

Class05: Data Visualization using GGPLOT

Gabriel Ulfohn

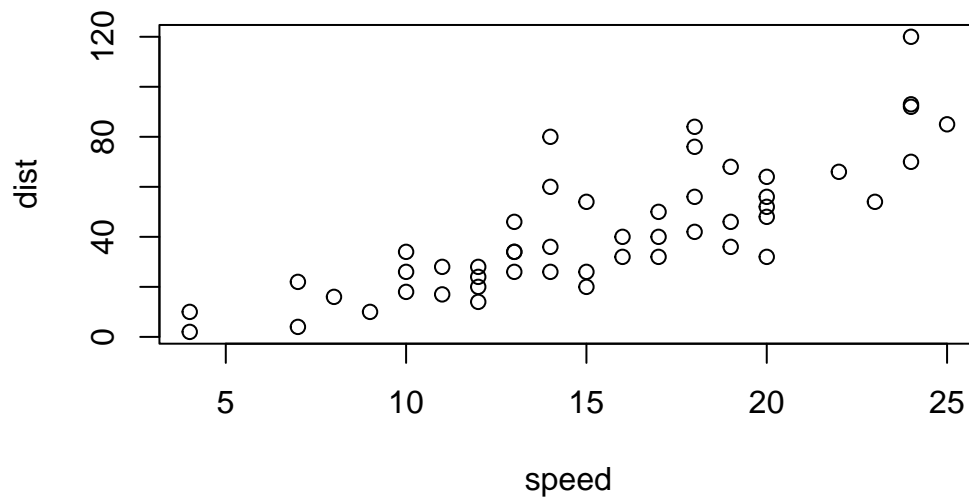
4/19/23



Base R plotting

Generating plot of class 05. Code plots cars dataset

```
plot(cars)
```



Ggplot2

first, we need to install ggplot2, using `install.packages` command

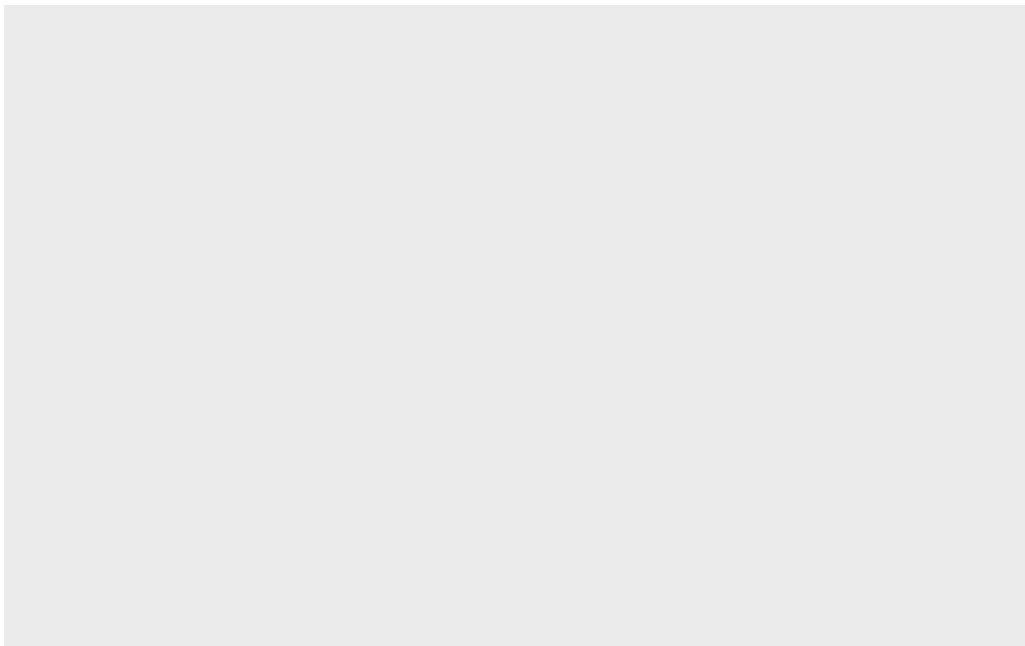
```
# install.packages("ggplot2")
```

after we must load the package

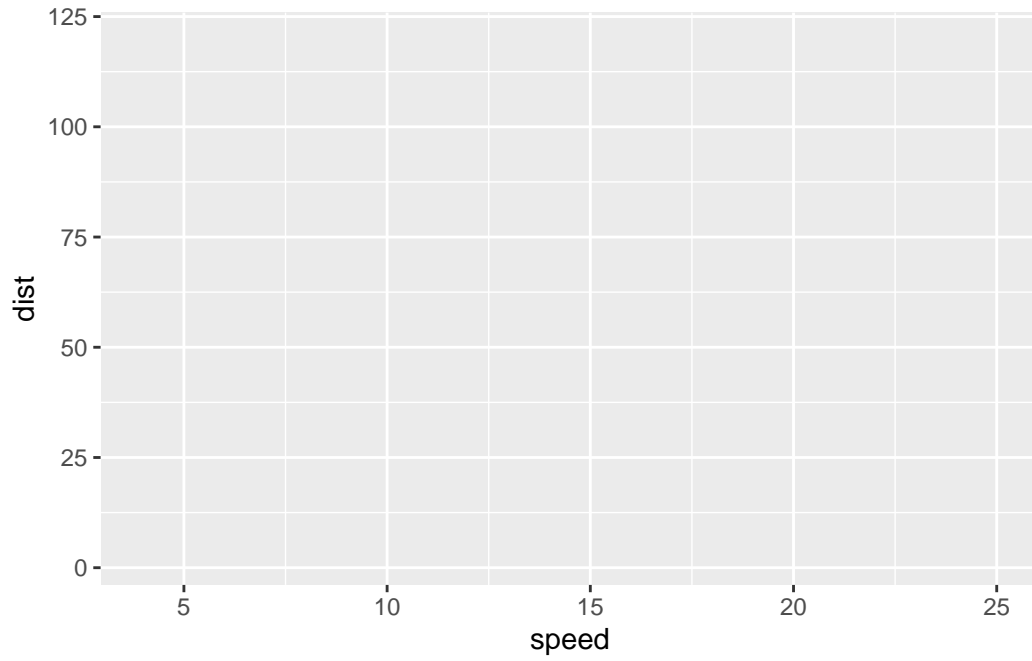
```
library(ggplot2)
```

we then must set the data to be related to cars, and setting all of our plot condition using `aes()` and `geom_point()`

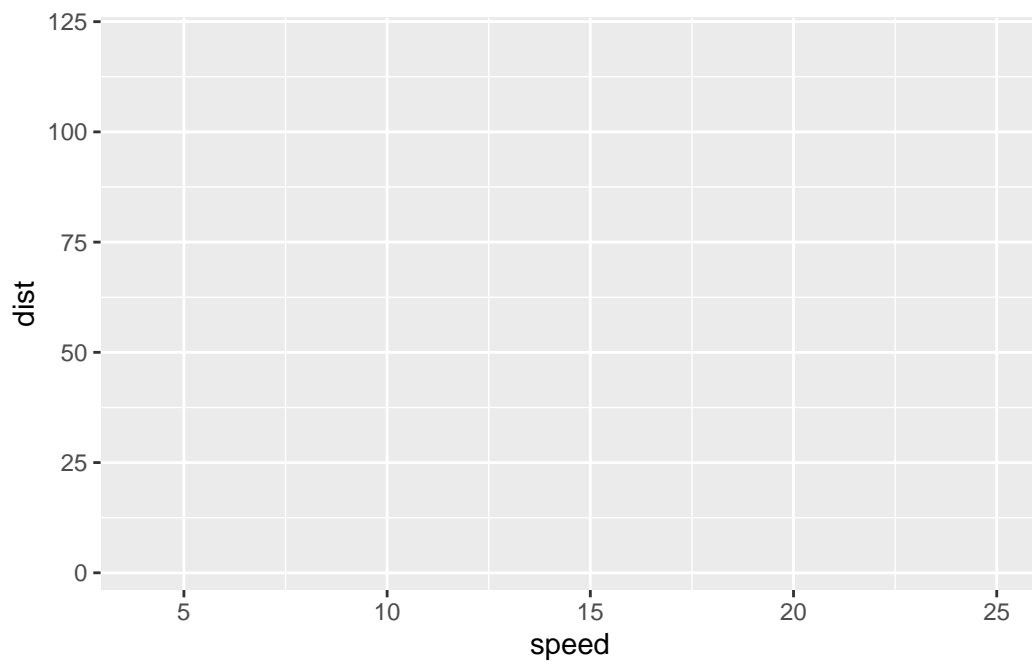
```
ggplot(data = cars)
```



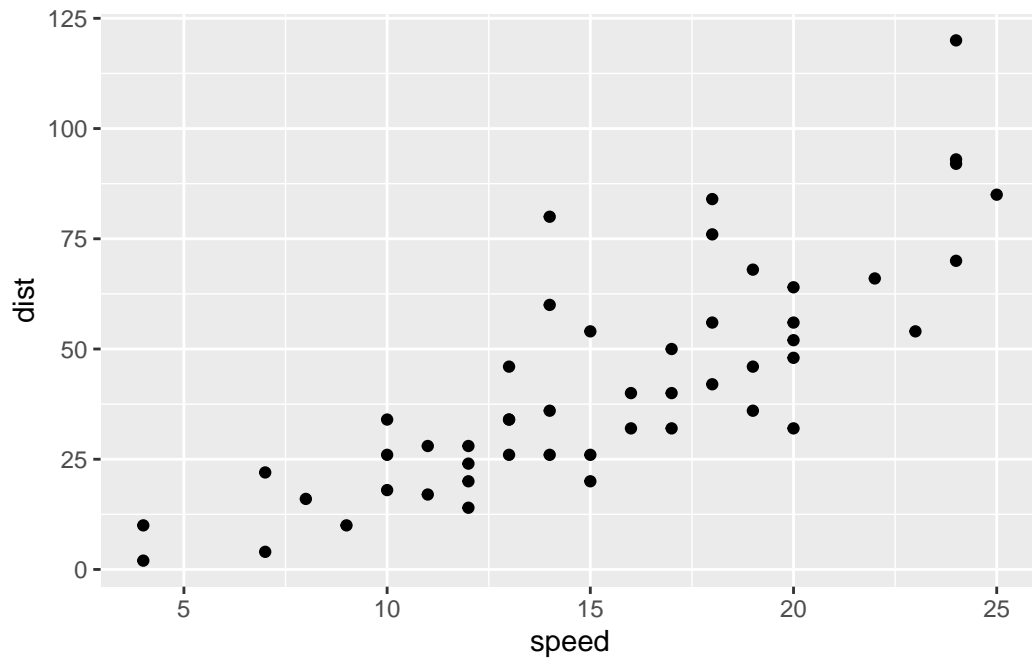
```
ggplot(data = cars) + aes(x=speed, y=dist)
```



```
ggplot(data = cars) + aes(x=speed, y=dist)
```

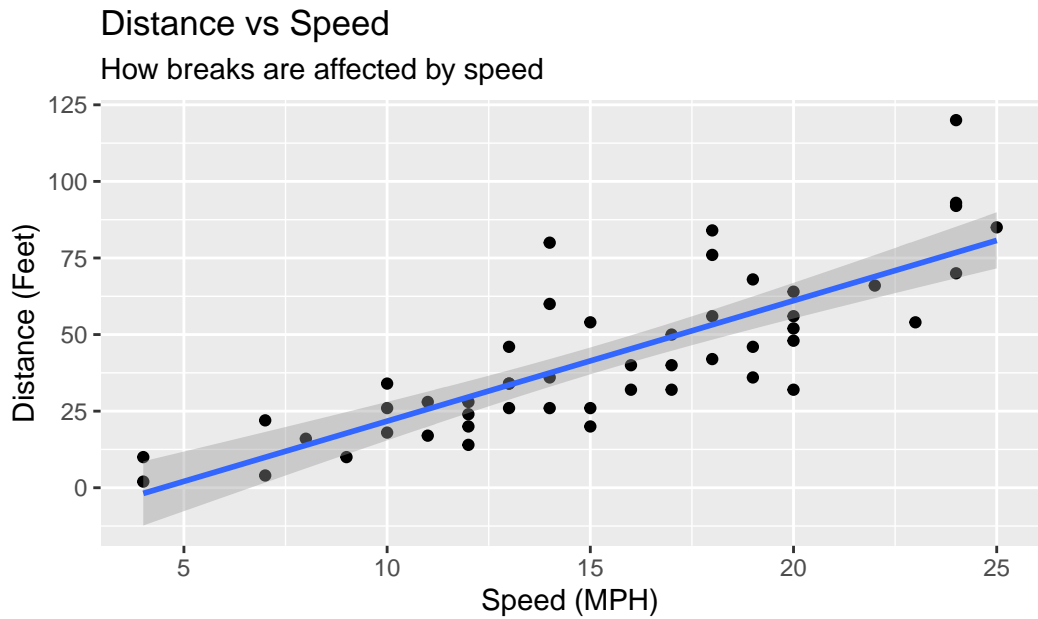


```
ggplot(data = cars) + aes(x=speed, y=dist) + geom_point()
```



```
ggplot(data = cars) + aes(x=speed, y=dist) + geom_point() + geom_smooth(method = "lm") + 1
```

`geom_smooth()` using formula = 'y ~ x'



BIMM143

Plotting gene expression data

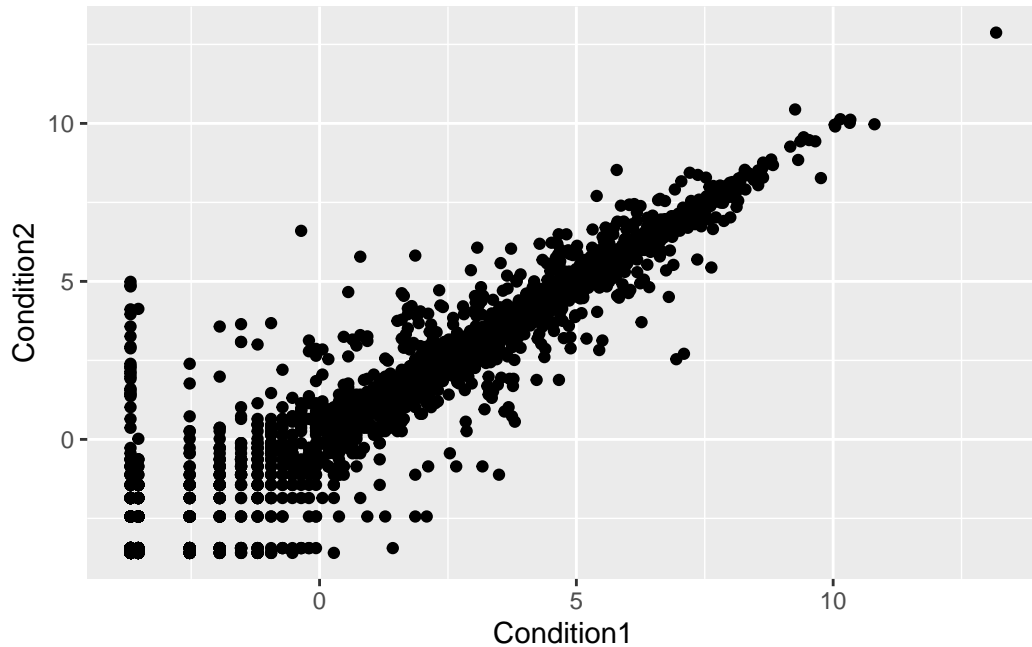
loading the data from the Url

```
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)
head(genes)
```

	Gene	Condition1	Condition2	State
1	A4GNT	-3.6808610	-3.4401355	unchanging
2	AAAS	4.5479580	4.3864126	unchanging
3	AASDH	3.7190695	3.4787276	unchanging
4	AATF	5.0784720	5.0151916	unchanging
5	AATK	0.4711421	0.5598642	unchanging
6	AB015752.4	-3.6808610	-3.5921390	unchanging

Initial ggplot

```
p1 <- ggplot(data = genes) + aes(x = Condition1, y = Condition2) + geom_point()
p1
```



lets change color

For which phases is data visualization important in our scientific workflows?

data visualization is very important overall, its important for the communication of the results, the detection of outliers, and Exploratory Data Analysis

True or False? The ggplot2 package comes already installed with R?

The ggplot2 package needs to be installed onto Rstudio and does come on the program, not only that but we must tell R to load that package.

Use the `nrow()` function to find out how many genes are in this dataset. What is your answer?

`nrow(Genes)`, and it allows me to say there are 5,196 genes in the data set

Use the `colnames()` function and the `ncol()` function on the `genes` data frame to find out what the column names are (we will need these later) and how many columns there are. How many columns did you find?

`Colnames(nol(genes))` There are 4 columns in the data set, Gene, Condition1, Condition2, and State

Use the `table()` function on the `State` column of this `data.frame` to find out how many 'up' regulated genes there are. What is your answer?

```
table(genes[,"State"])
```

Using your values above and 2 significant figures. What fraction of total genes is up-regulated in this dataset?

```
(table(genes[, "State"]) / nrow(genes)) * 100
```