Converting Annotate to ODS Graphics. Is It Possible?

Philip R Holland, Holland Numerics Limited

ABSTRACT

In the previous chapter I described how many standard SAS/GRAPH® plots can be converted easily to ODS Graphics, by using simple PROC SGPLOT or SGPANEL code. SAS/GRAPH Annotate code would appear, at first sight, to be much more difficult to convert to ODS Graphics, but, making use of its layering features, many Annotate plots can be replicated in a more flexible and repeatable way. This chapter will explain how to convert many of your Annotate plots, so they can be reproduced using Base SAS®.

INTRODUCTION

This chapter will compare the output from commonly used Annotate and SAS/GRAPH code for producing annotated graphs with equivalent code that just uses ODS Graphics. Each ODS Graphics program will be rated as follows:

- Easy: Replacing one Annotate data set and one SAS/GRAPH procedure statement with an ODS Graphics procedure. The conversion may also require some simple pre-processing of the input data.

- Difficult: Replacing one Annotate data set and one SAS/GRAPH procedure with code containing PROC TEMPLATE and PROC SGRENDER, or the conversion requires extensive pre-processing of the input data.

- Impossible: There is currently no corresponding ODS Graphics procedure in that version of SAS software to replicate the Annotate and SAS/GRAPH plot. However, the annotated plots selected for this paper are all possible to achieve with ODS Graphics, so none of the selection will be rated as impossible.

ERROR BARS

This type of plot is often created using a scatter plot and a fair amount of annotation. The sample data for SAS/GRAPH (**class_error_classic**) and ODS Graphics (**class_error_ods**) is generated using the following code:

```
PROC SORT DATA = sashelp.class OUT = class_error;
    BY sex height;
RUN;

PROC SUMMARY DATA = class_error NWAY;
    CLASS sex;
    VAR weight;
    OUTPUT OUT = class_error_se STDERR = weight_se;
RUN;

DATA class_error_classic (KEEP = sex height value)
     class_error_ods (KEEP = sex height value value_upper value_lower)
     ;
    MERGE class_error class_error_se;
    BY sex;
    value = weight;
    value_upper = value + weight_se;
    value_lower = value - weight_se;
    OUTPUT class_error_ods;
    OUTPUT class_error_classic;
```
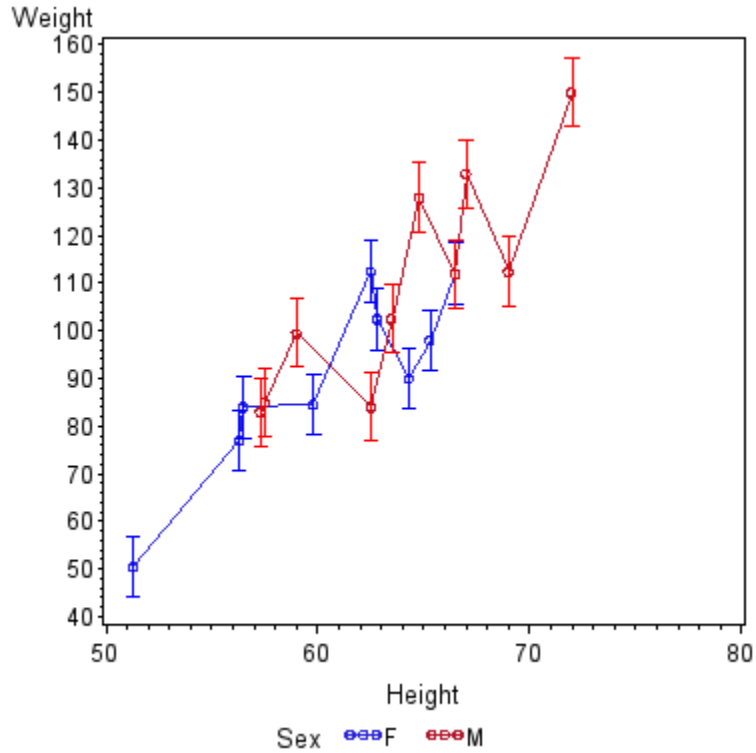
```
     value = value_upper;
     OUTPUT class_error_classic;
     value = value_lower;
     OUTPUT class_error_classic;
   RUN;
```

SAS/GRAPH from SAS 9.2

Error Bar plots can be created using PROC GPLOT and Annotate.


```
%LET height_offset = 0.3;

DATA class_error_anno;
    SET class_error_ods;
    BY sex;
    LENGTH function $8
           color $20
           when xsys ysys $1
           x y 8
           ;
    xsys = '2';
    ysys = '2';
    when = 'A';
    IF sex = 'M' THEN color = 'RED';
    ELSE color = 'BLUE';
    function = 'MOVE';
    x = height - &height_offset.;
    y = value_upper;
    OUTPUT;
```

```
         function = 'DRAW';
         x = height + &height_offset.;
         y = value_upper;
         OUTPUT;
         function = 'MOVE';
         x = height;
         y = value_upper;
         OUTPUT;
         function = 'DRAW';
         x = height;
         y = value_lower;
         OUTPUT;
         function = 'MOVE';
         x = height - &height_offset.;
         y = value_lower;
         OUTPUT;
         function = 'DRAW';
         x = height + &height_offset.;
         y = value_lower;
         OUTPUT;
    RUN;

    PROC GPLOT DATA = class_error ANNO = class_error_anno;
        SYMBOL V = CIRCLE I = JOIN;
        PLOT weight * height = sex / VAXIS = 40 TO 160 BY 10;
        LABEL weight = "Weight";
    RUN;
```
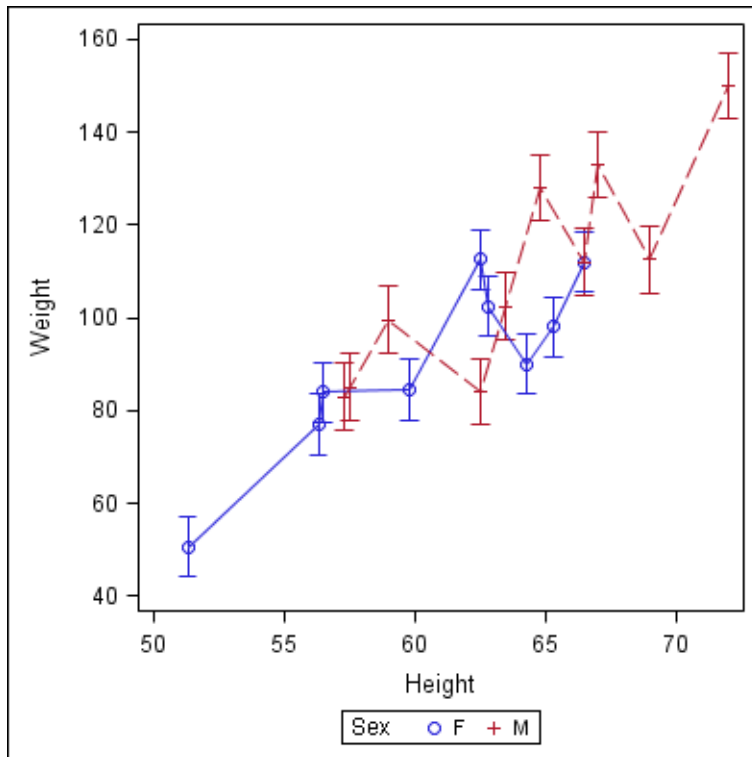
ODS Graphics from SAS 9.2 (Easy)

As was seen in the previous chapter, the error bars are drawn using the YERRORUPPER= and
YERRORLOWER= parameters.

```
PROC SGPLOT DATA = class_error_ods;
    SCATTER Y = value X = height / GROUP = sex YERRORUPPER = value_upper
                                             YERRORLOWER = value_lower;
    SERIES Y = value X = height / GROUP = sex;
    LABEL value = "Weight";
RUN;
```

POINT LABELS

The sample data for SAS/GRAPH (**class_point_classic**) and ODS Graphics (**class_point_ods**) is generated using the following code:
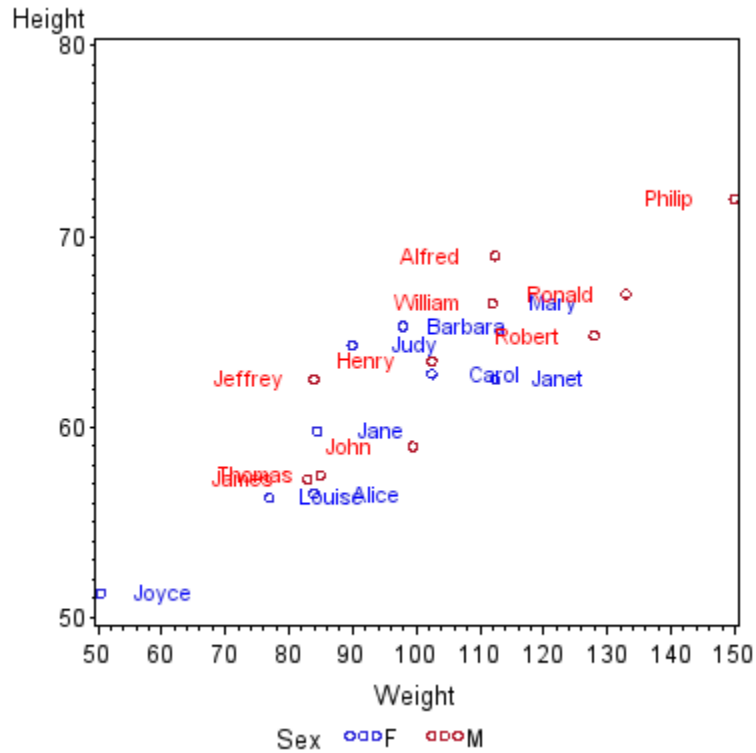
```
PROC SORT DATA = sashelp.class OUT = class_point;
    BY sex height;
RUN;

DATA class_point_ods;
    SET class_point;
    If sex = 'F' THEN weight2 = weight + 10;
                ELSE weight2 = weight - 10;
RUN;
```

SAS/GRAPH from SAS 9.2

Point Label plots can be created using PROC GPLOT and Annotate.

```
DATA class_point_anno;
    SET class_point_ods;
    BY sex;
    LENGTH function $8
           color $20
           position when xsys ysys hsys $1
           x y size 8
           ;
    xsys = '2';
    ysys = '2';
    hsys = 'D';
    when = 'A';
    size = '8';
    IF sex = 'M' THEN color = 'RED';
    ELSE color = 'BLUE';
    function = 'LABEL';
    x = weight2;
    y = height;
    text = name;
    position = '+';
    OUTPUT;
RUN;

PROC GPLOT DATA = class_point ANNO = class_point_anno;
    SYMBOL V = CIRCLE;
    PLOT height * weight = sex;
RUN;
```
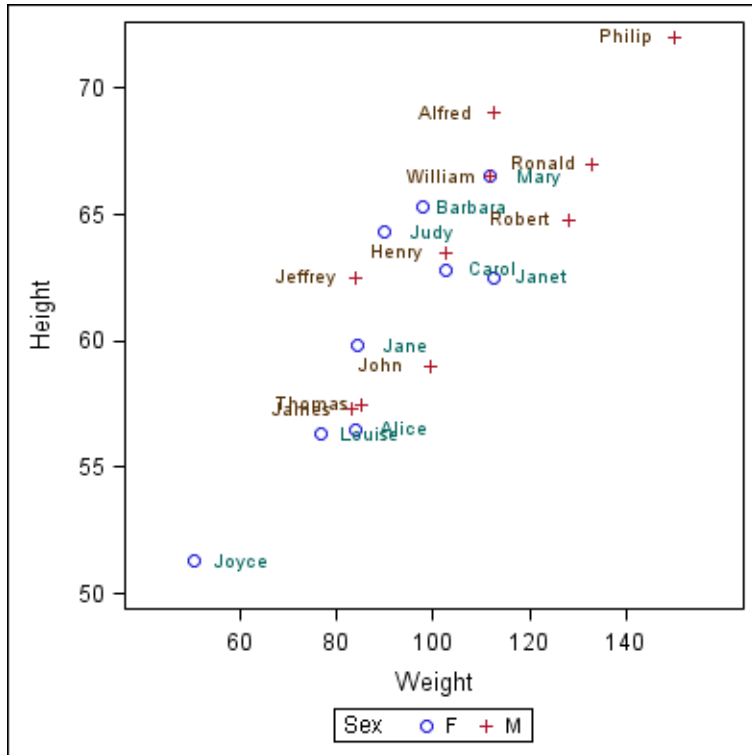
ODS Graphics from SAS 9.2 (Easy)

Point labels are plotted using a 2nd SCATTER statement.

5

```
PROC SGPLOT DATA = class_point_ods;
    SCATTER Y = height X = weight / GROUP = sex;
    SCATTER Y = height X = weight2 / GROUP = sex MARKERCHAR = name;
RUN;
```

BAR LABELS

The sample data for SAS/GRAPH (**class_bar**) and ODS Graphics (**class_bar_ods**) is generated using
the following code:

```
PROC SUMMARY DATA = sashelp.class NWAY;
    CLASS age sex;
    VAR height;
    OUTPUT OUT = class_bar N = count;
RUN;

DATA class_bar_ods;
    SET class_bar;
    BY age sex;
    LENGTH ccount $1;
    RETAIN total_count .;
    ccount = STRIP(PUT(count, 1.));
    IF FIRST.age THEN DO;
        total_count = 0;
        count_ods = count - 0.5;
    END;
    ELSE DO;
        count_ods = count;
    END;
```
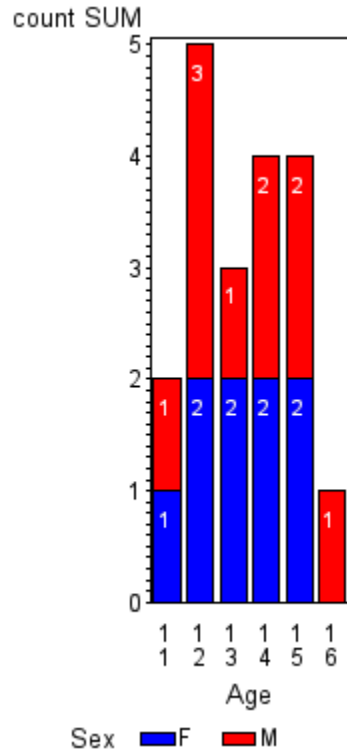
6

```
        total_count + count;
        total_count_ods = total_count - 0.5;
        IF sex = 'F' THEN total_count_odsf = total_count_ods;
        ELSE total_count_odsm = total_count_ods;
    RUN;
```

SAS/GRAPH from SAS 9.2

Bar Label plots can be created using PROC GCHART and Annotate.


```
DATA class_bar_anno;
    SET class_bar_ods;
    BY age sex;
    LENGTH function $8
           color $20
           position when xsys ysys hsys $1
           x y size 8
           ;
    xsys = '2';
    ysys = '2';
    hsys = 'D';
    when = 'A';
    size = '8';
    color = 'WHITE';
    function = 'LABEL';
    x = age;
    y = total_count;
    text = ccount;
    position = '8';
    OUTPUT;
RUN;
```

```
PROC GCHART DATA = class_bar ANNO = class_bar_anno;
    PATTERN1 VALUE = SOLID COLOR = BLUE;
    PATTERN2 VALUE = SOLID COLOR = RED;
    VBAR age / SUBGROUP = sex TYPE = SUM SUMVAR = count DISCRETE;
RUN;
```

ODS Graphics from SAS 9.3 (Easy)

The following example uses the VLINE statement to plot a series of point labels, without the points, onto the vertical bar chart. The unfortunate side-effect of this technique is that there is no absolute guarantee where the label will be displayed.
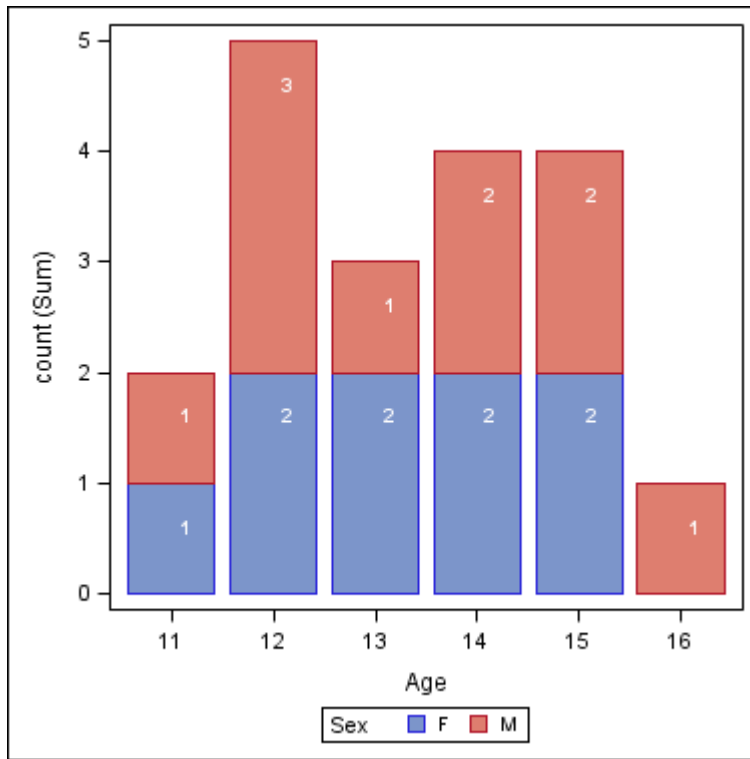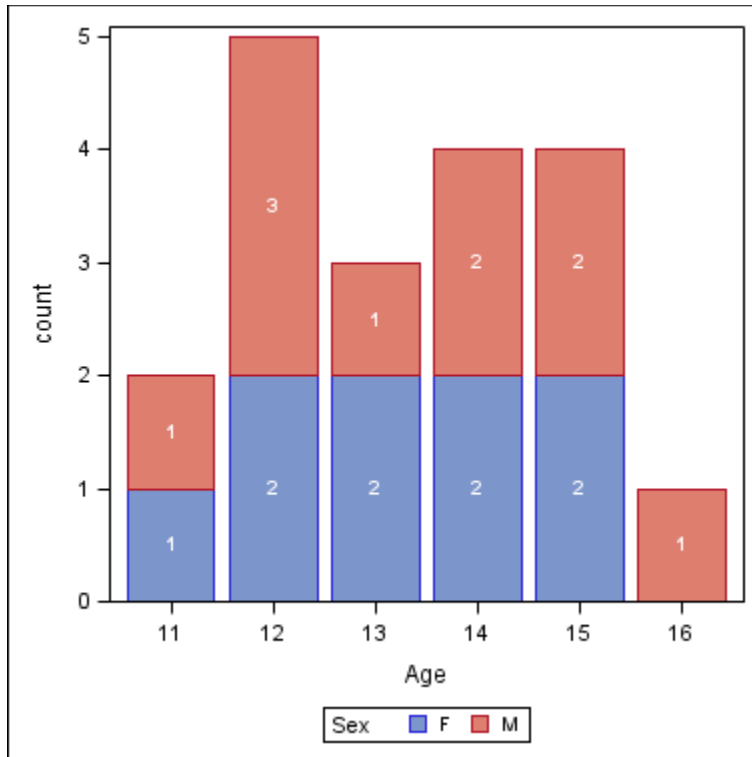


```
PROC SGPLOT DATA = class_bar_ods;
    VBAR age / GROUP = sex GROUPORDER = ASCENDING RESPONSE = count
            STAT = SUM;
    VLINE age / GROUP = sex GROUPORDER = ASCENDING
            RESPONSE = total_count_ods STAT = SUM
            DATALABEL = ccount DATALABELPOS = DATA
            DATALABELATTRS = (COLOR = WHITE WEIGHT = BOLD)
            LINEATTRS = (THICKNESS = 0);
RUN;
```

ODS Graphics from SAS 9.4 (Easy)

In SAS 9.4 the SEGLABEL option for VBAR is introduced, which allows labels to be placed in the center of any bar segment.

```
PROC SGPLOT DATA = class_bar_ods;
    VBARPARM CATEGORY = age RESPONSE = count / GROUP = sex
                    GROUPORDER = ASCENDING GROUPDISPLAY = STACK
                    DATALABEL = count DATALABELPOS = DATA SEGLABEL
                    SEGLABELATTRS = (COLOR = WHITE WEIGHT = BOLD);
RUN;
```

INFORMATION BOXES

The sample data for SAS/GRAPH and ODS Graphics (**class_info** and **class_info_range**) is generated using the following code:
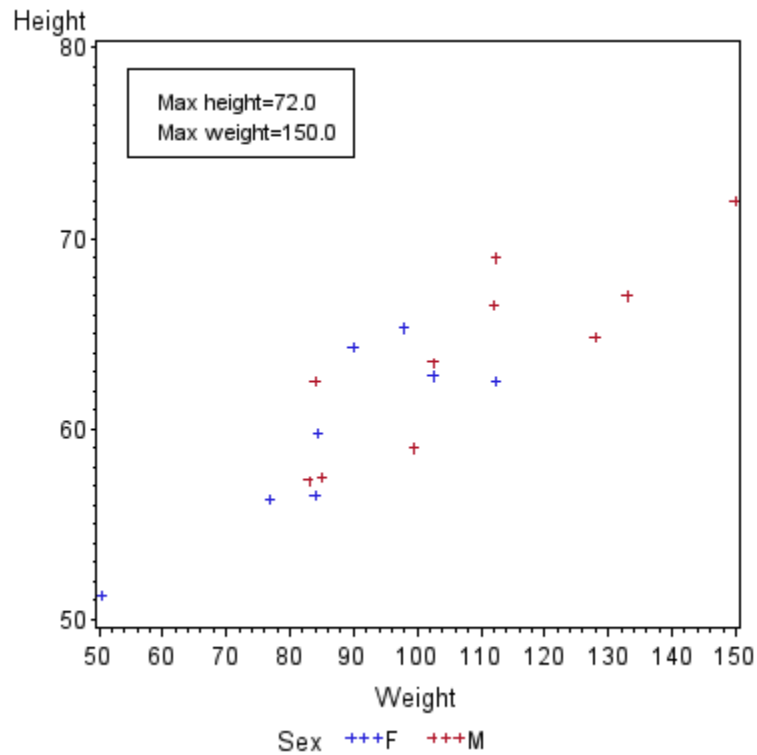
```
PROC SORT DATA = sashelp.class OUT = class_info;
    BY height weight;
RUN;

PROC SUMMARY DATA = class_info NWAY;
    VAR height weight;
    OUTPUT OUT = class_info_range MIN = min_height min_weight
                                  MAX = max_height max_weight;
RUN;
```

SAS/GRAPH from SAS 9.2

Information Box plots can be created using PROC GPLOT and Annotate.
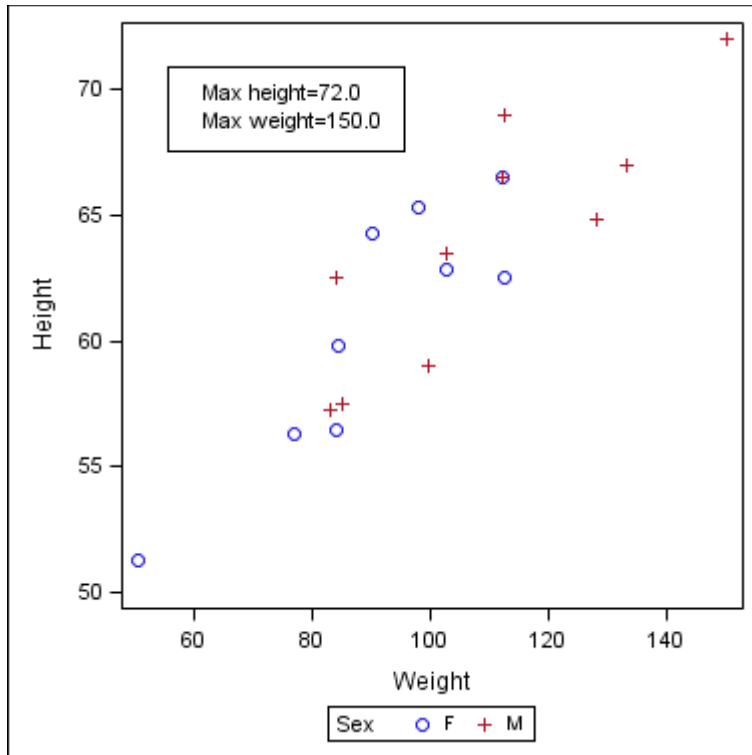
```
DATA class_info_anno (DROP = min_: max_: _:);
    SET class_info_range;
    %dclanno;
    LENGTH text $50;
    xsys = '1';
    ysys = '1';
    hsys = '3';
    when = 'A';
    %RECT(5, 95, 40, 80, BLACK, 1, 1);
    %LABEL(10, 90, "Max height=" || STRIP(PUT(max_height, 8.1)),
            BLACK, 0, 0, 3, Arial, 6);
    %LABEL(10, 85, "Max weight=" || STRIP(PUT(max_weight, 8.1)),
            BLACK, 0, 0, 3, Arial, 6);
RUN;
```

ODS Graphics from SAS 9.3 (Difficult)

This code uses the Annotate facility introduced to ODS Graphics in SAS 9.3.

```
DATA class_info_sganno (DROP = min_: max_: _:);
    SET class_info_range;
    LENGTH label $50;
    drawspace = 'DATAPERCENT';
    width = 40;
    anchor = 'TOPLEFT';
    function = 'RECTANGLE';
    height = 15;
    x1 = 5;
    y1 = 95;
    linecolor = 'BLACK';
    linethickness = 1;
    OUTPUT;
    anchor = 'LEFT';
    textsize = 8;
    height = .;
    function = 'TEXT';
    label = "Max height=" || STRIP(PUT(max_height, 8.1));
    x1 = 10;
    y1 = 90;
    OUTPUT;
    function = 'TEXT';
    label = "Max weight=" || STRIP(PUT(max_weight, 8.1));
    x1 = 10;
    y1 = 85;
    OUTPUT;
RUN;
```
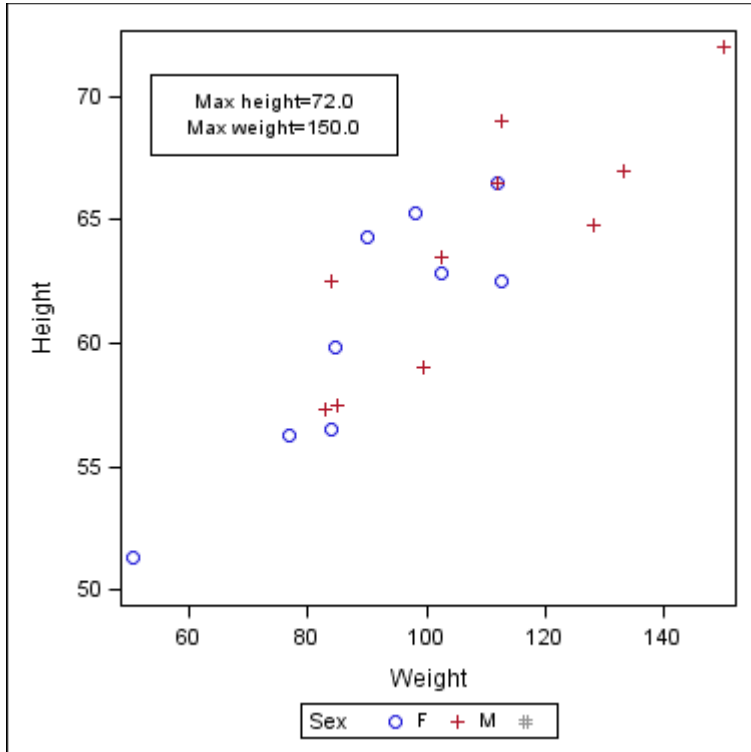
```
PROC SGPLOT DATA = class_info SGANNO = class_info_sganno;
    SCATTER X = weight Y = height / GROUP = sex;
RUN;
```

ODS Graphics from SAS 9.2 (Difficult)

If you prefer to use the layering techniques of ODS Graphics, then the same annotation can be achieved by plotting the information box and the text inside as separate overlaid plots.


```
DATA class_info_box (DROP = min_: max_: _:);
    SET class_info_range;
    LENGTH text $50;
    xbox = 5;
    ybox = 95;
    OUTPUT;
    xbox = 45;
    ybox = 95;
    OUTPUT;
    xbox = 45;
    ybox = 80;
    OUTPUT;
    xbox = 5;
    ybox = 80;
    OUTPUT;
    xbox = 5;
    ybox = 95;
    OUTPUT;
```

```
        xbox = .;
        ybox = .;
        xtext = 25;
        ytext = 90;
        text = "Max height=" || STRIP(PUT(max_height, 8.1));
        OUTPUT;
        xtext = 25;
        ytext = 85;
        text = "Max weight=" || STRIP(PUT(max_weight, 8.1));
        OUTPUT;
RUN;

DATA class_info_ods;
    SET class_info
        class_info_box
        ;
RUN;

PROC SGPLOT DATA = class_info_ods;
    SCATTER X = weight Y = height / GROUP = sex;
    SERIES X = xbox Y = ybox / LINEATTRS = (COLOR = BLACK) X2AXIS Y2AXIS;
    SCATTER X = xtext Y = ytext / MARKERCHAR = text
                                  MARKERCHARATTRS = (COLOR = BLACK)
                                  X2AXIS Y2AXIS;
    XAXIS OFFSETMIN = 0.02 OFFSETMAX = 0.02;
    X2AXIS OFFSETMIN = 0 OFFSETMAX = 0 MIN = 0 MAX = 100
           DISPLAY = (NOLABEL NOTICKS NOVALUES);
    Y2AXIS MIN = 0 MAX = 100 DISPLAY = (NOLABEL NOTICKS NOVALUES);
RUN;
```

CONCLUSIONS

Generating plots using ODS Graphics is based on the very simple application of graph layers, where individual graphs are drawn on top of each other to create the finished plot:

- Error Bars can be generated by plotting a SCATTER plot with YERRORUPPER= and YERRORLOWER= options on top of, or below, a SERIES plot from SAS 9.2.

- Point Labels can be generated by plotting the labels with a second SCATTER plot with MARKERCHAR options for the text, where the x-coordinates have been offset to improve readability from SAS 9.2.

- Bar Labels can be generated by plotting the labels with a VLINE plot on top of a VBAR chart in SAS 9.3, although the positioning of the labels can be offset slightly. In SAS 9.4 Bar Labels can be generated more precisely by using SEGLABEL options with a VBARPARM chart.

- Information Boxes can be generated with the SGANNO= option of PROC SGPLOT in SAS 9.3, which is the ODS Graphics equivalent of Annotate. However, by calculating the location of the box corners and the text, a similar Information Box can be drawn with more flexibility in SAS 9.2 using a simple input data set containing extra data coordinates, an extra SERIES statement for the box, and an extra SCATTER statement with MARKERCHAR options for the text.

REFERENCES

Holland, Philip R. June 2015. *SAS Programming and Data Visualization Techniques.* Apress.

CONTACT DETAILS

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

Author Name	Philip R Holland
Company	Holland Numerics Ltd
Address	94 Green Drift
City / Postcode	Royston, Hertfordshire, SG8 5BT, United Kingdom
Work Phone:	+44 7714 279085
Fax:	+44 1763 242486
Email:	phil@hollandnumerics.com
Web:	www.hollandnumerics.com/SASPAPER.HTM

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.