Paper DV06

Cracking the Curve: Unlocking Clinical Insights with Ridgeline Plots

Mariam Babayan, BioBrain Pro LLC, Yerevan, Armenia

ABSTRACT

Ridgeline plots represent an emerging data visualization technique that enables the comparison of multiple distributions across parameters, timepoints, or treatment arms. Within clinical trials, these plots provide a powerful means of uncovering trends in laboratory data and other repeated measures, revealing shifts and differences that may be hidden in traditional tables.

This paper demonstrates how ridgeline plots can be applied to ADaM laboratory (ADLB) data using R and the ggridges package, and discusses practical challenges such as handling skewed values, ensuring interpretability, and maintaining regulatory compliance. The implementation is demonstrated on a simulated ADLB dataset with laboratory categories including Chemistry, Hematology, and Urinalysis panels. The results highlight the ability of ridgeline plots to provide clear, engaging insights that support both exploratory analysis and communication with non-technical stakeholders, while noting that careful design and annotation are necessary to avoid misinterpretation.

INTRODUCTION

Data visualization plays a central role in clinical trial reporting. For decades, listings, summary tables, and boxplots have been the standard for communicating laboratory data and treatment effects. While these formats are robust, they may not always provide the clearest view of distributional patterns, especially when comparing multiple parameters or visits simultaneously.

Laboratory datasets (ADLB) in clinical trials are particularly suited to this visualization style. With hundreds of subjects, multiple timepoints, and diverse lab panels, tabular summaries can obscure meaningful insights. By layering distributions for ALT, AST, Hemoglobin, and other labs, ridgeline plots can expose subtle but important changes in safety signals or treatment responses.

This paper introduces ridgeline plots in the context of clinical trial laboratory reporting, explores their implementation using R, and discusses challenges and solutions for adoption in regulatory settings.

METHODS

DATA STRUCTURE

The input dataset is an ADLB-style dataset containing required variables such as:

- USUBJID – Subject Identifier
- PARAM / PARAMCD – Laboratory test and code
- AVAL – Analyzed value
- AVISIT – Visit label
- TRTP – Treatment group
- CATEGORY – Lab panel (Chemistry, Hematology, Urinalysis)

Simulated data were generated to represent common laboratory tests across categories, visits, and treatment groups.

RIDGELINE PLOT CONCEPT

A ridgeline plot is composed of multiple density plots stacked along a secondary axis. The X-axis represents the continuous measurement (e.g., ALT values), while the Y-axis represents parameters or visits. The overlapping “waves” allow rapid visual comparison of distributions.

IMPLEMENTATION IN R

Ridgeline plots were created using the ggridges package within ggplot2.

```
library(ggplot2)

library(ggridges)

ggplot(adlb, aes(x = AVAL, y = PARAM, fill = TRTP))

  + geom_density_ridges(alpha = 0.7, scale = 1.2) +

  labs(title = "Ridgeline Plot of Laboratory

  Parameters",

       x = "Value", y = "Parameter") +

  theme_minimal()
```

This produces layered distributions, with each laboratory parameter plotted as a ridge. Treatment groups are distinguished by color.

SHINY APPLICATION (EXPLORATORY USE)

For exploratory purposes, an interactive Shiny application was built to allow selection of categories, parameters, visits, and treatment arms. Screenshots of the application outputs are included to illustrate flexibility.

```
sidebarPanel( uiOutput("category_ui"),

  uiOutput("param_ui"), uiOutput("visit_ui"),

  uiOutput("trtp_ui"),

checkboxInput("add_baseline", "Overlay Baseline

  values", FALSE), checkboxInput("add_mean", "Show

  mean lines", FALSE),

checkboxInput("add_median", "Show median lines",

  FALSE)

)
```

This snippet illustrates how users can dynamically filter categories, parameters, visits, and treatment arms, while optionally overlaying baseline, mean, or median markers.

```
output$ridgeplot <- renderPlotly({

  ggplotly(plot_obj(), tooltip = c("x","y","fill"))

})
```

The core plotting engine uses ggplot2 with geom_density_ridges to generate ridgeline plots for the selected laboratory parameters.

```
output$ridgeplot <- renderPlotly({
    ggplotly(plot_obj(), tooltip = c("x","y","fill"))
})
```

Rendered plots are interactive (via Plotly) and supported by a data table view, enabling both graphical and tabular exploration.

USAGE IN PRACTICE

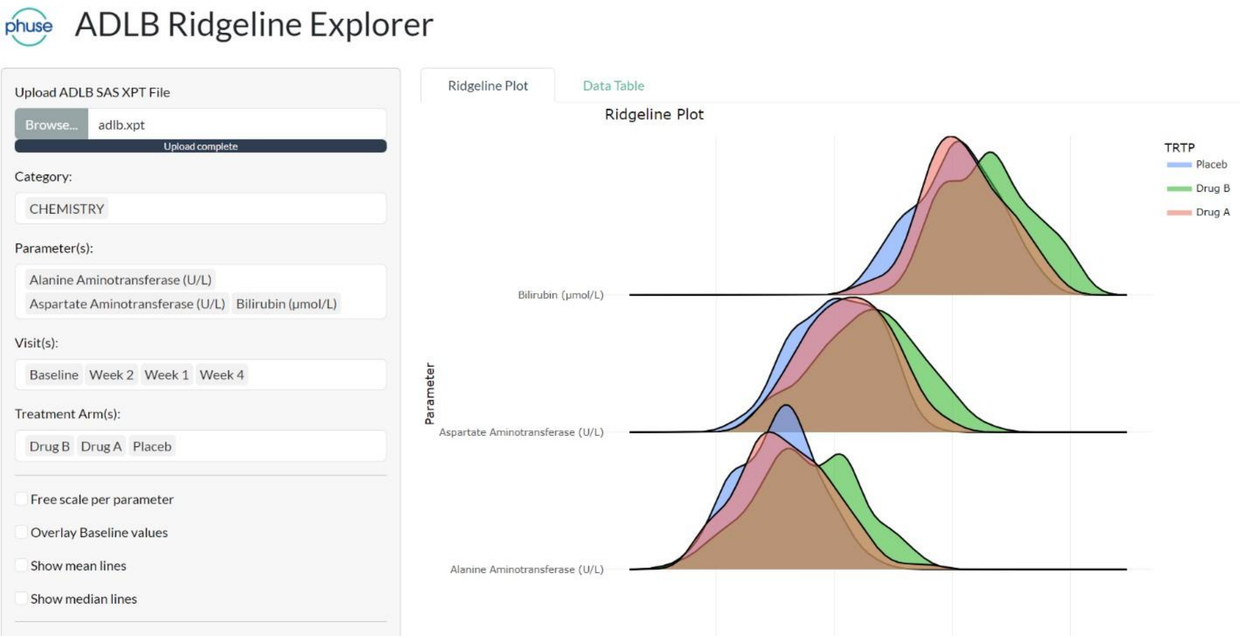
Ridgeline plots were incorporated into internal clinical reporting workflows to support exploratory analysis and team discussions. Key practical uses include:

- Rapid identification of distribution shifts across treatment arms and visits.
- Comparison of multiple lab parameters simultaneously to detect safety signals.
- Communication with non-statistical stakeholders, as layered visuals provide more intuitive insights than tables.
- Integration into R Shiny applications for interactive data exploration, enabling analysts to filter by visit, lab category, or treatment arm.
- Financially advantageous compared with generating separate plots for each category, as multiple parameters can be reviewed in a single visualization, reducing programming time and report production costs.

RESULTS

EXAMPLE 1: CHEMISTRY PARAMETERS

- ALT values in Drug A subjects are trending higher at Week 4.
- Overlap between Placebo and Drug B for AST, suggesting no treatment effect.
- Distribution widening for ALP, consistent with greater variability in Drug A.



Upload ADLB SAS XPT File: adlb.xpt

Category: CHEMISTRY

Parameter(s): Alanine Aminotransferase (U/L) Aspartate Aminotransferase (U/L) Bilirubin (µmol/L)

Visit(s): Baseline Week 2 Week 1 Week 4

Treatment Arm(s): Drug B Drug A Placeb

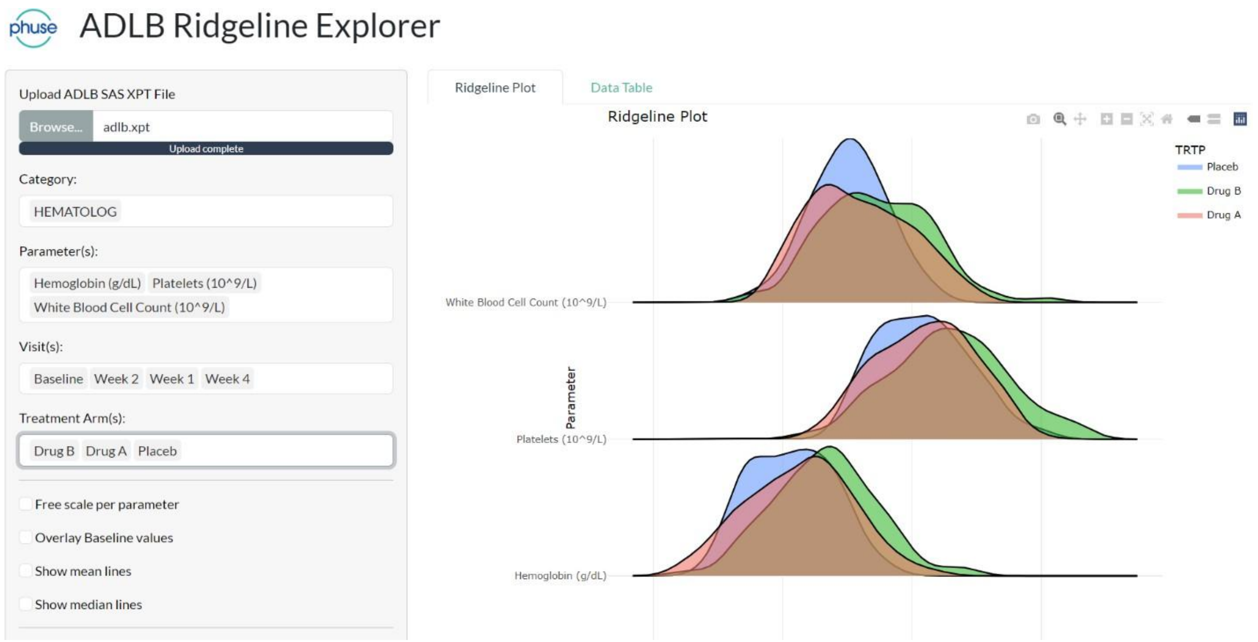
☐ Free scale per parameter

☐ Overlay Baseline values

☐ Show mean lines

☐ Show median lines

- Hemoglobin distributions remain stable across groups.
- Platelet counts show a downward shift for Drug B at Week 2.
- White Blood Cells demonstrate higher variance in the Placebo, possibly linked to baseline imbalances.



ADLB Ridgeline Explorer

Upload ADLB SAS XPT File: adlb.xpt

Upload complete

Category: HEMATOLOG

Parameter(s): Hemoglobin (g/dL) Platelets (10^9/L) White Blood Cell Count (10^9/L)

Visit(s): Baseline Week 2 Week 1 Week 4

Treatment Arm(s): Drug B Drug A Placeb

☐ Free scale per parameter

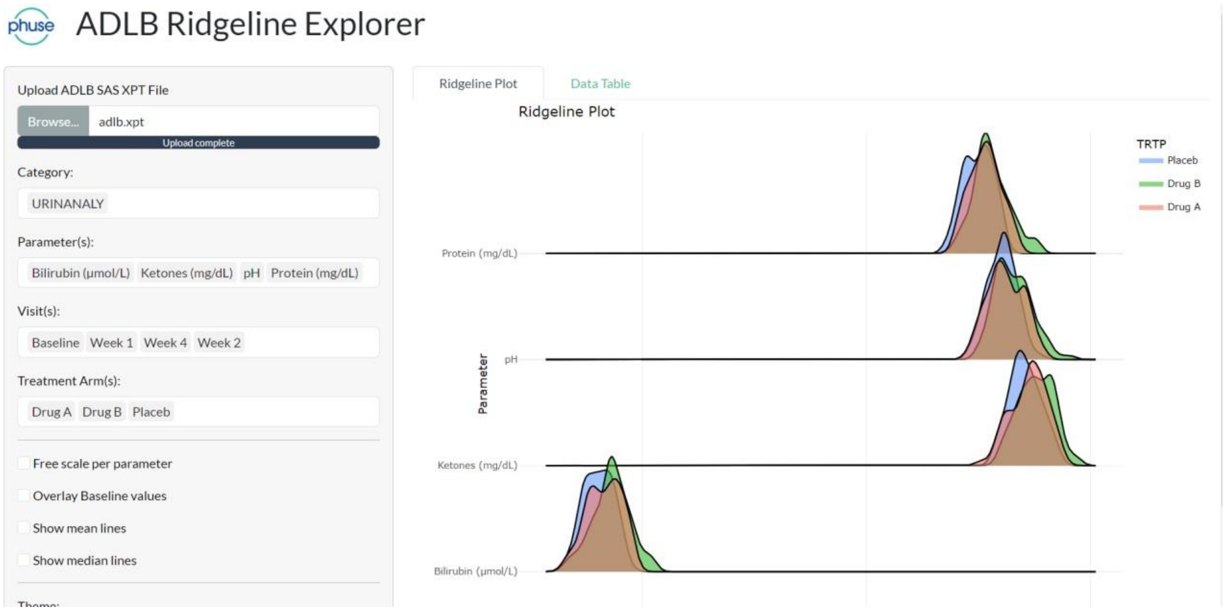
☐ Overlay Baseline values

☐ Show mean lines

☐ Show median lines

EXAMPLE 3: URINALYSIS PARAMETERS

- Bilirubin values are generally consistent across visits and treatment arms, with minimal variability.
- Ketones show a small peak shift for Drug B compared to Drug A and Placebo.
- pH distributions overlap across arms, suggesting no major treatment effect.
- Protein levels appear stable but with slightly broader variability in the Drug B group at later visits.



ADLB Ridgeline Explorer

Upload ADLB SAS XPT File: adlb.xpt

Upload complete

Category: URINANALY

Parameter(s): Bilirubin (µmol/L) Ketones (mg/dL) pH Protein (mg/dL)

Visit(s): Baseline Week 1 Week 4 Week 2

Treatment Arm(s): Drug A Drug B Placeb

☐ Free scale per parameter

☐ Overlay Baseline values

☐ Show mean lines

☐ Show median lines

GAPS AND LIMITATIONS

While ridgeline plots are powerful, several limitations exist in clinical trial applications:

1. **Data Quality** – Outliers, missing values, or skewed distributions may distort density estimates. Proper preprocessing is critical.
2. **Interpretation** – non-statistical reviewers may misinterpret density height or overlap as indicating sample size differences. Annotations and legends are essential.
3. **Regulatory Acceptance** – Ridgeline plots are currently exploratory; standard regulatory submissions rely on tables and boxplots. Plots may need justification in review documentation.
4. **Complexity** – High numbers of parameters or overlapping distributions can reduce readability. Careful selection of panels and visits is required.
5. **Tool Limitations** – While R allows flexible visualizations, similar outputs in SAS require advanced coding and may lack interactivity.

RECOMMENDATIONS FOR INDUSTRY PRACTICE

Based on our experience, the following best practices are recommended for implementing ridgeline plots in clinical programming workflows:

- **Prioritize clarity and interpretability:** Ensure visualizations are easy to read for non-statistical reviewers.
- **Preprocess data rigorously:** Handle outliers, missing values, and skewed distributions before plotting.
- **Use as a complementary tool:** Combine ridgeline plots with standard tables and boxplots for regulatory submissions.
- **Document assumptions and choices:** Record preprocessing steps, plotting parameters, and interpretation guidance for transparency.
- **Leverage interactive tools:** R Shiny or similar platforms can allow stakeholders to explore different parameters, visits, or treatment groups dynamically.
- **Plan for scalability:** Optimize plots for large datasets to maintain readability and computational efficiency.
- **Prepare for regulatory discussion:** Include guidance on interpretation and limitations when presenting ridgeline plots to regulatory reviewers.

EXPECTATION FOR THE FUTURE

Looking ahead, ridgeline plots have the potential to become a more standardized tool in clinical data visualization:

- **Integration into submission-ready workflows:** With further validation and alignment with regulatory expectations, ridgeline plots may complement traditional tables and boxplots in regulatory submissions.
- **Extension to other endpoints:** Beyond laboratory data, ridgeline plots could be applied to adverse event severities, patient-reported outcomes, pharmacokinetic measures, and efficacy endpoints.
- **Enhanced interactivity:** The Development of more robust R Shiny applications or web-based dashboards can allow stakeholders to explore complex datasets in real time.
- **Standardization and guidelines:** Future regulatory guidance may include formal recommendations for the use of ridgeline plots, ensuring consistent interpretation and acceptance.
- **Advanced statistical features:** Incorporating annotations for significance, confidence intervals, or baseline shifts could make ridgeline plots more informative and statistically rigorous.

By anticipating these developments, clinical programmers can prepare to adopt ridgeline plots in a compliant, practical, and forward-looking manner.

CHALLENGES AND CONSIDERATIONS

1. **Regulatory Standards:** Traditional plots (e.g., boxplots) remain the standard in submissions. Ridgeline plots are best suited for exploratory use or internal decision-making.
2. **Aesthetic vs. Readability:** Overuse of colors or overlapping ridges can reduce clarity. Balancing design with interpretability is critical.
3. **Tooling:** While SAS can generate simplified ridgeline-style plots, advanced features are more easily implemented in R

CONCLUSION

Ridgeline plots offer a novel way to visualize laboratory data, emphasizing distributional changes across visits and treatments. By layering density curves, these plots highlight subtle shifts and patterns that traditional summary tables may miss.

Although challenges exist particularly around regulatory acceptance and data quality ridgeline plots provide a valuable addition to the clinical programmer’s toolkit. With further refinement and standardization, they could extend beyond laboratory data into domains such as adverse event severities, patient-reported outcomes, or pharmacokinetic measures.

ACKNOWLEDGMENTS

The authors thank the PHUSE community for providing a platform to share innovative approaches and foster collaboration across the industry.

CONTACT INFORMATION

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

Author Name: Mariam Babayan
Company: BioBrain Pro
Address: 2/2 A Mikoyan Street, 0054, Yerevan, Armenia
Email: marbabayan@biobrainpro.net