

Advanced Clinical Data Visualisation Using Violin Plots in R

A Step-by-Step Approach

Author: Mrityunjay Kumar

Co-Author: Shashikant Kumar

Presentation No. – ET11

Stream: Emerging Trends (ET)

PHUSE US Connect 2025, Orlando, Florida 16–19 March

Agenda

1. Introduction to Violin Plot
2. Understanding through Examples
3. Data Structure for Violin Plot
4. How to Develop Violin Plot in R Studio
5. A Comparison with Box Plot, Histogram and Density Plot
6. Usage in Oncology
7. Limitations of Violin Plot
8. Conclusion

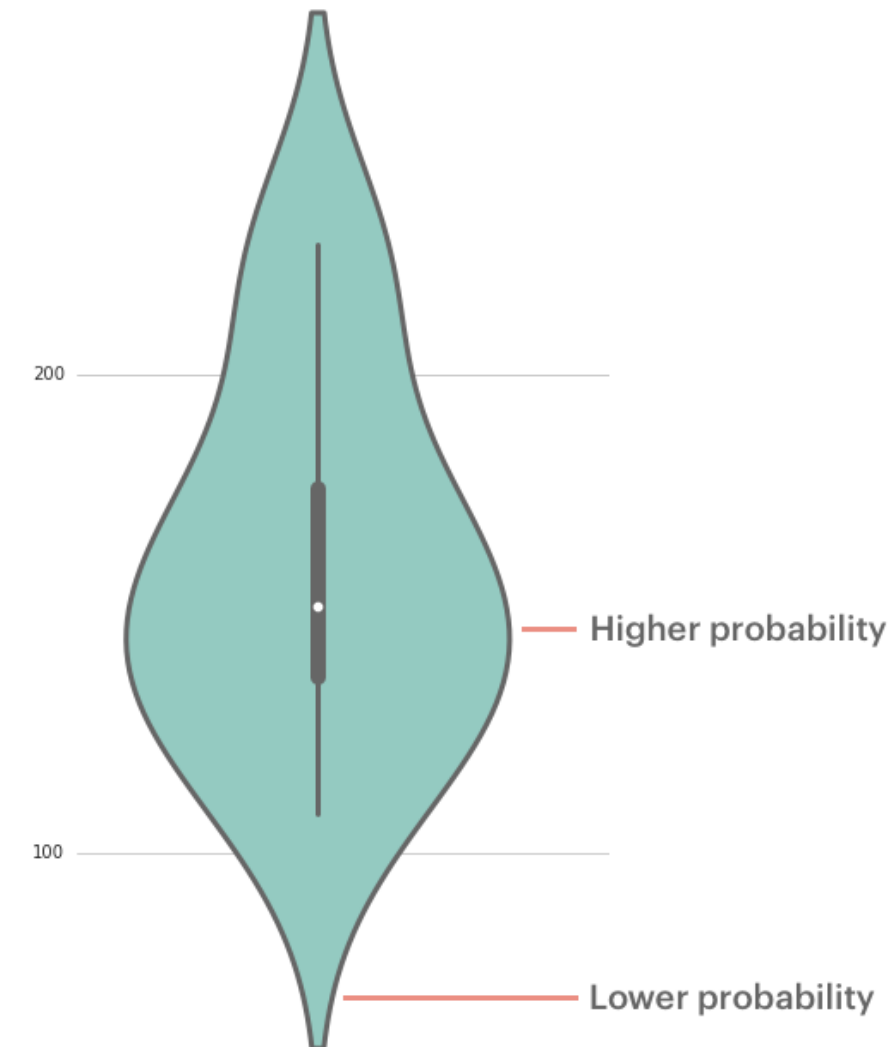
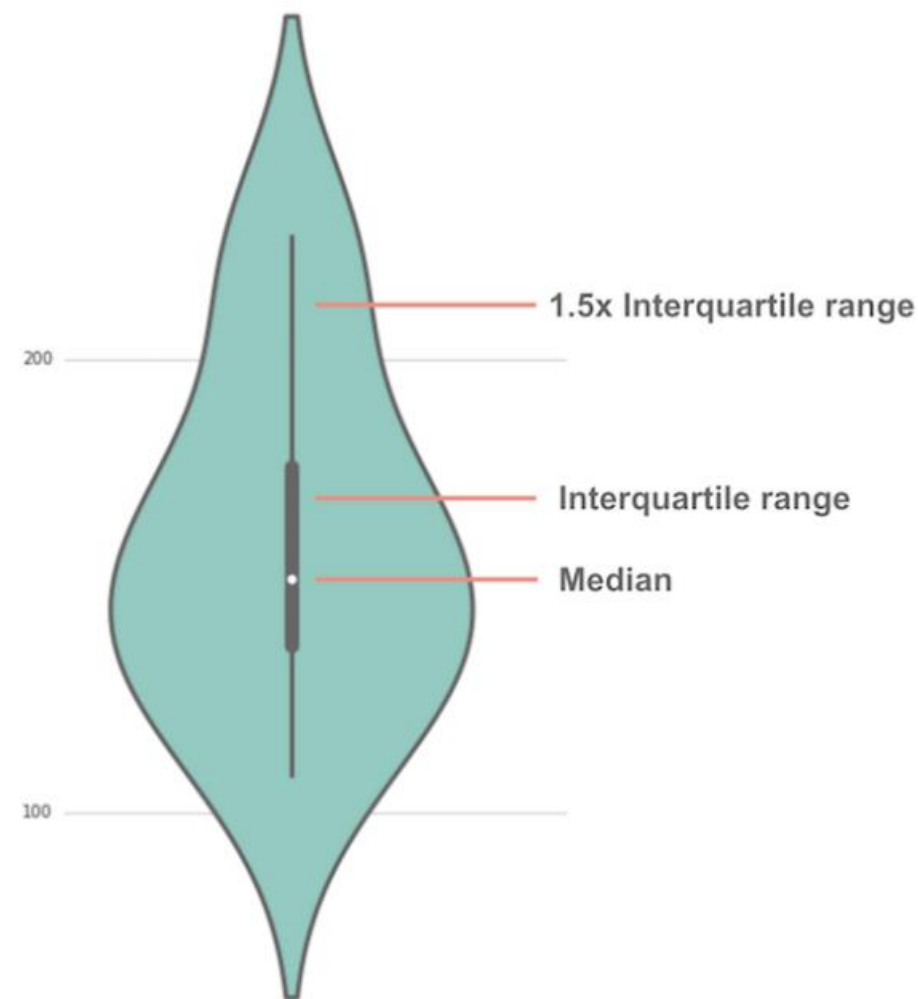


Introduction – Violin Plot

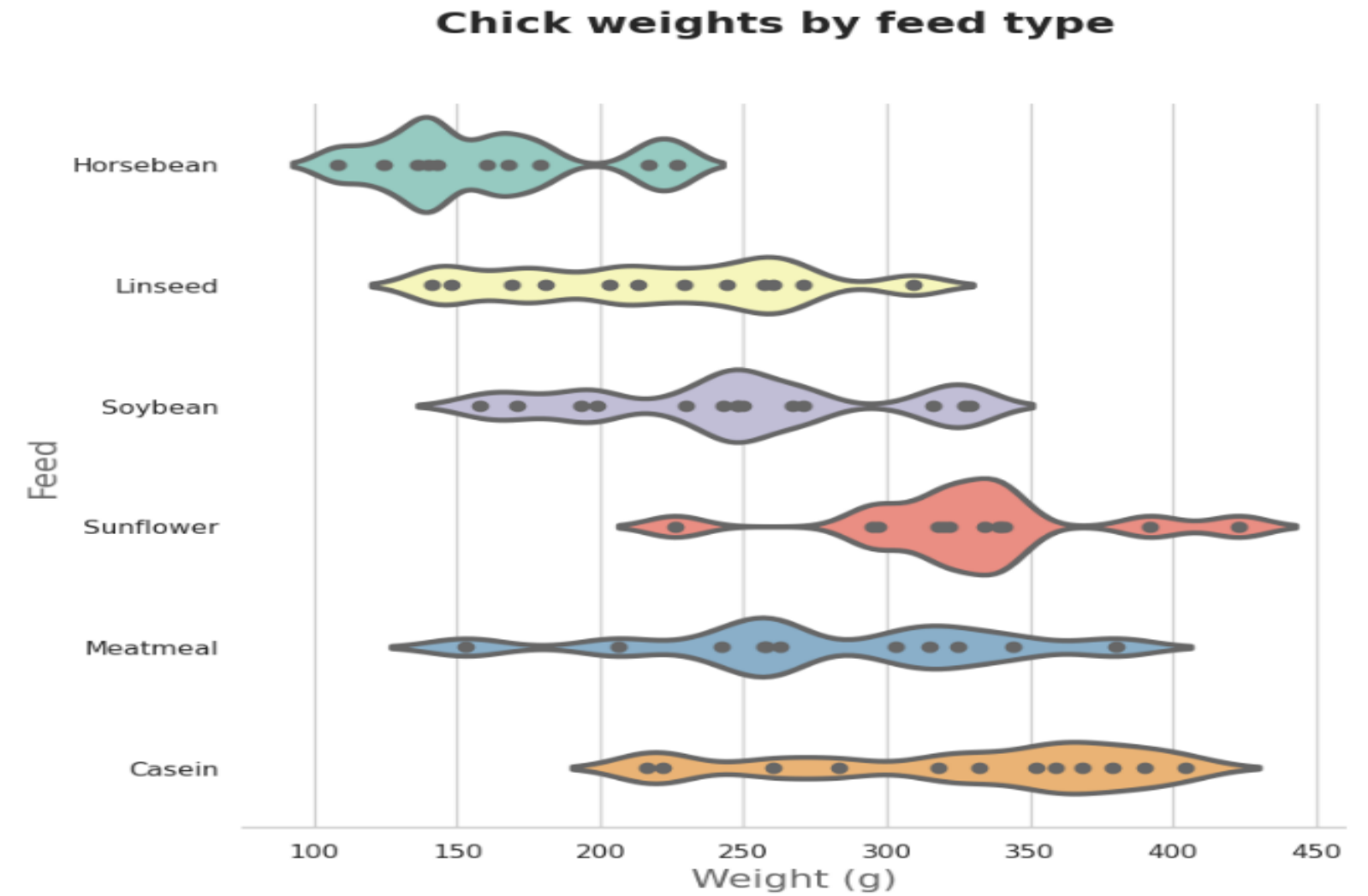
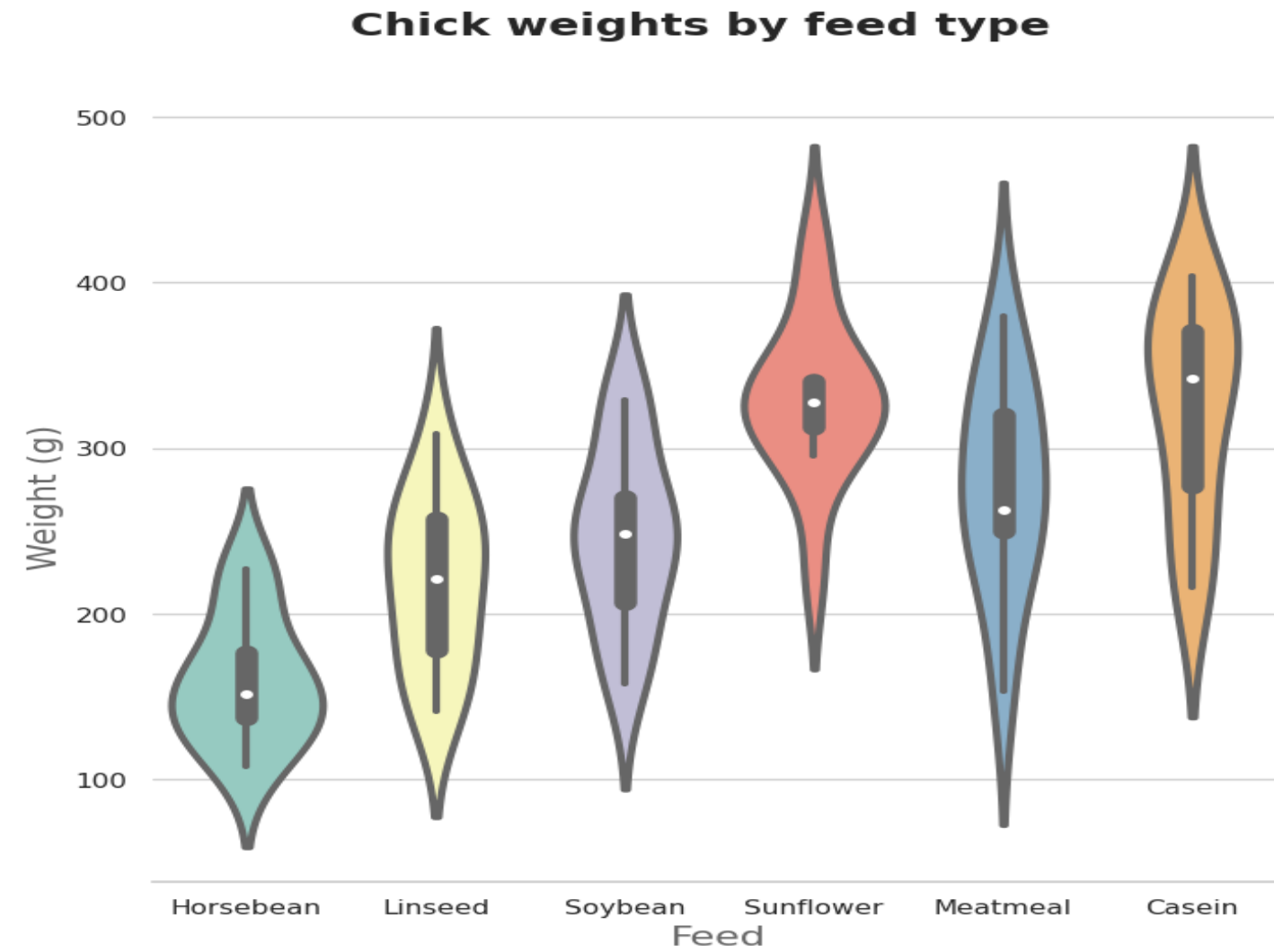
- Data Visualisation technique
- A Combination of box plot and density plot
- Shows peaks in the data
- Distributions of numeric data for one or more groups using density curves
- Displays summary statistics and the density of each variable

Why Violin Plot ?

- To observe the distribution of numeric data
- To make a comparison of distributions between multiple groups
- The peaks, valleys, and tails of each group's density curve can be compared to see where groups are similar or different

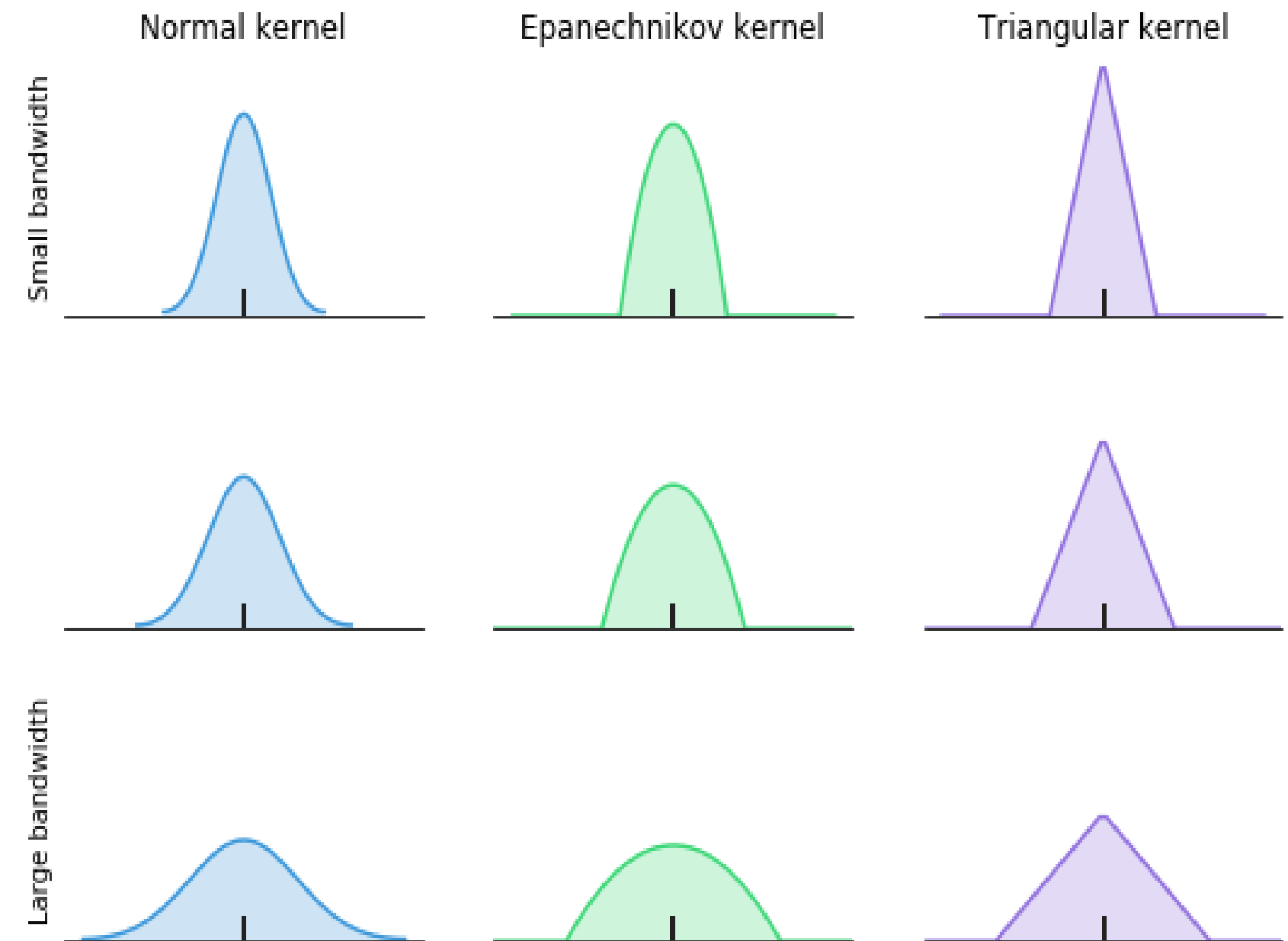


2. Understanding through Examples



Kernel Density Estimate (KDE)

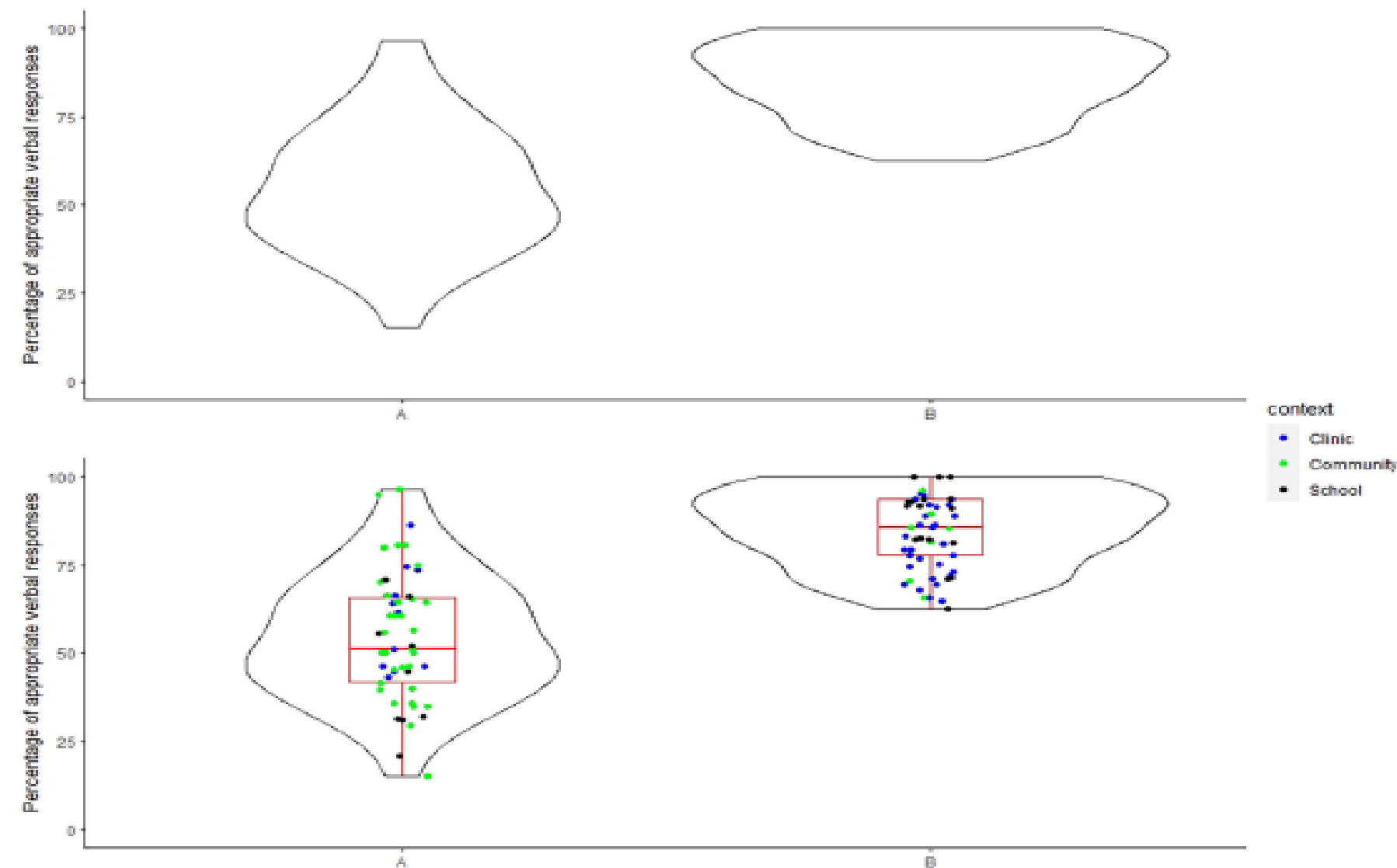
- KDE, is a less-frequently encountered depiction of data distribution, compared to the more common histogram
- Each data point contributes a small area around its true value
- The shape of this area is called the **kernel function**.
- Kernels can take different shapes from smooth bell curves to sharp triangular peaks
- Kernels can have different width, or bandwidth, affecting the influence of each individual data point



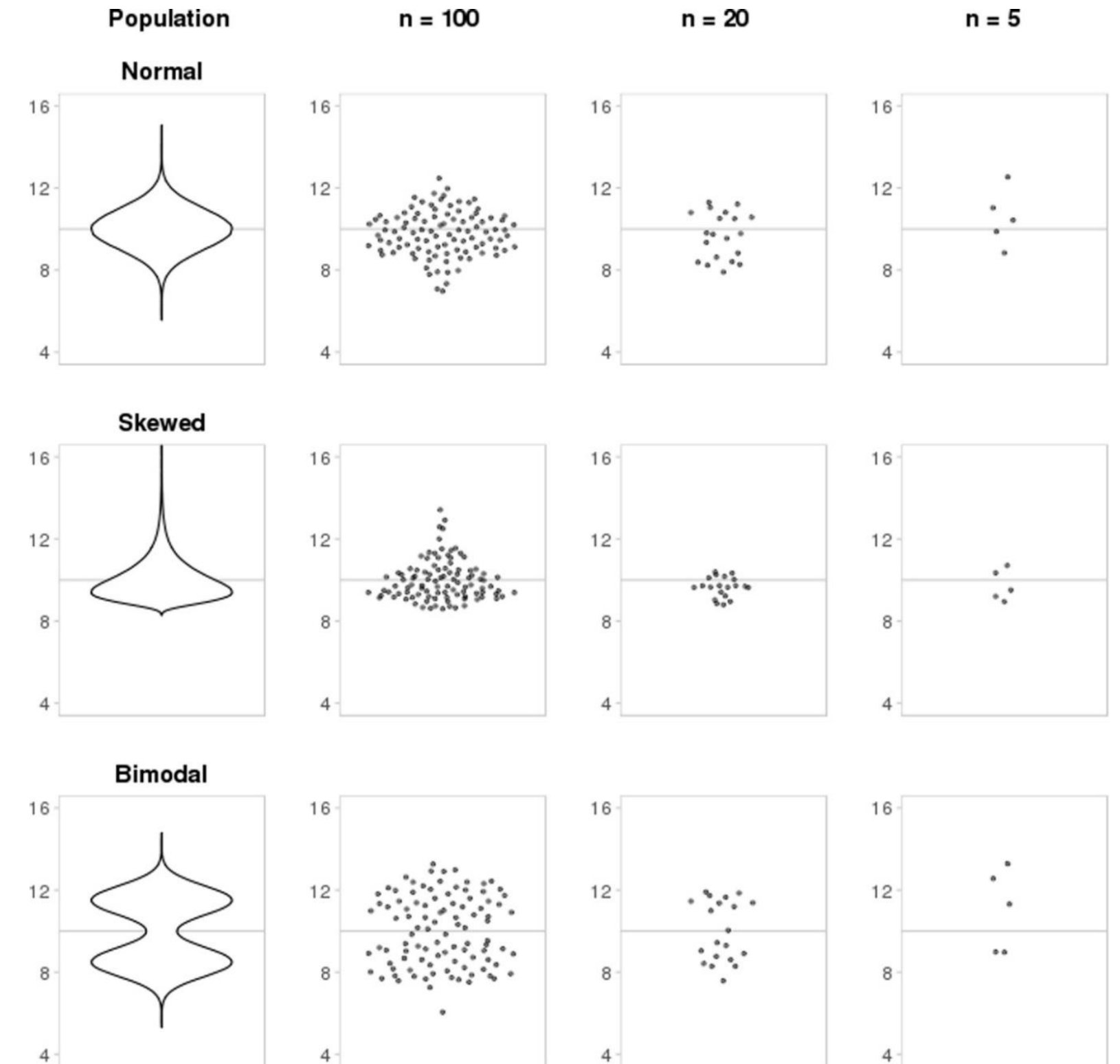
Understanding data distribution

Figure 1

Violin Plots for the Koegel et al. (1992) Data for Participant Tony



Note. The upper panel shows the basic violin plots. The lower panel shows the same violin plots with the addition of a boxplot and jitter of the raw scores. Blue indicates clinical setting, green indicates community setting, and black indicates school setting.



3. Data Structure for Violin Plot

- **CR (Complete Response):** 100% shrinkage
- **PR (Partial Response):** More than 30% shrinkage
- **SD (Stable Disease):** Between -30% and +20%
- **PD (Progressive Disease):** Growth beyond 20%

Subject ID	Treatment	Biomarker Level	Tumor Response (mm)	RECIST Category
10001	Drug A	23.5	-15	PR (Partial Response)
10002	Placebo	18.2	+10	PD (Progressive Disease)
10003	Drug A	27.8	-25	PR (Partial Response)
10004	Drug A	19.4	-5	SD (Stable Disease)
10005	Placebo	16.5	+12	PD (Progressive Disease)
10006	Drug A	30.1	-30	CR (Complete Response)
10007	Placebo	21.2	0	SD (Stable Disease)
10008	Drug A	25.3	-20	PR (Partial Response)
10009	Placebo	17.8	+8	PD (Progressive Disease)
10010	Drug A	29.5	-35	CR (Complete Response)

4. How to Develop Violin Plot in R Studio

- Load packages with `library()` function
- Packages includes `ggplot2`, `dplyr`, `gridRextra`

```
# Load library
library(ggplot2)
library(dplyr)
library(gridExtra)

# Set seed for reproducibility
set.seed(123)

# Create Tumor response & Biomarker dataset
oncology_data <- data.frame(
  Subject_ID = sprintf("%05d", 1:100),
  Treatment = rep(c("Drug A", "Placebo"), each = 50),
  Tumor_Response = c(rnorm(50, mean = -65, sd = 20),
                    rnorm(50, mean = -20, sd = 25)),
  Biomarker_Level = c(rnorm(50, mean = 15, sd = 5),
                    rnorm(50, mean = 20, sd = 6))
)

# Assign RECIST categories based on tumor response percentage
oncology_data <- oncology_data %>%
  mutate(RECIST = case_when(
    Tumor_Response <= -80 ~ "CR", # Complete Response
    Tumor_Response <= -30 ~ "PR", # Partial Response
    Tumor_Response > -30 & Tumor_Response < 20 ~ "SD", # Stable Disease
    Tumor_Response >= 20 ~ "PD" # Progressive Disease
  ))

# Display first 15 rows
head(oncology_data, 15)
```

```
> # Display first 15 rows
> head(oncology_data, 15)
```

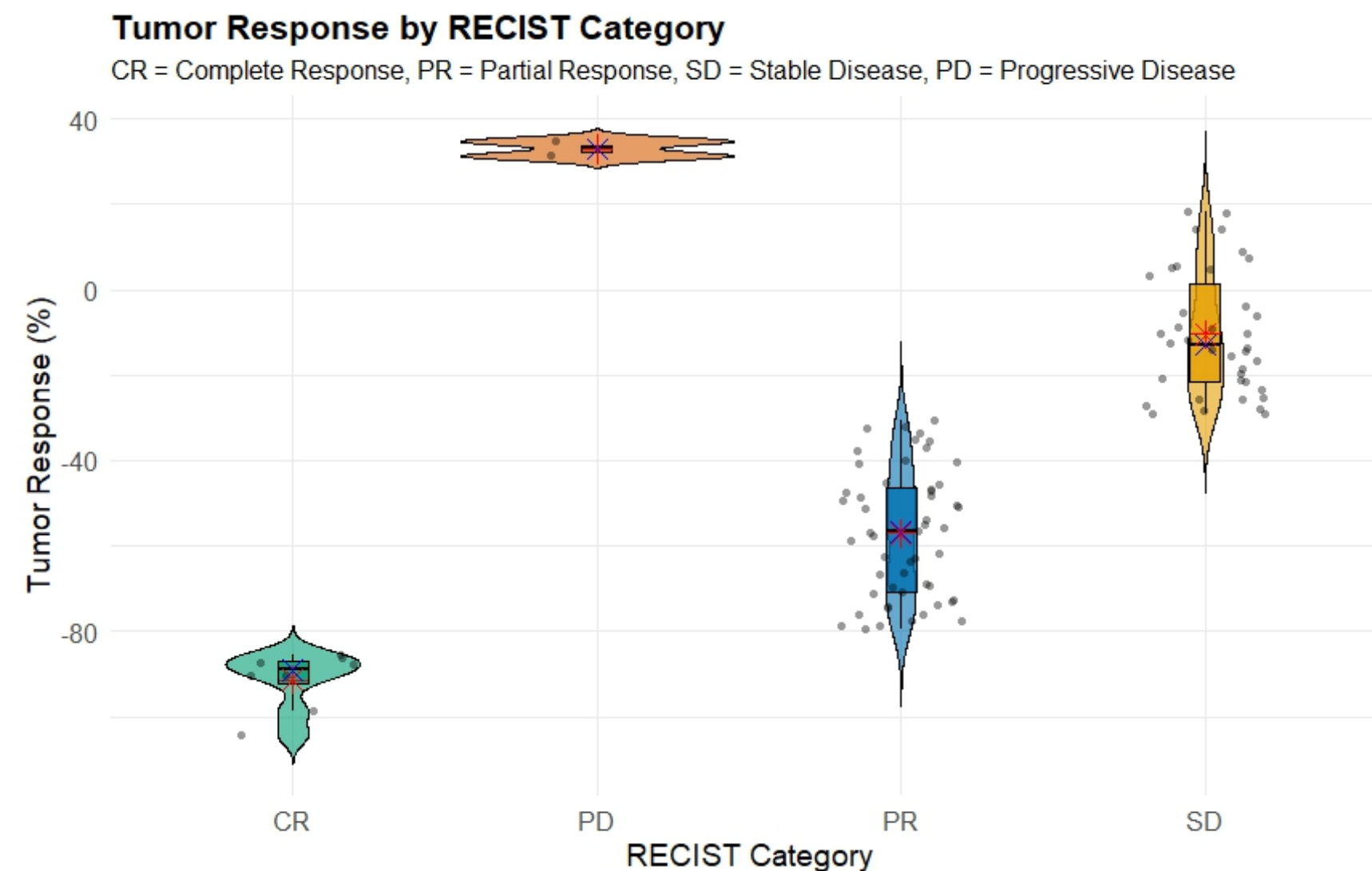
	Subject_ID	Treatment	Tumor_Response	Biomarker_Level	RECIST
1	00001	Drug A	-76.20951	11.447967	PR
2	00002	Drug A	-69.60355	16.284419	PR
3	00003	Drug A	-33.82583	13.766541	PR
4	00004	Drug A	-63.58983	13.262287	PR
5	00005	Drug A	-62.41425	10.241907	PR
6	00006	Drug A	-30.69870	14.774861	PR
7	00007	Drug A	-55.78168	11.075478	PR
8	00008	Drug A	-90.30122	6.660290	CR
9	00009	Drug A	-78.73706	13.098867	PR
10	00010	Drug A	-73.91324	19.594983	PR
11	00011	Drug A	-40.51836	12.123265	PR
12	00012	Drug A	-57.80372	18.039822	PR
13	00013	Drug A	-56.98457	6.910586	PR
14	00014	Drug A	-62.78635	14.722190	PR
15	00015	Drug A	-76.11682	17.597036	PR

Example 1: Violin Plot

Tumor Response (%) by RECIST Category

```
# Custom color scheme for RECIST
recist_colors <- c("CR" = "#009E73", "PR" = "#0072B2", "SD" = "#E69F00", "PD" = "#D55E00")

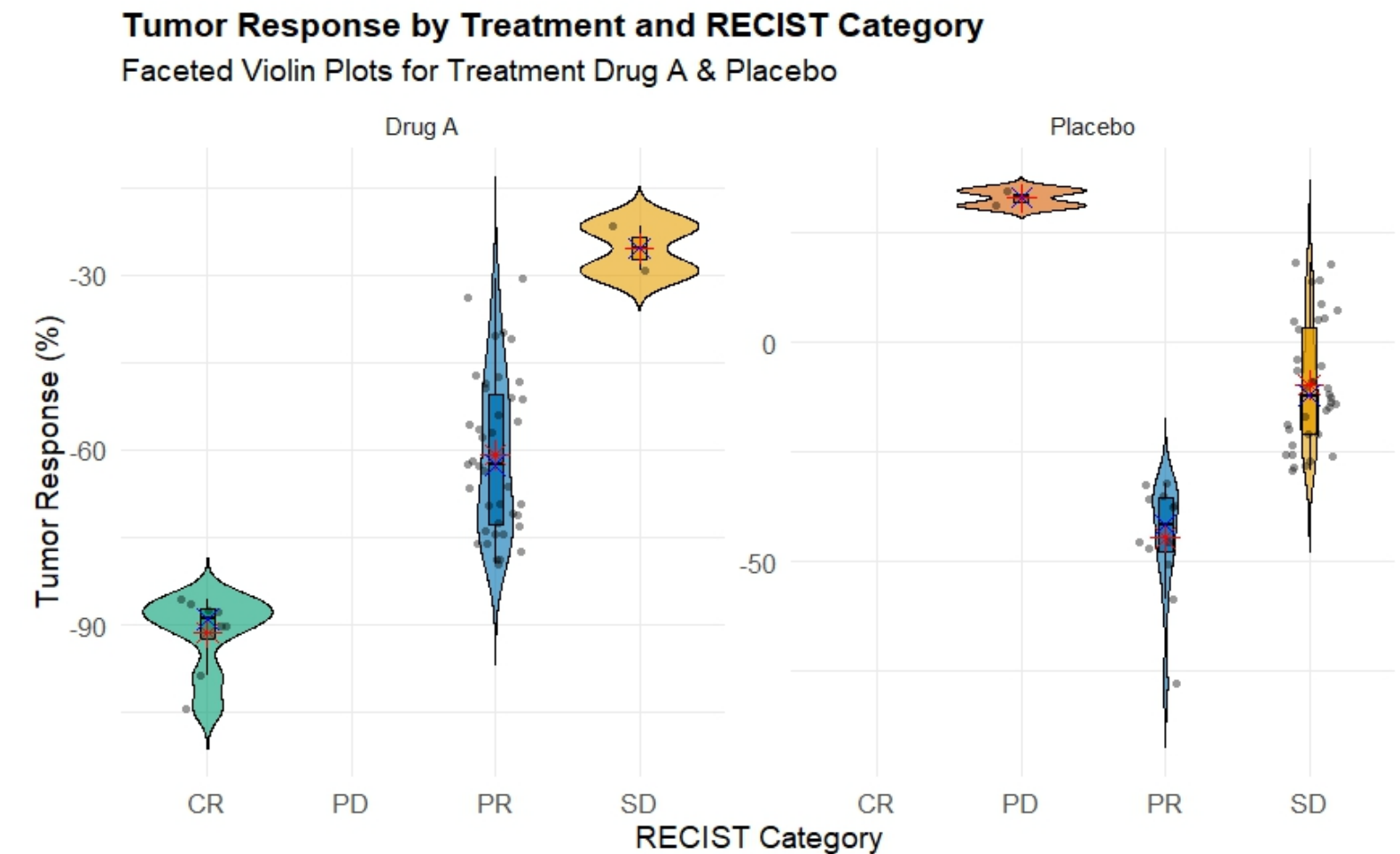
# Violin plot by RECIST category
ggplot(oncology_data, aes(x = RECIST, y = Tumor_Response, fill = RECIST)) +
  geom_violin(trim = FALSE, alpha = 0.6, color = "black") +
  geom_boxplot(width = 0.1, color = "black", alpha = 0.8, outlier.shape = NA) +
  geom_jitter(shape = 16, position = position_jitter(0.2), alpha = 0.4, color = "black") +
  stat_summary(fun = "mean", geom = "point", shape = 8, size = 4, color = "red") +
  stat_summary(fun = "median", geom = "point", shape = 4, size = 4, color = "blue") +
  scale_fill_manual(values = recist_colors) +
  labs(title = "Tumor Response by RECIST Category",
       subtitle = "CR = Complete Response, PR = Partial Response, SD = Stable Disease, PD = Progressive Disease",
       x = "RECIST Category",
       y = "Tumor Response (%)") +
  theme_minimal(base_size = 12) +
  theme(legend.position = "none",
       plot.title = element_text(face = "bold", size = 16),
       axis.title = element_text(size = 14),
       axis.text = element_text(size = 12))
```



Example 2: Faceted Violin Plot

Tumor Response by Treatment & RECIST

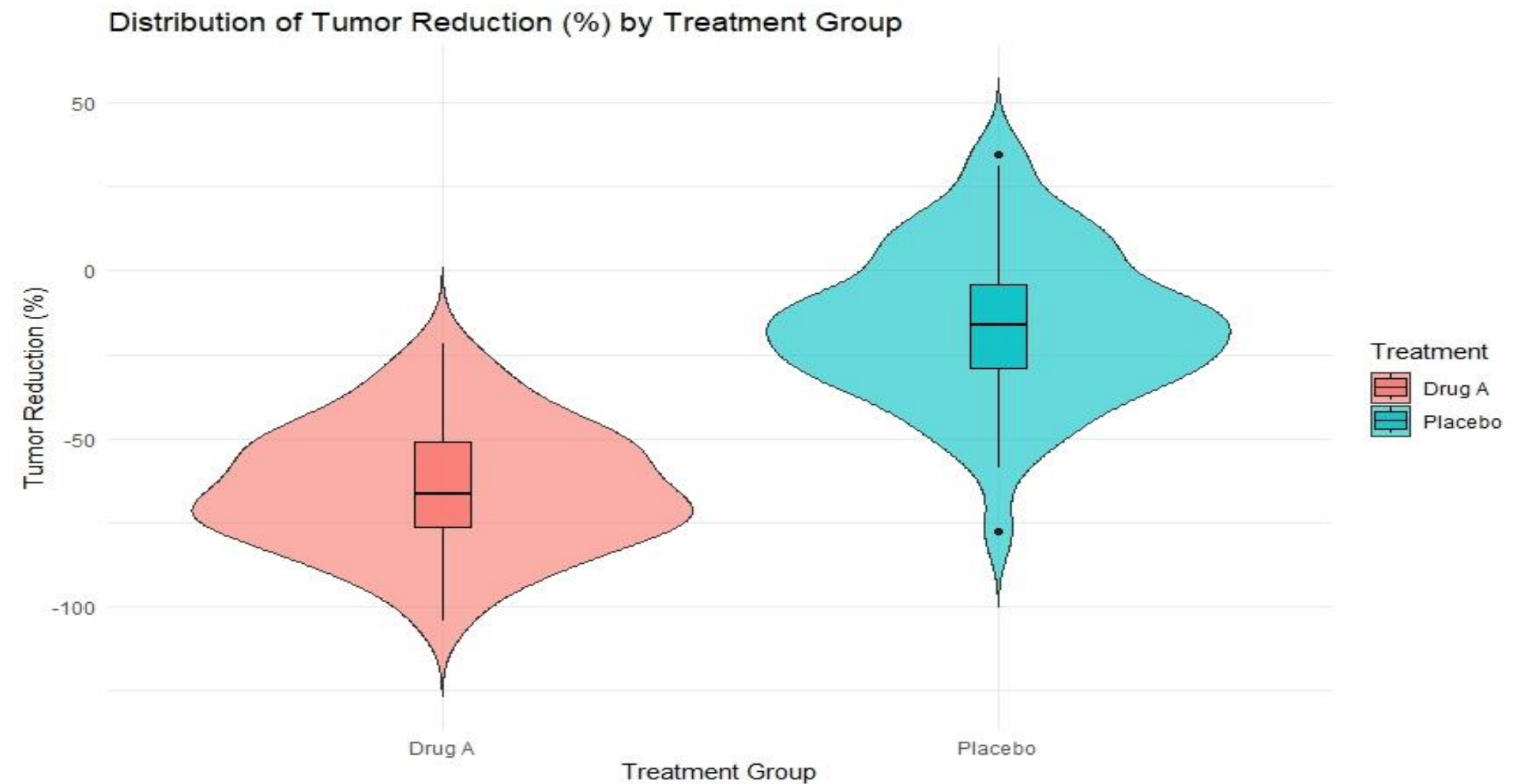
```
# Violin Plot with facet wrap for Treatment and RECIST category
ggplot(oncology_data, aes(x = RECIST, y = Tumor_Response, fill = RECIST)) +
  geom_violin(trim = FALSE, alpha = 0.6, color = "black") +
  geom_boxplot(width = 0.1, color = "black", alpha = 0.8, outlier.shape = NA) +
  geom_jitter(shape = 16, position = position_jitter(0.2), alpha = 0.4, color = "black") +
  stat_summary(fun = "mean", geom = "point", shape = 8, size = 4, color = "red") +
  stat_summary(fun = "median", geom = "point", shape = 4, size = 4, color = "blue") +
  scale_fill_manual(values = recist_colors) +
  labs(title = "Tumor Response by Treatment and RECIST Category",
       subtitle = "Faceted Violin Plots for Treatment Drug A & Placebo",
       x = "RECIST Category",
       y = "Tumor Response (%)") +
  facet_wrap(~Treatment, scales = "free_y") + # Facet by Treatment
  theme_minimal(base_size = 14) +
  theme(legend.position = "none",
       plot.title = element_text(face = "bold", size = 16),
       axis.title = element_text(size = 14),
       axis.text = element_text(size = 12))
```



Example 3: Violin Plot

Tumor Reduction by Treatment Group

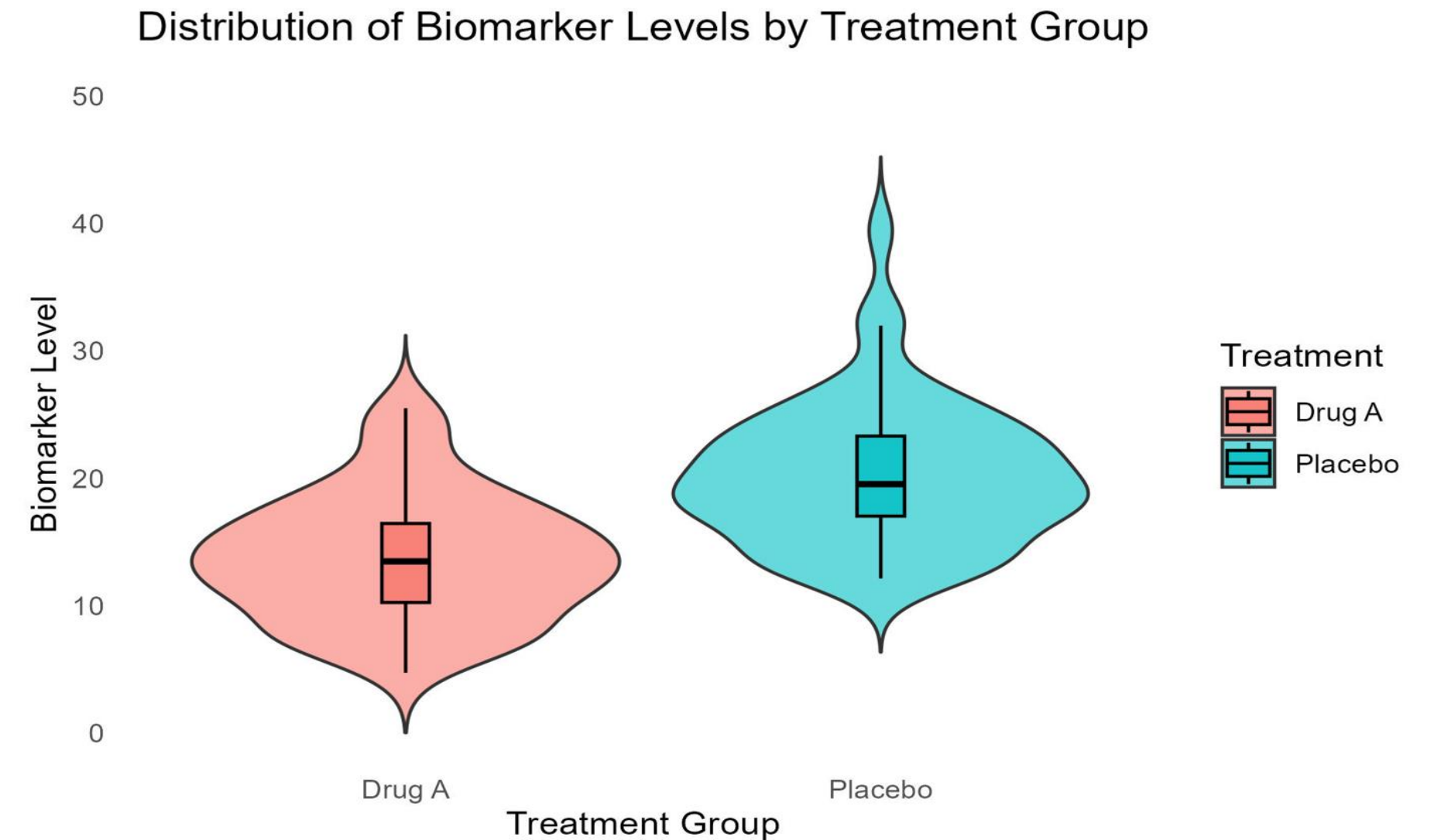
```
# Violin Plot for Tumor Reduction
ggplot(oncology_data, aes(x = Treatment, y = Tumor_Response, fill = Treatment)) +
  geom_violin(trim = FALSE, alpha = 0.6) +
  geom_boxplot(width = 0.1, color = "black", alpha = 0.8) +
  labs(title = "Distribution of Tumor Reduction (%) by Treatment Group",
       x = "Treatment Group",
       y = "Tumor Reduction (%)") +
  theme_minimal()
```



Example 4: Violin Plot

Biomarker Levels by Treatment Group

```
# Violin Plot for Biomarker Levels
ggplot(oncology_data, aes(x = Treatment, y = Biomarker_Level, fill = Treatment)) +
  geom_violin(trim = FALSE, alpha = 0.6) +
  geom_boxplot(width = 0.1, color = "black", alpha = 0.8) +
  labs(title = "Distribution of Biomarker Levels by Treatment Group",
       x = "Treatment Group",
       y = "Biomarker Level") +
  ylim(0, 50) +
  theme_minimal() +
  theme(plot.title = element_text(hjust = 0.5)) # Center-align title
```



5. A Comparison with Box Plot, Histogram and Density Plot

```
# Violin Plot
violin_plot <- ggplot(oncology_data, aes(x = Treatment, y = Biomarker_Level, fill = Treatment)) +
  geom_violin(trim = FALSE, alpha = 0.6) +
  geom_boxplot(width = 0.1, color = "black", alpha = 0.8) +
  labs(title = "Violin Plot: Biomarker Levels by Treatment", x = "Treatment", y = "Biomarker Level") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  theme(plot.title = element_text(hjust = 0.5))

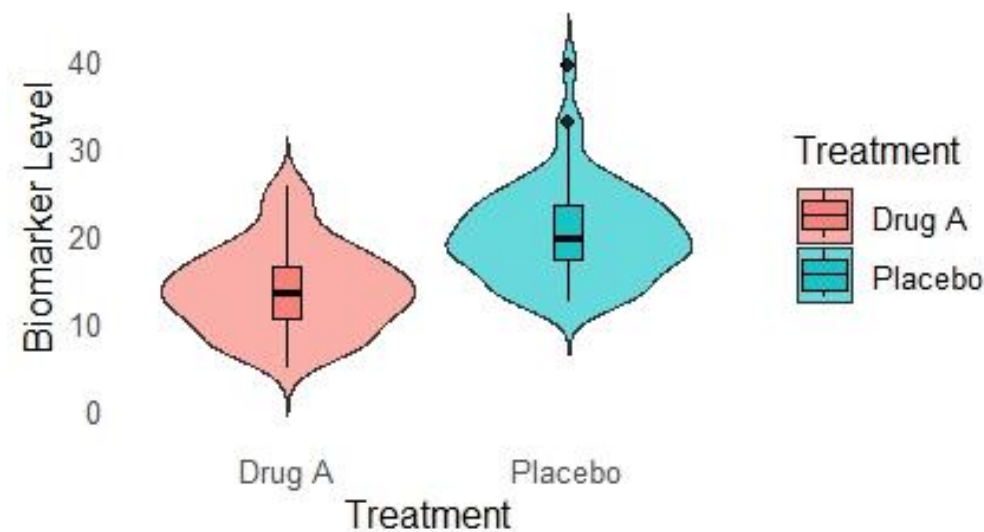
# Box Plot
box_plot <- ggplot(oncology_data, aes(x = Treatment, y = Biomarker_Level, fill = Treatment)) +
  geom_boxplot() +
  labs(title = "Box Plot: Biomarker Levels by Treatment", x = "Treatment", y = "Biomarker Level") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  theme(plot.title = element_text(hjust = 0.5))

# Histogram
histogram_plot <- ggplot(oncology_data, aes(x = Biomarker_Level, fill = Treatment)) +
  geom_histogram(alpha = 0.6, position = "identity", bins = 15) +
  labs(title = "Histogram: Biomarker Levels", x = "Biomarker Level", y = "Count") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  theme(plot.title = element_text(hjust = 0.5))

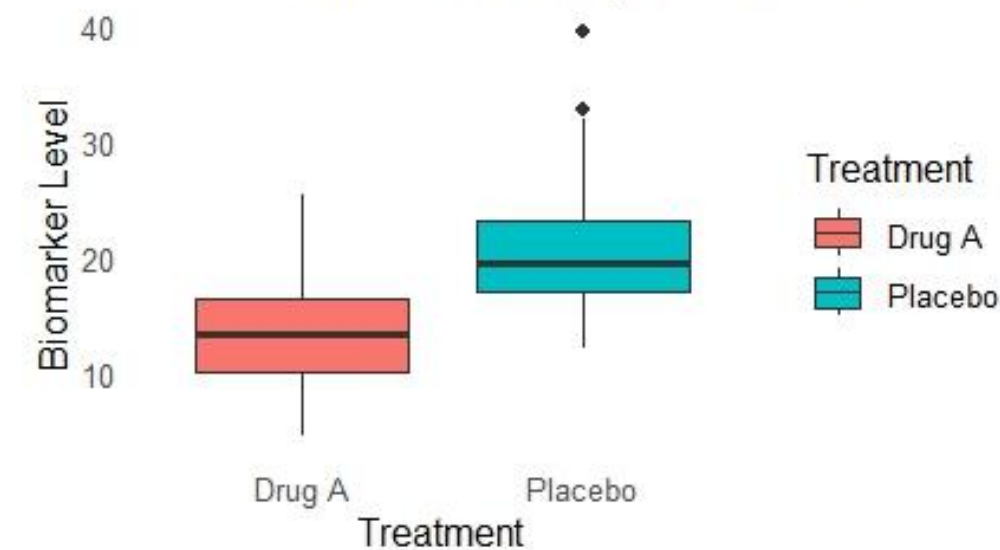
# Density Plot
density_plot <- ggplot(oncology_data, aes(x = Biomarker_Level, fill = Treatment)) +
  geom_density(alpha = 0.6) +
  labs(title = "Density Plot: Biomarker Levels", x = "Biomarker Level", y = "Density") +
  theme_minimal() +
  theme(
    panel.grid.major = element_blank(),
    panel.grid.minor = element_blank()
  ) +
  theme(plot.title = element_text(hjust = 0.5))

# Arrange all plots in a grid
grid.arrange(violin_plot, box_plot, histogram_plot, density_plot, ncol = 2)
```

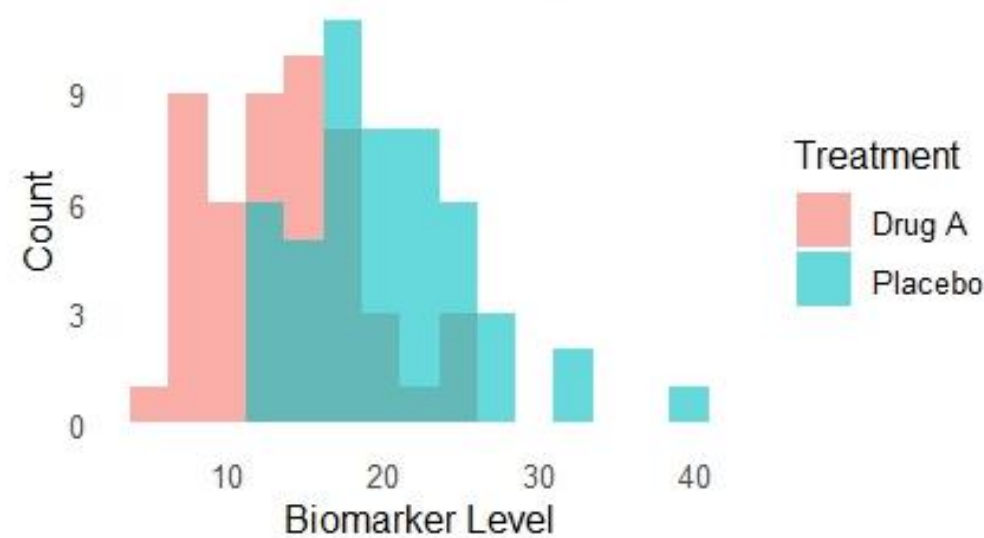
Violin Plot: Biomarker Levels by Treatment



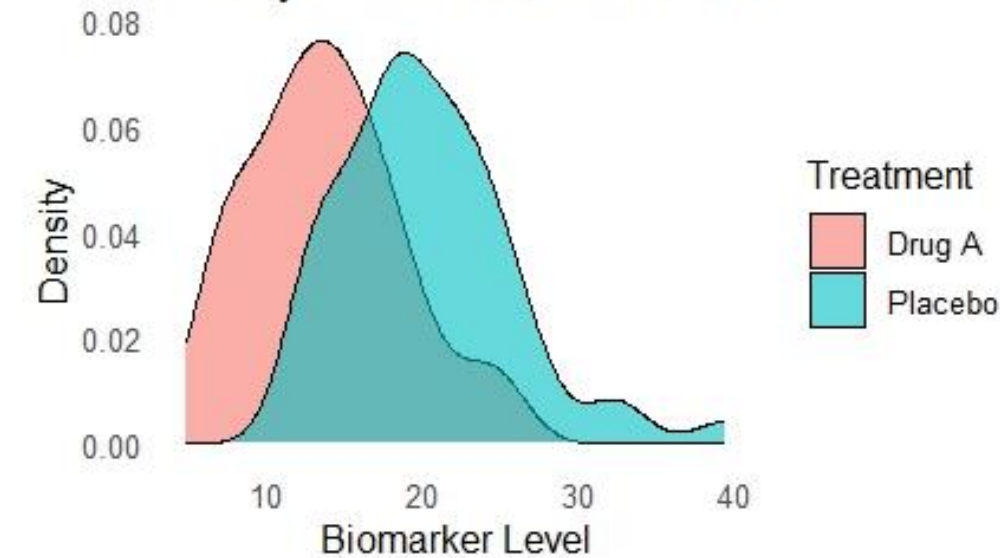
Box Plot: Biomarker Levels by Treatment



Histogram: Biomarker Levels



Density Plot: Biomarker Levels



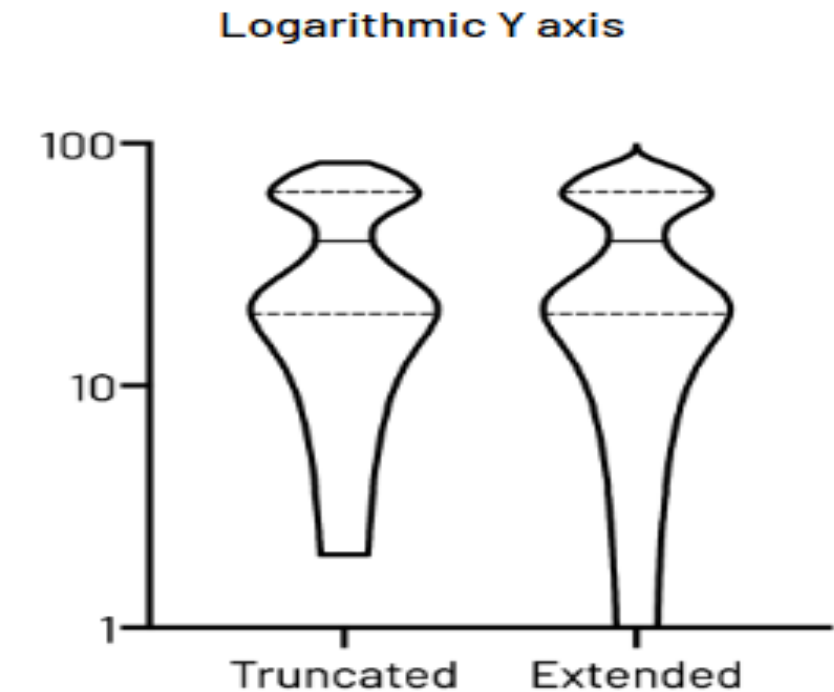
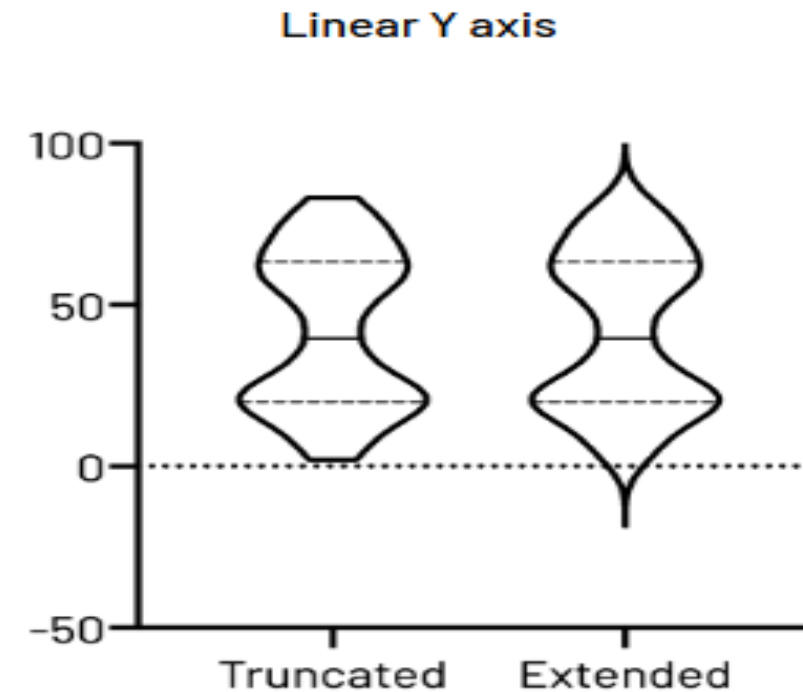
6. Usage in Oncology

Usage	How Violin Plots Help in Oncology Data Analysis
Data Distribution	Displays the full probability distribution, showing if biomarker levels or tumor responses follow a normal or skewed distribution.
Variability Assessment	Highlights the spread of data within each treatment group, indicating whether responses are consistent or highly variable.
Multimodal Distributions	Helps detect subpopulations in oncology data by revealing multiple peaks that standard boxplots might miss.
Comparing Treatment Responses	Enables visual comparison of different treatments by showing how their distributions differ in shape and spread.
Outlier Identification	Provides better context for outliers by showing density, which is useful in identifying exceptional responders or adverse effects.
Clinical Interpretation	Helps researchers and clinicians detect shifts in distributions, indicating treatment efficacy or toxicity concerns.
Oncology	Comparing biomarker levels (e.g., PD-L1, VEGF) across treatment arms
	Evaluating tumor shrinkage over time
	Representing survival analysis (PFS, OS) distributions

7. Limitations of Violin Plot

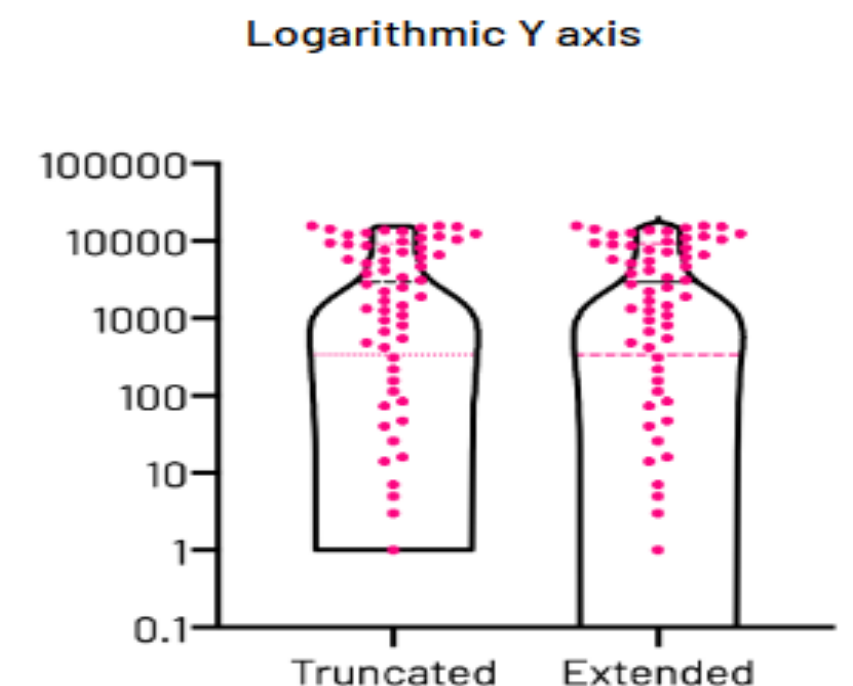
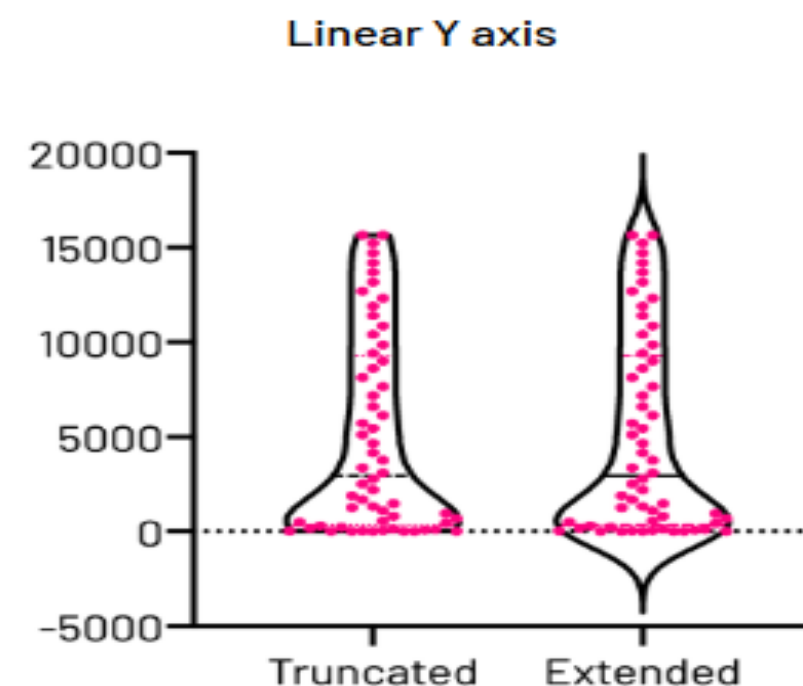
Issue # 1

- ✓ Even though the data used to generate a violin plot contains only positive numbers
- ✓ the violin itself may extend beyond zero into negative values
- ✓ This is problematic because logarithms can't be negative (or zero)

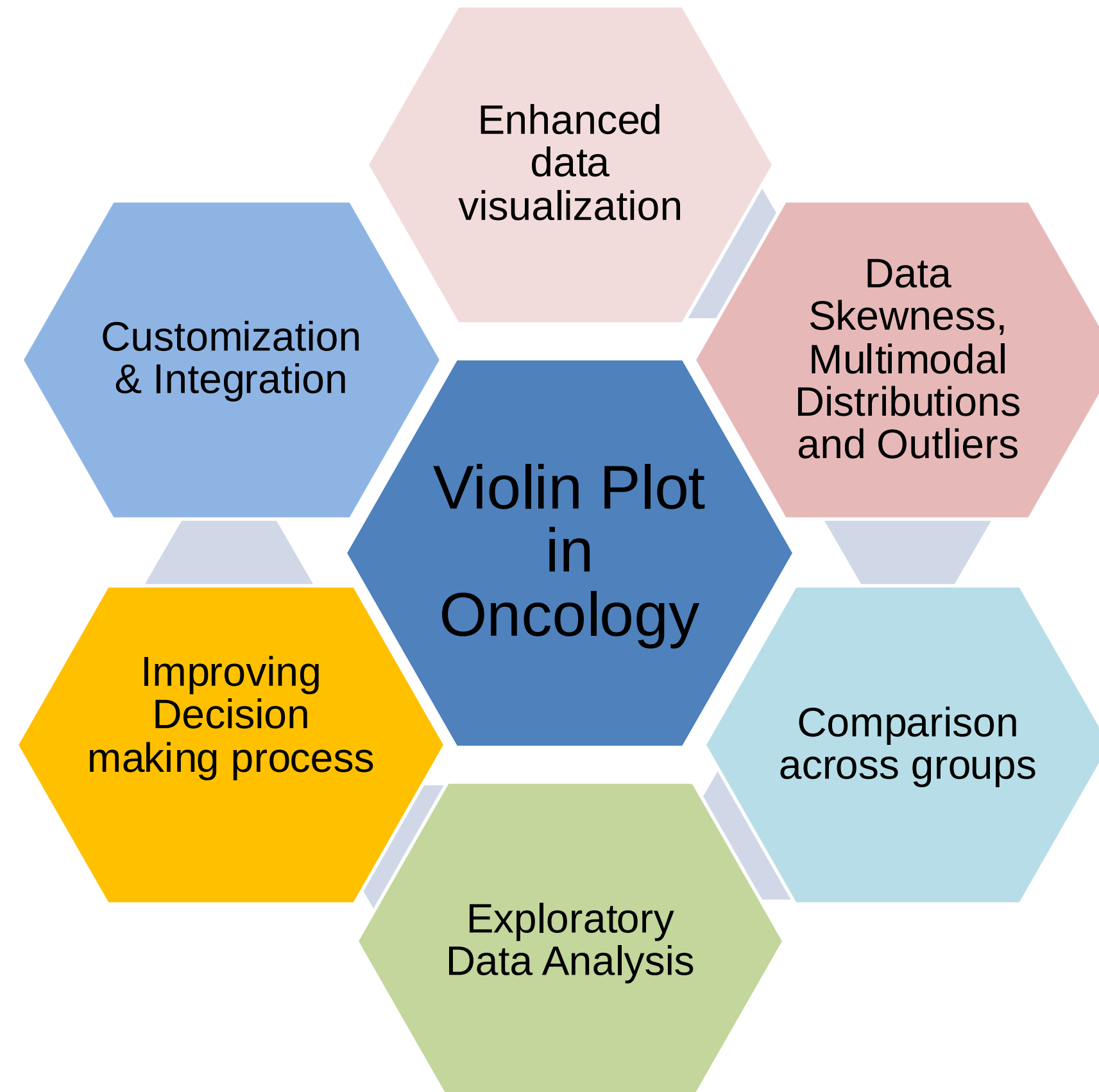


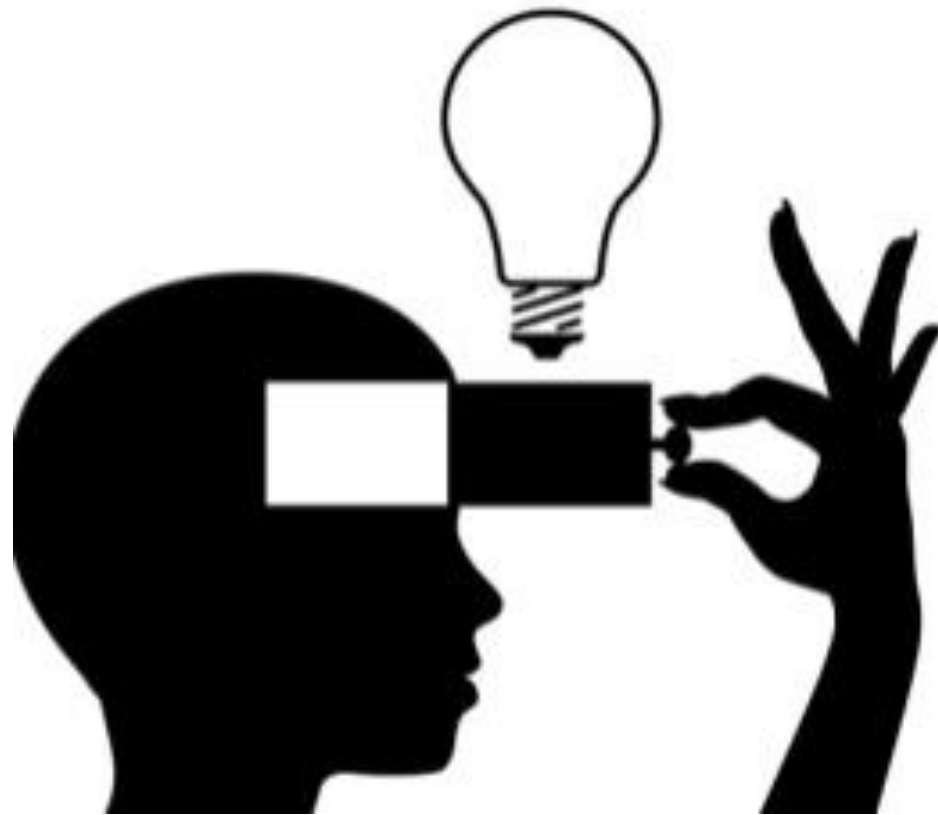
Issue # 2

- ✓ The width of violin plots is determined by examining the distance between values in a linear fashion
- ✓ This is problematic because the distance between values on a logarithmic axis is not uniform



8. Conclusion





“ Learning never
exhausts the
mind. ”

~Leonardo da Vinci

THANK YOU