Paper DV-06-2012

Are You Dense? Using Kernel Density Estimation (KDE) to Connect the Dots Amidst Uncertainty

Stephen Crosbie and David Corliss
Magnify, a Division of Marketing Associates, LLC
Detroit, Michigan and Wilmington Delaware

ABSTRACT

Data Visualization: Smoothing data for display using Kernel Density Estimation. Application of the KDE Procedure (provided with Base SAS®) with the Univar Statement provides smoothing for data in graphs. While PROC KDE provides a great resource for smoothing data, the bandwidth selection is fixed and does not account for uncertainty / variance measurements. This paper presents a macro for smoothing data that takes into account uncertainty at each observation point. The method can be used for data with normal and nearly-normal uncertainty distributions.

INTRODUCTION

This paper highlights Kernel Density Estimation and the KDE process for users of SAS® software—including a demonstration of how it adds value to histogram analysis. In the first section, an example shows the value-add of a kernel density estimate. Next, the theory behind kernel density estimation is discussed in familiar language—citing documentation from PROC KDE and other sources. Finally, three ways to compute kernel density estimates within SAS® software are presented, featuring a macro from Corliss using pointwise bandwidth.

MOTIVATION

One method of summarizing univariate data (one variable with many observations) is to use a histogram to show how the data is distributed. As a motivating example, a common scenario of college planning is introduced below.

SUMMARIZING COLLEGE TUITION DATA WITH A HISTOGRAM

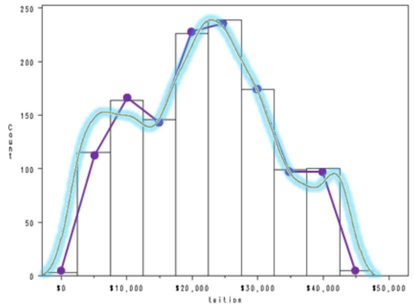
During a conversation with your child's high school counselor, you learn that your over-achieving teenager has set her sights on attending a 4-year private university… "*You don't want to let her down. Do you*?"

As a proud parent and data visualization expert, you can handle this! Using SAS® software and the UNIVARIATE procedure, you create a histogram from tuition data to use for planning (see Figure 1 and the Source Code section)

THE TUITION PICTURE ENHANCED WITH A KERNEL DENSITY ESTIMATE

While a histogram is functionally appropriate to display the wide-ranging tuition data, it can be visually frustrating to examine the distribution since the viewer is forced to mentally "Connect the Dots" between the bins. In addition, the bins sizes and midpoints selected appear to hide what are three subgroups within the distribution.

Instead of leaving this drawing exercise to the viewer's imagination, the graph can be enhanced by adding a kernel density estimate (KDE). Using a KDE to trace a density overlay line not only "Connects the Dots" for the viewer, but also adds-back characteristics of the data that might otherwise go unnoticed—for example, the three subgroups.



```
proc univariate data=work.college;
   histogram tuition
    /  kernel (k = normal
               c = SJPI
               color = green )
       midpoints = 0 to 50000 by 5000
       vscale=count;
   where sector in(2);
run;
```

Figure 1. Histogram and Kernel Density Estimate (enhanced) of 2010-2011 tuition at 4-year private, not-for-profit schools (data below in Source Code section, College Affordability and Transparency Center, 2012)

WHY USE A KERNEL DENSITY ESTIMATE?

As data visualization experts, we would like to accurately summarize data without sacrificing useful information. As seen in the college tuition example, a histogram can be visually frustrating and misleading, especially when bins or midpoints are not appropriately sized or placed (also discussed in the Theory section of this paper).

Kernel density estimates provide a smooth line to follow; however—similar to the “bin sizing” that can distort the data distribution in a histogram—bandwidth selection is an important part of a kernel density estimate. However, there are several automatic bandwidth selection methods available in SAS® software; these are discussed in the next section.

THE THEORY

Understanding how kernel density estimation works helps the data analyst generate a more effective histogram. This section walks through the basic components and calculations for the KDE process provided by SAS® software.

A KERNEL DENSITY ESTIMATE IS A NONPARAMETRIC METHOD

A kernel density estimate is a nonparametric graph—meaning that it lacks an underlying probability density function (Yeh, Shi-Tao, 2004). Instead, it is drawn based on the observations in the data. In other words, a kernel density estimate does not use regression to fit a line to the data. Other familiar, nonparametric graphing methods within SAS® software include the HISTOGRAM option within the UNIVARIATE procedure, the BOXPLOT and the LOESS procedures. As opposed to the above, examples of parametric regression in SAS® software include the GLM, REG, and NLIN procedures.

THE FORMULA BEHIND THE KDE PROCEDURE

While there is no density function, there is a mathematical expression for the KDE procedure:

$$\hat{f}(x) = \frac{1}{\sum_{i=1}^{n} W_i} \sum_{i=1}^{n} W_i \varphi_h(x - X_i)$$

Figure 2a. Kernel Density Estimate Function (SAS®, 2012)

$$\varphi_h(x) = \frac{1}{\sqrt{2\pi}h} \exp\left(-\frac{x^2}{2h^2}\right)$$

Figure 2b. Standard Normal Density Function With Bandwidth Adjustment (SAS®, 2012)

A simple interpretation of the PROC KDE calculation in Figure 2a. is as follows:
For each observation **i**, generate a kernel function **$\phi_h(x)$** centered at each **X_i** on the horizontal axis (see the red-dashed kernels in Figure 3) and compute the kernel density estimate $\hat{f}(x)$ as the weighted sum total of all observations’ kernel functions, each evaluated at the point x on the graph (solid blue line in Figure 3).

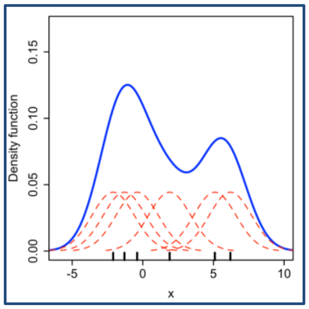


Figure 3. Kernel Density Estimate Based on Six Kernels (Wikipedia contributor Drleft, 2012)

THE KDE PROCEDURE USES THE STANDARD NORMAL DENSITY FUNCTION

As seen in Figure 2b, the KDE procedure uses the standard normal density function for each kernel **$\phi_h(x)$**. The parameter **h** in this function is used for the bandwidth, which affects the smoothness of the graph. In addition to the normal kernel, the UNIVARIATE procedure also supports triangular and quadratic kernels.

When the underlying probability distribution of the data is normal or near normal, it makes sense to use PROC KDE because the data will match the normal shape. Whenever the data are a collection of independent, identically distributed (i.i.d.) observations, the Central Limit Theorem reassures us that the distribution approaches normal as the number of observations increases.

THERE ARE SEVERAL METHODS FOR SELECTING BANDWIDTH

In most kernel density estimation techniques, the kernels have the same shape and bandwidth. While there are several bandwidth selection methods that automatically choose an appropriate kernel size, the most popular is the Sheather-Jones Plug-In (SJPI) method, which is used as a default in the KDE procedure.

Other bandwidth selection methods include Simple Normal Reference (SNR), SNR with Inter-quartile range (SNRQ), Silverman's Rule of Thumb (SROT) and Oversmoothed (OS), which are compared in Figure 4. More comprehensive, theoretical discussion of bandwidth selection is beyond the scope of this paper (Barnes, G.R. and P. B. Cerrito 2000).

An important note: the default bandwidth selection method used by the UNIVARIATE procedure (with the HISTOGRAM statement and KERNEL option) is a different method that minimizes approximate mean integrated standard error (MISE), not SJPI.

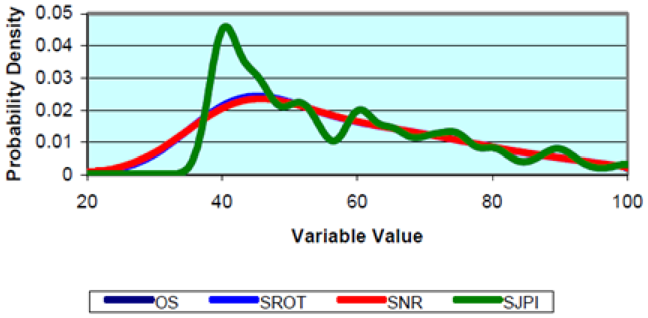


Figure 4. Comparison of Methods of Bandwidth Estimation (Barnes, G.R. and P. B. Cerrito, 2000)

TO ADJUST THE SMOOTHNESS OF THE GRAPH, USE THE BANDWIDTH MULTIPLIER (BWM)

The KDE process provides the BWM= option to allow the user to manually adjust smoothing, beyond the effect of the bandwidth selection method. A BWM value greater than 1 increases smoothing, while a value less than 1 makes the kernel bandwidth narrower. Changing the BWM is comparable to the "bin sizing" exercise used with histograms (see Figure 5 below), but seems easier and has a greater visual effect that makes the data "pop" when set correctly.

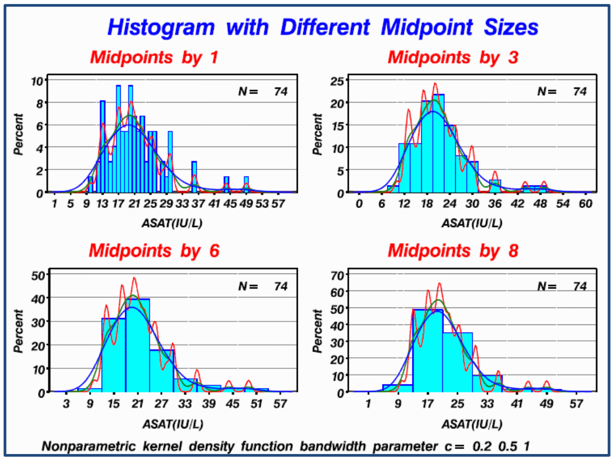


Figure 5a. Various Histogram Bin Sizes (Yeh, Shi-Tao, 2004)

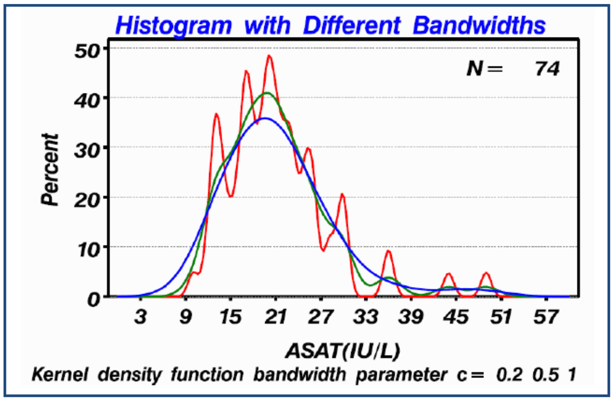


Figure 5b. Various KDE Bandwidths (Yeh, Shi-Tao, 2004)

OTHER BANDWIDTH ADJUSTMENT METHODS

Beyond PROC KDE, there are other ways to adjust bandwidth, including a method featured in this paper. Presented by Corliss at MWSUG in 2010, the method uses uncertainty measurements at each observation (a pointwise bandwidth adjustment). One requirement of using Corliss' macro is that the uncertainty is assumed to be normally distributed, as a normal kernel function is used. Other "adaptive," variable-bandwidth adjustment concepts may exist, but are outside the scope of this paper (Wikipedia).

KDE IN SAS® SOFTWARE

The KDE and UNIVARIATE procedures can create kernel density estimates. Each method has pros and cons; however, for the general user that is interested in ready-made solutions, PROC UNIVARIATE seems to be a more natural choice because it allows for separate control of the histogram bin sizing and midpoints.

THE KDE PROCEDURE

The key benefit of using PROC KDE is that the bandwidth multiplier option is available. If for some reason the overall smoothness of the density needs to be changed from the automatic bandwidth selection, the multiplier works well.

Using the data from Eisenstein in the Source Code section (used by Corliss), here is example for PROC KDE:

```
ods graphics on;
proc kde data=work.records;
   univar mu
   / method=SJPI
     /*BWM=1*/
     NGRID=501
     GRIDL=22500 GRIDU=57500
     /*PLOTS=(DENSITYOVERLAY)*/
     UNISTATS  /*out=outkde*/
     plots=histdensity
     ;
run;
ods graphics off;
```

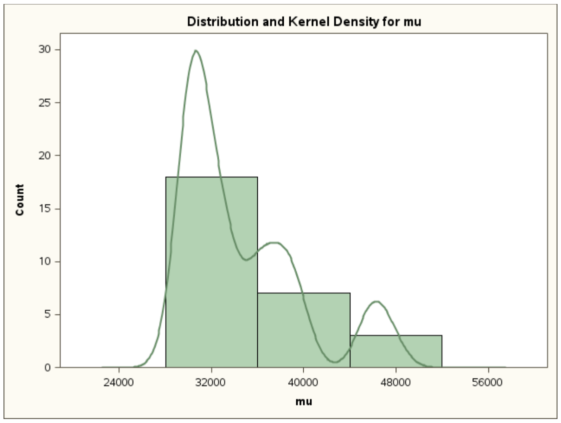


Figure 6. Output From PROC KDE

The UNIVAR statement is used (bivariate is outside the scope of this paper) with the data, with options such as the METHOD= option (for bandwidth selection method) and the NGRID= option (specifies the number of horizontal points to calculate for the density overlay). GRIDL= and GRIDU= serve as limits for the x-axis, while the UNISTATS option produces various statistical outputs. The PLOTS=HISTDENSITY option is used to generate both the KDE and the histogram.

THE UNIVARIATE PROCEDURE

For the purposes of this paper, the key benefit of using PROC UNIVARIATE is that there is more control from the HISTOGRAM statement. In addition, PROC UNIVARIATE can be used for many other purposes (beyond the scope of this paper). Since there is no bandwidth multiplier, the user must manually specify bandwidth values if the automatic selection does not produce a smooth graph as desired.

Using the same data as above, here is the example using PROC UNIVARIATE:

```
proc univariate data=work.records;
   histogram mu
   / kernel
      (k = normal
       c = SJPI /* Default is MISE*/
       /*w = 1*/
       color = green )
     midpoints = 22500 to 57500 by 5000
     vscale=count;
run;
```

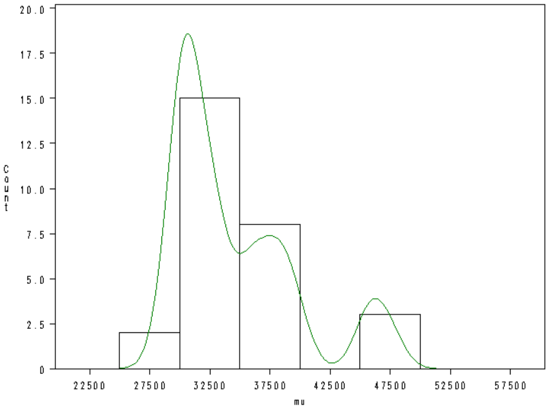


Figure 7. Output from PROC UNIVARIATE

The HISTOGRAM statement is used with the KERNEL option. Sub-options for selecting K = NORMAL density estimate and C = SJPI ensures that the density curve looks like the default from PROC KDE. The MIDPOINTS= option is what allows control of the histogram bin sizes (which PROC KDE lacks), and VSCALE = COUNT sets the vertical scale to match the style of PROC KDE.

POINTWISE UNCERTAINTY

Using the Eisenstein data and Corliss macro from the Source Code below, here is the example of what the “real” density looks like as a result of using the uncertainty measurements for pointwise bandwidth adjustment:

```
goptions reset = all;
ods graphics on;

axis1 order=(22500 to 57500 by 5000)
value=(height=2) label=(height=2 'x');
axis2 order=(0 to .012 by 0.001)
value=(height=2) label=none;
symbol1 value=none i=join c=black
mode=include line=1 w=2 ;
legend1 shape=line(10) value=(height=2
'y') across=4 label=none;

proc gplot data=work.final;
   plot y*x
   / overlay haxis=axis1 vaxis=axis2
legend=legend1;
   where x between 22500 and 57500;
run; quit;
ods graphics off;
```

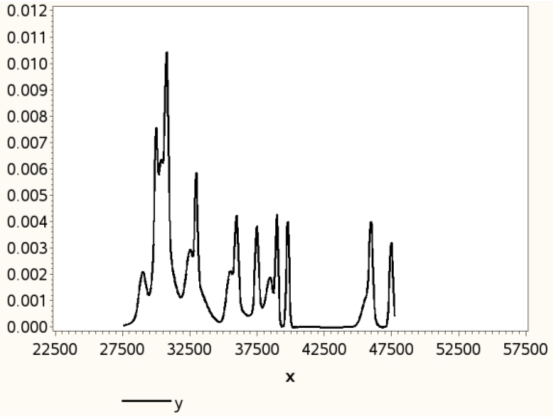


Figure 8. Output from PROC GPLOT and the Corliss KDE Macro for Pointwise Uncertainty

The GPLOT procedure is used to output the density overlay, using the macro from the Source Code section below.

It is important to note that the Corliss KDE macro uses the normal kernel, which assumes that the uncertainty distribution around each observation is normal (or near-normal).

There are opportunities for improvements to the Corliss macro, including adding controls for smoothing by using a bandwidth multiplier option or adding a vertical scaling factor to normalize the graph to the “count” scale used in other kernel density estimation techniques.

SAS SOURCE CODE

Source code for college tuition example:

```
/*Saved tuition data as 120829_College_Tuition2010.txt tab delimited*/

%let inloc=/<filepath>/;
data work.COLLEGE ;
%let _EFIERR_ = 0; /* set the ERROR detection macro variable */
infile "&inloc.120829_College_Tuition2010.txt" delimiter='09'x MISSOVER DSD
lrecl=32767 firstobs=2 ;
informat Sector best32. ;
informat Sector_name $40. ;
informat UnitID $6. ;
informat OPEID $8. ;
informat college $100. ;
informat State $2. ;
informat tuition comma6.0 ;
informat flg_hi $2. ;
informat flg_lo $2. ;
format Sector best12. ;
format Sector_name $40. ;
format UnitID $6.  ;
format OPEID $8. ;
format college $73. ;
format State $2. ;
format tuition dollar8. ;
format flg_hi $2. ;
format flg_lo $2. ;
input
Sector
Sector_name $
```

```
UnitID $
OPEID $
college $
State $
tuition
flg_hi $
flg_lo $
;
if _ERROR_ then call symputx('_EFIERR_',1);  /* set ERROR detection macro variable
*/
run;
```

Source code, including data from Eisenstein, PROC KDE, and the Corliss KDE macro code:

```
/*Hot DB White Dwarfs in Eisenstein et al. 2006*/
data work.records;
seq_num = _n_;
   dummy = 1;
   input name $16. mu 6.0 sigma 4.0;
cards;
J084916.1+013721 29000 250
J093759.5+091653 29550 975
J090232.1+071929 30000 250
J153852.3-012133 30000 1000
J093041.8+011508 30350 1825
J215514.4-075833 30500 750
J141349.4+571716 30500 250
J141258.1+045602 30750 375
J222833.8+141036 30750 1125
J234709.3+001858 30850 1075
J143227.2+363215 30850 925
J212403.1+114230 31000 500
J095256.6+015407 32400 800
J154201.4+502532 32500 250
J123750.4+085526 33000 1000
J164703.4+245129 33000 500
J084823.5+033216 33500 750
J001529.7+010521 35500 250
J090456.1+525030 36000 1250
J211149.5-053938 36000 500
J040854.6-043354 37500 1250
J140159.1+022126 37700 600
J092544.4+414803 38500 250
J134524.9-023714 39000 1000
J074538.1+312205 39800 1000
J113609.5+484318 45700 350
J081546.0+244603 46000 1250
J081115.0+270621 47500 1250
;
run;


%macro dokde(iter=500);
proc sort data=work.records; by mu; run;
proc sort data=work.records; by dummy; run;

/*CREATE MINIMUM AND MAXIMUM BASED ON 2Sigmas above/below each datapoint*/
**** minimum and maximum x-values ****;
data work.x_min;
   set work.records;
   x_min = mu - (2 * sigma);
   keep dummy x_min;
run;
```

```
proc sort data=x_min; by x_min; run;
data work.x_min; set work.x_min; by dummy; if first.dummy; keep dummy x_min; run;
data work.x_max;
   set work.records;
   x_max = mu + (2 * sigma);
   keep dummy x_max;
run;
proc sort data=x_max; by x_max; run;
data work.x_max; set work.x_max; by dummy; if last.dummy; keep dummy x_max; run;

data work.min_max;
   merge work.x_min work.x_max ;
   by dummy;
   x_range = x_max - x_min;
run;
data work.records;
   merge work.records work.min_max;
   by dummy;
run;

**** KDE Process ****;
data work.final;
   set work.records;
   by dummy;

   x = x_min;
   y_i            = (1/(sigma * ( SQRT(2 * constant('pi') ) ) ) )
                     * EXP((-0.5)*( ( (x - mu) / sigma )**2));
   retain y  0;
   y = y + y_i;

   if last.dummy then do;
      output work.final;
   end;
   keep x y ;
run;

%do i=1 %to &iter;
data work.tot;
   set work.records;
   by dummy;
   x = x_min + ((&i. / &iter.) * x_range);
   y_i            = (1/(sigma * ( SQRT(2 * constant('pi') ) ) ) )
                     * EXP((-0.5)*( ( (x - mu) / sigma )**2));
   retain y  0;
   y = y + y_i;

   if last.dummy then do;
      output work.tot;
   end;
   keep x y ;
run;
data work.final;
   set work.final work.tot;
run;
%end;
%mend dokde;

%dokde;
```

CONCLUSION

Using a kernel density estimate in combination with a histogram adds value to the visual presentation of data. SAS® software supports this task in both the KDE and UNIVARIATE procedures. This paper motivates users to familiarize themselves with the theory of kernel density estimation so that they can take control of the graphical output. A further challenge to the innovative programmer would be to use the macro from Corliss as a starting point for implementation of the newer “adaptive” bandwidth adjustment techniques.

REFERENCES

- Barnes, G.R., P. B. Cerrito, The Visualization of Continuous Data Using PROC KDE and PROC CAPABILITY, SUGI 2000 Proceedings, Paper 176-26
  http://www2.sas.com/proceedings/sugi26/p176-26.pdf
- College Affordability and Transparency Center (US DOE), 2012 list and data, accessed 8/31/2012
  http://collegecost.ed.gov/catc/Default.aspx and http://collegecost.ed.gov/catc/resources/CATClists2010.xls
- Corliss, David, Kernel Density Estimation as an Alternative to Binning in the Analysis of Survey Data, MWSUG 2010 Proceedings, Paper 130-2010
  http://www.mwsug.org/proceedings/2010/stats/MWSUG-2010-130.pdf
- Eisenstein, D.J., et al., 2006, ApJ, 132, 676 (Eisenstein et al. 2006)
- SAS, SAS/STAT 9.22 User’s Guide, Details: KDE Procedure – Kernel Density Estimates, accessed 8/31/2012
  http://support.sas.com/documentation/cdl/en/statug/63347/HTML/default/viewer.htm#statug_kde_sect012.htm
- Wikipedia, Variable Kernel Density Estimation, accessed 8/31/2012
  http://en.wikipedia.org/wiki/Variable_kernel_density_estimation
- Wikipedia contributor Drleft, Comparison of a histogram and a kernel density estimate, Wikipedia (image)
  http://upload.wikimedia.org/wikipedia/en/4/41/Comparison_of_1D_histogram_and_KDE.png
- Yeh, Shi-Tao, Grapical Display of Data – A Nonparametric Approach, NESUG 2004 Proceedings – Posters
  http://www.nesug.org/proceedings/nesug04/po/po10.pdf

CONTACT INFORMATION

Your comments and questions are valued and encouraged. Contact the authors at:

Name: Stephen Crosbie and David Corliss
Enterprise: Magnify, a Division of Marketing Associates, LLC
Address: 777 Woodward Ave, Suite 500
City, State ZIP: Detroit, MI 48226
Work Phone: 313-202-6346
Fax: 313-202-6249
E-mail: scrosbie@marketingassociates.com
Web: http://www.marketingassociates.com