Distribution_VIOLIN

2024-08-25

#— #title: “Distribution” #output: html_document #date: “2024-08-25” #—

#Load packages

Define APA theme

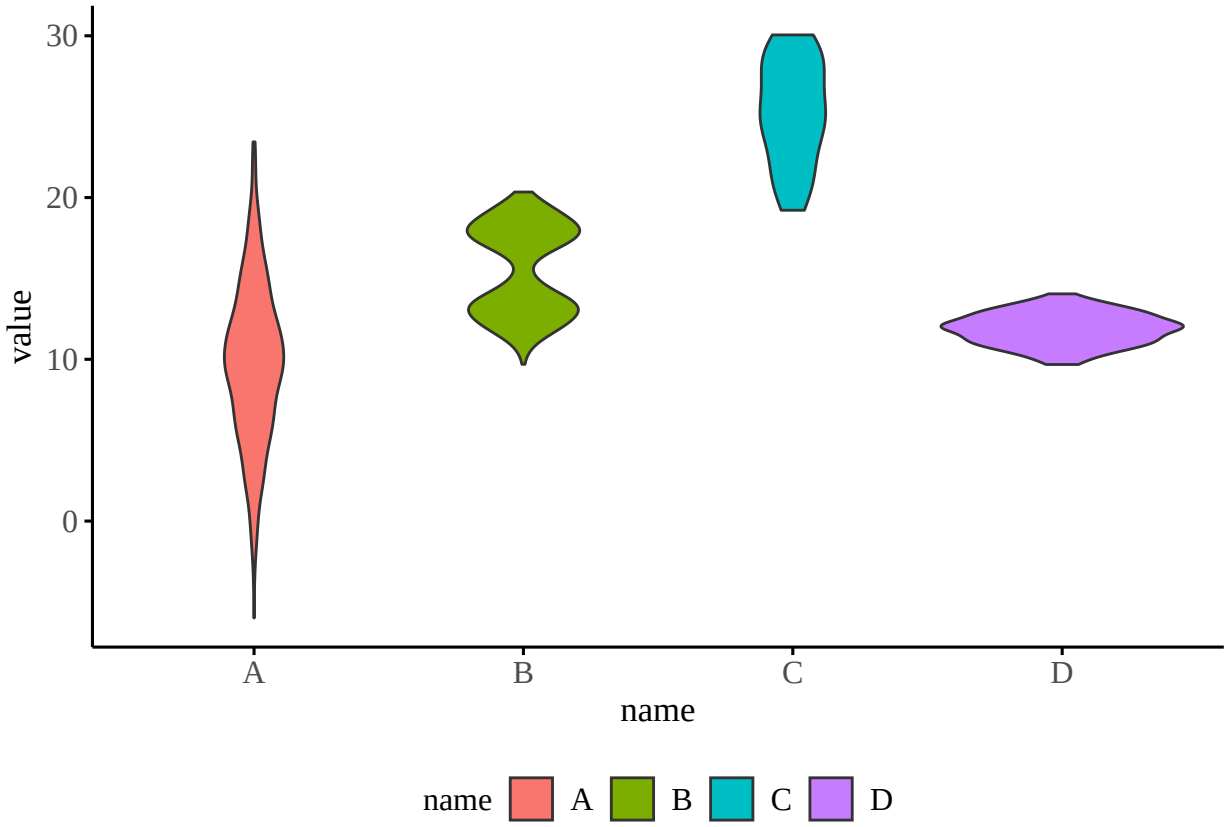
```
theme_apa <- function(base_size = 12, base_family = "serif") {
  theme_classic(base_size = base_size, base_family = base_family) +
    theme(
      # Title and subtitle
      plot.title = element_text(face = "bold", size = base_size * 1.2, hjust = 0.5),
      plot.subtitle = element_text(size = base_size, hjust = 0.5),

      # Axis titles and text
      axis.title = element_text(size = base_size * 1.1),
      axis.text = element_text(size = base_size),
      axis.line = element_line(color = "black"),
      axis.ticks = element_line(color = "black"),

      # Panel and background
      panel.grid.major = element_blank(),
      panel.grid.minor = element_blank(),
      panel.background = element_blank(),

      # Legend
      legend.title = element_text(size = base_size),
      legend.text = element_text(size = base_size),
      legend.position = "bottom",
      legend.key = element_blank()
    )
}
```

I. VIOLIN

Basic violin plot

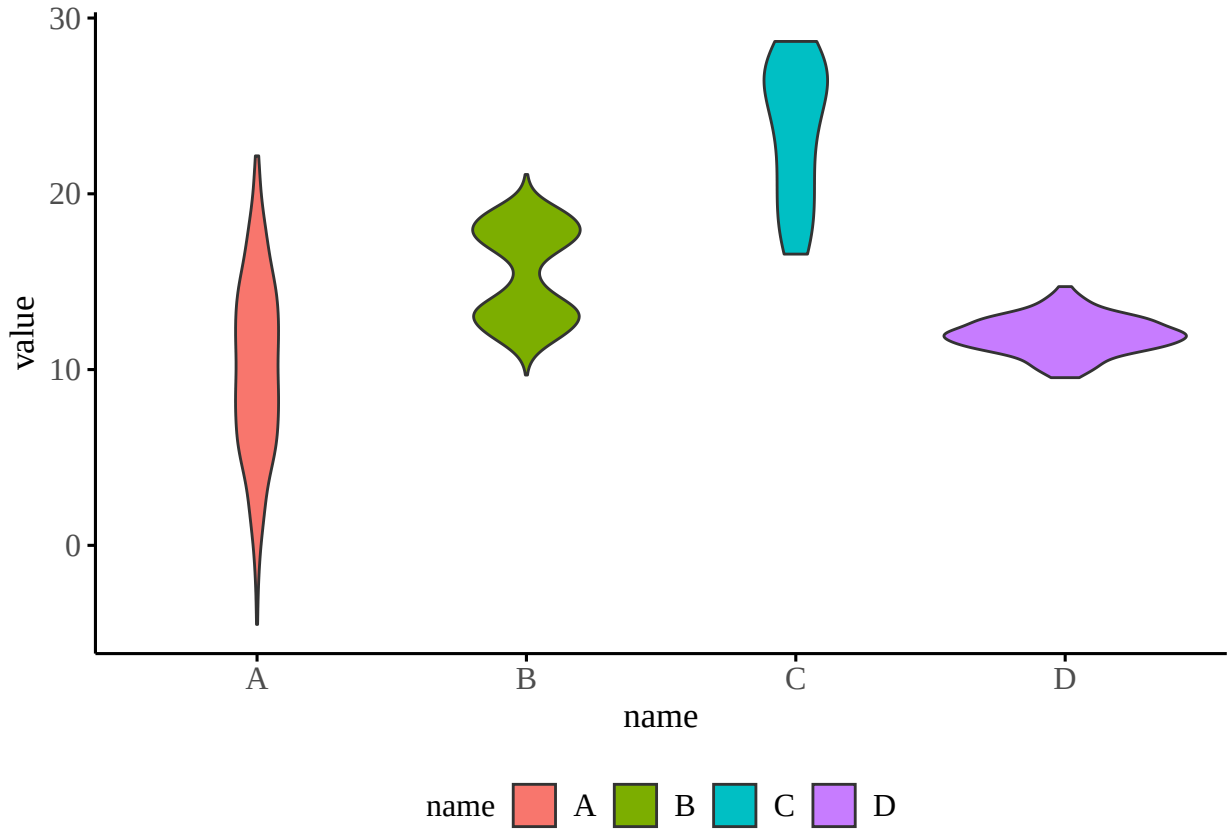
```
# Library
library(ggplot2)

# create a dataset
data <- data.frame(
  name=c( rep("A",500), rep("B",500), rep("B",500), rep("C",20), rep('D', 100)  ),
  value=c( rnorm(500, 10, 5), rnorm(500, 13, 1), rnorm(500, 18, 1), rnorm(20, 25, 4), rnorm(100, 12, 1)
)

# Most basic violin chart
```

```
p <- ggplot(data, aes(x=name, y=value, fill=name)) + # fill=name allow to automatically dedicate a colo
  geom_violin() + theme_apa()

p
```



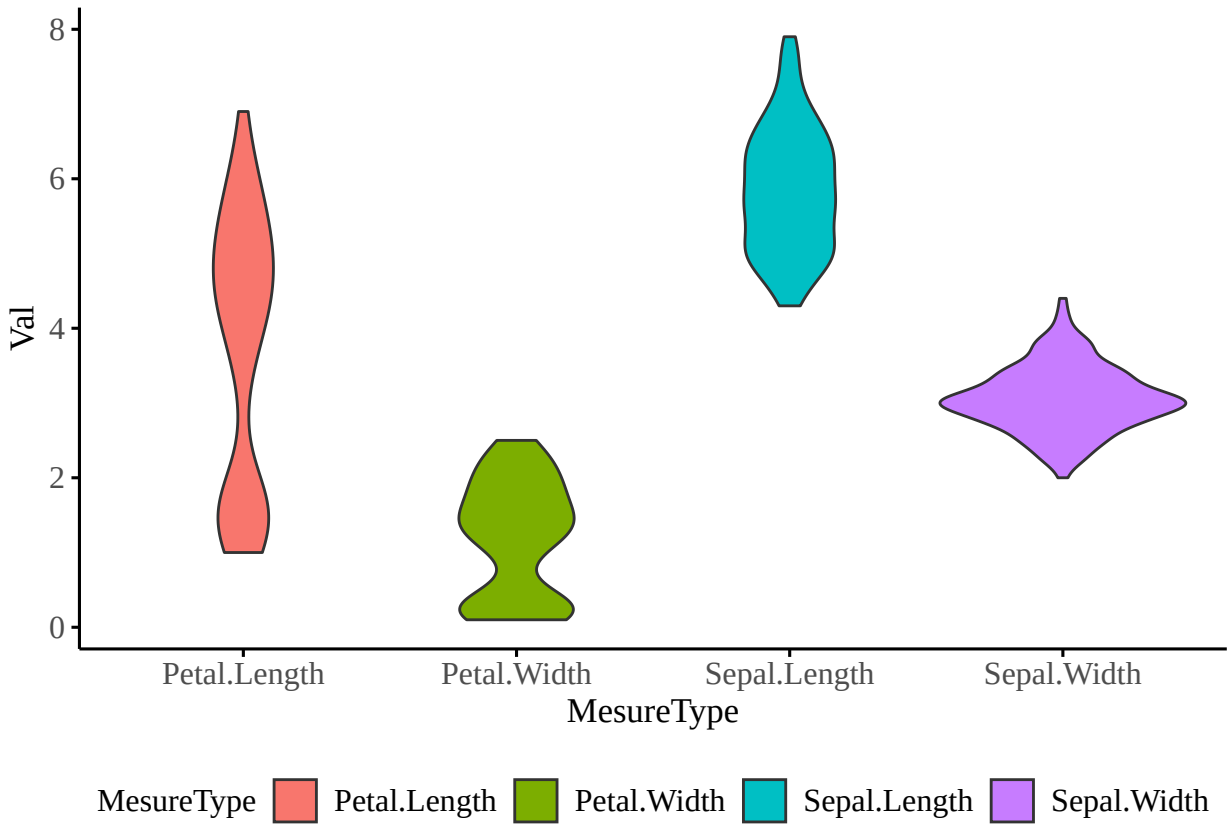
#Long format

```
# Library
library(ggplot2)
library(dplyr)

# Create data
data <- data.frame(
  name=c( rep("A",500), rep("B",500), rep("B",500), rep("C",20), rep('D', 100)  ),
  value=c( rnorm(500, 10, 5), rnorm(500, 13, 1), rnorm(500, 18, 1), rnorm(20, 25, 4), rnorm(100, 12, 1)
)



# Basic violin
ggplot(data, aes(x=name, y=value, fill=name)) +
  geom_violin() + theme_apa()
```



#Wide format

```
data_wide <- iris[ , 1:4]



library(tidyr)
library(ggplot2)
library(dplyr)
data_wide %>%
  gather(key="MesureType", value="Val") %>%
  ggplot( aes(x=MesureType, y=Val, fill=MesureType)) +
    geom_violin() + theme_apa()
```



#Violin plot with included boxplot: Including a boxplot within a violin plot can be useful for visualizing both the distribution of the data and its summary statistics. Additionally, displaying the sample size of each group along the X-axis is often an essential step.

```
# Libraries
library(ggplot2)
library(dplyr)
library(hrbrthemes)
library(viridis)
```

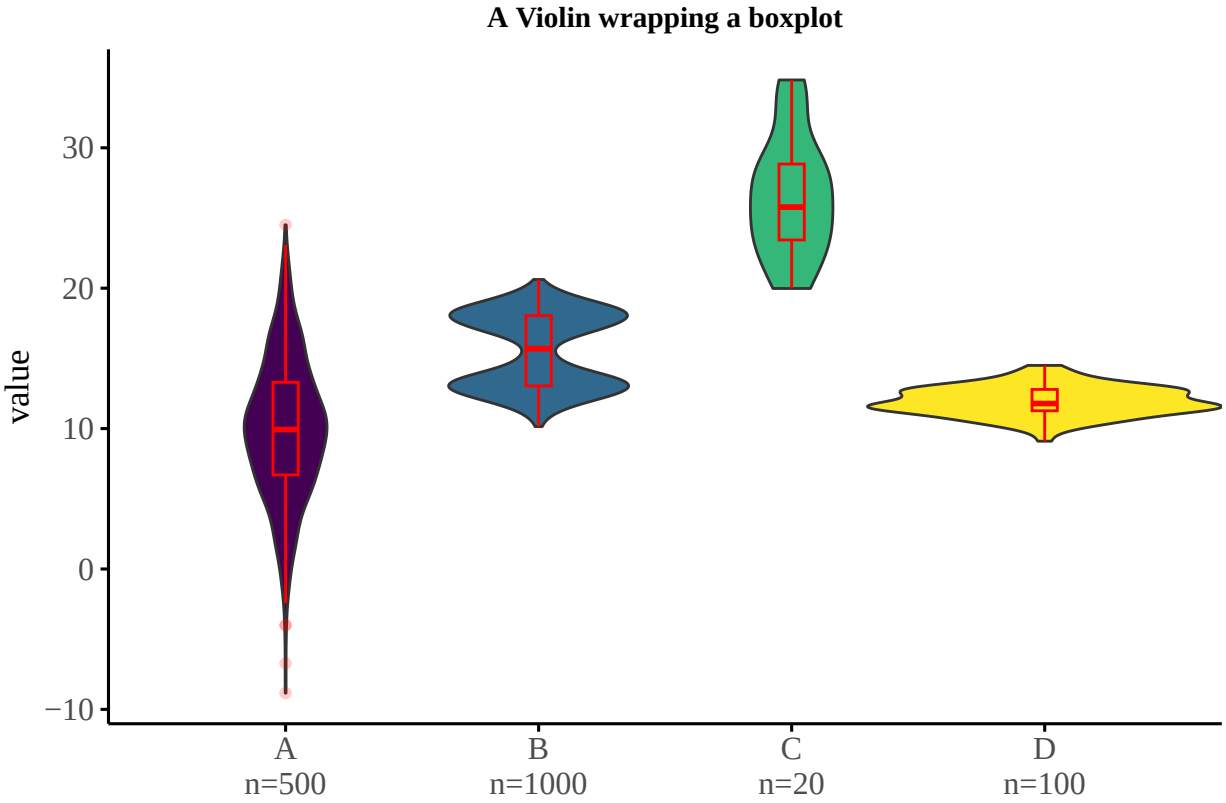
```
## Loading required package: viridisLite
```

```
# create a dataset
data <- data.frame(
  name=c( rep("A",500), rep("B",500), rep("B",500), rep("C",20), rep('D', 100)  ),
  value=c( rnorm(500, 10, 5), rnorm(500, 13, 1), rnorm(500, 18, 1), rnorm(20, 25, 4), rnorm(100, 12, 1)
)

# sample size
sample_size = data %>% group_by(name) %>% summarize(num=n())

# Plot
data %>%
  left_join(sample_size) %>%
  mutate(myaxis = paste0(name, "\n", "n=", num)) %>%
  ggplot( aes(x=myaxis, y=value, fill=name)) +
    geom_violin(width=1.4) +
    geom_boxplot(width=0.1, color="red", alpha=0.2) +
    scale_fill_viridis(discrete = TRUE) +
```

```
    theme_ipsum() +
    theme_apa() +
    theme(
      legend.position="none",
      plot.title = element_text(size=11)
    ) +
    ggtitle("A Violin wrapping a boxplot") +
    xlab("")
```

```
## Joining with `by = join_by(name)`
```

```
## Warning: `position_dodge()` requires non-overlapping x intervals.
```



#Horizontal violin: Violin plots are effective for comparing the distributions of multiple groups. To improve label readability, it often makes sense to create a horizontal version, making the group labels easier to read.

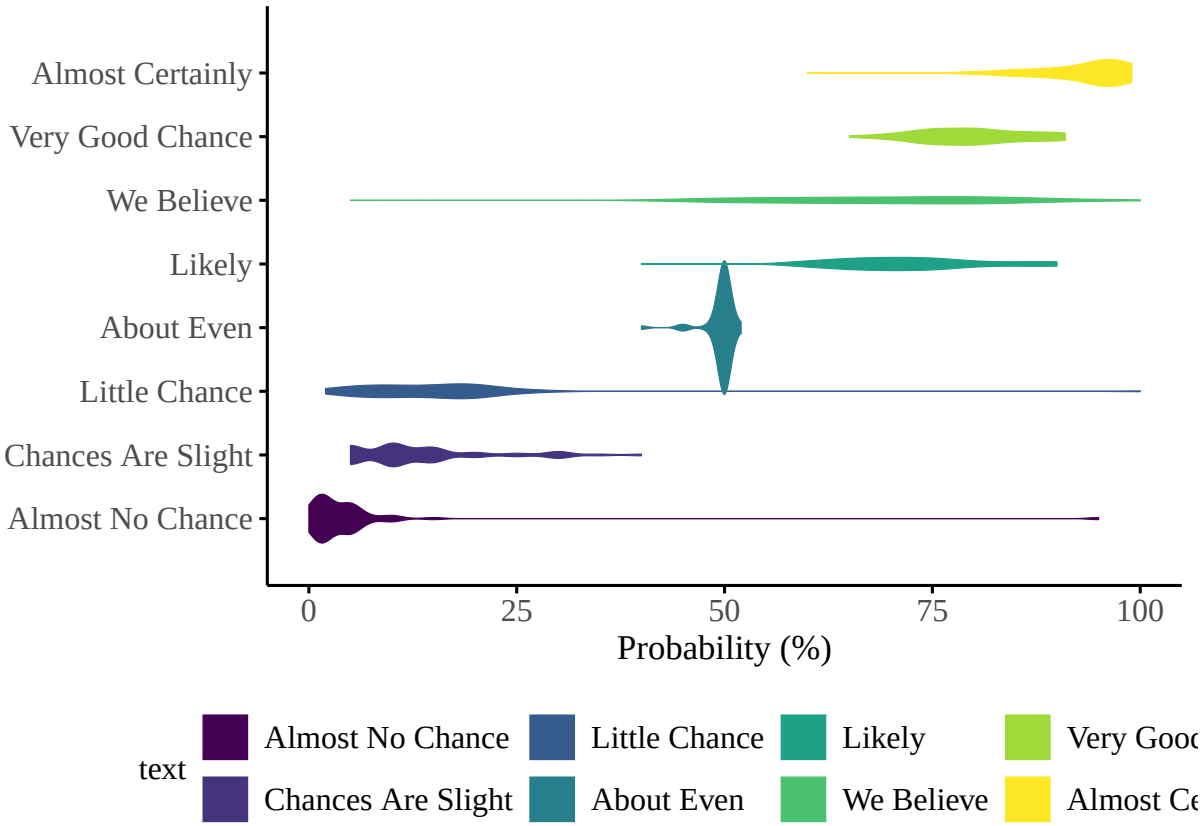
```
# Load dataset from github
data <- read.table("https://raw.githubusercontent.com/zonination/perceptions/master/probly.csv", header=

# Data is at wide format, we need to make it 'tidy' or 'long'
data <- data %>%
  gather(key="text", value="value") %>%
  mutate(text = gsub("\\.", " ",text)) %>%
  mutate(value = round(as.numeric(value),0)) %>%
  filter(text %in% c("Almost Certainly","Very Good Chance","We Believe","Likely","About Even", "Little 

# Plot
p <- data %>%
  mutate(text = fct_reorder(text, value)) %>% # Reorder data
  ggplot( aes(x=text, y=value, fill=text, color=text)) +
```

```
    geom_violin(width=2.1, size=0.2) +
    scale_fill_viridis(discrete=TRUE) +
    scale_color_viridis(discrete=TRUE) +
    theme_ipsum() +
    theme(
      legend.position="none"
    ) + theme_apa() +
    coord_flip() + # This switch X and Y axis and allows to get the horizontal version
    xlab("") +
    ylab("Probability (%)")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
p
```

```
## Warning: `position_dodge()` requires non-overlapping x intervals.
```



#Grouped violin chart: A grouped violin plot illustrates the distribution of a numeric variable across different groups and subgroups. In this case, the groups represent the days of the week, while the subgroups distinguish between Males and Females. The ggplot2 package facilitates this type of visualization using the position = "dodge" option within the geom_violin() function. The groups should be mapped to the x aesthetic, and the subgroups should be mapped to the fill aesthetic.

```
# Libraries
library(ggplot2)
library(dplyr)
```

```r
library(forcats)
library(hrbrthemes)
library(viridis)

# Load dataset from github
data <- read.table("https://raw.githubusercontent.com/holtzy/data_to_viz/master/Example_dataset/10_OneNu
  mutate(tip = round(tip/total_bill*100, 1))

# Grouped
data %>%
  mutate(day = fct_reorder(day, tip)) %>%
  mutate(day = factor(day, levels=c("Thur", "Fri", "Sat", "Sun"))) %>%
  ggplot(aes(fill=sex, y=tip, x=day)) +
    geom_violin(position="dodge", alpha=0.5, outlier.colour="transparent") +
    scale_fill_viridis(discrete=T, name="") +
    theme_apa()  +
    xlab("") +
    ylab("Tip (%)") +
    ylim(0,40)
```

```
## Warning in geom_violin(position = "dodge", alpha = 0.5, outlier.colour =
## "transparent"): Ignoring unknown parameters: `outlier.colour`
```

```
## Warning: Removed 2 rows containing non-finite outside the scale range
## (`stat_ydensity()`).
```