Swimmer Plots in Clinical Programming

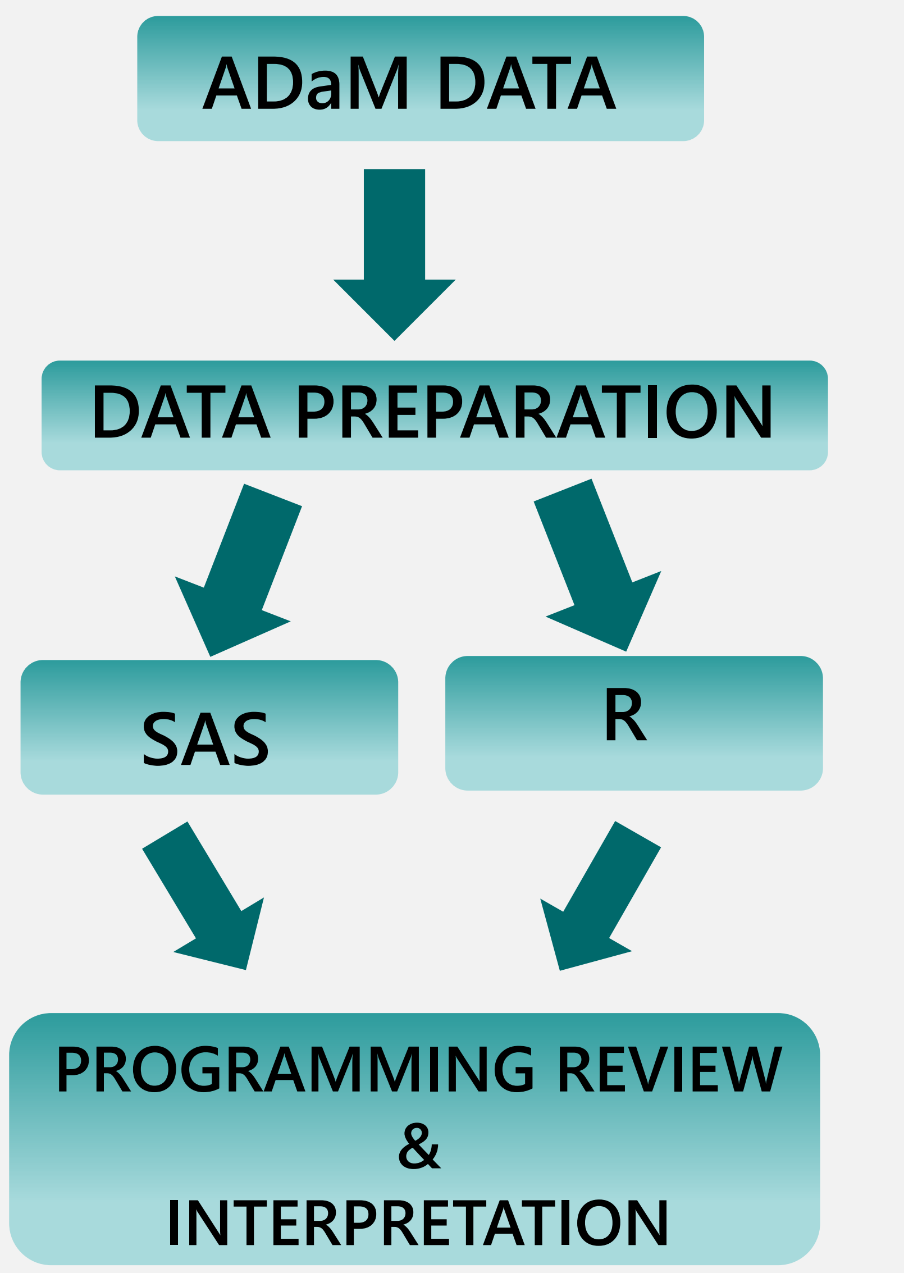
Implementation and Visualization Using SAS and R

Author: Ramya V Kini, Biostatistician, Atorus Research
Co-Author: Vinay S, Manager, Statistical Programming, Atorus Research

WHY SWIMMER PLOTS

- To Visualize individual subject treatment response over time.
- Commonly used in oncology trials to display subject-level treatment timelines.
- Shows whether, when, and for how long a subject responded to treatment.
- Simultaneously depicts treatment duration, response patterns, progression, and censoring.
- Reveals subject-level insights that are not evident from aggregate efficacy summaries.

END-TO-END FLOW



ADaM DATA

DATA PREPARATION

SAS

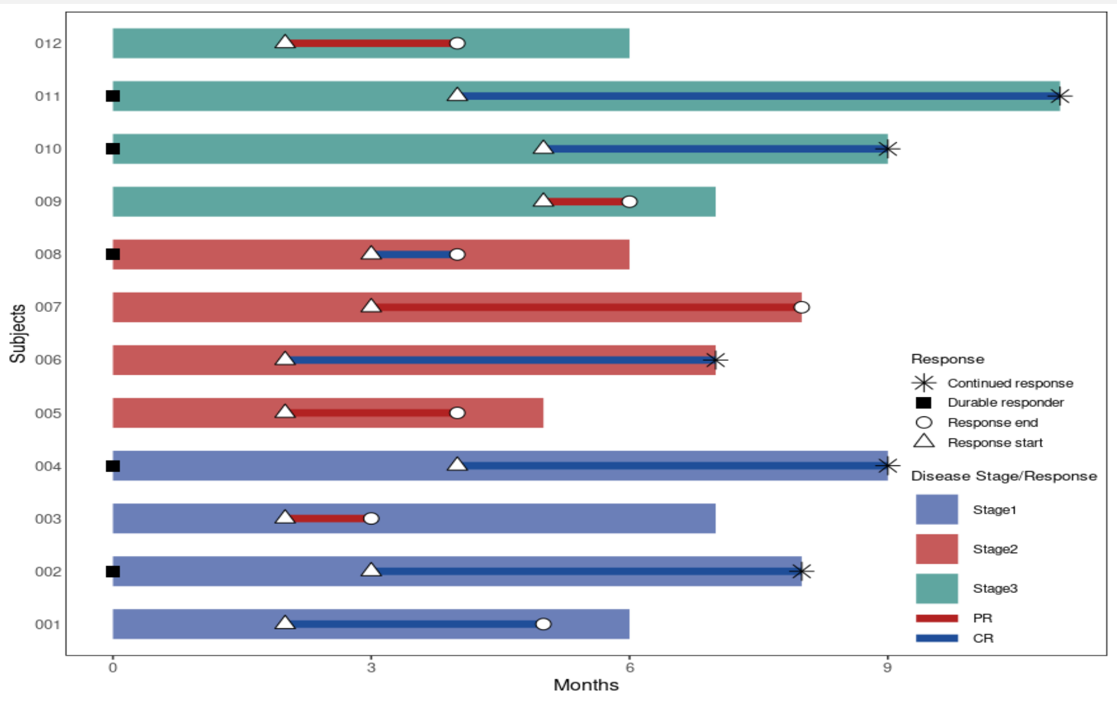
R

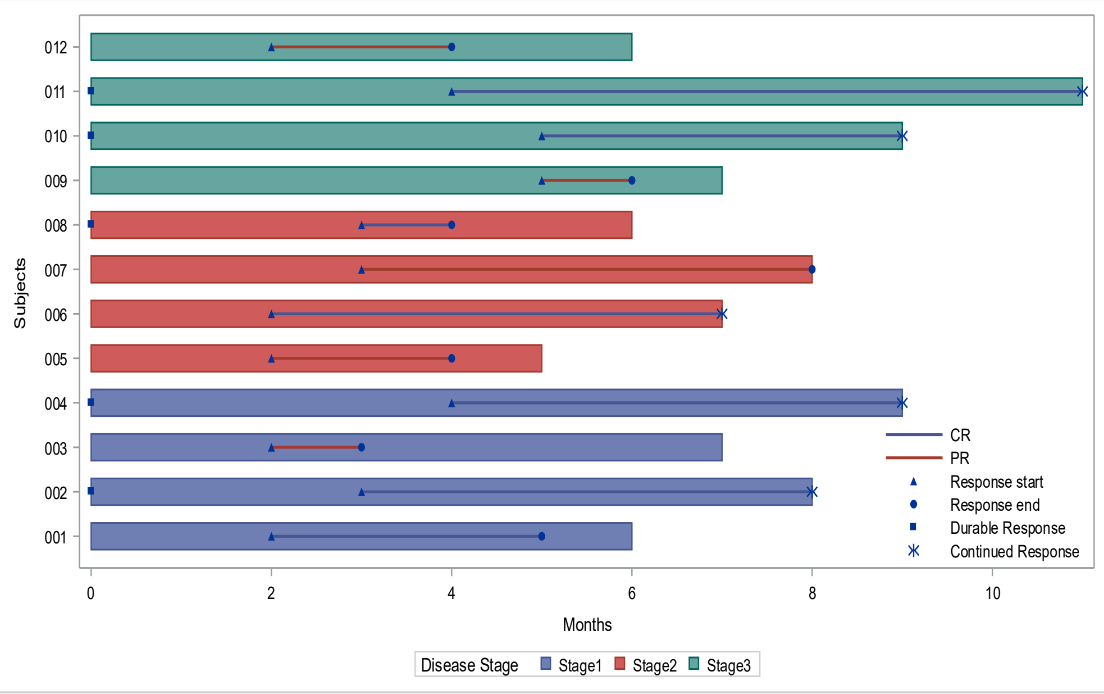
PROGRAMMING REVIEW
&
INTERPRETATION

DATA STRUCTURE

- One record per subject with treatment start and end.
- Response timing captured using start and end variables.
- Disease stage encoded as categorical variable.
- Durable and continued responses derived programmatically.
- Structure supports both SAS and R implementations.

SWIMMER PLOT COMPARISON
R (ggplot2) vs SAS (PROC SGPLOT)





R IMPLEMENTATION

```
# Treatment duration (main swimmer bar)
ggplot(df, aes(y = usubjid)) +
  geom_segment(
    aes(x = trt_start,
        xend = trt_end,
        yend = usubjid,
        color = stage),
    linewidth = 10,
    lineend = "butt"
  ) +
# Response duration
  geom_segment(
    aes(x = resp_start,
        xend = response_end_plot,
        yend = usubjid,
        color = resp_type),
    linewidth = 2.5
  ) +
# Response start
  geom_point(
    aes(x = resp_start,
        shape = "Response start"),
    size = 3.5, fill = "white"
  ) +
# Response end (ONLY if not continued)
  geom_point(
    data = df[df$continued == 0, ],
    aes(x = resp_end, shape = "Response end"),
    size = 3.5, fill = "white"
  ) +
# Durable responder
  geom_point(
    data = df[df$durable == 1, ],
    aes(x = trt_start,
        shape = "Durable responder"),
    size = 3.5, color = "black"
  ) +

# Continued response (asterisk)
  geom_point(
    data = df[df$continued == 1, ],
    aes(x = trt_end, shape = "Continued response"),
    size = 4, color = "black"
  ) +
# Legend and Axes
  scale_color_manual(
    values = c(stage_cols, resp_cols),
    breaks = c("Stage1","Stage2","Stage3",
               "PR","CR"),
    name = "Disease Stage/Response"
    ) +
  scale_shape_manual(
    values = c(
      "Continued response" = 8,
      "Durable responder"  = 15,
      "Response end"       = 21,
      "Response start"     = 24
    ),
    name = "Response"
  ) +

  labs(
    x = "Months",
    y = "Subjects"
    ) +

 theme(
   # Remove grids
   panel.grid.major = element_blank(),
   panel.grid.minor = element_blank(),

   # White background
   panel.background = element_rect(fill = "white", color = NA),
   plot.background  = element_rect(fill = "white", color = NA),

   # Axes & border
   axis.line.y  = element_blank(),
   axis.ticks.y = element_blank(),
   panel.border = element_rect(color = "black", fill = NA),

   # Legend position
   legend.position = c(1, 0),
   legend.justification = c(1, 0),
   legend.background = element_blank(),
   legend.key = element_blank(),
   legend.title = element_text(size = 9),
   legend.text  = element_text(size = 8),
   legend.key.size = unit(0.5, "cm"),
   legend.spacing.x = unit(0.1, "cm"),
   legend.spacing.y = unit(0.05, "cm")
 )
```

SAS IMPLEMENTATION

```
proc sgplot data=swimmer;
/*----------------------------------
    Treatment duration (main swimmer bar)
  ----------------------------------*/
  highlow y=usubjid
          low=trt_start
          high=trt_end
          / name="stage" type=bar
            group=stage
            barwidth=0.6;
  /*----------------------------------
    Response duration (overlay line)
  ----------------------------------*/
  highlow y=usubjid
          low=resp_start
          high=response_end_plot
          / name="response" type=line
            group=resp_type
            lineattrs=(thickness=2);

/*----------------------------------
  Response start marker
----------------------------------*/
scatter y=usubjid x=resp_start
        / markerattrs=(symbol=trianglefilled
                       size=8)
      name="respstart"
      legendlabel="Response start";

/*----------------------------------
  Response end marker
----------------------------------*/
scatter y=usubjid x=resp_end
        / markerattrs=(symbol=circlefilled
                       size=8)
      name="respend"
      legendlabel="Response end";


 /*----------------------------------
   Durable responder marker
 ----------------------------------*/
   scatter y=usubjid x=durable_x
      / markerattrs=(symbol=squarefilled
                     size=8)
        name="durable"
        legendlabel="Durable Response";

 /*----------------------------------
   Continued Response marker
 ----------------------------------*/
   scatter y=usubjid x=arrow_xend
     / markerattrs=(symbol=asterisk
                    size=10 )
       name="continued"
       legendlabel="Continued Response" ;

 /*----------------------------------
   Axes
 ----------------------------------*/
   xaxis label="Months";
   yaxis label="Subjects Received Study Drug"
   discreteorder=data;

   keylegend "stage"  / border
                        location=outside
   position=bottom across=5
   title="Disease Stage";
   keylegend  "response" "respstart"
       "respend" "durable" "continued"
      / noborder location=inside
        position=bottomright
        across=1;
   run;
```

SAS vs R: PROS & CONS

SAS (SGPLOT)
Pros:

- Standardized, validated outputs for clinical submissions.
- Attribute maps make consistent colour coding for disease stages easy.

Cons:

- Legends and arrows need extra coding/templates.
- Limited interactivity.

R (ggplot2)
Pros:

- Multiple visualization packages (ggplot2, ggswim, swimplot) enable flexible layered plots.
- Allows interactive Visualization using ggplotly.

Cons:

- Memory bound for huge datasets.
- Legends can get crowded with many aesthetics.

CONCLUSION

Both SAS and R generate equivalent subject-level swimmer plots when built on standardized input data. Differences arise primarily in customization flexibility and development efficiency.

SAS is preferred for standardized, submission-ready swimmer plots with consistent formatting and validated outputs.

R is well suited for exploratory analysis, offering extensive customization and optional interactive visualizations.

Beyond visualization, swimmer plots complement aggregate efficacy measures by visually linking individual response duration, durability to overall treatment effects, enhancing clinical interpretation.