



Enhance and enrich your plots using Graph Template Language (GTL)

Mathias Lebreton, Senior Programmer | PhUSE 2015, Vienna

Disclaimer



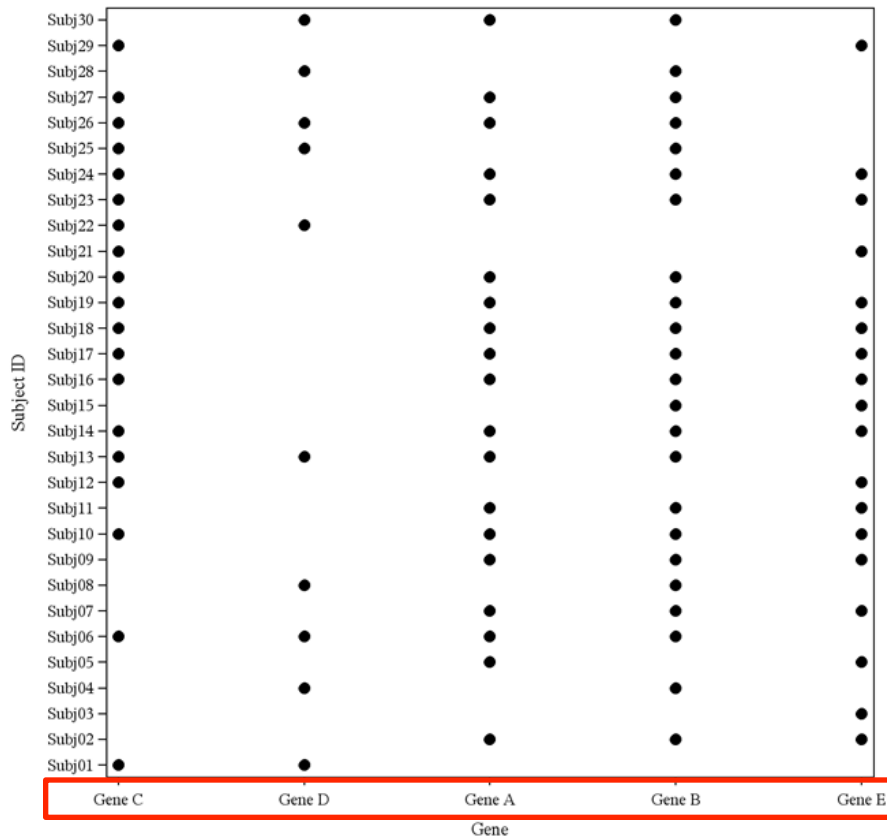
The opinions expressed in this presentation and on the following slides are solely those of the presenter and not necessarily those of Novartis. Novartis does not guarantee the accuracy or reliability of the information provided herein



■ GTL structure:

Input data derivation	{	<pre>proc sort data= input_data; by idpat Gene; run;</pre>
GTL template	{	<pre>proc template; define statgraph scatplot; begingraph /; layout overlay; <u><Plot statement>;</u> endlayout; endgraph; end; run;</pre>
SG render procedure	{	<pre>proc sgrender data= input_data template=scatplot; run;</pre>

Scatter plot: Gene alterations by subject ID



- Sample of input dataset sorted by **idpat** and **gene**:

idpat	gene
Subj01	Gene C
Subj01	Gene D
Subj02	Gene A
Subj02	Gene B
Subj02	Gene E

- Alphabetical order of axes expected

Sorting not controlled

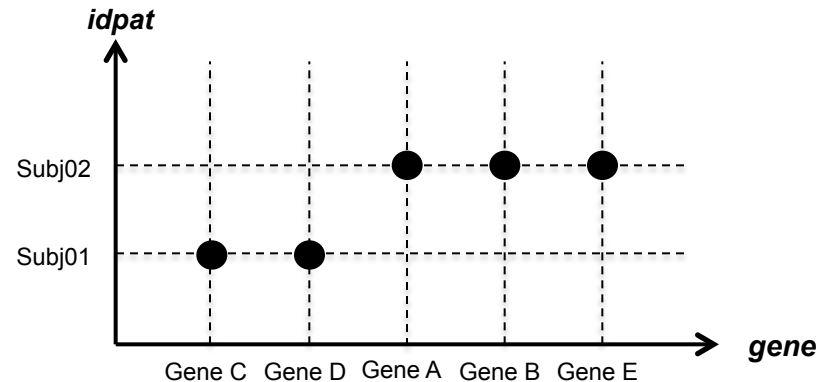
```
layout overlay;  
  scatterplot x=gene y=idpat;  
endlayout;
```

Control the axes display

- Graph and character variables:



idpat	gene
Subj01	Gene C
Subj01	Gene D
Subj02	Gene A
Subj02	Gene B
Subj02	Gene E

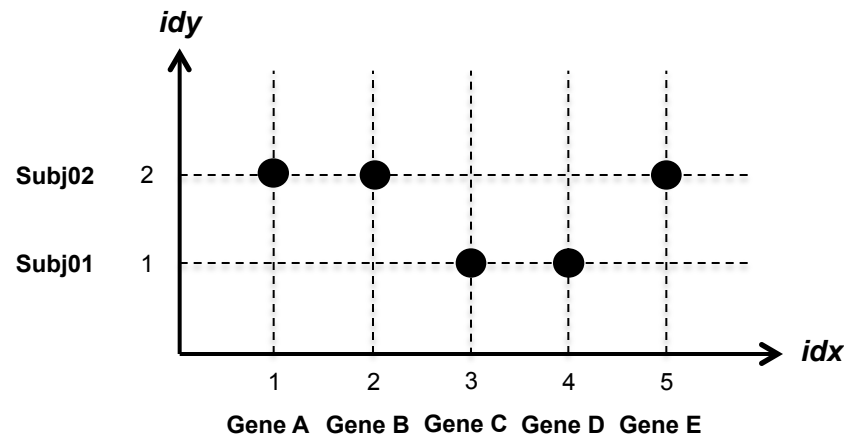


- From character variable to numeric equivalent + corresponding format:

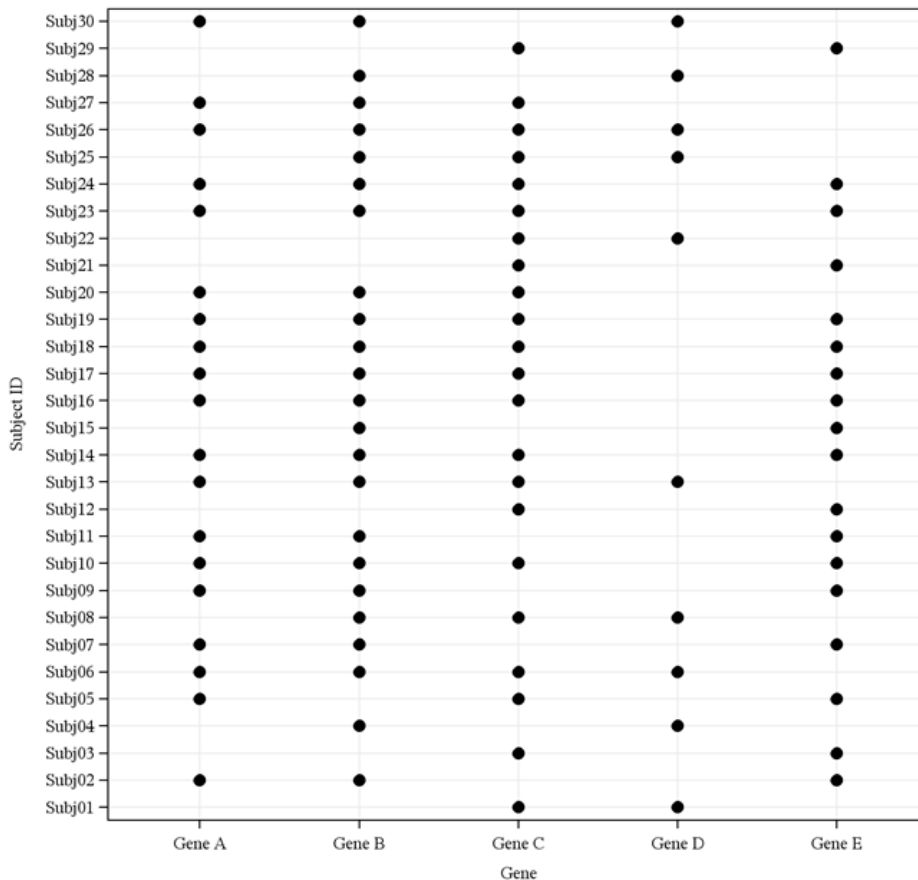


idpat	idy	gene	idx
Subj01	1	Gene C	3
Subj01	1	Gene D	4
Subj02	2	Gene A	1
Subj02	2	Gene B	2
Subj02	2	Gene E	5

Addition of numeric variables



Numeric variables for axes



```
layout overlay;  
  scatterplot y=idy x=idx;  
endlayout;
```

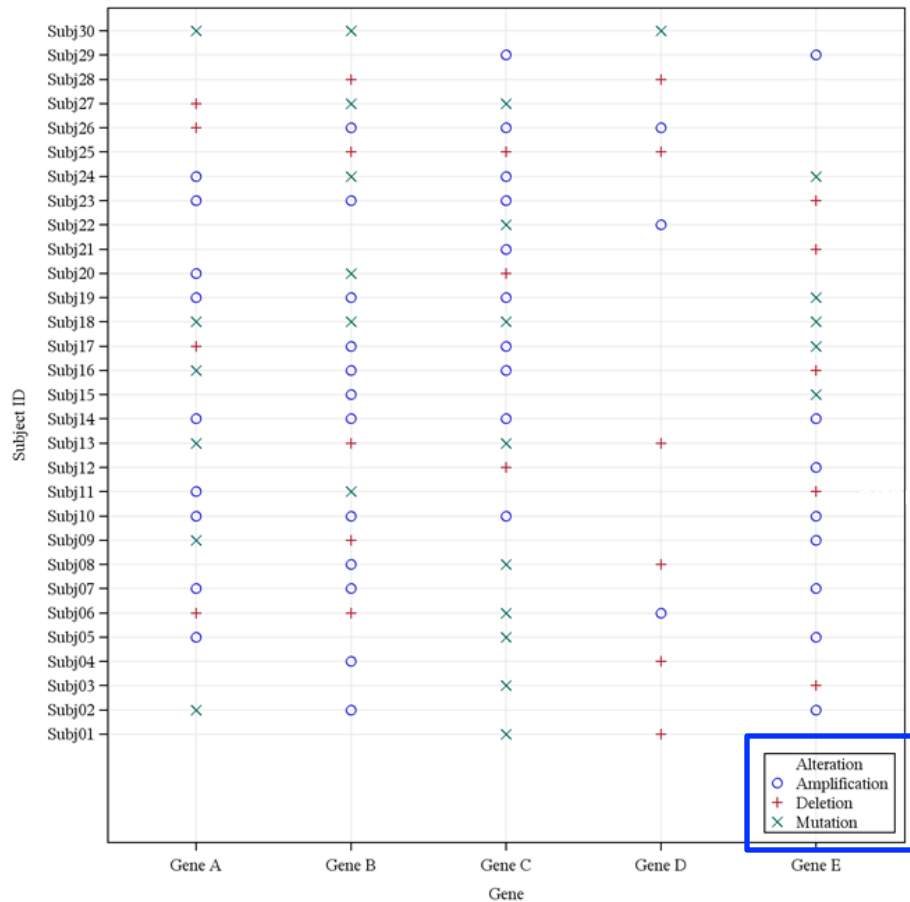
■ Control of axes:

- Expected display and order
- Display sorting independant of input dataset

■ Enrich by adding a group variable to distinguish the type of alterations :

- Type of alterations : Amplification, deletion or mutation

Addition of Group var : Type of alterations



```
layout overlay;
  scatterplot y=idy x=idx / group= alteration;
  <discretelegend statement>;
endlayout;
```

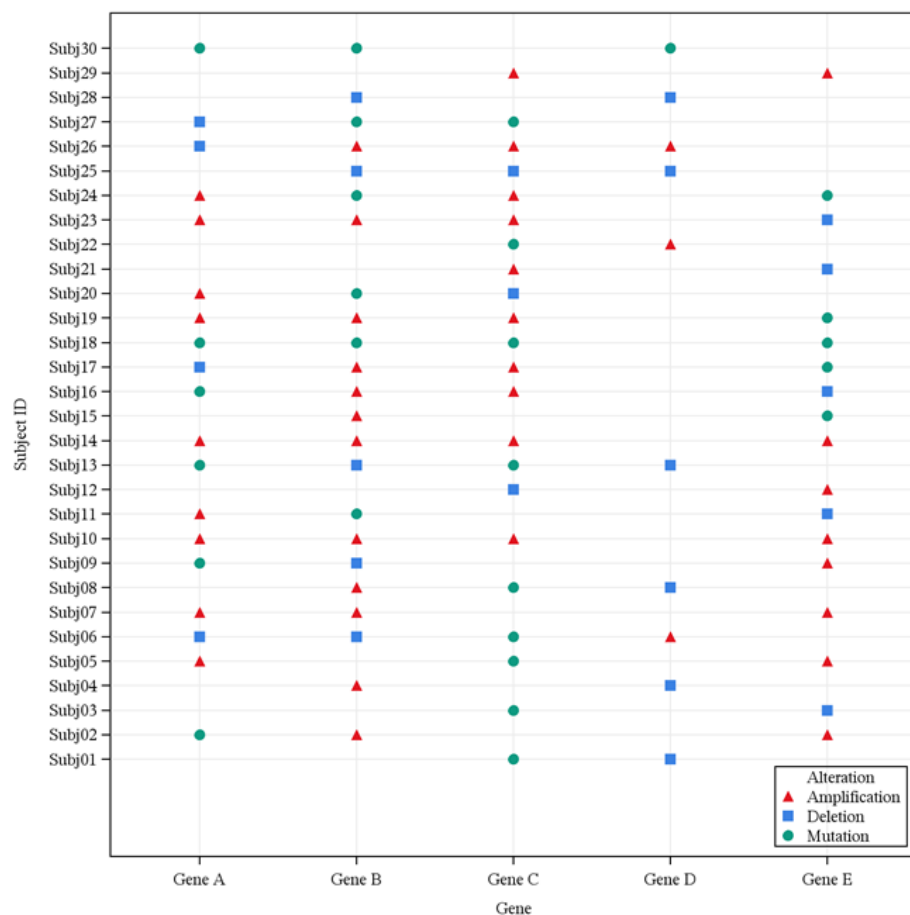
idpat	idy	gene	idx	alteration
Subj02	2	Gene B	2	Amplification
Subj02	2	Gene E	5	Amplification
Subj01	1	Gene D	4	Deletion
Subj01	1	Gene C	3	Mutation
Subj02	2	Gene A	1	Mutation

■ Display of legend:

- Input dataset **dependant**
- Sort input dataset by the group variable

- ~~Current symbol and color defined by default by SAS~~
- Control the color and symbol of each type of alteration

Attribute map dataset



Attribute map dataset:

ID	VALUE	MARKERCOLOR	MARKERSYMBOL
alt	Amplification	Red	trianglefilled
alt	Deletion	Blue	Squarefilled
alt	Mutation	Green	circlefilled

Name: **Attrib_data**

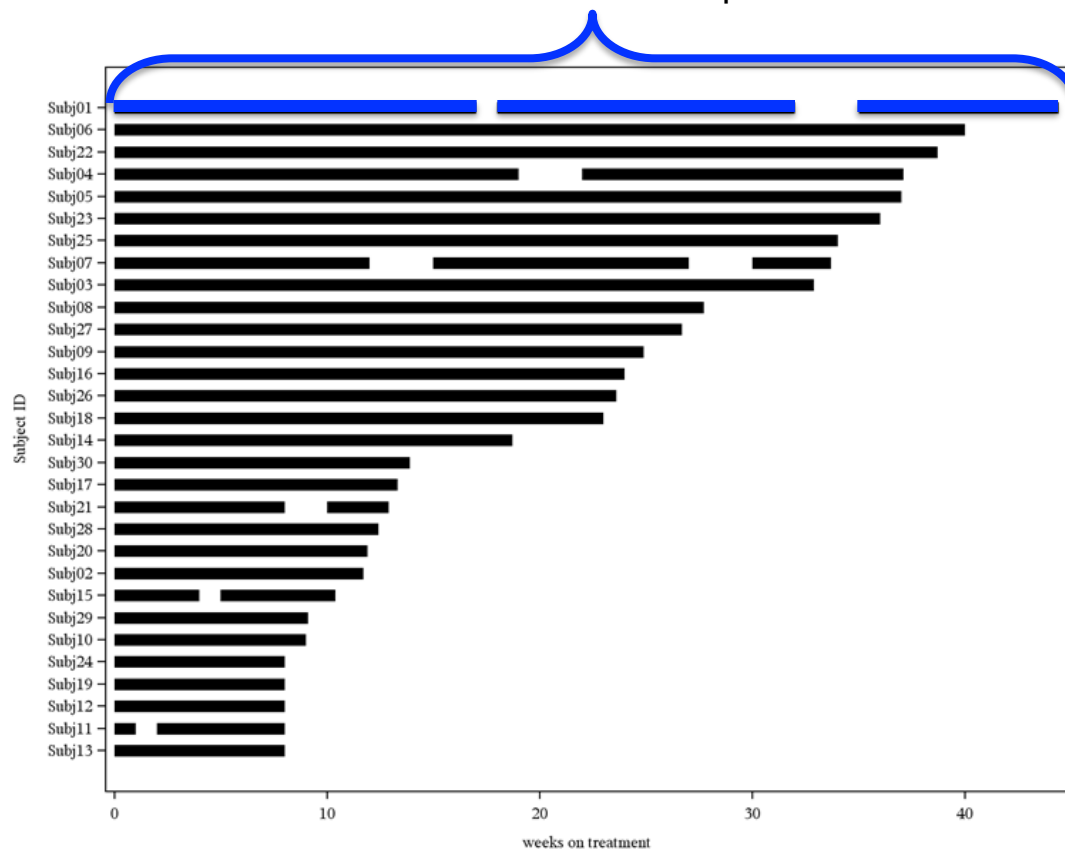
- ID and VALUE are mandatory
- Assign a color/symbol for each type of alterations
- Can be used in other graphs

```
proc sgrender data=input_data template=scatplot dattrmap=attrib_data;
  dattrvar alteration='alt';
run;
```



Swimming Plot: Drug exposure by subject ID

Overall treatment exposure

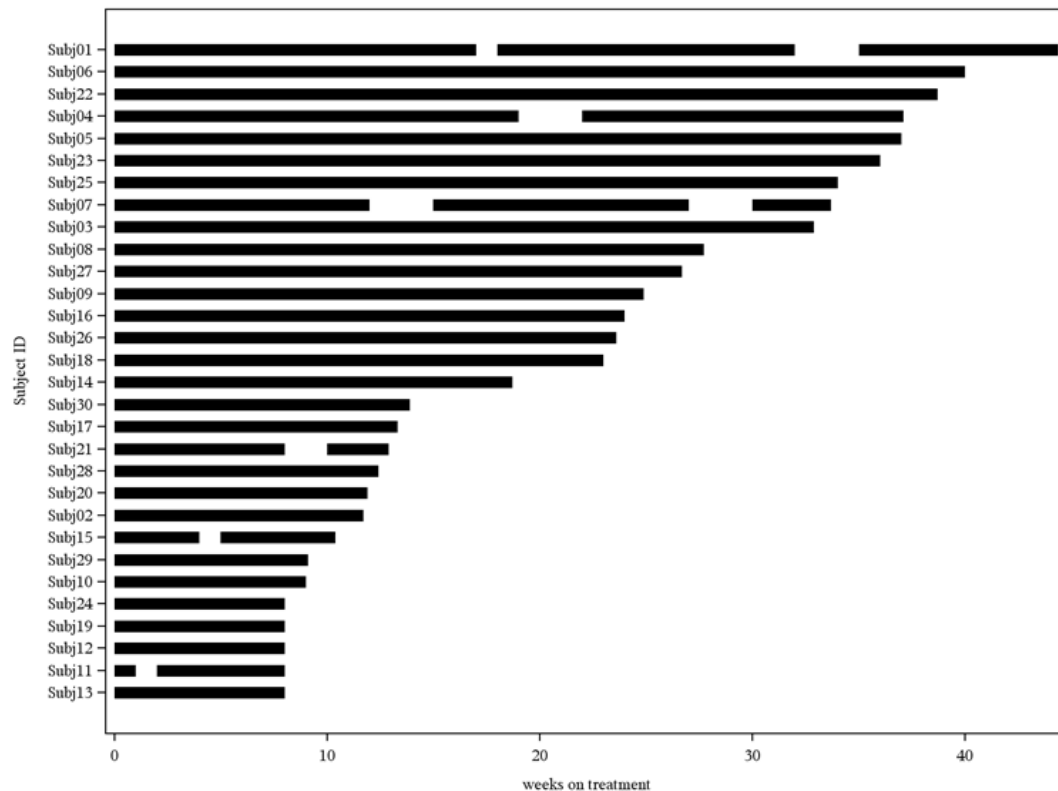


idpat	duration	start	end
Subj01	44	0	17
Subj01	44	18	32
Subj01	44	35	44

- Sorted by overall duration of exposure (duration)

```
layout overlay/;  
  hightlowplot y=idpat low=start high=end;  
endlayout;
```

Similarly...



1 – Basic graph

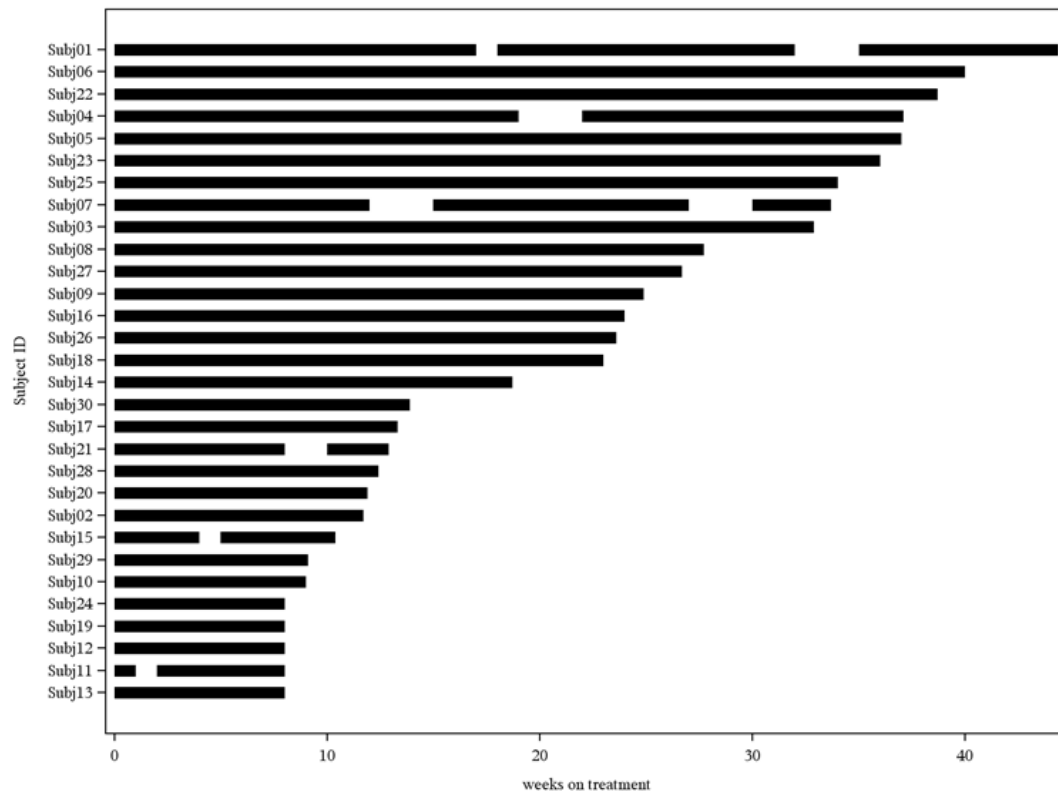
2 – Formatting of idpat to numeric

3 – Addition of group variable

4 – Addition of attribute map dataset

```
layout overlay/;  
  hightlowplot y=idpat low=start high=end;  
endlayout;
```

Similarly...



1 – Basic graph

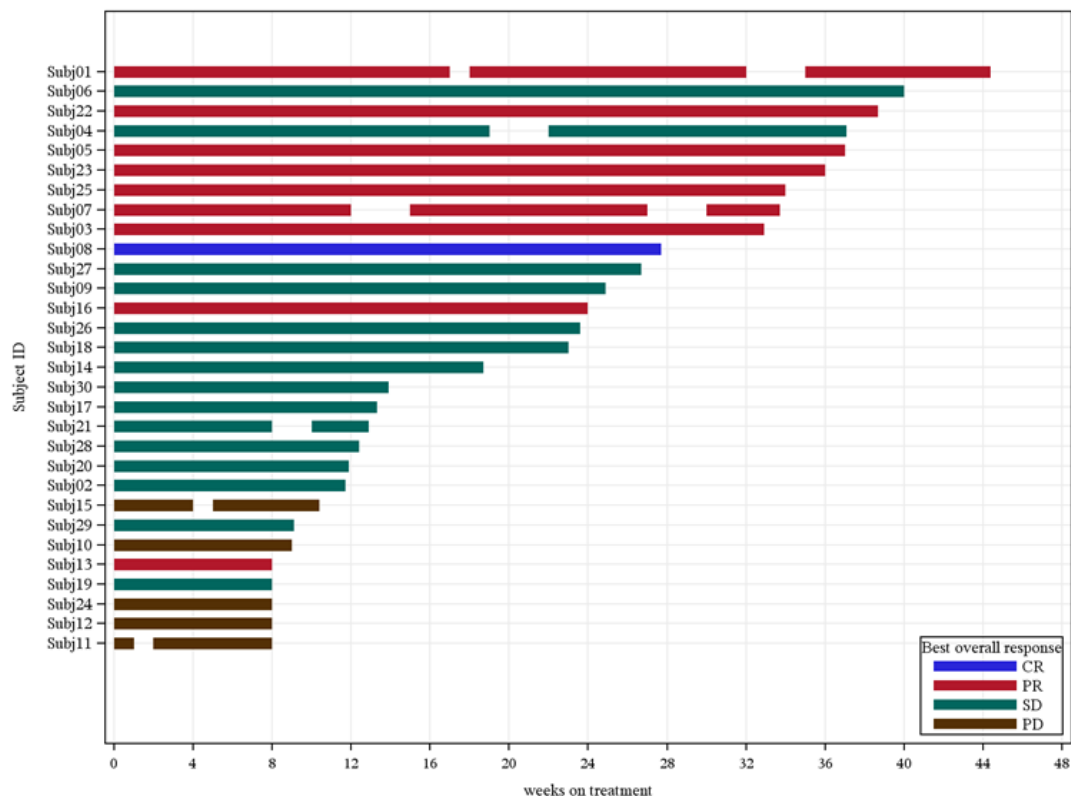
2 – Formatting of idpat to numeric

3 – Addition of group variable

4 – Addition of attribute map dataset

```
layout overlay/;  
  hightlowplot y=idy low=start high=end;  
endlayout;
```

Similarly...



1 – Basic graph

2 – Formatting of idpat to numeric

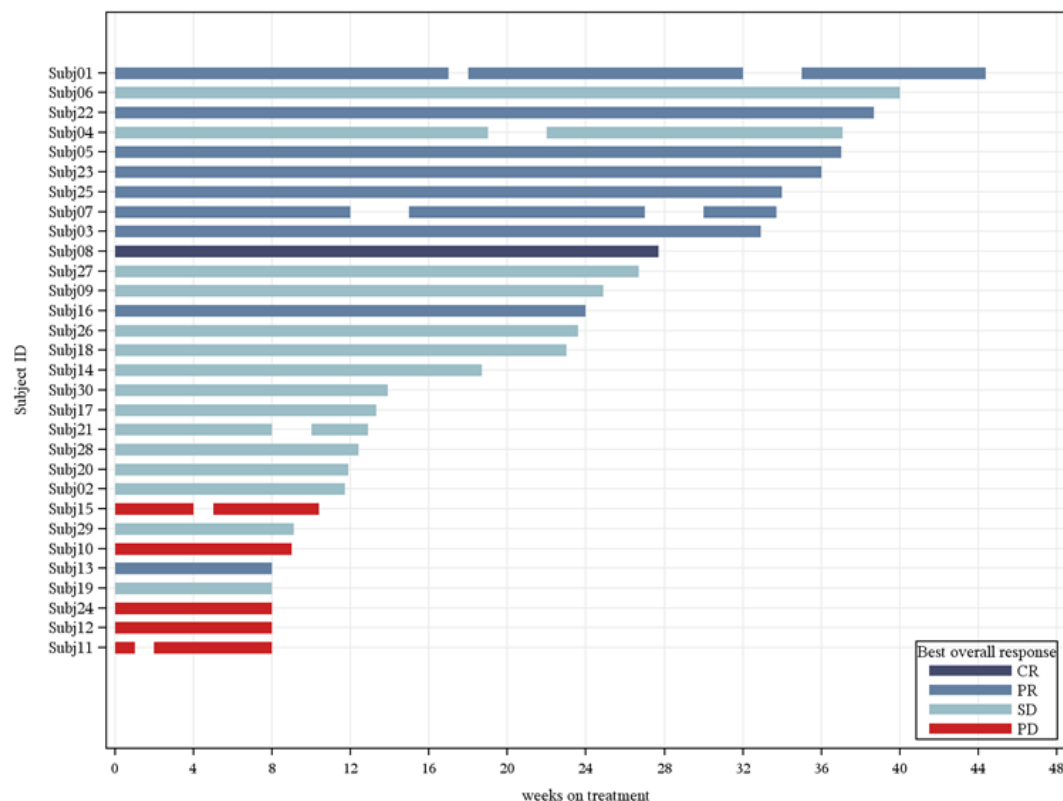
3 – Addition of group variable

4 – Addition of attribute map dataset

idpat	duration	start	end	bor
Subj01	44	0	17	PR
Subj01	44	18	32	PR
Subj01	44	35	44	PR

```
layout overlay/;  
  hightlowplot y=idpat low=start high=end /group=bor;  
endlayout;
```

Similarly...



1 – Basic graph

2 – Formatting of idpat to numeric

3 – Addition of group variable

4 – Addition of attribute map dataset

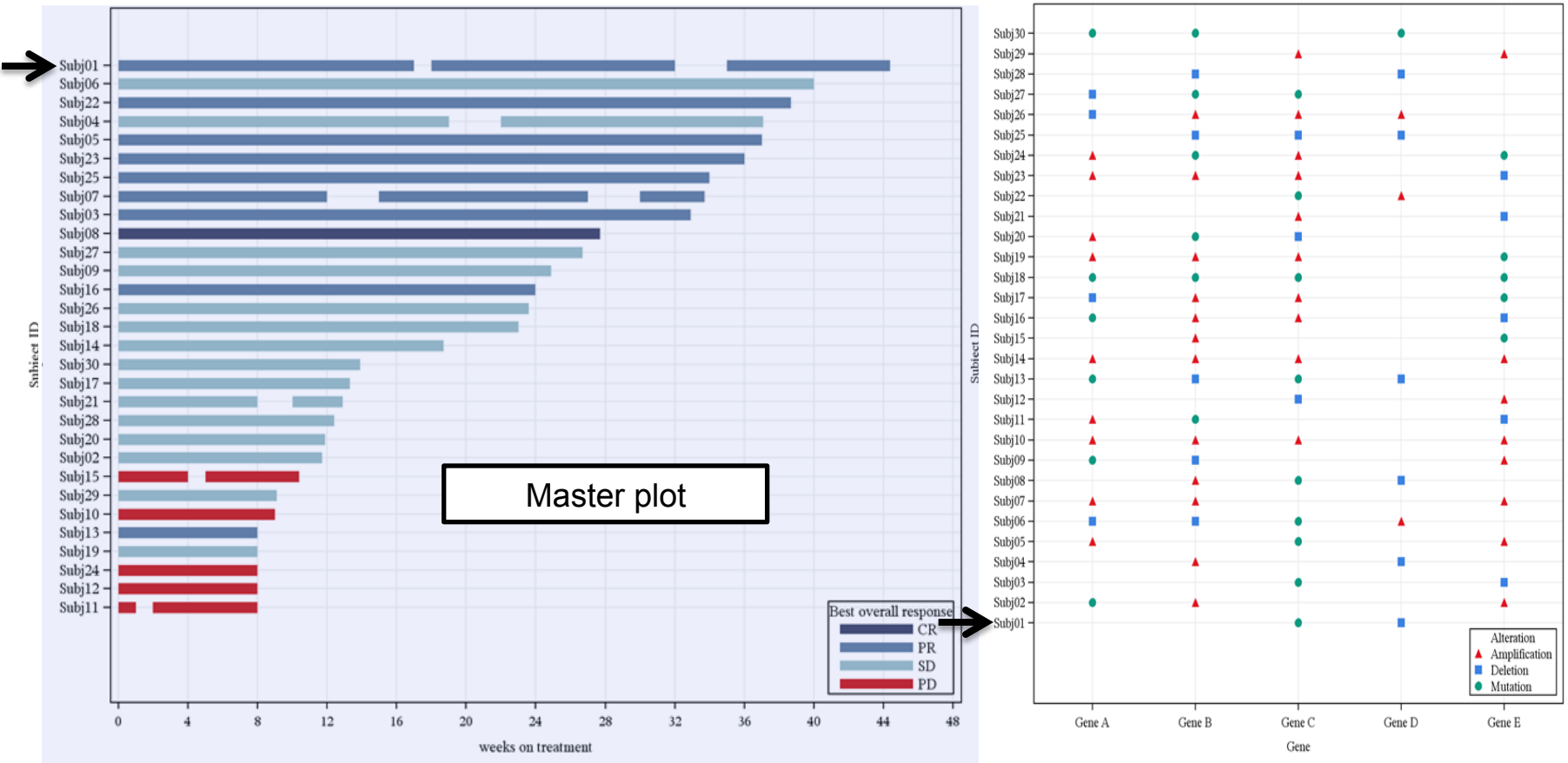
ID	VALUE	LINECOLOR
bor	CR	Dark blue
bor	PR	Blue
bor	SD	Light blue
bor	PD	Red

```
layout overlay;  
  hightlowplot y=idpat low=start high=end / group=bor;  
endlayout;
```



What's next ?

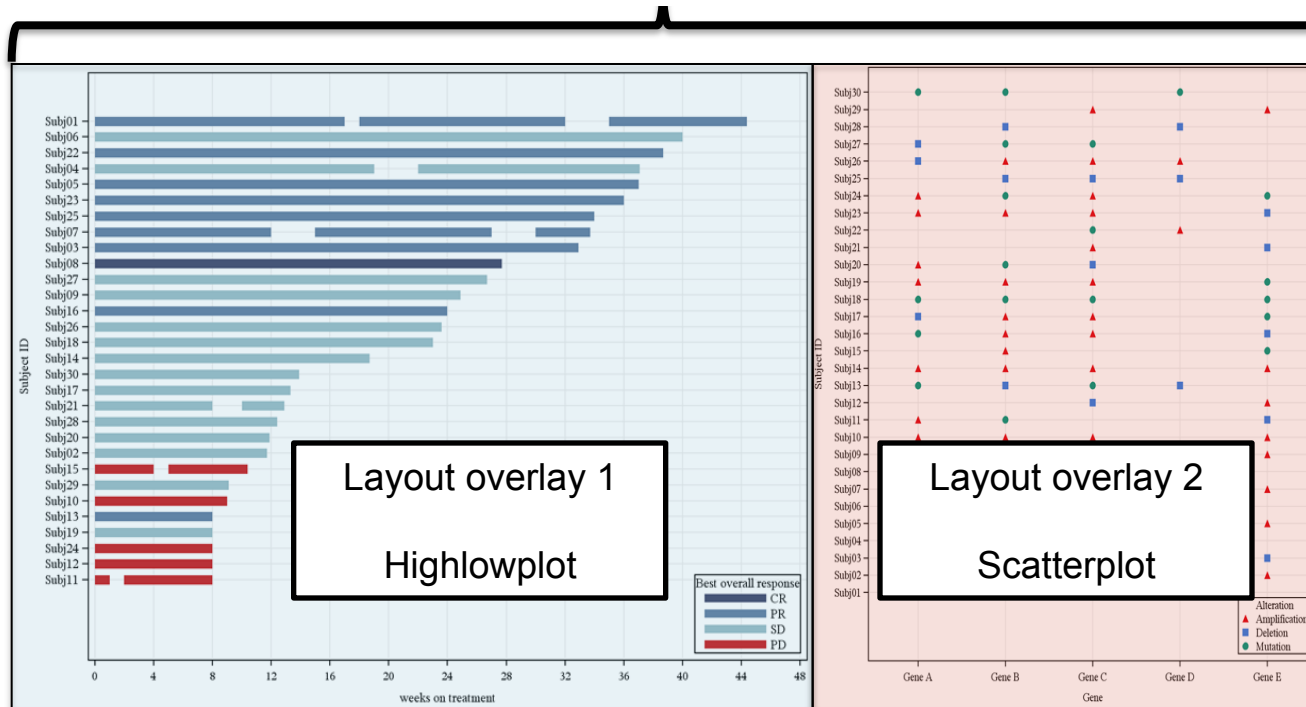
Combine two individual plots into one single graph



Combine GTL templates and input datasets

Layout lattice

Layout lattice
(1 row and 2 columns - 70% and 30%)



```
layout lattice /columns=2 rows=1 columnweights=(0.7 0.3);  
  layout overlay;  
    <Highlowplot statement>  
  endlayout;  
  layout overlay;  
    <Scatterplot statement>  
  endlayout;  
endlayout;
```

Combine input datasets

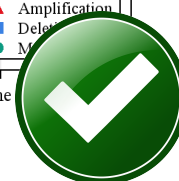
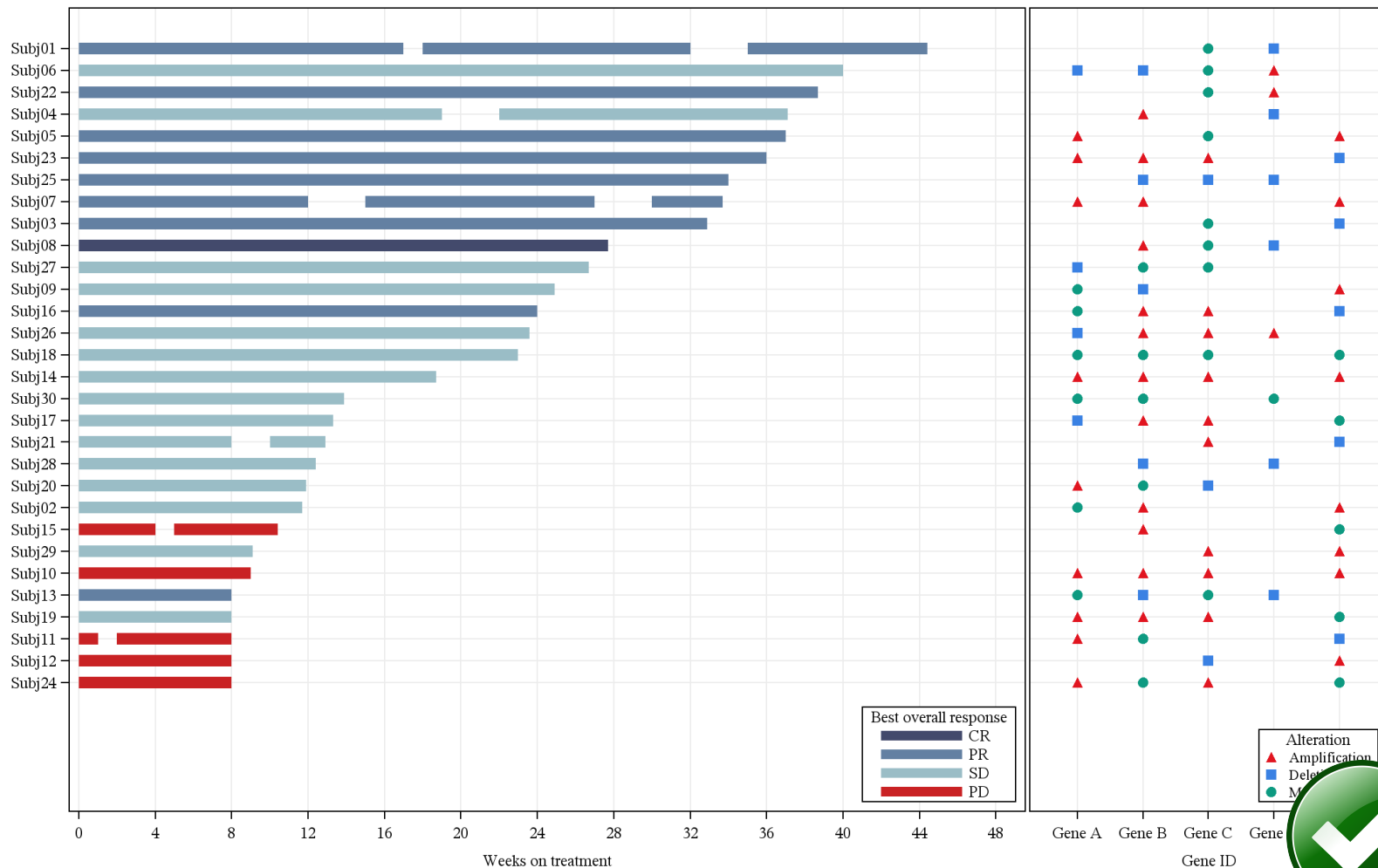


	idpat	duration	bor	Start	end	gene	alteration
Swimplot	Subj01	44	PR	0	17		
	Subj01	44	PR	18	32		
	Subj01	44	PR	35	44.4		
	Subj02	11	SD	0	11		
Scatterplot	Subj01	44				Gene C	Mutation
	Subj01	44				Gene D	Deletion
	Subj02	11				Gene A	Mutation
	Subj02	11				Gene B	Amplification
	Subj02	11				Gene E	Amplification
Scatterplot	Subj01					Gene D	Deletion
	Subj02					Gene A	Mutation
	Subj02					Gene B	Amplification
	Subj02					Gene E	Amplification

1. Set of input datasets
2. Retain variables needed for the sorting

- Same sorting in both plots
- Character variables used in X/Y-axes can then be formatted to numeric as seen in the previous cases

Combined plot

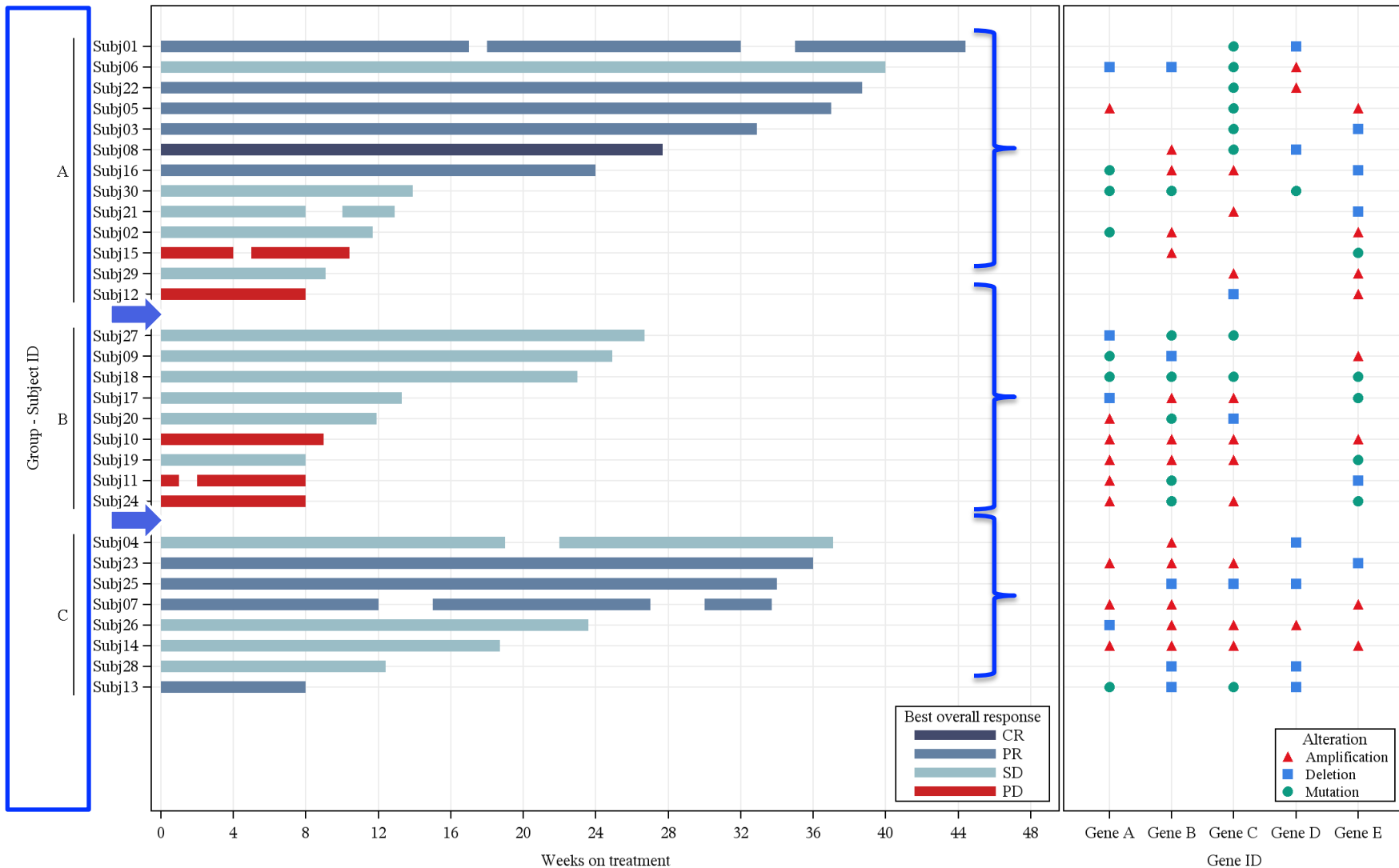




Understand the SAS default options to set them better

- Character variable in axes ?
 - Use numeric equivalent with corresponding format
- Addition of a group variable ?
 - Sort the dataset to control the legend order
 - Control colors and symbols with attribute map dataset
- Display multiple distinct plots in one graph?
 - Combine them with layout lattice and overlay
- More details and methods in the PhUSE paper CS08

Further customization is also becoming feasible ...



Thank you!

