

Graphs

1 Creating Histograms with Smoothed Lines

Earlier in the year, we learned how to create high resolution histograms using **Proc Univariate**. Today we are going to add a smoothed line to our histograms. These lines, when properly applied, reveal more clearly the underlying trend to our data.

Using the **HIST** option in the **Univariate** procedure we can add a smoothed line to our histogram using a Distribution (such as the normal), or using a Kernel.

When we use the distributional technique, we will specify a statistical distribution such as the Normal. SAS will then compute μ and σ and draw a normal curve overlaid on the histogram with the computed parameters.

The Kernel option (also known as nonparametric density estimation) will superimpose a Kernel density estimates on the histogram. Using a Kernel can be a more effective way to view the data than a histogram or using a distribution, since patterns will not be as effected by bin width or sampling variation. The main option used in Kernel is c = (bandwidth), if c is not specified SAS will choose the optimal bandwidth.

Lets see an example of both smoothing options:

```
Data one;
  set tmp1.school2;
  run;
* Performs univariate stats and histograms for the key variables;
proc univariate data = one;
  var sat skill income loans iq gpalst gpahs;
* Note that the histogram includes a normal distribution overlay;
  HISTOGRAM sat skill income loans iq gpalst gpahs/normal (W=3) ;
  run;

* Adds a kernal smoothed distribution, with kernal width of .5 ;
proc univariate data = one;
  var sat;
  HISTOGRAM sat/normal KERNEL (color = brown C = .5 L = 4 W = 1);
  run;

* Increases the kernal width to 1 and adds lognormal and
exponential distributions ;
proc univariate data = one;
  var income skill;
  HISTOGRAM income skill/normal(color = red) lognormal
  (color=green threshold=-60) exponential(color=purple
```

```

threshold=-60)
  KERNEL (color = black C = 1 L = 1 W = 4) ;
run;

* Sticks with the lognormal distribution and
increases the kernel width to 1.5;

proc univariate data = one;
  var income skill;
  HISTOGRAM income skill /lognormal (color=green threshold=-60)
  KERNEL (C = 1.5 L = 1 W = 3) ;
run;

* Theoretically changes the number of bins;
proc univariate data = one;
  var income skill;
  HISTOGRAM income skill /lognormal (color=green threshold=-60)
  KERNEL (C = 1.5 L = 1 W = 3) maxnbin = 15 ;
run;

```

2 Fitting a smoothed line with Proc Loess

Proc Loess is a procedure that we will use today to generate a fitted line to our data. Loess lines is a nonparametric method for estimating regression lines. We will use **Proc Loess** to obtain a fitted line to a scatter plot. From this plot we can investigate whether there is a linear relationship between our independent and explanatory variables.

Here is an example using the Hanes data set:

```

* BIVARIATE ANALYSES;
* Performs scatterplot with loess line for critical variables;

* Performs a scatter plot;
data one; set tmp1.nhanes; run;

proc gplot data = one;
  plot diabp*weight;
run;

* Obtains the Lowess graphs;
proc loess data = one;;
  model diabp=weight;
  ods output OutputStatistics=weightcheck;
run;

```

```

* The data must be sorted by the X axis to get interpolated points;
proc sort data = weightcheck;
  by weight;
run;

* Overlays the scatter plot and the loess line;
proc gplot data=weightcheck;
* title1 'Predicting Cholesterol by Weight';
  symbol1 color=black value=plus;
  symbol2 color=red interpol=join value=none;
  plot DepVar*weight Pred*weight/ overlay;
run; quit;

```

3 Checking Residuals with Proc Loess

Another way we can use **Proc Loess** is to check the assumptions of a linear regression model. This can be done by fitting a model with $\varepsilon = \mathbf{X}$. We then plot the DepVar and Pred by X (same as above) to check for dependence and constant variance.

Here is an example using the school data:

```

* Performs regression analysis ;
proc glm data = one;;
  model loans = income/solution;
  output out = resid predicted=pgpa residual=resid;
run;

* Note that the residuals are saved into a new file;
PROC loess data = resid;
  model resid = income /std;
  ods output OutputStatistics=grademodel;
run;

* Creates the plus and minus 1 std variables;
DATA grademodel2;
  set grademodel;
  HighPred = pred+predstd;
  LowPred = pred-predstd;
run;
PROC PRINT; RUN;
* remember to sort the data;
proc sort data = grademodel2;
  by income;
run;

```

```
* Obtains the residual plot with loess smoothing function ;
proc gplot data=grademodel2;
  title1 'Plot of Residuals for Predicting Loans as a Function of Income';
  symbol1 color=black value=plus;
  symbol2 color=red interpol=join value=none;
  symbol3 color=gold interpol=join value=none;
  symbol4 color=gold interpol=join value=none;
  plot DepVar*income Pred*income HighPred*income LowPred*income/ overlay;
run; quit;
```