Adapted from P. Smith, J. Picka

10 Regression Analysis in SAS

The following SAS code produces statistical descriptions and a simple regression analysis of a small data set. The data are the introductory example from Draper and Smith (1998).

The `DATA` step causes SAS to read data values directly from the input stream. In this example, the goal is to predict monthly steam usage (`STEAMUSE`) from average monthly temperature (`TEMP`), using a straight line regression.

Statistical characteristics of the data are examined using `PROC UNIVARIATE`. The `PLOT` option in the `PROC UNIVARIATE` statement cause SAS to produce crude histograms and boxplots. The `NORMAL` option causes SAS to test the hypothesis that the variable has a normal distribution. These options are given for illustrative purposes only, since we are not interested in the distribution of `TEMP` and do not believe that the `STEAMUSE` observations are identically distributed. `PROC UNIVARIATE` produces lots of output and is mainly used for exploratory purposes.

It is always useful to obtain scatter plots of the raw data in regression problems. The `PROC PLOT` step generates crude printer plots, but in this case it is sufficient to show that there is a rough linear trend with a negative slope and no wild outliers.

The regression analysis is performed using `PROC REG`. In this example we only specify the model to be estimated (in the `MODEL` statement). It is possible to get much more: plots, diagnostics and tests of model assumptions.

The `PROC GPLOT` step produces a high-resolution graph of the raw data with the regression line superimposed. The form of the graph is specified in the `SYMBOL` statement, which specifies that a least squares regression line should be used to "interpolate" between data points and that raw data points should be indicated by plus signs.

```
options ls=70 ;
title1 'STEAM DATA FROM CHAPTER 1 OF DRAPER & SMITH' ;
data steam ;
   input steamuse temp @@ ;  /* Note that SAS requires ONLY 2 obs
   per line (for 2 variables) without the @@ at end of input line. */
      datalines ;
10.98  35.3 11.13  29.7 12.51  30.8  8.40  58.8  9.27  61.4
```

```
 8.73  71.3  6.36  74.4  8.50  76.7  7.82  70.7  9.14  57.5
 8.24  46.4 12.19  28.9 11.88  28.1  9.57  39.1 10.94  46.8
 9.58  48.5 10.09  59.3  8.11  70.0  6.83  70.0  8.88  74.5
 7.68  72.1  8.47  58.1  8.86  44.6 10.36  33.4 11.08  28.6
       ;

proc univariate data=steam plot normal ;
   var steamuse temp ;
   title2 'Univariate Descriptive Statistics' ;

proc plot data=steam ;
   title2 'Scatterplot of Raw Data' ;
   plot steamuse*temp ;

proc reg data=steam ;
   title2 'Least Squares Analysis' ;
   model steamuse = temp ;
               /* NOTE: don't need 'run' between   PROC's */
proc gplot data=steam ;
   symbol i=rl value=PLUS ;
   plot steamuse*temp ;
   title2 'Observed Values and Estimated Regression Line' ;
run ;
```

The 25 data values produce about 3.5 pages of SAS output. Shorter output could have been generated using `PROC MEANS` or `PROC CORR`. However, these procedures can not produce histograms or test for normality.

The regression output from `PROC REG` appears on page 10 of the SAS output. The Analysis of Variance table shows how the total variation $\sum(Y_i - \bar{Y})^2$ is decomposed into a component "explained" by the regression model and an unexplained component described as Error. The F-test has a very small p value so that we conclude that the straight line model $Y_i = \beta_0 + \beta_1 x_i + e_i$ fits the data much better than the trivial model $Y_i = \beta_0 + e_i$. Because $R^2 = .7144$, we conclude that 71% of the variation in Y is "explained" by the linear regression relationship.

The estimated slope and intercept appear in the table headed "Parameter Estimates." For each parameter, the least squares estimate and estimated standard error are given. Also, SAS provides a Student t test of the null hypothesis that the true parameter is zero. Note that the square of the t statistic for the slope is equal to the F statistic for the model, in accordance with the theory.

Univariate Descriptive Statistics

The UNIVARIATE Procedure
Variable: steamuse

Moments

N	25	Sum Weights	25
Mean	9.424	Sum Observations	235.6
Std Deviation	1.63064149	Variance	2.65899167
Skewness	0.21135583	Kurtosis	-0.6061937
Uncorrected SS	2284.1102	Corrected SS	63.8158
Coeff Variation	17.3030718	Std Error Mean	0.3261283

...

The UNIVARIATE Procedure
Variable: steamuse

Tests for Normality

Test	--Statistic---		-----p Value------	
Shapiro-Wilk	W	0.971215	Pr < W	0.6760
Kolmogorov-Smirnov	D	0.110663	Pr > D	>0.1500
Cramer-von Mises	W-Sq	0.055016	Pr > W-Sq	>0.2500
Anderson-Darling	A-Sq	0.310084	Pr > A-Sq	>0.2500

...

The UNIVARIATE Procedure
Variable: steamuse

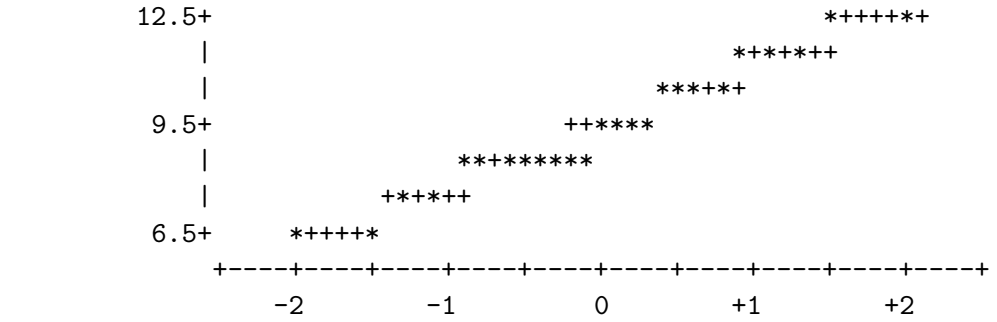
...

Stem	Leaf	#	Boxplot
12	25	2	
11	0119	4	
10	149	3	+-----+
9	1366	4	*--+--*
8	12455799	8	+-----+
7	78	2	
6	48	2	

----+----+----+----+

Variable: steamuse

Normal Probability Plot



The UNIVARIATE Procedure
Variable: temp

Moments

N	25	Sum Weights	25
Mean	52.6	Sum Observations	1315
Std Deviation	17.2655968	Variance	298.100833
Skewness	-0.1157664	Kurtosis	-1.5262541
Uncorrected SS	76323.42	Corrected SS	7154.42
Coeff Variation	32.8243285	Std Error Mean	3.45311936

...

Variable: temp

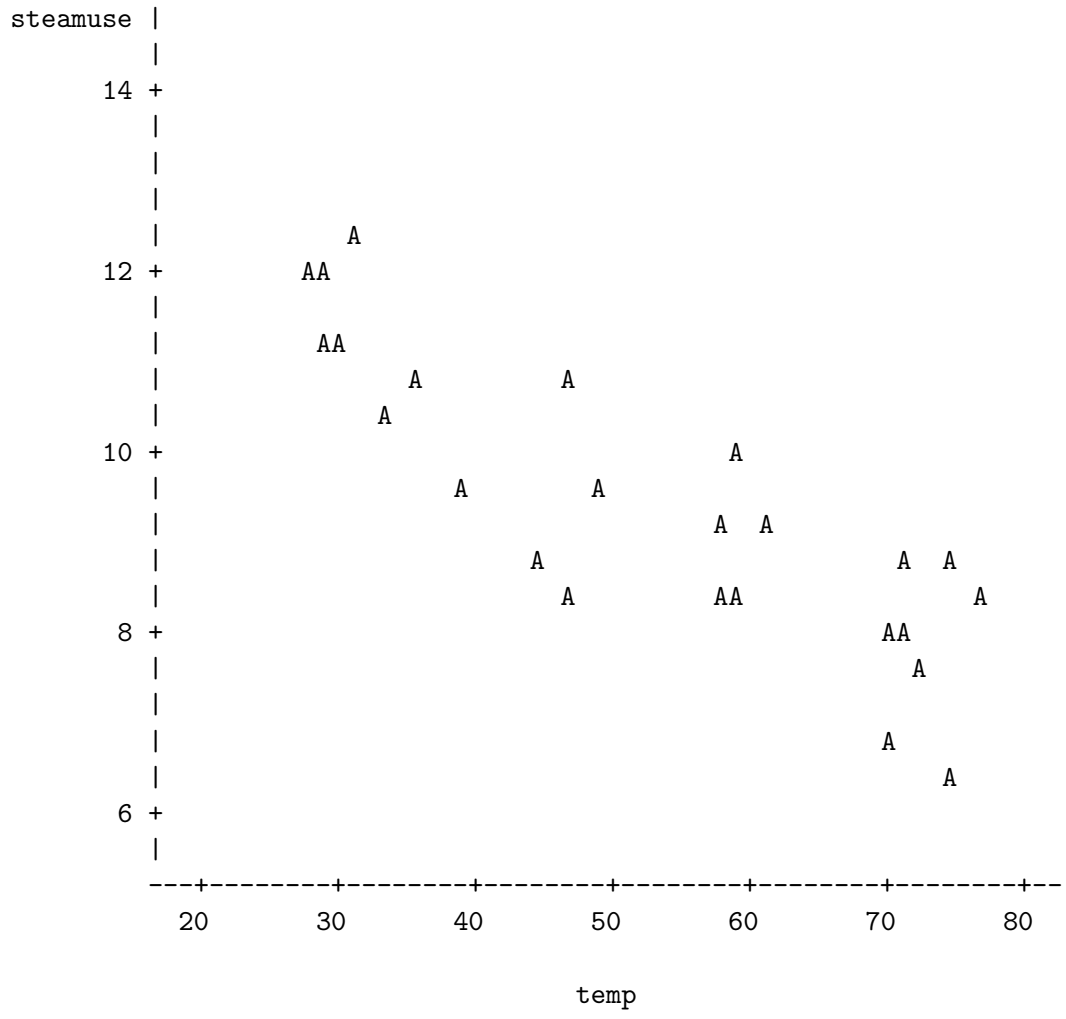
Tests for Normality

Test	--Statistic---		-----p Value------	
Shapiro-Wilk	W	0.897492	Pr < W	0.0162
Kolmogorov-Smirnov	D	0.163221	Pr > D	0.0847
Cramer-von Mises	W-Sq	0.115956	Pr > W-Sq	0.0676
Anderson-Darling	A-Sq	0.826067	Pr > A-Sq	0.0290

...

Scatterplot of Raw Data

Plot of steamuse*temp. Legend: A = 1 obs, B = 2 obs, etc.



Least Squares Analysis

The REG Procedure
Model: MODEL1
Dependent Variable: steamuse

Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	1	45.59240	45.59240	57.54	<.0001
Error	23	18.22340	0.79232		
Corrected Total	24	63.81580			

Root MSE	0.89012	R-Square	0.7144
Dependent Mean	9.42400	Adj R-Sq	0.7020
Coeff Var	9.44529		

Parameter Estimates

Variable	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	13.62299	0.58146	23.43	<.0001
temp	1	-0.07983	0.01052	-7.59	<.0001

We will see in the next segment of this handout that one can calculate 'by hand' how far out individual residuals are from a regression-model fit in **Splus**. To do this more automatically in SAS, try the INFLUENCE option within the MODEL statement of PROC REG. Another option you can use to highlight special 'outlying' features of individual observations is the COOKD output keyword under PROC REG.

```
Splus Example, 3/31/03
----------------------

> steamdat <- matrix(c(10.98, 35.3, 11.13, 29.7, 12.51, 30.8, 8.40, 58.8,
+   9.27, 61.4, 8.73, 71.3, 6.36, 74.4, 8.50, 76.7, 7.82, 70.7, 9.14, 57.5,
+   8.24, 46.4, 12.19, 28.9, 11.88, 28.1, 9.57, 39.1, 10.94, 46.8, 9.58,
+   48.5, 10.09, 59.3, 8.11, 70.0, 6.83, 70.0, 8.88, 74.5, 7.68, 72.1,
+   8.47, 58.1, 8.86, 44.6, 10.36, 33.4, 11.08, 28.6), ncol=2, byrow=T,
+   dimnames=list(NULL,c("steamuse","temp")))

## This is the dataset used in the little SAS illustration for
## PROC REG above.

> motif()
> plot(steamdat[,2],steamdat[,1], xlab="Temp", ylab="Use",
     main="Draper-Smith Steam Data Example")

## This produces a scatterplot, motivating line-fitting.

> lmtmp <- lm(steamuse ~ . , data=data.frame(steamdat))
> lmtmp
Call:
lm(formula = steamuse ~ ., data = data.frame(steamdat))

Coefficients:
 (Intercept)        temp
    13.62299 -0.07982869

Degrees of freedom: 25 total; 23 residual
Residual standard error: 0.8901245

                  Value Std. Error    t value    Pr(>|t|)
(Intercept)  13.62298927 0.58146349  23.428795 0.00000e+00
       temp  -0.07982869 0.01052358  -7.585697 1.05495e-07

> names(lmtmp)
 [1] "coefficients"  "residuals"     "fitted.values" "effects"
 [5] "R"             "rank"          "assign"        "df.residual"
 [9] "contrasts"     "terms"         "call"
> lines(steamdat[,2], lmtmp$fitted, lty=3)
```

```
> names(summary(lmtmp))
 [1] "call"    "terms"       "residuals"   "coefficients" "sigma"
 [6] "df"      "r.squared"   "fstatistic"  "cov.unscaled" "correlation"
> dim(model.matrix(lmtmp))
[1] 25  2

### Next we do two things to show how to 'interact' with the plot.
### The first is purely graphical: we highlight a few of the points by
### pointing and clicking at them, using "identify":

> identify(steamdat[,2],steamdat[,1])
### Now click successively with left mouse-button over the three
###     uppermost points in the plot and then the three lowermost,
###     and then click middle mouse-button
[1]  3 15 17 11 19  7
> printgraph(file="Steamplot.ps")
### These are the indices of the points clicked on: a great way
###   to identify outliers "visually"
```

More conventionally, we can try to identify outliers according to the "hat matrix":

if X denotes the design-matrix (in this case the 25x2 matrix model.matrix(lmtmp)) for a simple linear regression, for which the fitted variance is

```
> summary(lmtmp)$sigma^2
[1] 0.7923217
> sum(lmtmp$residuals^2)/23
[1] 0.7923217
```

then the theoretical vector of variances for the residuals from the linear-regression fit is

```
> rvar <- { mtmp <- model.matrix(lmtmp)
            0.7923217*diag(diag(25)-mtmp %*% solve(t(mtmp) %*%
               mtmp, t(mtmp))) }
```

So the 'standardized residuals' are:

```
> rstd <- lmtmp$resid/sqrt(rvar) ### standardized residuals
## of which only those with indices 3, 7, 11 seem 'significant':
```

```
> order(rstd)[c(1,25)]
[1] 11  3
> rstd[c(3,7,11)]
        3         7        11
 1.599349 -1.573203 -1.930487
```

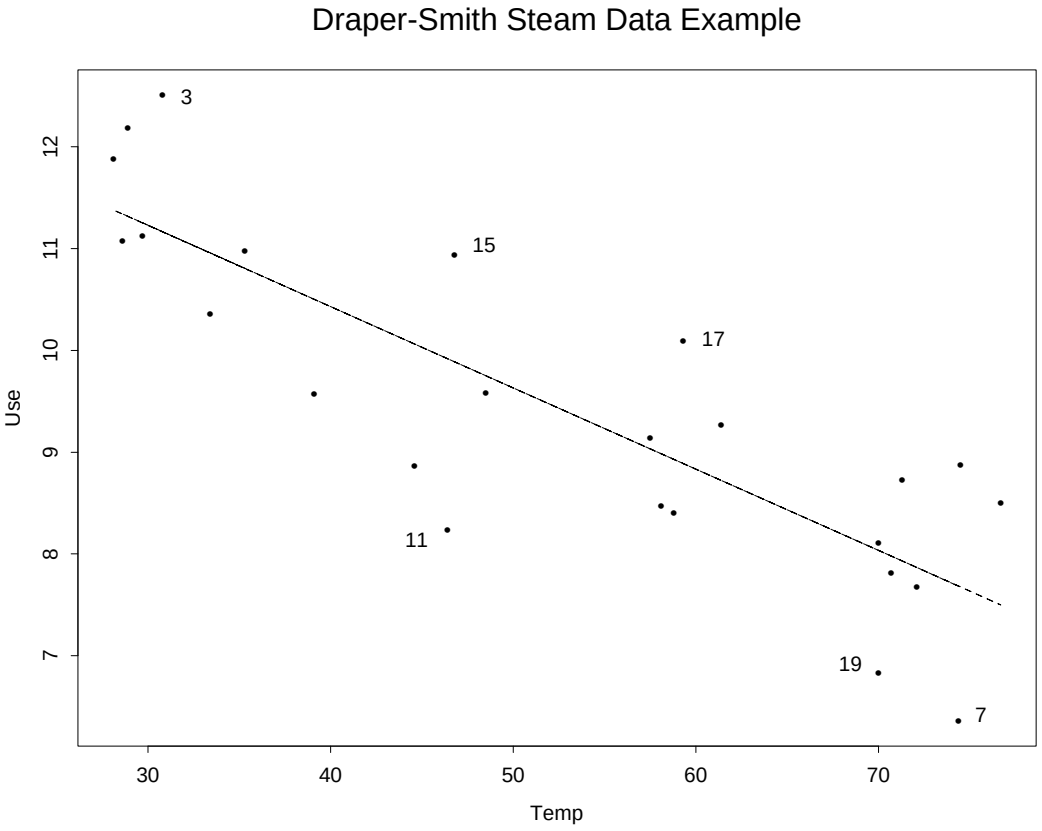


Figure 1: Scatterplot of Draper-Smith Steam Data including Splus fitted line and points highlighted with the **identify** function. Of the selected points, we found the standardized residuals of numbers 3, 7, 11 to be respectively 1.60, -1.57, -1.53.

If we had wanted to identify points by plotting the standardized residuals instead of the row-indices, the Splus command would be:

```
> identify(steamdat[,2],steamdat[,1], labels=round(rstd,2)))
```

10.1 Influential Points & Cook's Distance

Various methods exist to quantify either how badly an individual point is reproduced by a regression model, or how important the individual point is in affecting the values of fitted model parameters. Points which are very special by either of these criteria can be called 'outliers' and can be considered for removal from the dataset before reporting coefficients and assessing quality of fit, although removing points is often a very bad idea because the reality often is that observed populations must be viewed as superpositions of distinct or latent subpopulations which are not easy to recognize in advance.

We have already described under the Splus segment above the calculation of *standardized residuals*, `rstd` in the example. This measures whether the discrepancy between an observation and its predictor are larger than might occur by chance. (Compare its square to percentage points for a chi-square random variable with one degree of freedom.

Another approach to spotting residuals might be to plot lower and upper confidence limits for each of the observations. In Splus, in the plot already displayed in the Figure, we would do this with the statements

```
> lines(steamdat[,2],lmtmp$fitted + 1.645*sqrt(rvar), lty=6)
> lines(steamdat[,2],lmtmp$fitted -1.645*sqrt(rvar), lty=6)
```

Here we are calling points 'extreme' if their standardized residuals are significant (two-sided) at the 10% level.

The value of **Cook's distance** for each observation represents a measure of the degree to which the predicted values change if the observation is left out of the regression. If an observation has an unusually large value for the Cook's distance, it might be worth deleting it from the regression and seeing if the fit is improved. If no significant change in the coefficient estimates or the root MSE occurs, it is best not to delete the point from further analyses. (It might not be a good idea to delete it even if it looks 'influential' in this sense.) There may be no influential cases in a particular regression problem.

In SAS, the way to get a SAS output file with a calculation of predictors, residuals, standardized (or *studentized* residuals, Cook's distance, and lower and upper confidence limits for all of the observation values, the code would be written:

```
PROC REG data=SASstf.steam;
   MODEL steamuse = temp / alpha=.05 INFLUENCE ;
   OUTPUT out=SASstf.steamOUT predicted=steamhat
           residual=steamrsd student=stdresid
           cookd=steamcook  lcl = lowstm ucl = histm;
run;
```

In the resulting output matrix, the **steamcook** column shows largest values (respectively .152, .147, .122) for observations 3, 7, and 20. There is no direct correspondence between absolute studentized residual and Cook's distance, although it is true that the very largest Cook's distances almost always correspond to observations with large studentized residuals.

As a single indication of syntax and results for a `Proc Gplot` in SAS, consider the following:

```
  data steam;
     set SASstf.steamout;
   proc sort;
     by temp;

    symbol1  value = NONE  color=black  i=join  line = 3 ;
    symbol2  value = NONE  color=black  i=j  l=10 ;
    symbol3  value = circle color = black ;
    symbol4  value = square color = black i=j line 6;

    proc gplot data=steam ;
         title "Simultaneous plotting example" ;
           plot  steamuse * temp = 3  lowstm * temp = 1
              histm * temp = 2  steamhat * temp = 4 / overlay legend ;
   run;
```

The output from this SAS code is given in the following figure:

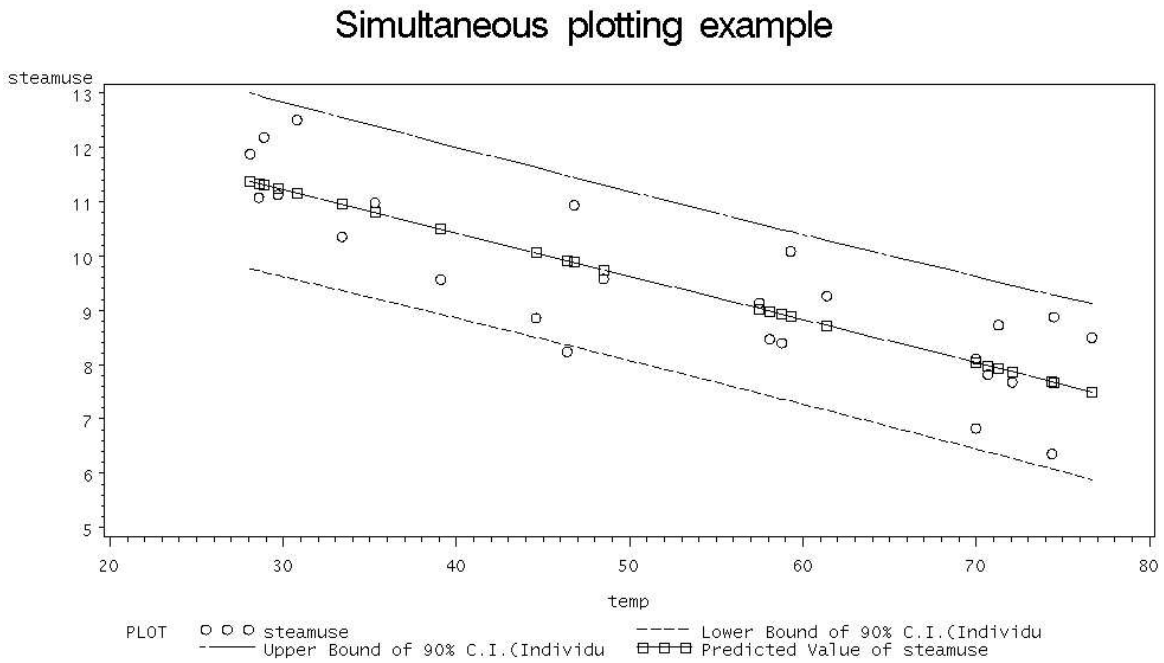


Figure 2: SAS plot of Draper-Smith Steam Data including fitted line and lower and upper prediction-interval points calculated for individual observations.