



Quanticate

CONFIDENCE COMES FROM QUALITY

Effective SAS greplay'ing and how to avoid stretching

By David Mottershead
Senior Programmer, Quanticate

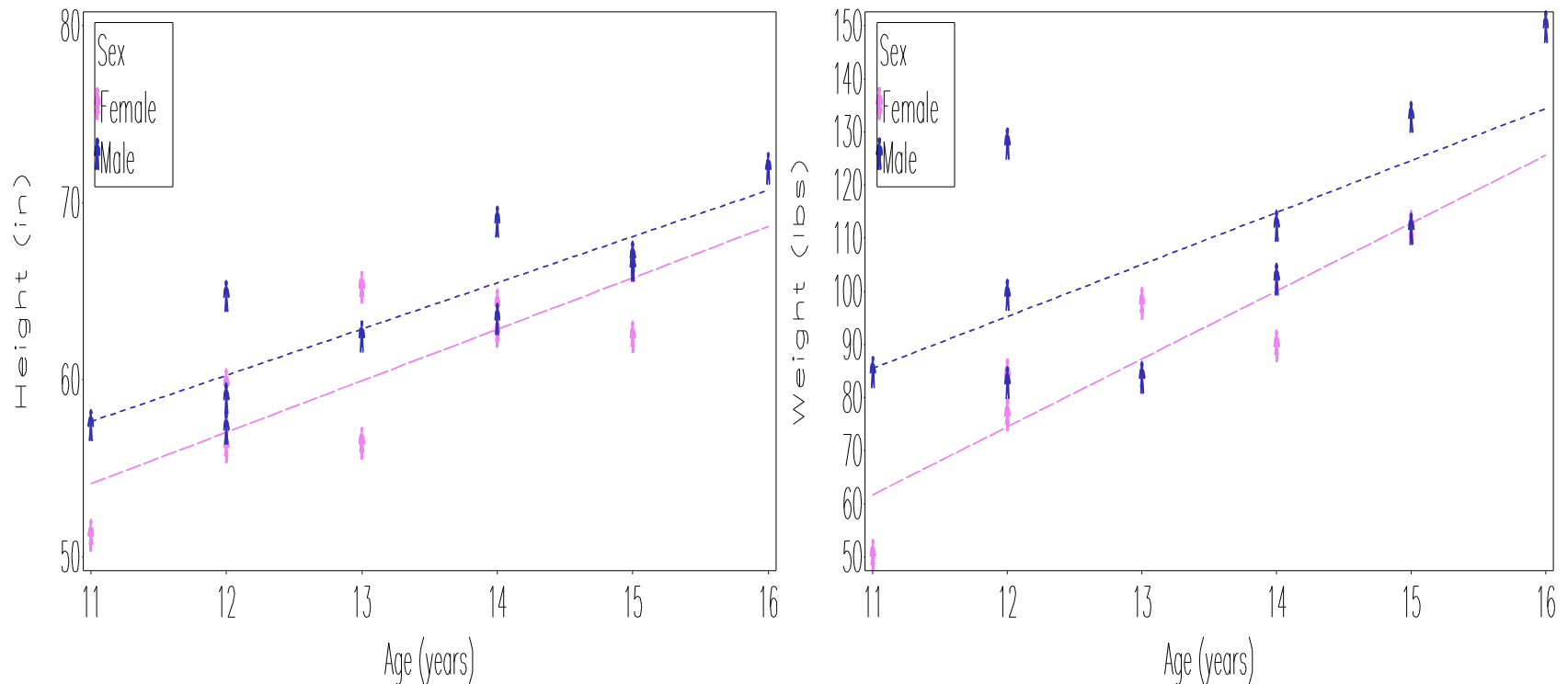
Introduction

- Review goptions
- Look at how text size is affected when goptions are changed
- Creating figures with proc gplot
- Creating titles and footnotes using proc gslide
- Creating a template with proc greplay
- Creating a figure with proc greplay

Introduction greplay stretching – the problem!



PhUSE 2011
Figure 1
Plot of height against age and weight against age

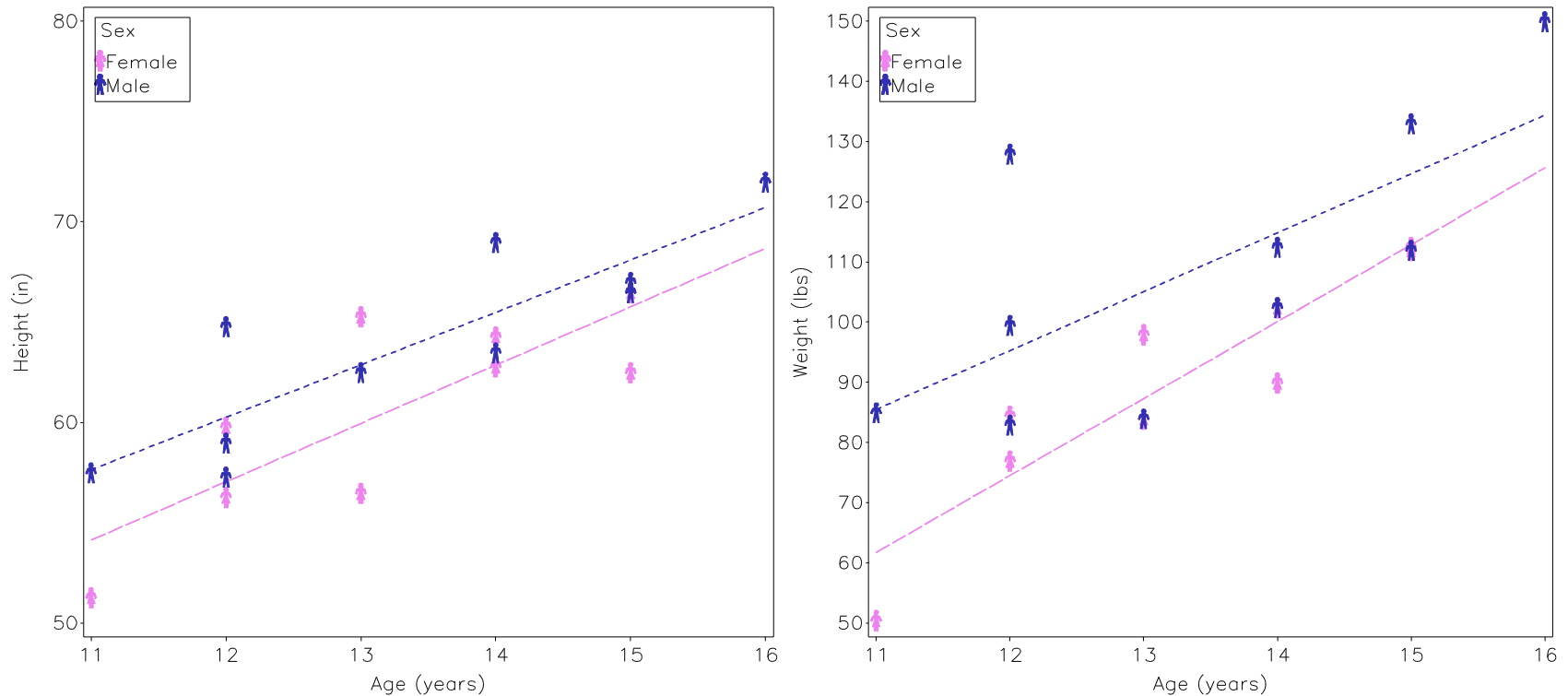


Author: David Mottershead

Introduction greplay stretching – what we want

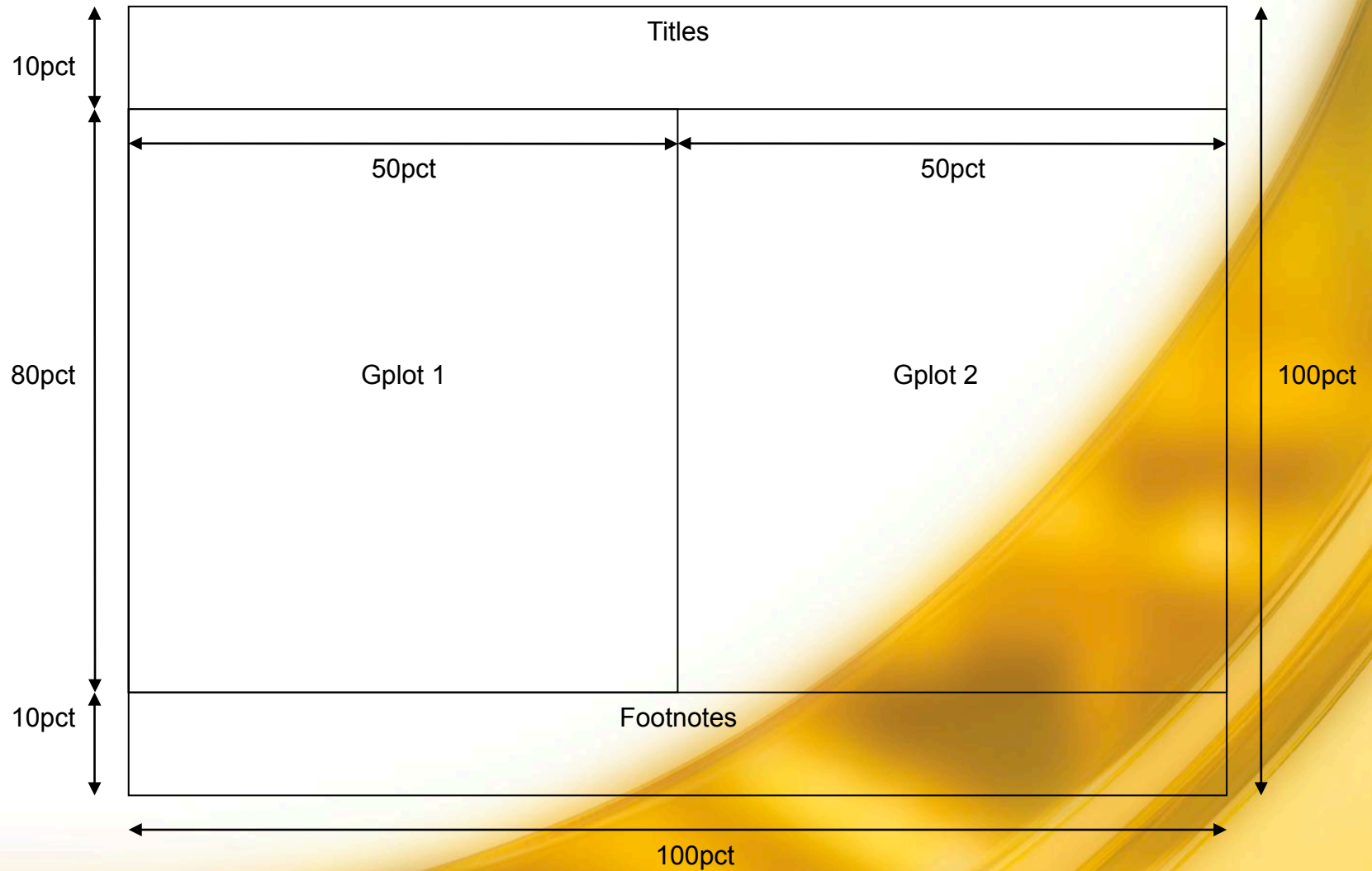


PhUSE 2011
Figure 1
Plot of height against age and weight against age



Author: David Mottershead

Diagram of figure



SAS goptions

- Goptions are used to control how graphical output is displayed.
- There are some key goptions which are used to control the size of figures:
 - Xmax: specifies the width of the figure
 - Ymax: specifies the height of the figure
 - Xpixels: specifies the number of pixels for the width of the figure
 - Ypixels: specifies the number of pixels for the height of the figure

SAS goptions (cont.)

- The required width of the combined figure is 24.5cm and the required height is 14.5cm.
- So the width and height of the gplot figures are calculated as follows:
 - The width of gplot figures = $0.5 * 24.5\text{cm} = 12.25\text{cm}$.
 - The height of gplot figures = $0.8 * 14.5\text{cm} = 11.6\text{cm}$.
- The goptions to define the dimensions of the combined figure and the gplot figures are shown below:

```
** Combined figure **;  
goptions xmax      = 24.5cm  
          xpixels   = 2450  
          ymax      = 14.5cm  
          ypixels   = 1450;
```

```
** Individual gplots **;  
goptions xmax      = 12.25cm  
          xpixels   = 1225  
          ymax      = 11.6cm  
          ypixels   = 1160;
```

SAS goptions (cont.)

- There are other important goptions which are required throughout the whole of the session:

```
goptions device = emf  
          htext  = 0.35cm  
          ftext  = simplex;
```

- Device - specifies the graphics device.
- Htext - specifies the text height.
- Ftext - specifies the font.

How is text size affected when the ymax goption is changed?



- The size of SAS text is controlled by specifying its height in units.
- The default SAS unit for specifying text height is cells.
- An important property of the cell unit is that heights are in relation to the height of the figure (ymax). In our example, if a text size of 1 cell is used to create both the gslide and gplot figures, the size of the text in the gplot figures will be 80 percent of the size of the text in the gslide.
- The easiest way to avoid this is to use an absolute unit such as cm or in.

How is text size affected when the ymax goption is changed? (cont.)

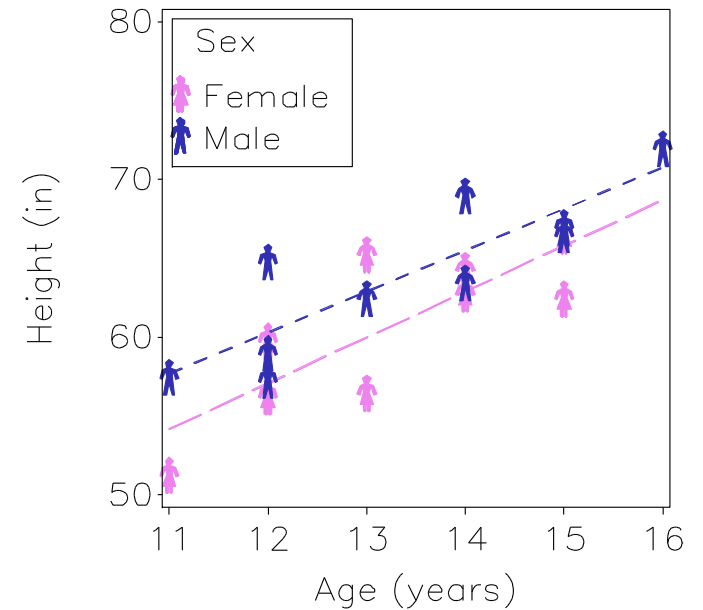


- Alternatively you can calculate the required text size in cells for the gplot figures so the text size will be 1 cell in the combined figure. Since the height of the gplot figures are 80 percent of the gslide,
$$1 \div 0.8 = 1.25.$$

Creating figures with proc gplot



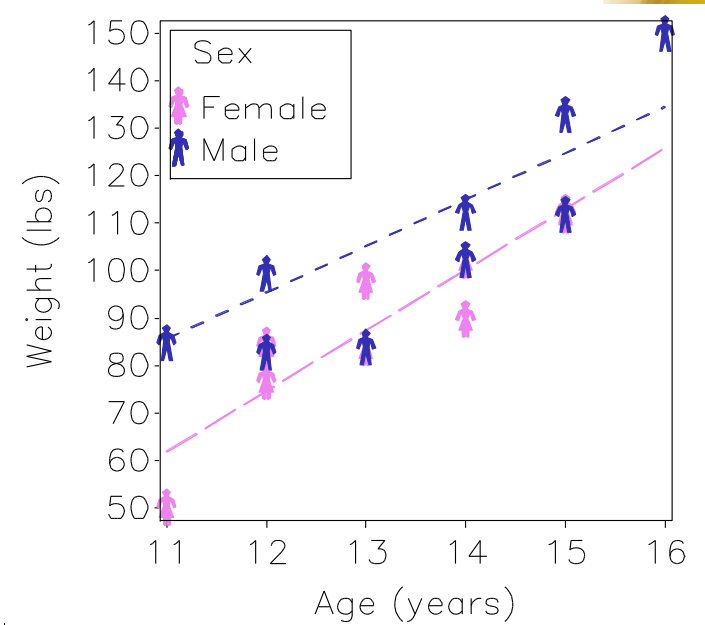
- The SAS/GRAPH procedure `proc gplot` has been used to create the figures. Some additional options have been used to enhance the look of the gplots:
 - Minor tick marks removed.
 - Symbol statements are used to define symbols, symbol colours and specify interpolation lines.
 - Axis statements are used to rotate the orientation of the vertical axis label so it is vertical, specify offsets for the axis so that symbols fit within the graph area and to reduce the height of the axis slightly so the full figure will display correctly using the `length` option.



Creating figures with proc gplot (cont.)



- A legend statement is used to place the legend within the graph area, which creates more space for the plot. The shape of the legend has also been manipulated so only one symbol is displayed next to the legend label rather than three.
- It is worth noting that SAS has a problem or **correctly** calculating the required length of the vertical axis for the graph to fit within the defined area. When this occurs the following warning is printed to the log:
 - WARNING: The left vertical axis labeled Height was too large as specified. Specify LENGTH=XX.X PERCENT.
- This can be corrected by specifying the length of the axis to be less than or equal to the length specified.



SAS catalogs and figure names



- By default
 - the figure will be stored in a catalogue called gseg in the work directory.
 - the name of the first figure created by the gplot procedure will be gplot. The next figure created will be gplot1, then gplot2 and so on.
- You can specify the name of the SAS catalogue to store the figure in using the gout= option in the gplot statement.
- You can change the name of the figure using the name= option in the plot statement to a name of your choice up to 8 characters.

SAS catalogs and figure names



- If a figure with the same name already exists within the catalogue it will not be overwritten but incremented with a 1 at the end, then a two at the end and so on.
- When developing code in SAS interactively it is worth being aware of this when changing gplot code and re-running a greplay procedure again because if the old figures are not deleted from the catalogue the new versions of the gplot figures will not be used in the greplay.
- The contents of the catalogue can be deleted using the following code.

```
proc datasets lib=work nolist memtype=cat;  
  delete <catalogue-name(s)>;  
quit;
```

Creating titles and footnotes using proc gslide



- Titles and footnotes should be presented consistently with the titles and footnotes in accompanying listings and tables. They also need to span across the whole of the graph area.
- A simple way to do this is using the SAS/GRAPH procedure proc gslide.

```
proc gslide;  
  title1 j=1 "PhUSE 2011";  
  title2 j=1 "Figure 1";  
  title3 j=1 "Plot of height against age and weight against age";  
  footnote1 j=1 "Author: David Mottershead";  
run;  
quit;
```

Using the greplay procedure to create a template



- The syntax to set up our template is as follows:

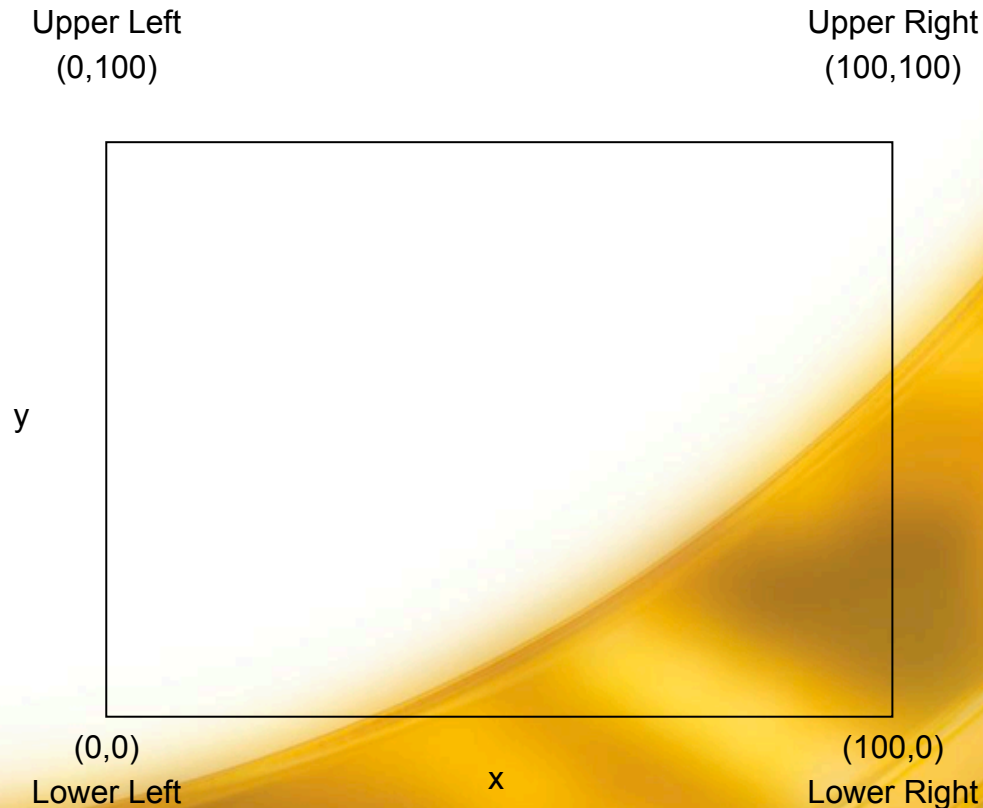
```
proc greplay tc=work.tempcat nofs;  
  tdef newtemp des='Two panel template'
```

- The tc= option defines the template catalogue where the template will be stored.
- The nofs option stops windows being used in interactive SAS.
- The tdef statement is used to define the new template which is called newtemp in our example.
- The description option associates a description with the template.

Using the greplay procedure to create a template (cont.)



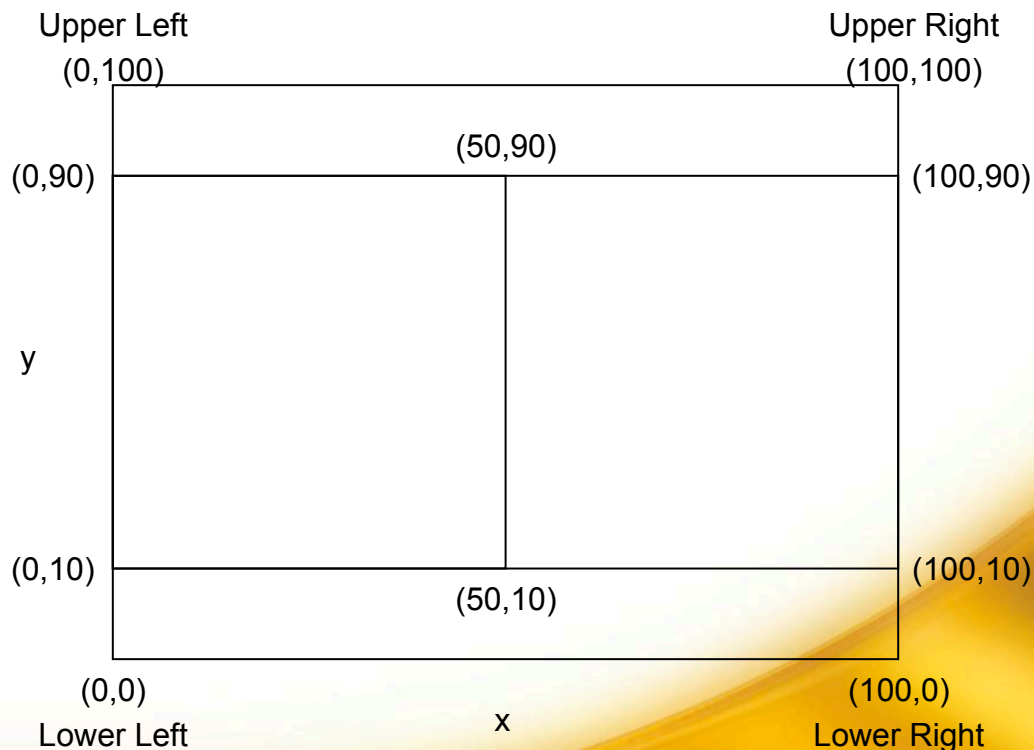
- The template is created by defining the coordinates of the 4 corners where each figure will be placed. The coordinates of the whole greplay area are shown in the below figure.



Using the greplay procedure to create a template (cont.)



- A visualisation and the code to create the template for the gslide (1) and gplot figures 1 and 2 (2 and 3 respectively) are shown below:



```
1/llx=0    llx=0  
   lrx=100  lry=0  
   ulx=0    uly=100  
   urx=100  ury=100
```

```
2/llx=0    llx=10  
   lrx=50   lry=10  
   ulx=0    uly=90  
   urx=50   ury=90
```

```
3/llx=50    llx=10  
   lrx=100  lry=10  
   ulx=50   uly=90  
   urx=100  ury=90
```

Using the greplay procedure to create the final figure



- We need to set up the filename and goptions to save our figure to an external location. Example code is shown below.

```
filename gsasfile "C:\temp\Combined Figure.emf";  
  
goptions gsfname = gsasfile  
        gsfmode = replace;
```
- The filename statement specifies where the figure will be stored.
- The goption gsfname associates the filename with the figure.
- The option gsfmode = replace is used so each time a figure is created it will be replace by the new one.

Using the greplay procedure to create the final figure (cont.)



- We use proc greplay again and specify the same template catalogue and nofs options.
- The template statement is used to specify the name of the template we want to use.
- The igout statement specifies the catalogue where we want the created figure to be stored.
- The treplay statement is used to specify figures from the catalogue which we want to place into the different template positions.

```
proc greplay tc=work.tempcat nofs;  
  template newtemp;  
  igout work.tempcat;  
  treplay 1:GSLIDE  
          2:FIGURE1  
          3:FIGURE2;  
run; quit;
```

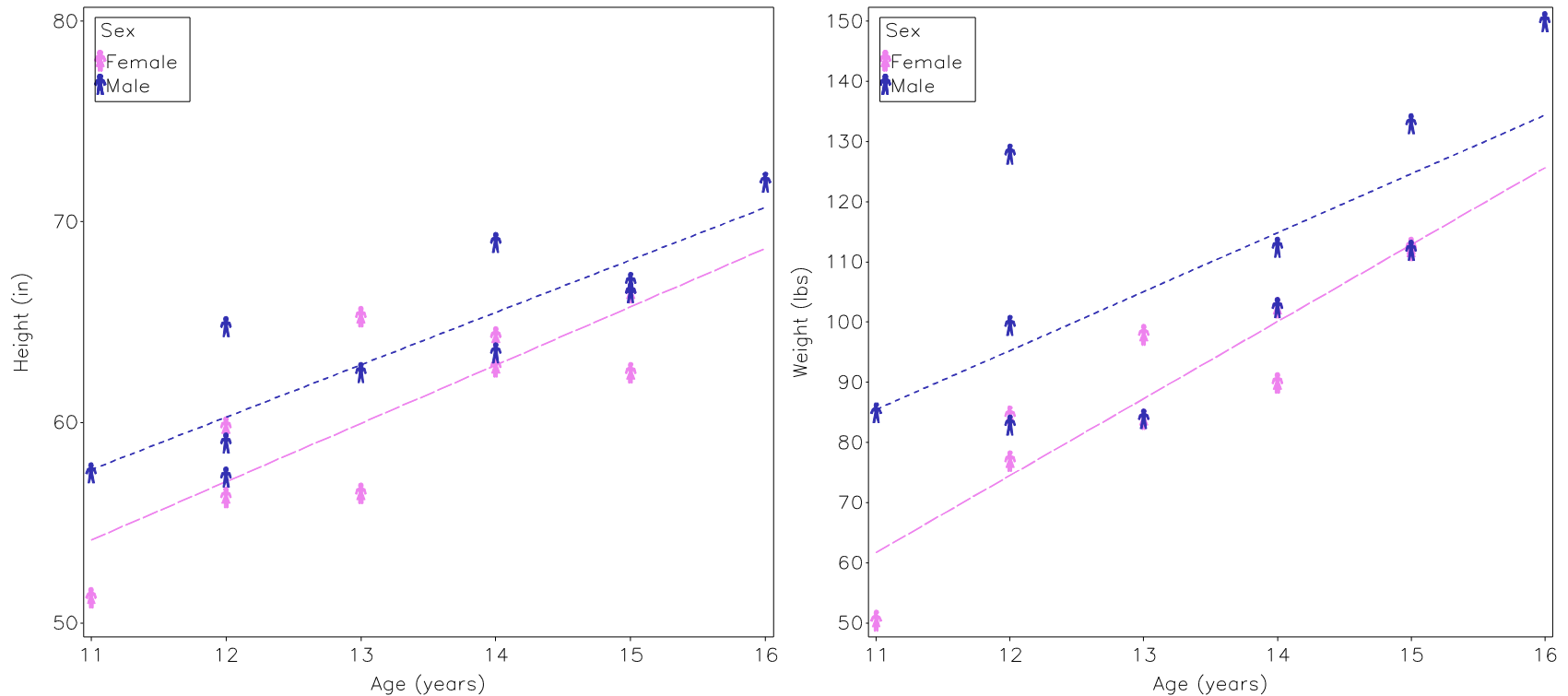
Using the greplay procedure to create the final figure (cont.)



PhUSE 2011

Figure 1

Plot of height against age and weight against age



Author: David Mottershead

Summary



- Review goptions
- Look at how text size is affected when goptions are changed
- Creating figures with proc gplot
- Creating titles and footnotes using proc gslide
- Creating a template with proc greplay
- Creating a figure with proc greplay

“...creating one page of output with multiple graphs ... can be very tedious.” “...graphs are scaled down to fit in the cookie cutter sections of the template. The scaling of the graphs can result in illegible fonts and problems with the lengths of the axis.” – Dirk Spruck.

End. Questions?

