ORIGIN ≡ 0

Probability Distributions

W. Stein

Statistics is based upon comparison of measurements collected from one or more limited samples with expected values characterizing the underlying population from which the samples have been drawn. In most cases, comparisons are made with one of a handful of reference distributions with properties that have been worked out.

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

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

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.**

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. In working with statistical tests, the classical way to estimate P(x) from x or the inverse was to consult obligatory tables. With the advent of the microcomputers, this is now much more efficiently handled by standard functions of four different types: d, p, q, and r.

Prototype in R:

COMMANDS:

```
? dnorm()
? dt()
? dchisq()
? df()
? dbinom()
```

Returns information about how to call standard d,p,q,r functions, e.g. for Normal Distribution:

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)
```

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 underlie many statistical methods.

$n := 50 \qquad i := 0 .. n$

$b := -\left(\frac{1}{2} \cdot n\right) \qquad c := 0.1$

$x_i := c \cdot (i + b)$

$\mu := 0 \qquad \sigma := 1$

$EN_A := dnorm(x, \mu, \sigma)$

$EN_B := dnorm[x, \mu, (2\sigma)]$

$EN_C := dnorm[x, \mu, (0.5\sigma)]$

$EN_D := dnorm[x, (\mu + 1), \sigma]$

$EN_E := dnorm[x, (\mu - 1), (0.5 \cdot \sigma)]$

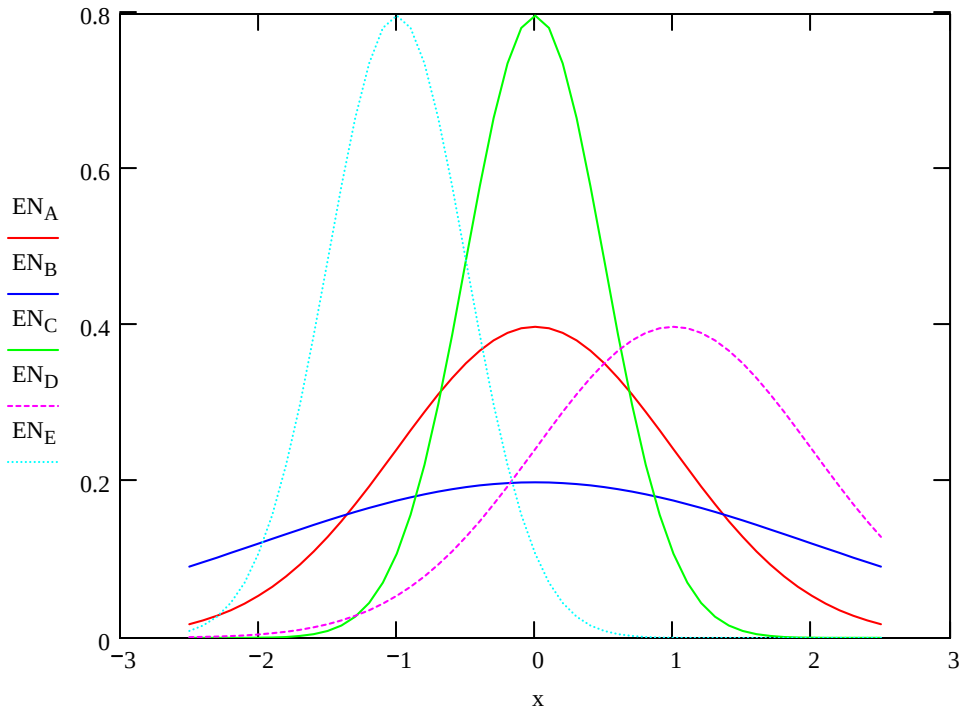
< Out of all possible values, we will arbitrarily look at a set of n points. At the scale we plot things here, this might as well be continuous...

< Arbitrary scaling factors so we can see things in the plot.

< Individual scaled values we plot on our x axis below.

< parameters of the standard normal curve where μ is the mean of the distribution and σ is the standard deviation

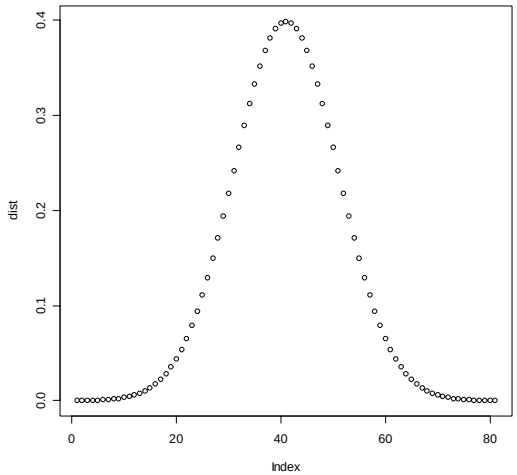
<- The Normal distribution defines a family of curves with different values of μ and σ.



Prototype in R:

COMMANDS:

```
# NORMAL DISTRIBUTION:
# SETS UP VARIABLE X AS A RANGE
x=seq(-4,4,0.1)
# SPECIFY MEAN (mu) &
# STANDARD DEVIATION (sigma)
mu=0
sigma=1
dist=dnorm(x,mu,sigma)
plot(dist, type="p")
```



Student's t Distribution:

Student's t distribution is similar to the Normal Distribution, but with heavier tails.

$n := 50 \qquad i := 0 .. n$

< 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...

$b := -\left(\frac{1}{2} \cdot n\right) \qquad c := 0.1$

< 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.

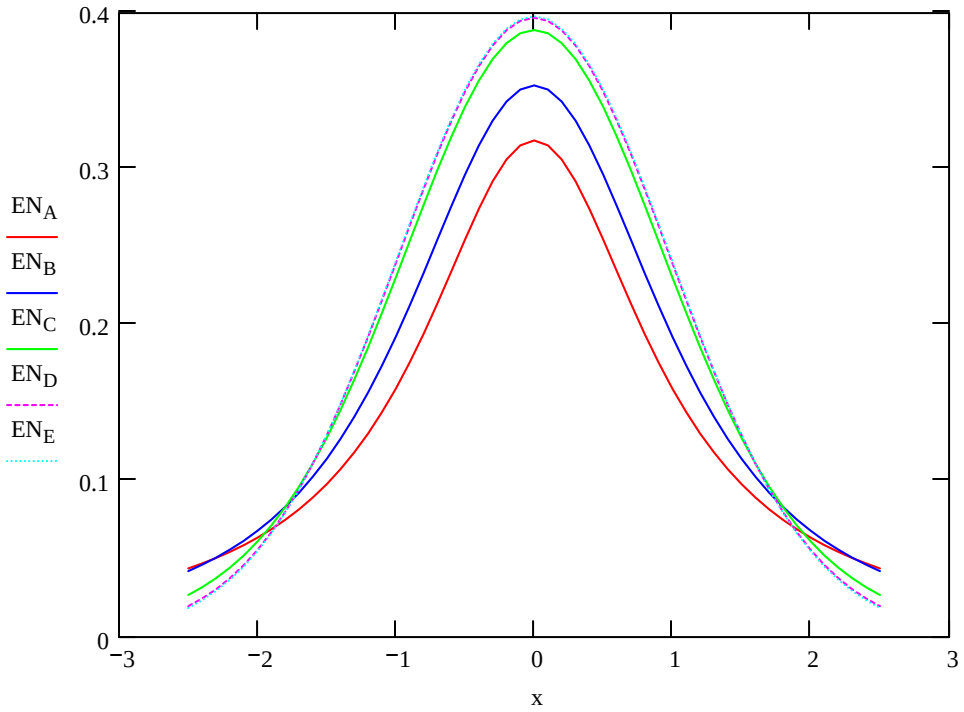
$df_t := 1$

< "degrees of freedom" for the t distribution

$EN_A := dt(x, df_t)$

$EN_B := dt(x, 2 \cdot df_t) \qquad EN_D := dt(x, 50 \cdot df_t)$

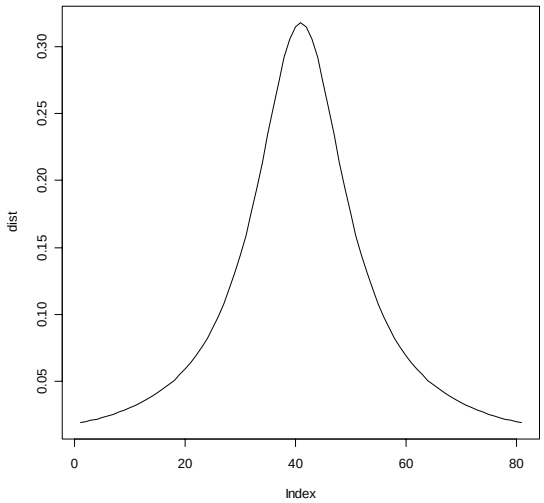
$EN_C := dt(x, 10 \cdot df_t) \qquad EN_E := dt(x, 100 \cdot df_t)$



Prototype in R:

COMMANDS:

```
#STUDENT'S t DISTRIBUTION:
# SETS UP VARIABLE X AS A RANGE
x=seq(-4,4,0.1)
# SPECIFY DEGREES OF FREEDOM
dft=1
#PLOTTING WITH LINE TYPE "1" (the letter l)
dist=dt(x,dft)
plot(dist,type="l")
```



Chi-Square (χ^2) Distribution:

This distribution is commonly encountered in statistics, especially in what is known as "Goodness of Fit" tests.

$n := 50$ $i := 0 .. n$

< 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...

$b := 1.4$ $c := 0.3$

< 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.

$df_{chisq} := 1$

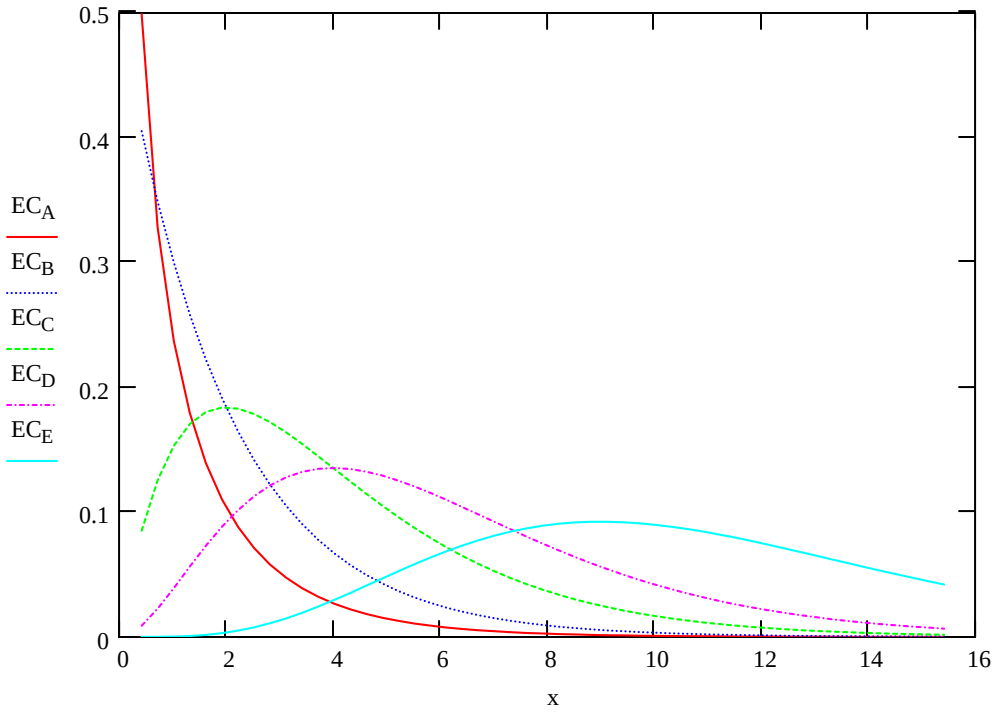
< d is a parameter for the χ^2 distribution called "degrees of freedom". Thus χ^2 defines a family of curves.

$EC_A := dchisq(x, df_{chisq})$

$EC_B := dchisq[x, (df_{chisq} + 1)]$ $EC_D := dchisq[x, (df_{chisq} + 5)]$

$EC_C := dchisq[x, (df_{chisq} + 3)]$ $EC_E := dchisq[x, (df_{chisq} + 10)]$

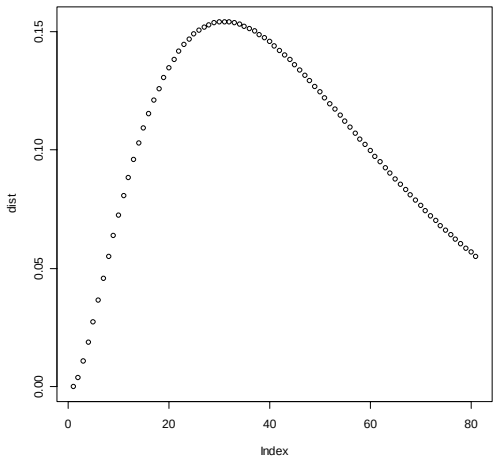
χ^2 family plotted below. As above, probability density the area under each curve.



Prototype in R:

COMMANDS:

```
# CHI-SQUARE DISTRIBUTION:
# SETS UP VARIABLE X AS A RANGE
x=seq(0,8,0.1)
# SPECIFY DEGREES OF FREEDOM
dfchisq=5
#PLOTTING
dist=dchisq(x,dfchisq)
plot(dist)
```



F Distribution:

Typical distributions an a wide variety of Linear Models tests.

$n := 50$ $i := 0 .. n$

< 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...

$b := 1.4$ $c := 0.3$

< 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.

$df_{F1} := 1$ $df_{F2} := 1$

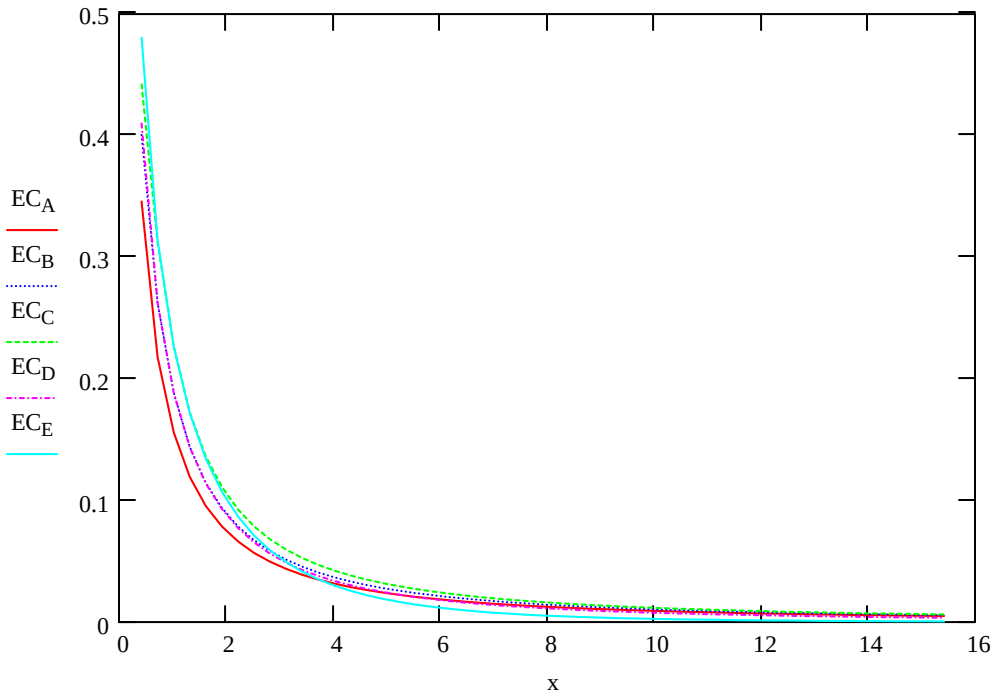
< d1 & d2 are numerator and denominator "degrees of freedom" respectively. Thus the F distribution defines a family of curves.

$EC_A := dF(x, df_{F1}, df_{F2})$

$EC_B := dF[x, (df_{F1} + 1), df_{F2}]$ $EC_D := dF[x, df_{F1}, (df_{F2} + 1)]$

$EC_C := dF[x, (df_{F1} + 10), df_{F2}]$ $EC_E := dF[x, df_{F1}, (df_{F2} + 10)]$

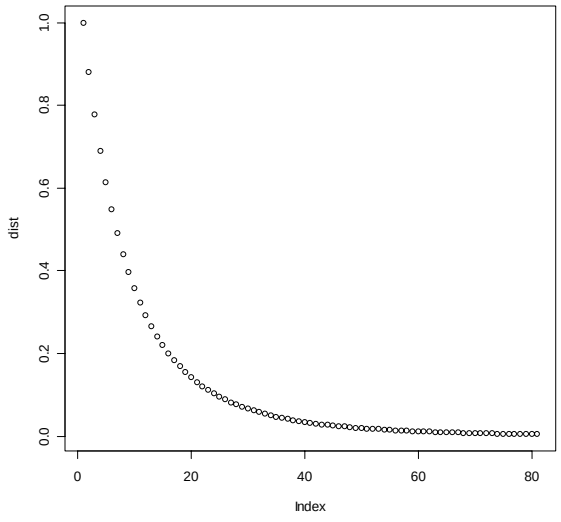
F family plotted below. As above, probability density the area under each curve.



Prototype in R:

COMMANDS:

```
# F DISTRIBUTION:
# SETS UP VARIABLE X AS A RANGE
x=seq(0,8,0.1)
# SPECIFY DEGREES OF FREEDOM
dfF1=2
dfF2=7
#PLOTTING
dist=df(x,dfF1,dfF2)
plot(dist)
```



Discrete Probability Distribution:

Binomial distribution:

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

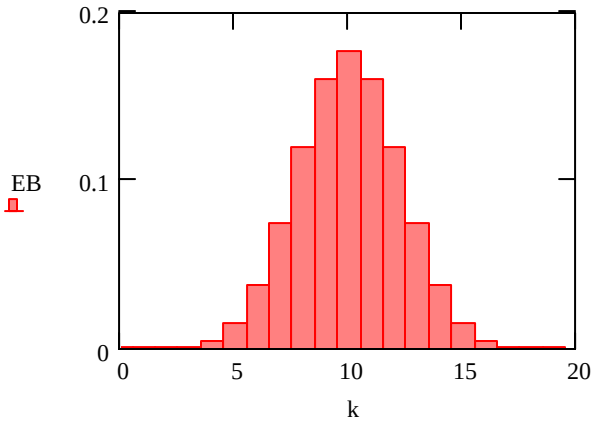
number of times one obtains a "heads" Note that this is a range of discrete possibilities (ranging from 0 to n) $i := 0 .. n - 1$

$k_i := i$

Expected probability for each k: (E_k): $EB := dbinom(k, n, p)$

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

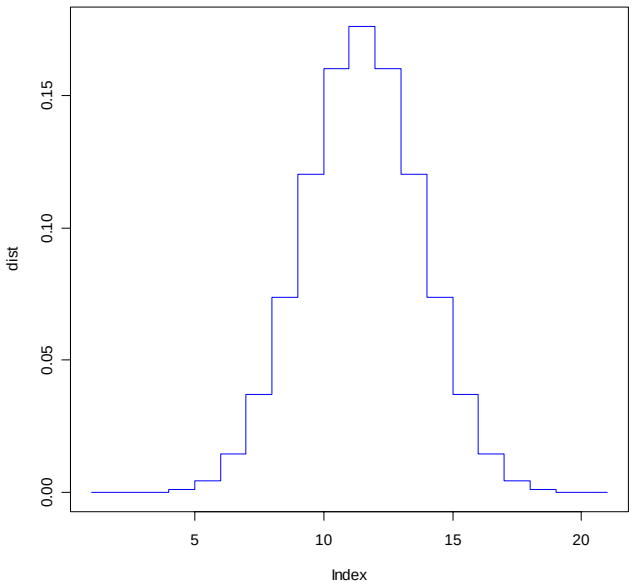


<- It is unlikely to find 0 or 20 heads as outcome of 20 coin tosses. An intermediate number is much more likely.

Prototype in R:

COMMANDS:

```
# BINOMIAL DISTRIBUTION:
# SETS UP VARIABLE X AS A RANGE
x=seq(0,20,1)
#NUMBER OF TRIALS:
n=20
#PROBABILITY
p=0.5
dist=dbinom(x,n,p)
plot(dist,type="s")
```



Calculating Probabilities & Quantiles:

The above graphs display the relationship between x values, observations (also called quantiles), and the probability that x is expected to have given an assumption of probability distribution for x (Normal, t, χ^2, F, or binomial, etc.). Standard "p" and "q" functions allow conversion from x to P(x) and vice versa. In all cases, the probability function is given as a CUMULATIVE probability Φ(x) starting from x values of minus infinity to x. In each case a specific cumulative probability function reqires that one specifies specific parameter values for the curve (μ,σ, df's), along with x OR Φ(x).

Normal Distribution:

$\mu := 0 \qquad \sigma := 1$ **< specify these to calculate one from a family of Normal Distributions**

x to Φ(x):

$x := 1.6449$ **< specify value of x**

$\text{pnorm}(x, \mu, \sigma) = 0.95$ **< Calculated cumulative probability Φ(x)**

Φ(x) to x:

$\Phi := 0.90$ **< specify Φ(x)**

$\text{qnorm}(\Phi, \mu, \sigma) = 1.2816$ **< Calculated quantile x**

Prototype in R:

COMMANDS:

```
#PQ FUNCTIONS FOR NORMAL DISTRIBUTION:
mu=0
sigma=1
x=1.6449
PHI=0.90
pnorm(x,mu,sigma)
qnorm(PHI,mu,sigma)
```

Student's t Distribution:

$dft := 1$ **< specify degrees of freedom for t distribution**

x to Φ(x):

$x := 1.6449$ **< specify value of x**

$\text{pt}(x, dft) = 0.8261$ **< Calculated cumulative probability Φ(x)**

Φ(x) to x:

$\Phi := 0.90$ **< specify Φ(x)**

$\text{qt}(\Phi, dft) = 3.0777$ **< Calculated quantile x**

Prototype in R:

COMMANDS:

```
#PQ FUNCTIONS FOR STUDENT'S t DISTRIBUTION:
dft=1
x=1.6449
PHI=0.90
pt(x,dft)
qt(PHI,dft)
```

Chi-Square (χ^2) Distribution:

$dfchisq := 1$ **< specify degrees of freedom for χ^2 distribution**

x to Φ(x):

$x := 1.6449$ **< specify value of x**

$pchisq(x, df_{chisq}) = 0.8003$ **< Calculated cumulative probability Φ(x)**

Φ(x) to x:

$\Phi := 0.90$ **< specify Φ(x)**

$qchisq(\Phi, df_{chisq}) = 2.7055$ **< Calculated quantile x**

Prototype in R:

COMMANDS:

```
#PQ FUNCTIONS FOR CHISQ DISTRIBUTION:
dfchisq=1
x=1.6449
PHI=0.90
pchisq(x,dfchisq)
qchisq(PHI,dfchisq)
```

F Distribution:

$df_{F1} := 2$ **< specify numerator df_{F1} degrees of freedom**

$df_{F2} := 7$ **< specify denominator df_{F2} degrees of freedom**

x to Φ(x):

$x := 1.6449$ **< specify value of x**

$pF(x, df_{F1}, df_{F2}) = 0.7403$ **< Calculated cumulative probability Φ(x)**

Φ(x) to x:

$\Phi := 0.95$ **< specify Φ(x)**

$qF(\Phi, df_{F1}, df_{F2}) = 4.7374$ **< Calculated quantile x**

Prototype in R:

COMMANDS:

```
#PQ FUNCTIONS FOR F DISTRIBUTION:
dfF1=2
dfF2=7
x=1.6449
PHI=0.95
pf(x,dfF1,dfF2)
qf(PHI,dfF1,dfF2)
```

Binomial Distribution:

$n := 20$ **< specify number of trials**

$p := 0.50$ **< specify probability of single event**

x to Φ(x):

$x := 5$ **< specify value of x (number of events observed)**

$pbinom(x, n, p) = 0.0207$ **< Calculated cumulative probability Φ(x)**

Φ(x) to x:

$\Phi := 0.95$ **< specify Φ(x)**

$qbinom(\Phi, n, p) = 14$ **< Calculated quantile x**

Prototype in R:

COMMANDS:
#PQ FUNCTIONS FOR BINOMIAL DISTRIBUTION:
n=20
p=0.50
x=5
PHI=0.95
pbinom(x,n,p)
qbinom(PHI,n,p)

Generating Pseudo-Random Numbers:

Standard computer packages also have randomize (r) functions that provide lists (more commonly called vectors) of numbers that are approximately random and follow the above probability distributions. Although appearing random, they *always* are not. Each value is generated by a "random number generator" computer procedure that is, by definition, not random. Sometimes, the resultant distributions are not even close.

$m := 1000$ **< specify how many x (quantile) values you want..**

$\mu := 0$ $\sigma := 1$ $df_t := 1$ $df_{chisq} := 3$ $df_{F1} := 2$ $df_{F2} := 7$ **< also specify parameter values.**

$n := 20$ $p := 0.5$

Making the vectors K:

$K_{normal} := rnorm(m, \mu, \sigma)$

$K_t := rt(m, df_t)$

$K_{chisq} := rchisq(m, df_{chisq})$

$K_F := rF(m, df_{F1}, df_{F2})$

$K_{binom} := rbinom(m, n, p)$

Prototype in R:

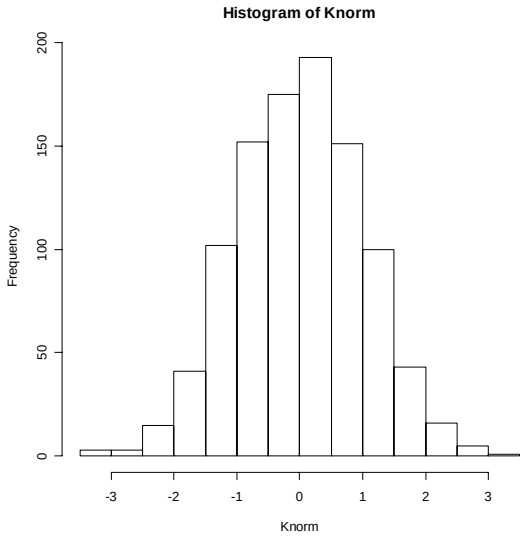
COMMANDS:

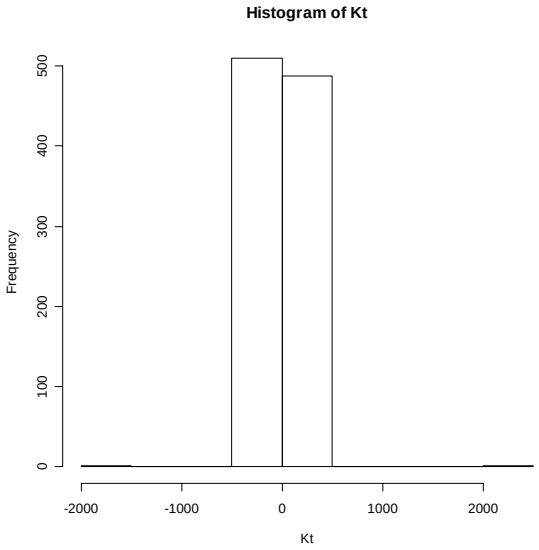
 # PSEUDO-RANDOM VECTORS:
 # SPECIFY VECTOR LENGTH m
m=1000
 # SPECIFY PARAMETERS:
mu=0
sigma=1
dft=1
dfchisq=3
dfF1=2
dfF2=7
n=20
p=0.50

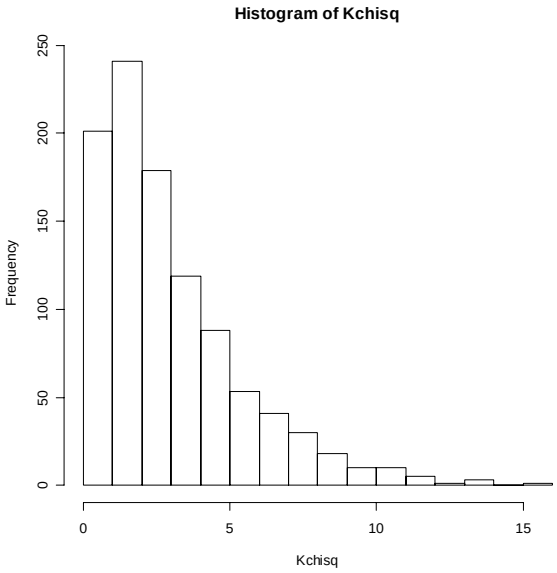
COMMANDS CONTINUED:

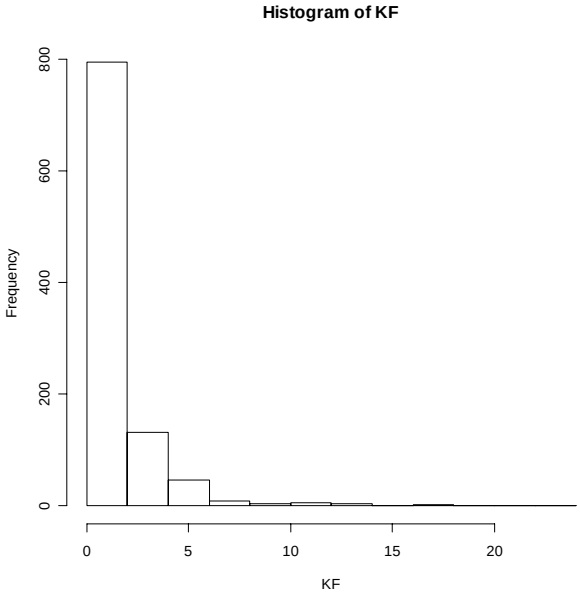
Knorm=rnorm(m,mu,sigma)
Kt=rt(m,dft)
Kchisq=rchisq(m,dfchisq)
KF=rf(m,dfF1,dfF2)
Kbinom=rbinom(m,n,p)

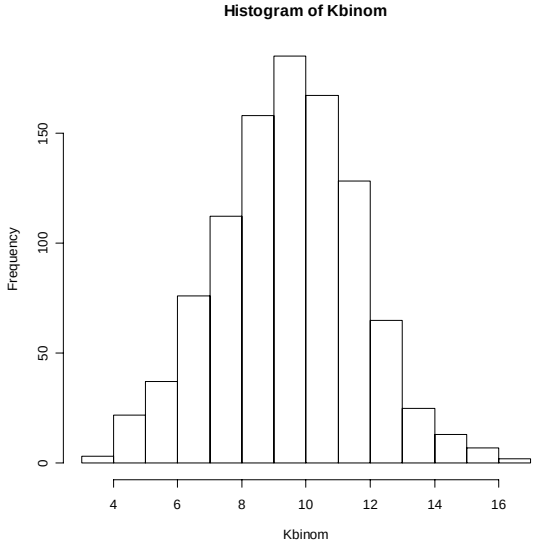
 # MAKE HISTOGRAMS:
hist(Knorm)
hist(Kt)
hist(Kchisq)
hist(KF)
hist(Kbinom)


Histogram of Knorm
Frequency
0
50
100
150
200
-3
-2
-1
0
1
2
3
Knorm


Histogram of Kt
Frequency
0
100
200
300
400
500
-2000
-1000
0
1000
2000
Kt


Histogram of Kchisq
Frequency
0
50
100
150
200
250
0
5
10
15
Kchisq


Histogram of KF
Frequency
0
200
400
600
800
0
5
10
15
20
KF


Histogram of Kbinom
Frequency
0
50
100
150
4
6
8
10
12
14
16
Kbinom