35.3629	25.6515	41.7775
34.9272	32.901	36.9534	26.7786	43.0759
36.7464	33.9955	39.4973	28.388	45.1047
36.14	33.6441	38.6359	27.8621	44.4179

^ values match those derived from R.

ORIGIN ≡ 0

Association and Correlation in "Simple" Regression

Once it is determined by ANOVA F or t-tests that an association between variables exists by testing H_0: $\beta = 0$, summary statistics such as the coefficient of determination (r^2 or R^2) and coefficient of correlation (r) may prove helpful. More usefully, further inferences may be made concerning the degree of association. This may be done by testing of H_0: $\beta = \beta_0$ or providing confidence limits on β. Alternatively, one can test and provide confidence limits on the population correlation coefficient (ρ) via its sample estimate (r). Tests using correlation are particularly useful when the reseacher is unwilling to specify in advance which of two variables (X or Y) should be considered independent versus dependent.

Coefficient of Determination (R^2) and Coefficient of Correlation (r):

From values defined in constructing Regression or the ANOVA table:

Coefficient of Correlation:

$$r := \frac{L_{xy}}{\sqrt{L_{xx} \cdot L_{yy}}} \qquad r := \sqrt{\frac{SS_R}{SS_T}} \qquad r := \sqrt{1 - \frac{SS_E}{SS_T}}$$ **< equivalent**

Coefficient of Determination:

$$R_{sq} := r^2$$

$$R_{sq} := \frac{L_{xy}^2}{L_{xx} \cdot L_{yy}} \qquad R_{sq} := \frac{SS_R}{SS_T} \qquad R_{sq} := 1 - \frac{SS_E}{SS_T}$$ **< equivalent**

Note: Values of r and R^2 range between -1 and 1. The closer R^2 is to -1 or 1, the stronger the linear relationship between the variables is *potentially* observed. R^2 or r near zero suggests no association. However, no single number can capture the situation exactly. It is possible, for data to show non-linear relationships, and for there to be high correlation/determination without necessarily a "good"regression fit or precison in prediction.

Correlation Coefficient Related to Sample Covariance & Standard Deviation:

$$s_{xy} := \frac{L_{xy}}{(n-1)} \qquad s_x := \sqrt{L_{xx}} \qquad s_y := \sqrt{L_{yy}}$$

$$r := \frac{s_{xy}}{s_x \cdot s_y}$$ **< correlation coefficient in terms of covariance & standard deviations**

Correlation coefficient related to regression slope (b as estimate of β):

$$b := r \cdot \sqrt{\frac{L_{yy}}{L_{xx}}} \qquad b := r \cdot \frac{s_y}{s_x}$$ **< equivalent**

$$r := b \cdot \sqrt{\frac{L_{xx}}{L_{yy}}} \qquad r := b \cdot \frac{s_x}{s_y}$$ **< equivalent**

Example:

Rosner Example 11.12 p. 477 K := READPRN("c:/2007BiostatsData/GreenTouchstone Study2.txt")

Calculating the Correlation:

$X := K^{\langle 0 \rangle}$ $Y := K^{\langle 1 \rangle}$ $X_{bar} := mean(X)$ $X_{bar} = 17.2258$ $Y_{bar} := mean(Y)$ $Y_{bar} = 32.0323$

$n := length(Y)$ $i := 0 .. n - 1$ $n = 31$

Sums of Squares and Cross Products (from means):

$$L_{xx} := \sum_i \left(X_i - X_{bar}\right)^2 \qquad L_{yy} := \sum_i \left(Y_i - Y_{bar}\right)^2 \qquad L_{xy} := \sum_i \left(X_i - X_{bar}\right) \cdot \left(Y_i - Y_{bar}\right)$$

$L_{xx} = 677.4194$ $L_{yy} = 680.9677$ $L_{xy} = 410.7742$

Estimated Regression Coefficients for Y = α+ βX:

$b := \frac{L_{xy}}{L_{xx}}$ $b = 0.6064$ $a := Y_{bar} - b \cdot X_{bar}$ $a = 21.5869$

Estimted values of Y (Y_{hat}): **Residuals:**

$Y_{hat_i} := a + b \cdot X_i$ $e_i := Y_{hat_i} - Y_i$

ANOVA Sums of Squares:

$$SS_T := \sum_i \left(Y_i - Y_{bar}\right)^2 \qquad SS_R := \sum_i \left(Y_{hat_i} - Y_{bar}\right)^2 \qquad SS_E := \sum_i \left(Y_i - Y_{hat_i}\right)^2$$

$SS_T = 680.9677$ $SS_R = 249.0856$ $SS_E = 431.8821$

Coefficient of Correlation:

$r := \frac{L_{xy}}{\sqrt{L_{xx} \cdot L_{yy}}}$ $r = 0.6048$ $\sqrt{\frac{SS_R}{SS_T}} = 0.6048$ $\sqrt{1 - \frac{SS_E}{SS_T}} = 0.6048$ **< equivalent**

Coefficient of Determination:

$R_{sq} := r^2$ $R_{sq} = 0.3658$ $\frac{L_{xy}^2}{L_{xx} \cdot L_{yy}} = 0.3658$ $\frac{SS_R}{SS_T} = 0.3658$ $1 - \frac{SS_E}{SS_T} = 0.3658$ **< equivalent**

Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/GreenTouchstone.txt")
> K
> X=K$Estriol
> Y=K$BirthWeight
> summary(lm(Y~X))
```

```
Call:
lm(formula = Y ~ X)

Residuals:
    Min      1Q  Median      3Q     Max
-8.14000 -2.07619 -0.07619  3.31743  6.86000

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  21.5869     2.6465   8.157  5.4e-09 ***
X             0.6064     0.1483   4.090 0.000313 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.859 on 29 degrees of freedom
Multiple R-Squared: 0.3658,     Adjusted R-squared: 0.3439
F-statistic: 16.73 on 1 and 29 DF,  p-value: 0.0003134
```

R^2 reported here >

One sample t-Test for $\beta = \beta_0$:

This test allows statistical appraisal of specific values for slope (β) not just whether β is zero.

Note: this test is a generalization of the t-Test for H_0: β = 0.

Assumptions:

- **Standard Linear Regression depends on specifying in advance which variable is to be considered 'dependent' and which 'independent'. This decision matters as changing roles for Y & X usually produces a different result.**
- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

< where: α is the y intercept of the regression line (translation)
β is the slope of the regression line (scaling coefficient)
ε is the error factor in prediction of Y given that it is a random variable with $N(\mu,\sigma^2)$

Hypotheses:

H_0: β = β_0 **< Slope of the Regression is β_0 - this value must be explicitly stated.**

H_1: β <> β_0 **< Two sided test**

Test Statistic:

$$t := \frac{b - \beta_0}{\sqrt{\frac{MS_E}{L_{xx}}}}$$

< b is unbiased point estimate of β

< MSE is Mean Square Error from ANOVA table also denoted $s^2_{x.y}$

< L_{xx} Corrected sums of squares of X as defined in Regression

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right) \qquad C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$$

$$C_1 := qt\left(\frac{\alpha}{2}, n-2\right) \qquad C_2 := qt\left(1 - \frac{\alpha}{2}, n-2\right)$$

Note degrees of freedom = (n-2)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$$P := \min[\, 2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2))\,]$$

Confidence Interval for the Regression (β):

$$CI_R := \left(b + C_1 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \quad b + C_2 \cdot \sqrt{\frac{MS_E}{L_{xx}}}\right)$$

< Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find the Lower Bound of the CI.

Example:

observed slope:

From above: $b = 0.6064$ **So let's test:** $\beta_0 := 0.5$

We also need from ANOVA: $MS_E := \frac{SS_E}{n-2}$ **< Note degrees of freedom = (n-2)**

One sample t-Test for $\beta = \beta_0$:

Assumptions:

- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$Y = \alpha + \beta X + \varepsilon$

Hypotheses:

H_0: $\beta = \beta_0 = 0.5$ < Slope of the Regression is β_0 - this value must be explicitly stated.

H_1: $\beta <> \beta_0$ < Two sided test

Test Statistic:

$$t := \frac{b - \beta_0}{\sqrt{\frac{MS_E}{L_{xx}}}} \qquad t = 0.7175$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n-2\right)$ $C_2 := qt\left(1-\frac{\alpha}{2}, n-2\right)$ **< Note degrees of freedom = (n-2)**

$C_1 = -2.0452$ $C_2 = 2.0452$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 0.7175$ $C_1 = -2.0452$ $C_2 = 2.0452$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2))\,]$ $P = 0.4788$

Confidence Interval for the Regression (β):

$$CI_R := \left(b + C_1 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \quad b + C_2 \cdot \sqrt{\frac{MS_E}{L_{xx}}} \right)$$

$CI_R = (0.3031 \quad 0.9096)$ **< same CI as in the test H_0: $\beta = 0$ ...**

Prototype in R:

This test you must do by hand. Obtain MS_E from anova(lm(Y~X)). Calculate t statistic with formula above. Use function qt() for C_1 & C_2.

One sample t-Test for $\rho = 0$:

This test, using ρ instead of β, is an equivalent alternative to the previous H_0: $\beta=\beta_0$ t-test.

Assumptions:

- **Standard Linear Regression depends on specifying in advance which variable is to be considered 'dependent' and which 'independent'. This decision matters as changing roles for Y & X usually produces a different result.**
- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

< where: α is the y intercept of the regression line (translation)
β is the slope of the regression line (scaling coefficient)
ε is the error factor in prediction of Y given that it is a random variable with $N(\mu,\sigma^2)$

$$\rho = \beta(\sigma_x/\sigma_y)$$

< correlation coefficient ρ defined in terms of Regression slope β and standard deviations σ_x & σ_y.

Hypotheses:

H_0: $\rho = 0$ **< No correlation**

H_1: $\rho <> 0$ **< Two sided test**

Test Statistic:

$$t := \frac{r \cdot \sqrt{n-2}}{\sqrt{1-r^2}}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1-\frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n-2\right)$ $C_2 := qt\left(1-\frac{\alpha}{2}, n-2\right)$

Note degrees of freedom = (n-2)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2))]$

Example:

From above, observed correlation coefficient: $r = 0.6048$

One sample t-Test for $\rho = 0$:

Assumptions:

- **$Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.**
- **$X_1, X_2, X_3, \ldots, X_n$ (independent variable) with each value of X_i matched to Y_i**

Model:

$$Y = \alpha + \beta X + \varepsilon$$

$$\rho = \beta(\sigma_x/\sigma_y)$$

Hypotheses:

H_0: $\rho = 0$ **< No correlation**

H_1: $\rho <> 0$ **< Two sided test**

Test Statistic:

$$t := \frac{r \cdot \sqrt{n-2}}{\sqrt{1-r^2}} \qquad t = 4.0897$$ **< Same value reported from t-test of H_0: $\beta = 0$**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$$C_1 := qt\left(\frac{\alpha}{2}, n-2\right) \qquad C_2 := qt\left(1-\frac{\alpha}{2}, n-2\right)$$ **< Note degrees of freedom = (n-2)**

$C_1 = -2.0452$ $C_2 = 2.0452$

Decision Rule:

IF $|t| > C$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 4.0897$ $C_1 = -2.0452$ $C_2 = 2.0452$

Probability Value:

$$P := \min[\, 2 \cdot pt(t, n-2), 2 \cdot (1 - pt(t, n-2)) \,]$$ $P = 0.0003$ **< Same result as t-test of H_0: $\beta = 0$**

Prototype in R:

COMMANDS:
> summary(lm(Y~X))

t statistic & P values are identical to the t-test for H_0: $\beta = 0$

Call:
lm(formula = Y ~ X)

Residuals:

Min	1Q	Median	3Q	Max
-8.14000	-2.07619	-0.07619	3.31743	6.86000

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	21.5869	2.6465	8.157	5.4e-09 ***
X	0.6064	0.1483	4.090	0.000313 ***

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Residual standard error: 3.859 on 29 degrees of freedom
Multiple R-Squared: 0.3658, Adjusted R-squared: 0.3439
F-statistic: 16.73 on 1 and 29 DF, p-value: 0.0003134

Fisher's One sample z-Test for $\rho = \rho_0$:

This test evaluates specific values of ρ_0 using Fisher's z-transformation approach. See Rosner p. 499ff for this.

Assumptions:

- Normal distribution for all variables used to compute correlation coefficient r.

Hypotheses:

H_0: $\rho = \rho_0$ **< Correlation Coefficient ρ_0 value must be explicitly stated.**

H_1: $\rho <> 0$ **< Two sided test**

Fisher's z-transformation:

$$z := \frac{1}{2} \cdot \ln\left(\frac{1 + r}{1 - r}\right)$$

Distribution of z:

z is Normally distributed: $N(\mu,\sigma^2)$ with: $z_0 = \mu$ $\mu := \frac{1}{2} \cdot \ln\left(\frac{1 + \rho_0}{1 - \rho_0}\right)$ $\sigma^2 := \frac{1}{n - 3}$

λ is the Normalized distribution of z ~ N(0,1) where: $\lambda := (z - z_0) \cdot \sqrt{n - 3}$

Test Statistic:

$$\lambda := (z - z_0) \cdot \sqrt{n - 3}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_N\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_N\left(1 - \frac{\alpha}{2}\right)$

< Note use of N(0,1) here!

$C_1 := \text{qnorm}\left(\frac{\alpha}{2}, 0, 1\right)$ $C_2 := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$

Decision Rule:

IF $|\lambda| > C$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 $\Phi_N(\lambda)$,1-2 $\Phi_N(\lambda)$)

$P := \min\left[2 \cdot \text{pnorm}(\lambda, 0, 1), 2 \cdot (1 - \text{pnorm}(\lambda, 0, 1))\right]$

Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find the Lower Bound of the CI.

Confidence Interval for ρ:

$z_1 := z + C_1 \cdot \frac{1}{\sqrt{n - 3}}$ $z_2 := z + C_2 \cdot \frac{1}{\sqrt{n - 3}}$ **< CI in units of z (i.e., "transformed")**

$\rho_1 := \frac{e^{2 \cdot z_1} - 1}{e^{2 \cdot z_1} + 1}$ $\rho_2 := \frac{e^{2 \cdot z_2} - 1}{e^{2 \cdot z_2} + 1}$ **< CI in units of ρ**

Example:

From above: $r = 0.6048$ **And let's test:** $\rho_0 := 0.7$

Fisher's One sample z-Test for $\rho = \rho_0$:

Assumptions:

- Normal distribution for all variables used to compute correlation coefficient r.

Hypotheses:

H_0: $\rho = \rho_0 = 0.7$ **< Correlation Coefficient ρ_0 value must be explicitly stated.**

H_1: $\rho <> 0$ **< Two sided test**

Fisher's z-transformation:

$$z := \frac{1}{2} \cdot \ln\left(\frac{1 + r}{1 - r}\right) \qquad z = 0.7007$$

Distribution of z:

z is Normally distributed: $N(\mu,\sigma^2)$ with: $z_0 = \mu$ $\mu := \frac{1}{2} \cdot \ln\left(\frac{1 + \rho_0}{1 - \rho_0}\right)$ $\sigma^2 := \sqrt{\frac{1}{n - 3}}$

λ is the Normalized distribution of z ~N(0,1) where: $\lambda := (z - z_0) \cdot \sqrt{n - 3}$

Test Statistic:

$\mu = 0.8673$

$z_0 := \mu$ $\lambda := (z - z_0) \cdot \sqrt{n - 3}$ $\lambda = -0.8817$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{qnorm}\left(\frac{\alpha}{2}, 0, 1\right)$ $C_2 := \text{qnorm}\left(1 - \frac{\alpha}{2}, 0, 1\right)$ **< Note use of N(0,1) here!**

$C_1 = -1.96$ $C_2 = 1.96$

Decision Rule:

IF $|\lambda| > C$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$\lambda = -0.8817$ $C_1 = -1.96$ $C_2 = 1.96$

Probability Value:

$$P := \min\left[2 \cdot \text{pnorm}(\lambda, 0, 1), 2 \cdot (1 - \text{pnorm}(\lambda, 0, 1))\right] \qquad P = 0.378$$

Confidence Interval for ρ:

$z_1 := z + C_1 \cdot \frac{1}{\sqrt{n - 3}}$ $z_2 := z + C_2 \cdot \frac{1}{\sqrt{n - 3}}$ **< CI in units of z (i.e., "transformed")**

$z_1 = 0.3303$ $z_2 = 1.0711$

< CI in units of ρ

$\rho_1 := \frac{\exp(2 \cdot z_1) - 1}{\exp(2 \cdot z_1) + 1}$ $\rho_2 := \frac{\exp(2 \cdot z_2) - 1}{\exp(2 \cdot z_2) + 1}$

Here we use the exponential function exp() for number e= 2.7183 Since we already used symbol e to refer to the residual vector above....

$\rho_1 = 0.3188$ $\rho_2 = 0.7899$

Assignment for Week 11

Today we begin our final push toward the end of the semester looking at **Linear Regression** first and then **ANOVA**. In fact, the two are closely related under an encompassing rubric called "linear modeling" or "glm" (for the general linear model). At heart, all of these methods involving specifying a **statistical model** allowing observations of a ***dependent variable*** to be interpreted in light of observations for one or more ***independent variables*** plus a general ***hypothesis of uncontrolled or unexplained variability*** often called "error" or "residual". Many different models can be used. In "Simple" and "Multiple" Linear Regression, a single dependent variable Y is specified in terms of an **intercept coefficient** α plus one or more **regression** (or slope) **coefficients** β_i exactly associated with the independent variables X_i (with i = 1 in "simple" or more than one in "multiple" regression). The first step in Linear Regression is to "fit" the regression – in other words find the "best" line describing the relationship between independent and dependent variables. One way to do this is the **least squares method** which involves finding a line through the points that minimizes the squared distances between points on the line itself, with the observations Y for each X.

Once fitted, the line becomes the **regression prediction** Y_{hat} of where the Expected (or mean) values of each Y are to be found, and the distance between Y_{hat} and Y becomes the **residual unexplained variance**. Of course, the smaller the residual, the better the fit between Y and X_i. To measure this fit, variance is usually expressed in terms of ***Sums of Squares*** – the numerator in variance calculations. Here, residual unexplained variance becomes the **Total Sums of Squares** SS_T that is **partitioned** into **Regression Sums of Squares** SS_R and **Within (or Error) Sums of Squares** SS_E such that $SS_R + SS_E = SS_T$. With this partition of variance, one sets up a **standard ANOVA table** displaying the **Source** of the variance, **Sums of Squares** SS, **degrees of freedom** df, and **Mean Squares** MS. From a standard ANOVA table, several **inference procedures** may be followed to test hypotheses about the **linear model parameters** with the fitted data.

Our objective this week is to prototype regression fitting and the associated tests with real data. Pick a data set from one of your data sources, and perform the following:

1. Fit your data using a "Simple" Linear Regression model. Also recover and display your regression predictions and residuals. Draw a graph displaying your results. [see Biostatistics Worksheet 39].
2. Calculate the ANOVA table. [see Worksheet 39]
3. Perform a F-Test for $\beta = 0$ and interpret the results. [see Worksheet 40]
4. Perform a t-Test for $\beta = 0$, calculate the confidence interval for β, and interpret the results. [see Worksheet 41]
5. Calculate confidence intervals for the regression prediction and for confidence interval for new observations. [see Worksheet 41]
6. Calculate the coefficient of determination and coefficient of correlation. [see Worksheet 42]
7. Perform a t-Test for $\beta = \beta_0$ (a value you wish to test), and interpret results. [see Worksheet 42]
8. Perform a t-Test for $\rho = 0$ (no correlation), and interpret results. [see Worksheet 42]
9. Perform Fisher's z-Test for $\rho = \rho_0$ (you supply the test value), and interpret results. [see Worksheet 42]

Find the Least Squares Fit

Description

The least squares estimate of ***b*** in the model

$$y = X\,b + e$$

is found.

Usage

```
lsfit(x, y, wt = NULL, intercept = TRUE, tolerance = 1e-07,
      yname = NULL)
```

Arguments

`x`	a matrix whose rows correspond to cases and whose columns correspond to variables.
`y`	the responses, possibly a matrix if you want to fit multiple left hand sides.
`wt`	an optional vector of weights for performing weighted least squares.
`intercept`	whether or not an intercept term should be used.
`tolerance`	the tolerance to be used in the matrix decomposition.
`yname`	names to be used for the response variables.

Details

If weights are specified then a weighted least squares is performed with the weight given to the *j*th case specified by the *j*th entry in `wt`.

If any observation has a missing value in any field, that observation is removed before the analysis is carried out. This can be quite inefficient if there is a lot of missing data.

The implementation is via a modification of the LINPACK subroutines which allow for multiple left-hand sides.

Value

A list with the following named components:

`coef`	the least squares estimates of the coefficients in the model (***b*** as stated

above).

`residuals` residuals from the fit.

`intercept` indicates whether an intercept was fitted.

`qr` the QR decomposition of the design matrix.

References

Becker, R. A., Chambers, J. M. and Wilks, A. R. (1988) *The New S Language*. Wadsworth & Brooks/Cole.

See Also

`lm` which usually is preferable; `ls.print`, `ls.diag`.

Examples

```
##-- Using the same data as the lm(.) example:
lsD9 <- lsfit(x = unclass(gl(2,10)), y = weight)
ls.print(lsD9)
```

Fitting Linear Models

Description

`lm` is used to fit linear models. It can be used to carry out regression, single stratum analysis of variance and analysis of covariance (although aov may provide a more convenient interface for these).

Usage

```
lm(formula, data, subset, weights, na.action,
   method = "qr", model = TRUE, x = FALSE, y = FALSE, qr = TRUE,
   singular.ok = TRUE, contrasts = NULL, offset, ...)
```

Arguments

`formula`	a symbolic description of the model to be fit. The details of model specification are given below.
`data`	an optional data frame, list or environment (or object coercible by as.data.frame to a data frame) containing the variables in the model. If not found in `data`, the variables are taken from `environment(formula)`, typically the environment from which `lm` is called.
`subset`	an optional vector specifying a subset of observations to be used in the fitting process.
`weights`	an optional vector of weights to be used in the fitting process. Should be `NULL` or a numeric vector. If non-NULL, weighted least squares is used with weights `weights` (that is, minimizing `sum(w*e^2)`); otherwise ordinary least squares is used.
`na.action`	a function which indicates what should happen when the data contain `NA`s. The default is set by the `na.action` setting of options, and is na.fail if that is unset. The "factory-fresh" default is na.omit. Another possible value is `NULL`, no action. Value na.exclude can be useful.
`method`	the method to be used; for fitting, currently only `method = "qr"` is supported; `method = "model.frame"` returns the model frame (the same as with `model = TRUE`, see below).
`model, x, y, qr`	logicals. If `TRUE` the corresponding components of the fit (the model frame, the model matrix, the response, the QR decomposition) are returned.
`singular.ok`	logical. If `FALSE` (the default in S but not in **R**) a singular fit is an error.

`contrasts`	an optional list. See the `contrasts.arg` of model.matrix.default.
`offset`	this can be used to specify an *a priori* known component to be included in the linear predictor during fitting. This should be `NULL` or a numeric vector of length either one or equal to the number of cases. One or more offset terms can be included in the formula instead or as well, and if both are specified their sum is used. See model.offset.
`...`	additional arguments to be passed to the low level regression fitting functions (see below).

Details

Models for `lm` are specified symbolically. A typical model has the form `response ~ terms` where `response` is the (numeric) response vector and `terms` is a series of terms which specifies a linear predictor for `response`. A terms specification of the form `first + second` indicates all the terms in `first` together with all the terms in `second` with duplicates removed. A specification of the form `first:second` indicates the set of terms obtained by taking the interactions of all terms in `first` with all terms in `second`. The specification `first*second` indicates the *cross* of `first` and `second`. This is the same as `first + second + first:second`.

If the formula includes an offset, this is evaluated and subtracted from the response.

If `response` is a matrix a linear model is fitted separately by least-squares to each column of the matrix.

See model.matrix for some further details. The terms in the formula will be re-ordered so that main effects come first, followed by the interactions, all second-order, all third-order and so on: to avoid this pass a `terms` object as the formula (see aov and `demo(glm.vr)` for an example).

A formula has an implied intercept term. To remove this use either `y ~ x - 1` or `y ~ 0 + x`. See formula for more details of allowed formulae.

`lm` calls the lower level functions lm.fit, etc, see below, for the actual numerical computations. For programming only, you may consider doing likewise.

All of `weights`, `subset` and `offset` are evaluated in the same way as variables in `formula`, that is first in `data` and then in the environment of `formula`.

Value

`lm` returns an object of class `"lm"` or for multiple responses of class `c("mlm", "lm")`. The functions `summary` and anova are used to obtain and print a summary and analysis of variance table of the results. The generic accessor functions `coefficients`, `effects`,

`fitted.values` and `residuals` extract various useful features of the value returned by `lm`.
An object of class `"lm"` is a list containing at least the following components:

`coefficients`	a named vector of coefficients
`residuals`	the residuals, that is response minus fitted values.
`fitted.values`	the fitted mean values.
`rank`	the numeric rank of the fitted linear model.
`weights`	(only for weighted fits) the specified weights.
`df.residual`	the residual degrees of freedom.
`call`	the matched call.
`terms`	the terms object used.
`contrasts`	(only where relevant) the contrasts used.
`xlevels`	(only where relevant) a record of the levels of the factors used in fitting.
`offset`	the offset used (missing if none were used).
`y`	if requested, the response used.
`x`	if requested, the model matrix used.
`model`	if requested (the default), the model frame used.

In addition, non-null fits will have components `assign`, `effects` and (unless not requested) `qr` relating to the linear fit, for use by extractor functions such as `summary` and effects.

Using time series

Considerable care is needed when using `lm` with time series.

Unless `na.action = NULL`, the time series attributes are stripped from the variables before the regression is done. (This is necessary as omitting `NA`s would invalidate the time series attributes, and if `NA`s are omitted in the middle of the series the result would no longer be a regular time series.)

Even if the time series attributes are retained, they are not used to line up series, so that the time shift of a lagged or differenced regressor would be ignored. It is good practice to prepare a `data` argument by ts.intersect`(..., dframe = TRUE)`, then apply a suitable `na.action` to that data frame and call `lm` with `na.action = NULL` so that residuals and fitted values are time series.

Note

Offsets specified by `offset` will not be included in predictions by predict.lm, whereas those specified by an offset term in the formula will be.

Author(s)

The design was inspired by the S function of the same name described in Chambers (1992). The implementation of model formula by Ross Ihaka was based on Wilkinson & Rogers (1973).

References

Chambers, J. M. (1992) *Linear models.* Chapter 4 of *Statistical Models in S* eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

Wilkinson, G. N. and Rogers, C. E. (1973) Symbolic descriptions of factorial models for analysis of variance. *Applied Statistics*, **22**, 392–9.

See Also

summary.lm for summaries and anova.lm for the ANOVA table; aov for a different interface.

The generic functions coef, effects, residuals, fitted, vcov.

predict.lm (via predict) for prediction, including confidence and prediction intervals; confint for confidence intervals of *parameters*.

lm.influence for regression diagnostics, and glm for **generalized** linear models.

The underlying low level functions, lm.fit for plain, and lm.wfit for weighted regression fitting.

More `lm()` examples are available e.g., in anscombe, attitude, freeny, LifeCycleSavings, longley, stackloss, swiss.

Examples

```
## Annette Dobson (1990) "An Introduction to Generalized Linear
Models".
## Page 9: Plant Weight Data.
ctl <- c(4.17,5.58,5.18,6.11,4.50,4.61,5.17,4.53,5.33,5.14)
trt <- c(4.81,4.17,4.41,3.59,5.87,3.83,6.03,4.89,4.32,4.69)
group <- gl(2,10,20, labels=c("Ctl","Trt"))
weight <- c(ctl, trt)
anova(lm.D9 <- lm(weight ~ group))
summary(lm.D90 <- lm(weight ~ group - 1))# omitting intercept
summary(resid(lm.D9) - resid(lm.D90)) #- residuals almost identical
```

```
opar <- par(mfrow = c(2,2), oma = c(0, 0, 1.1, 0))
plot(lm.D9, las = 1)      # Residuals, Fitted, ...
par(opar)

## model frame :
stopifnot(identical(lm(weight ~ group, method = "model.frame"),
                    model.frame(lm.D9)))

### less simple examples in "See Also" above
```

Extract Model Fitted Values

Description

`fitted` is a generic function which extracts fitted values from objects returned by modeling functions. `fitted.values` is an alias for it.

All object classes which are returned by model fitting functions should provide a `fitted` method. (Note that the generic is `fitted` and not `fitted.values`.)

Methods can make use of napredict methods to compensate for the omission of missing values. The default and nls methods do.

Usage

```
fitted(object, ...)
fitted.values(object, ...)
```

Arguments

`object`	an object for which the extraction of model fitted values is meaningful.
`...`	other arguments.

Value

Fitted values extracted from the object `x`.

References

Chambers, J. M. and Hastie, T. J. (1992) *Statistical Models in S*. Wadsworth & Brooks/Cole.

See Also

coefficients, glm, lm, residuals.

Predict method for Linear Model Fits

Description

Predicted values based on linear model object.

Usage

```
## S3 method for class 'lm':
predict(object, newdata, se.fit = FALSE, scale = NULL, df = Inf,
        interval = c("none", "confidence", "prediction"),
        level = 0.95, type = c("response", "terms"),
        terms = NULL, na.action = na.pass, pred.var = res.var/weights,
        weights = 1, ...)
```

Arguments

`object`	Object of class inheriting from `"lm"`
`newdata`	An optional data frame in which to look for variables with which to predict. If omitted, the fitted values are used.
`se.fit`	A switch indicating if standard errors are required.
`scale`	Scale parameter for std.err. calculation
`df`	Degrees of freedom for scale
`interval`	Type of interval calculation.
`level`	Tolerance/confidence level
`type`	Type of prediction (response or model term).
`terms`	If `type="terms"`, which terms (default is all terms)
`na.action`	function determining what should be done with missing values in `newdata`. The default is to predict `NA`.
`pred.var`	the variance(s) for future observations to be assumed for prediction intervals. See Details.
`weights`	variance weights for prediction. This can be a numeric vector or a one-sided model formula. In the latter case, it is interpreted as an expression evaluated in `newdata`
`...`	further arguments passed to or from other methods.

Details

`predict.lm` produces predicted values, obtained by evaluating the regression function in the frame `newdata` (which defaults to `model.frame(object)`. If the logical `se.fit` is `TRUE`, standard errors of the predictions are calculated. If the numeric argument `scale` is set (with optional `df`), it is used as the residual standard deviation in the computation of the standard errors, otherwise this is extracted from the model fit. Setting `intervals` specifies computation of confidence or prediction (tolerance) intervals at the specified `level`, sometimes referred to as narrow vs. wide intervals.

If the fit is rank-deficient, some of the columns of the design matrix will have been dropped. Prediction from such a fit only makes sense if `newdata` is contained in the same subspace as the original data. That cannot be checked accurately, so a warning is issued.

If `newdata` is omitted the predictions are based on the data used for the fit. In that case how cases with missing values in the original fit is determined by the `na.action` argument of that fit. If `na.action = na.omit` omitted cases will not appear in the residuals, whereas if `na.action = na.exclude` they will appear (in predictions, standard errors or interval limits), with residual value `NA`. See also `napredict`.

The prediction intervals are for a single observation at each case in `newdata` (or by default, the data used for the fit) with error variance(s) `pred.var`. This can be a multiple of `res.var`, the estimated value of *sigma^2*: the default is to assume that future observations have the same error variance as those used for fitting. If `weights` is supplied, the inverse of this is used as a scale factor. For a weighted fit, if the prediction is for the original data frame, `weights` defaults to the weights used for the model fit, with a warning since it might not be the intended result. If the fit was weighted and newdata is given, the default is to assume constant prediction variance, with a warning.

Value

`predict.lm` produces a vector of predictions or a matrix of predictions and bounds with column names `fit`, `lwr`, and `upr` if `interval` is set. If `se.fit` is `TRUE`, a list with the following components is returned:

`fit`	vector or matrix as above
`se.fit`	standard error of predicted means
`residual.scale`	residual standard deviations
`df`	degrees of freedom for residual

Note

Variables are first looked for in `newdata` and then searched for in the usual way (which will include the environment of the formula used in the fit). A warning will be given if the variables found are not of the same length as those in `newdata` if it was supplied.

Offsets specified by `offset` in the fit by `lm` will not be included in predictions, whereas those specified by an offset term in the formula will be.

Notice that prediction variances and prediction intervals always refer to *future* observations, possibly corresponding to the same predictors as used for the fit. The variance of the *residuals* will be smaller.

Strictly speaking, the formula used for prediction limits assumes that the degrees of freedom for the fit are the same as those for the residual variance. This may not be the case if `res.var` is not obtained from the fit.

See Also

The model fitting function `lm`, `predict`, `SafePrediction`

Examples

```
## Predictions
x <- rnorm(15)
y <- x + rnorm(15)
predict(lm(y ~ x))
new <- data.frame(x = seq(-3, 3, 0.5))
predict(lm(y ~ x), new, se.fit = TRUE)
pred.w.plim <- predict(lm(y ~ x), new, interval="prediction")
pred.w.clim <- predict(lm(y ~ x), new, interval="confidence")
matplot(new$x,cbind(pred.w.clim, pred.w.plim[,-1]),
        lty=c(1,2,2,3,3), type="l", ylab="predicted y")

## Prediction intervals, special cases
##  The first three of these throw warnings
w <- 1 + x^2
fit <- lm(y ~ x)
wfit <- lm(y ~ x, weights = w)
predict(fit, interval = "prediction")
predict(wfit, interval = "prediction")
predict(wfit, new, interval = "prediction")
predict(wfit, new, interval = "prediction", weights = (new$x)^2)
predict(wfit, new, interval = "prediction", weights = ~x^2)
```

ANOVA for Linear Model Fits

Description

Compute an analysis of variance table for one or more linear model fits.

Usage

```
## S3 method for class 'lm':
anova(object, ...)

anova.lmlist(object, ..., scale = 0, test = "F")
```

Arguments

`object, ...`	objects of class `lm`, usually, a result of a call to `lm`.
`test`	a character string specifying the test statistic to be used. Can be one of `"F"`, `"Chisq"` or `"Cp"`, with partial matching allowed, or `NULL` for no test.
`scale`	numeric. An estimate of the noise variance *sigma^2*. If zero this will be estimated from the largest model considered.

Details

Specifying a single object gives a sequential analysis of variance table for that fit. That is, the reductions in the residual sum of squares as each term of the formula is added in turn are given in as the rows of a table, plus the residual sum of squares.

The table will contain F statistics (and P values) comparing the mean square for the row to the residual mean square.

If more than one object is specified, the table has a row for the residual degrees of freedom and sum of squares for each model. For all but the first model, the change in degrees of freedom and sum of squares is also given. (This only make statistical sense if the models are nested.) It is conventional to list the models from smallest to largest, but this is up to the user.

Optionally the table can include test statistics. Normally the F statistic is most appropriate, which compares the mean square for a row to the residual sum of squares for the largest model considered. If `scale` is specified chi-squared tests can be used. Mallows' *Cp* statistic is the residual sum of squares plus twice the estimate of *sigma^2* times the residual degrees of freedom.

Value

An object of class `"anova"` inheriting from class `"data.frame"`.

Warning

The comparison between two or more models will only be valid if they are fitted to the same dataset. This may be a problem if there are missing values and **R**'s default of `na.action = na.omit` is used, and `anova.lmlist` will detect this with an error.

Note

Versions of **R** prior to 1.2.0 based F tests on pairwise comparisons, and this behaviour can still be obtained by a direct call to `anovalist.lm`.

References

Chambers, J. M. (1992) *Linear models.* Chapter 4 of *Statistical Models in S* eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

See Also

The model fitting function `lm`, `anova`.

`drop1` for so-called 'type II' anova where each term is dropped one at a time respecting their hierarchy.

Examples

```
## sequential table
fit <- lm(sr ~ ., data = LifeCycleSavings)
anova(fit)

## same effect via separate models
fit0 <- lm(sr ~ 1, data = LifeCycleSavings)
fit1 <- update(fit0, . ~ . + pop15)
fit2 <- update(fit1, . ~ . + pop75)
fit3 <- update(fit2, . ~ . + dpi)
fit4 <- update(fit3, . ~ . + ddpi)
anova(fit0, fit1, fit2, fit3, fit4, test="F")

anova(fit4, fit2, fit0, test="F") # unconventional order
```

Summarizing Linear Model Fits

Description

`summary` method for class `"lm"`.

Usage

```
## S3 method for class 'lm':
summary(object, correlation = FALSE, symbolic.cor = FALSE, ...)

## S3 method for class 'summary.lm':
print(x, digits = max(3, getOption("digits") - 3),
      symbolic.cor = x$symbolic.cor,
      signif.stars = getOption("show.signif.stars"), ...)
```

Arguments

`object`	an object of class `"lm"`, usually, a result of a call to `lm`.
`x`	an object of class `"summary.lm"`, usually, a result of a call to `summary.lm`.
`correlation`	logical; if `TRUE`, the correlation matrix of the estimated parameters is returned and printed.
`digits`	the number of significant digits to use when printing.
`symbolic.cor`	logical. If `TRUE`, print the correlations in a symbolic form (see `symnum`) rather than as numbers.
`signif.stars`	logical. If `TRUE`, “significance stars” are printed for each coefficient.
`...`	further arguments passed to or from other methods.

Details

`print.summary.lm` tries to be smart about formatting the coefficients, standard errors, etc. and additionally gives “significance stars” if `signif.stars` is `TRUE`.

Correlations are printed to two decimal places (or symbolically): to see the actual correlations print `summary(object)$correlation` directly.

Value

The function `summary.lm` computes and returns a list of summary statistics of the fitted linear model given in `object`, using the components (list elements) `"call"` and `"terms"` from its argument, plus

`residuals`	the *weighted* residuals, the usual residuals rescaled by the square root of the weights specified in the call to `lm`.
`coefficients`	a *p x 4* matrix with columns for the estimated coefficient, its standard error, t-statistic and corresponding (two-sided) p-value. Aliased coefficients are omitted.
`aliased`	named logical vector showing if the original coefficients are aliased.
`sigma`	the square root of the estimated variance of the random error *sigma^2 = 1/(n-p) Sum(w[i] R[i]^2),* where *R[i]* is the *i*-th residual, `residuals[i]`.
`df`	degrees of freedom, a 3-vector *(p, n-p, p*)*, the last being the number of non-aliased coefficients.
`fstatistic`	(for models including non-intercept terms) a 3-vector with the value of the F-statistic with its numerator and denominator degrees of freedom.
`r.squared`	*R^2*, the "fraction of variance explained by the model", *R^2 = 1 - Sum(R[i]^2) / Sum((y[i]- y*)^2),* where *y** is the mean of *y[i]* if there is an intercept and zero otherwise.
`adj.r.squared`	the above *R^2* statistic "*adjusted*", penalizing for higher *p*.
`cov.unscaled`	a *p x p* matrix of (unscaled) covariances of the *coef[j], j=1, ..., p*.
`correlation`	the correlation matrix corresponding to the above `cov.unscaled`, if `correlation = TRUE` is specified.
`symbolic.cor`	(only if `correlation` is true.) The value of the argument `symbolic.cor`.
`na.action`	from `object`, if present there.

See Also

The model fitting function `lm`, `summary`.

Function `coef` will extract the matrix of coefficients with standard errors, t-statistics and p-values.

Examples

```
##-- Continuing the  lm(.) example:
coef(lm.D90)# the bare coefficients
sld90 <- summary(lm.D90 <- lm(weight ~ group -1))# omitting intercept
sld90
coef(sld90)# much more
```

ORIGIN ≡ 0

Multiple Regression

Multiple Regression is an extension of the technique of linear regression that describes the relationship between a single dependent variable (Y) and multiple independent (predictor) variables (X_1, X_2, X_3, ...). Typically, multiple regression involves specifying one, or sometimes several, linear models, constructing the multiple regression, and then testing hypotheses often involving several regression coefficients (β_1, β_2, β_3, ...) corresponding to each of the X variables.

Assumptions:

- Multiple Linear Regression depends on specifying in advance which variable is considered 'dependent' and which others 'independent'. This decision matters as changing roles for Y versus X's usually produces a different result.

- Y_1, Y_2, Y_3, ... , Y_n (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.

- k Vectors of Independent Variables:

- $X_{1,1}$, $X_{1,2}$, $X_{1,3}$, ... , $X_{1,n}$ (independent variable) with each value of $X_{1,i}$ matched to Y_i
- $X_{2,1}$, $X_{2,2}$, $X_{2,3}$, ... , $X_{2,n}$ (independent variable) with each value of $X_{2,i}$ matched to Y_i
...
- $X_{k,1}$, $X_{k,2}$, $X_{k,3}$, ... , $X_{k,n}$ (independent variable) with each value of $Xk_{,i}$ matched to Y_i

Model:

$$Y_i = \alpha + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \beta_3 X_{3,i} + ... + \beta_k X_{k,i} + \varepsilon_i \qquad \text{for i = 1 to n}$$

where: α is the y intercept of the regression line (translation).
β_i's are the regression coefficient (i.e., "slope") for each X_i of the regression line.
ε_i's are the residuals (i.e. "error") in prediction of Y given that it is a random variable with $N(\mu,\sigma^2)$

Note that this is one of many possible linear models involving $X_{,i}$'s that may be squared or higher order functions of an original variable (i.e., X^2, X^3, etc.) or cross products of two original variables (i.e., $X_{a,i}X_{b,i}$ etc.). This is the wonderful and extremely powerful world of Linear Modeling.

Least Squares Estimation of the Regression Line:

Calculations in Multiple Regression are extensive, and best visualized using matrix algebra where sums of squares and cross products are implicit in matrix manipulations:

X:	**becomes a (n X k+1) Matrix of values with a first column of 1's and each of k vectors above comprising a subsequent columns.**
X^{-1}	**is the Inverse Matrix of X, such that $X^{-1}X = I$ (the identity matrix).**
X^T	**is the transpose matrix of X where rows and columns are reversed.**
Y	**is the vector of Y_i's arrayed as a column of numbers.**
b	**is the vector of regression coefficients including α plus all β_i's arrayed as a single column of numbers.**
Y_{hat}	**is the vector of fitted values Y_i arrayed as a column of numbers.**
e	**is the vector of residuals e_i arrayed as a column of numbers.**

Estimated Regression Coefficients (b):

$$b := \left(X^T X\right)^{-1} \cdot \left(X^T Y\right)$$

< Note: all calculations here involve MathCad's matrix algebra functions!

Estimated values of Y (Y_{hat}):

$$Y_{hat} := X \cdot b$$

Residuals (e):

$$e := Y - Y_{hat}$$

Example: Rosner Table 11.9 p. 511.

K := READPRN("c:/2007BiostatsData/Rosner Table 11.9a.txt")

$Y := K^{\langle 2 \rangle}$ **< dependent variable is the 3rd column of K**

n := length(Y) n = 16

i := 0 .. n − 1 $L_i := 1$

$X := \text{augment}\left(L, K^{\langle 0 \rangle}, K^{\langle 1 \rangle}\right)$

^ independent variables are first two columns of K

Assumptions:

- vector Y is the dependent variable and a random sample ~ N(μ,σ²).
- matrix X are the independent variables matched to Y

Model:

$$Y_i = \alpha + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \varepsilon_i$$

$$K = \begin{pmatrix} 135 & 3 & 89 \\ 120 & 4 & 90 \\ 100 & 3 & 83 \\ 105 & 2 & 77 \\ 130 & 4 & 92 \\ 125 & 5 & 98 \\ 125 & 2 & 82 \\ 105 & 3 & 85 \\ 120 & 5 & 96 \\ 90 & 4 & 95 \\ 120 & 2 & 80 \\ 95 & 3 & 79 \\ 120 & 3 & 86 \\ 150 & 4 & 97 \\ 160 & 3 & 92 \\ 125 & 3 & 88 \end{pmatrix} \quad Y = \begin{pmatrix} 89 \\ 90 \\ 83 \\ 77 \\ 92 \\ 98 \\ 82 \\ 85 \\ 96 \\ 95 \\ 80 \\ 79 \\ 86 \\ 97 \\ 92 \\ 88 \end{pmatrix} \quad X = \begin{pmatrix} 1 & 135 & 3 \\ 1 & 120 & 4 \\ 1 & 100 & 3 \\ 1 & 105 & 2 \\ 1 & 130 & 4 \\ 1 & 125 & 5 \\ 1 & 125 & 2 \\ 1 & 105 & 3 \\ 1 & 120 & 5 \\ 1 & 90 & 4 \\ 1 & 120 & 2 \\ 1 & 95 & 3 \\ 1 & 120 & 3 \\ 1 & 150 & 4 \\ 1 & 160 & 3 \\ 1 & 125 & 3 \end{pmatrix}$$

Estimated Regression Coefficients (b):

$$b := \left(X^T X\right)^{-1} \cdot \left(X^T Y\right)$$

Estimated values of Y (Y_{hat}):

$$Y_{hat} := X \cdot b$$

Residuals (e):

$$e := Y - Y_{hat}$$

$$b = \begin{pmatrix} 53.450194 \\ 0.125583 \\ 5.887719 \end{pmatrix} \quad Y_{hat} = \begin{pmatrix} 88.0671 \\ 92.0711 \\ 83.6717 \\ 78.4119 \\ 93.3269 \\ 98.5867 \\ 80.9235 \\ 84.2996 \\ 97.9588 \\ 88.3036 \\ 80.2956 \\ 83.0438 \\ 86.1833 \\ 95.8386 \\ 91.2067 \\ 86.8113 \end{pmatrix} \quad e = \begin{pmatrix} 0.9329 \\ -2.0711 \\ -0.6717 \\ -1.4119 \\ -1.3269 \\ -0.5867 \\ 1.0765 \\ 0.7004 \\ -1.9588 \\ 6.6964 \\ -0.2956 \\ -4.0438 \\ -0.1833 \\ 1.1614 \\ 0.7933 \\ 1.1887 \end{pmatrix}$$

Values confirmed in Table 11.10 p. 512 >

Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/Rosner Table 11.9.txt")
> K
> Y=K$SBP
> X1=K$Birthwt
> X2=K$Age
> lm(Y~X1+X2)  < Note formula format for Linear Model...

Call:
lm(formula = Y ~ X1 + X2)

Coefficients:
(Intercept)          X1          X2
    53.4502      0.1256      5.8877

> plot.lm(lm(Y~X1+X2))          < Diagnostic plots for assessing Normality Assumption.
> summary(lm(Y~X1+X2))

Call:
lm(formula = Y ~ X1 + X2)

Residuals:
    Min      1Q  Median      3Q     Max
-4.0438 -1.3481 -0.2395  0.9688  6.6964

Coefficients:
            Estimate   Std. Error  t value  Pr(>|t|)
(Intercept) 53.45019    4.53189    11.794   2.57e-08 ***
X1           0.12558    0.03434    3.657    0.00290 **
X2           5.88772    0.68021    8.656    9.34e-07 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.479 on 13 degrees of freedom
Multiple R-Squared: 0.8809,     Adjusted R-squared: 0.8626
F-statistic: 48.08 on 2 and 13 DF,  p-value: 9.844e-07

> anova(lm(Y~X1+X2))

Analysis of Variance Table

Response: Y
           Df  Sum Sq  Mean Sq  F value   Pr(>F)
X1          1  130.54   130.54   21.238  0.0004901 ***
X2          1  460.50   460.50   74.923  9.342e-07 ***
Residuals  13   79.90     6.15
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
> predict(lm(Y~X1+X2),confidence=0.95,interval="prediction")

       fit      lwr      upr
1 88.06709 82.41169 93.72250
2 92.07106 86.45813 97.68400
3 83.67168 77.94349 89.39987
4 78.41187 72.49403 84.32972
5 93.32689 87.68236 98.97143
6 98.58670 92.53982 104.63358
7 80.92354 75.05303 86.79405
8 84.29959 78.65438 89.94481
9 97.95878 91.90561 104.01195
10 88.30356 82.21745 94.38968
11 80.29562 74.44843 86.14282
12 83.04376 77.21015 88.87738
13 86.18334 80.64367 91.72302
14 95.83856 89.84909 101.82803
15 91.20667 84.91020 97.50314
16 86.81126 81.25746 92.36506
Warning message:
Predictions on current data refer to _future_ responses
 in: predict.lm(lm(Y ~ X1 + X2), confidence = 0.95, interval = "prediction")
```

Prototype in SYSTAT:

SYSTAT Rectangular file C:\Documents and Settings\Wm Stein\Desktop\Biostatistics Spring 2007\Week 11\Data\Rosner Table 11.syd,
created Mon Apr 02, 2007 at 15:20:33, contains variables:

BLANKBIRTHWTAGESBP
<Bookmark(3)>

Dep Var: SBP N: 16 Multiple R: 0.93857 Squared multiple R: 0.88091

Adjusted squared multiple R: 0.86259 Standard error of estimate: 2.47917

Effect	Coefficient	Std Error	Std Coef	Tolerance	t	P(2 Tail)
CONSTANT	53.45019	4.53189	0.00000	.	11.79424	0.00000
BIRTHWT	0.12558	0.03434	0.35208	0.98859	3.65746	0.00290
AGE	5.88772	0.68021	0.83323	0.98859	8.65580	0.00000

Analysis of Variance

Source	Sum-of-Squares	df	Mean-Square	F-ratio	P
Regression	591.03564	2	295.51782	48.08063	0.00000
Residual	79.90186	13	6.14630		

***** WARNING *****
Case 10 is an outlier (Studentized Residual = 6.75638)

Durbin-Watson D Statistic 2.214
First Order Autocorrelation -0.121

ORIGIN ≡ 0

Inference in Multiple Regression

A variety of tests may be employed testing regression coefficients in Multiple Regression in ways that are analogous to those in "Simple" Linear Regression.

Assumptions:

- $Y_1, Y_2, Y_3, \ldots, Y_n$ (dependent variable) is a random sample ~ $N(\mu,\sigma^2)$.

- X Matrix of One and column vectors of Independent Variables:

- $X_{1,1}, X_{1,2}, X_{1,3}, \ldots, X_{1,n}$ (independent variable) with each value of $X_{1,i}$ matched to Y_i
- $X_{2,1}, X_{2,2}, X_{2,3}, \ldots, X_{2,n}$ (independent variable) with each value of $X_{2,i}$ matched to Y_i
...
- $X_{k,1}, X_{k,2}, X_{k,3}, \ldots, X_{k,n}$ (independent variable) with each value of $Xk_{,i}$ matched to Y_i

Model:

$$Y_i = \alpha + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \beta_3 X_{3,i} + \ldots + \beta_k X_{k,i} + \varepsilon_i \qquad \text{for } i = 1 \text{ to } n$$

where: α is the y intercept of the regression line (translation).
β_i's are the regression coefficient (i.e., "slope") for each X_i of the regression line.
ε_i's are the residuals (i.e. "error") in prediction of Y given that it is a random variable with $N(\mu,\sigma^2)$

Note that this is one of many possible linear models involving $X_{,i}$'s that may be squared or higher order functions of an original variable (i.e., X^2, X^3, etc.) or cross products of two original variables (i.e., $X_{a,i}X_{b,i}$ etc.). This is the wonderful and extremely powerful world of Linear Modeling.

Least Squares Estimation of the Regression Line:

Estimated Regression Coefficients (b):

$$b := \left(X^T X\right)^{-1} \cdot \left(X^T Y\right)$$

Note: all calculations here involve MathCad's matrix algebra functions!

Estimated values of Y (Y_{hat}):

$$Y_{hat} := X \cdot b$$

Residuals (e):

$$e := Y - Y_{hat}$$

$$n := \text{length}(Y) \qquad k := \text{cols}(X) - 1$$

< calculations needed for size of problem for ANOVA table

ANOVA Table for Multiple Regression:

Sums of Squares:	Degrees of Freedom:	Mean Squares:	
$SS_R := \sum \left(Y_{hat} - \text{mean}(Y)\right)^2$	$df_R := k$	$\frac{SS_R}{df_R}$	**< Regression**
$SS_E := \left(Y - Y_{hat}\right)^2$	$df_E := n - k - 1$	$\frac{SS_E}{df_E}$	**< Residual**
$SS_T := \sum (Y - \text{mean}(Y))^2$	$df_T := n - 1$	$\frac{SS_T}{df_T}$	**< Total**

F-Test (ANOVA) for H_0: all β's = 0 versus H_1: not all β's are 0:

Hypotheses:

H_0: $\beta_1 = \beta_2 = \beta_3 = ... = \beta_k = 0$ **< Note specificity of test here!**

H_1:At least one β_i is NOT 0 **< Two sided test**

Test Statistic:

$F := \frac{MS_R}{MS_E}$ **< F is the ratio of sample variances**

Sampling Distribution:

If Assumptions hold and H_0 is true, then $F \sim F_{(k)/(n-k-1)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1-\alpha)$ $CV := qF(1-\alpha, k, n-k-1)$ **note: df = k, (n-k-1)**

Decision Rule:

IF |F| > CV, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = 1- Φ_F(F) $P := 1 - pF(F, k, n-k-1)$

Example: Rosner Table 11.9 p. 511.

$K := READPRN("c:/2007BiostatsData/Rosner Table 11.9a.txt")$

$Y := K^{\langle 2 \rangle}$ **< dependent variable is the 3rd column of K**

$n := length(Y)$ $n = 16$

$i := 0..n-1$ $L_i := 1$

$X := augment(L, K^{\langle 0 \rangle}, K^{\langle 1 \rangle})$

^ independent variables are first two columns of K

$$K = \begin{pmatrix} 135 & 3 & 89 \\ 120 & 4 & 90 \\ 100 & 3 & 83 \\ 105 & 2 & 77 \\ 130 & 4 & 92 \\ 125 & 5 & 98 \\ 125 & 2 & 82 \\ 105 & 3 & 85 \\ 120 & 5 & 96 \\ 90 & 4 & 95 \\ 120 & 2 & 80 \\ 95 & 3 & 79 \\ 120 & 3 & 86 \\ 150 & 4 & 97 \\ 160 & 3 & 92 \\ 125 & 3 & 88 \end{pmatrix} \quad Y = \begin{pmatrix} 89 \\ 90 \\ 83 \\ 77 \\ 92 \\ 98 \\ 82 \\ 85 \\ 96 \\ 95 \\ 80 \\ 79 \\ 86 \\ 97 \\ 92 \\ 88 \end{pmatrix} \quad X = \begin{pmatrix} 1 & 135 & 3 \\ 1 & 120 & 4 \\ 1 & 100 & 3 \\ 1 & 105 & 2 \\ 1 & 130 & 4 \\ 1 & 125 & 5 \\ 1 & 125 & 2 \\ 1 & 105 & 3 \\ 1 & 120 & 5 \\ 1 & 90 & 4 \\ 1 & 120 & 2 \\ 1 & 95 & 3 \\ 1 & 120 & 3 \\ 1 & 150 & 4 \\ 1 & 160 & 3 \\ 1 & 125 & 3 \end{pmatrix}$$

Assumptions:

- **vector Y is the dependent variable and a random sample ~ $N(\mu,\sigma^2)$.**
- **matrix X are the independent variables matched to Y**

Model:

$$Y_i = \alpha + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \varepsilon_i$$

Estimated Regression Coefficients (b):

$b := (X^T X)^{-1} \cdot (X^T Y)$

Estimated values of Y (Y_{hat}):

$Y_{hat} := X \cdot b$

Residuals (e):

$e := Y - Y_{hat}$

$$b = \begin{pmatrix} 53.450194 \\ 0.125583 \\ 5.887719 \end{pmatrix} \quad Y_{hat} = \begin{pmatrix} 88.0671 \\ 92.0711 \\ 83.6717 \\ 78.4119 \\ 93.3269 \\ 98.5867 \\ 80.9235 \\ 84.2996 \\ 97.9588 \\ 88.3036 \\ 80.2956 \\ 83.0438 \\ 86.1833 \\ 95.8386 \\ 91.2067 \\ 86.8113 \end{pmatrix} \quad e = \begin{pmatrix} 0.9329 \\ -2.0711 \\ -0.6717 \\ -1.4119 \\ -1.3269 \\ -0.5867 \\ 1.0765 \\ 0.7004 \\ -1.9588 \\ 6.6964 \\ -0.2956 \\ -4.0438 \\ -0.1833 \\ 1.1614 \\ 0.7933 \\ 1.1887 \end{pmatrix}$$

$n := \text{length}(Y)$ $k := \text{cols}(X) - 1$

ANOVA Table for Multiple Regression:

Sums of Squares:		Degrees of Freedom:		Mean Squares:	
$SS_R := \sum (Y_{hat} - \text{mean}(Y))^2$	$SS_R = 591.0356$	$df_R := k$	$df_R = 2$	$MS_R := \frac{SS_R}{df_R}$	$MS_R = 295.5178$
$SS_E := \sum (Y - Y_{hat})^2$	$SS_E = 79.9019$	$df_E := n - k - 1$	$df_E = 13$	$MS_E := \frac{SS_E}{df_E}$	$MS_E = 6.1463$
$SS_T := \sum (Y - \text{mean}(Y))^2$	$SS_T = 670.9375$	$df_T := n - 1$	$df_T = 15$	$MS_T := \frac{SS_T}{df_T}$	$MS_T = 44.7292$

Test Statistic:

^ values confirmed Table 11.10 p. 512

$F := \frac{MS_R}{MS_E}$ **< F is the ratio of sample variances** $F = 48.0806$ **< confirmed p. 516**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF(1 - \alpha, k, n - k - 1)$ $CV = 3.8056$ **note: df = k, (n-k-1)**

Decision Rule:

IF |F| > CV, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F = 48.0806$ $CV = 3.8056$

Probability Value:

$P := 1 - pF(F, k, n - k - 1)$ $P = 9.8443 \times 10^{-7}$ **< confirmed p. 516**

Prototype in R:

COMMANDS:

```
> K=read.table("c:/2007BiostatsData/Rosner Table 11.9.txt")
> K
> Y=K$SBP
> X1=K$Birthwt
> X2=K$Age
> anova(lm(Y~X1+X2))
```

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X1	1	130.54	130.54	21.238	0.0004901 ***
X2	1	460.50	460.50	74.923	9.342e-07 ***
Residuals	13	79.90	6.15		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

^ Note that in order to obtain appropriate value for MS_R, one must first sum the *partial* Sum of Squares for X_1 & X_2 and also sum the degrees of freedom for X_1 & X_2. The MS_R can be calculated by hand, along with a new F value that is one half the sum for X_1 and X_2 reported here.

Prototype in SYSTAT:

SYSTAT Rectangular file C:\Documents and Settings\Wm Stein\Desktop\Biostatistics Spring 2007\Week 11\Data\Rosner Table 11.syd,
created Mon Apr 02, 2007 at 15:20:33, contains variables:

Dep Var: SBP N: 16 Multiple R: 0.93857 Squared multiple R: 0.88091

Adjusted squared multiple R: 0.86259 Standard error of estimate: 2.47917

Effect	Coefficient	Std Error	Std Coef	Tolerance	t	P(2 Tail)
CONSTANT	53.45019	4.53189	0.00000	.	11.79424	0.00000
BIRTHWT	0.12558	0.03434	0.35208	0.98859	3.65746	0.00290
AGE	5.88772	0.68021	0.83323	0.98859	8.65580	0.00000

Analysis of Variance

Source	Sum-of-Squares	df	Mean-Square	F-ratio	P
Regression	591.03564	2	295.51782	48.08063	0.00000
Residual	79.90186	13	6.14630		

***** WARNING *****
Case 10 is an outlier (Studentized Residual = 6.75638)

Durbin-Watson D Statistic 2.214
First Order Autocorrelation -0.121

^ SYSTAT reports things our way in its ANOVA table

t-Test for H_0: $\beta_j = 0$ and all other β_i's <> 0:

Note: This t-test approach is equivalent to the Partial F-Test approach (Rosner p. 519) as far as inference on coefficients for each independent variable goes. However partial F-test approaches (also called "maximum liklihood" or "full and reduced model" methods) are generally more useful and form the core of so-called "General Linear Modeling" strategies. Most statistical packagaes including R & SYSTAT routinely report partial F in addition to, or instead of, t-test results.

Hypotheses:

H_0: $\beta_j = 0$ and all other β_i's <> 0 **< *one only* of the regression parameter is zero**

H_0: β_j <> 0 and all other β_i's <> 0 **< Two sided test**

Calculating Standard Errors for regression parameters β:

$i := 1 .. k$

$L_{xx_{i-1}} := \sum \left(X^{\langle i \rangle} - \text{mean}\left(X^{\langle i \rangle}\right)\right)^2$ **< corrected Sums of Squares for each independent variable in matrix X**

$SE_B := \sqrt{\frac{MS_E}{L_{xx}}}$ **< Standard Error for each b. See Rosner p. 483.**

Test Statistic:

$$t_{i-1} := \frac{b_i}{SE_{B_{i-1}}}$$

Sampling Distribution:

If Assumptions hold and H_0 is true, then $F \sim F_{(k)/(n-k-1)}$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n - k - 1\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k - 1\right)$ **Note degrees of freedom = (n-k-1)**

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t) for each

$P_i := \min\left[\left[2 \cdot pt\left[t_i, (n - k - 1)\right]\right], \left[2 \cdot \left[1 - pt\left[t_i, (n - k - 1)\right]\right]\right]\right]$

< Note that C_1 and C_2 are explicitly evaluated above so C_1 is already negative in value. So it is added to X_{bar} here to find the Lower Bound of the CI.

Example: (same data as above)

t-Test for H_0: $\beta_j = 0$ and all other β_i's <> 0:

Hypotheses:

H_0: $\beta_j = 0$ and all other β_i's <> 0 < *one only* of the regression parameter is not zero

H_0: β_j <> 0 and all other β_i's <> 0 < Two sided test

Calculating Standard Errors for regression parameters β:

$i := 1 .. k \qquad k = 2$

$L_{xx_{i-1}} := \sum \left(X^{\langle i \rangle} - \text{mean}\left(X^{\langle i \rangle}\right)\right)^2 \qquad L_{xx} = \begin{pmatrix} 5273.4375 \\ 13.4375 \end{pmatrix}$

$SE_B := \sqrt{\frac{MS_E}{L_{xx}}} \qquad SE_B = \begin{pmatrix} 0.0341 \\ 0.6763 \end{pmatrix}$ < close but not quite the same as Table 11.10

Test Statistic:

$MS_E = 6.1463$

$t_{i-1} := \frac{b_i}{SE_{B_{i-1}}} \qquad t = \begin{pmatrix} 3.6785 \\ 8.7056 \end{pmatrix}$ ^ this is the same

Critical Value of the Test:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$C_1 := qt\left(\frac{\alpha}{2}, n - k - 1\right) \qquad C_2 := qt\left(1 - \frac{\alpha}{2}, n - k - 1\right)$ < Note degrees of freedom = (n-k-1)

$C_1 = -2.1604 \qquad C_2 = 2.1604$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = \begin{pmatrix} 3.6785 \\ 8.7056 \end{pmatrix}$ < for β_1 < for β_2 $\qquad C_1 = -2.1604 \qquad C_2 = 2.1604$

Probability Value:

$i := 0 .. k - 1$

$P_i := \min\left[\left[2 \cdot pt\left[t_i, (n - k - 1)\right]\right], \left[2 \cdot \left[1 - pt\left[t_i, (n - k - 1)\right]\right]\right]\right] \qquad P = \begin{pmatrix} 0.0028 \\ 8.7597 \times 10^{-7} \end{pmatrix}$ < for β_1 < for β_2

> summary(lm(Y~X1+X2))

Call:
lm(formula = Y ~ X1 + X2)

Residuals:

Min	1Q	Median	3Q	Max
-4.0438	-1.3481	-0.2395	0.9688	6.6964

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)	
(Intercept)	53.45019	4.53189	11.794	2.57e-08	***
X1	0.12558	0.03434	3.657	0.00290	**
X2	5.88772	0.68021	8.656	9.34e-07	***

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Residual standard error: 2.479 on 13 degrees of freedom
Multiple R-Squared: 0.8809, Adjusted R-squared: 0.8626
F-statistic: 48.08 on 2 and 13 DF, p-value: 9.844e-07

t & P values approximately > match... rounding?

ORIGIN ≡ 0

Interpreting Regression Results from Statistics Packages

"Industrial strength" Statistical packages, such as those provided by R, SYSTAT, SPSS, Minitab or SAS are clearly the way to go for routine analysis of these stastical problems. Each provides slightly different output that is characteristically dense with information. The packages also provide multiple diagnostic tools for determining the appropriateness of Linear Regression to different datasets. Provided in this sheet is a brief summary of terms useful for interpreting results based on things we have seen in previous Worksheets.

SYSTAT Output of Multiple Linear Regression:

**SYSTAT Rectangular file C:\Documents and Settings\Wm Stein\Desktop\Biostatistics Spring 2007\Week 11\Data\Rosner Table 11.syd,
created Mon Apr 02, 2007 at 15:20:33, contains variables:**

BLANKBIRTHWTAGESBP

Dep Var: SBP N: 16 Multiple R: 0.93857 Squared multiple R: 0.88091

Adjusted squared multiple R: 0.86259 Standard error of estimate: 2.47917

Effect	Coefficient	Std Error	Std Coef	Tolerance	t	P(2 Tail)
CONSTANT	53.45019	4.53189	0.00000	.	11.79424	0.00000
BIRTHWT	0.12558	0.03434	0.35208	0.98859	3.65746	0.00290
AGE	5.88772	0.68021	0.83323	0.98859	8.65580	0.00000

Analysis of Variance

Source	Sum-of-Squares	df	Mean-Square	F-ratio	P
Regression	591.03564	2	295.51782	48.08063	0.00000
Residual	79.90186	13	6.14630		

***** WARNING *****
Case 10 is an outlier (Studentized Residual = 6.75638)

Durbin-Watson D Statistic 2.214
First Order Autocorrelation -0.121

Dependent Variable:

SBP is the name I gave to the dependent variable Y in the SYSTAT data table.

N:

Number of matched Y with X's in the study.

Multiple R & Squared multiple R:

Squared multiple R is the Coefficient of Determination:

$$\frac{SS_R}{SS_T}$$ **< from the ANOVA table**

Multiple R is the Coefficient of Correlation:

$\sqrt{\frac{SS_R}{SS_T}}$ **< square-root of the Coefficient of Determination**

Effect:

These are the names employed for the Independent portion ofthe the Regression Equation:
$Y_i = \alpha + \beta_1 X_{1,i} + \beta_2 X_{2,i} + \beta_3 X_{3,i} + \ldots + \beta_k X_{k,i} + \varepsilon_i$

"Constant" is the variable name for coefficient α - it involves only translation in dependent variable Y
"BIRTHWT" are the variable names I gave to the two independent variables X
"AGE" in this study

Coefficients:

These are estimates of the regression coefficients α & β_i

Standard Errors:

These are standard errors of the Regression coefficients:

$SE_{b_i} := \sqrt{\frac{MS_E}{Lxx_i}}$ **for coefficients estimates of β_i** **< See Rosner p. 483**

$SE_a := \sqrt{MS_E \cdot \left(\frac{1}{n} + \frac{X_{bar}^2}{L_{xx}}\right)}$ **for coefficient estimate of α**

Standardized Coefficients:

These are Standardized Regression Coefficients: **< See Rosner p. 513**

$b_{s_i} := b_i \cdot \left(\frac{s_{x_i}}{s_y}\right)$ **coefficients standardized by multiplying the ratio of standard deviations for each estimate of α & β_i**

Standardized coefficients are useful because relative magnitude reported are all in the same units of standard deviation, whereas the "raw" Regression coefficients also reflect the magnitude of the scale for each X variable, and these may be greatly different.

t & P(2 Tail):

These are the t-statistics and Probability values calculated in the t-test:

H_0: $\beta_j = 0$ and all other β_i's <> 0
H_0: β_j <> 0 and all other β_i's <> 0

< See Worksheet 44

Note that a test for α has not been given...

Analysis of Variance:

< See Worksheets 40 & 44

This is the standard ANOVA table for the multiple regression

In Source:

Given here are standard names for portions of the chart:

"Regression" is a row for reporting SS_R, df_R & MS_R Sometimes, as in R output more than one "partial regression" row needs to be summed for all "Regression" values.

ORIGIN ≡ 0

Regression and General Linear Models:

Multiple Linear Regression is not restricted to cardinal data, but may be readily adapted to other forms of data. In this guise - the so called "General Linear Model" or "GLM", is a very wide-ranging method that can be shown to unify much of standard statistics including t-tests, $\chi 2$, ANOVA, and many non-parametric statistical techniques. Shown here is one standard extension to independent variables in data classes through the judicious use of "dummy coding". Because all variables can be "binned" into data classes (as in histograms), this approach has broad application.

Example: Cats data previously analyzed in Biostatistics Worksheet 27:

cats := READPRN("c:/2007BiostatsData/dcats.txt")

cats =

	0	1	2	3
0	1	0	2	7
1	2	0	2	7.4
2	3	0	2	9.5
3	4	0	2.1	7.2
4	5	0	2.1	7.3
5	6	0	2.1	7.6
6	7	0	2.1	8.1
7	8	0	2.1	8.2
8	9	0	2.1	8.3
9	10	0	2.1	8.5
10	11	0	2.1	8.7
11	12	0	2.1	9.8
12	13	0	2.2	7.1
13	14	0	2.2	8.7
14	15	0	2.2	9.1
15	16	0	2.2	9.7

Calculations in R:

t-test approach: two populations equal variance

COMMANDS:

```
> dcats=read.table("c:/2007BiostatsData/dcats.txt")
> dcats
> X1=dcats$Bwt[dcats$Sex=="0"]
> X2=dcats$Bwt[dcats$Sex=="1"]
> t.test(X1,X2,alternative="two.sided",var.equal=T)
```

Two Sample t-test

data: X1 and X2
t = -7.3307, df = 142, p-value = 1.590e-11
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
-0.6861584 -0.3946927
sample estimates:
mean of x mean of y
2.359574 2.900000 **< mean difference:** 2.9 – 2.359574= 0.5404

Regression approach: with dummy variable for Sex: F=0 M=1

COMMANDS:

```
> Y=dcats$Bwt
> X=dcats$Sex
> summary(lm(Y~X))
```

Call:
lm(formula = Y ~ X)

Residuals:

Min	1Q	Median	3Q	Max
-0.90000	-0.25957	-0.05957	0.30000	1.00000

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	2.35957	0.06051	38.997	< 2e-16 ***
X	0.54043	0.07372	7.331	1.59e-11 ***

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Residual standard error: 0.4148 on 142 degrees of freedom
Multiple R-Squared: 0.2745, Adjusted R-squared: 0.2694
F-statistic: 53.74 on 1 and 142 DF, p-value: 1.590e-11

t statistic& P match >

difference in means calculated above match estimate for coefficient $\beta 1$ here.

2.9 – 2.359574 = 0.5404

Assignment for Week 12

Let's broaden our attack on Regression and ANOVA this week using both R an SPSS.

Using a dataset of choice, including several I have placed in the Data Section this week, do the following:

Multiple Regression – As discussed in lecture, multiple regression involves extension of the regression technique using least squares to the commonly encountered situations with more than one independent variable.

1. Fit your data using a Multiple Linear Regression model with at least two independent variables. Also recover and display your regression predictions and residuals. Draw graphs displaying your results comparing the dependent variable with each multiple independent variables. [see Biostatistics Worksheet 39 for making graphs & 43 for Multiple Regression].
2. Calculate the ANOVA table. [see Worksheet 44 & 45]
3. Perform a F-Test for H_0: all β's = 0 versus H_1: not all β's are 0 and interpret the results. [see Worksheet 44 & 45]
4. Perform a t-Test for H_0: $\beta_j = 0$ and all other β_i's <> 0, and interpret the results. [see Worksheet 44 & 45]
5. Now try the same thing again using SPSS. The important thing to think about here is how to structure your data in an appropriate way in order to use SPSS's Graphical User's Interface (GUI). My sense is that SPSS is generally less flexible than R, but once you figure out how it works, probably more efficient.

One-Way ANOVA for fixed effects – Now find data that will work with this approach. Some datasets already have categorical variables as text fields whereas others list categories as dummy variables. See Biostatistics Worksheet 46 and dataset dcats.txt for an example of this.

6. Run the dcats.dta set both as a regression and as an ANOVA and compare your results. Are the number n_i in each sample of the ANOVA equal? If not, drop some observations from dcat.txt to make them equal and re-run Regression and ANOVA. Do these results differ? If so, how? What parts are comparable?
7. Using data you chose, perform the F-Test for All $\alpha_i = 0$ in One-Way ANOVA with Fixed Effects Model, and compare the results in both R and SPSS.
8. Assuming you can with the data (if not, find another dataset that makes this test meaningful), now perform the t-Test for $\alpha_i = \alpha_j$ versus H_1: $\alpha_i <> \alpha_j$ for specific i's & j's you choose (try more than one). Interpret your results.

End of Term Assignment for Graduate Students

The Department of Biological Sciences mandates that I require something 'extra' from graduate students in courses simultaneously taught at both graduate and undergraduate levels. Whereas this course primarily consists of introducing the theoretical framework of statistics with quite a bit of practice with specific procedures, the objectives of all students are basically the same. However, graduate students have chosen a field of study with its own peculiar problems, methodology, literature, and style of 'scientific' reportage. No doubt, all fields utilize some sort of statistical appraisal, and *it is important to become familiar with techniques actually used by your colleagues-to-be*. Therefore, we have an excellent rationale for this **end-term requirement**.

Here's what I want you to hand in by the last day of term:

The report need not be all that long or involved. As far as I'm concerned the fewer words the better. What I want is a brief summary of the literature in your chosen field of study. In fact, I need this information as I endeavor to better fit this course in the future to its intended target audience. Please provide me with the following:

1. I need a **basic summary** of the **research objectives** of your field of study. What, and how do leading researchers report their findings. What kinds of study are typically conducted. Who is the intended audience?
2. What journals or other forms of publications represent the most prestigious outlets for research in your field? What, typically, is reported in these journals?
3. I need an **annotated bibliography** of 10-15 recent papers reporting statistical results in your field. Please include a ***complete and correctly formatted reference list***. Following each reference, provide me a sentence or two describing the statistical tests utilized. Please be as specific as possible. Some tests probably have been covered in this course, whereas others may not have been. Either way, try to be as specific as you can about: **statistical hypothesis tested**, **assumptions** of the test (if stated by the authors or that you would expect to be in force), **statistical model utilized**, and each paper's **conclusion**.
4. Finally, I need a list of statistical procedures/tests that in your opinion, would constitute an **important set** for your field of study.

Factors

Description

The function `factor` is used to encode a vector as a factor (the terms ‘category’ and ‘enumerated type’ are also used for factors). If `ordered` is `TRUE`, the factor levels are assumed to be ordered. For compatibility with S there is also a function `ordered`.

`is.factor`, `is.ordered`, `as.factor` and `as.ordered` are the membership and coercion functions for these classes.

Usage

```
factor(x = character(), levels = sort(unique.default(x), na.last =
TRUE),
       labels = levels, exclude = NA, ordered = is.ordered(x))
ordered(x, ...)

is.factor(x)
is.ordered(x)

as.factor(x)
as.ordered(x)
```

Arguments

`x`	a vector of data, usually taking a small number of distinct values.
`levels`	an optional vector of the values that `x` might have taken. The default is the set of values taken by `x`, sorted into increasing order.
`labels`	*either* an optional vector of labels for the levels (in the same order as `levels` after removing those in `exclude`), *or* a character string of length 1.
`exclude`	a vector of values to be excluded when forming the set of levels. This should be of the same type as `x`, and will be coerced if necessary.
`ordered`	logical flag to determine if the levels should be regarded as ordered (in the order given).
`...`	(in `ordered(.)`): any of the above, apart from `ordered` itself.

Details

The type of the vector `x` is not restricted.

Ordered factors differ from factors only in their class, but methods and the model-fitting functions treat the two classes quite differently.

The encoding of the vector happens as follows. First all the values in `exclude` are removed from `levels`. If `x[i]` equals `levels[j]`, then the `i`-th element of the result is `j`. If no match is found for `x[i]` in `levels`, then the `i`-th element of the result is set to `NA`.

Normally the 'levels' used as an attribute of the result are the reduced set of levels after removing those in `exclude`, but this can be altered by supplying `labels`. This should either be a set of new labels for the levels, or a character string, in which case the levels are that character string with a sequence number appended.

`factor(x, exclude=NULL)` applied to a factor is a no-operation unless there are unused levels: in that case, a factor with the reduced level set is returned. If `exclude` is used it should also be a factor with the same level set as `x` or a set of codes for the levels to be excluded.

The codes of a factor may contain `NA`. For a numeric `x`, set `exclude=NULL` to make `NA` an extra level (`"NA"`), by default the last level.

If `"NA"` is a level, the way to set a code to be missing is to use `is.na` on the left-hand-side of an assignment. Under those circumstances missing values are printed as `<NA>`.

`is.factor` is generic: you can write methods to handle specific classes of objects, see InternalMethods.

Value

`factor` returns an object of class `"factor"` which has a set of integer codes the length of `x` with a `"levels"` attribute of mode `character`. If `ordered` is true (or `ordered` is used) the result has class `c("ordered", "factor")`.
Applying `factor` to an ordered or unordered factor returns a factor (of the same type) with just the levels which occur: see also `[.factor` for a more transparent way to achieve this.
`is.factor` returns `TRUE` or `FALSE` depending on whether its argument is of type factor or not. Correspondingly, `is.ordered` returns `TRUE` when its argument is ordered and `FALSE` otherwise.
`as.factor` coerces its argument to a factor. It is an abbreviated form of `factor`.
`as.ordered(x)` returns `x` if this is ordered, and `ordered(x)` otherwise.

Warning

The interpretation of a factor depends on both the codes and the `"levels"` attribute. Be careful only to compare factors with the same set of levels (in the same order). In particular, `as.numeric` applied to a factor is meaningless, and may happen by implicit

coercion. To “revert” a factor `f` to its original numeric values, `as.numeric(levels(f))[f]` is recommended and slightly more efficient than `as.numeric(as.character(f))`.

The levels of a factor are by default sorted, but the sort order may well depend on the locale at the time of creation, and should not be assumed to be ASCII.

Comparison operators and group generic methods

There are `"factor"` and `"ordered"` methods for the group generic `Ops`, which provide methods for the Comparison operators. (The rest of the group and the `Math` and `Summary` groups generate an error as they are not meaningful for factors.)

Only == and `!=` can be used for factors: a factor can only be compared to another factor with an identical set of levels (not necessarily in the same ordering) or to a character vector. Ordered factors are compared in the same way, but the general dispatch mechanism precludes comparing ordered and unordered factors.

All the comparison operators are available for ordered factors. Sorting is done by the levels of the operands: if both operands are ordered factors they must have the same level set.

Note

Storing character data as a factor is more efficient storage if there is even a small proportion of repeats. On a 32-bit machine storing a string of *n* bytes takes *28 + 8*ceiling((n+1)/8)* bytes whereas storing a factor code takes 4 bytes. (On a 64-bit machine 28 is replaced by 56 or more.) Only if they were computed from the same values (or in some cases read from a file: see `scan`) will identical strings share storage.

References

Chambers, J. M. and Hastie, T. J. (1992) *Statistical Models in S*. Wadsworth & Brooks/Cole.

See Also

`[.factor` for subsetting of factors.

`gl` for construction of “balanced” factors and `C` for factors with specified contrasts. `levels` and `nlevels` for accessing the levels, and `unclass` to get integer codes.

Examples

```
(ff <- factor(substring("statistics", 1:10, 1:10), levels=letters))
as.integer(ff)  # the internal codes
```

```
factor(ff)      # drops the levels that do not occur
ff[, drop=TRUE] # the same, more transparently

factor(letters[1:20], label="letter")

class(ordered(4:1)) # "ordered", inheriting from "factor"

## suppose you want "NA" as a level, and to allowing missing values.
(x <- factor(c(1, 2, "NA"), exclude = ""))
is.na(x)[2] <- TRUE
x  # [1] 1    <NA> NA, <NA> used because NA is a level.
is.na(x)
# [1] FALSE  TRUE FALSE
factor()
```

Anova Tables

Description

Compute analysis of variance (or deviance) tables for one or more fitted model objects.

Usage

```
anova(object, ...)
```

Arguments

`object`	an object containing the results returned by a model fitting function (e.g., `lm` or `glm`).
`...`	additional objects of the same type.

Value

This (generic) function returns an object of class `anova`. These objects represent analysis-of-variance and analysis-of-deviance tables. When given a single argument it produces a table which tests whether the model terms are significant.
When given a sequence of objects, `anova` tests the models against one another in the order specified.
The print method for `anova` objects prints tables in a "pretty" form.

Warning

The comparison between two or more models will only be valid if they are fitted to the same dataset. This may be a problem if there are missing values and **R**'s default of `na.action = na.omit` is used.

References

Chambers, J. M. and Hastie, T. J. (1992) *Statistical Models in S*, Wadsworth & Brooks/Cole.

See Also

`coefficients`, `effects`, `fitted.values`, `residuals`, `summary`, `drop1`, `add1`.

[Package *stats* version 2.4.1 Index]

ANOVA for Linear Model Fits

Description

Compute an analysis of variance table for one or more linear model fits.

Usage

```
## S3 method for class 'lm':
anova(object, ...)

anova.lmlist(object, ..., scale = 0, test = "F")
```

Arguments

`object, ...`	objects of class `lm`, usually, a result of a call to `lm`.
`test`	a character string specifying the test statistic to be used. Can be one of `"F"`, `"Chisq"` or `"Cp"`, with partial matching allowed, or `NULL` for no test.
`scale`	numeric. An estimate of the noise variance *sigma^2*. If zero this will be estimated from the largest model considered.

Details

Specifying a single object gives a sequential analysis of variance table for that fit. That is, the reductions in the residual sum of squares as each term of the formula is added in turn are given in as the rows of a table, plus the residual sum of squares.

The table will contain F statistics (and P values) comparing the mean square for the row to the residual mean square.

If more than one object is specified, the table has a row for the residual degrees of freedom and sum of squares for each model. For all but the first model, the change in degrees of freedom and sum of squares is also given. (This only make statistical sense if the models are nested.) It is conventional to list the models from smallest to largest, but this is up to the user.

Optionally the table can include test statistics. Normally the F statistic is most appropriate, which compares the mean square for a row to the residual sum of squares for the largest model considered. If `scale` is specified chi-squared tests can be used. Mallows' *Cp* statistic is the residual sum of squares plus twice the estimate of *sigma^2* times the residual degrees of freedom.

Value

An object of class `"anova"` inheriting from class `"data.frame"`.

Warning

The comparison between two or more models will only be valid if they are fitted to the same dataset. This may be a problem if there are missing values and **R**'s default of `na.action = na.omit` is used, and `anova.lmlist` will detect this with an error.

Note

Versions of **R** prior to 1.2.0 based F tests on pairwise comparisons, and this behaviour can still be obtained by a direct call to `anovalist.lm`.

References

Chambers, J. M. (1992) *Linear models.* Chapter 4 of *Statistical Models in S* eds J. M. Chambers and T. J. Hastie, Wadsworth & Brooks/Cole.

See Also

The model fitting function `lm`, `anova`.

`drop1` for so-called 'type II' anova where each term is dropped one at a time respecting their hierarchy.

Examples

```
## sequential table
fit <- lm(sr ~ ., data = LifeCycleSavings)
anova(fit)

## same effect via separate models
fit0 <- lm(sr ~ 1, data = LifeCycleSavings)
fit1 <- update(fit0, . ~ . + pop15)
fit2 <- update(fit1, . ~ . + pop75)
fit3 <- update(fit2, . ~ . + dpi)
fit4 <- update(fit3, . ~ . + ddpi)
anova(fit0, fit1, fit2, fit3, fit4, test="F")

anova(fit4, fit2, fit0, test="F") # unconventional order
```

ORIGIN ≡ 0

One-Way Analysis of Variance with Fixed Effects Model

Analysis of Variance (ANOVA) are a broad class of statistical models that fall under the GLM framework. However unlike typical regression where all variables are usually continuous, the independent variable(s) in ANOVA involve membership in classes. Since more than two classes may be present, this approach allows extension of the t-test strategy to comparisions of multiple populations. Since ANOVA is ubiquitous in many experimental settings in biology, its proficient use is often viewed as evidence of good experimental design.

Data Structure:

k groups with not necessarily the same numbers of observations and different means.

Let index i,j indicate the ith column (treatment class) and jth row (object).

One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n1	n2	n3		nk
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

Model:

$Xi,j = \mu + \alpha_i + \varepsilon_{i,j}$ < **where:**

μ is the grand mean of all objects.
α_i is the mean of i = $\mu + \alpha_i$ for each class i.
$\varepsilon_{i,j}$ is the error term specific to each object i,j

Restriction:

$\sum_i n_i \cdot \alpha_i := 0$ < **allows estimation of k parameters.**

Other restrictions are also possible:

$\sum_i \alpha_i := 0$ **or** $\alpha_k := 0$ < **See Rosner p. 558**

Assumptions:

ε_{ij} are a random sample ~ N(0,σ^2)

Number & Means:

$n := \sum_i n_i$ < **total number of observations**

$GM := \frac{1}{n} \cdot \left(\sum_i \sum_j X_{i,j} \right)$ < **grand mean - sample estimate of μ**

$Xbar_i := mean\left(X^{\langle i \rangle}\right)$

Sums of Squares:

$SS_T := \sum_i \sum_j \left(X_{i,j} - GM\right)^2$ < **Total Sum of Squares**

$SS_W := \sum_i \sum_j \left(X_{i,j} - Xbar_i\right)^2$ < **Within (Error) Sum of Squares**

$SS_B := \sum_i \sum_j \left(Xbar_i - GM\right)^2$ < **Between (Treatment) Sum of Squares**

One-Way ANOVA Table:

Source:	SS	df	MS
Between	SS_B	$k - 1$	$\frac{SS_B}{k-1}$
Within	SS_W	$n - k$	$\frac{SS_W}{n-k}$
TOTAL	SS_T		

Example:

Vital Capacity Data in this week's Data folder:

V := READPRN("c:/2007BiostatsData/vital.txt")

$n_0 := 12$ $n_0 = 12$

$n_1 := 39 - 11$ $n_1 = 28$

$n_2 := 83 - 39$ $n_2 = 44$

$n := n_0 + n_1 + n_2$ $n = 84$

< determined by looking at row numbers in the first column...

$V =$

	0	1	2	3
0	1	1	39	4.62
1	2	1	40	5.29
2	3	1	41	5.52
3	4	1	41	3.71
4	5	1	45	4.02
5	6	1	49	5.09
6	7	1	52	2.7
7	8	1	47	4.31
8	9	1	61	2.7
9	10	1	65	3.03
10	11	1	58	2.73
11	12	1	59	3.67
12	13	2	29	5.21
13	14	2	29	5.17
14	15	2	33	4.88
15	16	2	32	4.5

$j0 := 0 .. n_0 - 1$ $j1 := 0 .. n_1 - 1$ $j2 := 0 .. n_2 - 1$

$X0_{j0} := \left(V^{\langle 3 \rangle}\right)_{j0}$ $X1_{j1} := \left(V^{\langle 3 \rangle}\right)_{j1+n_0}$ $X2_{j2} := \left(V^{\langle 3 \rangle}\right)_{j2+n_0+n_1}$

$X_{bar_0} := mean(X0)$ $X_{bar_1} := mean(X1)$ $X_{bar_2} := mean(X2)$

$X_{bar} = \begin{pmatrix} 3.949167 \\ 4.471786 \\ 4.462045 \end{pmatrix}$ **< means for each class**

$GM := mean\left(V^{\langle 3 \rangle}\right)$ $GM = 4.392024$ $\frac{n_0 \cdot X_{bar_0} + n_1 \cdot X_{bar_1} + n_2 \cdot X_{bar_2}}{n} = 4.392024$

^ Grand mean

Sums of Squares:

$i := 0 .. 2$

$SS_T := \sum_{j0} \left(X0_{j0} - GM\right)^2 + \sum_{j1} \left(X1_{j1} - GM\right)^2 + \sum_{j2} \left(X2_{j2} - GM\right)^2$ $SS_T = 47.641$

$SS_W := \sum_{j0} \left(X0_{j0} - X_{bar_0}\right)^2 + \sum_{j1} \left(X1_{j1} - X_{bar_1}\right)^2 + \sum_{j2} \left(X2_{j2} - X_{bar_2}\right)^2$ $SS_W = 44.8936$

$SS_B := \sum_{j0} \left(X_{bar_0} - GM\right)^2 + \sum_{j1} \left(X_{bar_1} - GM\right)^2 + \sum_{j2} \left(X_{bar_2} - GM\right)^2$ $SS_B = 2.7473$

$SS_B + SS_W = 47.641$

One-Way ANOVA Table:

$k := 3$ **< number of classes**

Source:	SS	df	MS	
Between	$SS_B = 2.7473$	$k - 1 = 2$	$MS_B := \frac{SS_B}{k-1}$	$MS_B = 1.3737$
Within	$SS_W = 44.8936$	$n - k = 81$	$MS_W := \frac{SS_W}{n-k}$	$MS_W = 0.5542$
TOTAL	$SS_T = 47.641$			

Prototype in R:

COMMANDS:

```
> V=read.table("c:/2007BiostatsData/vital.txt")
> V
> Y=V$vital.capacity
> X=V$group
```

NOTE that there is a RIGHT WAY and a WRONG WAY to do ANOVA in R:

WRONG WAY:

```
> anova(lm(Y~X))

Analysis of Variance Table

Response: Y
           Df  Sum Sq Mean Sq F value  Pr(>F)
X           1   1.609   1.609  2.8658 0.09428 .
Residuals  82  46.032   0.561
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

What this does is produce an ANOVA on the Linear Regression of the dependent variable (Y) with the values reported for the independent variable (X). These values are class indicators (1,2,3) and are meaningless. The fact that this is a Linear Regression ANOVA can be seen in the report of 1 DF for variable X in the ANOVA chart...

RIGHT WAY:

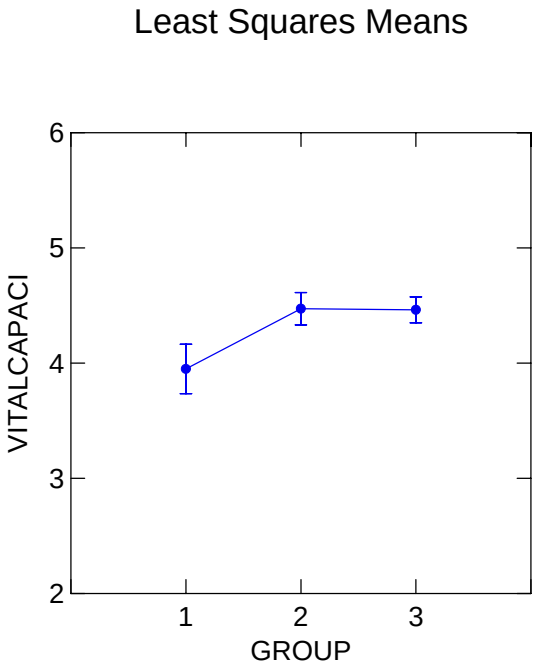
```
> X=factor(V$group)    < here the values of variable V$group are converted into class "factors"
> X
> anova(lm(Y~X))

Analysis of Variance Table

Response: Y
           Df  Sum Sq Mean Sq F value  Pr(>F)
X           2   2.747   1.374  2.4785 0.09021 .
Residuals  81  44.894   0.554
```

^ These values match above.

```
> V
   group age vital.capacity
1      1  39          4.62
2      1  40          5.29
3      1  41          5.52
4      1  41          3.71
5      1  45          4.02
6      1  49          5.09
7      1  52          2.70
8      1  47          4.31
9      1  61          2.70
10     1  65          3.03
11     1  58          2.73
12     1  59          3.67
13     2  29          5.21
14     2  29          5.17
15     2  33          4.88
16     2  32          4.50
17     2  31          4.47
18     2  29          5.12
19     2  29          4.51
20     2  30          4.85
21     2  21          5.22
22     2  28          4.62
23     2  23          5.07
24     2  35          3.64
25     2  38          3.64
26     2  38          5.09
27     2  43          4.61
28     2  39          4.73
29     2  38          4.58
30     2  42          5.12
31     2  43          3.89
32     2  43          4.62
33     2  37          4.30
34     2  50          2.70
35     2  50          3.50
36     2  45          5.06
37     2  48          4.06
38     2  51          4.51
39     2  46          4.66
40     2  58          2.88
41     3  27          5.29
42     3  25          3.67
43     3  24          5.82
44     3  32          4.77
45     3  23          5.71
46     3  25          4.47
47     3  32          4.55
48     3  18          4.61
49     3  19          5.86
50     3  26          5.20
51     3  33          4.44
52     3  27          5.52
53     3  33          4.97
54     3  25          4.99
55     3  42          4.89
56     3  35          4.09
57     3  35          4.24
58     3  41          3.88
59     3  38          4.85
60     3  41          4.79
61     3  36          4.36
62     3  36          4.02
63     3  41          3.77
64     3  41          4.22
65     3  37          4.94
66     3  42          4.04
67     3  39          4.51
68     3  41          4.06
69     3  43          4.02
70     3  41          4.99
71     3  48          3.86
72     3  47          4.68
73     3  53          4.74
74     3  49          3.76
75     3  54          3.98
76     3  48          5.00
77     3  49          3.31
78     3  47          3.11
79     3  52          4.76
80     3  58          3.95
81     3  62          4.60
82     3  65          4.83
83     3  62          3.18
84     3  59          3.03
```

Prototype in SYSTAT:

```
          Effects coding used for categorical variables in model.

Categorical values encountered during processing are:
GROUP (3 levels)
          1,        2,        3

Dep Var: VITALCAPACI   N: 84   Multiple R: 0.24014   Squared multiple R: 0.05767


                         Analysis of Variance

Source             Sum-of-Squares   df  Mean-Square     F-ratio       P

GROUP                     2.74734    2      1.37367     2.47846     0.09021

Error                    44.89362   81      0.55424


                    Durbin-Watson D Statistic     1.692
                    First Order Autocorrelation   0.126
```



ORIGIN ≡ 0

F-Test for H_0: All $\alpha_i = 0$ in One-Way ANOVA with Fixed Effects Model

Inferences on the means of the multiple populations indicated by the class ("factor" or "group") variable follow directly from the ANOVA table.

Data Structure:

k groups with not necessarily the same numbers of observations and different means.

Let index i,j indicate the ith column (treatment class) and jth row (object).

One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n1	n2	n3		nk
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

F Test for Overall Comparison of Class Means:

Model:

$X_{i,j} = \mu + \alpha_i + \varepsilon_{i,j}$ < where:

μ is the grand mean of all objects.
α_i is the mean of i = $\mu + \alpha_i$ for each class i.
$\varepsilon_{i,j}$ is the error term specific to each object i,j

Restriction:

$\sum_i n_i \cdot \alpha_i := 0$ < allows estimation of k parameters.

Other restrictions are also possible:

$\sum_i \alpha_i := 0$ or $\alpha_k := 0$ < See Rosner p. 558

Assumptions:

ε_{ij} are a random sample ~ N(0,σ^2)

One-Way ANOVA Table:

Source:	SS	df	MS
Between	SS_B	$k - 1$	$\frac{SS_B}{k-1}$
Within	SS_W	$n - k$	$\frac{SS_W}{n-k}$
TOTAL	SS_T		

Hypotheses:

H_0: $\alpha_i = 0$ *for all* i < All treatment class deviations from the grand mean are 0

H_1: *At least one* α_i <> 0 < Two sided test

Test Statistic:

$F := \frac{MS_B}{MS_W}$ < Ratio of "between" versus "within" Mean Squares

Distribution of the test Statistic F:

If H_0 is true then F ~F((k-1),(n-k)) where: k = number of classes
n = total number of observations

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1-\alpha)$ $CV := qF[1-\alpha,(k-1),(n-k)]$

Decision Rule:

IF F > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(Φ_F(F),1-Φ_F(F)

$P := \min[\, pF[F,(k-1),(n-k)], pF[F,(k-1),(n-k)]\,]$

Example:

Vital Capacity Data in this week's Data folder:

One-Way ANOVA Table:

$k := 3$ **< number of classes**

$n := 84$

From R:

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	**2**	**2.747**	**1.374**	**2.4785**	**0.09021 .**
Residuals	**81**	**44.894**	**0.554**		

Source:	**SS**	**df**	**MS**	
Between	$SS_B := 2.747$	$k-1=2$	$MS_B := \frac{SS_B}{k-1}$	$MS_B = 1.3735$
Within	$SS_W := 44.894$	$n-k=81$	$MS_W := \frac{SS_W}{n-k}$	$MS_W = 0.5542$
TOTAL	$SS_T := SS_B + SS_W$	$SS_T = 47.641$		

Test Statistic:

$F := \frac{MS_B}{MS_W}$ $F = 2.4781$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF[1-\alpha,(k-1),(n-k)]$ $CV = 3.1093$

Decision Rule:

IF F > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F = 2.4781$ $CV = 3.1093$

Probability Value:

$P := \min[\, pF[F,(k-1),(n-k)], 1-pF[F,(k-1),(n-k)]\,]$ $P = 0.0902$

^ values match R output above

Another Example:

Iris dataset:

R COMMANDS:

```
> Iris=read.table("c:/2007BiostatsData/iris.txt")
> Iris
> Y=Iris$Sepal.Length
> length(Y)
> X=Iris$Species
> anova(lm(Y~X))
```

$k := 3$ **< number of classes**

$n := 150$ **< number of objects**

One-Way ANOVA Table:

From R:

Analysis of Variance Table
Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	**2**	**63.212**	**31.606**	**119.26**	**< 2.2e-16 *****
Residuals	**147**	**38.956**	**0.265**		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

Source:	SS	df	MS	
Between	$SS_B := 63.212$	$k - 1 = 2$	$MS_B := \frac{SS_B}{k-1}$	$MS_B = 31.606$
Within	$SS_W := 38.956$	$n - k = 147$	$MS_W := \frac{SS_W}{n-k}$	$MS_W = 0.265$
TOTAL	$SS_T := SS_B + SS_W$	$SS_T = 102.168$		

Test Statistic:

$F := \frac{MS_B}{MS_W}$ $F = 119.2649$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF[1 - \alpha, (k-1), (n-k)]$ $CV = 3.0576$

Decision Rule:

IF F > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F = 119.2649$ $CV = 3.0576$

Probability Value:

$P := \min[\, pF[F, (k-1), (n-k)], 1 - pF[F, (k-1), (n-k)]\,]$ $P = 0$

Note: Rejection of H_0 here allows one to continue testing for values of *specific* α_i's

ORIGIN ≡ 0

t-Test for H_0: $\alpha_i = \alpha_j$ versus H_1: $\alpha_i <> \alpha_j$ in One-Way ANOVA with Fixed Effects Model

When the F-Test for ANOVA rejects the null hypothesis that all α_i's = 0, then one usually wants to determine *specifically which* of the α_i's <> 0. This test allow us to do this within the context of multiple possible tests in ANOVA.

Data Structure:

k groups with not necessarily the same numbers of observations and different means.

Let index i,j indicate the ith column (treatment class) and jth row (object).

One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n1	n2	n3		nk
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

t-Test for Comparison of Means for Specific Class Pairs:

Model:

$X_{i,j} = \mu + \alpha_i + \varepsilon_{i,j}$ < **where:**

μ is the grand mean of all objects.
α_i is the mean of i = $\mu + \alpha_i$ for each class i.
$\varepsilon_{i,j}$ is the error term specific to each object i,j

Restriction:

$\sum_i n_i \cdot \alpha_i := 0$ < **allows estimation of k parameters.**

Other restrictions are also possible:

$\sum_i \alpha_i := 0$ **or** $\alpha_k := 0$ < **See Rosner p. 558**

Assumptions:

ε_{ij} are a random sample ~ N(0,σ^2)

One-Way ANOVA Table:

Source:	SS	df	MS
Between	SS_B	$k - 1$	$\frac{SS_B}{k-1}$
Within	SS_W	$n - k$	$\frac{SS_W}{n-k}$
TOTAL	SS_T		

Hypotheses:

H_0: $\alpha_i = \alpha_j$ *for specific* i & j < **Means in treatment classes i & j are the same as grand mean**

H_1: $\alpha_i <> \alpha_j$ *for specific* i & j < **Two sided test**

Test Statistic:

$$t := \frac{X_{bar_i} - X_{bar_j}}{\sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)}}$$

< **Normalized distance between mean of class i & j**

Distribution of the test Statistic t:

If H_0 is true then t ~t(n-k) **where: k = number of classes**
n = total number of observations

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k\right)$

Note degrees of freedom = (n-k)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,]$

Example: Vital Capacity dataset by group

R COMMANDS:

```
> V=read.table("c:/2007BiostatsData/vital.txt")
> summary(V)
```

```
     group            age        vital.capacity
Min.   :1.000   Min.   :18.00   Min.   :2.700
1st Qu.:2.000   1st Qu.:32.00   1st Qu.:3.935
Median :3.000   Median :41.00   Median :4.530
Mean   :2.381   Mean   :40.55   Mean   :4.392
3rd Qu.:3.000   3rd Qu.:48.00   3rd Qu.:4.947
Max.   :3.000   Max.   :65.00   Max.   :5.860
```

```
> attach(V)
> X1=vital.capacity[group=="1"]
> X2=vital.capacity[group=="2"]
> X3=vital.capacity[group=="3"]
> summary(X1)
```

$k := 3$ **< number of classes = groups**

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.700   2.955   3.865   3.949   4.737   5.520
```

```
> length(X1)
[1] 12
```

$n_1 := 12$ $X_{bar_1} := 3.949$ **< number of objects & mean of X1**

```
> summary(X2)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.700   4.240   4.615   4.472   5.062   5.220
```

```
> length(X2)
[1] 28
```

$n_2 := 28$ $X_{bar_2} := 4.472$ **< number of objects & mean of X2**

```
> length(X3)
[1] 44
```

$n_3 := 44$

$n := n_1 + n_2 + n_3$ $n = 84$

```
> Y=vital.capacity
> X=factor(group)
> anova(lm(Y~X))
```

One-Way ANOVA Table:

From R:

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	2	2.747	1.374	2.4785	0.09021 .
Residuals	81	44.894	0.554		

--

$MS_W := 0.554$ **< MS Residuals**

Hypotheses:

H_0: $\alpha_i = \alpha_j$ *for specific* i & j **< Means in treatment classes i & j are the same as grand mean**

H_1: $\alpha_i <> \alpha_j$ *for specific* i & j **< Two sided test**

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_2}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad t = -2.0365 \qquad X_{bar_1} - X_{bar_2} = -0.523$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n - k\right)$ $C_1 = -1.9897$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k\right)$ $C_2 = 1.9897$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -2.0365$ $C_1 = -1.9897$ $C_2 = 1.9897$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,]$ $P = 0.045$

Prototype in SYSTAT:

```
Effects coding used for categorical variables in model.

Categorical values encountered during processing are:
GROUP (3 levels)
          1,        2,        3

Dep Var: VITALCAPACI   N: 84   Multiple R: 0.24014   Squared multiple R: 0.05767
                         Analysis of Variance

Source             Sum-of-Squares   df  Mean-Square     F-ratio       P

GROUP                     2.74734     2      1.37367     2.47846     0.09021

Error                    44.89362    81      0.55424

  ROW GROUP
    1  1
    2  2
    3  3
  Using least squares means.
  Post Hoc test of VITALCAPACI
  ----------------------------------------------------------
  Using model MSE of 0.554 with 81 df.
  Matrix of pairwise mean differences:
                          1           2           3
               1        0.00000
               2        0.52262     0.00000
               3        0.51288    -0.00974     0.00000
  Fisher's Least-Significant-Difference Test.
  Matrix of pairwise comparison probabilities:
                          1           2           3
               1        1.00000
               2        0.04516     1.00000
               3        0.03747     0.95697     1.00000
```

< values match for difference in means

and Probability

Prototype in R:

COMMANDS:

```
> V=read.table("c:/2007BiostatsData/vital.txt")
> V
> attach(V)
> Y=vital.capacity
> X=factor(group)
> anova(lm(Y~X))
```

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	2	2.747	1.374	2.4785	0.09021 .
Residuals	81	44.894	0.554		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

```
> pairwise.t.test(Y,X,p.adj="none")
```

^ values match SYSTAT above

Pairwise comparisons using t tests with pooled SD

data: Y and X

	1	2
2	0.045	-
3	0.037	0.957

< pairwise Probabilities match SYSTAT

P value adjustment method: none

Another Example:

Iris dataset

Sepal Length:

R COMMANDS:

```
> iris=read.table("c:/2007BiostatsData/iris.txt")
> summary(iris)
```

Sepal.Length	Sepal.Width	Petal.Length	Petal.Width
Min. :4.300	Min. :2.000	Min. :1.000	Min. :0.100
1st Qu.:5.100	1st Qu.:2.800	1st Qu.:1.600	1st Qu.:0.300
Median :5.800	Median :3.000	Median :4.350	Median :1.300
Mean :5.843	Mean :3.057	Mean :3.758	Mean :1.199
3rd Qu.:6.400	3rd Qu.:3.300	3rd Qu.:5.100	3rd Qu.:1.800
Max. :7.900	Max. :4.400	Max. :6.900	Max. :2.500

Species

setosa :50
versicolor:50
virginica :50

$k := 3$ **< number of classes = species**

$n_1 := 50$

$n := n_1 + n_2 \quad n = 100$ **< number of objects**

```
> attach(iris)
> X1=Sepal.Length[Species="setosa"]
> X2=Sepal.Length[Species=="virginica"]
> summary(X1)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
4.300	4.800	5.000	5.006	5.200	5.800

$X_{bar_1} := 5.006$

```
> summary(X2)
```

Min.	1st Qu.	Median	Mean	3rd Qu.	Max.
4.900	6.225	6.500	6.588	6.900	7.900

```
> Y=Sepal.Length
> X=Species
> anova(lm(Y~X))
```

$X_{bar_2} := 6.588$

One-Way ANOVA Table:

From R: **Analysis of Variance Table**

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	**2**	**63.212**	**31.606**	**119.26**	**< 2.2e-16 *****
Residuals	**147**	**38.956**	**0.265**		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

$MS_W := 0.265$ **< MS Residuals**

Hypotheses:

H_0: $\alpha_i = \alpha_j$ *for specific* i & j **< Means in treatment classes i & j are the same as grand mean**

H_1: $\alpha_i <> \alpha_j$ *for specific* i & j **< Two sided test**

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_2}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad t = -15.3657 \qquad X_{bar_1} - X_{bar_2} = -1.582$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n - k\right)$ $C_1 = -1.9847$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k\right)$ $C_2 = 1.9847$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -15.3657$ $C_1 = -1.9847$ $C_2 = 1.9847$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,]$ $P = 0$

Results in SYSTAT:

```
Effects coding used for categorical variables in model.

Categorical values encountered during processing are:
SPECIES$ (3 levels)
   setosa, versicolor, virginica
3 case(s) deleted due to missing data.

Dep Var: SEPALLENGTH   N: 150   Multiple R: 0.78658   Squared multiple R: 0.61871

                          Analysis of Variance
Source             Sum-of-Squares   df  Mean-Square     F-ratio       P
SPECIES$                63.21213     2     31.60607   119.26450     0.00000
Error                   38.95620   147      0.26501

     Durbin-Watson D Statistic     2.043
     First Order Autocorrelation  -0.028
     COL/
     ROW SPECIES$
       1  setosa
       2  versicolor
       3  virginica
     Using least squares means.
     Post Hoc test of SEPALLENGTH
     ----------------------------------------------------------------------
     Using model MSE of 0.265 with 147 df.
     Matrix of pairwise mean differences:
                              1           2           3
                  1        0.00000
                  2        0.93000     0.00000
                  3        1.58200     0.65200     0.00000
     Fisher's Least-Significant-Difference Test.
     Matrix of pairwise comparison probabilities:
                              1           2           3
                  1        1.00000
                  2        0.00000     1.00000
                  3        0.00000     0.00000     1.00000
```

^ **Although agreeing, the probabilities here are simply to small to use as prototype.**

ORIGIN ≡ 0

t-Test for Linear Contrasts H_0: L = 0 versus H_1: L <> 0 in One-Way ANOVA with Fixed Effects Model

When the F-Test for ANOVA rejects the null hypothesis that all α_i's = 0, then one usually wants to determine *specifically which* of the α_i's <> 0. This test is a generalization of the t-test comparing pairs of means.
Here any linear combination L = $c_1Xbar_1 + c_2Xbar_2 + c_3Xbar_3 + ... + c_kXbar_k$ can be tested where $X_1, X_2, X_3 .. X_k$ represent samples derived from different populations 1,2, 3 ...k.

Data Structure:

k groups with not necessarily the same numbers of observations and different means.

Let index i,j indicate the ith column (treatment class) and jth row (object).

One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n1	n2	n3		nk
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

Linear Combination:

$$L := \sum_{i=0}^{k-1} c_i \cdot X_{bar_i}$$ < $L = c_1Xbar_1 + c_2Xbar_2 + c_3Xbar_3 + ... + c_kXbar_k$

With Further Condition as Linear Contrast:

$$\sum_{i=0}^{k-1} c_i := 0$$ < coefficients of the Linear combination must add to zero

t-Test for Linear Contrasts H_0: L = 0 versus H_1: L <> 0:

Model:

$X_{i,j} = \mu + \alpha_i + \varepsilon_{i,j}$ < where: μ is the grand mean of all objects.
α_i is the mean of i = $\mu + \alpha_i$ for each class i.
$\varepsilon_{i,j}$ is the error term specific to each object i,j

$L = c_1Xbar_1 + c_2Xbar_2 + c_3Xbar_3 + ... + c_kXbar_k$ < definition of Linear Contrast

$$\mu_L := \sum_{i=0}^{k-1} c_i \cdot \alpha_i$$ < mean of the Linear Contrast

Restrictions:

$$\sum_i n_i \cdot \alpha_i := 0$$ < allows estimation of k parameters.
Other restrictions are also possible:

$$\sum_i \alpha_i := 0$$ or $\alpha_k := 0$ < See Rosner p. 558

$$\sum_{i=0}^{k-1} c_i := 0$$ < restriction for the linear contrast

Assumptions:

ε_{ij} are a random sample ~ $N(0,\sigma^2)$

One-Way ANOVA Table:

Source:	SS	df	MS
Between	SS_B	$k - 1$	$\frac{SS_B}{k-1}$
Within	SS_W	$n - k$	$\frac{SS_W}{n-k}$
TOTAL	SS_T		

Hypotheses:

H_0: $\mu_L = 0$ **< Means of Linear Contrast is zero**

H_1: $\mu_L <> 0$ **< Two sided test**

Test Statistic:

$$t := \frac{L}{\sqrt{MS_W \cdot \sum_i \frac{(c_i)^2}{n_i}}}$$

< Linear Contrast normalized by standard Error

Distribution of the test Statistic t:

If H_0 is true then t ~t(n-k) **where: k = number of classes**
n = total number of observations

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := \text{inverse}\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := \text{inverse}\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k\right)$

Note degrees of freedom = (n-k)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t)

$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,]$

Example: Rosner Pulmonary Disease Example 12.10 p. 572

$MS_W := 0.636$ **< given from ANOVA Table 12.3 p. 564**

$$X_{bar} := \begin{pmatrix} 3.78 \\ 3.30 \\ 3.32 \\ 3.23 \\ 2.73 \\ 2.59 \end{pmatrix} \quad n := \begin{pmatrix} 200 \\ 200 \\ 50 \\ 200 \\ 200 \\ 200 \end{pmatrix}$$

< means & number of observations given from summary Table 12.1 p. 558

Linear contrast vector of coefficients >

$$c := \begin{pmatrix} 1.0 \\ 0 \\ 0 \\ -0.1 \\ -0.7 \\ -0.2 \end{pmatrix}$$

$k := 6$ **< number of classes**

$N := \sum n$ $N = 1050$ **< total number of observations**

The Linear Contrast:

$L := c^T \cdot X_{bar}$ **< matrix algebra multiplication of vectors** $L = (1.028)$ **< confirmed p. 572**

Standard Error of the Linear Contrast:

$i := 0 .. k - 1$

$$\sqrt{MS_W \cdot \sum_i \frac{(c_i)^2}{n_i}} = 0.07$$ < **confirmed p. 572**

Hypotheses:

H_0: $\mu_L = 0$ < **Means of Linear Contrast is zero**

H_1: $\mu_L <> 0$ < **Two sided test**

Test Statistic:

$$t := \frac{L}{\sqrt{MS_W \cdot \sum_i \frac{(c_i)^2}{n_i}}}$$ $t = (14.6899)$ < **confirmed p. 572**

Critical Value of the Test:

$\alpha := 0.05$ < **Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, N - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, N - k\right)$

$C_1 = -1.9622$ $C_2 = 1.9622$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = (14.6899)$ $C_1 = -1.9622$ $C_2 = 1.9622$

Probability Value:

P = minimum(2 Φ_t(t),1-2 Φ_t(t) $N - k = 1044$ < **confirmed p. 572**

$P := \min[2 \cdot pt(t, N - k), 2 \cdot (1 - pt(t, N - k))]$ $P = 0$ < **confirmed p. 572**

Another Example: Vital Capacity dataset by group

R COMMANDS:

```
> V=read.table("c:/2007BiostatsData/vital.txt")
> summary(V)
```

```
     group            age        vital.capacity
Min.   :1.000   Min.   :18.00   Min.   :2.700
1st Qu.:2.000   1st Qu.:32.00   1st Qu.:3.935
Median :3.000   Median :41.00   Median :4.530
Mean   :2.381   Mean   :40.55   Mean   :4.392
3rd Qu.:3.000   3rd Qu.:48.00   3rd Qu.:4.947
Max.   :3.000   Max.   :65.00   Max.   :5.860
```

```
> attach(V)
> X1=vital.capacity[group=="1"]
> X2=vital.capacity[group=="2"]
> X3=vital.capacity[group=="3"]
> summary(X1)
```

$k := 3$ < **number of classes = groups**

```
> length(X1)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.700   2.955   3.865   3.949   4.737   5.520
```

[1] 12 $nn_0 := 12$ $Xn_{bar_0} := 3.949$ < **number of objects & mean of X1**

```
> summary(X2)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  2.700   4.240   4.615   4.472   5.062   5.220
```

```
> length(X2)
```

[1] 28 $nn_1 := 28$ $Xn_{bar_1} := 4.472$ < **number of objects & mean of X2**

> summary(X3)

Min. 1st Qu. Median Mean 3rd Qu. Max.
3.030 4.010 4.530 4.462 4.902 5.860

> length(X3)

[1] 44 $nn_2 := 44$ $Xn_{bar_2} := 4.462$ **< number of objects & mean of X3**

> Y=vital.capacity
> X=factor(group)
> anova(lm(Y~X))

$N := \sum nn$ $N = 84$ $nn = \begin{pmatrix} 12 \\ 28 \\ 44 \end{pmatrix}$

One-Way ANOVA Table:

From R: **Analysis of Variance Table**

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	**2**	**2.747**	**1.374**	**2.4785**	**0.09021 .**
Residuals	**81**	**44.894**	**0.554**		

--

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

$MS_W := 0.554$ **< MS Residuals**

$Xn_{bar} = \begin{pmatrix} 3.949 \\ 4.472 \\ 4.462 \end{pmatrix}$ $nn = \begin{pmatrix} 12 \\ 28 \\ 44 \end{pmatrix}$ **< means & number of observations given from summary Table 12.1 p. 558**

$k := 3$ **< number of classes**

$N = 84$ **< total number of observations**

Linear contrast vector of coefficients > $c := \begin{pmatrix} 1 \\ -1 \\ 0 \end{pmatrix}$

The Linear Contrast:

$L := c^T \cdot Xn_{bar}$ **< matrix algebra multiplication of vectors** $L = (-0.523)$

Standard Error of the Linear Contrast:

$i := 0 .. k - 1$

$$\sqrt{MS_W \cdot \sum_i \frac{(c_i)^2}{nn_i}} = 0.2568$$

Standard Error of two-sample t-test in Biostatistics 49:

$$\sqrt{MS_W \cdot \left(\frac{1}{nn_0} + \frac{1}{nn_1}\right)} = 0.2568$$

^ same result

Hypotheses:

H_0: $\mu_L = 0$

H_1: μ_L <> 0

Test Statistic:

$$t := \frac{L}{\sqrt{MS_W \cdot \sum_i \frac{(c_i)^2}{nn_i}}}$$ $t = (-2.0365)$

from two-sample t-test:

$$t := \frac{Xn_{bar_0} - Xn_{bar_1}}{\sqrt{MS_W \cdot \left(\frac{1}{nn_0} + \frac{1}{nn_1}\right)}}$$ $t = (-2.0365)$

^ same result

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, N - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, N - k\right)$

$C_1 = -1.9897$ $C_2 = 1.9897$

< different again from the two-sample case.

Decision Rule:

IF $|t| > C$, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = (-2.0365)$ $C_1 = -1.9897$ $C_2 = 1.9897$

Probability Value:

$P = \text{minimum}(2\,\Phi_t(t), 1-2\,\Phi_t(t))$

$P := \min[\,2 \cdot pt(t, N-k), 2 \cdot (1 - pt(t, N-k))\,]$ $P = 0.045$

Prototype in SYSTAT:

Use GLM in the Statistics Menu, Estimate Model and under Category specify variable "group" as a categorical variable. The Run estimate. Now go back to GLM Hypothesis Test, specify "group" in the Effects box, and select the Contrast Button. In the pop-up box, write the contrast vector as c^T above. Run Hypothesis.

```
Effects coding used for categorical variables in model.

Categorical values encountered during processing are:
GROUP (3 levels)
          1,        2,        3

Dep Var: VITAL   N: 84   Multiple R: 0.24014   Squared multiple R: 0.05767


                            Analysis of Variance

Source              Sum-of-Squares   df  Mean-Square     F-ratio       P

GROUP                     2.74734     2      1.37367     2.47846     0.09021

Error                    44.89362    81      0.55424


Durbin-Watson D Statistic      1.692
First Order Autocorrelation    0.126
```

Test for effect called: GROUP

A Matrix

1	2	3
0.00000	1.00000	-1.00000

Test of Hypothesis

Source	SS	df	MS	F	P
Hypothesis	2.29430	1	2.29430	4.13952	0.04516
Error	44.89362	81	0.55424		

^ equivalent Probability (P) reported here on the basis of an equivalent F test.

Assignment for Week 13 and 20 point Quiz

For your final assignment of this term – *and a Quiz* – Using a small data set of your own, I want you to do the following using *hand calculations* (as calculations like this might be on Exam 3):

1. *Biostatistics* 39 - Perform "simple" linear regression:

a. show calculations for: L_{xx}, L_{yy}, and L_{xy}.

b. calculate α and β.

c. calculate regression predictions (Y_{hat}) and residuals.

2. *Biostatistics* 40 – Calculate the ANOVA for Regression standard table:

a. show calculations for Sum of Squares: SS Total, SS Regression & SS error.

b. Show the completed ANOVA chart with degrees of freedom & Mean Squares.

3. *Biostatistics* 40 – Perform the omnibus F test for the Regression

Show your work including assumptions, model, hypotheses, decision rule, probability values & result.

4. *Biostatistics* 41 – Calculate the following confidence/prediction intervals for your "simple" regression:

a. Confidence interval for regression slope.

b. Confidence interval for your regression predictions.

c. Prediction interval for *new observations* based on your regression.

5. *Biostatistics* 42 – Calculate the following for your "simple" regression:

a. coefficient of correlation.

b. coefficient of determination.

6. *Biostatistics* 42 – Perform the following tests based on your "simple" regression:

a. Test for a *specified value* of regression slope (β_0).

b. Test for presence of correlation (ρ).

7. *Biostatistics* 43 – Perform a multiple regression *using R* based on a dataset that you compose. Extract and report the ANOVA table from R's results.

8. *Biostatistics* 44 – Using the output from R:

a. Perform the omnibus F test for the Regression – show all work.

b. Test each slope parameter (β) separately – show your work *including calculation of the test statistic*.

9. *Biostatistics* 47 – Perform a One-Way ANOVA for fixed effects. Show calculations for:

a. Sums of Squares & Mean Squares

b. Show your ANOVA standard table

10. *Biostatistics* 48 – Perform the omnibus F test for the One-Way ANOVA – show all work.

11. *Biostatistics* 49 – Perform the test for equivalence of means between specific pairs of groups in One-Way ANOVA – show your work.

12. *Biostatistics* 50 – Perform a test for a chosen Linear Contrast in One-Way ANOVA – show all work.

13. *Biostatistics* 51 – Perform a Two-Way ANOVA with fixed effects:

a. Set up the ANOVA standard table – show your work.

14. *Biostatistics* 51 – Perform the omnibus F tests for the Two-Way ANOVA – show all work.

15. *Biostatistics* 52 – Perform an example Kruskal-Wallis test. Show all work.

16. *Biostatistics* 53 – Perform a specific Fishers LSD comparison between two means in a One-Way ANOVA. Show all work in the test and calculate the associated confidence interval.

17. *Biostatistics* 54 – Perform the omnibus F tests for Repeated Measures One-Way ANOVA – show all work.

18. *Biostatistics* 55 – Perform an example Kruskal-Wallis test. Show all work.

ORIGIN ≡ 0

Two-Way ANOVA - Equal Sample Sizes

The ANOVA approach analyzes means from multiple populations with membership in each sample determined by discrete values of a classification variable. The Two-Way (and higher) ANOVA stategy extends the system of classifications to two (or more) variables. Here we look at analysis of fully randomized ballanced designs in which numbers of observatios in each class (or block) of data are all the same.

Data Structure:

Data are structured as an R X C Contingency Table with cells representing simultaneous classification by two variables. Numeric values Yij for n objects are placed in each cell

Let index i,j indicate the ith row (treatment classes of Variable R) and jth column (treatment classes of Variable C)

Treatment Classes of Variable R:	Two-Way ANOVA Treatment Classes of Variable C:				
	#1	#2	#3	…	#j
#1	n	n	n	…	n
#2	n	n	n	…	n
#3	n	n	n	…	n
…	…	…	…	…	…
#i	n	n	n	…	n
Each cell consists of n replicates with means $Ybar_{ij}$					

Also let: $Ybar_{i.}$ **= mean over all columns for row i.**
$Ybar_{j.}$ **= mean over all rows for column j.**
$Ybar_{..}$ **= overall mean.**

Model:

$$Y_{i,j} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$$ **where:**

μ is a constant = grand mean of all objects.
α_i is effect coefficient for classes i in Variable R.
β_j is effect coefficient for classes j in Variable C.
γ_{ij} is interaction coefficient for classes i,j between Variables R and C.
ε_{ijk} is the error term specific to each object i,j,k

Restrictions:

$$\sum_i \alpha_i := 0 \qquad \sum_j \beta_j := 0 \qquad \sum_i \gamma_{ij} := 0 \qquad \sum_j \gamma_{ij} := 0$$ **for all i & j**

Assumptions:

– ε_{ijk} are a random sample ~ N(0,σ^2)
– variance is homogeneous across cells

Data in Each Cell:

$Y_{i,j,k}$ **< k Observations in each cell defined by rows i, & columns j**

Number & Means:

N **< Total number of observations found by multiplying k by the number of cells**

$Y_{bar_{ij.}}$ **< Cell means - averages of observations within a cell**

$Y_{bar_{i.}}$ **< Row means - averages for each row**

$Y_{bar_{.j}}$ **< Column means - averages for each column**

$Y_{bar_{..}}$ **< Grand mean - average of all observations**

Sums of Squares:

$I := 2$ **< total number of rows**

$J := 2$ **< total number of columns**

$K := 5$ **< total number of replicates = n**

$$SS_{Rows} := J \cdot K \cdot \sum_{i} \left(Ybar_{i.} - Ybar_{..} \right)^2$$ **< Sum of Squares for Rows**

$$SS_{Cols} := I \cdot K \cdot \sum_{j} \left(Ybar_{.j} - Ybar_{..} \right)^2$$ **< Sum of Squares for Columns**

$$SS_{Int} := K \cdot \left[\sum_{j} \sum_{i} \left(Ybar_{ij} - Ybar_{i.} - Ybar_{.j} + Ybar_{..} \right)^2 \right]$$ **< Sum of Squares for Interactions**

$$SS_{E} := \sum_{k} \sum_{j} \sum_{i} \left(Y_{ijk} - Ybar_{ij} \right)^2$$ **< Sum of Squares for Error (Within)**

$$SS_{T} := \sum_{k=0}^{N-1} \left(Y_{ijk} - Ybar_{..} \right)^2$$ **< Total Sum of Squares**

Example: Calcium Concentration measured with two factors: Sex & Hormone Treatment:

$Z := READPRN("c:/2007BiostatsData/ZarExample12.1a.txt")$

$$Z = \begin{pmatrix} 1 & 16.5 & 0 & 0 \\ 2 & 18.4 & 0 & 0 \\ 3 & 12.7 & 0 & 0 \\ 4 & 14 & 0 & 0 \\ 5 & 12.8 & 0 & 0 \\ 6 & 14.5 & 1 & 0 \\ 7 & 11 & 1 & 0 \\ 8 & 10.8 & 1 & 0 \\ 9 & 14.3 & 1 & 0 \\ 10 & 10 & 1 & 0 \\ 11 & 39.1 & 0 & 1 \\ 12 & 26.2 & 0 & 1 \\ 13 & 21.3 & 0 & 1 \\ 14 & 35.8 & 0 & 1 \\ 15 & 40.2 & 0 & 1 \\ 16 & 32 & 1 & 1 \\ 17 & 23.8 & 1 & 1 \\ 18 & 28.8 & 1 & 1 \\ 19 & 25 & 1 & 1 \\ 20 & 29.3 & 1 & 1 \end{pmatrix}$$

Data in Each Cell:

$$a := \begin{pmatrix} 16.5 \\ 18.4 \\ 12.7 \\ 14.0 \\ 12.8 \end{pmatrix} \quad b := \begin{pmatrix} 14.5 \\ 11.0 \\ 10.8 \\ 14.3 \\ 10.0 \end{pmatrix} \quad c := \begin{pmatrix} 39.1 \\ 26.2 \\ 21.3 \\ 35.8 \\ 40.2 \end{pmatrix} \quad d := \begin{pmatrix} 32.0 \\ 23.8 \\ 28.8 \\ 25.0 \\ 29.3 \end{pmatrix}$$

Number & Means:

$n := length(a)$ $n = 5$ **< number in each cell**

$N := 4 \cdot n$ $N = 20$ **< total number of observations**

$$YB_{bar} := \begin{pmatrix} mean(a) & mean(b) \\ mean(c) & mean(d) \end{pmatrix} \quad YB_{bar} = \begin{pmatrix} 14.88 & 12.12 \\ 32.52 & 27.78 \end{pmatrix}$$ **< cell means ($Ybar_{ij.}$)**

$i := 0 .. 1$ $j := 0 .. 1$ $k := 0 .. n - 1$

$$YR_{bar_i} := \frac{1}{2} \cdot \sum_{j} YB_{bar_{i,j}} \quad YR_{bar} = \begin{pmatrix} 13.5 \\ 30.15 \end{pmatrix}$$ **< row means ($Ybar_{i..}$)**

$$YC_{bar_j} := \frac{1}{2} \cdot \sum_{i} YB_{bar_{i,j}} \quad YC_{bar} = \begin{pmatrix} 23.7 \\ 19.95 \end{pmatrix}$$ **< column means ($Ybar_{.j.}$)**

$Y := Z^{\langle 1 \rangle}$ $Ybar := mean(Y)$ $Ybar = 21.825$ **< Grand Mean ($Ybar_{...}$)**

Sums of Squares:

$I := 2$ **< total number of rows**

$J := 2$ **< total number of columns**

$K := 5$ **< total number of replicates = n**

$$SS_R := J \cdot K \cdot \sum_i \left(YR_{bar_i} - Y_{bar}\right)^2$$ **< Sum of Squares for Rows**

$$SS_C := I \cdot K \cdot \sum_j \left(YC_{bar_j} - Y_{bar}\right)^2$$ **< Sum of Squares for Columns**

$$SS_I := K \cdot \left[\sum_j \sum_i \left(YB_{bar_{i,j}} - YR_{bar_i} - YC_{bar_j} + Y_{bar}\right)^2\right]$$ **< Sum of Squares for Interactions**

$$SS_E := \sum_k \left(a_k - YB_{bar_{0,0}}\right)^2 + \sum_k \left(b_k - YB_{bar_{0,1}}\right)^2 + \sum_k \left(c_k - YB_{bar_{1,0}}\right)^2 + \sum_k \left(d_k - YB_{bar_{1,1}}\right)^2$$

^ Sum of Squares Error (Within)

$$SS_T := \sum_{k=0}^{N-1} \left(Y_k - Y_{bar}\right)^2$$ **< Total Sum of Squares**

Two-Way ANOVA Table:

Source:	SS	df	MS	
Rows	$SS_R = 1386.1125$	$I - 1 = 1$	$MS_R := \frac{SS_R}{I-1}$	$MS_R = 1386.1125$
Columns	$SS_C = 70.3125$	$J - 1 = 1$	$MS_C := \frac{SS_C}{J-1}$	$MS_C = 70.3125$
Interactions	$SS_I = 4.9005$	$(I-1)\cdot(J-1) = 1$	$MS_I := \frac{SS_I}{(I-1)\cdot(J-1)}$	$MS_I = 4.9005$
Within	$SS_E = 366.372$	$I \cdot J \cdot (K-1) = 16$	$MS_E := \frac{SS_E}{I\cdot J\cdot(K-1)}$	$MS_E = 22.8983$
TOTAL	$SS_T = 1827.6975$	$I \cdot J \cdot K - 1 = 19$	$MS_T := \frac{SS_T}{I\cdot J\cdot K - 1}$	$MS_T = 96.1946$

Prototype in R:

COMMANDS:

```
> Z=read.table("c:/2007BiostatsData/ZarExample12.1.txt")
> Z
> C=factor(Z$Sex)
> R=factor(Z$HormTR)
> Y=Z$CaConc
> anova(lm(Y~R*C))
```

< Note that Z$Sex and Z$HormTR are already factors here, so the factor() function has no effect, but is a safe approach anyway.

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
R	**1**	**1386.11**	**1386.11**	**60.5336**	**7.943e-07 *** **
C	**1**	**70.31**	**70.31**	**3.0706**	**0.09886 .**
R:C	**1**	**4.90**	**4.90**	**0.2140**	**0.64987**
Residuals	**16**	**366.37**	**22.90**		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1**

F-Tests in Two-Way ANOVA with Fixed Effects Model:

Model:

$Y_{i,j} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$ **where:**

μ is a constant = grand mean of all objects.
α_i is effect coefficient for classes i in Variable R.
β_j is effect coefficient for classes j in Variable C.
γ_{ij} is interaction coefficient for classes i,j between Variables R and C.
ε_{ijk} is the error term specific to each object i,j,k

Restrictions:

$$\sum_i \alpha_i := 0 \qquad \sum_j \beta_j := 0 \qquad \sum_i \gamma_{ij} := 0 \qquad \sum_j \gamma_{ij} := 0$$

Assumptions:

– ε_{ij} are a random sample ~ N(0,σ^2)
– variance is homogeneous across cells

F-Test for H_0: All $\alpha_i = 0$

Hypotheses:

H_0: $\alpha_i = 0$ *for all* i **< All treatment class deviations from the grand mean are 0**
H_1: *At least one* $\alpha_i <> 0$ **< Two sided test**

Test Statistic:

$$F_1 := \frac{MS_R}{MS_E}$$ **< Ratio of "row" versus "within" Mean Squares**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1 - \alpha)$ $CV := qF[1 - \alpha, (I - 1), I \cdot J \cdot (K - 1)]$ **< Note: df = I-1, IJ(K-1)**

Decision Rule:

IF F_1 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

$P_1 = 1 - \Phi_F(F_1)$ $P_1 := 1 - pF[F_1, I - 1, I \cdot J \cdot (K - 1)]$

F-Test for H_0: All $\beta_j = 0$

Hypotheses:

H_0: $\beta_j = 0$ *for all* j **< All treatment class deviations from the grand mean are 0**
H_1: *At least one* $\beta_j <> 0$ **< Two sided test**

Test Statistic:

$$F_2 := \frac{MS_C}{MS_E}$$ **< Ratio of "column" versus "within" Mean Squares**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1-\alpha)$ $CV := qF[1-\alpha,(J-1),I\cdot J\cdot(K-1)]$ **< Note: df = J-1, IJ(K-1)**

Decision Rule:

IF F_2 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P_2 = 1- $\Phi_F(F_2)$ $P_2 := 1 - pF[F_2, J-1, I\cdot J\cdot(K-1)]$

F-Test for H_0: All $\gamma_{ij} = 0$

Hypotheses:

H_0: γ_{ij} = 0 *for all* ij **< All interactions between the two variables is 0**

H_1: *At least one* γ_{ij} <> 0 **< Two sided test**

Test Statistic:

$F_3 := \frac{MS_I}{MS_E}$ **< Ratio of "interactions" versus "within" Mean Squares**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1-\alpha)$ $CV := qF[1-\alpha,(I-1)\cdot(J-1),I\cdot J\cdot(K-1)]$

Decision Rule:

^ Note: df = (I-1)(J-1), IJ(K-1)

IF F_3 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P_3 = 1- $\Phi_F(F_3)$ $P_3 := 1 - pF[F_3,(I-1)\cdot(J-1),I\cdot J\cdot(K-1)]$

Example: Continuing the Above ANOVA analysis on Sex and Hormone Treatment

F-Tests in Two-Way ANOVA with Fixed Effects Model:

Model:

$Y_{i,j} = \mu + \alpha_i + \beta_j + \gamma_{ij} + \varepsilon_{ijk}$ **where:**

μ is a constant = grand mean of all objects.
α_i is effect coefficient for classes i in Variable R.
β_j is effect coefficient for classes j in Variable C.
γ_{ij} is interaction coefficient for classes i,j between Variables R and C.
ε_{ijk} is the error term specific to each object i,j,k

Restrictions:

$\sum_i \alpha_i := 0$ $\sum_j \beta_j := 0$ $\sum_i \gamma_{ij} := 0$ $\sum_j \gamma_{ij} := 0$

Assumptions:

– ε_{ij} are a random sample ~ N(0,σ^2)
– variance is homogeneous across cells

F-Test for H_0: All $\alpha_i = 0$

Hypotheses:

H_0: $\alpha_i = 0$ *for all* i **< All treatment class deviations from the grand mean are 0**

H_1: *At least one* $\alpha_i <> 0$ **< Two sided test**

Test Statistic:

$$F_1 := \frac{MS_R}{MS_E} \qquad F_1 = 60.5336$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF[1 - \alpha, (I - 1), I \cdot J \cdot (K - 1)]$ $CV = 4.494$

Decision Rule:

IF F_1 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F_1 = 60.5336$ $CV = 4.494$

Probability Value:

$P_1 := 1 - pF[F_1, I - 1, I \cdot J \cdot (K - 1)]$ $P_1 = 7.9431 \times 10^{-7}$

F-Test for H_0: All $\beta_j = 0$

Hypotheses:

H_0: $\beta_j = 0$ *for all* j **< All treatment class deviations from the grand mean are 0**

H_1: *At least one* $\beta_j <> 0$ **< Two sided test**

Test Statistic:

$$F_2 := \frac{MS_C}{MS_E} \qquad F_2 = 3.0706$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF[1 - \alpha, (J - 1), I \cdot J \cdot (K - 1)]$ $CV = 4.494$

Decision Rule:

IF F_2 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F_2 = 3.0706$ $CV = 4.494$

Probability Value:

$P_2 := 1 - pF[F_2, J - 1, I \cdot J \cdot (K - 1)]$ $P_2 = 0.0989$

F-Test for H_0: All $\gamma_i = 0$

Hypotheses:

H_0: $\gamma_{ij} = 0$ *for all* ij **< All interactions between the two variables is 0**

H_1: *At least one* $\gamma_{ij} <> 0$ **< Two sided test**

Test Statistic:

$F_3 := \frac{MS_I}{MS_E}$ $F_3 = 0.214$ **< Ratio of "interactions" versus "within" Mean Squares**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := qF\left[1 - \alpha, (I - 1) \cdot (J - 1), I \cdot J \cdot (K - 1)\right]$ $CV = 4.494$

Decision Rule:

IF F_3 > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F_3 = 0.214$ $CV = 4.494$

Probability Value:

$P_3 := 1 - pF\left[F_3, (I - 1) \cdot (J - 1), I \cdot J \cdot (K - 1)\right]$ $P_3 = 0.6499$

Prototype in R: ANOVA Table from R above:

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
R	**1**	**1386.11**	**1386.11**	**60.5336**	**7.943e-07 *****
C	**1**	**70.31**	**70.31**	**3.0706**	**0.09886 .**
R:C	**1**	**4.90**	**4.90**	**0.2140**	**0.64987**
Residuals	**16**	**366.37**	**22.90**		

Signif. codes: 0 '*' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1 ^ these values match**

ORIGIN ≡ 0

Kruskal-Wallis Test

The Kruskal-Wallis is a non-parametric analog to the One-Way ANOVA F-Test of means. It is useful when the k samples appear not to come from underlying Normal Distributions, or when variance in the different samples are of greatly different magnitudes (non-homogeneous). As with other rank-based tests, it does not have as much power as the fully parametric tests, but nevertheless enjoys wide use. Note that when the number of samples k=2, this test is identical to the Mann-Whitney Test.

Data Structure:

k groups with not necessarily the same numbers of observations and different means.

Let index i,j indicate the ith column (treatment class) and jth row (object).

One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	…	#k
1					
2					
3					
…					
n	n1	n2	n3		nk
means:	Xbar.1	Xbar.2	Xbar.3	…	Xbar.k

Assumptions:

- Observations in each class(block) are a random sample.
- Observations in each block are independent of observations in other class.
- Underlying distribution of observations in each cell are continuous.
- Measurement scale is at least ordinal.

Hypotheses:

H_0: Δ= 0 < No population differences in treatment

H_1: Δ <> 0 < Two Sided Test

Criterion for Normal Approximation:

- IF $n_i \geq 5$ THEN Normal Approximation Applies

  OTHERWISE use Special Tables e.g. Rosner Table 15 p. 844

Normal Approximation:

Rank Data and Sum:

- Pool the data over all treatment classes - Total sample size N = Σn_i
- Assign Data to Ranks. In the case of ties, t observations in a rank are assigned the appropriate average rank.
- Compute the Rank Sum (R_i) for each treatment class i.

Test Statistic:

$$H_S := \frac{12}{N \cdot (N+1)} \cdot \sum_i \frac{(R_i)^2}{n_i} - 3 \cdot (N+1)$$

< where R_i are the Rank sums for each treatment class i

IF no ties, THEN: $H := H_S$ < no correction factor...

OTHERWISE:

correction factor > $\Omega := 1 - \frac{\sum_{j=1}^{g} \left[(t_j)^3 - t_j \right]}{N^3 - N}$

< t represent the number of observations that are tied in groups 1 to g

$$H := \frac{H_S}{\Omega}$$

< Corrected Test Statistic

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \text{inverse}\Phi_{\chi 2}(1-\alpha)$ $C := \text{qchisq}(1-\alpha, k-1)$ **< Note: df = (k-1)**

Decision Rule:

IF H > C THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi 2}(H)\right)$ $P := (1 - \text{pchisq}(H\, k-1))$

Example:

Zar Example 10.11 p. 199: pH was measured multiple times (n_i's) in Four ponds (treatment classes):

$k := 4$ **< treatment classes**

$Z := \text{READPRN}(\text{"c:/2007BiostatsData/ZarExample10.11.txt"})$

$\text{Pond}_0 := Z^{\langle 0 \rangle}$ $n_0 := \text{length}(\text{Pond}_0)$ $n_0 = 8$

$\text{Pond}_1 := Z^{\langle 1 \rangle}$ $n_1 := \text{length}(\text{Pond}_1)$ $n_1 = 8$

$\text{Pond}_2 := Z^{\langle 2 \rangle}$ $n_2 := \text{length}(\text{Pond}_2) - 1$ $n_2 = 7$

$\text{Pond}_3 := Z^{\langle 3 \rangle}$ $n_3 := \text{length}(\text{Pond}_3)$ $n_3 = 8$

$N := \sum n$ $N = 31$ **< total observations**

$$Z = \begin{pmatrix} 7.68 & 7.71 & 7.74 & 7.71 \\ 7.69 & 7.73 & 7.75 & 7.71 \\ 7.7 & 7.74 & 7.77 & 7.74 \\ 7.7 & 7.74 & 7.78 & 7.79 \\ 7.72 & 7.78 & 7.8 & 7.81 \\ 7.73 & 7.78 & 7.81 & 7.85 \\ 7.73 & 7.8 & 7.84 & 7.87 \\ 7.76 & 7.81 & 9999 & 7.91 \end{pmatrix}$$

9999 = missing datapont ^

Assumptions:

- **Observations in each class(block) are a random sample.**
- **Observations in each block are independent of observations in other class.**
- **Underlying distribution of observations in each cell are continuous.**
- **Measurement scale is at least ordinal.**

Hypotheses:

H_0: Δ= 0 **< No population differences in treatment**

H_1: Δ <> 0 **< Two Sided Test**

Criterion for Normal Approximation:

- **IF** $n_i \geq 5$ **THEN Normal Approximation Applies**
  OTHERWISE use Special Tables

$$n = \begin{pmatrix} 8 \\ 8 \\ 7 \\ 8 \end{pmatrix}$$

< so qualifies for Normal Approximation

Normal Approximation:

Rank Data and Sum:

Rank of each observation:

$$\text{Ranks} := \begin{pmatrix} 1 & 6 & 13.5 & 6 \\ 2 & 10 & 16 & 6 \\ 3.5 & 13.5 & 18 & 13.5 \\ 3.5 & 13.5 & 20 & 22 \\ 8 & 20 & 23.5 & 26 \\ 10 & 20 & 26 & 29 \\ 10 & 23.5 & 28 & 30 \\ 17 & 26 & 9999 & 31 \end{pmatrix}$$

$R_0 := \sum \text{Ranks}^{\langle 0 \rangle}$

$R_1 := \sum \text{Ranks}^{\langle 1 \rangle}$

$j := 0 .. n_2 - 1$

$R_2 := \sum_j \text{Ranks}_{j,2}$

$R_3 := \sum \text{Ranks}^{\langle 3 \rangle}$

$$R = \begin{pmatrix} 55 \\ 132.5 \\ 145 \\ 163.5 \end{pmatrix}$$

< Rank Sums for each Pond (Treatment class)

Tied groups:

$$t := \begin{pmatrix} 2 \\ 3 \\ 3 \\ 4 \\ 3 \\ 2 \\ 3 \end{pmatrix} \qquad g := \text{length}(t) \qquad g = 7$$

Test Statistic:

$i := 0 .. k - 1$

$$H_S := \frac{12}{N \cdot (N + 1)} \cdot \sum_i \frac{(R_i)^2}{n_i} - 3 \cdot (N + 1)$$

$H_S = 11.8761$

IF no ties, THEN: $H := H_S$

OTHERWISE:

$$H := \frac{H_S}{1 - \frac{\sum_j \left[(t_j)^3 - t_j\right]}{N^3 - N}}$$

$H = 11.9435$ **< corrected test statistic**

$$\text{sort}\left(\text{stack}\left(Z^{\langle 0 \rangle}, Z^{\langle 1 \rangle}, Z^{\langle 2 \rangle}, Z^{\langle 3 \rangle}\right)\right) = \begin{pmatrix} 7.68 \\ 7.69 \\ 7.7 \\ 7.7 \\ 7.71 \\ 7.71 \\ 7.71 \\ 7.72 \\ 7.73 \\ 7.73 \\ 7.73 \\ 7.74 \\ 7.74 \\ 7.74 \\ 7.74 \\ 7.75 \\ 7.76 \\ 7.77 \\ 7.78 \\ 7.78 \\ 7.78 \\ 7.79 \\ 7.8 \\ 7.8 \\ 7.81 \\ 7.81 \\ 7.81 \\ 7.84 \\ 7.85 \\ 7.87 \\ 7.91 \\ 9999 \end{pmatrix} \qquad \begin{matrix} \textbf{t =} \\ \\ \textbf{< 2} \\ \\ \textbf{< 3} \\ \\ \\ \textbf{< 3} \\ \\ \\ \\ \textbf{< 4} \\ \\ \\ \\ \textbf{< 3} \\ \\ \\ \textbf{< 2} \\ \\ \textbf{< 3} \\ \\ \\ \textbf{^ 7 tied groups} \end{matrix}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I**

$C := \text{qchisq}(1 - \alpha, k - 1)$ $C = 7.8147$

Decision Rule:

IF H > C THEN REJECT H_0 OTHER

$H = 11.9435$ $C = 7.8147$

Probability Value:

$P := (1 - \text{pchisq}(H, k - 1))$ $P = 0.0076$

^ all values confirmed by Zar p. 199

Prototype in R:

COMMANDS:

```
> Z=read.table("c:/2007BiostatsData/ZarExample10.11a.txt",na.strings="NA")
> kruskal.test(Z)
```

Kruskal-Wallis rank sum test

data: ZZ

Kruskal-Wallis chi-squared = 11.9435, df = 3, p-value = 0.007579 < Values match

ORIGIN ≡ 0

Single and Multiple Simultaneous Confidence Intervals in ANOVA Tests

Similar to previous statistical t-Tests, Confidence Intervals may be specified to indicate values of the test statistic in comparison with Critical Values (derived from the inverse cumulative probability t function qt) over which H_0 will *not* be rejected, or equivalently, values of probability greater than a previously specified α. In using ANOVA, however, an important complication arises. In two population t-Tests, only a single comparison between population means (μ_1 with μ_2) is made. In ANOVA, greater than two populations is standard and multiple pairwise or linear contrast comparisons (for instance μ_1 with μ_2 *and* μ_1 with μ_3 *and* μ_2 with μ_3 for three populations) are often of interest. In most cases, these comparisions are made simultaneously, and are therefore dependent upon the same sample data. The existence of multple dependent probabilities derived from each comparison implies that the joint probabity of a *family of comparisons* together is greater than *each one separately*. Thus, if one specifies $\alpha = 0.05$ for one one interval (or test) then *familywise* α for all together is always greater (i.e., less significant).

Multiple t-Test / Fisher's LSD Test for Specific Treatment Pairs:

Model:

$X_{i,j} = \mu + \alpha_i + \varepsilon_{i,j}$ < where:

μ is the grand mean of all objects.

α_i is the mean of i = $\mu + \alpha_i$ for each class i.

$\varepsilon_{i,j}$ is the error term specific to each object i,j

Restriction:

$\sum_i n_i \cdot \alpha_i := 0$ < allows estimation of k parameters.

Other restrictions are also possible:

$\sum_i \alpha_i := 0$ or $\alpha_k := 0$ < See Rosner p. 558

Assumptions:

ε_{ij} are a homogeneous random sample ~ $N(0,\sigma^2)$

One-Way ANOVA Table:

Source:	SS	df	MS
Between	SS_B	$k - 1$	$\frac{SS_B}{k-1}$
Within	SS_W	$n - k$	$\frac{SS_W}{n-k}$
TOTAL	SS_T		

Hypotheses:

H_0: $\alpha_i = \alpha_j$ *for specific* i & j < Means in treatment classes i & j are the same as grand mean

H_1: $\alpha_i <> \alpha_j$ *for specific* i & j < Two sided test

Test Statistic:

$$t := \frac{X_{bar_i} - X_{bar_j}}{\sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)}}$$

< Normalized distance between mean of class i & j

Critical Value of the Test:

$\alpha := 0.05$ < Probability of Type I error must be explicitly set

$C_1 := inverse\Phi_t\left(\frac{\alpha}{2}\right)$ $C_2 := inverse\Phi_t\left(1 - \frac{\alpha}{2}\right)$

$C_1 := qt\left(\frac{\alpha}{2}, n - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2}, n - k\right)$

Note degrees of freedom = (n-k)

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum(2 $\Phi_t(t)$,1-2 $\Phi_t(t)$

$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,]$

Confidence Interval:

$$CI := \left[X_{bar_i} - X_{bar_j} + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \quad X_{bar_i} - X_{bar_j} + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \right]$$

^ Note that C_1 & C_2 are explicitly evaluated above, so added to the difference in sample means here.

Example: Vital Capacity dataset by group

R COMMANDS:

```
> V=read.table("c:/2007BiostatsData/vital.txt")
> attach(V)
> X1=vital.capacity[group=="1"]
> X2=vital.capacity[group=="2"]
> X3=vital.capacity[group=="3"]
> summary(X1)
> length(X1)
> summary(X2)
> length(X2)
> summary(X3)
> length(X3)
> Y=vital.capacity
> X=factor(group)
> anova(lm(Y~X))
```

$k := 3$ **< number of classes = groups**

Min. 1st Qu. Median Mean 3rd Qu. Max.
2.700 2.955 3.865 3.949 4.737 5.520

[1] 12 $n_1 := 12$ $X_{bar_1} := 3.949$ **< number of objects & mean of X1**

Min. 1st Qu. Median Mean 3rd Qu. Max.
2.700 4.240 4.615 4.472 5.062 5.220

[1] 28 $n_2 := 28$ $X_{bar_2} := 4.472$ **< number of objects & mean of X2**

Min. 1st Qu. Median Mean 3rd Qu. Max.
3.030 4.010 4.530 4.462 4.902 5.860

[1] 44 $n_3 := 44$ $X_{bar_3} := 4.462$ **< number of objects & mean of X3**

$n := n_1 + n_2 + n_3$ $n = 84$

One-Way ANOVA Table:

From R:

Analysis of Variance Table

Response: Y

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
X	2	2.747	1.374	2.4785	0.09021 .
Residuals	81	44.894	0.554		

--

$MS_W := 0.554$ **< MS Residuals**

Hypotheses:

H_0: $\alpha_i = \alpha_j$ *for specific* i & j < Means in treatment classes i & j are the same as grand mean

H_1: $\alpha_i <> \alpha_j$ *for specific* i & j < Two sided test

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2}, n-k\right)$ $C_1 = -1.9897$ $C_2 := qt\left(1-\frac{\alpha}{2}, n-k\right)$ $C_2 = 1.9897$

Single Comparisons:

Between populations 1 & 2:

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_2}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}} \qquad t = -2.0365 \qquad X_{bar_1} - X_{bar_2} = -0.523$$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -2.0365$ $C_1 = -1.9897$ $C_2 = 1.9897$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n-k), 2 \cdot (1 - pt(t, n-k))\,]$ $P = 0.045$

Confidence Interval:

$$CI_{12} := \left[\left(X_{bar_1} - X_{bar_2}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \quad \left(X_{bar_1} - X_{bar_2}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \right]$$

$CI_{12} = (-1.034 \quad -0.012)$

Between populations 1 & 3:

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_3}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)}} \qquad t = -2.1163 \qquad X_{bar_1} - X_{bar_3} = -0.513$$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -2.1163$ $C_1 = -1.9897$ $C_2 = 1.9897$

Probability Value:

$P := \min[\, 2 \cdot pt(t, n-k), 2 \cdot (1 - pt(t, n-k))\,]$ $P = 0.0374$

Confidence Interval:

$$CI_{13} := \left[\left(X_{bar_1} - X_{bar_3}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)} \quad \left(X_{bar_1} - X_{bar_3}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)} \right]$$

$CI_{13} = (-0.9953 \quad -0.0307)$

Between populations 2 & 3:

Test Statistic:

$$t := \frac{X_{bar_2} - X_{bar_3}}{\sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)}} \qquad t = 0.0556 \qquad X_{bar_2} - X_{bar_3} = 0.01$$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 0.0556$ $C_1 = -1.9897$ $C_2 = 1.9897$

Probability Value:

$$P := \min[\, 2 \cdot pt(t, n - k), 2 \cdot (1 - pt(t, n - k))\,] \qquad P = 0.9558$$

Confidence Interval:

$$CI_{23} := \left[\left(X_{bar_2} - X_{bar_3}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)} \quad \left(X_{bar_2} - X_{bar_3}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)} \right]$$

$CI_{23} = (-0.348 \quad 0.368)$

Note that these are Separate and Single Confidence Intervals. Considered as a joint statement of probability, the *familywise probability* of Type I error α is potentially much higher. From the mathematical end of things, statisticians routinely caution experimenters about the potential pitfalls of "data snooping" (to borrow a term from Neter et al. 1996). By this, they mean running a large number of simultaneous tests or confidence intervals, and then proceeding to report significant findings as if discovered outside the context of the others, or worse, as the result of a strategically-chosen *a priori* experimental design. The problem is that if enough simultaneous tests are run, the laws of probability predict that some tests will end up showing significance merely due to chance. This mathematically-based caution is certainly correct. Given this, are more than one significant planned or unplanned result in ANOVA tests to be considered valid or not? Much depends on what exactly is meant by the foundational concept of α in often widely differing theoretical and experimental contexts. Whereas mathematicians might like to draw a bright line between *a priori* and *post hoc*, in experimental practice rarely is the distinction so clear. All experiments exist within a framework of pre-existing literature and laboratory/field practice for data collection. So *of course* biologists regularly engage in "data snooping" in conceiving of problems, designing studies and analyzing results. They could hardly do otherwise...

In my opinion, the a *priori* vs *post hoc* distinction is of interest from both theoretical and practical standpoints, and to be aware of the issues involved makes it possible to construct stronger scientific arguments. The distinction also points to clear limitations in statistical reasoning in the sciences to the extent that all of it must be acknowledged to be nothing more than an approximation. If one's data are overwealmingly clear, then difficulties in the approximation don't really matter. However, if the data are unclear, then how one employs the approximation may influence what one might say within a test, but not necessarily what one might conclude. The take-home message remains the same - the data remain unclear, and biological interpretation, and experimental replication, must necessarily take precedence over mathematical methodology.

Simultaneous Inference Procedures:

Several methods have been developed to adjust Probabilities of Tests and associated Confidence Interval widths to accomodate familywise assessments. Some methods explicitly permit "data snooping" whereas others do not. It will be beyond the scope of this course to worry about how these adjustments are calculated, but it is important to be aware of how, and under what circumstances, each procedure is employed. As a practical matter, of course, standard statistical packages offer a full battery of possibilites and if the data permits, use of the "most conservative" (i..e, widest confidence intervals) is often considered evidence of good experimental design.

Multiple t-Test / Fisher's LSD Test for Specific Treatment Pairs:

Although described above in the context of single tests, in fact, Fisher's LSD (Least Significant Difference) Tests is often available as one of the available "multiple test" options in standard statistical packages. It is useful to know that they are the same. Fisher's LSD is often employed when the researcher feels that "data snooping" is not a major issue in the study and/or the number of multiple comparisions are relatively low. Of course, this is a judgement call. So if the data permits, use of one of the procedures below is more "conservative" and is often judged to be more prudent. Many studies report both.

Prototype in SYSTAT:

Data cut & pasted from Excel to a SYSTAT Datasheet. Dependent Variable was named 'VC' and Independent categorical variable named "GROUP". ANOVA option chosen and variables assigned. Posthoc tests turned on with LSD as option.

```
          Effects coding used for categorical variables in model.
           Categorical values encountered during processing are:
                                GROUP (3 levels)
                                   1,        2,        3
  Dep Var: VC   N: 84   Multiple R: 0.24014   Squared multiple R: 0.05767
                                Analysis of Variance
  Source             Sum-of-Squares   df  Mean-Square     F-ratio       P
 GROUP                     2.74734     2      1.37367     2.47846     0.0902
 Error                    44.89362    81      0.55424


                        COL/
                       ROW GROUP
                          1  1
                          2  2
                          3  3
              Using least squares means.
                  Post Hoc test of VC
         -Using model MSE of 0.554 with 81 df.
          Matrix of pairwise mean differences:
                          1           2           3
              1        0.00000
              2        0.52262     0.00000                 < differences match above
              3        0.51288    -0.00974     0.00000

       Fisher's Least-Significant-Difference Test.
      Matrix of pairwise comparison probabilities:
                          1           2           3
              1        1.00000
              2        0.04516     1.00000                 < Probabilities match above
              3        0.03747     0.95697     1.00000
```

Bonferroni Multiple Comparisons Procedure:

If a specific and relatively small set of simultaneous tests are desired, this procedure will often give the narrowest confidence intervals, and is preferred. Since the Bonferroni method requires identifying a specific set of simultaneous tests, it is not appropriate for "data snooping".

Methodology:

Bonferroni intervals can be easily calculated given g - the number of simultaneous tests:

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C_1 := qt\left(\frac{\alpha}{2 \cdot g}, n - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2 \cdot g}, n - k\right)$ **< critical values modified to account for number of tests g**

Bonferroni Confidence Interval for Multiple Comparisons:

$$CI_B := \left[X_{bar_i} - X_{bar_j} + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \quad X_{bar_i} - X_{bar_j} + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \right]$$

^ same as for Single CI but with adjusted Critical Values

Example: Data from above.

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$g := 3$ **< Three tests set explicitly (1-2, 1-3, 2-3)**

$C_1 := qt\left(\frac{\alpha}{2 \cdot g}, n - k\right)$ $C_2 := qt\left(1 - \frac{\alpha}{2 \cdot g}, n - k\right)$ **< critical values modified to account for number of tests g**

Between populations 1 & 2:

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_2}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$ $t = -2.0365$ $X_{bar_1} - X_{bar_2} = -0.523$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -2.0365$ $C_1 = -2.4447$ $C_2 = 2.4447$

Probability Value:

$P := \min[\, 2 \cdot g \cdot pt(t, n - k), 2 \cdot g \cdot (1 - pt(t, n - k))\,]$ $P = 0.134899$

Confidence Interval:

$$CI_{12} := \left[\left(X_{bar_1} - X_{bar_2}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \quad \left(X_{bar_1} - X_{bar_2}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)} \right]$$

$CI_{12} = (-1.1508 \quad 0.1048)$

Between populations 1 & 3:

Test Statistic:

$$t := \frac{X_{bar_1} - X_{bar_3}}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)}} \qquad t = -2.1163 \qquad X_{bar_1} - X_{bar_3} = -0.513$$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = -2.1163$ $C_1 = -2.4447$ $C_2 = 2.4447$

Probability Value:

$P := \min[\, 2 \cdot g \cdot pt(t, n - k), 2 \cdot g \cdot (1 - pt(t, n - k))\,]$ $P = 0.1122$

Confidence Interval:

$$CI_{13} := \left[\left(X_{bar_1} - X_{bar_3}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)} \quad \left(X_{bar_1} - X_{bar_3}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_3}\right)} \right]$$

$CI_{13} = (-1.1056 \quad 0.0796)$

Between populations 2 & 3:

Test Statistic:

$$t := \frac{X_{bar_2} - X_{bar_3}}{\sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)}} \qquad t = 0.0556 \qquad X_{bar_2} - X_{bar_3} = 0.01$$

Decision Rule:

IF |t| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$t = 0.0556$ $C_1 = -2.4447$ $C_2 = 2.4447$

Probability Value:

$P := \min[\, 2 \cdot g \cdot pt(t, n - k), 2 \cdot g \cdot (1 - pt(t, n - k))\,]$ $P = 2.8675$

Confidence Interval:

$$CI_{13} := \left[\left(X_{bar_2} - X_{bar_3}\right) + C_1 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)} \quad \left(X_{bar_2} - X_{bar_3}\right) + C_2 \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_2} + \frac{1}{n_3}\right)} \right]$$

$CI_{13} = (-0.4299 \quad 0.4499)$

Prototype in SYSTAT:

Bonferroni Adjustment.
Matrix of pairwise comparison probabilities:

		1	2	3
	1	1.00000		
values are close but don't exactly match >	2	0.13549	1.00000	
	3	0.11241	1.00000	1.00000

Tukey Multiple Comparisons Procedure:

This procedure is designed to provide a simultaneous probability of α when comparing means of all possible pairs of populations within the ANOVA data structure. When sample sizes n_i differ, this procedure is also called the Tukey-Kramer Procedure. "Data snooping" is permitted with this procedure as long as one is restricts "snooping" to pairwise comparisons of population means.

Methodology:

Tukey intervals are calculated by consulting a *studentized range distribution.*

Tukey Test Statistic:

$$Q := \frac{\sqrt{2} \cdot \left(X_{bar_i} - X_{bar_j}\right)}{\sqrt{MS_W \cdot \left(\frac{1}{n_1} + \frac{1}{n_2}\right)}}$$

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \frac{1}{\sqrt{2}} \cdot qstudentizedrange(1 - \alpha, k, n - k)$ **< critical value constructed from "studentized" range distribution.**

Decision Rule:

IF |Q| > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

$P := \min(pstudentizedrange(Q, k, n - k))$

Tukey Confidence Interval for Multiple Comparisons:

$$CI_T := \left[X_{bar_i} - X_{bar_j} - C \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \quad X_{bar_i} - X_{bar_j} + C \cdot \sqrt{MS_W \cdot \left(\frac{1}{n_i} + \frac{1}{n_j}\right)} \right]$$

Example: Data from above.

Output from SYSTAT:

```
                Post Hoc test of VC

     Using model MSE of 0.554 with 81 df.
     Matrix of pairwise mean differences:

              1          2          3
1       0.00000
2       0.52262    0.00000
3       0.51288   -0.00974    0.00000

        Tukey HSD Multiple Comparisons.
Matrix of pairwise comparison probabilities:

              1          2          3
1       1.00000
2       0.11049    1.00000
3       0.09304    0.99839    1.00000
```

Scheffé Multiple Comparisons Procedure:

This procedure is designed to provide a simultaneous probability of α for all possible linear contrasts within the ANOVA data structure. Since all possible Linear Constrasts in a dataset involves an infinite set of possible comparisons including pairwise comparisons, the Tukey procedure will typically give smaller Confidence Intervals for only pairwise comparisons, and the Bonferroni procedure will give smaller Confidence Intervals, for a specific limited set of any kind of comparisons. Thus the Scheffé is a conservative approach that allows "data snooping" and is often preferred for methodological reasons - if the data will permit it. Often the data does not. In using this test, many researchers relax the criterion of "acceptable" *familywise* Type I error α a little ($\alpha = 0.1$ is often considered acceptable for multiple comparisons when $\alpha = 0.05$ is considered acceptable for single comparisons).

Methodology:

Scheffé intervals are calculated by constructing an unbiased point estimate of the mean of a Linear Combination of interest L_{hat}, standard deviation s_L, and Critical Values calculated from the F distribution.

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$S := \sqrt{(n-1)\cdot qF(1-\alpha, g-1, N-g)}$ **< where g is the number of Populations in the ANOVA data structure, and N is the total number of observations Σn_i.**

Scheffé Confidence Interval for Multiple Comparisons:

$$CI_S := \left[L_{hat} - S \cdot \left[MS_W \cdot \sum_{i=1}^{g} \frac{(c_i)^2}{n_i} \right] \quad L_{hat} + S \cdot \left[MS_W \cdot \sum_{i=1}^{g} \frac{(c_i)^2}{n_i} \right] \right]$$

^ where Mean Squares Within (Error) is modified by coefficients c_i squared for the Linear combination and sample sizes n_i.

If the ANOVA F-Test for H_0: All $\alpha_i = 0$ rejects H_0 then the Scheffé Procedure is guaranteed to find at least one contrast such that H_0: $L_i = 0$ is also rejected.

Output from SYSTAT:

```
Post Hoc test of VC

Using model MSE of 0.554 with 81 df.
Matrix of pairwise mean differences:

          1          2          3
1     0.00000
2     0.52262    0.00000
3     0.51288   -0.00974    0.00000

          Scheffe Test.
Matrix of pairwise comparison probabilities:

          1          2          3
1     1.00000
2     0.13284    1.00000
3     0.11329    0.99854    1.00000
```

Holm Simultaneous Testing Procedure:

This procedure is an iterative refinement of the Bonferroni approach designed to provide a simultaneous probability of α for a specific set of tests. Holm sometimes rejects a null hypothesis that Bonferroni would not with the same data and is, thus, more powerful. However, Holm is computationally more complex and lacks direct computation of Confidence Intervals. Although Holm may be the preferred method for theoretical reasons, power consideration by itself may not necessarily be a good reason for chosing the test. As with Bonferroni, this method is unsuitable for "data snooping".

ORIGIN ≡ 0

Repeated Measures One-Way Analysis of Variance with Fixed Effects Model

As indicated previously, One-Way ANOVA with Fixed Effects Model (also termed "Single Factor" and "Between Groups" ANOVA) represents an extension of the Two-Sample t-Test with equal variance to analyses involving $k \geq 2$ groups (often termed "treatments" or "factor levels"). The ANOVA extension of the Paired t-Test, in which data are matched exactly across groups ("treatments" or "factor levels"), are called Repeated One-Way ANOVA designs (also termed "Within-Subjects" Single-Factor ANOVA). They are also sometimes called "Radomized Block" studies emphasizing the importance of proper experimental design in the presentation of treatments to multiple individuals ("objects" or "replicates") within the study. Such concerns were also present in the Paired t-Test but become much more so in Repeated-Measures ANOVA.

Data Structure:

k groups (treatments) exactly matched within individuals (objects). Typically, the order in which specific treatments are presented to individuals is randomized and exactly matched over the n replicates.

Let index i,j indicate the ith column (treatment class) and jth row (object).

Repeated Measures One-Way ANOVA					
	Treatment Classes:				
Objects (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n	n	n		n
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

Model:

$$X_{i,j} = \mu + \rho_j + \alpha_i + \varepsilon_{i,j}$$

< where:

μ is the grand mean of all objects.
ρ_j is a random effect for each object j
α_i is a constant effect for each class i.
$\varepsilon_{i,j}$ is the error term specific to each object i,j

Restriction:

$$\sum_i \alpha_i := 0$$

< allows estimation of k parameters.

Assumptions:

ρ_j are a random sample ~ N(0, σ_ρ^2)

ε_{ij} are a random sample ~ N(0,σ^2)

ρ_j and ε_{ij} are independent.

Number & Means:

n **< total number of objects = matched observations**

$N := n \cdot k$ **< Total number of observations**

$$GM := \frac{1}{N} \cdot \left(\sum_i \sum_j X_{i,j} \right)$$

< grand mean - sample estimate of μ

$$Xbar_i := mean\left(X^{\langle i \rangle}\right)$$

< means for individuals across treatments

$$Xbar_j := mean\left[\left(X^T\right)^{\langle i \rangle}\right]$$

< means for treatments across individuals (using matrix transpose function)

Sums of Squares:

$$SS_{TOT} := \sum_i \sum_j \left(X_{i,j} - GM\right)^2$$ **< Total Sum of Squares**

$$SS_I := k \cdot \sum_j \left(Xbar_j - GM\right)^2$$ **< Sums of Squares for Individuals (Objects or Subjects)**

$$SS_T \cdot \sum_i \left(Xbar_i - GM\right)^2$$ **< Sums of Squares for Treatments**

$$SS_E := \sum_i \sum_j \left(X_{i,j} - Xbar_i - Xbar_j + GM\right)^2$$ **< Between (Treatment) Sum of Squares**

Repeated Measures One-Way ANOVA Table:

Source:	SS	df	MS
Individuals	SS_I	$n - 1$	$\frac{SS_B}{n-1}$
Treatment	SS_T	$k - 1$	$\frac{SS_W}{k-1}$
Error	SS_E	$(k-1)\cdot(n-1)$	$\frac{SS_W}{(k-1)\cdot(n-1)}$
TOTAL	SS_{TOT}		

Example:

WineTest.txt Data in this week's Data folder -
6 Judges each rate 4 wines in a taste test (Neter et al p 1169):

W := READPRN("c:/2007BiostatsData/WineTest.txt")

$$W = \begin{pmatrix} 1 & 20 & 24 & 28 & 28 \\ 2 & 15 & 18 & 23 & 24 \\ 3 & 18 & 19 & 24 & 23 \\ 4 & 26 & 26 & 30 & 30 \\ 5 & 22 & 24 & 28 & 26 \\ 6 & 19 & 21 & 27 & 25 \end{pmatrix}$$

X := submatrix(W, 0, 5, 1, 4) **< extracting only the data**

Number & Means:

n := rows(X) k := cols(X) i := 0 .. cols(X) − 1 j := 0 .. rows(X) − 1

N := n · k

$$GM := \frac{1}{N} \cdot \left(\sum_i \sum_j X_{j,i}\right)$$

n = 6
k = 4

GM = 23.6667

$$X = \begin{pmatrix} 20 & 24 & 28 & 28 \\ 15 & 18 & 23 & 24 \\ 18 & 19 & 24 & 23 \\ 26 & 26 & 30 & 30 \\ 22 & 24 & 28 & 26 \\ 19 & 21 & 27 & 25 \end{pmatrix}$$

$XbarT_i := mean\left(X^{\langle i \rangle}\right)$

$XbarI_j := mean\left[\left(X^T\right)^{\langle j \rangle}\right]$

$$XbarT = \begin{pmatrix} 20 \\ 22 \\ 26.6667 \\ 26 \end{pmatrix}$$

$$XbarI = \begin{pmatrix} 25 \\ 20 \\ 21 \\ 28 \\ 25 \\ 23 \end{pmatrix}$$

< treatment means for each individual

individual means for each treatment ^

Sums of Squares:

$$SS_{TOT} := \sum_i \sum_j \left(X_{j,i} - GM\right)^2 \qquad SS_{TOT} = 373.3333$$

$$SS_I := k \cdot \sum_j \left(XbarI_j - GM\right)^2 \qquad SS_I = 173.3333$$

$$SS_T := \sum_i n \cdot \left(XbarT_i - GM\right)^2 \qquad SS_T = 184$$

$$SS_E := \sum_i \sum_j \left(X_{j,i} - XbarT_i - XbarI_j + GM\right)^2 \qquad SS_E = 16$$

Repeated Measures One-Way ANOVA Table:

Source:	SS	df	MS	
Individuals	$SS_I = 173.3333$	$n - 1$	$MS_I := \frac{SS_I}{n-1}$	$MS_I = 34.6667$
Treatment	$SS_T = 184$	$k - 1$	$MS_T := \frac{SS_T}{k-1}$	$MS_T = 61.3333$
Error	$SS_E = 16$	$(k-1)\cdot(n-1)$	$MS_E := \frac{SS_E}{(k-1)\cdot(n-1)}$	$MS_E = 1.0667$
TOTAL	$SS_{TOT} = 373.3333$			

^ values confirmed Neter et al. p. 1171

F Test for Overall Comparison of Class Means:

Hypotheses:

H_0: $\alpha_i = 0$ *for all* i **< All treatment class deviations from the grand mean are 0**

H_1: *At least one* $\alpha_i <> 0$ **< Two sided test**

Test Statistic:

$$F := \frac{MS_T}{MS_E}$$

< Ratio of "treatment" versus "error" Mean Squares

Distribution of the test Statistic F:

If H_0 is true then F ~F((k-1),(k-1)(n-1)) **where: k = number of classes**
n = number of individuals

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$CV := \text{inverse}\Phi_F(1-\alpha)$ $CV := qF[1-\alpha,(k-1),(k-1)\cdot(n-1)]$

Decision Rule:

IF F > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

P = minimum($\Phi_F(F)$,1-$\Phi_F(F)$)

$P := \min[\, pF[F,(k-1),(n-1)],1 - pF[F,(k-1),(N-1)]\,]$

^ Note that C_1 & C_2 are explicitly evaluated above, so added to the difference in sample means here.

Example:

Continuing our Example from Above...

F Test for Overall Comparison of Class Means:

Hypotheses:

H_0: α_i = 0 *for all* i **< All treatment class deviations from the grand mean are 0**

H_1: *At least one* α_i <> 0 **< Two sided test**

Test Statistic:

$F := \frac{MS_T}{MS_E}$ $F = 57.5$ **< confirmed Neter et al. p. 1170**

Critical Value of the Test:

$\alpha := 0.01$ **< Probability of Type I error must be explicitly set**

$CV := qF\left[1-\alpha,(k-1),(k-1)\cdot(n-1)\right]$ $CV = 5.417$

Decision Rule:

IF F > C, THEN REJECT H_0 OTHERWISE ACCEPT H_0

$F = 57.5$ $CV = 5.417$ **< values confirmed Neter et al. p. 1170**

Probability Value:

$P := \min[\, pF[F,(k-1),(n-1)],1 - pF[\,F,(k-1),(k-1)\cdot(n-1)\,]\,]$ $P = 1.8538 \times 10^{-8}$

ORIGIN ≡ 0

Friedman Two-Way Analysis of Variance by Ranks Test

The Friedman Two-Way ANOVA by Ranks Test is the non-parametric analog to the One-Way Repeated Measures ANOVA. The object here is to compare observations exactly matched across treatment classes for replicate individuals.

Data Structure:

k groups (treatments) exactly matched within individuals (objects). Typically, the order in which specific treatments are presented to individuals is randomized and exactly matched over the n replicates.

Let index i,j indicate the ith column (treatment class) and jth row (object). $X_{i,j}$ represents the rank or average rank of the treatment for each individual.

Friedman's Two-Way ANOVA by Ranks					
	Treatment Classes:				
Individuals (Replicates)	#1	#2	#3	...	#k
1					
2					
3					
...					
n	n	n	n		n
means:	Xbar.1	Xbar.2	Xbar.3	...	Xbar.k

Assumptions:

- **The n Individuals represent a random sample.**
- **Underlying distribution of observations in treatment cells are continuous.**
- **Observations are of at least ordinal scale.**

Hypotheses:

H_0: Δ= 0 < **No population differences in treatment**

H_1: Δ <> 0 < **Two Sided Test**

Criterion for Approximation:

- **IF** $n_i \geq 8$ **THEN Approximation Applies OTHERWISE the test is conservative.**

Rank Data and Sum:

- **n = number of individuals, k = number of treatment classes**
- **Assign Data for treatment class to a Ranks considering each Individual. In the case of ties, t observations in a rank are assigned the appropriate average rank.**
- **Compute the Rank Sum (R_i) for each treatment class i.**

Test Statistic:

$$Fr_S := \frac{12}{n \cdot k \cdot (k+1)} \cdot \left[\sum_i \left(R_i\right)^2 \right] - 3 \cdot n \cdot (k+1)$$

< **where R_i are the Rank sums for each treatment class i**

IF no ties, THEN: $Fr := Fr_S$ < **no correction factor...**

OTHERWISE:

$$Fr := \frac{Fr_S}{1 - \frac{\sum_{j=1}^{g} \left[\left(t_j\right)^3 - t_j \right]}{n \cdot \left(k^3 - k\right)}}$$

correction factor >

< **t represent the number of observations that are tied in groups 1 to g**

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := \text{inverse}\Phi_{\chi 2}(1-\alpha)$ $C := \text{qchisq}(1-\alpha, k-1)$ **< Note: df = (k-1)**

Decision Rule:

IF Fr > C THEN REJECT H_0 OTHERWISE ACCEPT H_0

Probability Value:

$P := \left(1 - \Phi_{\chi 2}(H)\right)$ $P := (1 - \text{pchisq}(Fr\ k - 1))$

Example:

Sheskin Table 19.1 p. 455 6 Individuals (Subjects) ranked for three Treatments (Conditions) :

$Z := \text{READPRN}(\text{"c:/2007BiostatsData/Sheskin.txt"})$

$X := \text{submatrix}(Z, 0, 5, 1, 3)$

$n := \text{length}\left(X^{\langle 0 \rangle}\right)$ $n = 6$

$k := \text{length}\left[\left(X^{T}\right)^{\langle 0 \rangle}\right]$ $k = 3$

$$X = \begin{pmatrix} 9 & 7 & 4 \\ 10 & 8 & 7 \\ 7 & 5 & 3 \\ 10 & 8 & 7 \\ 7 & 5 & 2 \\ 8 & 6 & 6 \end{pmatrix} \qquad Z = \begin{pmatrix} 1 & 9 & 7 & 4 \\ 2 & 10 & 8 & 7 \\ 3 & 7 & 5 & 3 \\ 4 & 10 & 8 & 7 \\ 5 & 7 & 5 & 2 \\ 6 & 8 & 6 & 6 \end{pmatrix}$$

Assumptions:

- **The n Individuals represent a random sample.**
- **Underlying distribution of observations in treatment cells are continuous.**
- **Observations are of at least ordinal scale.**

Hypotheses:

H_0: $\Delta = 0$ **< No population differences in treatment**

H_1: $\Delta <> 0$ **< Two Sided Test**

Criterion for Approximation:

- **IF $n_i \geq 8$ THEN Approximation Applies OTHERWISE the test is conservative.**

Rank Data and Sum:

$n = 6$ $k = 3$

$$X = \begin{pmatrix} 9 & 7 & 4 \\ 10 & 8 & 7 \\ 7 & 5 & 3 \\ 10 & 8 & 7 \\ 7 & 5 & 2 \\ 8 & 6 & 6 \end{pmatrix} \qquad X_R := \begin{pmatrix} 3 & 2 & 1 \\ 3 & 2 & 1 \\ 3 & 2 & 1 \\ 3 & 2 & 1 \\ 3 & 2 & 1 \\ 3 & 1.5 & 1.5 \end{pmatrix}$$

< rankings are determined by numerical values of treatments seen for each individual separately.

$i := 0 .. k - 1$

$R_i := \sum X_R^{\langle i \rangle}$

$$R = \begin{pmatrix} 18 \\ 11.5 \\ 6.5 \end{pmatrix}$$

< rank sums for each column of X_R

Test Statistic:

$$Fr_s := \frac{12}{n \cdot k \cdot (k+1)} \cdot \left[\sum_i \left(R_i\right)^2\right] - 3 \cdot n \cdot (k+1) \qquad Fr_s = 11.0833$$

^ confirmed Sheskin p. 456

IF no ties, THEN: $Fr := Fr_s$ **< no correction factor...**

$g := 1 \qquad t_1 := 2$

OTHERWISE:

$$Fr := \frac{Fr_S}{1 - \frac{\sum_{j=1}^{g} \left[\left(t_j\right)^3 - t_j \right]}{n \cdot \left(k^3 - k\right)}}$$

correction factor >

< t represent the number of observations that are tied in groups 1 to g

$Fr = 11.5652$

^ Fr and correction factor confirmed Sheskin p. 457

Critical Value of the Test:

$\alpha := 0.05$ **< Probability of Type I error must be explicitly set**

$C := qchisq(1 - \alpha, k - 1)$ $C = 5.9915$

Decision Rule:

IF Fr > C THEN REJECT H_0 OTHERWISE ACCEPT H_0

$Fr = 11.5652$ $C = 5.9915$

Probability Value:

$P := (1 - pchisq(Fr, k - 1))$ $P = 0.0031$

Prototype in R:

COMMANDS:

```
X=read.table("c:/2007BiostatsData/Sheskin.txt")
X
Y=as.matrix(X)
Y
friedman.test(Y)
```

Friedman rank sum test

data: Y
Friedman chi-squared = 11.5652, df = 2, p-value = 0.003081

^ values the same as the corrected version Fr above.