Class 5: Data visualization

Bianca Barriga

##Plotting in R

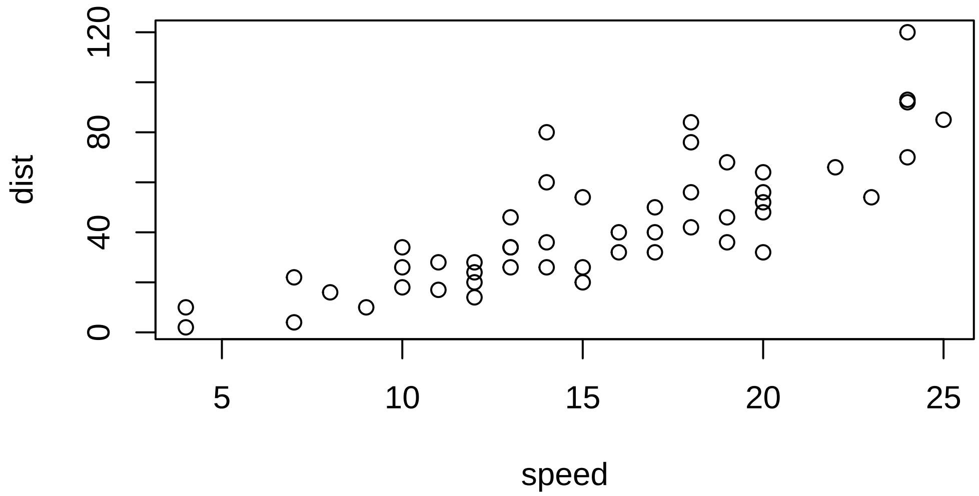
R has many plotting and visualization systems including “base” R. #Head function to print out first few rows, can also specify number of rows #head(dataset, n = i)

codechunks for management

```
head(cars, n=10)
```

```
   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
```

```
plot(cars)
```



Base R plots can be quite simple for basic plots when compared to systems like ggplot.

#how to not plot in ggplot

```
#ggplot2(cars)
```

It will produce and error because ggplot is not installed. To use an add on package like ggplot. I have to first install it onto the computer.

##**How to install a package**

We use the function 'install.packages()' with the name of the package we want to install.

Packages like ggplot need to be loaded from the library before every use, using the library function.

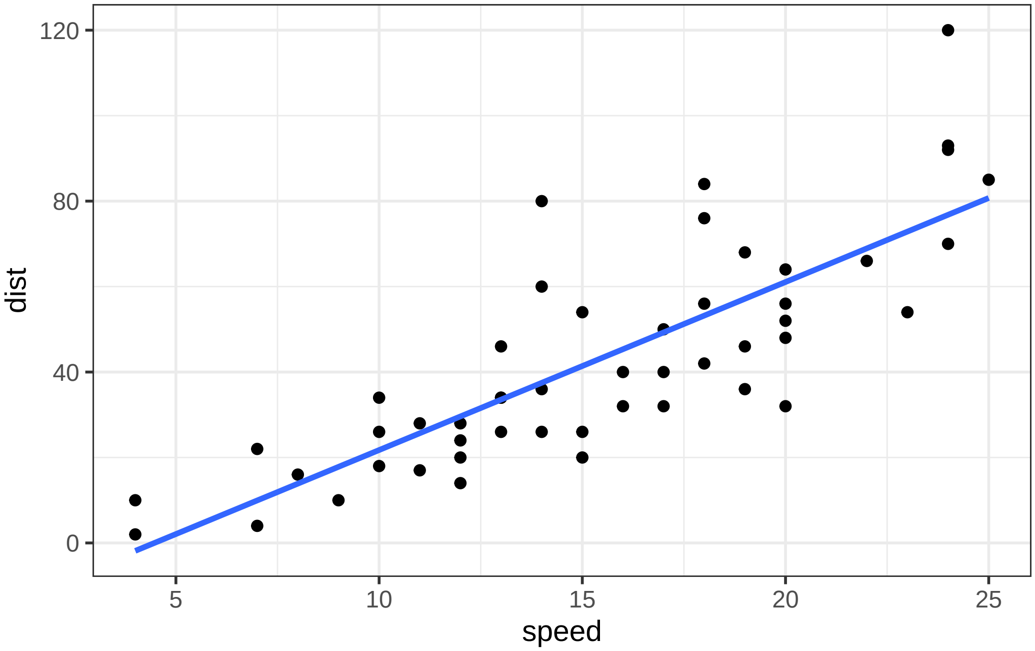
```
library(ggplot2)
```

##**ggplot minimum inputs** ggplot is much more requires more inputs than base R plot function. Ggplot requires 3 inputs at minimum:

- **Data** (this is the data.fram with the stuff we want to plot)
- **Aesthetics** or aes() for short (how the data map to the plot)
- **Geometery** (geom_point(), geomline() the plot type)

```r
ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point() +
  theme_bw() +
  geom_smooth(se=FALSE, method = lm)
```

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



A plot of some gene expression data

The code to read the data:

```r
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
```

> Q. How many genes are in this dataset?

```r
nrow(genes)
```

```
[1] 5196
```

#Example of inline code

There are 5196 genes in this dataset.

> How many genes are up-regulated?

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

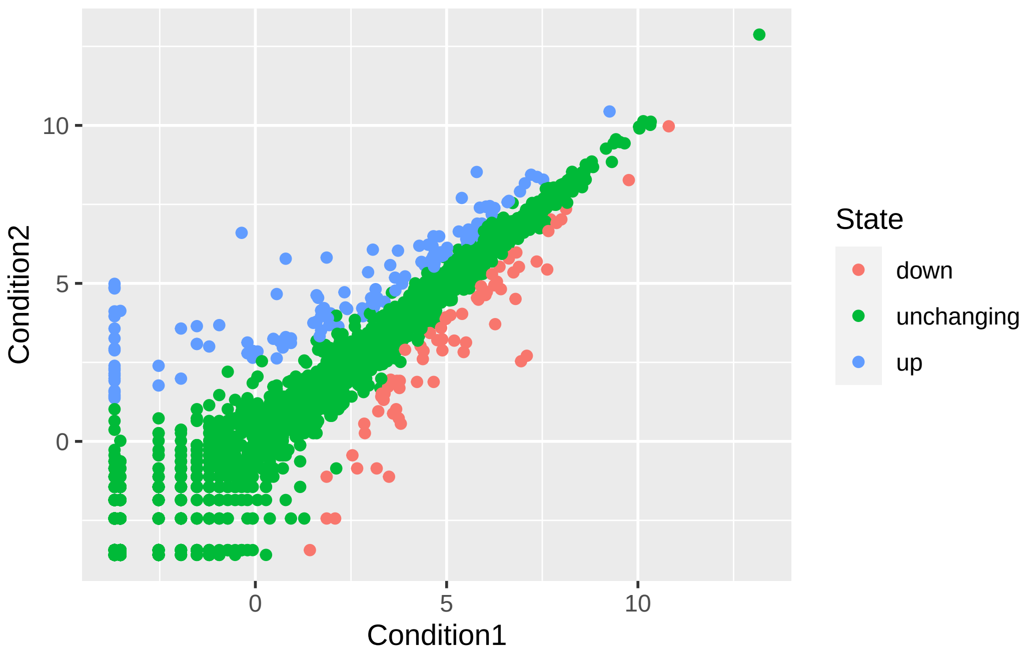
```

      down unchanging         up 
        72       4997        127 
```

```r
sum(genes$State == "up")
```

```
[1] 127
```

plotting

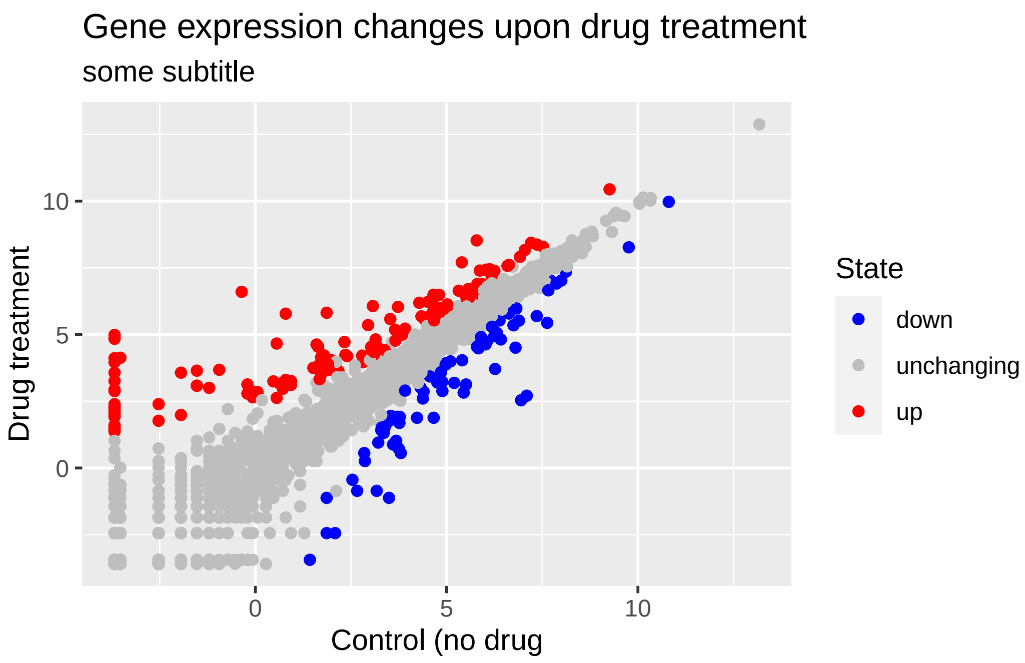
```r
 ggplot(genes) +
  aes(x = Condition1, y=Condition2, color=State) +
  geom_point(alpha = 1.0)
```



I can save any ggplot oject for use later so I dont need to type it all out again. Here I save my starting plot to object **p** then I can add layers to **p** later on.

```
p <- ggplot(genes) +
  aes(x = Condition1, y=Condition2, color=State) +
  geom_point(alpha = 1.0)
```

```
p + scale_color_manual(
  values=c("blue","gray","red")) +
  labs(title = "Gene expression changes upon drug treatment", subtitle = "some subtitle",
```



A more complex ggplot example

One of the big winds wiht ggplot is how easy it is to facet your data into sub-plots.

```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.

gapminder <- read.delim(url)
```

> Q. How many countrys are in this dataset?

```
length(unique(gapminder$country))
```

```
[1] 142
```

> Q. How many years are in this dataset?

```
length(unique(gapminder$year))
```

```
[1] 12
```

```
range(gapminder$year)
```

```
[1] 1952 2007
```

Q. How to find country with smallest population

```
min(gapminder$pop)
```

```
[1] 60011
```

#how to index - first where is this min value in the popvector

```
ind <- which.min(gapminder$pop)
```

```
gapminder$country[ind]
```

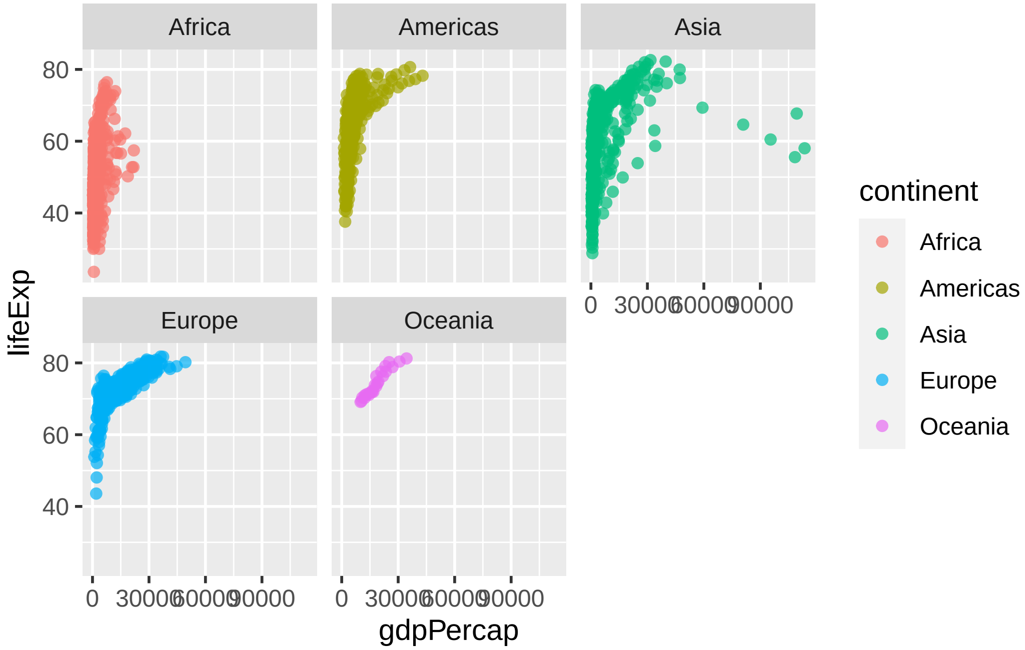
```
[1] "Sao Tome and Principe"
```

```
gapminder[ind,]
```

```
                   country continent year lifeExp   pop gdpPercap
1297 Sao Tome and Principe    Africa 1952  46.471 60011  879.5836
```

Plotting gdb vs. life expectancy

```
ggplot(gapminder) +
  aes(x=gdpPercap, y=lifeExp ,color=continent) +
  geom_point(alpha = 0.7) +
  facet_wrap(~continent)
```


Africa
Americas
Asia
Europe
Oceania
lifeExp
80
60
40
0
30000
60000
90000
gdpPercap
continent
● Africa
● Americas
● Asia
● Europe
● Oceania