ORIGIN ≡ 0

Linear Model Regression Diagnostics

W. Stein

Regression analysis often involves a repetitive process fitting different linear models relating the dependent variable Y with different sets of independent variables X_i. Diagnostic plots or calculations are often helpful in deciding whether particular X_i's belong in or out of the model. They are also helpful in identifying specific cases, typically rows in the dataset (such as K below), that are somehow mis-specified often by mis-measurement or mistaken entry into the data table. Other cases of "mis-specification" may in fact involve a real phenomenon requiring additional study. Presented here are several important diagnostic techniques useful for identifying mis-specification. What one does with the information, however, is very much part of the science of the study, as opposed to the calculation of statistics. Examples below comes from Chapter 10 in Kuter et al. (KNNL) A*pplied Linear Statistical Models* 5th Edition.

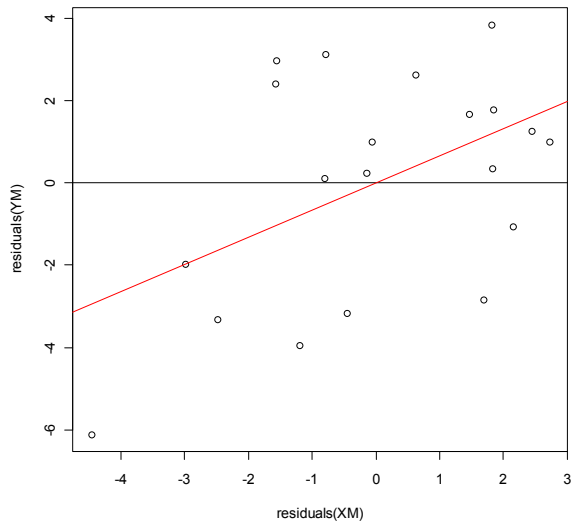
Example & Calculations in R:

Body Fat Example KNNL Table 7.1
Compare results in Tables 10.3 & 10.4

Added Variable Plots:

```
#DIAGNOSTICS FOR LINEAR MODELS
require(car) #DOWNLOAD {car} FROM CRAN WEBSITE
#READ STRUCTURED DATA TABLE WITH NUMERIC CODED FACTOR
K=read.table("c:/2008LinearModelsData/BodyFatR.txt")
K
attach(K)
options(digits=6)
cor(K)

#MAKING ADDED VARIABLE PLOT FOR X2 GIVEN X1 IN THE MODEL
FM=lm(Y~X1+X2)
YM=lm(Y~X1)
XM=lm(X2~X1)
plot(residuals(XM),residuals(YM))
abline(0,0)
abline(0,coefficients(FM)[3],col="red")
av.plots(FM,X2) #ALTERNATE ADDED VARIABLE PLOT FUNCTION IN {car}
```

```
> K
      X1   X2   X3     Y
1   19.5 43.1 29.1 11.9
2   24.7 49.8 28.2 22.8
3   30.7 51.9 37.0 18.7
4   29.8 54.3 31.1 20.1
5   19.1 42.2 30.9 12.9
6   25.6 53.9 23.7 21.7
7   31.4 58.5 27.6 27.1
8   27.9 52.1 30.6 25.4
9   22.1 49.9 23.2 21.3
10  25.5 53.5 24.8 19.3
11  31.1 56.6 30.0 25.4
12  30.4 56.7 28.3 27.2
13  18.7 46.5 23.0 11.7
14  19.7 44.2 28.6 17.8
15  14.6 42.7 21.3 12.8
16  29.5 54.4 30.1 23.9
17  27.7 55.3 25.7 22.6
18  30.2 58.6 24.6 25.4
19  22.7 48.2 27.1 14.8
20  25.2 51.0 27.5 21.1
```




```
> cor(K)
          X1        X2        X3        Y
X1 1.000000 0.9238425 0.4577772 0.843265
X2 0.923843 1.0000000 0.0846675 0.878090
X3 0.457777 0.0846675 1.0000000 0.142444
Y  0.843265 0.8780896 0.1424440 1.000000
```

Residuals of Y on X's already in the model, and of a new X_i on X_i's already in the model are regressed. If the scatter is better fit by a sloped line though the origin (here in red), then the zero line (in black), evidence suggests the new X will provide an additional reduction in SSE if added to the existing model. Slope of the line is equivalent to the regression coefficient for the new variable after entry into an expanded model.

Diagnostics for Outlying Y values:

```
#SETTING UP AND MAKING HAT MATRIX
n=length(Y)
X=model.matrix(FM)
p=ncol(X)
H=X%*%solve(t(X)%*%X)%*%t(X) #HAT MATRIX
h=diag(H) #LEVERAGE
hatvalues(FM)  #FUNCTION FOR DIRECT CALCULATION

#CALCULATIONS FOR OUTLYING Y OBSERVATIONS
sd=summary(FM)$sigma
sd              #SQUARE-ROOT OF MSE
MSE=sd^2
SSE=(n-p)*MSE      #SUM OF SQUARES ERROR
#FINDING RESIDUALS
e=residuals(FM)    #RESIDUAL
#CALCULATING SEMI-STUDENTIZED RESIDUALS KNNL Eq. 10-9
SEMISTUDRES=e/sd     #SEMI-STUDENTIZED RESIDUALS
#CALCULATING STUDENTIZED RESIDUALS KNNL Eq.10.19 & 10.20
STUDRES=e/sqrt(MSE*(1-h))  #STUDENTIZED RESIDUALS
rstandard(FM)          #FUNCTION FOR DIRECT CALCULATION
#CALCULATING STUDENTIZED DELETED RESIDUALS KNNL Eq. 10.26
t=e*sqrt((n-p-1)/(SSE*(1-h)-e^2))
DELSTUDRES=t        #STUDENTIZED DELETED RESIDUALS
rstudent(FM)          #FUNCTION FOR DIRECT CALCULATION
YRESULTS=cbind(e,h,SEMISTUDRES,STUDRES,DELSTUDRES)
YRESULTS
```

< n = number of cases

< p = number of parameters

< hat matrix H

< h = main diagonal of H

< sd = standard deviation derived from summary function in R

< MSE = sd^2

< SSE = df_E * MSE

< e = residuals extracted from FM

< calculation of Y diagnostics

< put into a table for display

In all "corrected" residual measures, one is looking for large values for particular cases as possible indication of mis-specification. Large values indicate possible outliers in Y even if they aren't visable in bivariate plots of Y vs individual X_i's.

Deleted Studentized Residuals (last column) are perhaps the most perceptive measure. Here "corrected" residual for a particular case is computed with regard to regression fit for all but that case, thus maximizing apparent lack of fit if the point is an outlier in Y.

A formal Bonferroni t-test is available to allow statement that a particular Deleted Studentized Residual is beyond expectations (see KNNL p. 395), but in my opinion rarely is this the objective of disgnostic work.

```
> YRESULTS
             e         h  SEMISTUDRES      STUDRES   DELSTUDRES
1  -1.682709311 0.2010125 -0.661659239 -0.740226105 -0.729985403
2   3.642931179 0.0588948  1.432439373  1.476580603  1.534254132
3  -3.175970140 0.3719330 -1.248825315 -1.575791342 -1.654329572
4  -3.158465120 0.1109401 -1.241942154 -1.317151878 -1.348484207
5  -0.000288658 0.2480103 -0.000113503 -0.000130889 -0.000126981
6  -0.360815519 0.1286162 -0.141876508 -0.151986758 -0.147549094
7   0.716199189 0.1555175  0.281617156  0.306452923  0.298127621
8   4.014732755 0.0962878  1.578635716  1.660606955  1.760092492
9   2.655105736 0.1146356  1.044015878  1.109547978  1.117648740
10 -2.474811541 0.1102443 -0.973122279 -1.031649181 -1.033728421
11  0.335806380 0.1203365  0.132042648  0.140784859  0.136661066
12  2.225511014 0.1092663  0.875094654  0.927216363  0.923178504
13 -3.946861346 0.1783818 -1.551947954 -1.712151262 -1.825902725
14  3.447456194 0.1480068  1.355576525  1.468608324  1.524763051
15  0.570587104 0.3332120  0.224360932  0.274759900  0.267150092
16  0.642298478 0.0952774  0.252558608  0.265524411  0.258132342
17 -0.850946475 0.1055947 -0.334601225 -0.353802041 -0.344509100
18 -0.782919881 0.1967928 -0.307852443 -0.343501635 -0.334408084
19 -2.857288765 0.0669542 -1.123516401 -1.163129129 -1.176171277
20  1.040448727 0.0500853  0.409115531  0.419762514  0.409356417
```

Diagnostics for Outlying X values:

```
#CALCULATIONS FOR OUTLYING X OBSERVATIONS
#LEVERAGE KNNL p. 398
LEVERAGE=diag(H)  #LEVERAGE = MAIN DIAGONAL OF HAT MATRIX
#DFFITS KNNL Eq. 10.30a
DFFITS=t*sqrt(h/(1-h))  #DFFITS
dffits(FM)          #FUNCTION FOR DIRECT CALCULATION
#COOK'S DISTANCE KNNL Eq. 10.33b
COOKSDIST=((e^2)/(p*MSE))*(h/(1-h)^2)
cooks.distance(FM)     #FUNCTION FOR DIRECT CALCULATION
COOKSPC=pf(COOKSDIST,p,n-p) #F-DIST PERCENTILES OF COOK'S DISTANCE
#DFBETAS KNNL +++ NO EXPLICIT FORMULA GIVEN +++
DFBETAS=dfbetas(FM)    #FUNCTION FOR DIRECT CALCULATION
XRESULTS=cbind(e,LEVERAGE,DFFITS,COOKSDIST,COOKSPC,DFBETAS)
XRESULTS
```

```
> XRESULTS                                                          --------- DFBETAS ----------
               e  LEVERAGE       DFFITS   COOKSDIST     COOKSPC  (Intercept)           X1           X2
1  -1.682709311 0.2010125 -3.66147e-01 4.59505e-02 1.35334e-02 -3.05182e-01 -1.31486e-01  2.32032e-01
2   3.642931179 0.0588948  3.83810e-01 4.54812e-02 1.33331e-02  1.72573e-01  1.15025e-01 -1.42613e-01
3  -3.175970140 0.3719330 -1.27307e+00 4.90157e-01 3.06265e-01 -8.47101e-01 -1.18252e+00  1.06690e+00
4  -3.158465120 0.1109401 -4.76348e-01 7.21619e-02 2.59185e-02 -1.01612e-01 -2.93520e-01  1.96072e-01
5  -0.000288658 0.2480103 -7.29235e-05 1.88340e-09 1.17857e-13 -6.37212e-05 -3.05275e-05  5.02371e-05
6  -0.360815519 0.1286162 -5.66865e-02 1.13652e-03 5.51802e-05  3.96772e-02  4.00811e-02 -4.42676e-02
7   0.716199189 0.1555175  1.27937e-01 5.76494e-03 6.27314e-04 -7.75275e-02 -1.56129e-02  5.43163e-02
8   4.014732755 0.0962878  5.74521e-01 9.79385e-02 3.99070e-02  2.61431e-01  3.91126e-01 -3.32453e-01
9   2.655105736 0.1146356  4.02165e-01 5.31335e-02 1.67022e-02 -1.51352e-01 -2.94656e-01  2.46909e-01
10 -2.474811541 0.1102443 -3.63873e-01 4.39570e-02 1.26886e-02  2.37749e-01  2.44601e-01 -2.68809e-01
11  0.335806380 0.1203365  5.05458e-02 9.03799e-04 3.91411e-05 -9.02089e-03  1.70564e-02 -2.48452e-03
12  2.225511014 0.1092663  3.23337e-01 3.51544e-02 9.15878e-03 -1.30493e-01  2.24580e-02  6.99961e-02
13 -3.946861346 0.1783818 -8.50781e-01 2.12150e-01 1.13411e-01  1.19415e-01  5.92420e-01 -3.89491e-01
14  3.447456194 0.1480068  6.35514e-01 1.24893e-01 5.59078e-02  4.51744e-01  1.13172e-01 -2.97704e-01
15  0.570587104 0.3332120  1.88852e-01 1.25753e-02 2.00654e-03 -3.00428e-03 -1.24757e-01  6.87693e-02
16  0.642298478 0.0952774  8.37683e-02 2.47493e-03 1.77071e-04  9.30846e-03  4.31135e-02 -2.51250e-02
17 -0.850946475 0.1055947 -1.18373e-01 4.92614e-03 4.95950e-04  7.95121e-02  5.50436e-02 -7.60901e-02
18 -0.782919881 0.1967928 -1.65527e-01 9.63647e-03 1.35019e-03  1.32052e-01  7.53287e-02 -1.16100e-01
19 -2.857288765 0.0669542 -3.15071e-01 3.23601e-02 8.11248e-03 -1.29603e-01 -4.07203e-03  6.44293e-02
20  1.040448727 0.0500853  9.39971e-02 3.09679e-03 2.47677e-04  1.01905e-02  2.29080e-03 -3.31415e-03
```

Leverage:

Leverage values, the main diagonal of the hat matrix, range between 0 & 1 and sum to the number of parameter in FM including the intercept (i.e., = p). Leverage values for each case measure how far the set of X_i's for a specific case diverge from the centroid (average point for all X_i's). Large leverage values indicate most influence on where the regression fit between Y & X_i's is placed (leverage of 1 indicates fit with residual of zero). To identify outliers in X_i's, one therefore looks for large leverage values (> 0.5 for moderate/large datasets AND/OR values that are > 2p/n) and for leverage values that are considerably larger than average for all cases. Calculation of leverage may also be used to determine whether a certain set of X_i's are suitable for predicting a new Y given the regression fit - see KNNL p. 400.

For above FM:

$p := 3 \quad n := 20 \qquad 2 \cdot \frac{p}{n} = 0.3$

DFFITS:

DFFITS is a measure of the *influence* (i.e., practical effect) a particular case may have on the fitted regression, and therefore residual, for each case. Influence is determined by considering how deviant a particular residual in Y may be, plus also considering that case's leverage in X. Calculation of DFFITS conceptually involves calculating the difference in fitted values when a specific case is either included (Yhat) or excluded ($Yhat_{(i)}$). This difference is scaled by an estimation of the standard deviation of $Yhat_{(i)}$. Actual calculation is by explicit formula derived from the Deleted Studentized Residuals above. KNNL offer the following guidelines for determining whether a particular case (DFFITS value) is to be considered influential or not:

1) *Influential* if DFFITS > 1 for small to medium sized datasets.

2) *Influential* if DFFITS > $2 \cdot \sqrt{\frac{p}{n}}$ for large datasets.

For above FM:

$p := 3 \quad n := 20 \qquad 2 \cdot \sqrt{\frac{p}{n}} = 0.7746$

Cook's Distance:

Cook's Distance is an *aggregate influence* a particular case may have on all residuals of a linear fit. A sum of squares of the difference for each residual value, with and without a particular case included, is calculated and then scaled by a standardizing measure. Calculation of Cook's Distance is by explicit formula involving residuals (e) and the main diagonal of the Hat matrix (h_i). Cook's Distance is often plotted against case index number to identify instances particularly large aggregate influence. Comparison can also be made with the F distribution (see COOKSPC column in above chart) as follows:

For above FM:

$p := 3 \quad n := 20 \qquad D := 0.490157$

^ for case i = 3 above and KNNL p. 404

$pF(D, p, n - p) = 0.306265$ **< measure of influence**

If for a case:
pF(D,p,n-p) < .30 then little aggregate influence can be inferred.

However if,
pF(D,p,n-p) > .50 then the case has "major" influence of the fitted regression - see KNNL p. 403.

DFBETAS:

DFBETAS measure the difference in regression coefficient values (one for each column in the design matrix X including intercept) when a particular case is included versus excluded. Sign of DFBETAS indicate direction of change of each coefficient, whereas magnitude indicates relative influence, for each case. No explicit formula involving residuals (e) or Hat matrix (H or h) was given by KNNL, implying that unlike the other measures, multiple regressions need to be fit in order to calculate DFBETAS - paperwork best left to computer algorithms. For evaluation, KNNL suggest that a case is *influential* if:

DFBETAS > 1 for small to medium sized datasets, or
DFBETAS > $\frac{2}{\sqrt{n}}$ for large datasets.

For above FM:

$p := 3 \quad n := 20 \qquad \frac{2}{\sqrt{n}} = 0.4472$

Variance Inflation Factors (VIF) and Tolerance:

```
#CALCULATING VARIANCE INFLATION FACTORS
FM=lm(Y~X1+X2+X3)     #CONSTRUCTING FM WITH X1+X2+X3
XM=cbind(X1,X2,X3)      #CONSTRUCTING MATRIX OF Xi's
R=cor(XM)               #CORRELATION MATRIX OF Xi's
R
INVR=solve(R)           #INVERSE OF R
INVR
VIF=diag(INVR)          #VIF = VARINACE INFLATION FACTORS
VIF
vif(FM)                 #VIF FUNCTION IN {car} PACKAGE
mean(VIF)               #AVERAGE VIF
TOLERANCE=1/VIF     #TOLERANCE FACTORS IN AUTOMATED REGRESSIONS
TOLERANCE
```

Variance Inflation Factors comprise a formal method for detecting multicolinearity between independent variables in a multiple regression. Multicolinearity has detrimental effects on reliability of predictions involving Y_{hat} and on stability of regresson coefficients when different X_i's are added or deleted from a regression model. When there are multiple X_i's, multicolinearity is almost always present to some extent but difficult to detect by simple means involving bivariate plots.

The VIF vector is simply calculated as the main diagonal of the Inverse of the Correlation Matrix. VIF values are 1 when the X_i's are uncorrelated. Otherwise VIF's are larger than 1 and unbound. The *largest* VIF is used to measure the extent of multicolinearity among the X_i's, with max(VIF) > 10 indicating 'serious' multicolinearity adversely influencing regression coefficient values. The *average* of VIF's is also used to detect adverse influence in terms of how much the estimates (b's) of regression coefficients may differ from true/actual (β's) values. Mean values > 1 are indicative of problematic levels of multicolinearity.

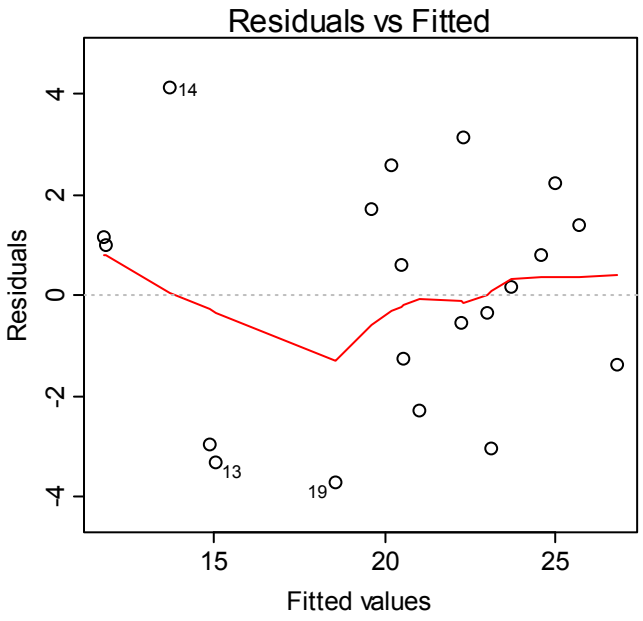
Tolerance is the inverse of each VIF value. The term is often employed in automatic computer programs involved in serial (stepwise) regressions. Frequently used cut-off values below which an Xi will not be allowed to enter a stepwise regression are often: 0.01, 0.001, or 0.0001.

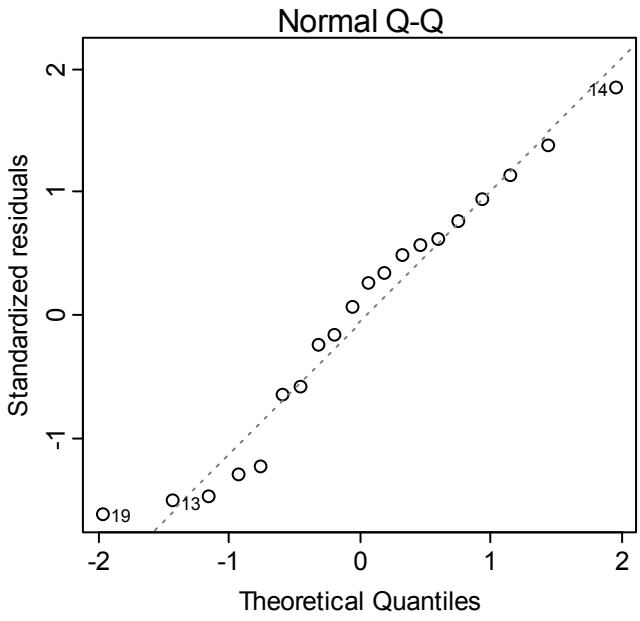
```
> R
          X1        X2        X3
X1 1.000000 0.9238425 0.4577772
X2 0.923843 1.0000000 0.0846675
X3 0.457777 0.0846675 1.0000000

> INVR
          X1       X2       X3
X1  708.843 -631.915 -270.989
X2 -631.915  564.343  241.495
X3 -270.989  241.495  104.606

> VIF
     X1       X2       X3
708.843 564.343 104.606

> TOLERANCE
         X1          X2         X3
0.00141075 0.00177197 0.00955968
```

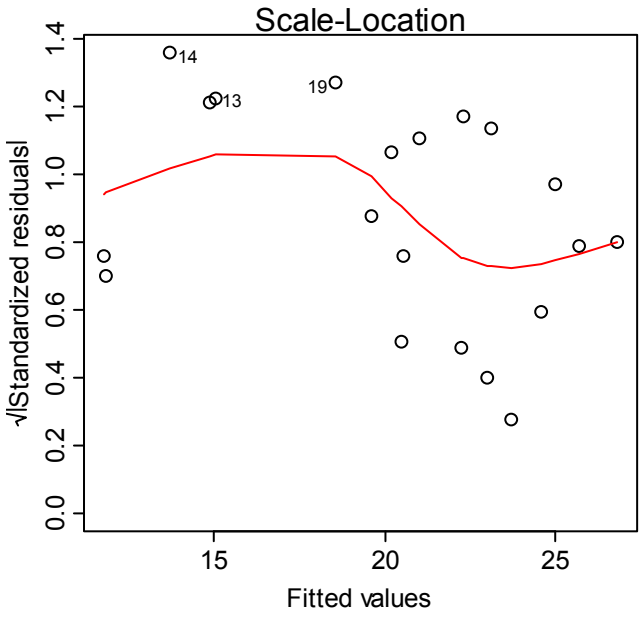
BUILT IN Diagnostic Plots in R:

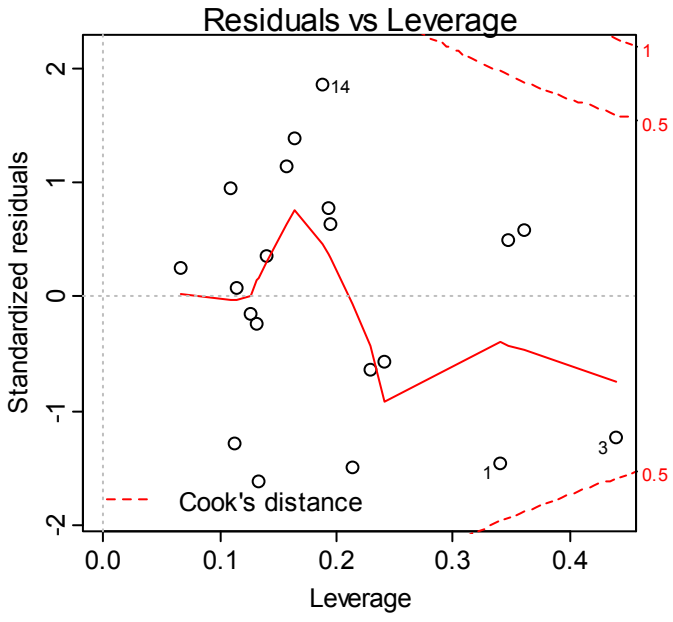
```
#DIAGNOSTIC PLOTTING FUNCTIONS
par(mfrow=c(2,2), mex=0.6)
plot(FM)
par(mfrow=c(1,1), mex=1)
```

< Built-into plot()









Constructing your own plot in R:

```
#MORE DIAGNOSTIC PLOTS from P.Dalgaard p. 184
par(mfrow=c(2,2), mex=0.6)
plot(rstandard(FM))
plot(rstudent(FM))
plot(dffits(FM),type="l")
matplot(dfbetas(FM),type="l",col="red")
lines(sqrt(cooks.distance(FM)),lwd=2,col="blue")
par(mfrow=c(1,1), mex=1)
```

< constructed by extracting interesting variables. Note use of framing with par()