

What could ODS graphics do about Box Plot?

Tongda Che, MSD R&D (China) Co. Ltd., Shanghai, China

ABSTRACT

Box Plot is commonly used to graphically present data's distribution. The most common boxplot would show data's median, first quartile, third quartile, minimum point and maximum point. And in many cases, we would need to add other features on the boxplot, like display outliers, combining scatter plot or display summary dataset above the plot...etc. This paper would like to discuss how to accomplish these features with ODS graphics.

Key words: Box Plot, ODS graphics

PART I. BRIEF INTRODUCTION ON ODS GRAPHICS

ODS GRAPHICS HISTORY

As an extension of SAS ODS output system, since SAS 9.2, ODS graphics began to provide flexible ways to generate graphs. Starting from SAS 9.3, ODS graphics have been included in SAS BASE, from then on, people would not need SAS GRAPH license to use this function. Because of the advent of ODS graphics, SAS graph capability thus have tremendously improved. ODS graphics can easily generate graph with good looking and can achieve functions that may be hard in other SAS graph tools.

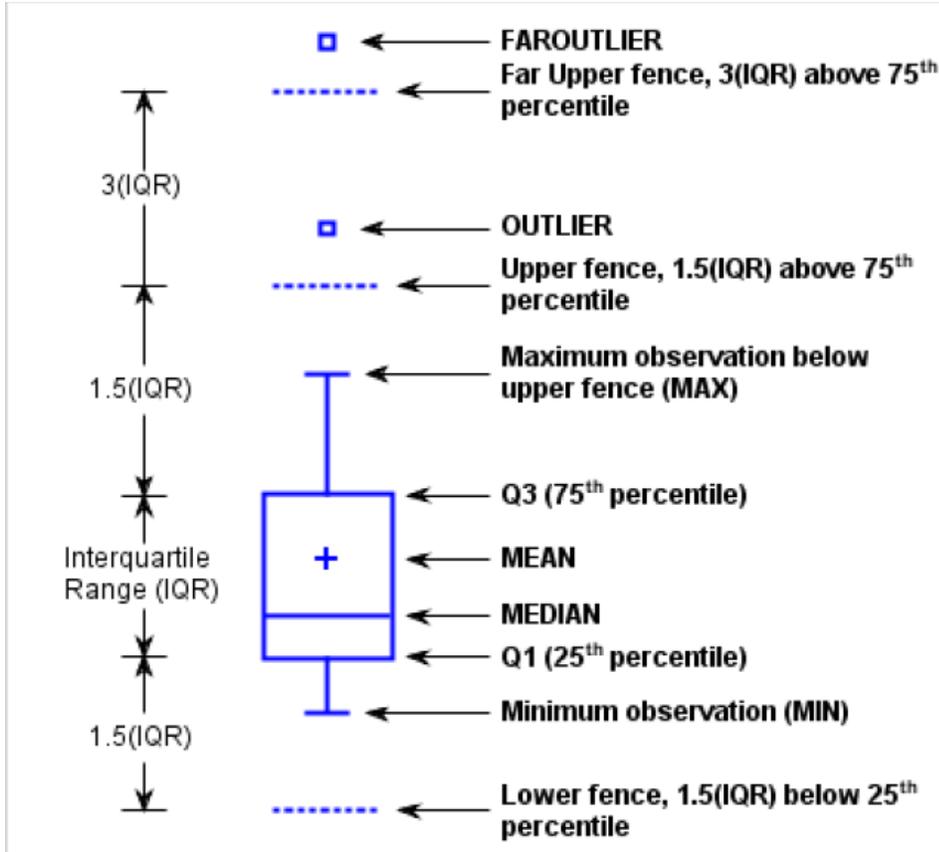
ODS GRAPHICS CATEGORY

ODS graphics could provide graphs under multiple situations:

1. Automatic graph from SAS statistical procedures. With ODS GRAPHICS option ON, graph would be automatic generated when running statistical procedures.
2. Graphs created from Statistical Graphics (SG) procedures. For most commonly used graph, SG procedures could fully accomplish your requirement.
3. Graphs created from Graphic Template Language (GTL). GTL could create graphs exactly based on your specific requirements.
4. Graphs created from the ODS Graphics Designer application. It is a point and click tool to create graph, also a good learning tool for studying GTL. Run the macro %sgdesign, this tool would come out.

ABOUT BOX PLOT

A box plot summarizes the data and indicates the median, upper and lower quartiles, and minimum and maximum values. The plot provides a quick visual summary that easily shows center, spread, range, and any outliers.



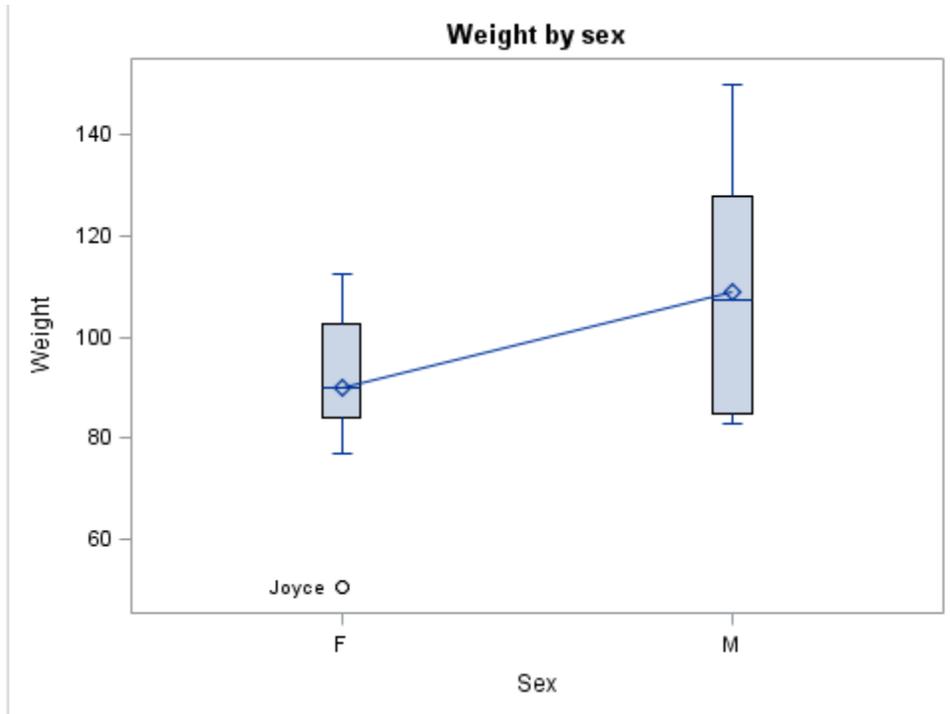
Before the advent of ODS graphics, SAS have methods of BOXPLOT or GPLOT procedure to generate boxplot. In this paper, I would like to discuss how to achieve boxplot with ODS graphs easily.

PART II.VBOX/HBOX STATEMENT IN SG PROCEDURE

Reference code:

```
proc sgplot data=sashelp.class;  
title "Weight by sex";  
vbox weight / category=sex datalabel=name boxwidth=0.1 connect=mean;  
run;  
title;
```

Output:



With SGPLOT/SGPANEL procedures, statement VBOX/HBOX could generate the boxplot easily, VBOX for vertical boxplot and HBOX for horizontal boxplot. Options would be provided to modify the graph output on appearance, axes, data tip, group, label, plot and plot reference.

The above reference code is to display the height distribution categorized by sex based on class dataset in sashelp library. The boxwidth is set to 0.1 while the default is 0.4. Outlier's name is displayed as label. Option 'connect=mean' specify a connect line joins mean from box to box, you could also choose to connect points of median,Q1,Q3,min or max.

OVERLAYING WITH OTHER BOX STYLE

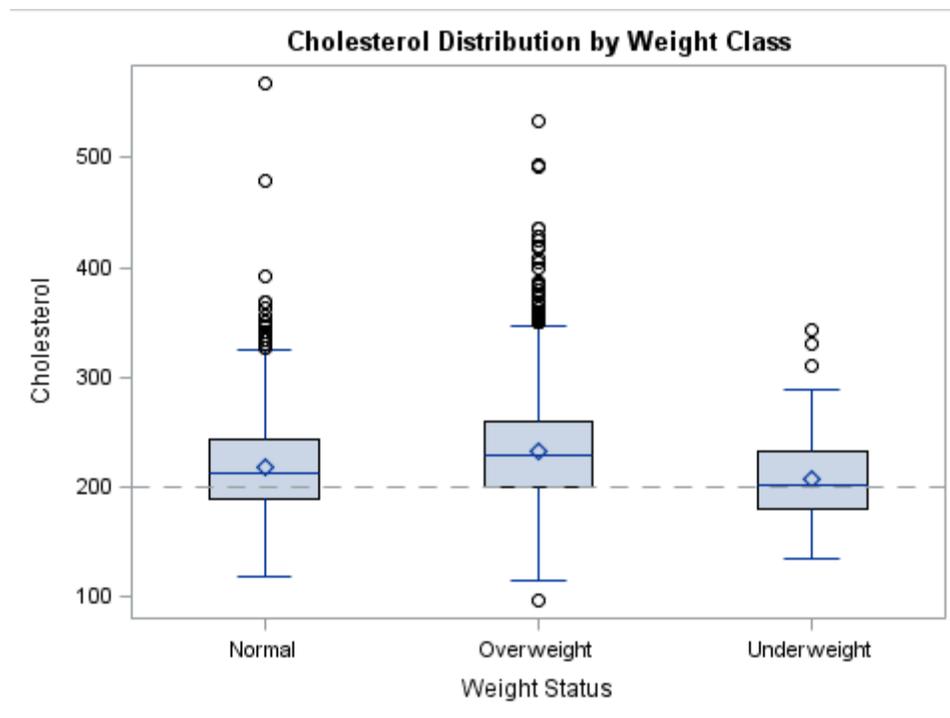
The feature of enable layering one or more plot statement provides SGPLOT/SGPANEL great flexibility to produce specific or complex tables. Before SAS version 9.4, unfortunately boxplot have compatibility restriction thus could only overlaying with boxplot itself. But starting with SAS version 9.4, it could overlay with basic format plots like REFLINE(reference line),TEXT,SCATTER and other.

OVERLAY WITH REFERENCE LINE

Reference Code:

```
proc sgplot data=sashelp.heart;  
title "Cholesterol Distribution by Weight Class";  
vbox cholesterol / category=weight_status;  
refline 200/lineattrs=(pattern=dash);  
run;  
title;
```

Output:



Multiple connect lines

Reference Code:

```
proc means data=sashelp.heart;
  var cholesterol;
  class weight_status;
  output out=heart1 mean=mean median=median;
run;

data heart2;
  keep weight_status mean median cholesterol;
  set sashelp.heart heart1(where=(type_>0));
run;

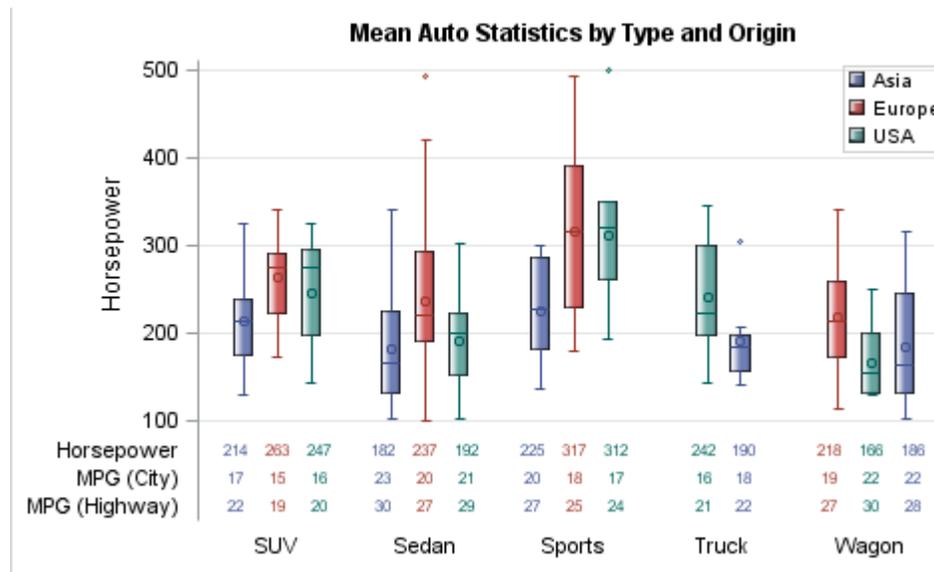
/*--Box plot with connect and group colors--*/
ods graphics / reset ANTIALIASMAX=5300 width=5in height=3in imagename='Box_Group_Multi_Connect';
title 'Cholesterol by Weight';
proc sgplot data=heart2 noautolegend noborder;
  vbox cholesterol / category=weight_status group=weight_status;
  series x=weight_status y=mean / name='mean' legendlabel='Mean';
  series x=weight_status y=median / lineattrs=(pattern=dash) name='median' legendlabel='Median';
  keylegend "mean" "median" / linelength=32 location=inside across=1 position=topright;
run;
```

Output:

<Paper title>, continued

```
title h=10pt 'Mean Auto Statistics by Type and Origin';
proc sgplot data=sashelp.cars(where=(type ne 'Hybrid')) noborder;
  format mpg_city mpg_highway horsepower 3.0;
  vbox horsepower / category=type group=origin name='a'
    groupdisplay=cluster dataskin=gloss
    meanattrs=(size=6) outlierattrs=(size=5);
  xaxistable horsepower mpg_city mpg_highway / class=origin
    classdisplay=cluster stat=mean
    colorgroup=origin location=inside nostatlabel;
  xaxis display=(nolabel noticks noline);
  keylegend 'a' / location=inside position=topright across=1 title='';
  yaxis grid;
run;
```

Output:



PART III.GTL

If simple overlaying feature could not accomplish your request, GTL would be a good choice. Unlike SG procedures which are designed to meet most need, GTL is designed for complex plot.

DISPLAY SUMMARY TABLE

REFERENCE CODE:

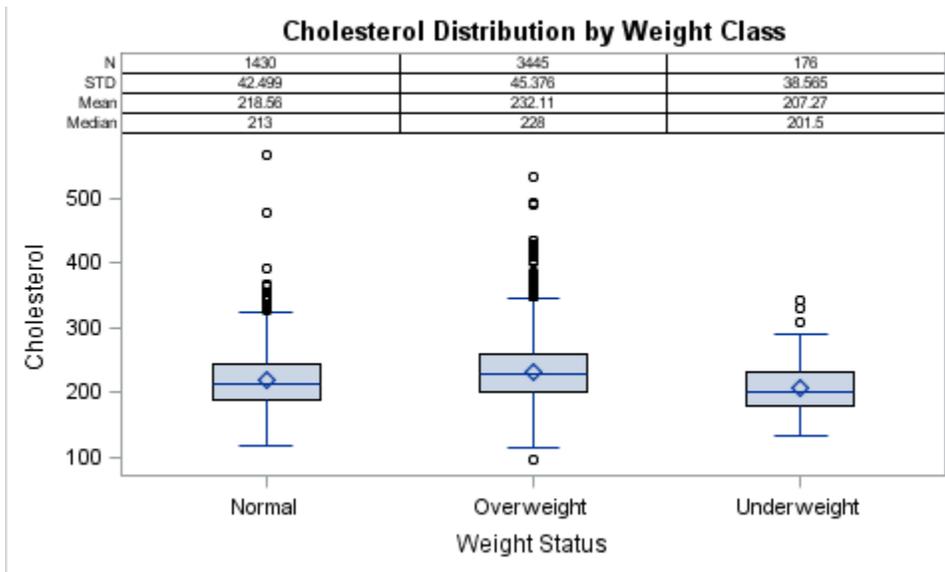
```
proc summary data=sashelp.heart nway qntldef=5;
  class WEIGHT_STATUS;
  var CHOLESTEROL;
  output out=summary(drop=_type_ _freq_) n=N mean=Mean median=Median q1=Q1
  q3=Q3 std=STD / noinherit;
run;
proc sort data=sashelp.heart out=heart1;
  by weight_status;
run;
data heart2;
  merge heart1 summary;
  by weight_status;
run;
title;

proc template;
define statgraph sgdesign;
dynamic x value ;
```

```
begingraph;
  entrytitle halign=center 'Cholesterol Distribution by Weight Class';
  /* entryfootnote halign=left 'Type in your footnote...';*/
  layout lattice / rowdatarange=data columndatarange=data rowgutter=10 columngutter=10;
  layout overlay / xaxisopts=( discreteopts=( tickvaluefitpolicy=splitrotate));
  innerMargin / align=top;
  blockplot x=x block=n /display=(values label outline) valuehalign=center
labelattrs=graphdatatext valueattrs=graphdatatext;
  blockplot x=x block=std /display=(values label outline) valuehalign=center
labelattrs=graphdatatext valueattrs=graphdatatext;
  blockplot x=x block=mean /display=(values label outline) valuehalign=center
labelattrs=graphdatatext valueattrs=graphdatatext;
  blockplot x=x block=median /display=(values label outline) valuehalign=center
labelattrs=graphdatatext valueattrs=graphdatatext;
  endInnerMargin;
  boxplot y=value x=x;
  endlayout;
endgraph;
end;
run;
```

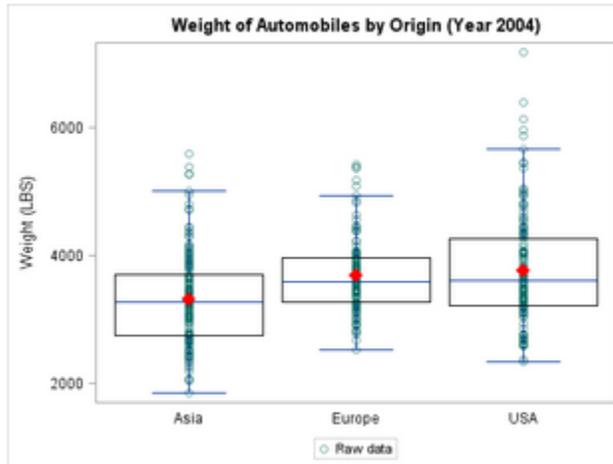
```
proc sgrender data=heart2 template=sgdesign;
dynamic value="CHOLESTEROL" x="'WEIGHT_STATUS'n" ;
run;
```

Output:



PART IV.SOME TIPS

JITTERING



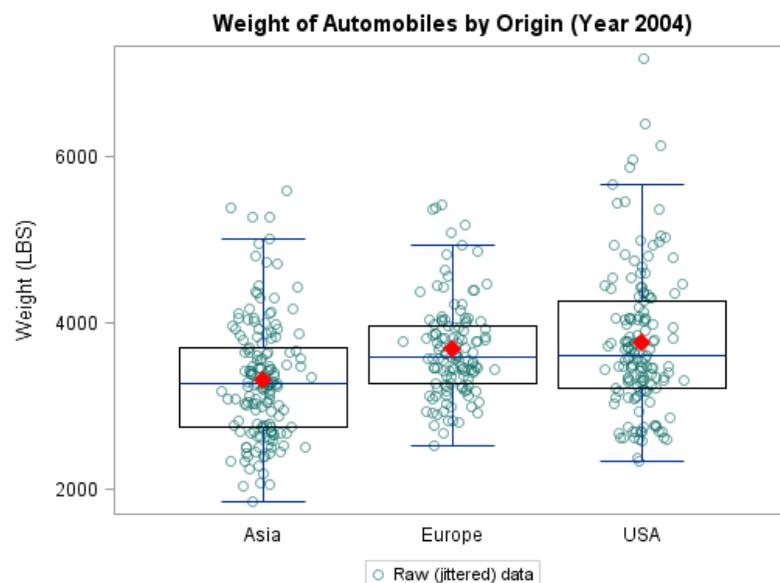
Raw Data with Box overlays (No Jitter)

The above graph is the example of overplotting. If you display a discrete ordinal variable directly, the effect would be like above. TO avoid this, we need the act of jittering.

Before SAS 9.4, people usually use statistical method to control this, for example, the simplest way to achieve this is adding random uniform noise to displayed variable, like: $X = \text{eval}(0.08 * \text{rannor}(57)) + \text{original_x}$

Starting from SAS 9.4 , JITTER option in SCATTER statement could achieve this automatically.

Jittered effect would be like the following graph:



BOXPLOTPARM IN GTL

Sometimes there would be requirements of developing a non-standard boxplot, inBOX that case, it may be not appropriate to use BOXPLOT statement. BOXPLOTPARM statement is a good alternate. The main difference for BOXPLOTPARM statement is that this statement need precomputed input data.

PART V. CONCLUSION

ODS graphics is a good tool to generate boxplot.

REFERENCE

Sanjay Matange.(2016). Clinical Graphs Using SAS. SAS Institute.

SAS® 9.4 Graph Template Language User's Guide, Third Edition

SAS® 9.4 ODS Graphics: Procedures Guide, Sixth Edition

Sanjay Matange. Graphically Speaking: Data Visualization with a focus on SAS ODS Graphics.
<http://blogs.sas.com/content/graphicallyspeaking/>

CONTACT INFORMATION

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

Name: Che, Tongda

Enterprise: MSD R&D (China) Co. Ltd.

E-mail: tong.da.che@Merck.com

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.