Overlaying Scatter Plots on Box Plots in the Presence of Ties

Charles G. Minard[1]

[1]Dan L. Duncan Institute for Clinical and Translational Research, Baylor College of Medicine, Houston, TX

Abstract

Graphical methods are important for efficient and effective visualization of data. Many graphical methods are available, and combining different types of graphics can yield interesting and effective results. Box plots are commonly used to compare the distributions of continuously measured variables across two or more groups, and overlaying a scatter plot on a box plot can provide additional information about distributional differences or similarities. However, ties may be common when the response variable is not measured with high fidelity such as when time is measured in days as an integer value. Random jitter could be used to create separation between points when ties are present in the response variable. However, this can create a figure that is distorted and somewhat difficult to interpret. Evenly distributing ties across the horizontal scale creates a figure that is clearer and more informative.

The purpose of this paper is to present SAS code for generating a scatter plot overlaid on a box plot in the presence of ties. The data set requirements to generate this figure are discussed, and the SAS procedures TEMPLATE and SGRENDER are used to produce the final graphic. A data set comparing the number of days from hospital discharge to readmission stratified by groups of patients is used as an example throughout.

Introduction

Box plots are commonly used to visually compare distributions and summary statistics of a continuously measured response variable across two or more distinct groups. Overlaying a scatter plot on a box plot can provide additional information about distributional differences or similarities. However, this type of overlay is incompatible with the SGPLOT procedure. Pratt [1] previously demonstrated using the TEMPLATE and SGRENDER procedures to address this issue. However, this figure may be unclear or misleading in the presence of ties. Random jitter can be used to create separation between points, but this can also create a figure that is distorted and difficult to interpret.

An improved figure overlaying a scatter plot on a box plot can be accomplished by evenly distributing the distance between points over the horizontal scale. The resulting figure clearly displays each point of the scatter plot and distributes the tied observations evenly over the box plot for each group.

Data set requirements

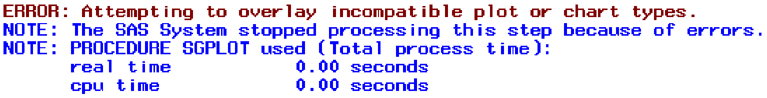
The minimum data set requirements to overlay a scatter plot on a box plot include two variables: *group* and *response*. The *group* variable should consist of sequential integers identifying the group to which each observation belongs. The *response* variable, in general, may be discrete or continuous. However, the intended application here is for continuously measured variables that are measured with low fidelity.

For example, consider a fabricated data set on hospital readmissions. This data set, used throughout this paper, involves the time to readmission among patients discharged from a hospital. The response variable (*Days*) measures the time from hospital discharge to readmission in days, and it consists of integer values from 0 to 30. The grouping variable (*Group*) is coded as 0, 1, or 2. The complete data set is presented in Table 1 in the appendix, and the name of the data set is *mydata*.

Incompatibility

The SGPLOT procedure is capable of overlaying many different types of plots. However, scatter plots and box plots are incompatible because the VBOX statement assumes a categorical variable on the x-axis while the SCATTER statement assumes a continuous x-axis. This is true even when the CATEGORY option on the VBOX statement specifies an integer value variable. The following code attempts to overlay a scatter plot on a vertical box plot, but an error message is received.

```
proc sgplot data=mydata;
  vbox days / category=group nooutliers;
  scatter x=group y=days;
run;
```



```
ERROR: Attempting to overlay incompatible plot or chart types.
NOTE: The SAS System stopped processing this step because of errors.
NOTE: PROCEDURE SGPLOT used (Total process time):
      real time           0.00 seconds
      cpu time            0.00 seconds
```

Minimum SAS Code

The following SAS code provides the minimum level of information required to overlay a scatter plot on a box plot using the TEMPLATE and SGRENDER procedures. This code creates a graph template named **scatterbox**, defines the LAYOUT, BOXPLOT and SCATTER statements, and renders the code. Note that the DISPLAY= option for the BOXPLOT statement specifically excludes outliers. This intentional exclusion is to avoid confusion between markers for outliers resulting from the box plots and markers that result from the scatter plots.

```
proc template;
  define statgraph scatterbox;
    begingraph;
      layout overlay;
        boxplot x=group y=days / display=(mean median caps);
        scatterplot x=group y=days;
      endlayout;
    endgraph;
  end;
proc sgrender data=mydata template=scatterbox;
run;
```

Figure 1 is the product of the above code. If no ties are present in the data, then the scatter plot is appropriate. However, there are multiple instances of ties within each group in the hospital readmissions data set. Thus, the scatter plot in Figure 1 contains many overlapping data points that are indistinguishable from one another. The box plots are informative with respect to the distribution of the data points and summary statistics, but the scatter plots are not completely representative of the respective distributions.

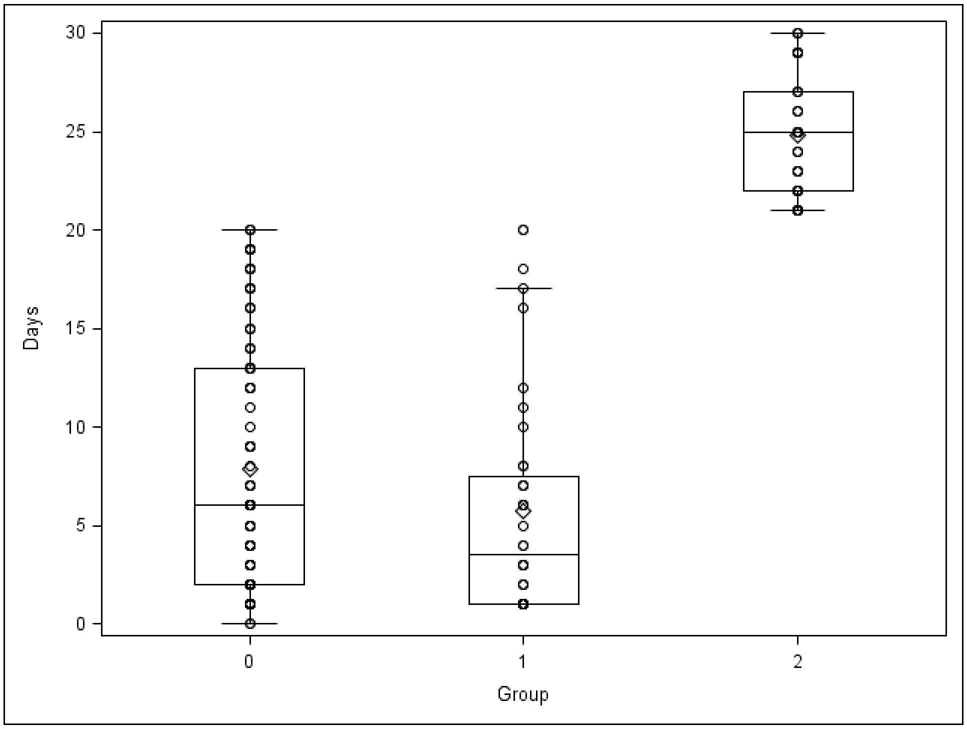


Figure 1: Overlapping data points.

Random Jitter

The scatter plot in Figure 1 above is generated by defining the x-axis by the grouping variable in the BOXPLOT and SCATTERPLOT statements (i.e., X=group). In our example, the only possible values for the grouping variable are 0, 1, and 2 for every observation in the data set. Thus, marker overlap occurs for identical values in the response variable. However, random jitter could be used to create separation between data points. The following DATA step creates a new variable that is equal to the grouping variable (an integer) plus some random value.

```
data mydata_random;
  set mydata;
  shift=group+rannor(0)/10;
run;
```

The new data set (mydata_random) includes a new variable (*shift*) which is set equal to the grouping variable plus some random value generated from a normal distribution. Now, we can replace the earlier SAS code with the following statements.

```
proc template;
  define statgraph scatterbox;
    begingraph;
      layout overlay /
          x2axisopts=(display=(line) linearopts=(viewmin=0 viewmax=2));
        boxplot x=group y=days /
          xaxis=x
          display=(mean median caps);
        scatterplot x=shift y=days /
          xaxis=x2;
      endlayout;
    endgraph;
  end;
proc sgrender data=mydata_random template=scatterbox;
run;
```

Several important changes were made compared with the earlier code. First, note that primary x-axis is defined by the box plot (XAXIS=X) and a second x-axis has been created for the scatter plot (XAXIS=X2). Display options for the second x-axis are controlled through the X2AXISOPTS option in the LAYOUT statement. Here, tick marks and numerical values are excluded from the figure by specifying DISPLAY=(LINE). The data range of the axis is specified by VIEWMIN and VIEWMAX which respectively represent the minimum and maximum values of the grouping variable. Also, the new variable (*shift*) is only defined for the scatter plot. The original grouping variable is used for the box plot because the box plot assumes the x-axis is represented by a discrete, categorical variable.

Figure 2 presents the results of executing the new commands. The data points have been shifted about the horizontal center of each box plot by some random distance. The scatter plots create generally distinguishable points, but the distributions remain difficult to compare and interpret. Furthermore, it is possible that a random distance is acquired that is sufficiently large enough to shift the data points into the vicinity of an adjacent group or out of the range of the x-axis entirely.

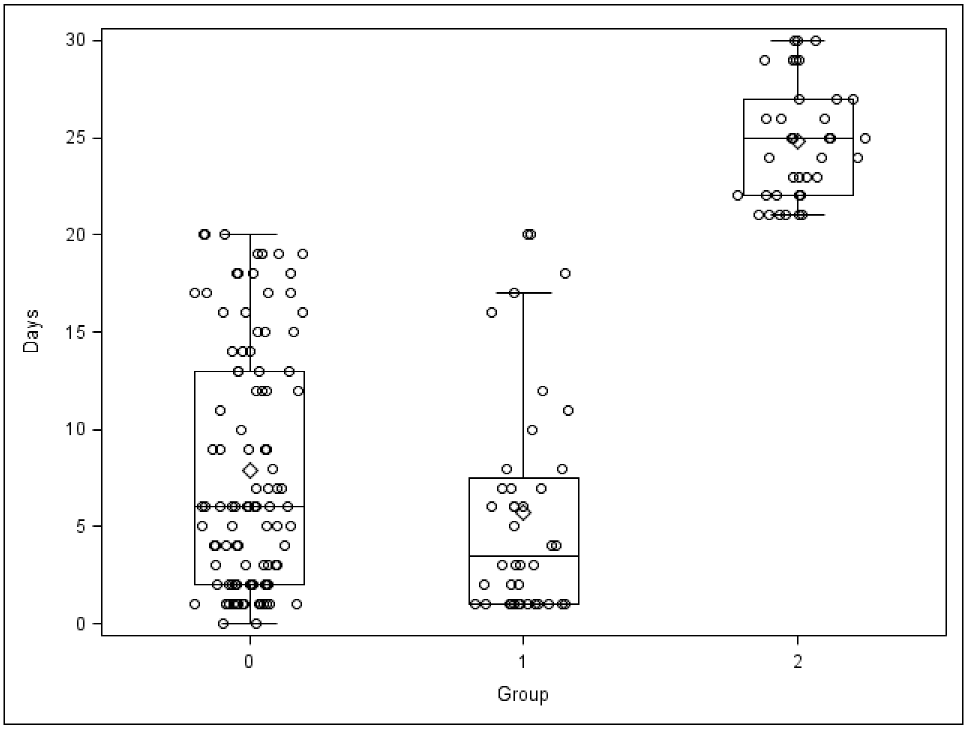


Figure 2: Random dispersion.

ScatterBox macro

Rather than shifting the data points by some random distance, a clearer figure can be generated by evenly spacing identical data points and centering the points over the horizontal scale within each box plot. The ScatterBox macro performs the data manipulation steps required, and is executed using the following macro call function.

%scatterbox(dsn= , group= , response= , delta=);

The complete macro is available in the Appendix. The four arguments that are requested by the macro include:

1)	dsn	Data set name
2)	group	Name of grouping variable
3)	response	Name of response variable
4)	delta	Distance between markers on figure (default=0.01)

The macro counts the number of repeated observations within each group, evenly distributes the markers, and centers the distribution on the box plot within each group. The distance between markers is determined by *delta*. The default value is 0.01, but his value may need to be adjusted depending on the data set. Figure 3 presents the figure for the hospital readmissions data set using the SCATTERBOX macro assuming *delta* = 0.035.

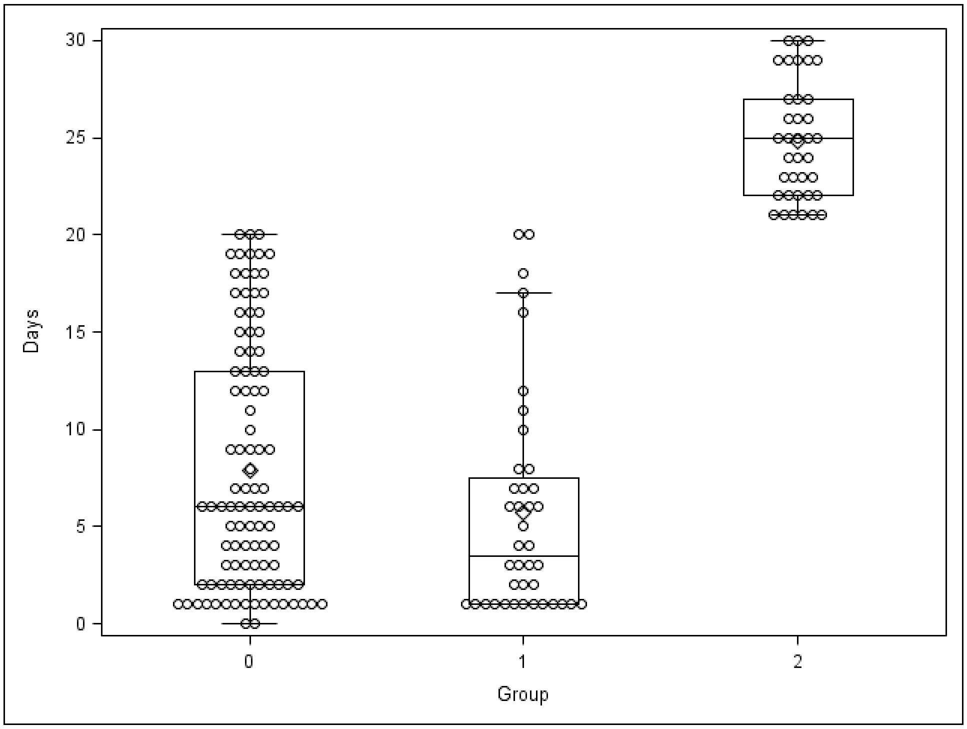


Figure 3: Evenly distributed over horizontal scale.

Customized figures may be created by editing the statements in the TEMPLATE and other procedures. For example, the following TEMPLATE, FORMAT, and SGRENDER procedures were used to generate Figure 4.

```
proc template;
  define statgraph scatterbox;
    begingraph;
      layout overlay /
        xaxisopts=(label='Group' type=discrete)
        x2axisopts=(display=(line)
          linearopts=(viewmin=&grpmin viewmax=&grpmax))
        yaxisopts=(label=”&response”);

        boxplot x=&group y=&response /
          xaxis=x
          display=(caps mean median connect)
          meanattrs=(symbol=trianglefilled color=red)
          connectattrs=(color=red)
          ;
        scatterplot x=shift y=&response /
          xaxis=x2
          markerattrs=(color=charcoal)
          ;
      endlayout;
    endgraph;
  end;
run;
proc format;
  value group 0='A' 1='B' 2='C';
run;
proc sgrender data=scatterbox2 template=scatterbox;
  format group group.;
run;
```

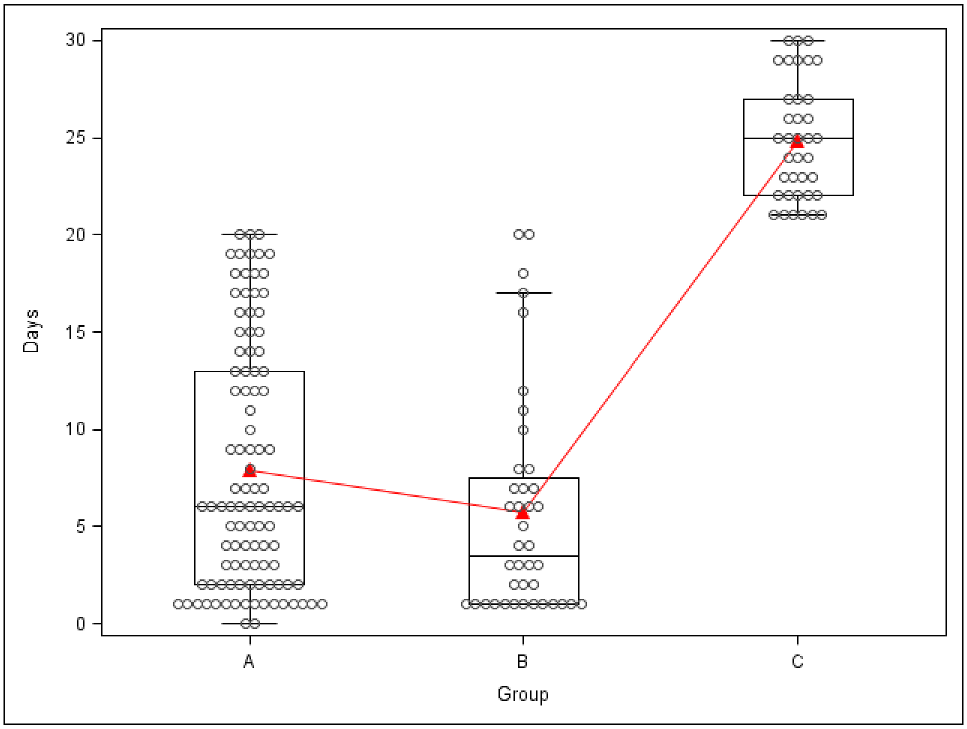


Figure 4: Customized figure.

Conclusions

Overlaying scatter plots on box plots can produce informative and comparable figures when the response variable is continuously measured. The TEMPLATE and SGRENDER procedures may be used to generate such figures, but appropriate data modification may be required to generate informative results, particularly in the presence of ties. The %SCATTERBOX macro, presented here, performs the necessary data manipulation and generates the figure. Evenly distributing tied observations over the horizontal scale within each box plot produces a clear figure that is easily interpretable and meaningful.

Acknowledgments

This work was funded, in part, by the Dan L. Duncan Institute for Clinical and Translational Research at Baylor College of Medicine, Houston, TX. The author would also like to thank Meng-Fen (Jessie) Wu for her assistance and insight.

References

[1] Jesse M Pratt. The graph template language: Beyond the SAS/GRAPH® procedures. *SAS Global Forum 2012*, Paper 285-2012, 2012.

Author Contact Information:
Charles G. Minard, PhD
Dan L. Duncan Institute for Clinical and Translational Research
Baylor College of Medicine
One Baylor Plaza, BCM 122
Houston, TX 77030
minard@bcm.edu

Appendix

Hospital Readmissions Data Set

Table 1: Number of days to readmission (*mydata*).

ID	Group	Days	ID	Group	Days	ID	Group	Days	ID	Group	Days
1	0	1	46	2	24	91	0	18	136	0	19
2	1	1	47	1	6	92	2	23	137	0	9
3	1	17	48	2	30	93	2	25	138	1	1
4	0	6	49	2	22	94	2	21	139	0	19
5	1	6	50	2	22	95	0	1	140	0	19
6	0	17	51	1	4	96	0	1	141	0	13
7	2	21	52	0	7	97	1	7	142	1	1
8	1	1	53	0	3	98	1	20	143	1	2
9	0	7	54	0	6	99	0	2	144	1	8
10	0	1	55	2	22	100	1	5	145	0	7
11	0	17	56	0	1	101	0	2	146	0	0
12	2	22	57	2	23	102	0	18	147	1	3
13	1	1	58	1	6	103	0	1	148	2	22
14	0	12	59	2	27	104	0	16	149	1	3
15	0	9	60	1	1	105	0	13	150	0	2
16	0	16	61	2	21	106	0	4	151	0	20
17	0	12	62	0	4	107	0	1	152	0	6
18	0	14	63	1	7	108	0	5	153	0	8
19	0	18	64	0	12	109	0	15	154	0	14
20	0	6	65	0	5	110	1	10	155	2	29
21	2	23	66	0	0	111	2	29	156	0	2
22	2	24	67	1	6	112	0	1	157	0	1
23	0	16	68	2	29	113	0	1	158	0	2
24	0	2	69	2	25	114	0	9	159	2	29
25	0	1	70	0	6	115	0	2	160	2	23
26	1	1	71	0	6	116	2	26	161	1	2
27	1	8	72	0	2	117	1	1	162	1	1
28	0	2	73	0	13	118	2	21	163	1	7
29	1	3	74	0	3	119	0	20	164	0	6
30	0	5	75	0	3	120	2	30	165	0	6
31	2	24	76	1	16	121	0	7	166	0	2
32	0	3	77	0	10	122	0	4	167	0	6
33	1	11	78	0	6	123	2	26	168	1	12
34	2	25	79	0	4	124	2	25	169	0	1
35	0	19	80	0	14	125	0	11	170	0	4
36	1	1	81	2	29	126	2	30	171	0	19
37	0	15	82	0	17	127	1	1	172	0	20
38	0	3	83	0	1	128	0	6	173	2	21
39	1	2	84	0	1	129	2	26	174	0	5
40	2	27	85	1	18	130	0	2	175	1	4
41	2	21	86	1	1	131	0	3	176	0	17
42	0	1	87	2	27	132	1	20	177	0	15
43	2	25	88	0	9	133	0	9	178	0	4
44	1	1	89	0	1	134	0	12	179	0	5
45	0	13	90	1	3	135	0	18			

SCATTERBOX Macro

```
/*The purpose of this macro is to produce a scatter plot overlaid on a
box plot in the presence of ties.*/
%macro ScatterBox (dsn, group, response, delta);
  %if (&dsn=) %then %do;
    data _null_;
      file print;
      put ”ERROR: Please specify the data set name
 (macro variable: dsn).”;
    run;
    %return;
  %end;

  %if (&group=) %then %do;
    data _null_;
      file print;
      put ”ERROR: Please specify the name of the grouping variable
 (macro variable: group).”;
    run;
    %return;
  %end;

  %if (&response=) %then %do;
    data _null_;
      file print;
      put ”ERROR: Please specify the name of the response variable
 (macro variable: response).”;
    run;
    %return;
  %end;

  %if (&delta=) %then %let delta=0.01;

  /*Sort the data set by the grouping variable then the response variable*/
  proc sort data=&dsn; by &group &response; run;

  /*Create macro variables to identify the minimum and maximum group values*/
  data _null_;
    set &dsn;
    if _n_=1 then call symputx(’grpmin’,&group);
    call symputx (’grpmax’,&group);
  run;

  /*Create a counter that counts the number of ties by observation value
  within each group*/
  data scatterbox1;
    set &dsn;
    by &group &response;

    if first.&response then count=1;
    else count+1;
  run;

  /*Sort the Scatterbox1 data set by group, response value, and descending counts*/
  proc sort data=scatterbox1; by &group &response descending count; run;

  /*Create a new data set and uniformly distribute ties over the horizontal axis*/
  data scatterbox2;
    set scatterbox1;
    by &group &response;

    retain maxcount;
    if first.&response then maxcount=count;

    shift=&group+&delta*(count-1)-(0.5*maxcount*&delta)+.5*&delta;
  run;

  /*Sort final data set*/
  proc sort data=scatterbox2; by &group &response count; run;

  /*Create template for figure*/
  proc template;
    define statgraph scatterbox;
      begingraph;
        layout overlay /
          xaxisopts=(label=’Group’ type=discrete)
          x2axisopts=(display=(line) linearopts=(viewmin=&grpmin viewmax=&grpmax))
          yaxisopts=(label=”&response”);
          boxplot x=&group y=&response /
            xaxis=x
            display=(caps mean median);

          scatterplot x=shift y=&response /
            xaxis=x2;
        endlayout;
      endgraph;
    end;
  run;

  /*Apply template to modified data set*/
  proc sgrender data=scatterbox2 template=scatterbox;
  run;

%mend ScatterBox;
```