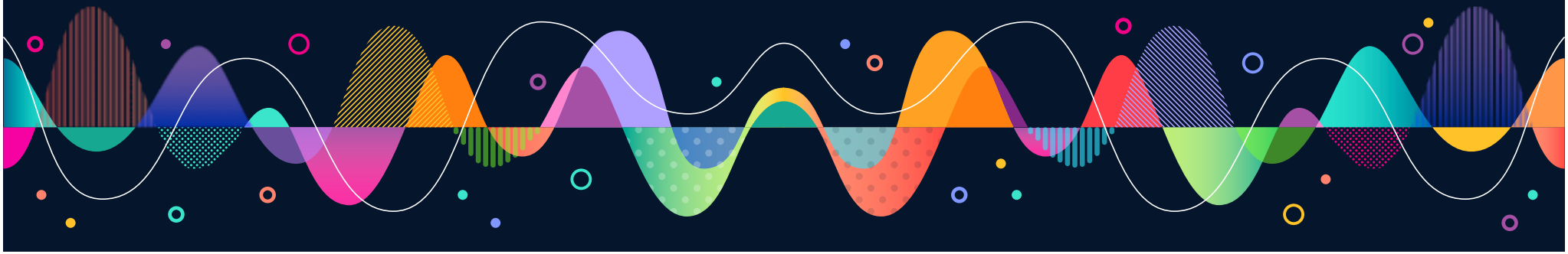
An Introduction to ggplot2

Roy Roberts and Chris Bird



Included Data Sets

- storms (dplyr)
  - Storm tracks data
  - `stm <-tibble(storms)%>%`

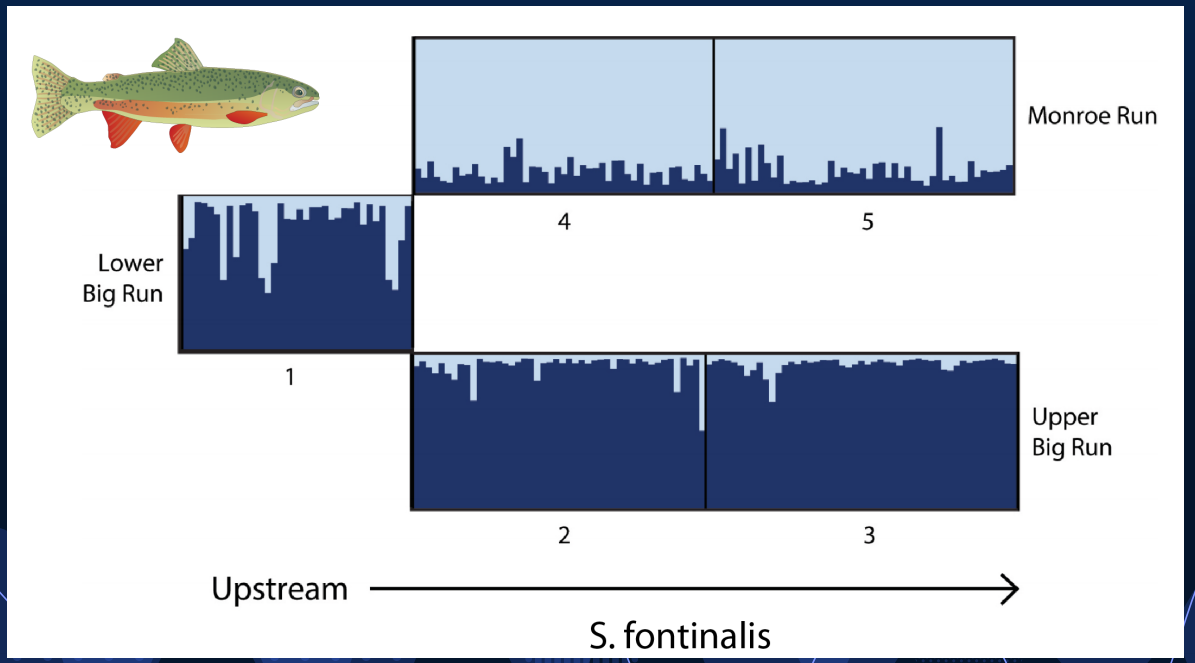
    `storms_ymdh()`

- diamonds (ggplot2)
  - Prices of over 50,000 round cut diamonds
  - `dmd <-tibble(diamonds)`

Scientific Visualizations Should be Clear

- What is being plotted?
- What are the axes?
- What do the colors, shapes, & sizes represent?
- What is the message you want to convey?

What's wrong with this figure?



Philosophy of ggplot2

- The grammar of graphics (Wilkinson 2006)
  - Describe a graph using text
- **Data** – data frame to plot
- **Aesthetic Mapping (aes)**
  - which parts of data frame are associated with which parts of plot
    - X-axis, Y-axis, colors of points, etc
- **Geometry (geom)**
  - Lines, points, etc
- **Coordinates**
  - cartesian, polar
- **Faceting**
  - Panels
- **Statistics**
  - Fit line, smoothing, etc
- **Theme**
  - Adjust look and feel of the graphics

Building a ggplot

- Plot the wind speed for every storm in the year 1998
- Add data

  `data = stm`
- Add aesthetics (aes)

  `x = ymdh`

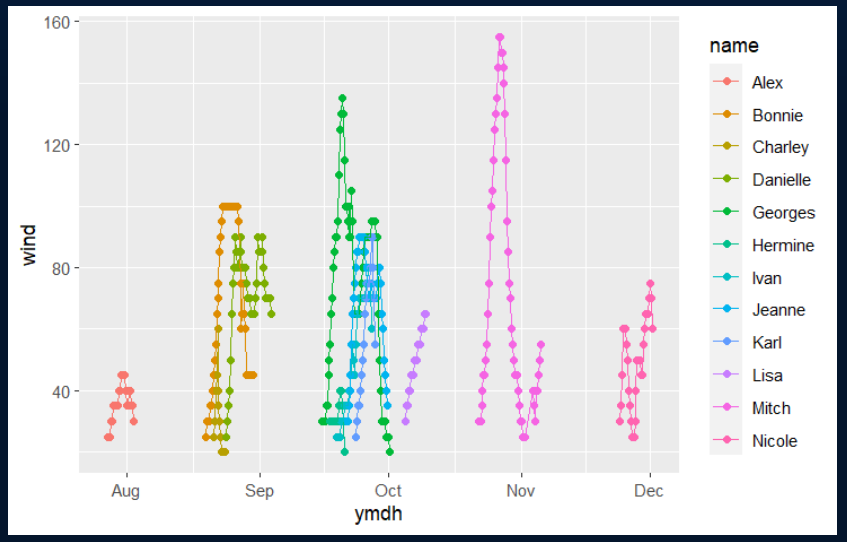
  `y = wind`

  `colour = name`
- Add data point geometry

  `geom_point()`
- Add line geometry

  `geom_line()`

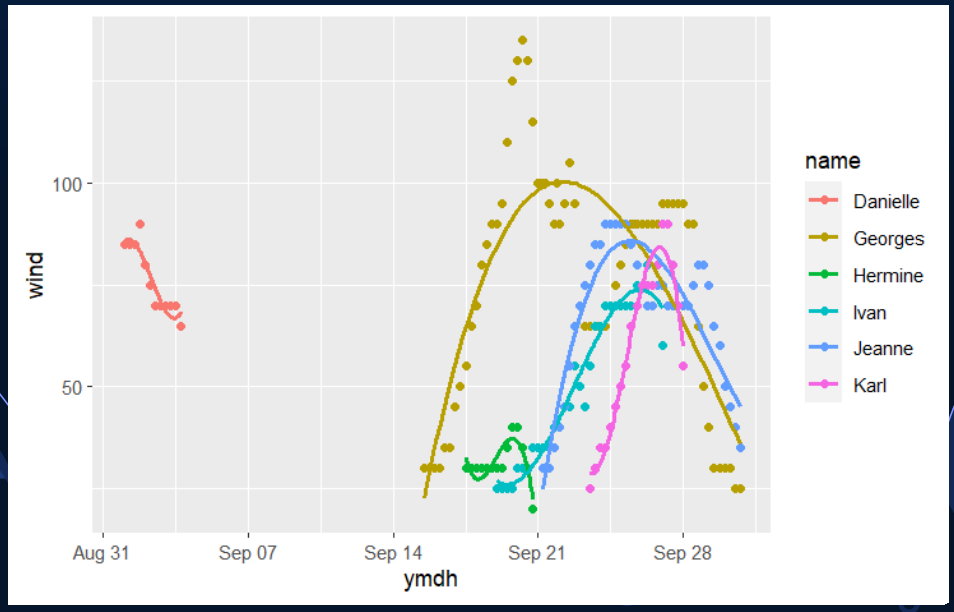
```
stm %>%
  filter(year == 1998) %>%
  ggplot(aes(y = wind,
             x = ymdh,
             colour = name)) +
  geom_point() +
  geom_line()
```



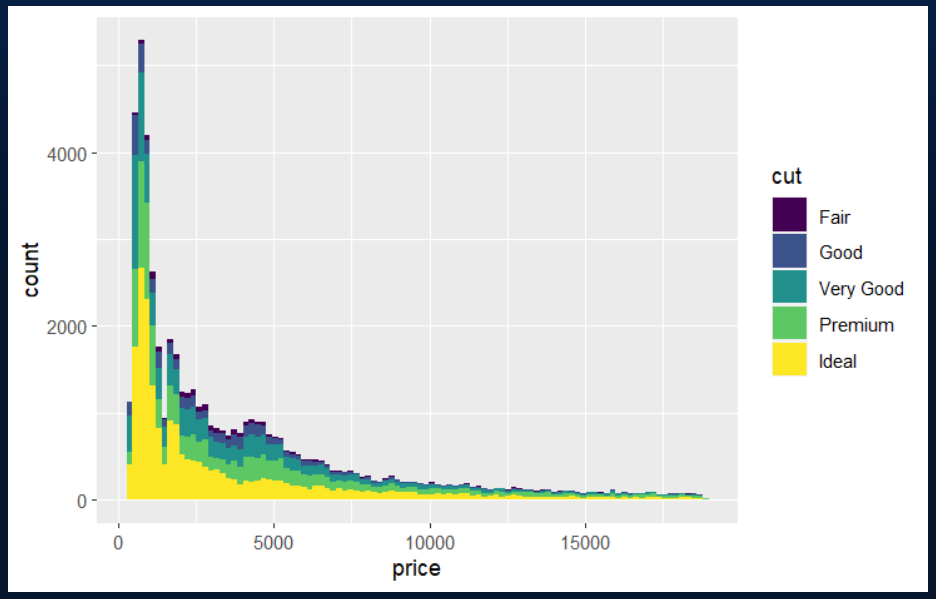
Scatter Plot with Trend Lines

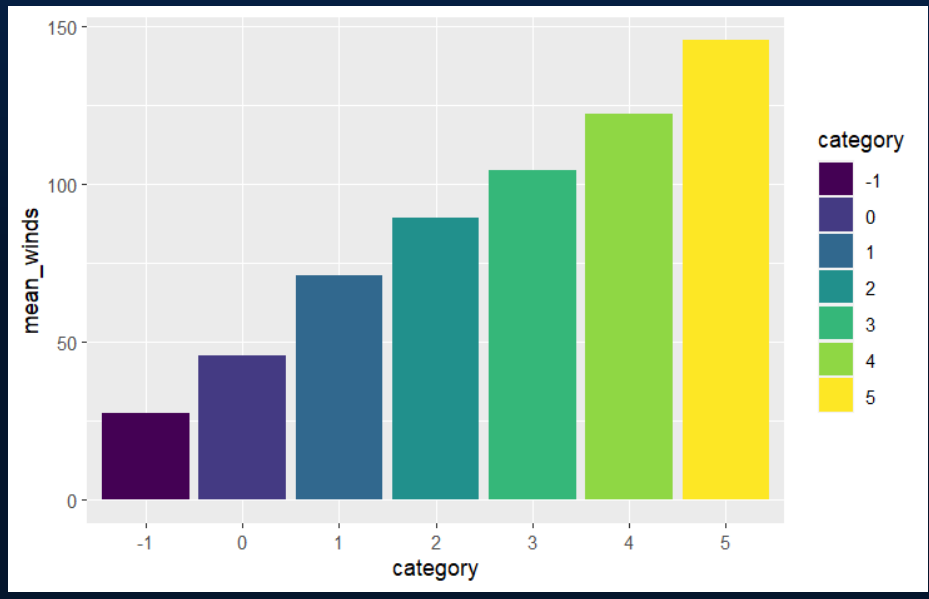
- Visualize relationship between two continuous variables
- Fit lines can be added
  - `geom_smooth()`
    - Uses loess smoother by default
  - `geom_smooth(method="lm")`
    - Linear model fit line
  - `geom_smooth(method="lm", formula = y ~ poly(x,3))`
    - 3rd order polymonial model fit line
  - `geom_smooth(se=FALSE)`
    - Remove error "ribbon"

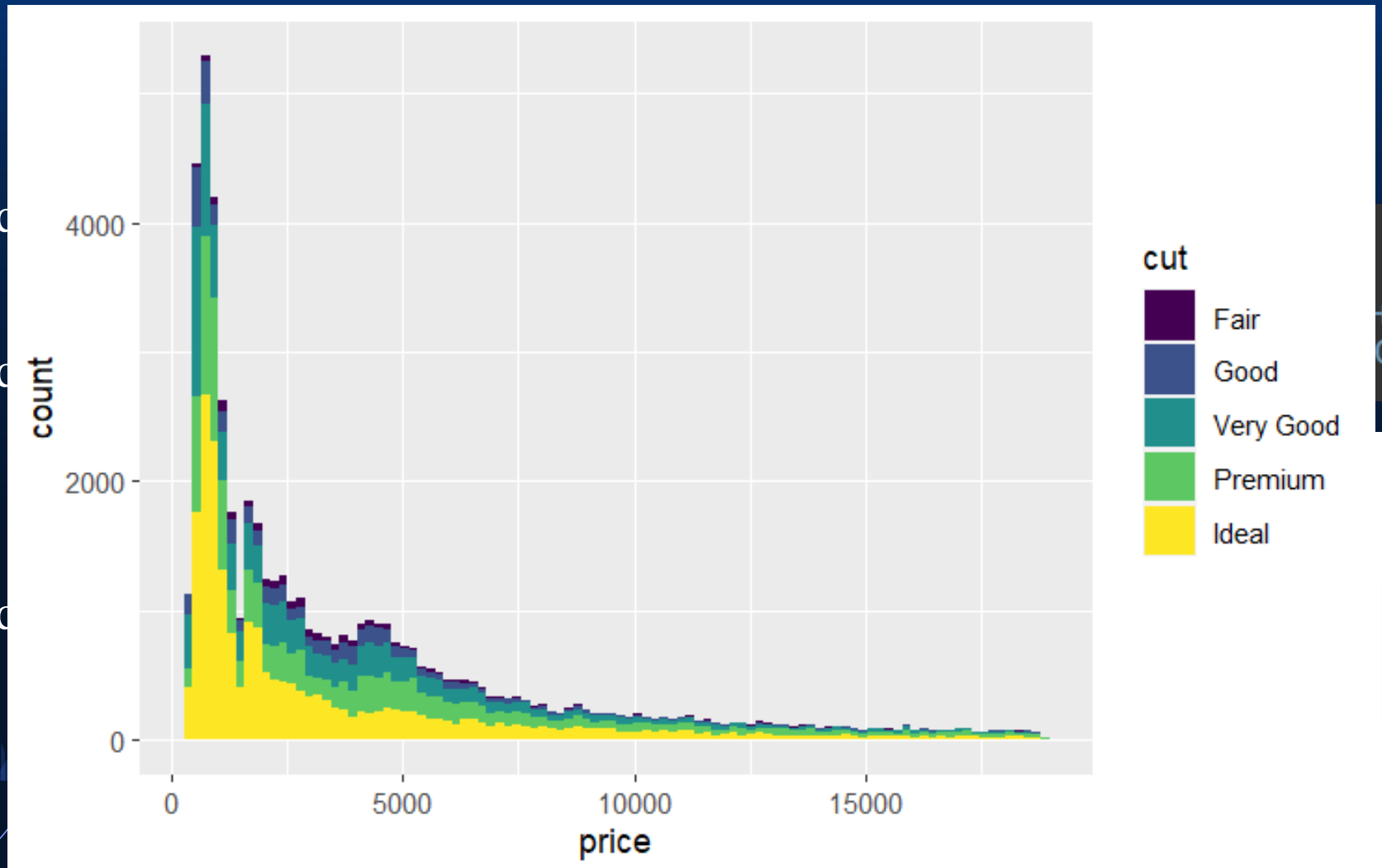
```
stm %>%
  filter(year == 1998,
         month == "9") %>%
  ggplot(aes(y = wind,
             x = ymdh,
             color = name)) +
  geom_point() +
  geom_smooth(method = "lm",
              formula = y ~ poly(x,3),
              se = FALSE)
```



Histogram VS Bar Plot



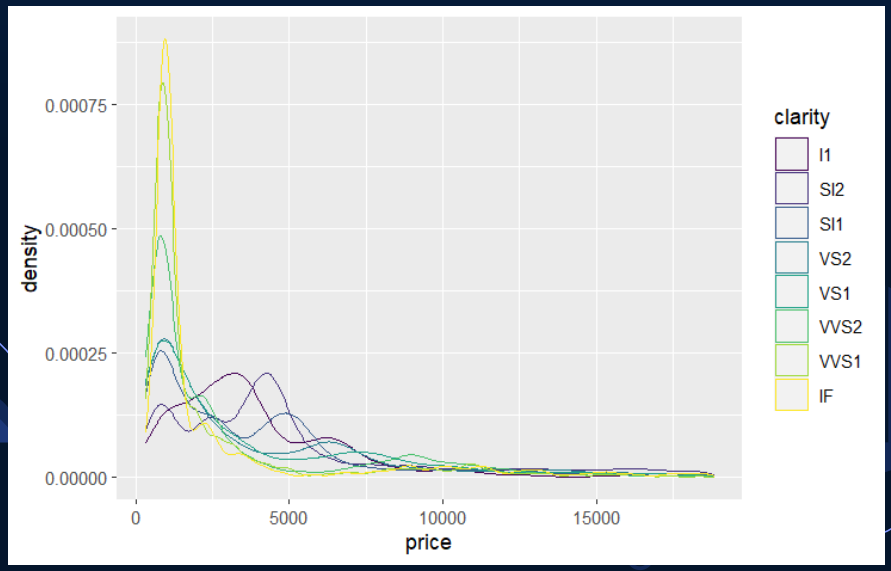



4000
2000
0
count
0
5000
10000
15000
price
cut
Fair
Good
Very Good
Premium
Ideal

Density Plot

- Add data
  - `data = dmd`
- Add aesthetics
  - `aes(x = price,`
    `colour = cut)`
  - Color changes the color of the line
  - Fill colors the area below the line
- Add data point geometry
  - `geom_density()`

```
dmd %>%
  ggplot(aes(x = price,
             color = clarity)) +
  geom_density()
```



Bar Plot

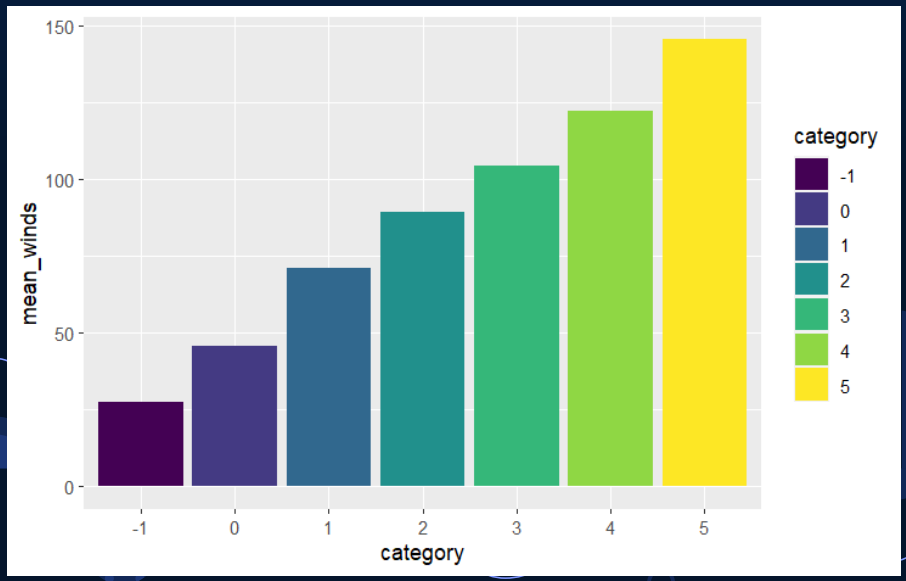
- Summarise storms, grouping by category, to calculating the mean wind speed
- Add aesthetics

```
aes(x = [          ],
    y = [          ],
    fill = [       ])
```

- Add data point geometry

```
geom_col()
```

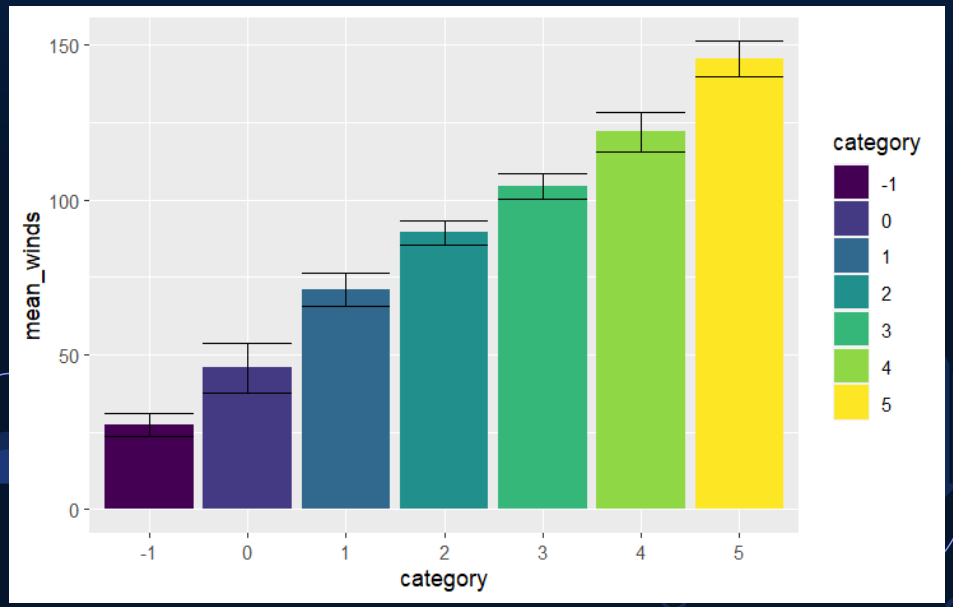
```
stm %>%
  group_by(category) %>%
  summarise(mean_winds = mean(wind)) %>%
  ungroup() %>%
  ggplot(aes(x = category,
             y = mean_winds,
             fill = category)) +
  geom_col()
```



Error Bars

- Error bars are typically plotted based upon values that you calculate
- Standard deviation can be calculated by adding it to summarise()
- Adding error bars
- **geom_errorbar(aes(ymax, ymin))**

```
stm %>%
  group_by(category) %>%
  summarise(mean_winds = mean(wind),
            sd_winds = sd(wind)) %>%
  ungroup() %>%
  ggplot(aes(x = category,
             y = mean_winds,
             fill = category)) +
  geom_col() +
  geom_errorbar(aes(ymin = mean_winds - sd_winds,
                    ymax = mean_winds + sd_winds))
```



Box and Violin Plots

- Add data
  - `data = dmd`
- Add aesthetics
  - `aes(x = cut,`
    `y = depth,`
    `fill = cut)`

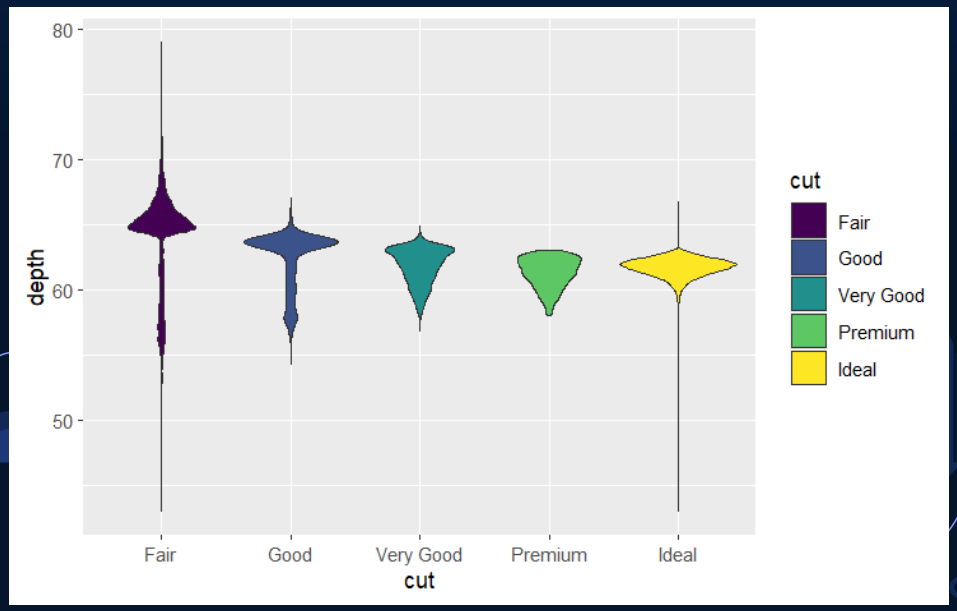
Add data point geometry

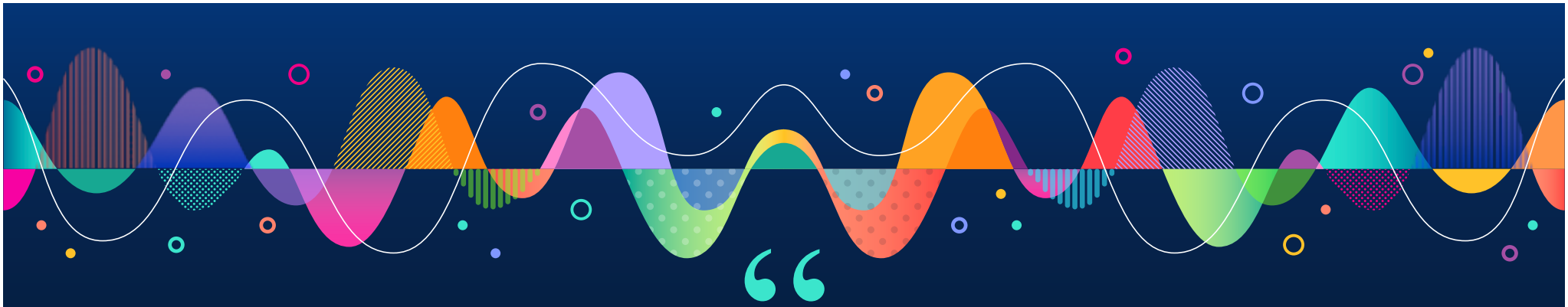
- `geom_boxplot()`

or

- `Geom_violinplot()`

```
dmd %>%
  ggplot(aes(x = cut,
             y = depth,
             fill = cut)) +
  geom_violin()
```



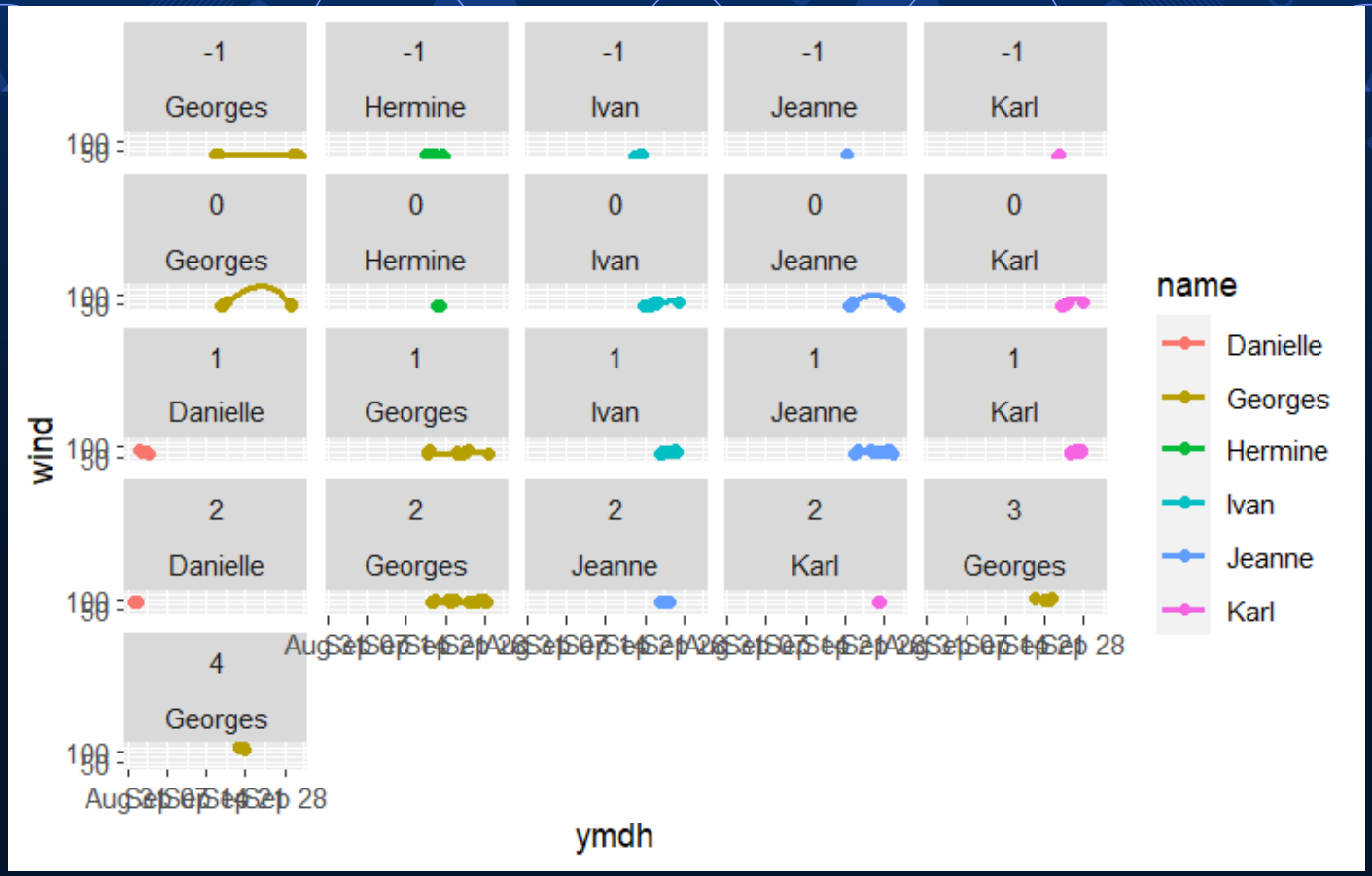


ggplot2 Assignment 1

Using the provided PSMC data, make an informative plot

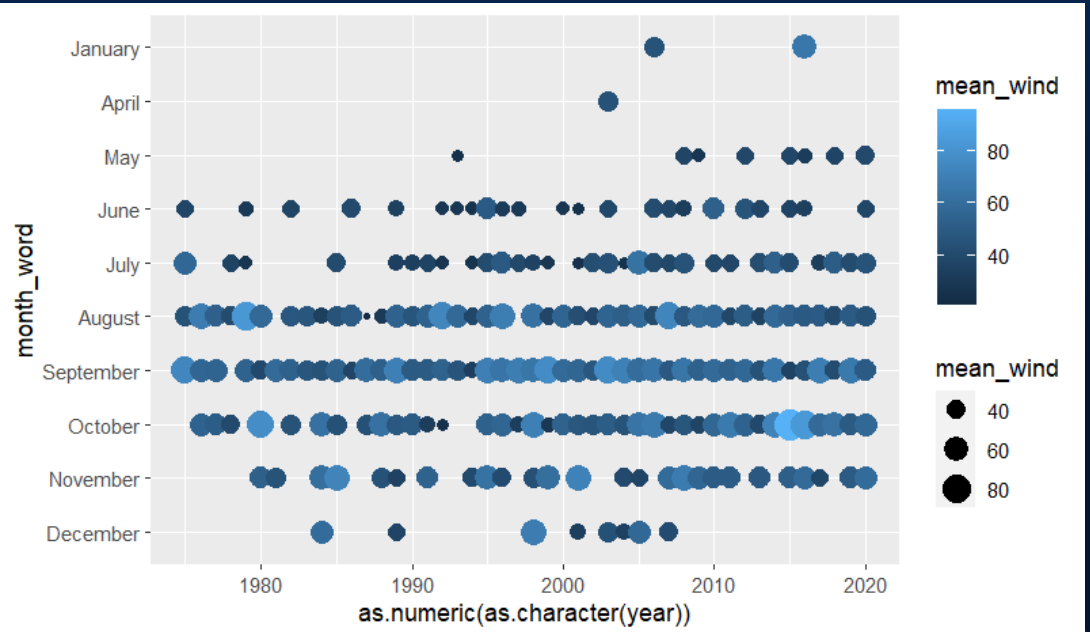
Faceting

- Faceting visually splits your data
- Creates a matrix of panels based on specified variables
  - `facet_grid(column2~column1)`
  - `facet_wrap(column2~column1)`
- Faceting options
  - `scale = "free_x" or "free_y", or "free"`
  - `rows = Num.`
  - `columns = Num.`


-1
Georges
-1
Hermine
-1
Ivan
-1
Jeanne
-1
Karl
0
Georges
0
Hermine
0
Ivan
0
Jeanne
0
Karl
1
Danielle
1
Georges
1
Ivan
1
Jeanne
1
Karl
2
Danielle
2
Georges
2
Jeanne
2
Karl
3
Georges
4
Georges
wind
ymdh
name
Danielle
Georges
Hermine
Ivan
Jeanne
Karl

Plot Modification

- Modify the x and y axes, & determine how colors, sizes, shapes behave
- **`fill`**
  - Color fill an object
- **`colour`**
  - Color of border of object, point, line
- **`size`**
  - Size of point
- **`shape/linetype`**
  - Shape of point/line
- **`alpha`**
  - Level of transparency of point or bar
- Discrete variables get discrete colors, continuous variables get continuous color



Custom Colour and Fill

- library(RColorBrewer)
  - display.brewer.all()
  - scale_colour_brewer()
  - scale_brewer_manual()
- Hexcodes
  - https://htmlcolorcodes.com/
  - https://vis4.net/palettes
- scale_colour_manual()
- scale_fill_manual()

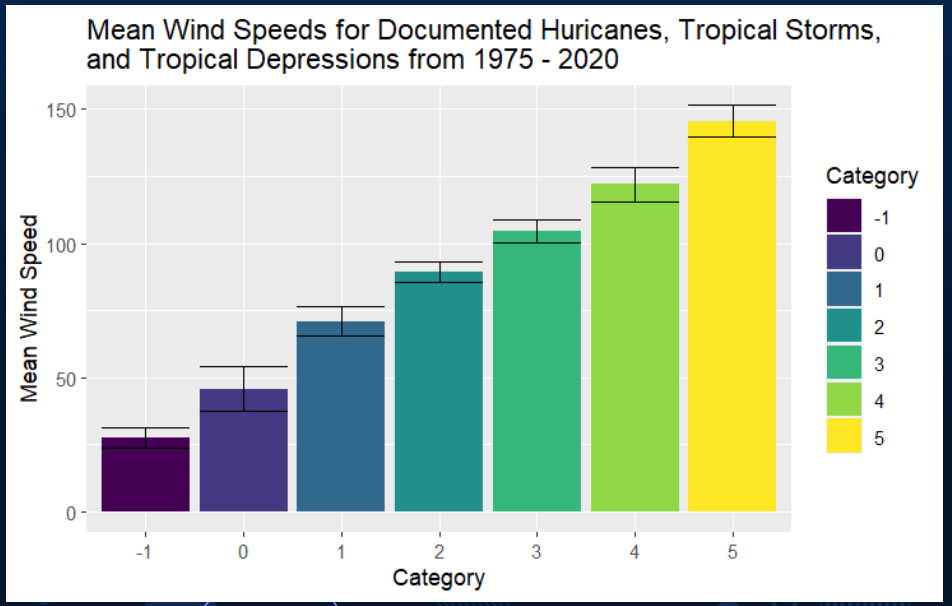
Labels

- You can change axis and title labels

```
xlab("x_axis_label")

ylab("y_axis_label")

labs(colour = "legend_label",
     fill = "legend_label",
     title = "plot_title")
```

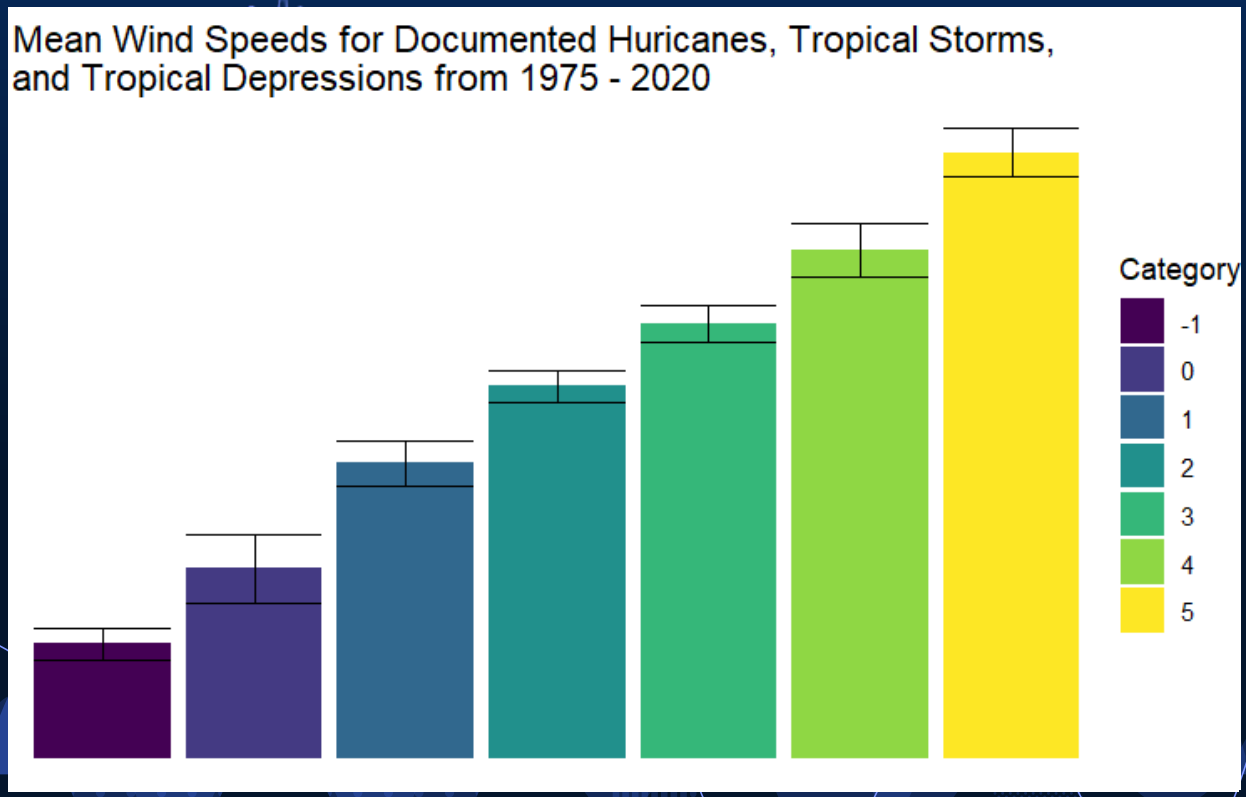


Labels

```
(themes_plot <- stm %>%
  group_by(category) %>%
  summarise(mean_winds = mean(wind),
            sd_winds = sd(wind)) %>%
  ungroup() %>%
  ggplot(aes(x = category,
             y = mean_winds,
             fill = category)) +
  geom_col() +
  geom_errorbar(aes(ymin = mean_winds - sd_winds,
                    ymax = mean_winds + sd_winds)) +
  xlab("Category") +
  ylab("Mean Wind Speed") +
  labs(fill = "Category",
       title = "Mean Wind Speeds for Documented Huricanes,
       Tropical Storms,\nand Tropical Depressions from 1975 - 2020"))
```

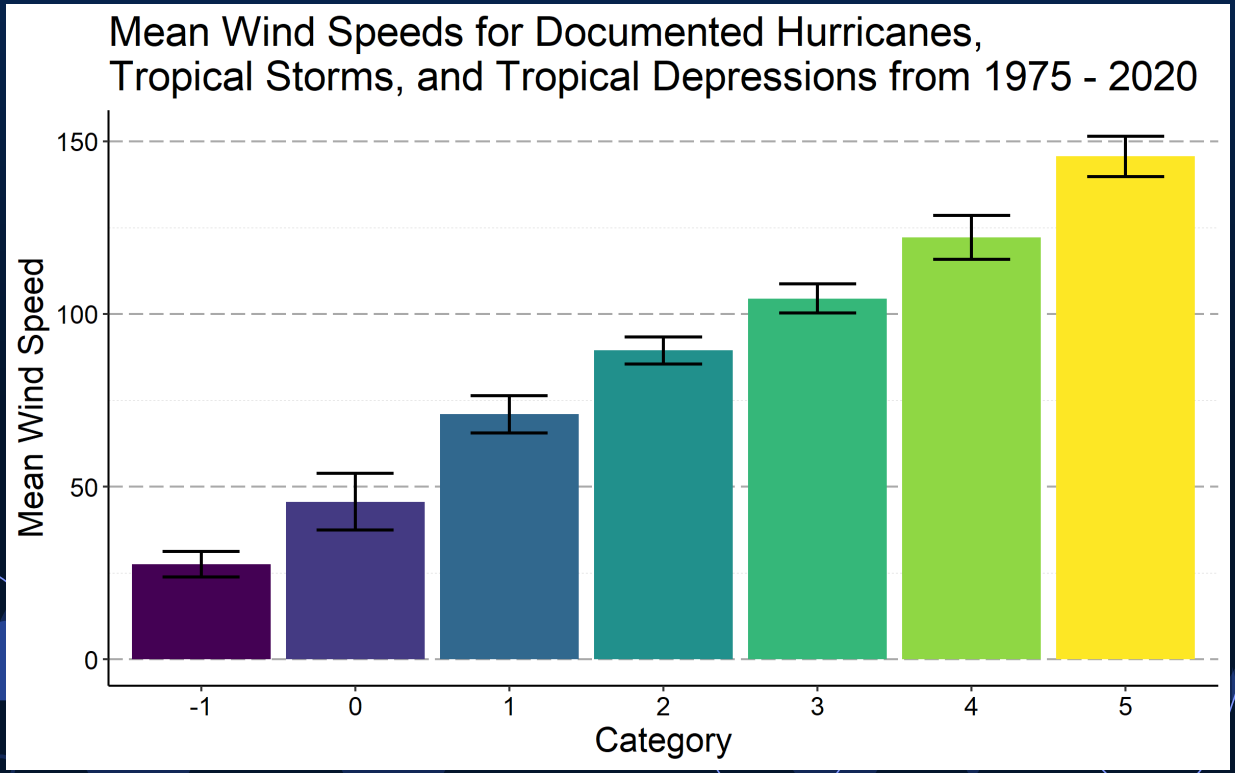
Themes

- Change the look of the plot area by adding a theme
  - theme_bw()
  - theme_classic()
  - theme_dark()
  - theme_minimal()
  - theme_void()

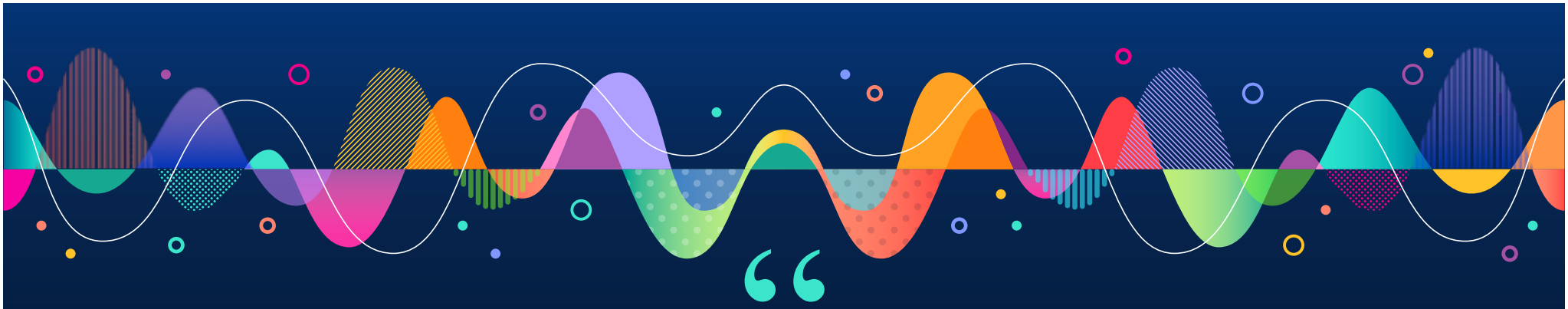


Changing Font and Other Miscellaneous Details

- theme()
- There are more then 90 different options on changing elements of your plot



```r
stm %>%
  group_by(category) %>%
  summarise(mean_winds = mean(wind),
            sd_winds = sd(wind)) %>%
  ungroup() %>%
  ggplot(aes(x = category,
             y = mean_winds,
             fill = category)) +
  geom_col() +
  geom_errorbar(aes(ymin = mean_winds - sd_winds,
                    ymax = mean_winds + sd_winds),
                width = .5,
                size = .75) +
  xlab("Category") +
  ylab("Mean Wind Speed") +
  labs(fill = "Category",
       title = "Mean Wind Speeds for Documented Huricanes, Tropical Storms,\nand Tropical Depressions from 1975 - 2020") +
  theme_classic() +
  theme(panel.grid.major.y = element_line(size = .5,
                                          linetype = "longdash",
                                          colour = "darkgrey"),
        panel.grid.minor.y = element_line(size = .1,
                                          linetype = "dashed",
                                          colour = "lightgrey"),
        axis.title = element_text(size = 16, colour = "black"),
        axis.text = element_text(size = 12, colour = "black"),
        title = element_text(size = 20, colour = "black"),
        legend.position = "none")
```



ggplot2 Assignment 2

Using the provided PSMC data, make a nice plot or enhance the previous plot you made from assignment 1