

Class 05 Data Visualization

Monica Lin (PID: A15524235)

2021-10-12

```
# Anything you type in this file must be run in the console below  
# The code typed here will not auto-run  
  
# Lets start with a scatterplot  
cars
```

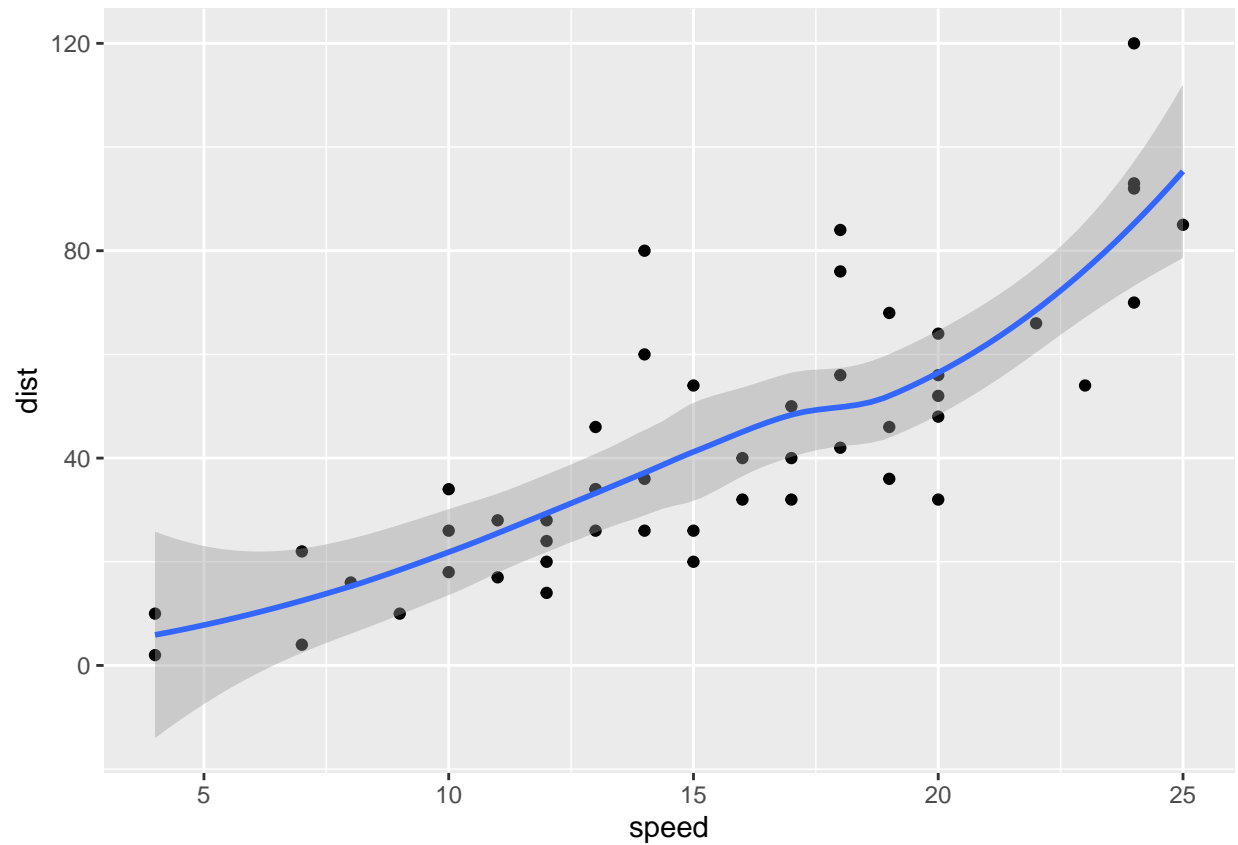
```
##      speed dist  
## 1      4      2  
## 2      4     10  
## 3      7      4  
## 4      7     22  
## 5      8     16  
## 6      9     10  
## 7     10     18  
## 8     10     26  
## 9     10     34  
## 10     11     17  
## 11     11     28  
## 12     12     14  
## 13     12     20  
## 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
```

```
# We have already installed ggplot, but if not, type the following line:
# install.packages("ggplot2")
# Before we can use it, we need to load it up!
library(ggplot2)

# Every ggplot has a data + aes + geoms layer
ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth()
```

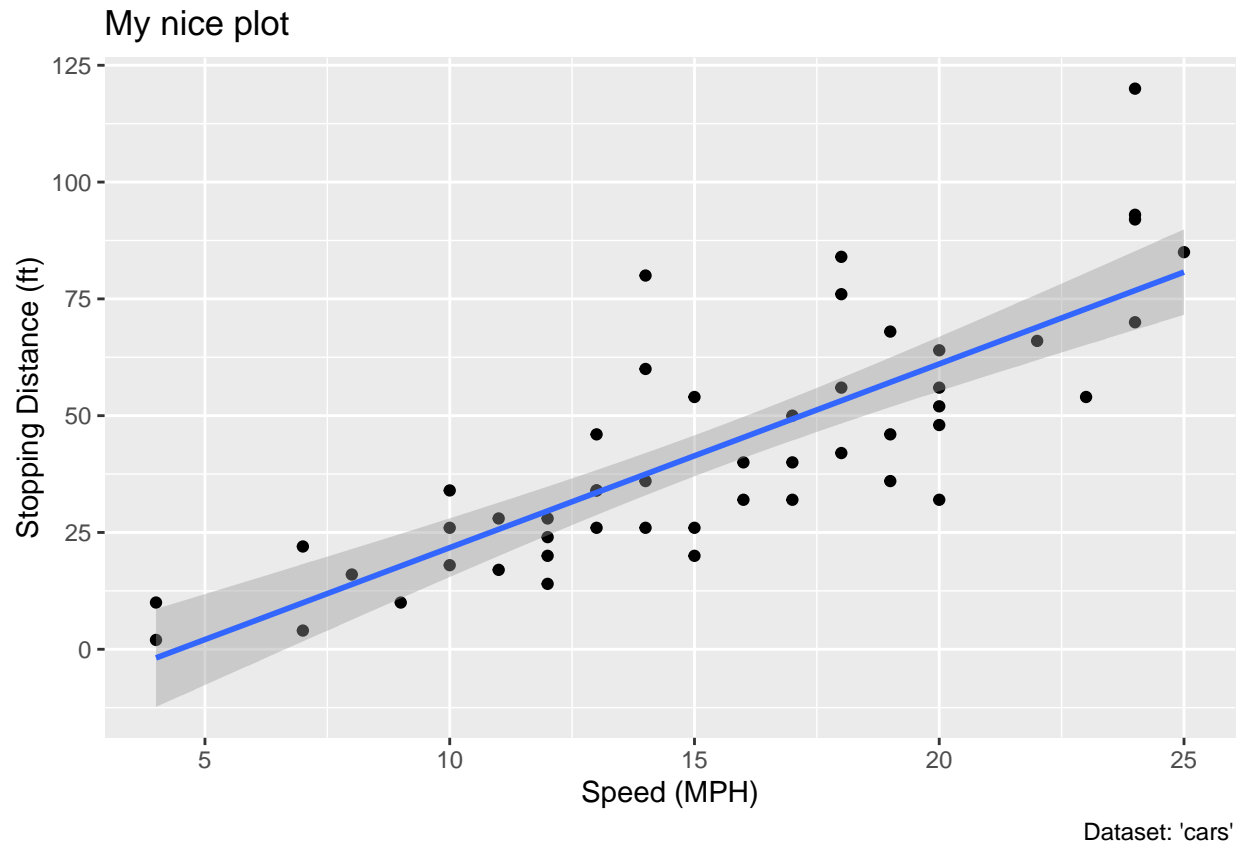
```
## 'geom_smooth()' using method = 'loess' and formula 'y ~ x'
```



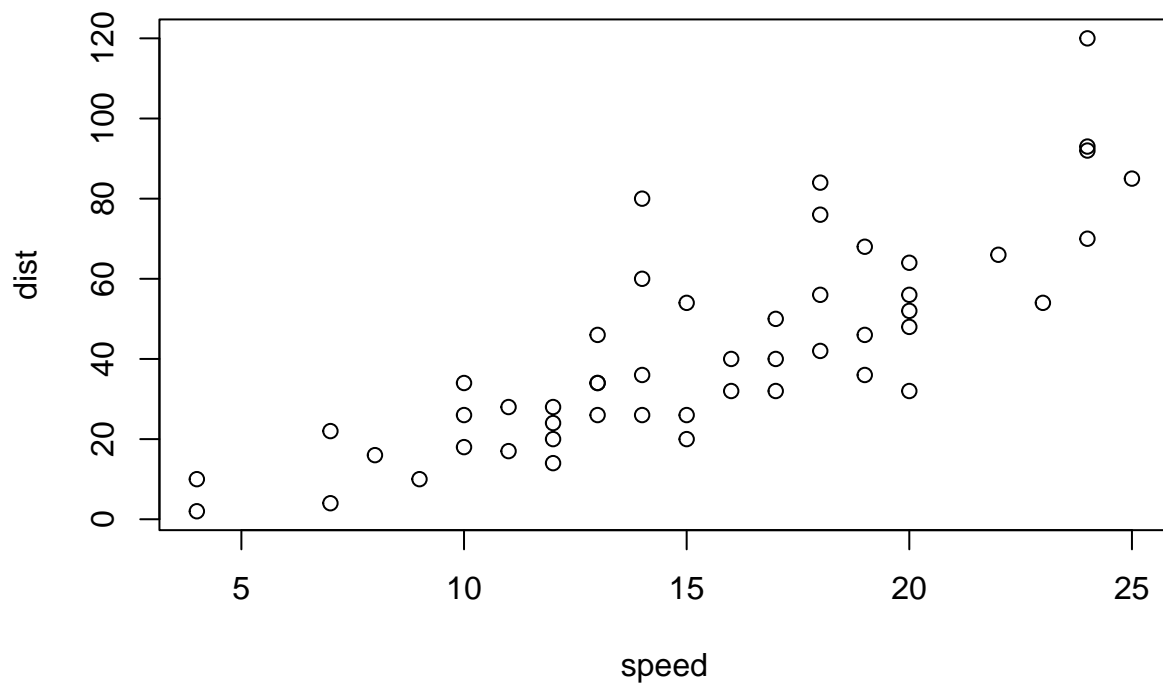
```
# Change to a linear model
p <- ggplot(data=cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  geom_smooth(method="lm")

p + labs(title="My nice plot",
         x="Speed (MPH)",
         y="Stopping Distance (ft)",
         caption="Dataset: 'cars'")
```

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



```
# Base graphics is shorter  
plot(cars)
```

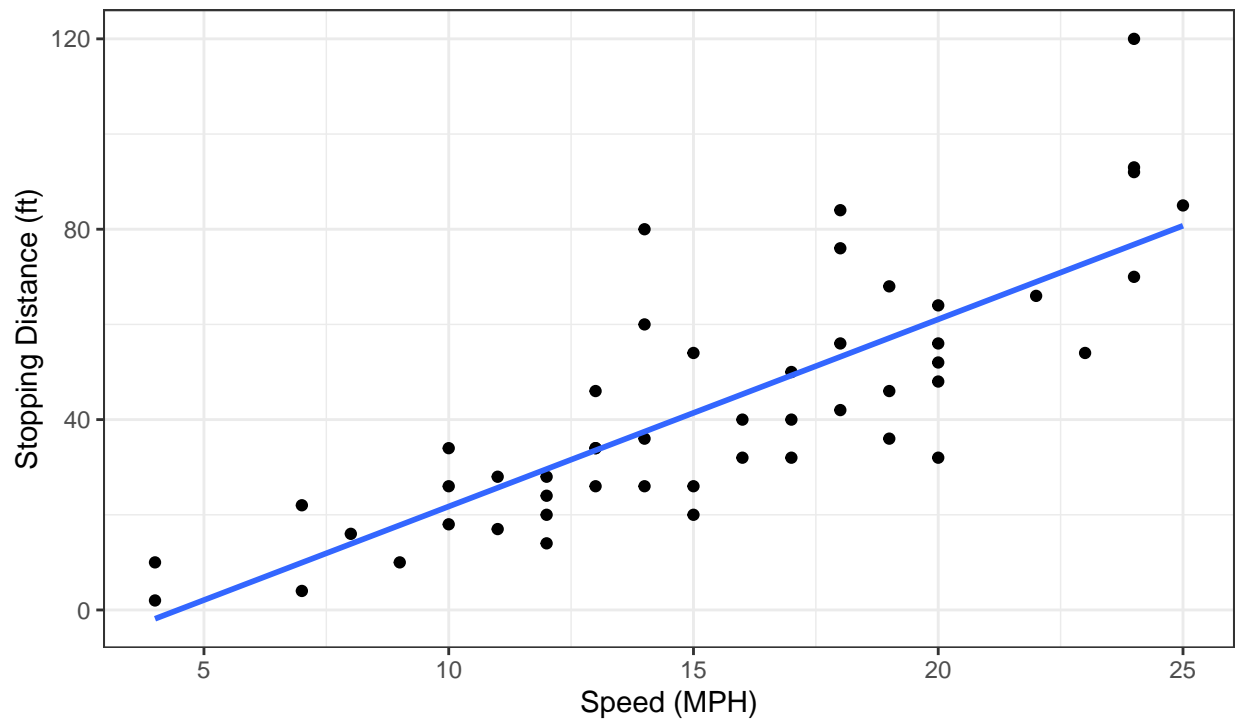


```
# Can add B/W theme
ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  labs(title="Speed and Stopping Distances of Cars",
        x="Speed (MPH)",
        y="Stopping Distance (ft)",
        subtitle = "Monica Lin",
        caption="Dataset: 'cars'") +
  geom_smooth(method="lm", se=FALSE) +
  theme_bw()
```

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

Speed and Stopping Distances of Cars

Monica Lin



Dataset: 'cars'

```
# Adding more plot aesthetics through aes()
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
```

```
# How many genes in this dataset?
nrow(genes)
```

```
## [1] 5196
```

```
# Use colnames() and ncol() on the genes data.frame
colnames(genes)
```

```
## [1] "Gene"      "Condition1" "Condition2" "State"
```

```
ncol(genes)
```

```
## [1] 4
```

```
# Use table() on the State column of this data.frame  
# How many 'up' regulated genes are there?  
table(genes$State)
```

```
##  
##      down  unchanged      up  
##      72      4997      127
```

```
# What % are up/down regulated?  
table(genes$State) / nrow(genes) * 100
```

```
##  
##      down  unchanged      up  
## 1.385681 96.170131 2.444188
```

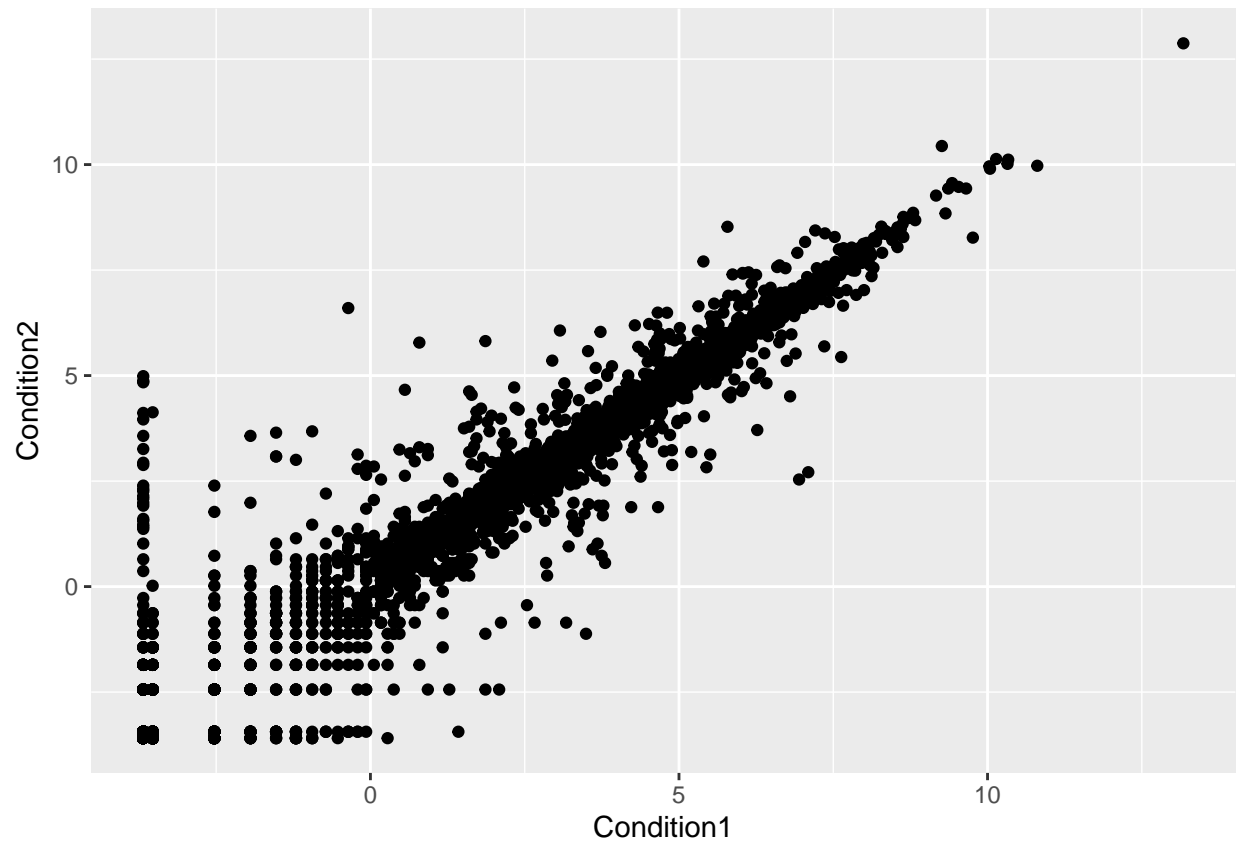
```
# Use the round() function to auto display sig figs  
round(table(genes$State) / nrow(genes) * 100, 2)
```

```
##  
##      down  unchanged      up  
##    1.39    96.17    2.44
```

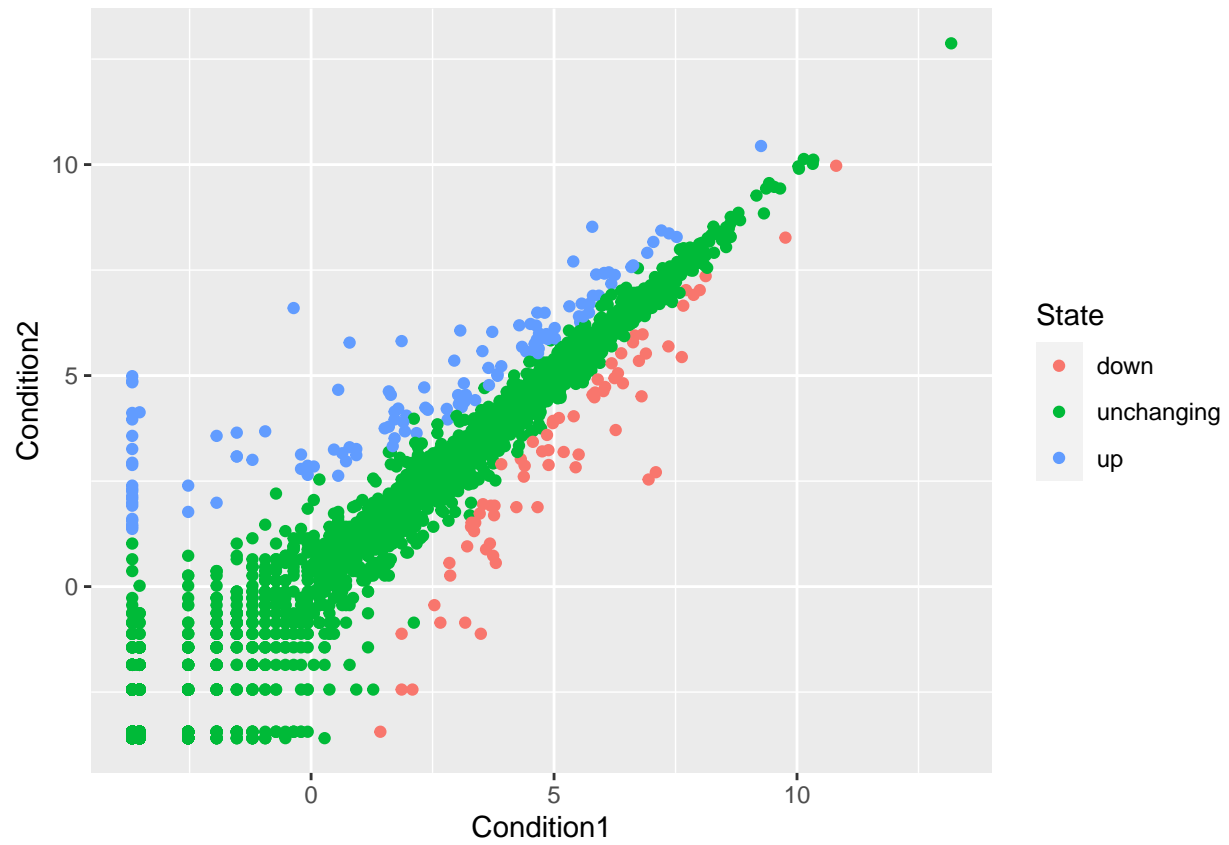
```
# Alternative way to round  
prec <- table(genes$State) / nrow(genes) * 100  
round(prec, 3)
```

```
##  
##      down  unchanged      up  
##    1.386    96.170    2.444
```

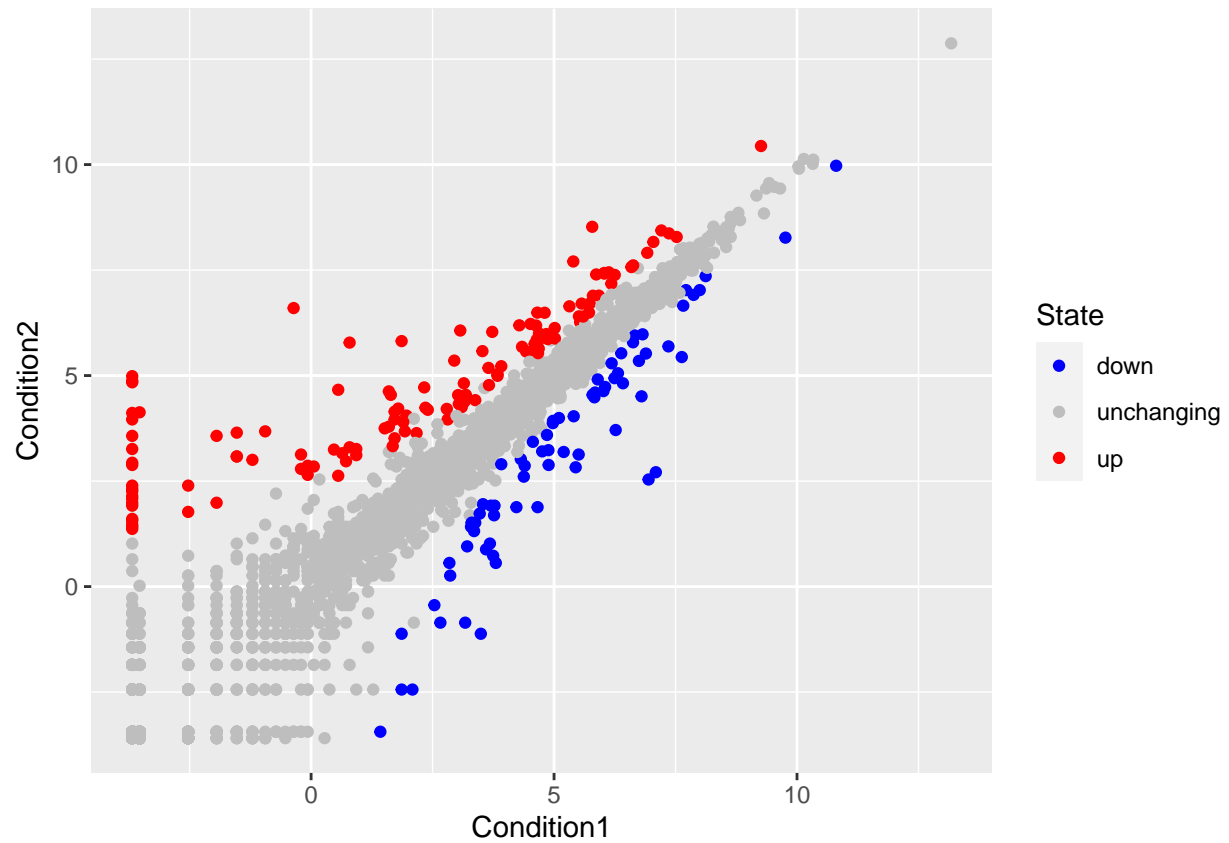
```
# Display the genes data  
ggplot(genes) +  
  aes(x=Condition1, y=Condition2) +  
  geom_point()
```



```
# State column tells stat sig  
p <- ggplot(genes) +  
  aes(x=Condition1, y=Condition2, col=State) +  
  geom_point()  
p
```

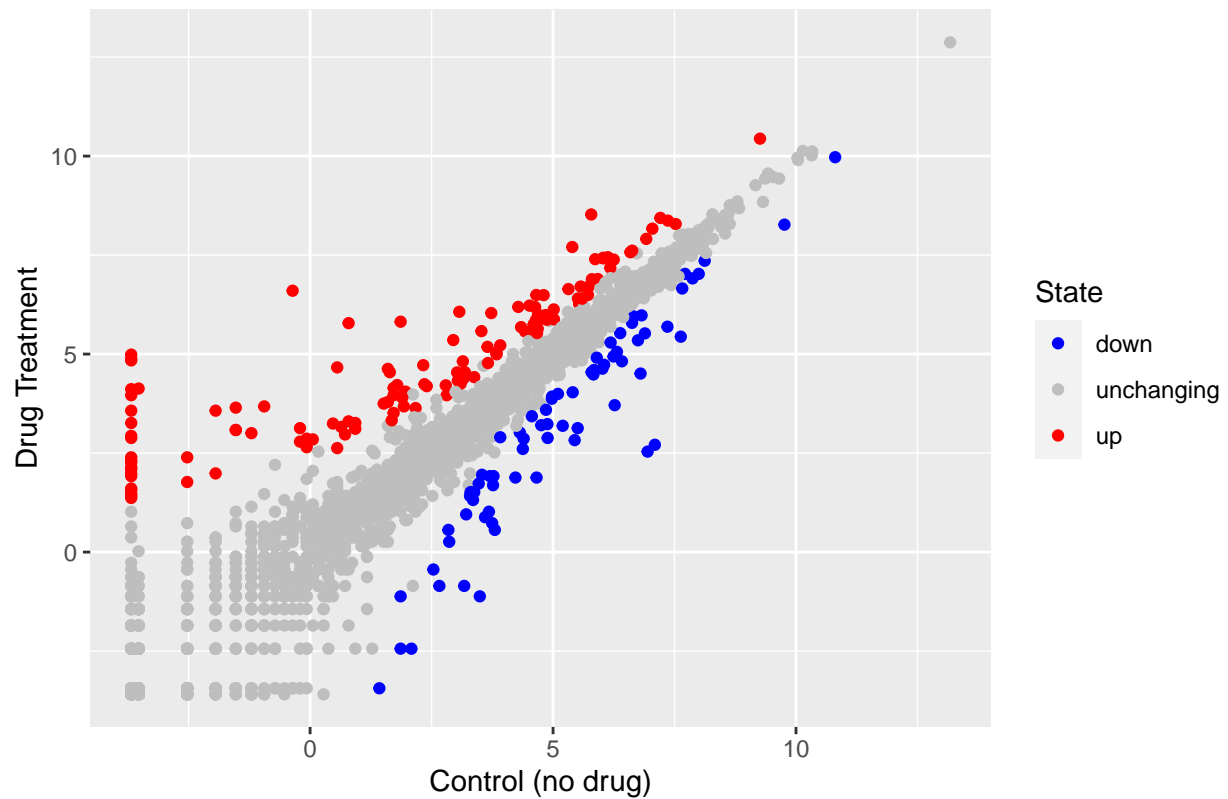



```
# Change to user-friendly colors  
p + scale_color_manual(values=c("blue", "gray", "red"))
```



```
# Add labels
p + scale_color_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



```
# OPTIONAL: Going Further
# gapminder dataset contains econ + demographic data on various countries since 1952
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.tsv"
gapminder <- read.delim(url)
```