

SAS/Graph Annotate Macro Tutorial

Kathy Shelley, Department of Statistics, Iowa State University

ABSTRACT

The utility of SAS/Graph is for production graphs that are run repeatedly at regular intervals. Because the graphs are produced from a programming language, their design can be totally controlled and reproduced without hand-editing. The Annotate Macros enable one to easily program labels to enhance and clarify the default graphs. This tutorial will empower the user to automatically label graph components with a small number of SAS program statements. Examples focus on bar charts, but the concepts can be extended to enable one to create labels for maps, scatterplots, box plots, etc.

INTRODUCTION

Presented are two bar charts that include bar labels that are statistics, such as number of observations, percentages and averages. These statistics are first generated using SAS procedures such as Proc Freq or Proc Summary. Then labels are created using these statistics in an annotate data set. These special types of data sets are most easily created using SAS' annotate macros. Included are sample graphs accompanied by their SAS programs.

MOTIVATION

Many people use spreadsheet programs to generate graphs because they are easy to produce on the fly. One complaint is that the design of these graphs is inflexible- a what-you-see-is-what-you-get product. Most corporations and large institutions produce reports on a periodic basis that include tables and graphs. SAS' annotate macros enable one to use data generated labeling instead of manually changing the labels for each periodic report. This saves time and ensures accuracy.

FUNDAMENTAL MACRO STATEMENTS

An annotate macro data set should initially include the following statements:

```
%annomac;  
%dclanno;
```

The **%annomac** statement makes the annotate macros available to the SAS code. The **%dclanno** statement sets default lengths for the variables included in the annotate data set.

The bar chart examples in this paper use the **%label** macro to place data-based values at the end of each bar. The placement of the label is determined by the values in the data set. This is set by the **%system** macro. In our examples, we use the following **%system** macro:

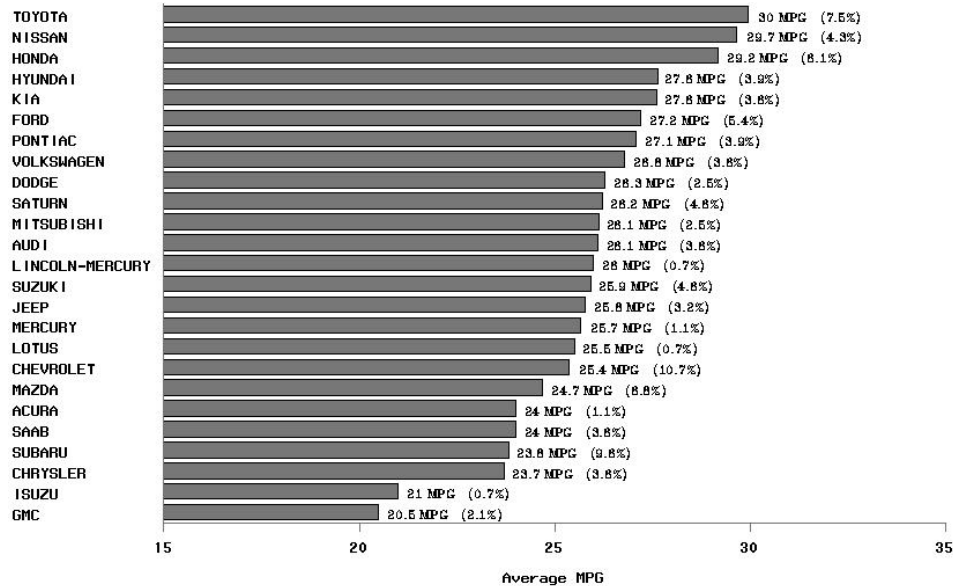
```
%system(2,2,3);  
*system(xsys,ysys,hsys);  
Xsys=2 uses a variable value for horizontal "x" label positioning  
Ysys=2 uses a variable value for vertical "y" label positioning  
Hsys=3 defines size of label text as percent of the graphics  
area
```

FIRST BAR CHART EXAMPLE: A SIMPLE HORIZONTAL BAR CHART WITH ANNOTATED LABELS

The following graph displays average combined city/highway miles per gallon (MPG) for 2007 4 cylinder vehicles sold in the United States. (The source of the data set used for both bar charts was downloaded from the website <http://www.fueleconomy.gov/feg/download.shtml>). At the end of each bar, the average MPG is displayed for each manufacturer, along with a percentage figure that represents the % of the total number of 4 cylinder vehicles.

2007 Average MPG by Manufacturer for 4 Cylinder Vehicles

% represents (manufacturer 4 Cyl)/(total 4 Cyl)



The general format of the label macro is:

%LABEL(x,y,text-string,color,angle,rotate,size,style,position)

The following **%label** statement was used to display labels in this bar chart:

%LABEL (avgmpg,,lbl,black,0,0,2,triplex,6);

Lbl, which concatenates avgmpg and percent, is the text variable that is displayed at the end of each bar. The average, avgmpg, is computed from **Proc Summary** and the percentage is computed from **Proc Freq**. **Size="2"** defines the size of the text to be 2 percent of the graphics area. **Style="Triplex"** is the font used and **Position="6"** means that the text is placed to the right of the (x,y) coordinate.

Following is the program code with comments that generated this graph. Annotate data sets for bar charts require a variable named **midpoint**. In this case, **midpoint=mfr**, the manufacturer. The "y" value is not needed in the **%label** statement because the midpoint variable, mfr, is used for vertical placement of the label. The variable, **avgmpg**, is used for horizontal placement of the label.

* the car data, **data.csv**, came from website <http://www.fueleconomy.gov/feg/download.shtml>;

```
PROC IMPORT OUT= WORK.car_data
```

```
    DATAFILE= "data.csv"
```

```
    DBMS=CSV REPLACE;
```

```
    GETNAMES=YES;
```

```
    DATAROW=2;
```

```
RUN;
```

*compute label values of mean=averageMPG and N=number of observations;

```
proc summary noprint data=car_data;
```

```
class mfr;
```

```
var COMB_MPG__GUIDE_;
```

```
output out=car_means mean=mean n=n;
```

```
run;
```

```
data car_means;
```

```
set car_means;
```

```
if _type_=1;
```

```
drop _type__freq_ ;
```

```
run;
```

```
proc freq data=car_data;
```

```
tables mfr/nocum out=outpct;
```

```

run;
proc sort data=outpct; by mfr;run;
proc sort data=car_means; by mfr; run;
* this step merges the statistics from the summary and freq procedures;
data both;
merge car_means outpct;
by mfr;
run;

data macrodata;
*set car_means;
set both;
%annomac;
*Compiles Annotate macros and makes them available for use;
%dclanno;
*declare all variables ;
%system(2,2,3);
*system(xsys,ysys,hsys);
*xsys=2 uses x data value for horizontal label positioning;
*ysys=2 uses y data value for vertical label positioning;
*hsys=3 uses size of text as percent of graphics area;
avgmpg=round(mean, .1);
pct=round(percent, .1);
lbl= "||compress(avgmpg)|| " MPG" || " (" || compress(pct) || "%)";
*class of vehicle is compact, midsize, suv2wd, suv4wd, pickup2wd, pickup4wd;
midpoint=mfr;
%LABEL (avgmpg, . , lbl, black, 0, 0, 2, triplex, 6);
*%LABEL(x, y, text-string, color, angle, rotate, size, style, position);
* rotate = rotation angle of characters;
* angle = angle of text with respect to horizontal, i.e. 90=perpendicular;
run;

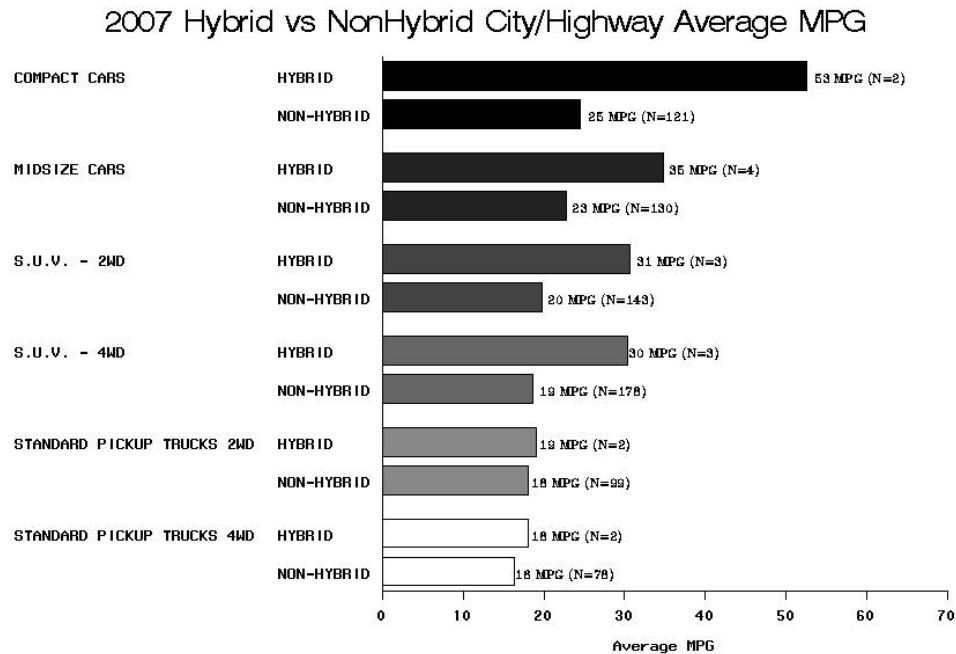
*clean graph catalogue;
proc greplay nofs igout=work.gseg; delete _all_; run; quit;
title j=c "2007 Average MPG by Manufacturer for 4 Cylinder Vehicles";
title2 j=c "% represents (manufacturer 4 Cyl)/(total 4 Cyl)";
goptions ctext=black colors=(gray77);
*note: need to extend x axis beyond max avgmpg value to make room for label;
axis1 order=(15 to 35 by 5)
label=('Average MPG')
minor=none
offset=(0,0);
*axis2 is for mfr;
axis2 label=none;
pattern1 v=solid;
proc gchart data=car_means anno=macrodata;
hbar mfr / descending
sumvar=mean
noframe
raxis=axis1
maxis=axis2
nostats
coutline=black;

run;
quit;

```

SECOND BAR CHART EXAMPLE: A GROUPED HORIZONTAL BAR CHART WITH ANNOTATED LABELS

The following graph compares average combined city/highway miles per gallon (MPG) for 2007 hybrid versus non-hybrid vehicles sold in the United States.



In this example, **Lbl**, which concatenates avgmpg and the number of vehicles, is the text variable that is displayed at the end of each bar. We only need to use **Proc Summary** to obtain both the average and the number of observations.

Following is the program code with comments that generated this graph. This grouped bar chart requires both a group and midpoint variable to enable the placement of the annotate labels. In this case, **midpoint=hybrid** and **group=class**, the type/size of vehicle. The "y" value is not needed in the %label statement because the midpoint variable, hybrid, is used for vertical placement of the label. The variable, **avgmpg**, is used for horizontal placement of the label.

```

data car_data;
length hybrid $10. ;
set car_data;
if _n_=1 then delete;
if hybrid=" " then hybrid="NON-HYBRID";
run;
proc sort data=car_data;
by hybrid class descending COMB_MPG__GUIDE_ car_line ;
run;
data compare_hybrid;
set car_data;
class1=compress(class);
if class1 in("COMPACTCARS", "MIDSIZECARS", "S.U.V.-2WD", "S.U.V.-4WD",
"STANDARDPICKUPTRUCKS2WD", "STANDARDPICKUPTRUCKS4WD");
keep class mfr hybrid car_line COMB_MPG__GUIDE_;

```

```

run;
*compute label values of mean=averageMPG and N=number of observations;
proc summary noprint data=compare_hybrid;
class class hybrid;
var COMB_MPG__GUIDE_;
output out=hybrid_car_means mean=mean n=n stderr=stderr ;
run;
data hybrid_data;
set hybrid_car_means;
if _type_=3;
drop _type__freq_ ;
run;
data macrodata;
set hybrid_data;
*not that hybrid_data contains these variables:
class(y) , hybrid, mean (x), n and stderr from proc summary;
%annomac;
*Compiles Annotate macros and makes them available for use;
%aclanno;
*declare all variables ;
%system(2,2,3);
*system(xsys,ysys,hsys);
*xsys=2 uses x data value for horizontal label positioning;
*ysys=2 uses y data value for vertical label postioning;
*hsys=3 uses size of text as percent of graphics area;
avgmpg=round(mean*100/100);
lbl="||compress(avgmpg)|| " MPG" || " (N="||compress(n)||)";
*class of vehicle is compact, midsize, suv2wd, suv4wd, pickup2wd, pickup4wd;
group=class;
midpoint=hybrid;
%LABEL (avgmpg, . , lbl, black, 0, 0, 2, triplex, 6);
%LABEL(x, y, text-string, color, angle, rotate, size, style, position);
* rotate = rotation angle of characters;
* angle = angle of text with respect to horizontal, i.e. 90=perpendicular;
run;
*clean graph catalogue;
proc greplay nofs igout=work.gseg; delete _all_; run; quit;
title j=c "2007 Hybrid vs NonHybrid City/Highway Average MPG";
goptions ctext=black
colors=(gray00, gray22, gray44, gray66, gray88, grayff;
*note: need to extend x axis beyond max avgmpg value to make room for label;
axis1 order=(0 to 70 by 10)
label=('Average MPG')
minor=none
offset=(0,0);
*axis2 is for hybrid;
axis2 label=none;
*axis3 is for class;
axis3 label=none;
proc gchart data=compare_hybrid anno=macrodata;
hbar hybrid / type=mean
sumvar=COMB_MPG__GUIDE_
noframe
group=class
patternid=group
raxis=axis1
maxis=axis2
gaxis=axis3
nostats
coutline=black;
run;

```

quit;

CONCLUSION

This paper has given two examples of placing data-driven labels on bar graphs with the use of annotate macros. The annotate data set is a SAS data set where labels are created from other SAS data sets using normal data set coding. The annotate macro data set provides positioning and values for the labels. It can be used with any type of SAS data-driven graph. This enables one to create labels from any type of statistics so graphs can display percentages, averages, standard deviations and numbers of observations.

CONTACT INFORMATION

Any comments or questions would be appreciated. Contact the author:

Kathy Shelley
Iowa State University, Department of Statistics
117 Snedecor Hall
Ames, IA 50011
Work Phone: 515-294-7776
Fax: 515-294-4040
E-mail: Kathy@iastate.edu

SAS® and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration. Other brand and product names are trademarks of their respective companies.