

# Class 5: Data Viz with ggplot

AE Pasquinelli

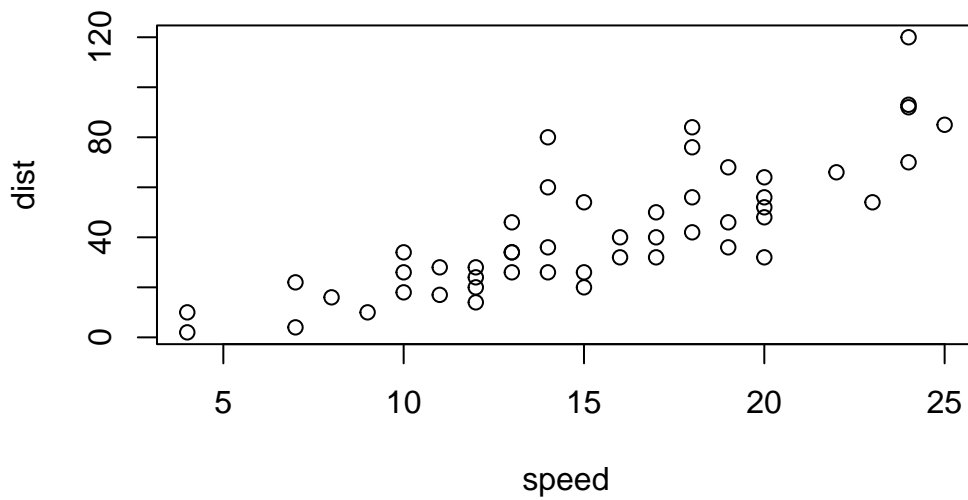
R has lots of ways to make figures and graphs in particular. One that comes with R out of the box is called “**base**” **R** - the `plot()` function.

```
cars
```

	speed	dist
1	4	2
2	4	10
3	7	4
4	7	22
5	8	16
6	9	10
7	10	18
8	10	26
9	10	34
10	11	17
11	11	28
12	12	14
13	12	20
14	12	24
15	12	28
16	13	26
17	13	34
18	13	34
19	13	46
20	14	26
21	14	36
22	14	60
23	14	80
24	15	20
25	15	26
26	15	54

27	16	32
28	16	40
29	17	32
30	17	40
31	17	50
32	18	42
33	18	56
34	18	76
35	18	84
36	19	36
37	19	46
38	19	68
39	20	32
40	20	48
41	20	52
42	20	56
43	20	64
44	22	66
45	23	54
46	24	70
47	24	92
48	24	93
49	24	120
50	25	85

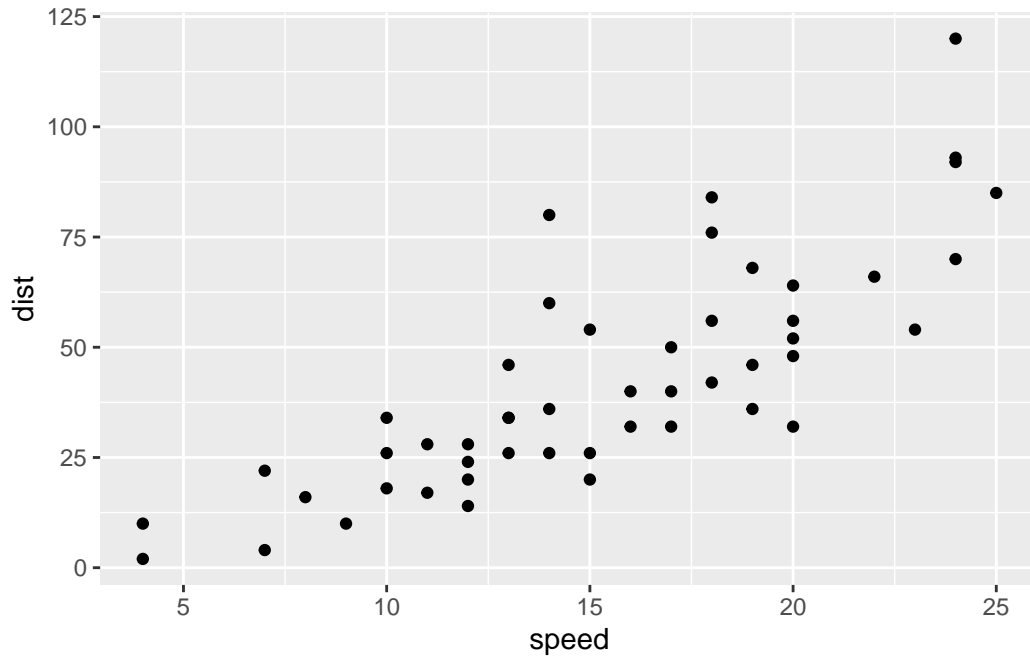
```
plot(cars)
```



A very popular package in this area is called **ggplot2**. Before I can use any add on package like this I must install it with the `install.packages("ggplot")` command/function

Then use the package I need to load it with a `library(ggplot2)` call.

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

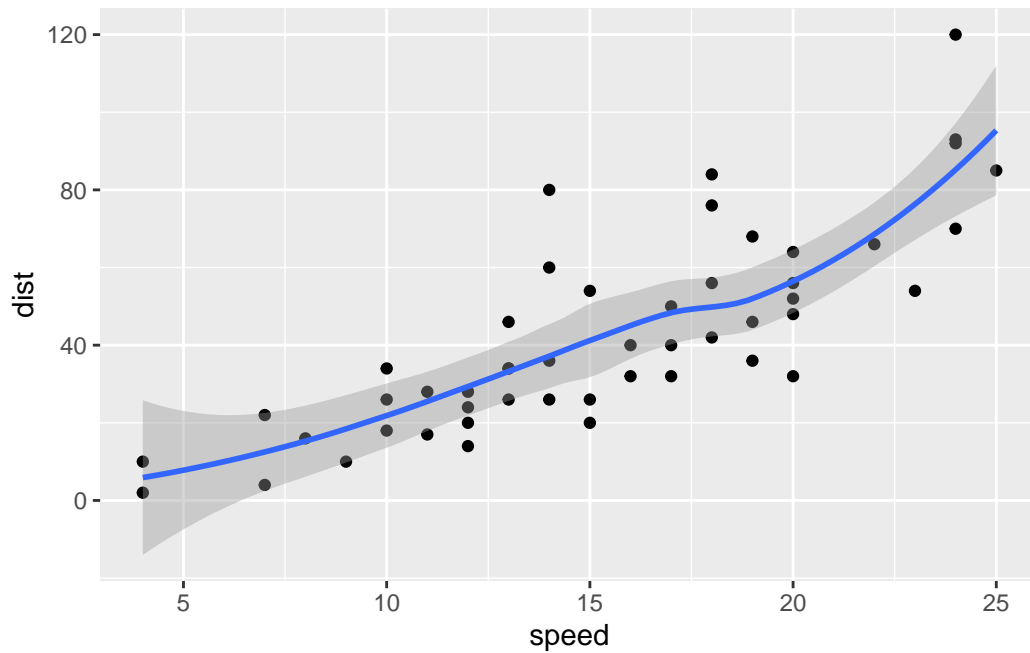


Every ggplot has at least 3 layers - **data** (data.frame with the numbers or stuff you want to plot) - **aesthetics** (mapping of your data columns to your plot) - **geoms** (there are tons of these, basics are `geom_point()`, `geom_line()`, `geom_col()`)

For “simple” plots like this one base R code will be much shorter than ggplot code. Lets fit a line and show it on plot

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

`geom\_smooth()` using method = 'loess' and formula = 'y ~ x'

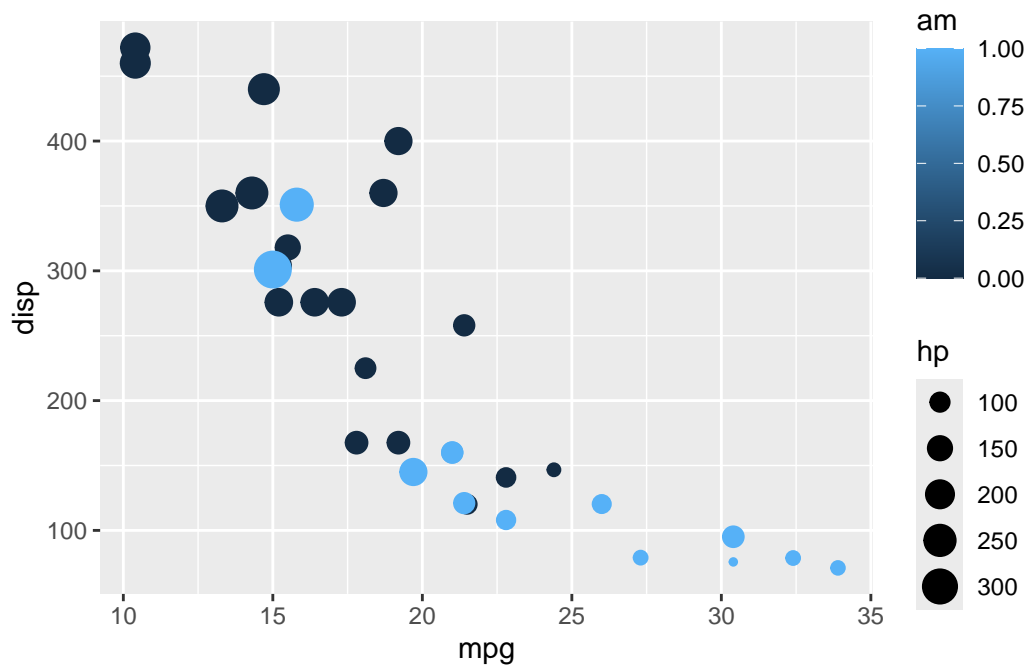


```
head(mtcars)
```

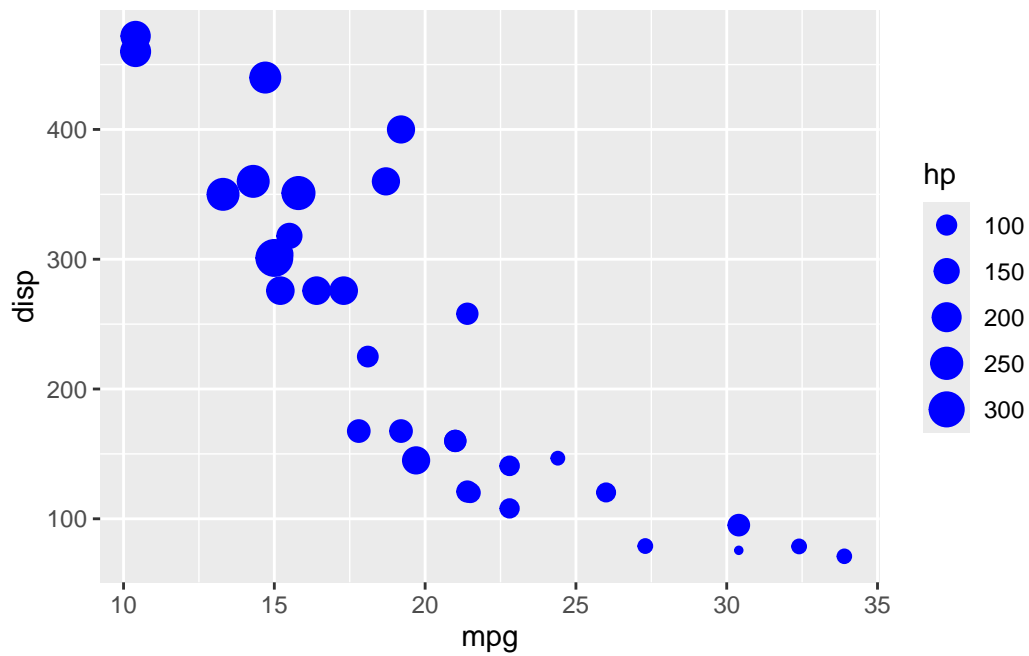
	mpg	cyl	disp	hp	drat	wt	qsec	vs	am	gear	carb
Mazda RX4	21.0	6	160	110	3.90	2.620	16.46	0	1	4	4
Mazda RX4 Wag	21.0	6	160	110	3.90	2.875	17.02	0	1	4	4
Datsun 710	22.8	4	108	93	3.85	2.320	18.61	1	1	4	1
Hornet 4 Drive	21.4	6	258	110	3.08	3.215	19.44	1	0	3	1
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Make me a ggplot of the mtcars data set using mpg vs disp and set the size of the points to the hp.

```
ggplot(mtcars) + aes(x=mpg, y=disp, col=am, size=hp) + geom_point()
```



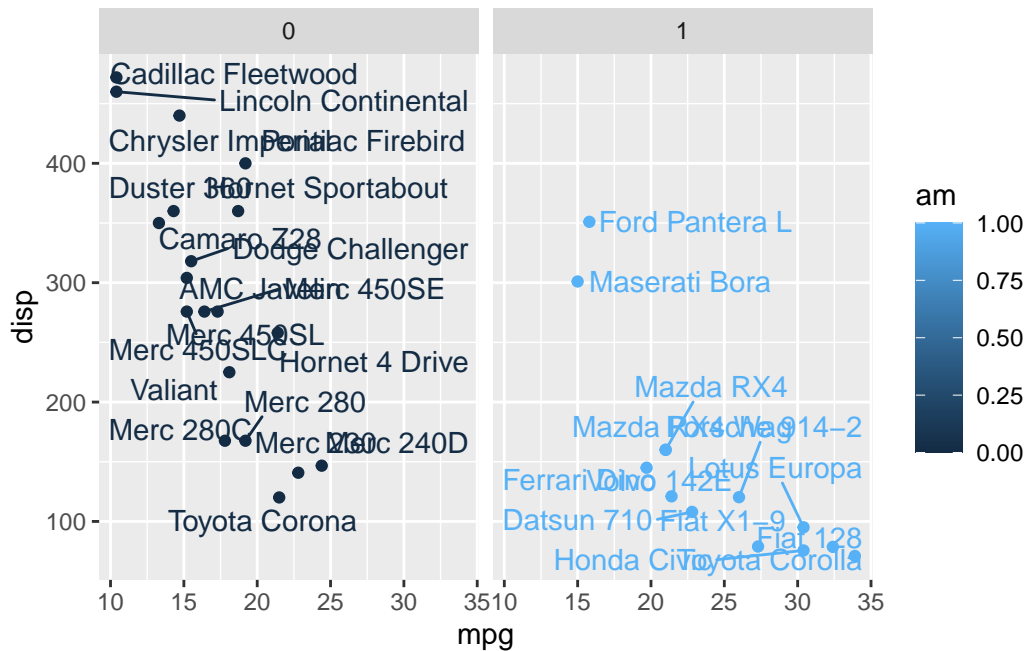
```
ggplot(mtcars) + aes(x=mpg, y=disp, size=hp) + geom_point(col="blue")
```



```
#comment ggplot(mtcars) + aes(x=mpg, y=disp, col=am, labels=rownames, size=hp) +  
geom_point() + facet_wrap(~am)
```

```
library(ggrepel)
```

```
ggplot(mtcars) + aes(x=mpg, y=disp, col=am, label=rownames(mtcars)) + geom_point() + facet_w
```



```
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

## Gene expression list

```
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 `nrow(genes)` genes in this dataset.

```
unique(genes$State)
```

```
[1] "unchanging" "up"          "down"
```

The `table` function is a super useful utility to tell me how many entries of each type there are

```
table(genes$State) / nrow(genes) * 100
```

down	unchanging	up
1.385681	96.170131	2.444188

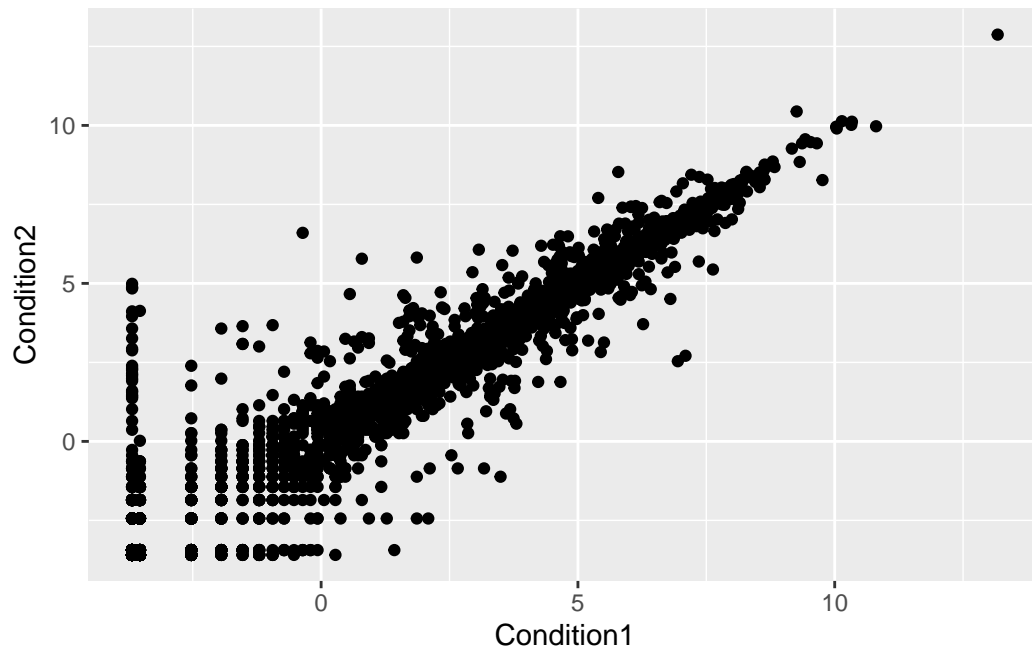
```
round(table(genes$State) / nrow(genes), 3)
```

down	unchanging	up
0.014	0.962	0.024

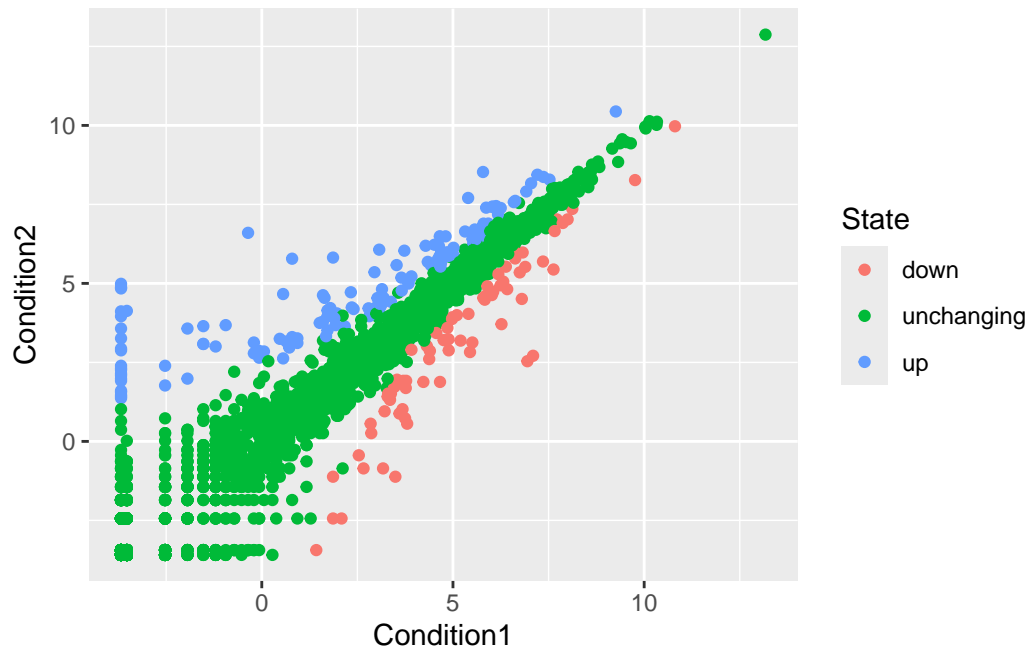
The functions `nrow()`, `ncol()`, and `table()` are ones I want you to know

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

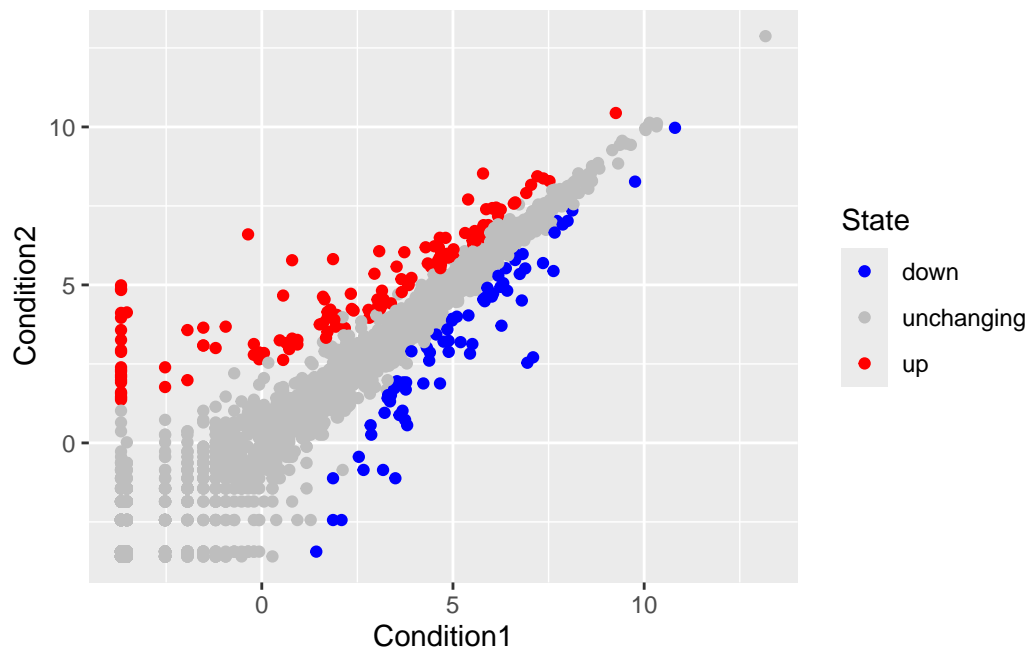




```
p <- ggplot(genes) +  
  aes(x=Condition1, y=Condition2, col=State) +  
  geom_point()  
p
```

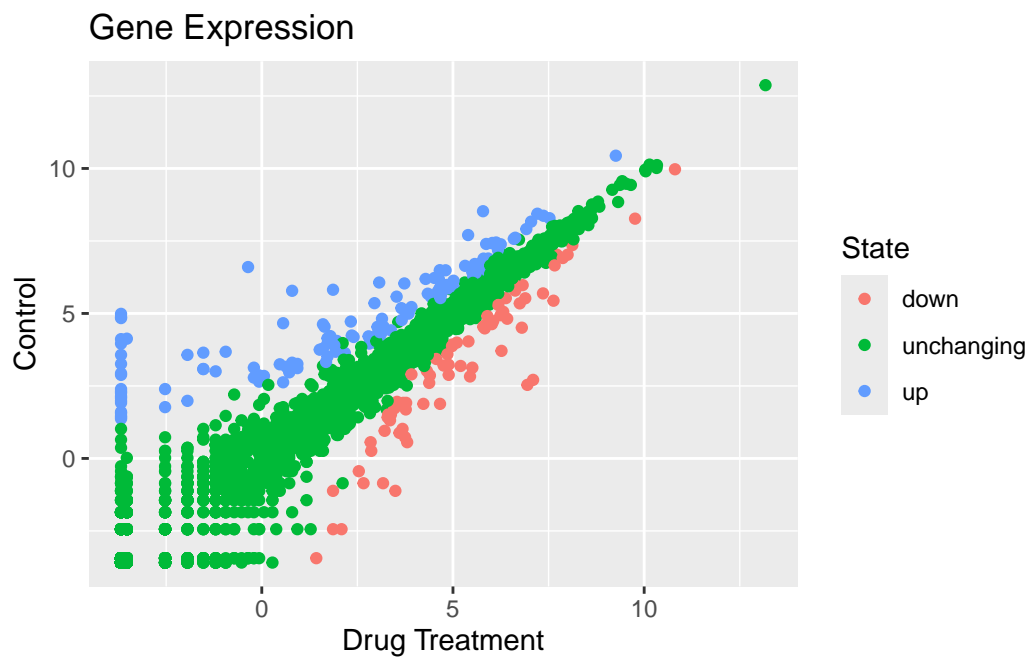


```
p + scale_colour_manual( values=c("blue","gray","red") )
```

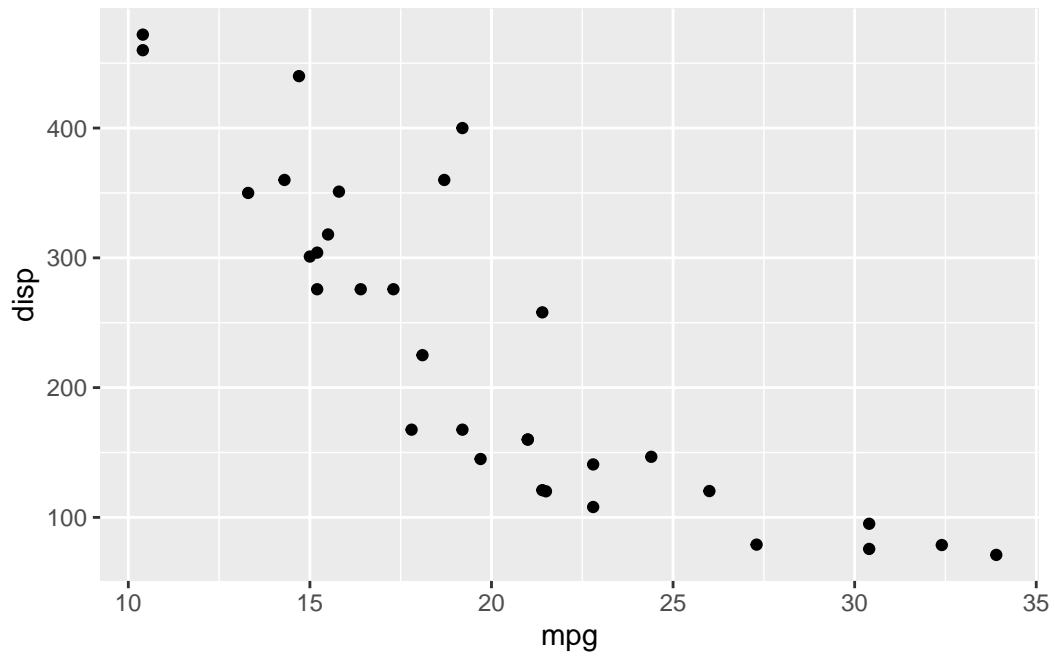


```
#labs(x = "Drug Treatment", y = "Control (no drug)")
p <- ggplot(genes) +
  aes(x=Condition1, y=Condition2, col=State) +
  labs(x="Drug Treatment",
       y="Control",
       title="Gene Expression") +
  geom_point()
```

p



```
ggplot(mtcars) + aes(mpg, disp) + geom_point()
```



```
ggsave("myplot.pdf")
```

Saving 5.5 x 3.5 in image

Key points

Multi-plot