

DV-01:

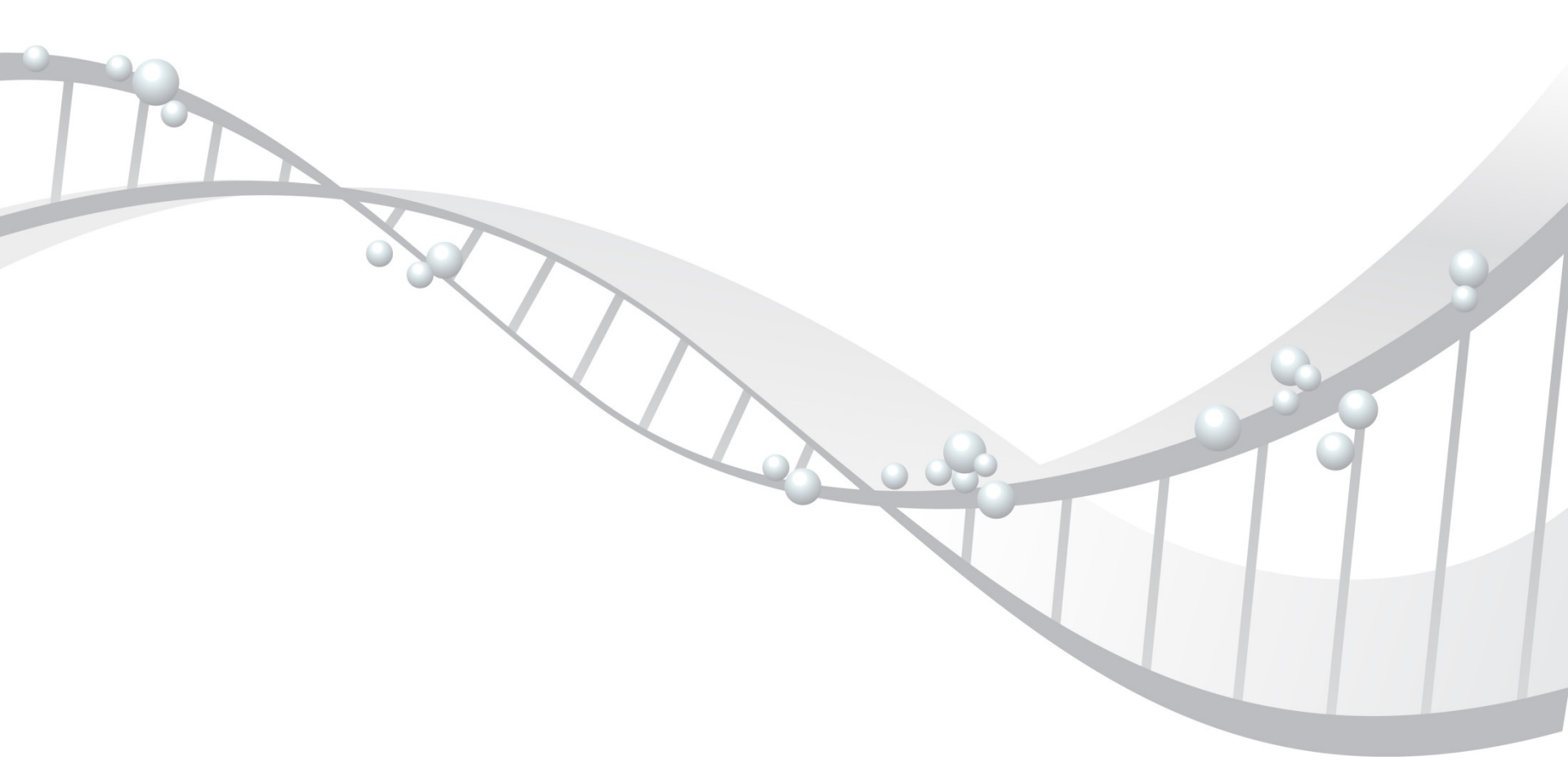
Turning Fantasy into Reality - Creating Custom Data Visualisations for Future Reporting

James Diserens
Veramed Limited
November 2018



Agenda

- Data Visualisations & Study Reporting.
- SAS Graphics.
 - SAS and R for graphs.
 - Graph Template Language (GTL).
- Embedded Line Plots.
- Lattice Scatter.
- Swimmer Plots.



DATA VISUALISATIONS & STUDY REPORTING

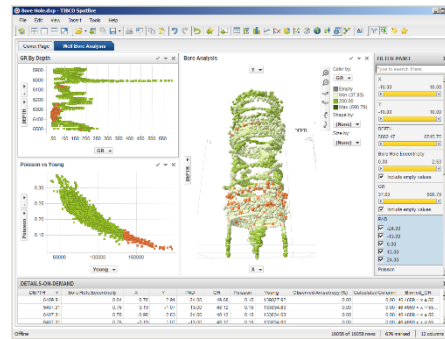
Data Visualisations

- Comprehend information quickly.
- Identify relationships and patterns.
- Communicate the story to others.



“I’ll give you a thousand words for it.”

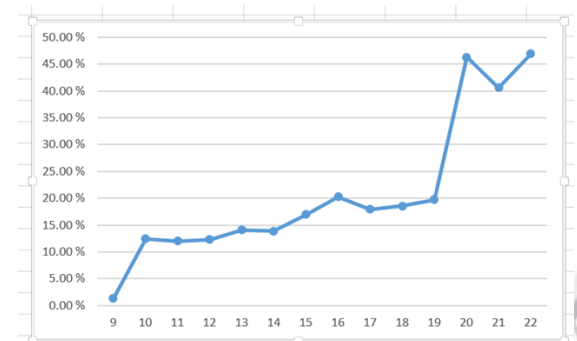
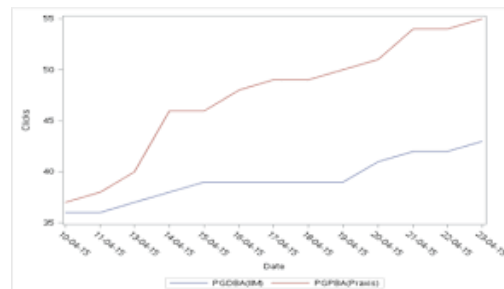
TIBCO® Spotfire®



Tools for Figures



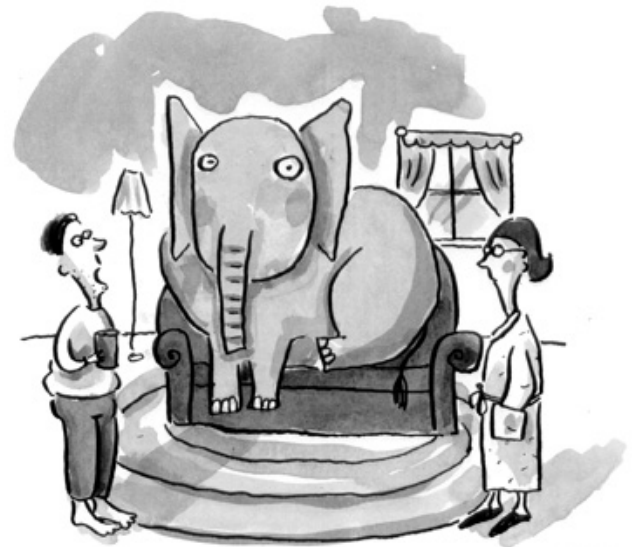
sas®

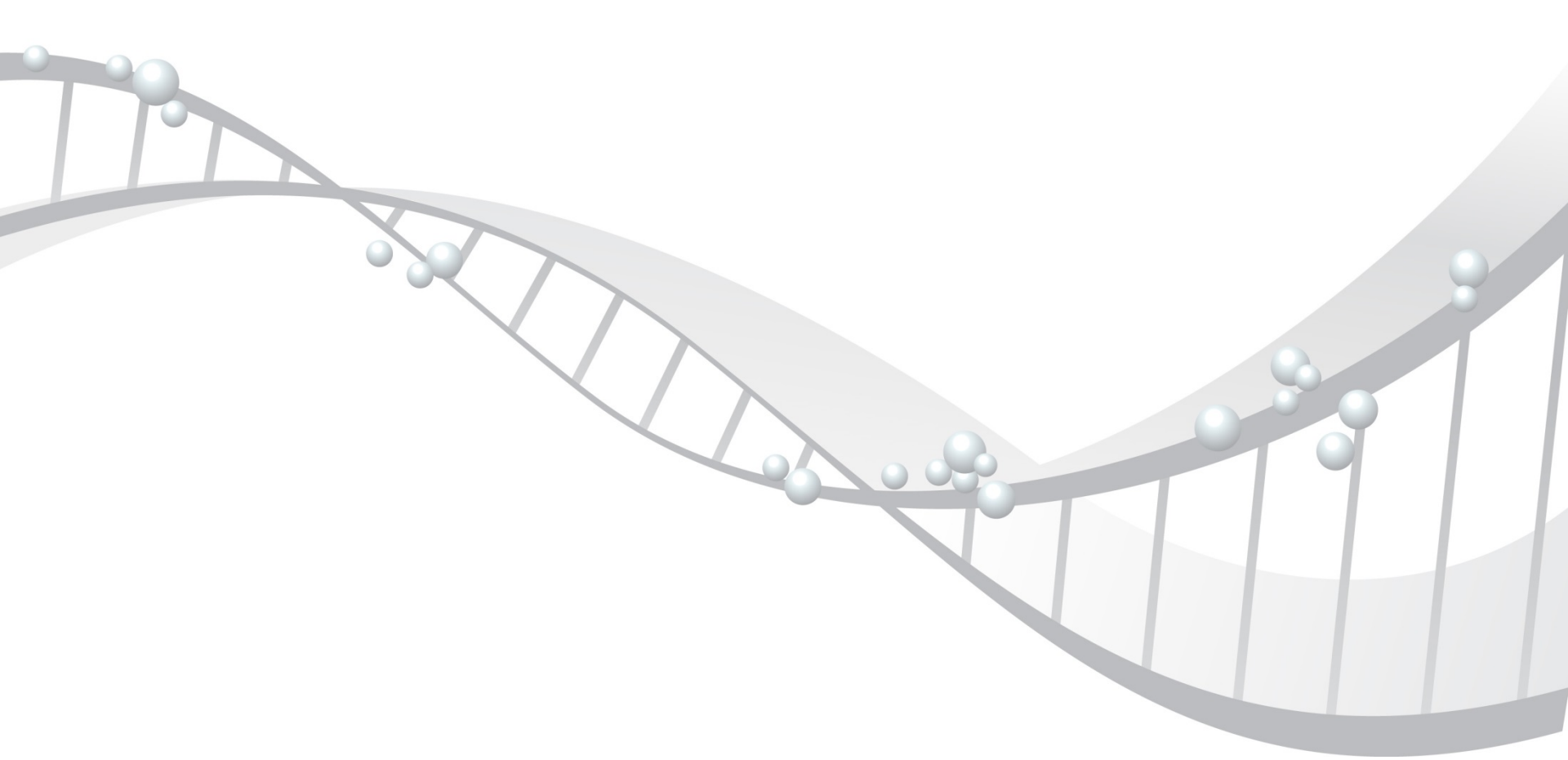


veramed

Study Reporting Figures

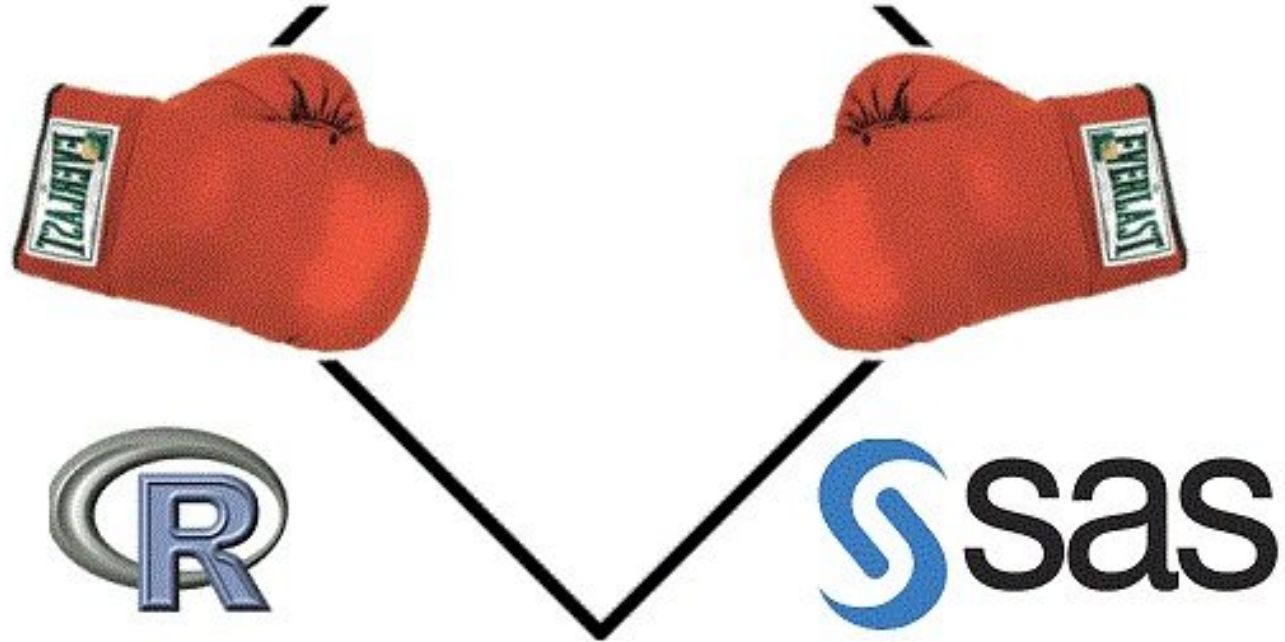
- Based on Analysis Plans & shells.
- Restrictive when compared to exploratory analysis.
 - Interactive software
 - Greater freedoms/flexibility
- Programming frameworks are based in SAS.



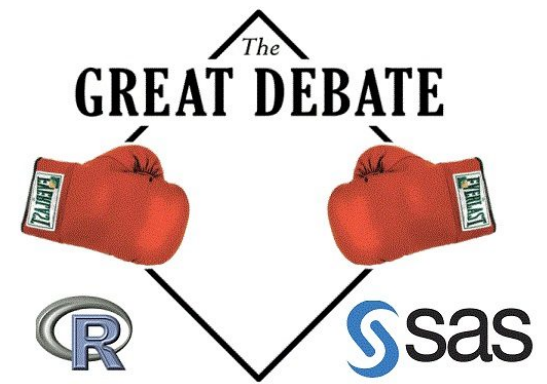


SAS GRAPHICS

The
GREAT DEBATE



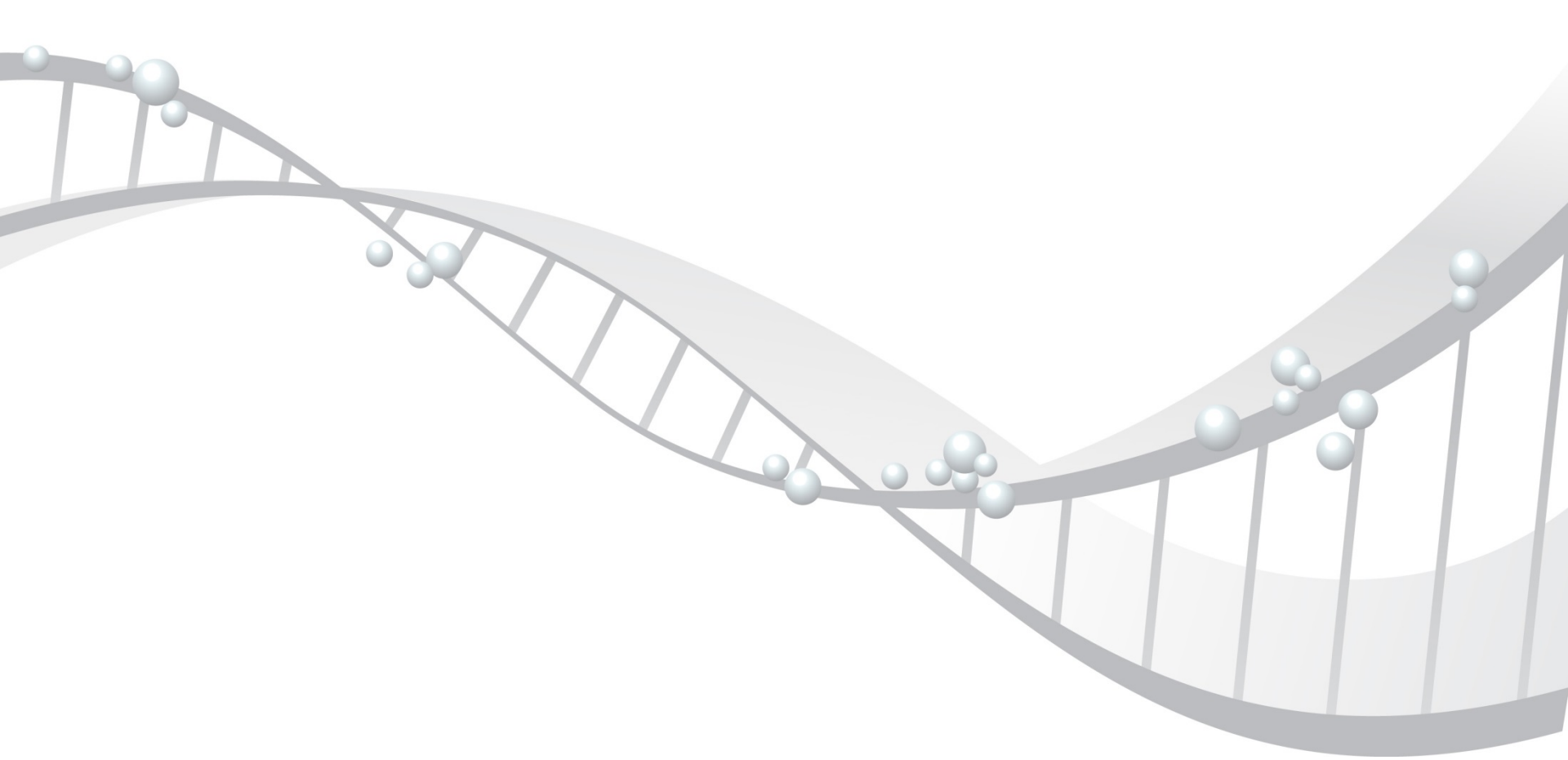
SAS v R



- “R is better than SAS for graphs”?
 - Does it affect the way we use SAS graphics?
 - SAS is functional but customisations are difficult.
 - SAS requires additional software.
- Cross-over of skills is important.

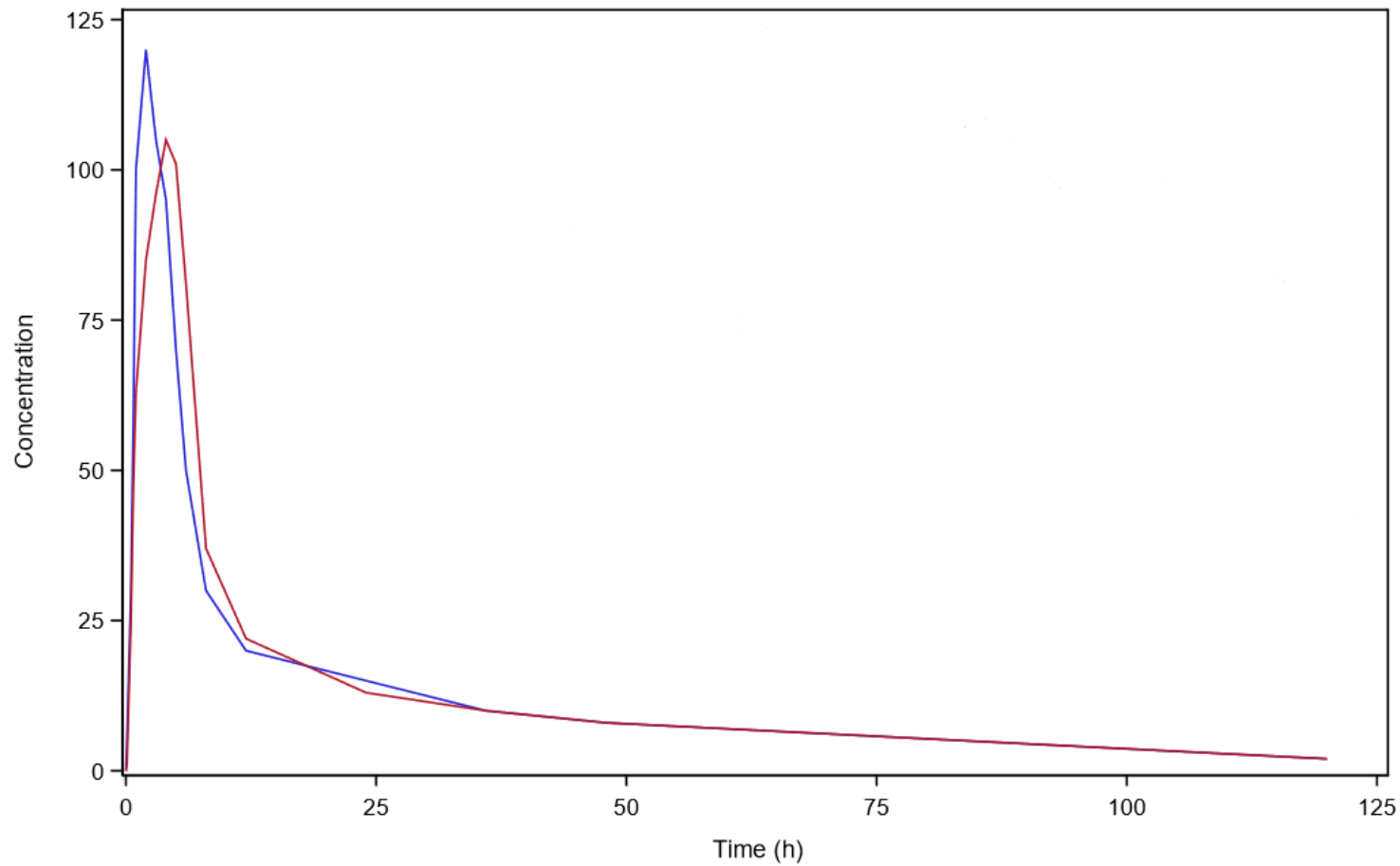
SAS Graphics

- Standard graphical procedures:
 - e.g. PROC SGPLOT & PROC SGPANEL.
 - 16 different types of plots possible.
 - Inflexible to customise.
- Graph Template Language (GTL):
 - PROC TEMPLATE.
 - Flexible but complicated.
 - Produces graphs in standard image file formats such as PNG and JPEG .



EMBEDDED LINE GRAPHS

Embedded Line graphs



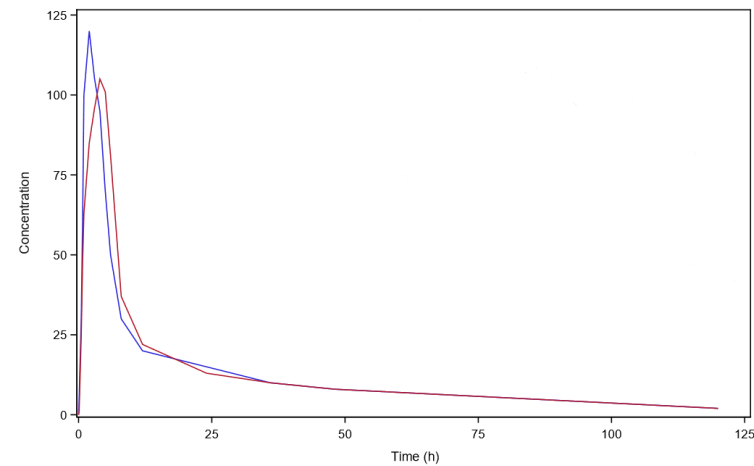
Embedded Line graphs

- Issues:

- Data condensed in small area of the figure.
- Difficult to distinguish potential differences.
- Large amounts of white space.

- Improvements:

- Focus on a smaller area.
- Maintains the overall picture.
- Embed a section of the graph in the white space.



Embedded Line graphs

```
ods graphics on / reset=index imagename="insert" width=20cm height=20cm imagefmt=png;  
ods listing image_dpi=300 gpath=fout;
```

```
|proc sgplot data=&dsin.(where=(&xvar. < &zxmax.)) noautolegend;  
  series x=&xvar. y=&yvar. / group=&group. lineattrs=(pattern=1);  
  xaxis label="&xlabel" labelattrs=(size=16) valueattrs=(size=16);  
  yaxis label="&ylabel" labelattrs=(size=14) valueattrs=(size=14);  
run;
```

```
ods graphics off; ods listing; filename fout clear; goptions reset=all;
```

```
** Proc template for the main plot area;  
|proc template;  
  define style MyDefault;  
    parent = Styles.Listing;  
    class GraphData1 /linestyle = 1 contrastcolor = GraphColors('gcdatal');  
    class GraphData2 / linestyle = 1 contrastcolor = GraphColors('gcdata2');  
  end;  
  define statgraph insert;  
  dynamic grpl;  
    begingraph;  
      layout overlay /xaxisopts = (label="&xlabel") yaxisopts = (label="&ylabel");  
      seriesplot x=&xvar. y=&yvar. /group=&group. name="main";  
      discretelegend "main" / title='Treatment' across=2 border=true;  
    endlayout;  
    drawimage "&lib\insert.png" / border=false layer=front anchor=topright x=99 y=99 height=60 width=60;  
  endgraph;  
end;  
run;
```

Embedded Line graphs

- Create the smaller plot using SGPLOT.
- Save as a PNG file.

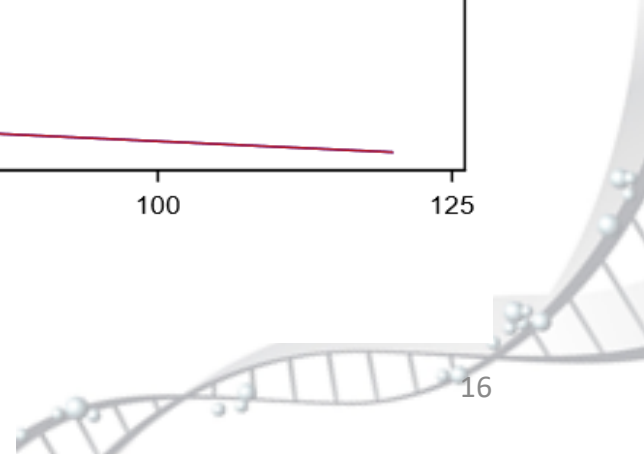
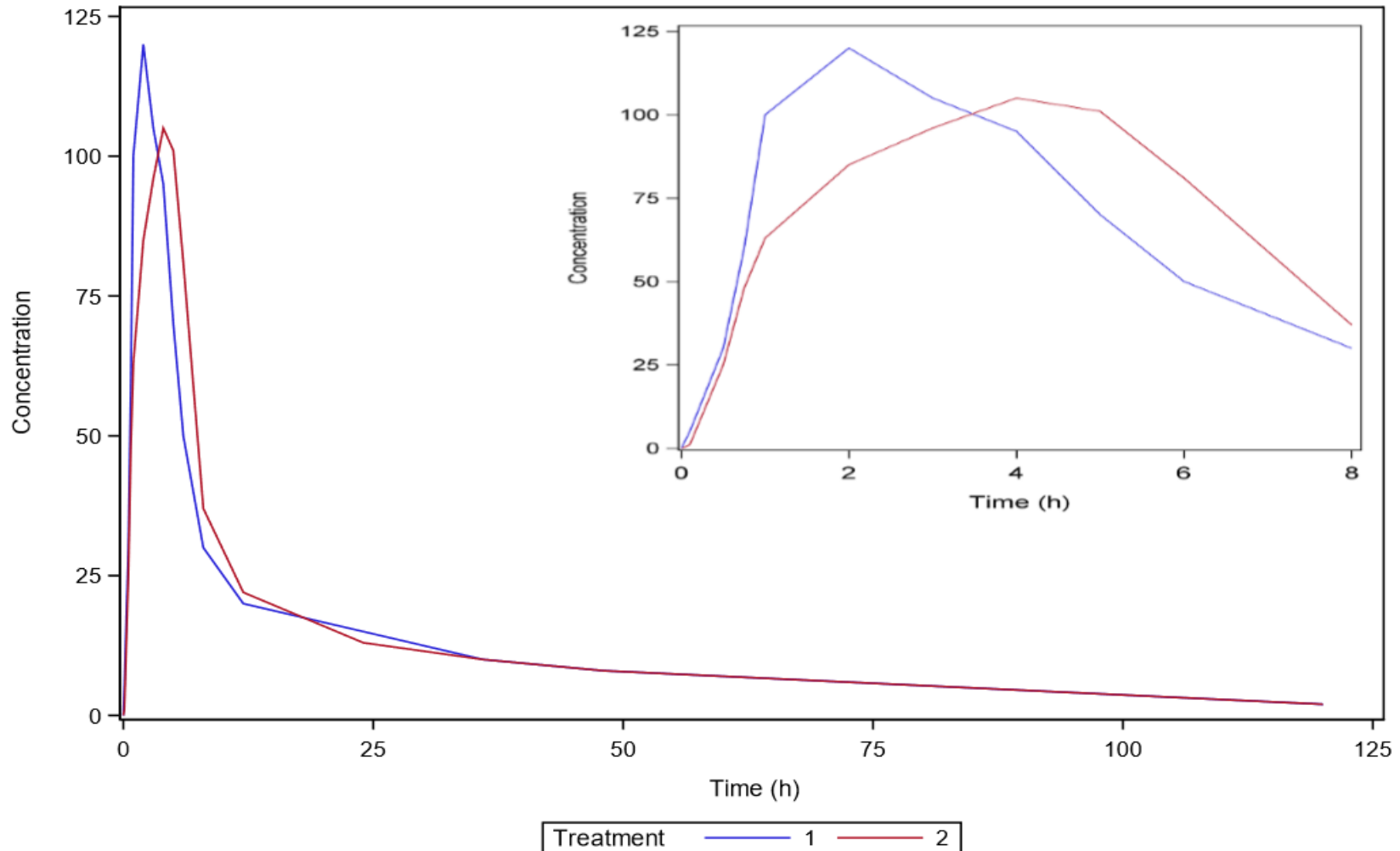
```
** Create PNG image for insert plot;  
ods graphics on / reset=index imagename="insert" width=20cm height=20cm imagefmt=png;  
ods listing image_dpi=300 gpath=fout;
```

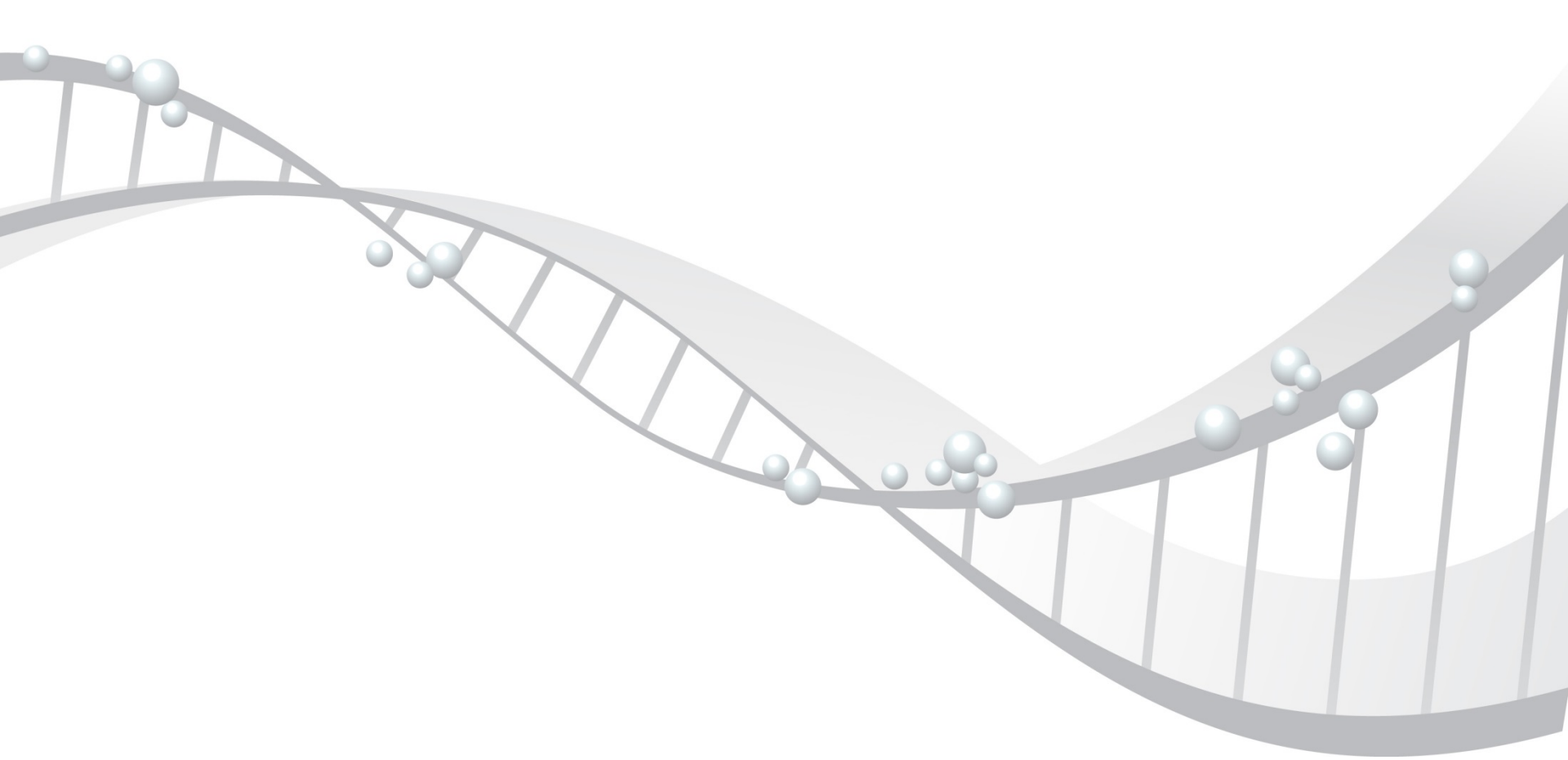
- Create the larger plot with PROC TEMPLATE.
- Redraw the first image.

```
/* Re-draw png image from series plot */  
drawimage "&lib\insert.png" / border=false layer=front anchor=topright  
x=99 y=99 height=60 width=60;
```

- Produce overall plot – PROC SGRENDER.

Embedded Line graphs

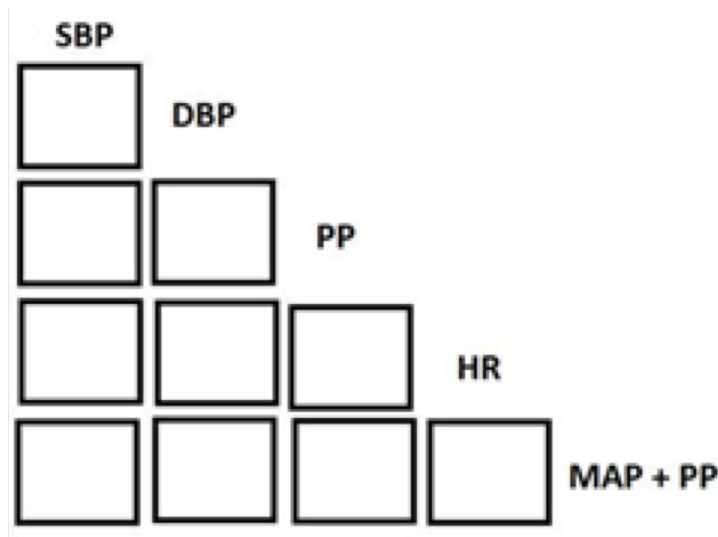




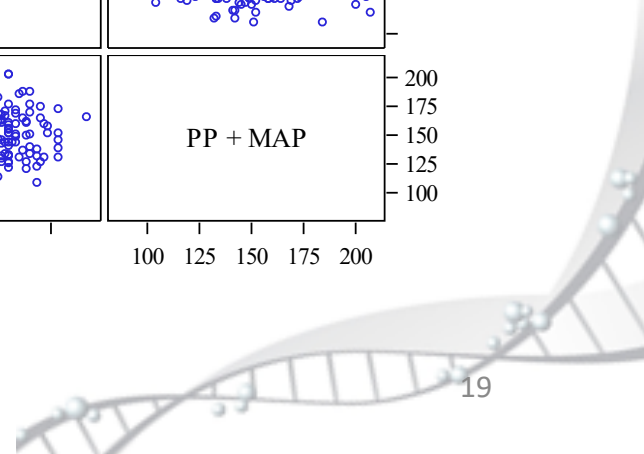
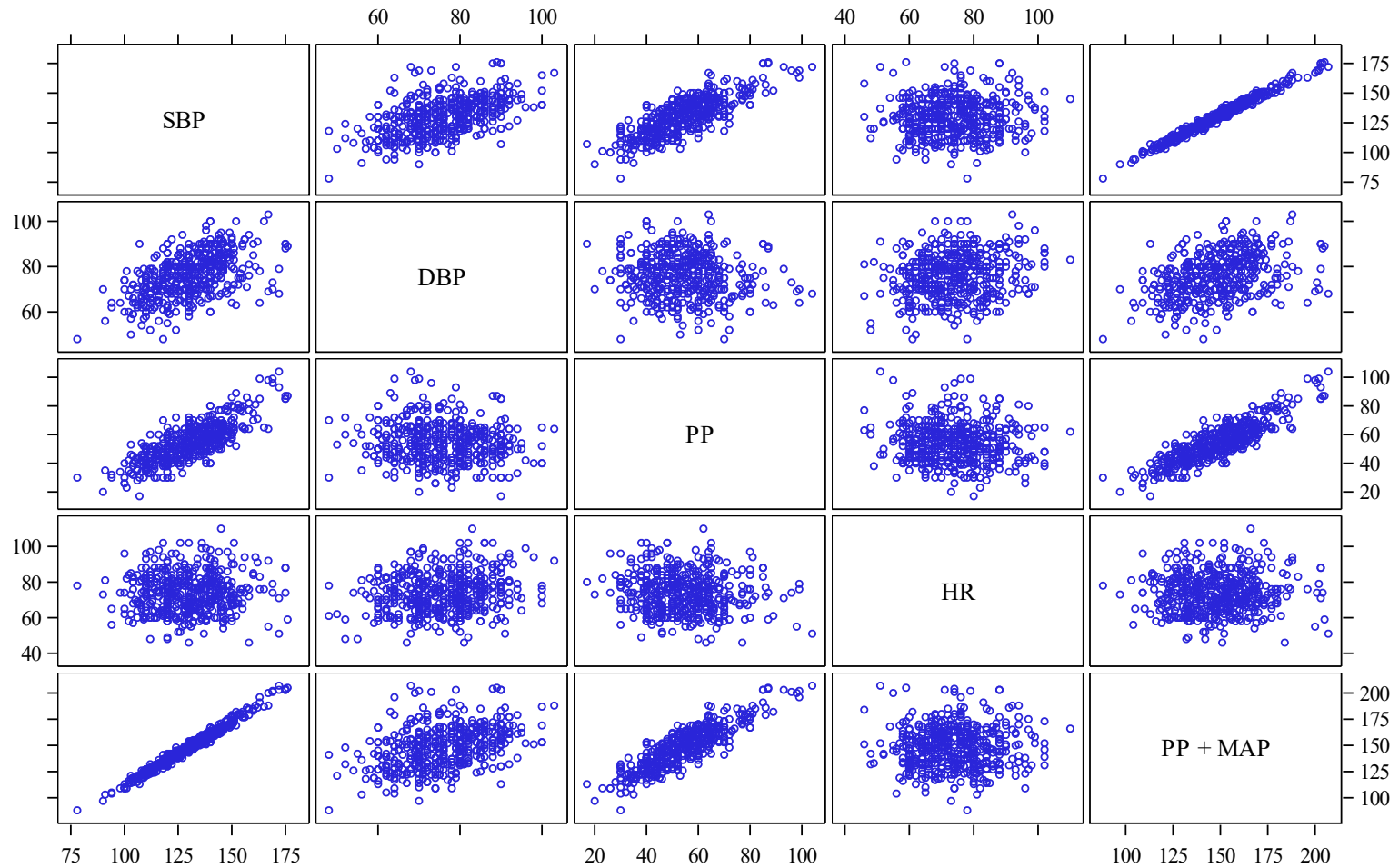
LATTICE PLOTS

LATTICE PLOTS

- Overview of multiple scatter plots.
- Compare relationship of multiple parameters.



SGSCATTER



SGSCATTER



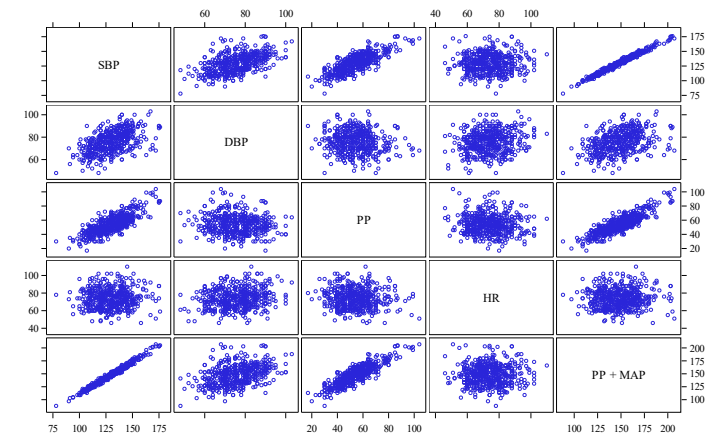
Issues:

- Repeat plots with flipped axis.
- Data condensed in small area of the figure.
- Increased noise and confusion.



Improvements:

- Each combination plotted once.
- Consistent axes and scale.
- Increase area of plot used.



LATTICE SCATTER

- PROC TEMPLATE.
- Create a blank n by n grid.

```
/* Create n * n dimension matrix */  
layout lattice / columns=&numvar. rows=&numvar. rowgutter=5 columngutter=5  
                rowdatarange=union columndatarange=union;
```

- Set up blank axis in each section.

```
/* Set up n rows for i variables */  
rowaxes;  
  %do i=1 %to &numvar.;  
    rowaxis / tickvalueattrs=(size=&fontsize.) labelattrs=(size=&fontsize.)  
             linearopts=(tickvaluesequence=(start=&minval. end=&maxval. increment=&valby.)  
                          tickvaluepriority=true);  
  %end;  
endrowaxes;  
  
/* Set up n columns for j variables */  
columnaxes;  
  %do j=1 %to &numvar.;  
    columnaxis / tickvalueattrs=(size=&fontsize.) labelattrs=(size=&fontsize.)  
               linearopts=(tickvaluesequence=(start=&minval. end=&maxval. increment=&valby.)  
                           tickvaluepriority=true);  
  %end;  
endcolumnaxes;
```

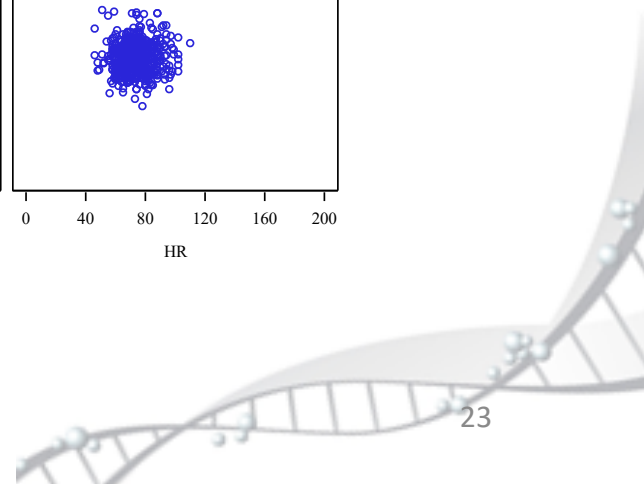
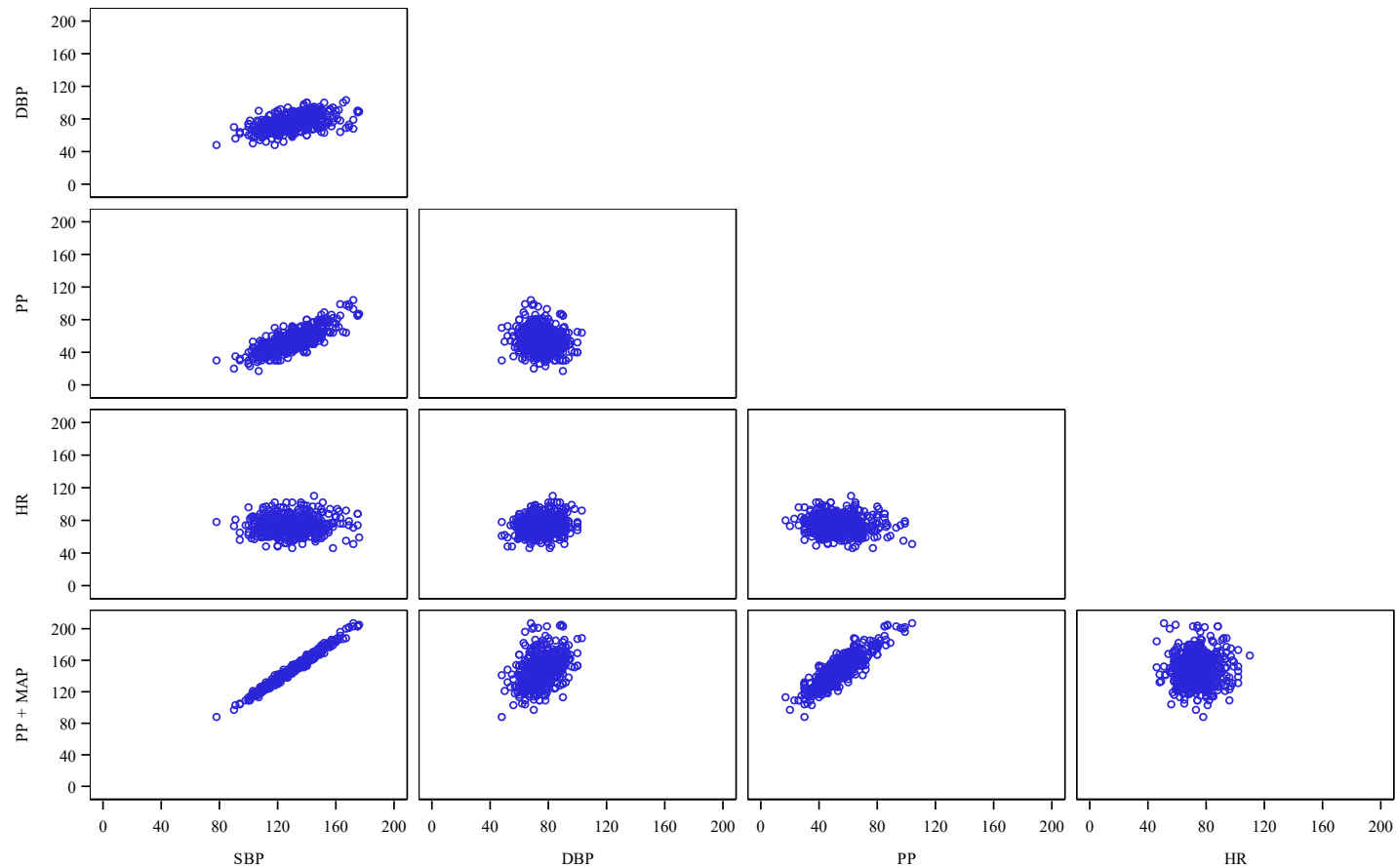
LATTICE SCATTER

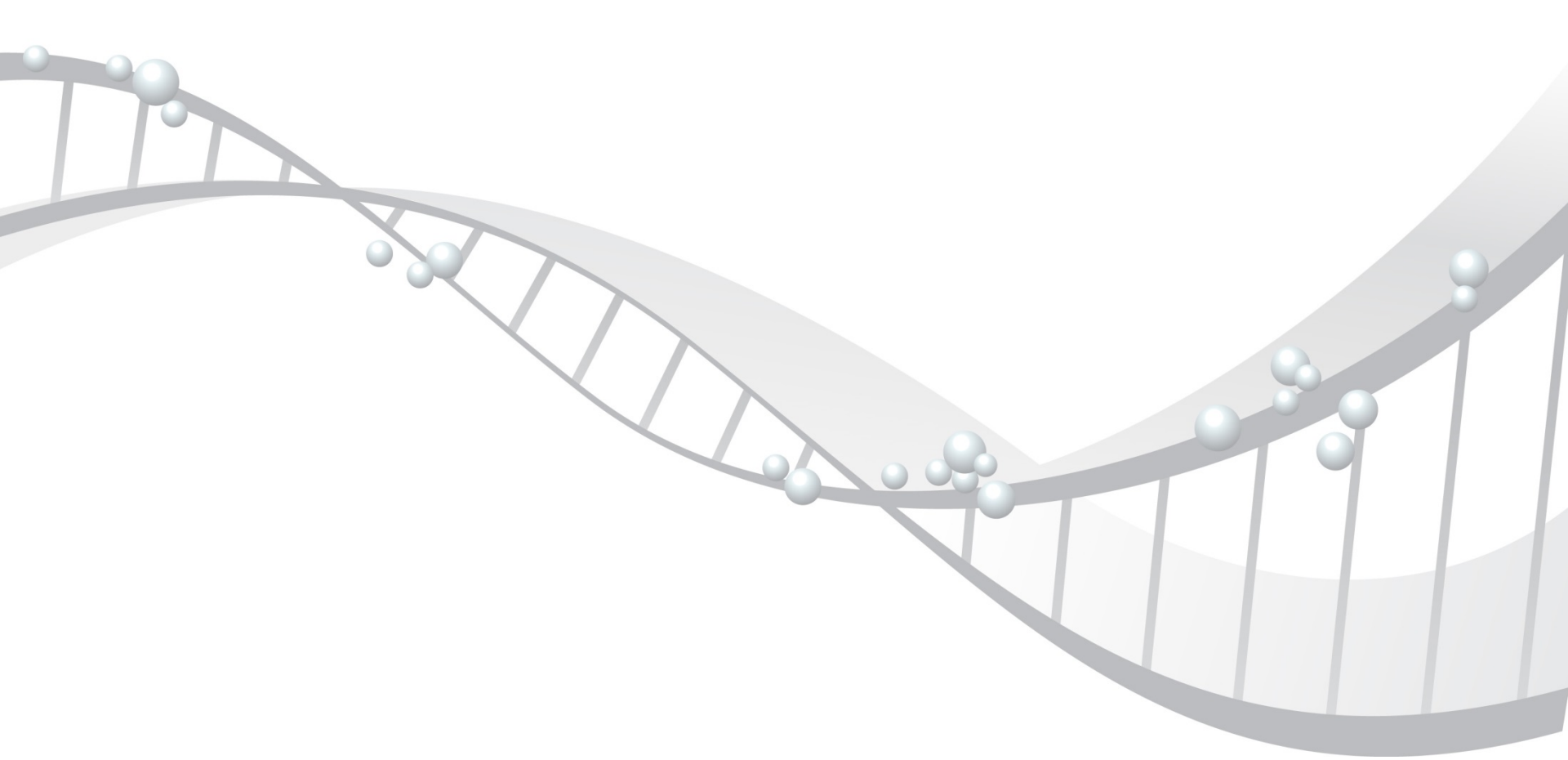
- Create individual plots with LAYOUT OVERLAY.
- Create blank plots in the unused area.

```
/* Create individual scatter plots */
/* Loop through to complete each row at a time where k = row, l = column */
/* Plot var(k) on x axis and var(l) on y axis */
  %do k = 1 %to &numvar.;
    %let y = %eval(&k. + 1);
    %do l = 1 %to &k.;
      layout overlay;
      scatterplot y=%%&var&y. x=%%&var&l. / group=trt;
    endlayout;
  %end;

/* Create blank space at any point where l >= k */
  %do z=&l. %to &numvar.;
    layout overlay;
    entry '';
  endlayout;
  %end;
%end;
```

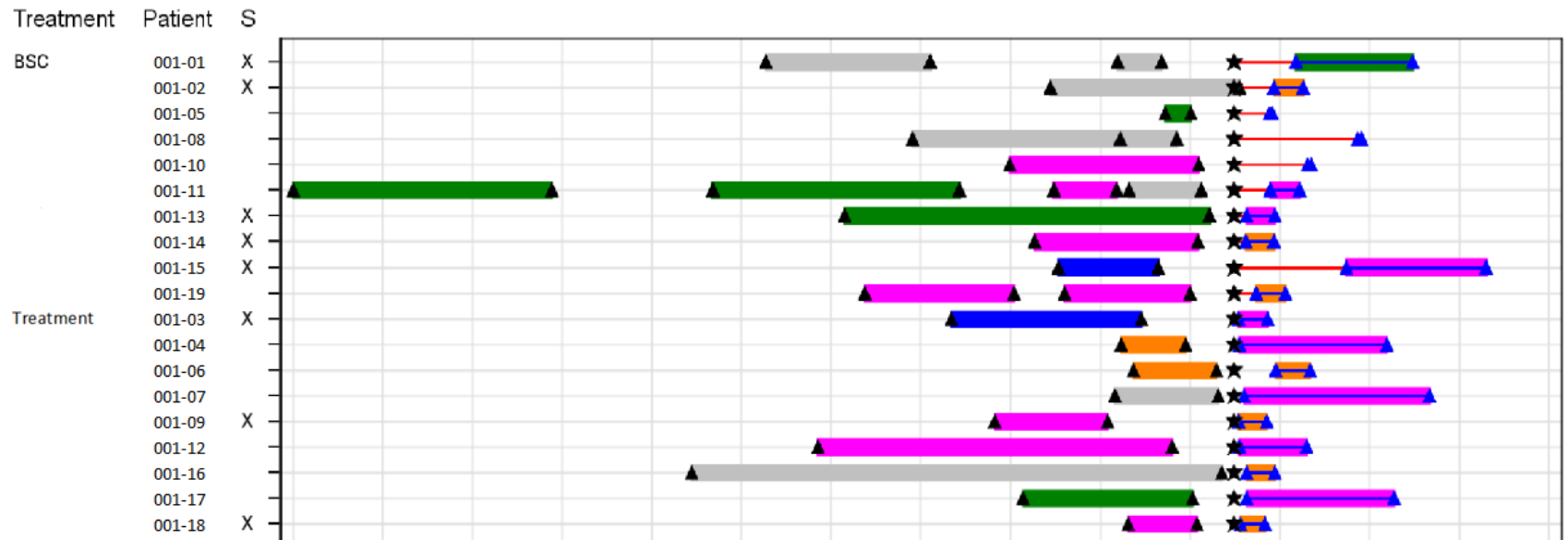
LATTICE SCATTER





SWIMMER PLOTS

SWIMMER PLOTS



Treatment duration (days)

Best response on therapy

PR UNK PD
SD CR

BSC treatment Start/End treatment
Start/End Chemotherapy Randomisation S = Prior Surgery

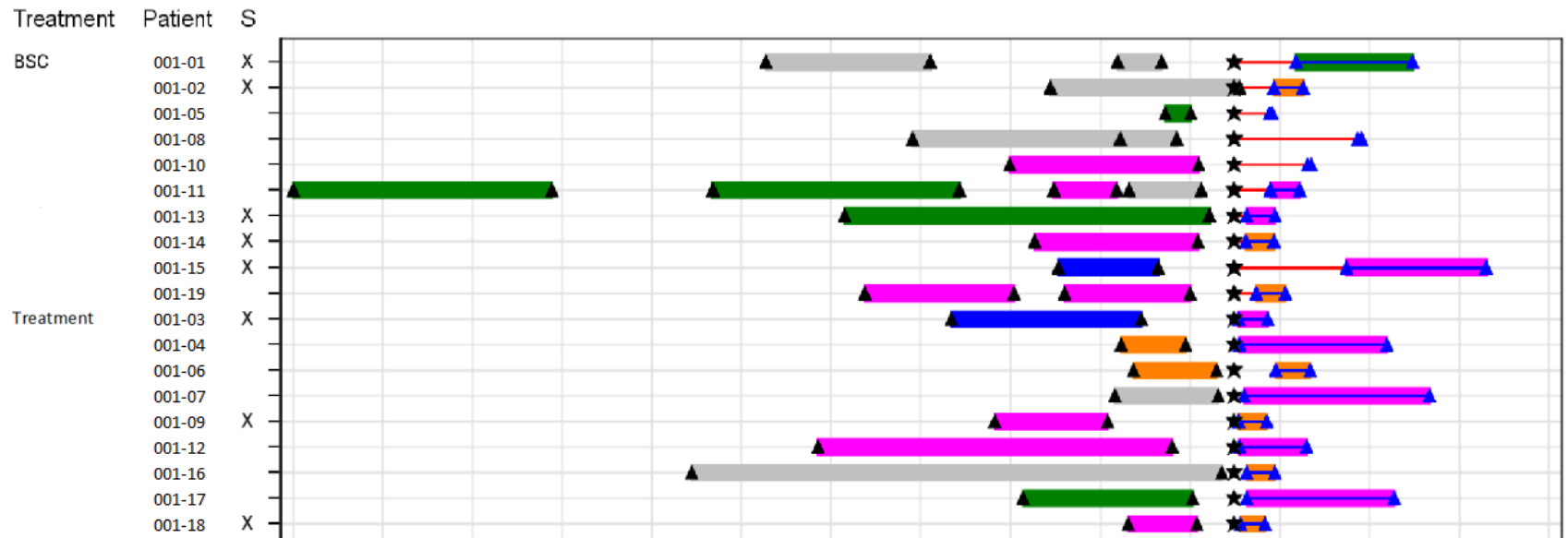
SWIMMER PLOTS

- Plotting time lines of subjects.
 - prior therapies
 - study treatment
 - best response
 - Adverse events
- Relationship between events and outcomes.
- Overlay of individual events.
 - Treatment
 - Surgery
- Visual of how subjects respond to treatment/events.

SWIMMER PLOTS

- Process data into single format.
 - Convert dates into relative days.
 - Number of days from a reference point.
 - Start and end days for bars.
- Reference point:
 - dosing date.
 - randomisation date.
 - first date to be displayed.

SWIMMER PLOTS



Treatment duration (days)

Best response on therapy

PR UNK PD
SD CR

BSC treatment Start/End treatment
Start/End Chemotherapy Randomisation S = Prior Surgery

SWIMMER PLOTS

```
|proc sgplot data=plots dattrmap=getattrs uniform=scale;

    highlow y=subjid low=ady high=aendy / type=bar group=aval groupdisplay=overlay
                                         barwidth=0.6 name='a' attrid=Color ;
    yaxis reverse display=(nolabel novalues) grid fitpolicy=splitalways;
    xaxis label="Treatment duration (days)" grid values=(0 to 1400 by 100);

    highlow y=subjid low=adyb high=aendyb / type=line group=avalt name='b'
                                         lineattrs=(pattern=1 thickness=1pt)
                                         groupdisplay=overlay attrid=Color;

    scatter y=subjid x=adyt / markerattrs=(symbol=TriangleFilled color=blue size=7)
                               name='c' legendlabel="Start/End CAP7.1 treatment";
    scatter y=subjid x=aendyt/ markerattrs=(symbol=TriangleFilled color=blue size=7);
    scatter y=subjid x=adyc / markerattrs=(symbol=TriangleFilled color=black size=7)
                               name='e' legendlabel="Start/End Chemotherapy";
    scatter y=subjid x=adyr / markerattrs=(symbol=StarFilled color=white size=7)
                               name='g' legendlabel="S = Prior Surgery";
    scatter y=subjid x=adyr / markerattrs=(symbol=StarFilled color=black size=7)
                               name='f' legendlabel="Randomisation";

    yaxistable trt0lp / position=left label="Treatment" labelpos=bottom ;
    yaxistable subjid / position=left label="Patient" labelpos=bottom ;
    yaxistable surg / position=left label="S" labelpos=bottom ;

    keylegend 'a' / position=bottomleft across = 3 title='Best response on therapy';
    keylegend 'b' 'c' 'e' 'f' 'g' / position=bottomright across = 3;

run;
```

SWIMMER PLOTS

```
highlow y=subjid low=ady high=aendy / type=bar group=aval groupdisplay=overlay  
                                         barwidth=0.6 name='a' attrid=Color ;  
yaxis reverse display=(nolabel novalues) grid fitpolicy=splitalways;  
xaxis label="Treatment duration (days)" grid values=(0 to 1400 by 100);
```

- Bars and lines plotted with HIGHLOW.
 - Low = relative start day
 - High = relative end day
 - Type = Line or Bar
 - Group = response type
- Axis set up on first of the plots.

SWIMMER PLOTS

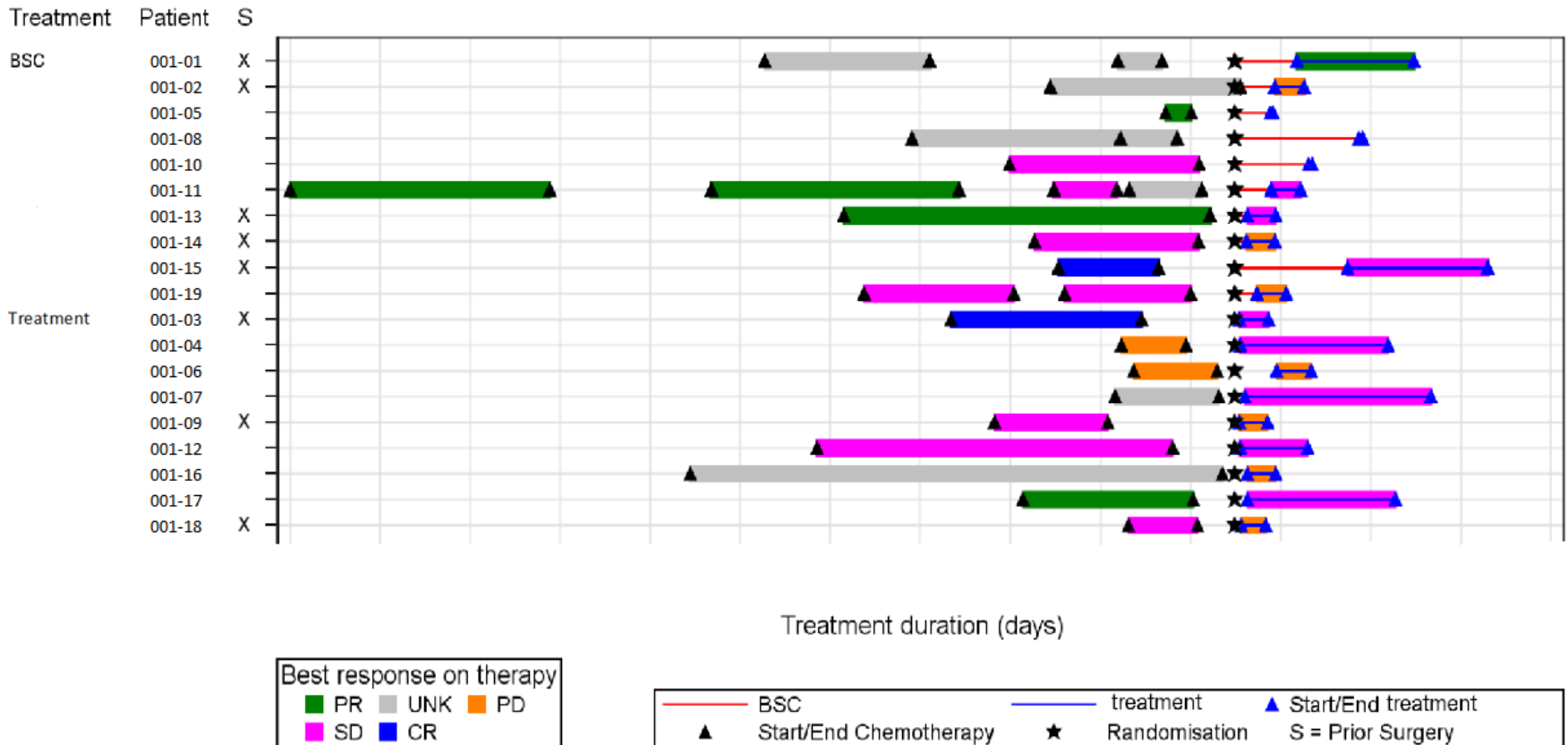
```
scatter y=subjid x=adyt / markerattrs=(symbol=TriangleFilled color=blue size=7)
                                name='c' legendlabel="Start/End CAP7.1 treatment";
scatter y=subjid x=aendyt/ markerattrs=(symbol=TriangleFilled color=blue size=7);
scatter y=subjid x=adyc / markerattrs=(symbol=TriangleFilled color=black size=7)
                                name='e' legendlabel="Start/End Chemotherapy";
```

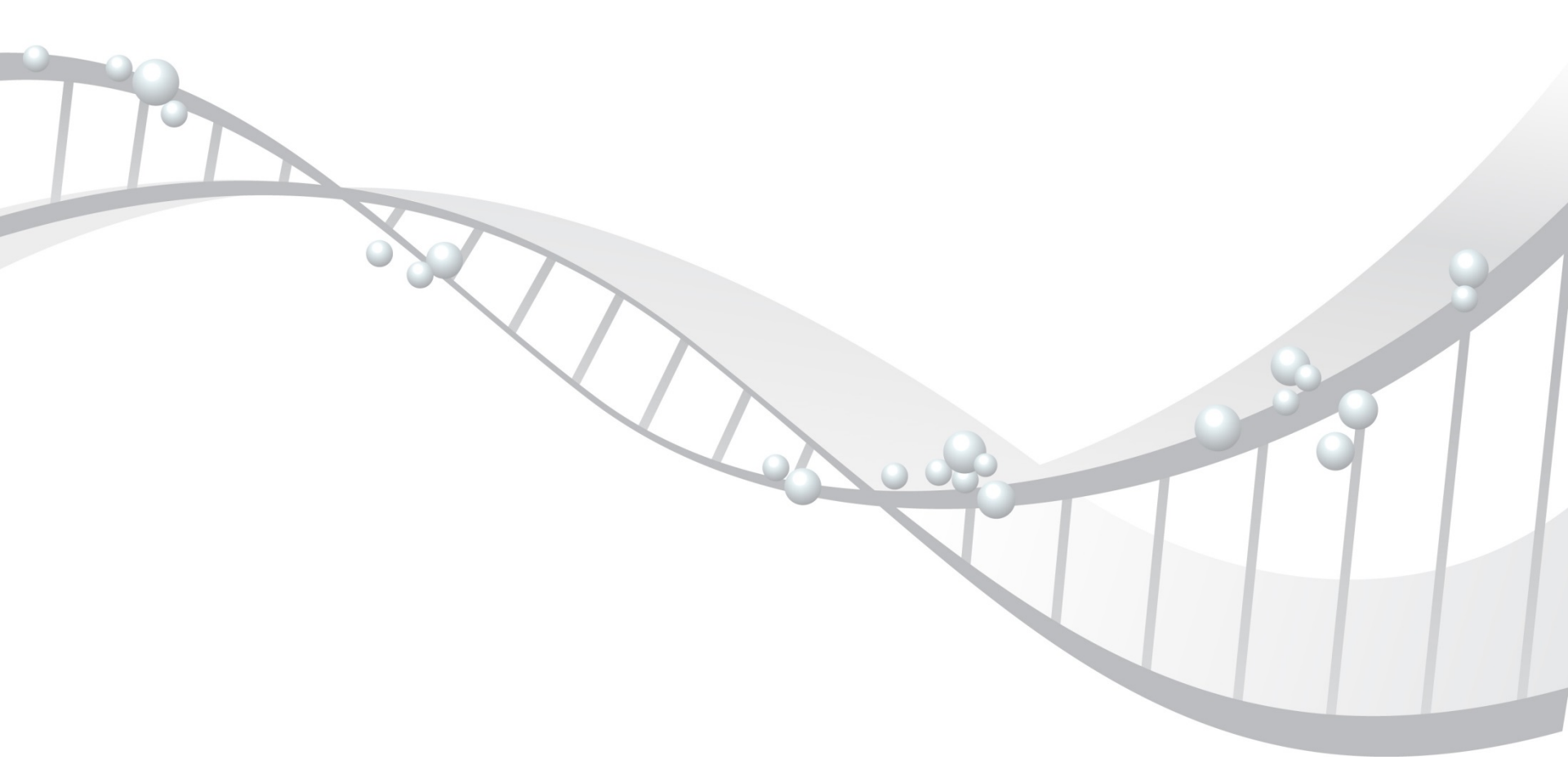
- SCATTER for each marker point.
 - X = Relative day
 - Markerattrs = properties of marker
 - Name = reference for the legend
 - Legendlabel = Label to use in the legend

- Customise legends KEYLEGEND.

```
keylegend 'a' / position=bottomleft across = 3 title='Best response on therapy';
keylegend 'b' 'c' 'e' 'f' 'g' / position=bottomright across = 3;
```

SWIMMER PLOTS





SUMMARY

SUMMARY

- GTL allows greater customisation of SAS graphs.
- Greater potential for more complex study reporting graphs.
- Create templates in advance.



Questions?