Create_Plots

eCornell

10/7/2021

Step 1: Load your data and the R packages.

Load data from the National Oceanic and Atmospheric Administration's Atlantic hurricane database and convert variables to factors where necessary.

```
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.0     v stringr   1.5.1
## v lubridate 1.9.3     v tibble    3.2.1
## v purrr     1.0.2     v tidyr     1.3.1
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become error
```

```
# read in the storm data:
storms <- read.csv("storms.csv")

# set the storm category to be a factor:
storms$Category <- factor(storms$Category, levels = -1:5)

# set the measurement date/time to be a factor:
storms$Date <- factor(storms$Date, levels = unique(storms$Date))
# View(storms) # Look at the storms data
```

Step 2: Set up a blank plotting region.

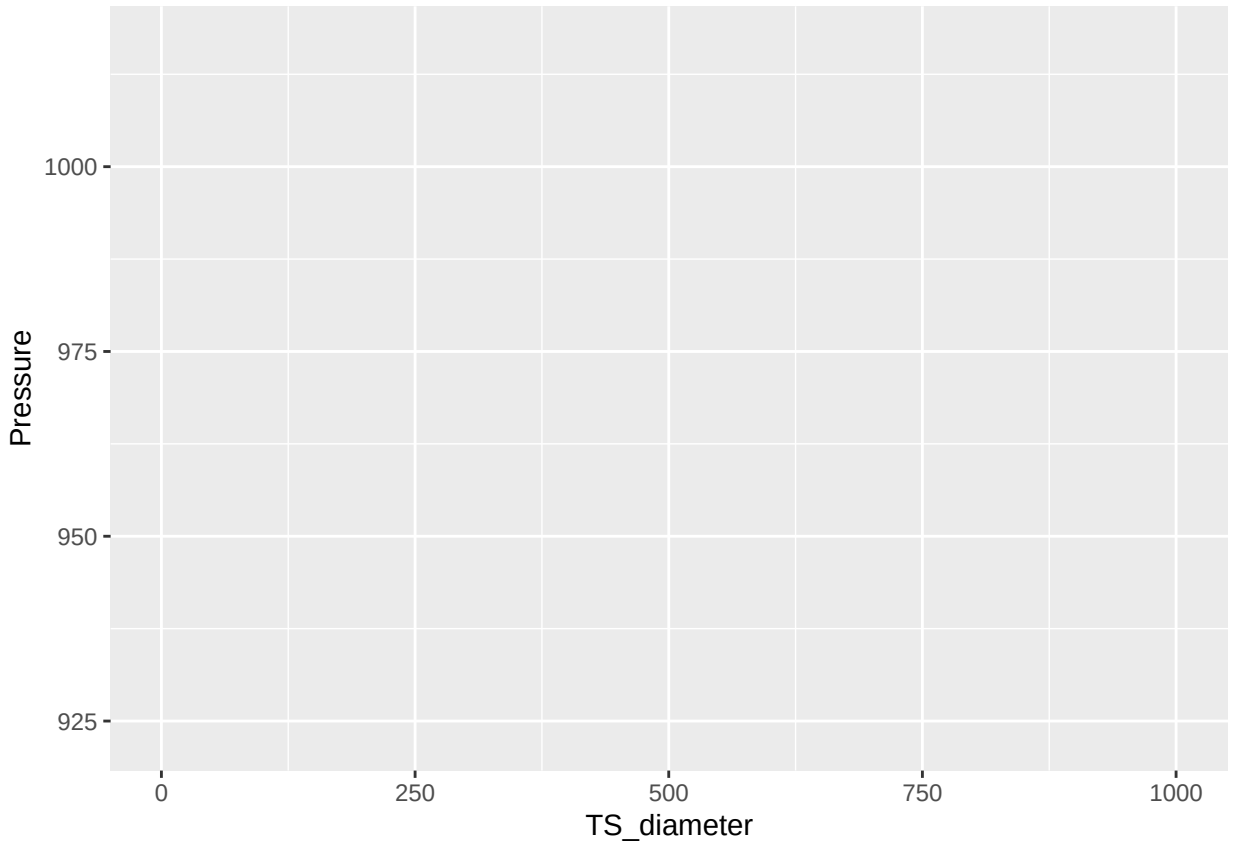
You can run the ggplot() command with no arguments to set up a blank plot:

```
ggplot()
```

Step 3: Specify the data set, x-, and y-axis variables.

The first argument that ggplot() takes is the name of the data set. The second argument is the aes() function. The x- and y-axis variables are specified within the aes() function:

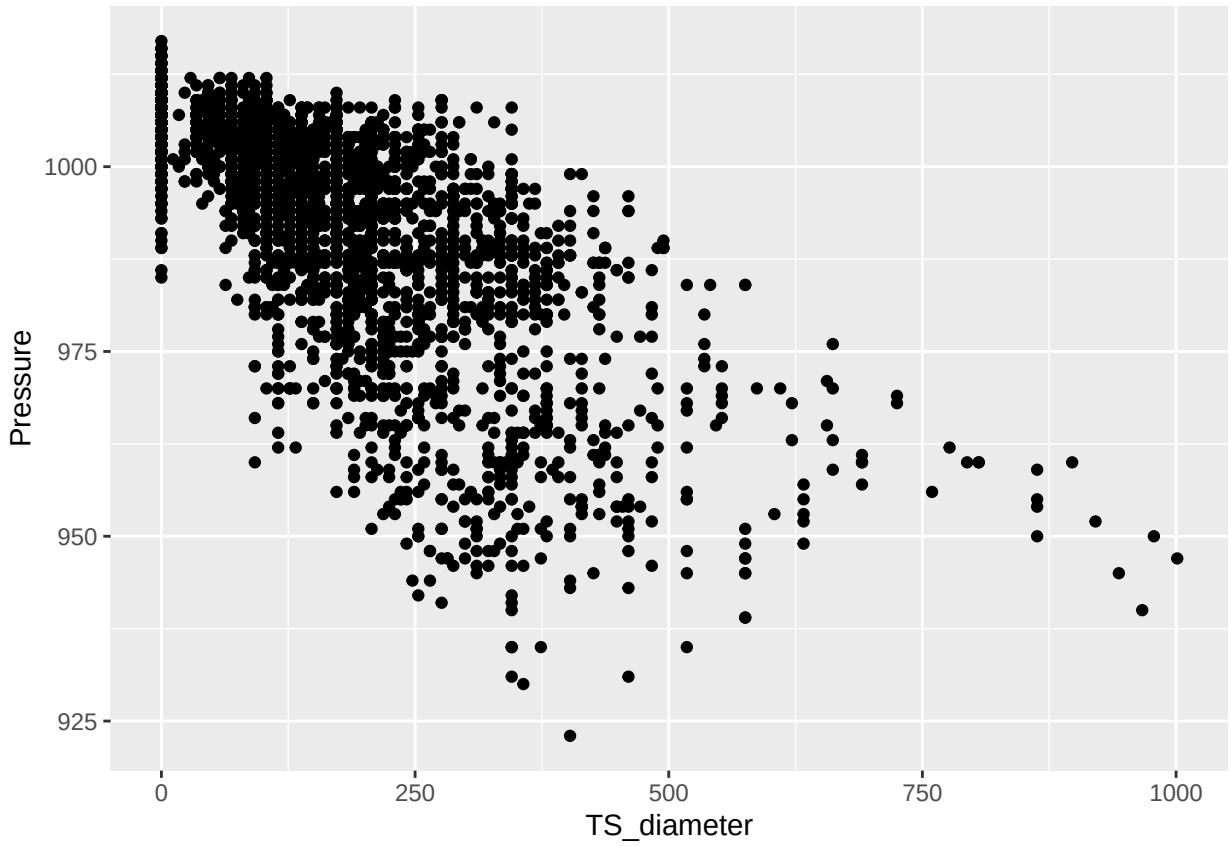
```
ggplot(storms, aes(x = TS_diameter, y = Pressure))
```



Step 4: Add a geometric object.

You're creating a scatterplot, so you want to add a layer of points on top of your labeled plotting region. You can do so by adding the function geom_point() to the ggplot() command:

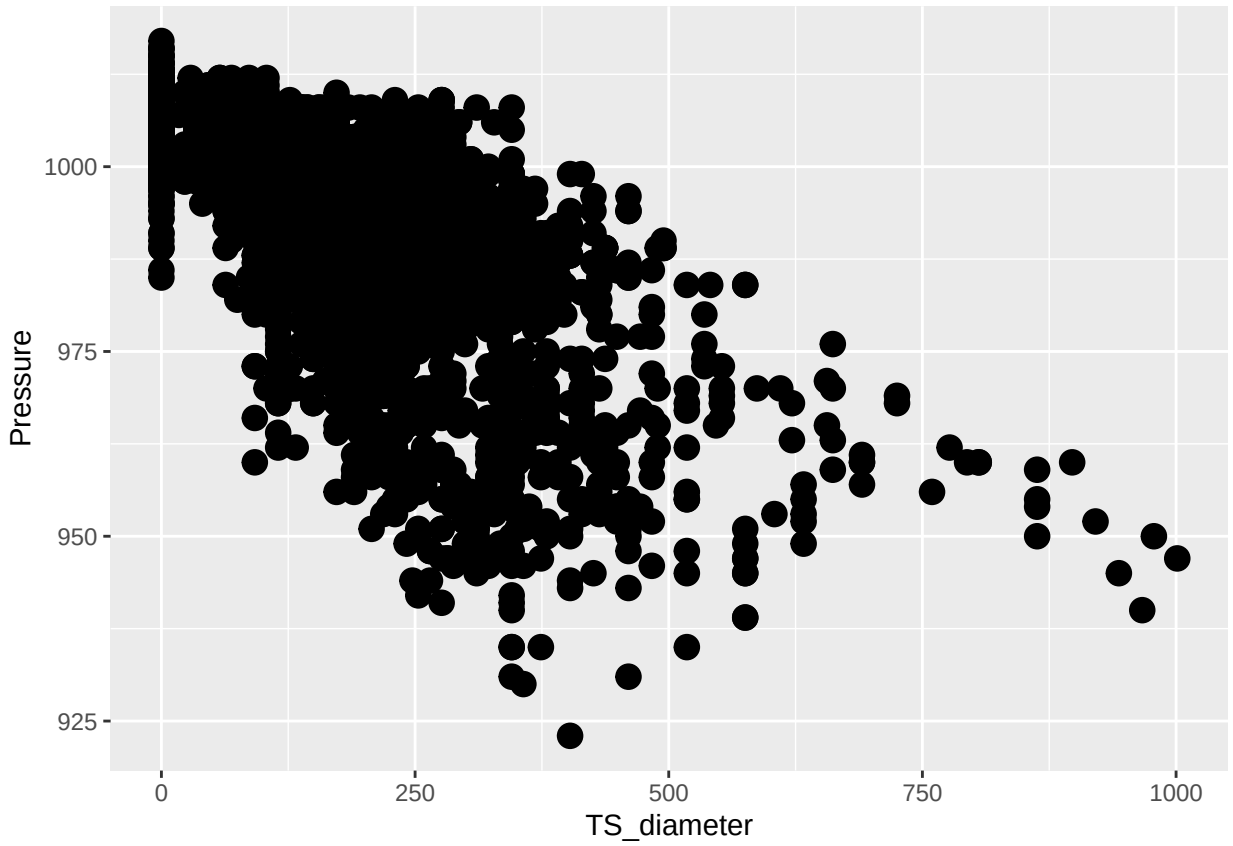
```
ggplot(storms, aes(x = TS_diameter, y = Pressure))+
   geom_point()
```



Step 5: Increase the size of the geometric object.

If you want to make the points on your plot bigger, you can do so by adding an argument that specifies size within the geom_point() function:

```
ggplot(storms, aes(x = TS_diameter, y = Pressure))+
   geom_point(size = 4)
```



Step 6: Create and add the theme to your plot.

Create a theme:

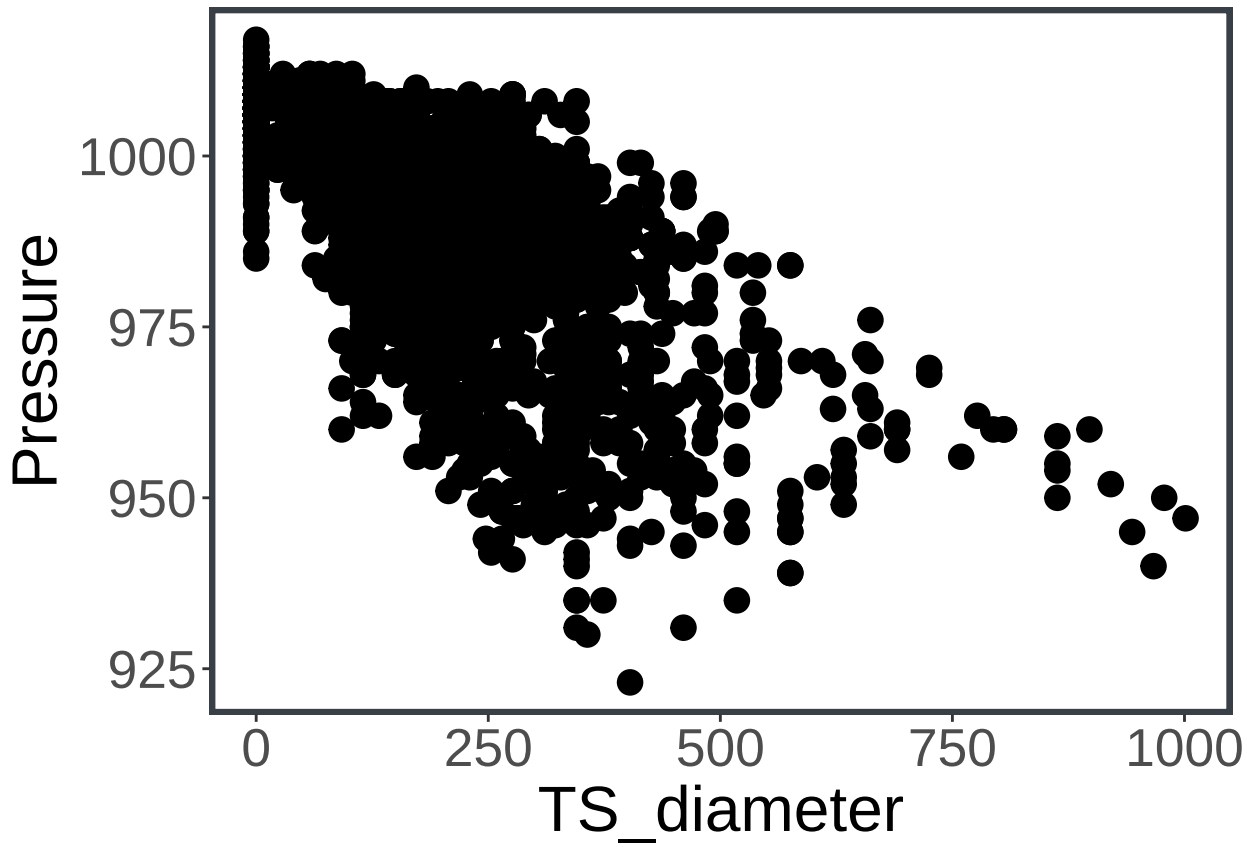
```
theme <-  theme(plot.margin = margin(5, 5, 5, 5, "pt"),
          panel.grid.major = element_blank(),
          panel.grid.minor = element_blank(),
          panel.background = element_blank(),
          panel.border = element_rect(colour = "#393f47", fill = NA, size = 2),
          axis.text = element_text(size = 20),
          axis.title.x = element_text(size = 24),
          axis.title.y = element_text(size = 24),
          plot.title = element_text(face = "bold", size = 30))
```

```
## Warning: The ‘size‘ argument of ‘element_rect()‘ is deprecated as of ggplot2 3.4.0.
## i Please use the ‘linewidth‘ argument instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
ourTheme <- list(theme, scale_color_manual(values = c('#393f47', '#b31b1b',
       '#fbb040', '#92b2c4')))
```

You can then add a theme to your plot by including a plus sign and the name of your theme. In this case, the theme is called ourTheme:

```
ggplot(storms, aes(x = TS_diameter, y = Pressure))+
  geom_point(size = 4) +
  ourTheme
```


Pressure
1000
975
950
925
0
250
500
750
1000
TS_diameter