TOOL

ggplot() Basics

This tool will help you use `ggplot()` to create different types of high-quality data visualizations.

1 Set up your data

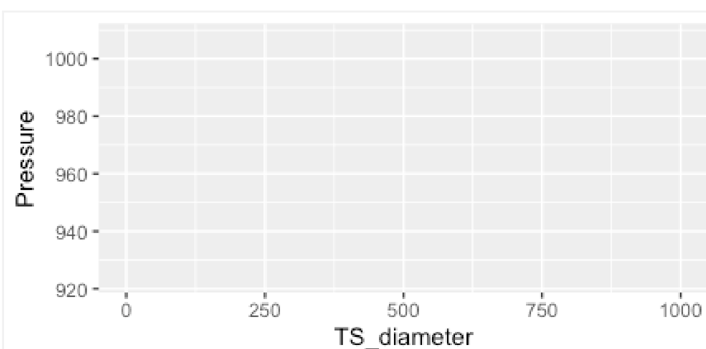
These plots use data from the `storms.csv` file you used during the course. To set that data up for use with this tool, use the following code:

```
library(tidyverse)
storms <- read.csv("storms.csv")
storms$Category <- factor(storms$Category, levels = -1:5) # set the storm
category to be a factor
storms$Date <- factor(storms$Date, levels = unique(storms$Date)) # set the
measurement date/time to be a factor
# create data set with observations only for Hurricanes Katrina, Sandy, and
# Wilma:
sampleStorms <- storms %>% filter(Name %in% c("Katrina", "Sandy", "Wilma"))
sandy <- storms %>% filter(Name == "Sandy") # only use observations for Sandy
```

2 Build the first layer of your plot

Plots made in ggplot are built from layers. The first layer contains the data set and the aesthetic mappings, which include the x and y variables and any aesthetic features that apply to all subsequent layers:

```
ggplot(sampleStorms, aes(x = TS_diameter, y = Pressure))
```

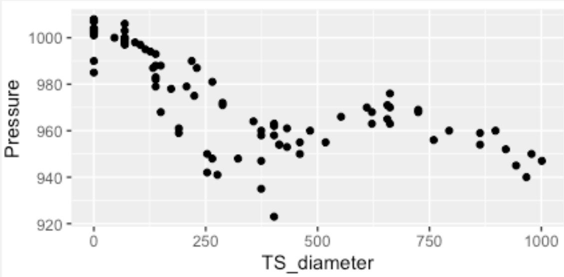


3 Specify plot geometry

To plot data, tell R what kind of plot you want — scatter, bar, or line — by specifying the type of geometry your plot will have.

geom_point()

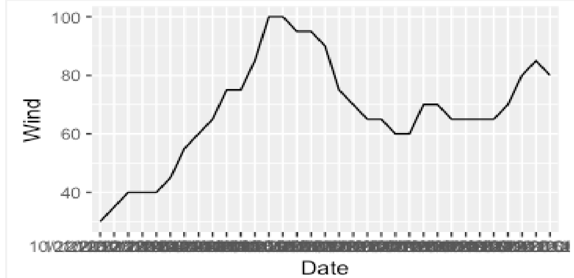
Scatterplot of sampleStorms data




```
ggplot(sampleStorms, aes(x = TS_diameter,
 y = Pressure)) +
  geom_point()
```

geom_line()

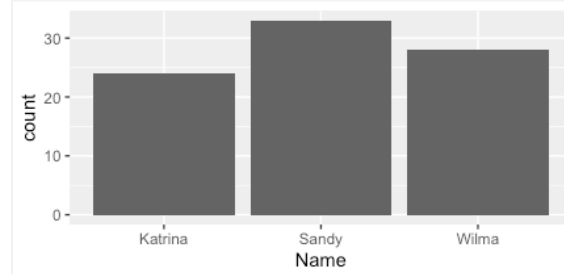
Line plot of sandy data




```
ggplot(sandy, aes(x = Date, y = Wind,
 group = 1)) +
  geom_line()
```

geom_bar()

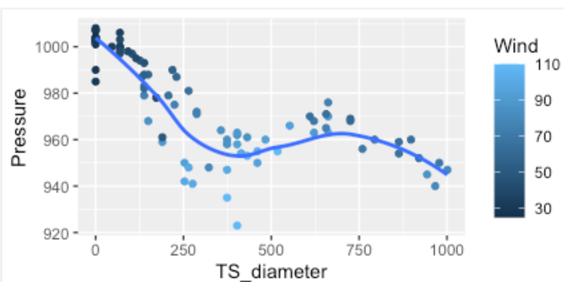
Barplot of sampleStorms data



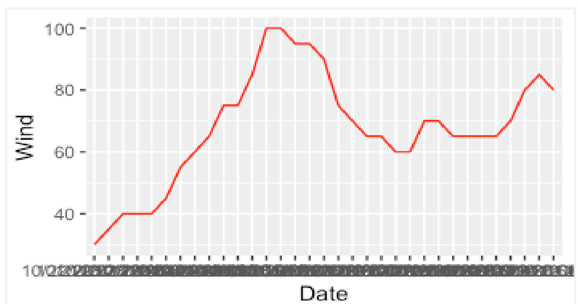

```
ggplot(sampleStorms, aes(x = Name)) +
  geom_bar()
```

4 Adjust plot features

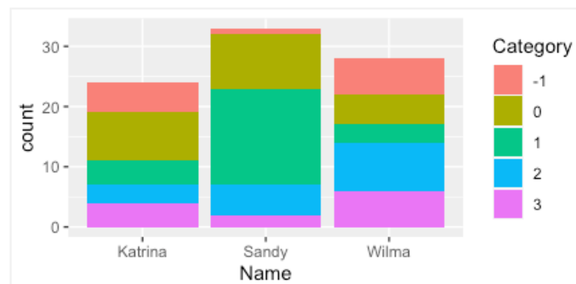
Now that you've set up your basic plot, you can adjust its features. To adjust features like color, position, and size based on the value of a variable, add them within the `aes()` function. To set these features to a fixed value, add them outside of the `aes()` function.




```
ggplot(sampleStorms, aes(x = TS_diameter,
  y = Pressure, color = Wind)) +
   geom_point() +
   geom_smooth(se = FALSE)
```




```
ggplot(sandy, aes(x = Date, y = Wind,
  group = 1)) +
   geom_line(color = "red")
```



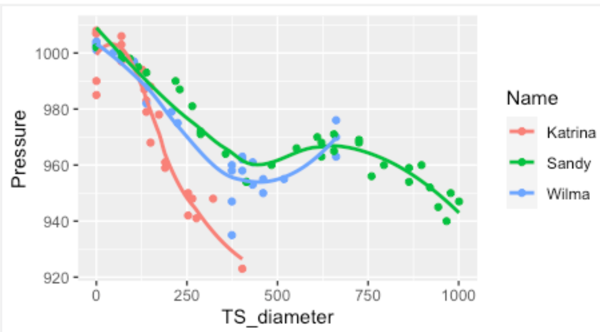

```
ggplot(sampleStorms, aes(x = Name,
 fill = Category)) +
  geom_bar()
```

5 Layer additional features

If there are multiple geometric objects in the same plot, you may want to make aesthetic adjustments to one, several, or all of these geometries. You can control which geometries are changed by selectively adjusting the plot layers:

Adjust all layers

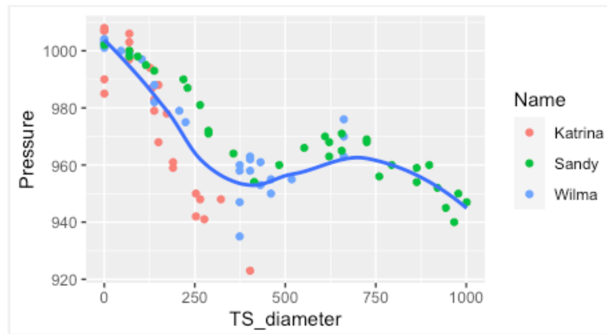
Aesthetic adjustments in the main layer, `ggplot()`, apply to all geometric objects. Here, color is applied to both the points and curves.




```
ggplot(data = sampleStorms, aes(x = TS_
 diameter, y = Pressure, color = Name)) +
  geom_point() +
  geom_smooth(se = FALSE)
```

Adjust some layers

Aesthetic adjustments in one layer apply only to that geometric object. Here, color is applied only to the points.

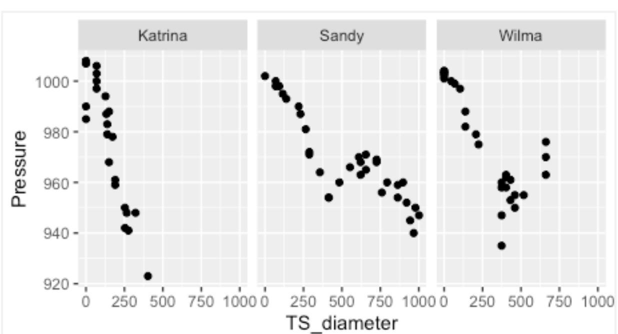



```
ggplot(data = sampleStorms, aes(x = TS_
 diameter, y = Pressure)) +
  geom_point(aes(color = Name)) +
  geom_smooth(se = FALSE)
```

6 Creating groups of plots

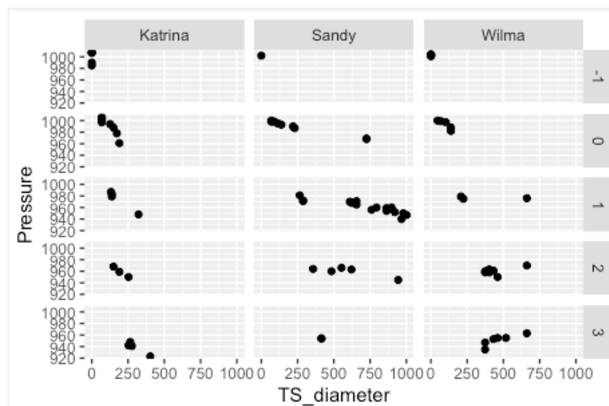
You can create groups of plots, where the groups are based on the values of one or two categorical variables:

`facet_wrap()` creates a group of graphs based on one categorical variable:




```
ggplot(data = sampleStorms, aes(x = TS_
 diameter, y = Pressure)) +
  geom_point() +
  facet_wrap(~ Name)
```

`facet_grid()` creates a group of graphs based on two categorical variables:




```
ggplot(data = sampleStorms, aes(x = TS_
 diameter, y = Pressure)) +
  geom_point() +
  facet_grid(Category ~ Name)
```