

Class 05 : Data Visualization with ggplot2

Sindy

Table of contents

A more interesting plot 5

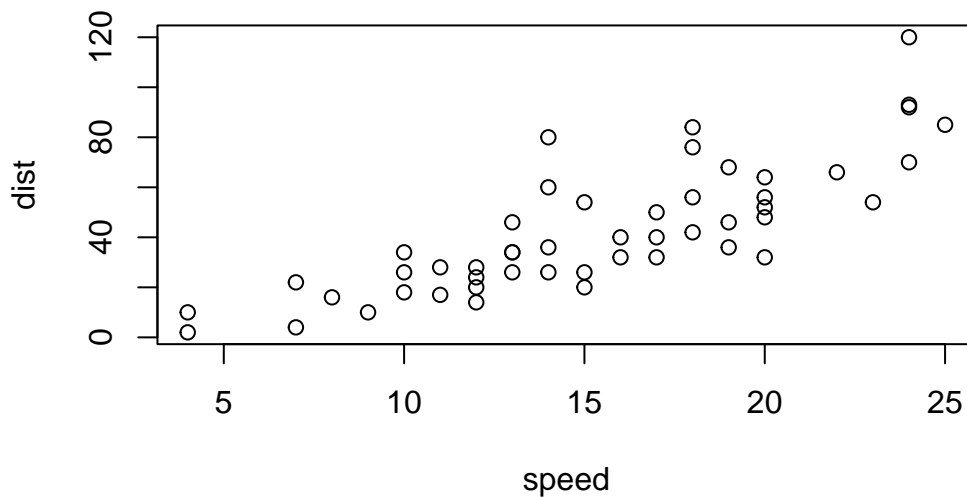
To submit - change line 4 format to pdf from html, then to hide chunks of r script that are not running correctly type in the thing below within the r code chunk `#!/ eval:false` # Our first plot

R has base graphics

```
head(cars)
```

```
  speed dist
1     4    2
2     4   10
3     7    4
4     7   22
5     8   16
6     9   10
```

```
plot(cars)
```



How would I plot this with 'ggplot2'? NO!

We need to install and load the ggplot2 package first. Type in **install.packages()**

Put the name of the package inside the parenthesis and use quotes

After installing, you need to load the package each time you want to use it. Use the function **library()**. You only need to load the packages that you will be using for that project. If R automatically loaded all the packages available when it opened, you'd slow your computer down. You need to load the library here in the source, as well as in the console (R brain). This way it will be saved into the environment of this particular project.

Once loaded, you can start plotting (with a few more steps)!

Every ggplot2 needs at least 3 layers

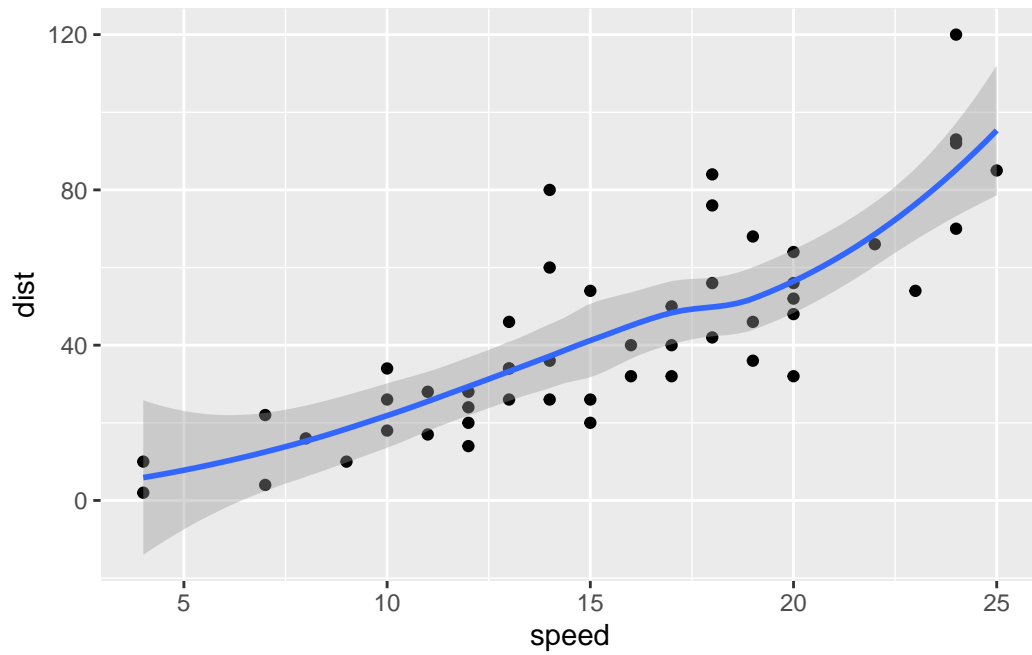
- **Data** (i.e the data.frame we have)
- **Aes** (the aesthetic mapping of our data to what we want to plot)
- **Geoms** (how we want to plot this stuff)

```
library(ggplot2)
ggplot(data=cars) + aes(x=speed, y=dist) + geom_point()
```



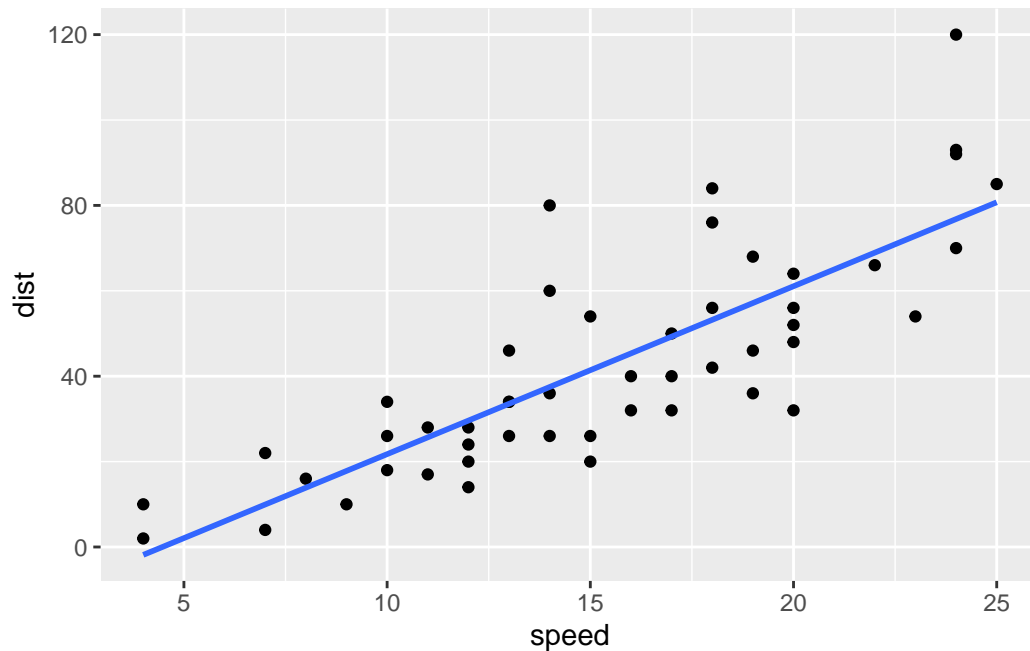
```
ggplot(data=cars) + aes(x=speed, y=dist) + geom_point() + geom_smooth()
```

``geom_smooth()`` using `method = 'loess'` and formula `'y ~ x'`



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

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



A more interesting plot

First read data from online

```
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

```
nrow(genes)
```

```
[1] 5196
```

There are 5196 genes in this data set.

Using `r nrow(genes)` allows you to insert a line of r (or any other program) code

The columns are named Gene, Condition1, Condition2, State and there are 4 of them

```
table(genes[,4])[3]
```

```
up  
127
```

```
table(genes$State)
```

```
down  unchanging      up  
   72      4997      127
```

```
nrow(genes)
```

```
[1] 5196
```

```
table(genes$State)/nrow(genes)
```

```
down  unchanging      up  
0.01385681 0.96170131 0.02444188
```

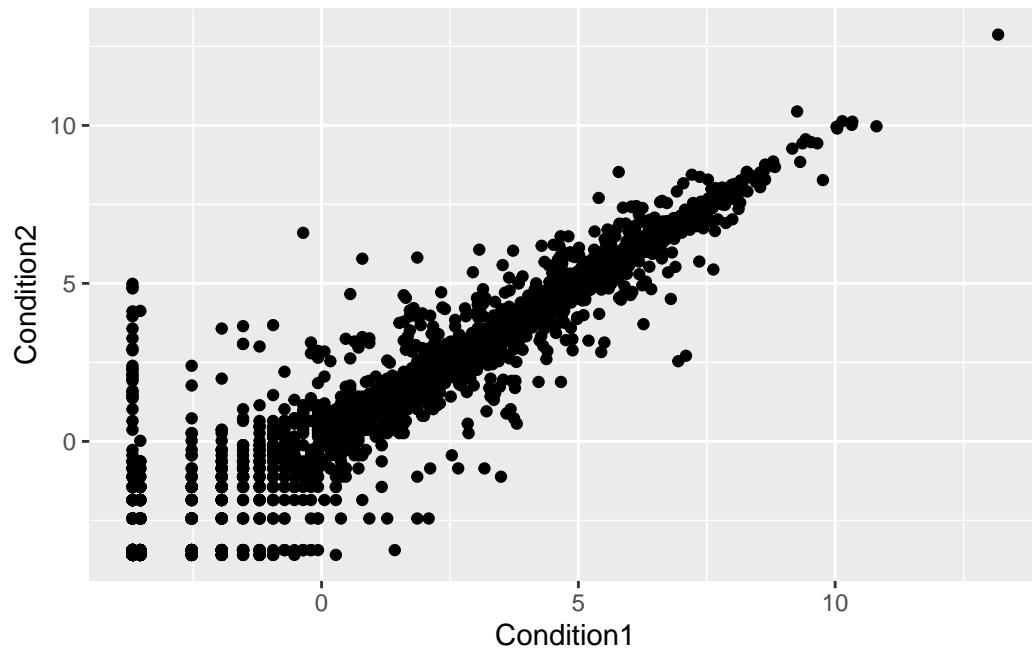
```
round(table(genes$State)/nrow(genes) *100, 2)
```

```
down  unchanging      up  
1.39      96.17      2.44
```

There are 127 ‘up’ regulated genes

(Extra Note:plotly is a package where you can hover over a point and see the gene)

```
ggplot(genes) +  
  aes(x=Condition1, y=Condition2) +  
  geom_point()
```



```
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  geom_point()
p + scale_colour_manual( values=c("blue","gray","red")) +
  labs(title="Gene Expression Changes Upon Drug Treatment",
       x="Control (no drug) ",
       y="Drug Treatment")
```

Gene Expression Changes Upon Drug Treatment

