

# BIMM 143 Class 5

Xaler Lu PID(A17388454)

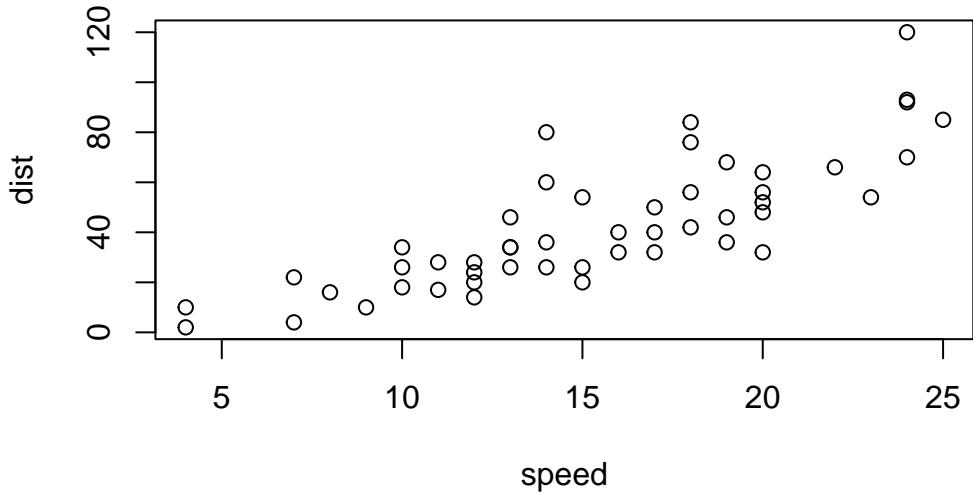
## Background

R has many ways to make figures like “base R” graphics (eg. `plot()`) and tons of add-on packages like `ggplot2`. For example, here we make the sample plot with both “base R” and with `ggplot2`.

```
head(cars)
```

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

```
plot(cars)
```



However, I do not have `ggplot` downloaded yet, so I will need `install.packages()`.

**N.B.** We never run an install command in a quarto code chunk or we will end up reinstalling packages many times

We would then need to load the ggplot from the library with `library()`. Afterwards, every ggplot would need at least three things:

- **data**: The stuff you want to be plotted
- **aesthetics**: How the data map to the plot
- **geommetry**: The type of plot

```
library(ggplot2)
```

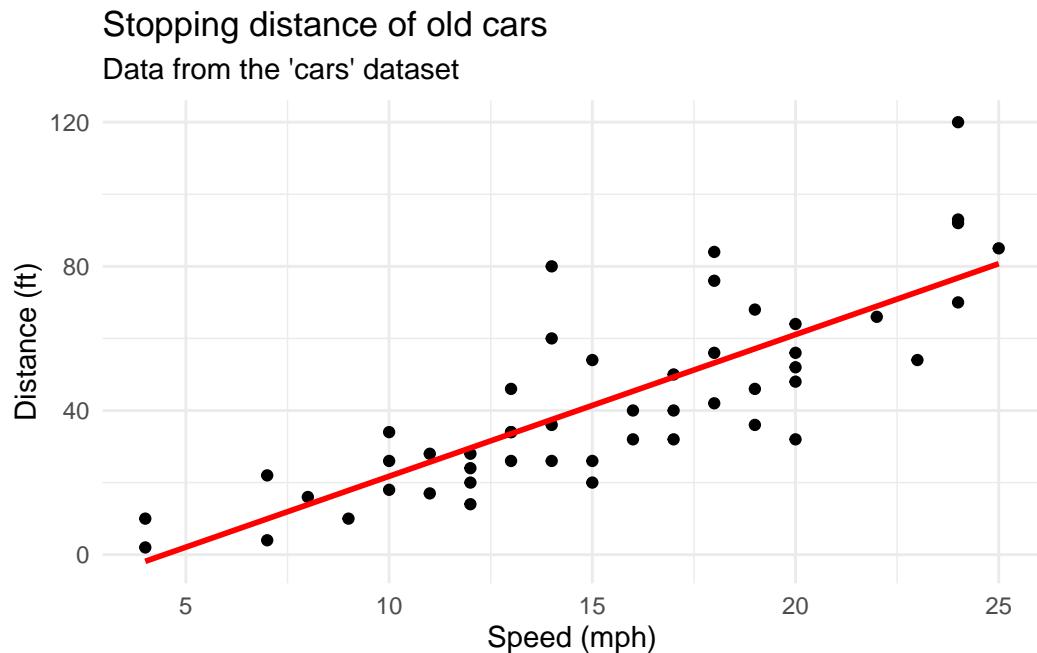
```
Warning: package 'ggplot2' was built under R version 4.3.3
```

```
p <- ggplot(cars) +
  aes(x=speed, y=dist) +
  geom_point(color = 'black') +
  geom_smooth(method="lm", se=FALSE, formula = 'y ~ x', color = 'red') +
```

```
labs(title="Stopping distance of old cars",
     subtitle="Data from the 'cars' dataset",
     x = "Speed (mph)",
     y = "Distance (ft)")
```

We set a variable for the ggplot to later add a theme with `theme`.

```
p + theme_minimal()
```



## Gene Expression Plot

Here is some code to answer some questions

```
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
```

```
ncol(genes)
```

```
[1] 4
```

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

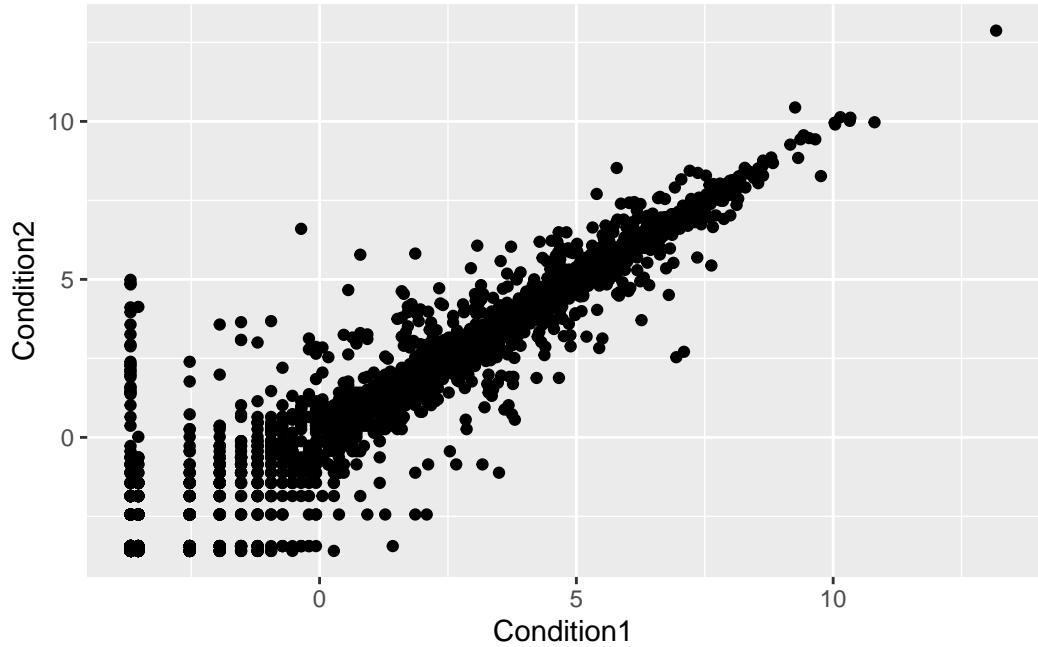
	down	unchanging	up
	72	4997	127

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

	down	unchanging	up
	1.39	96.17	2.44

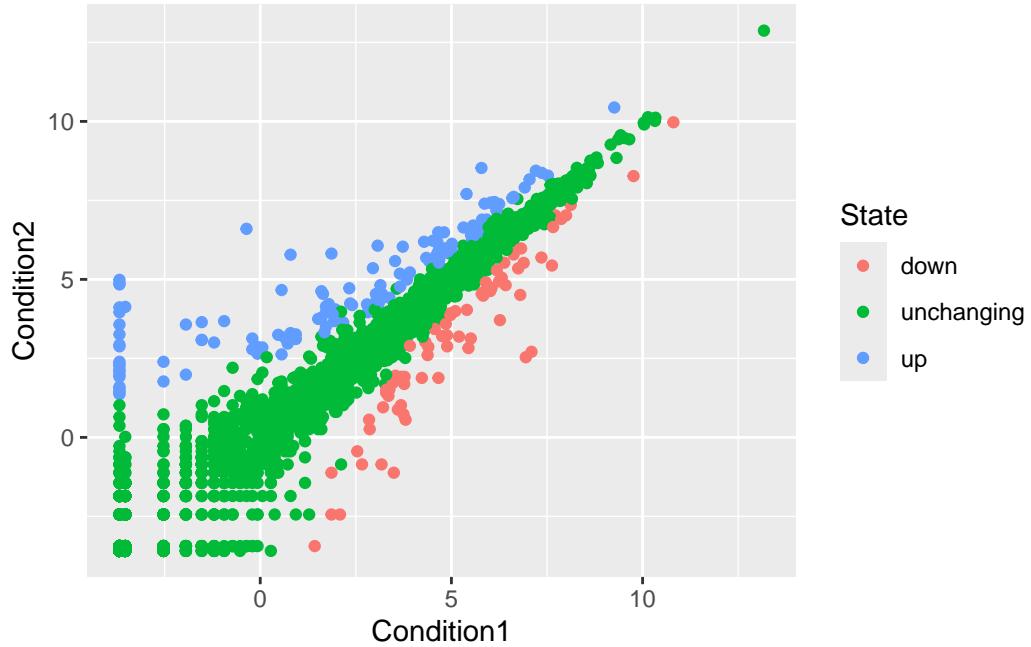
Version 1 of the ggplot

```
ggplot(genes) +
  aes(Condition1, Condition2) +
  geom_point()
```



Version 2 of the ggplot with colors by State.

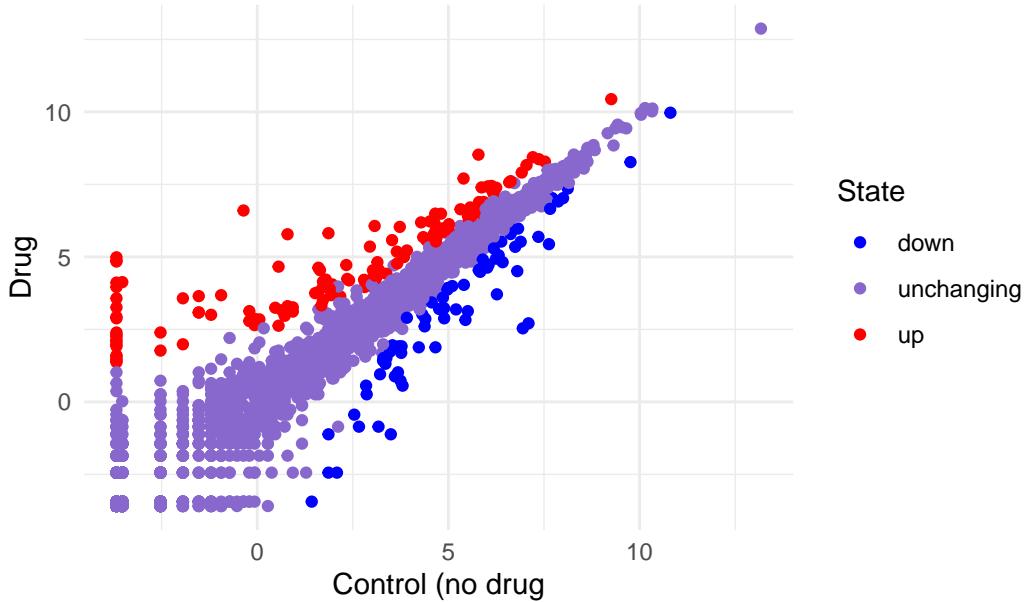
```
ggplot(genes) +  
  aes(Condition1, Condition2, color = State) +  
  geom_point()
```



Version 3 of the `ggplot` to further modify the colors with `scale_color_manual` and to add titles with `labs`

```
ggplot(genes) +
  aes(Condition1, Condition2, col = State) +
  geom_point() +
  scale_color_manual( values = c("blue","#8968CD", "red")) +
  labs(x = "Control (no drug",
       y = "Drug",
       title = "Gene Expression Changes upon GLP-1 drug") +
  theme_minimal()
```

## Gene Expression Changes upon GLP-1 drug



## Going Further

Setting up

```
library(gapminder)
```

```
Warning: package 'gapminder' was built under R version 4.3.3
```

```
library(dplyr)
```

```
Attaching package: 'dplyr'
```

```
The following objects are masked from 'package:stats':
```

```
filter, lag
```

```
The following objects are masked from 'package:base':
```

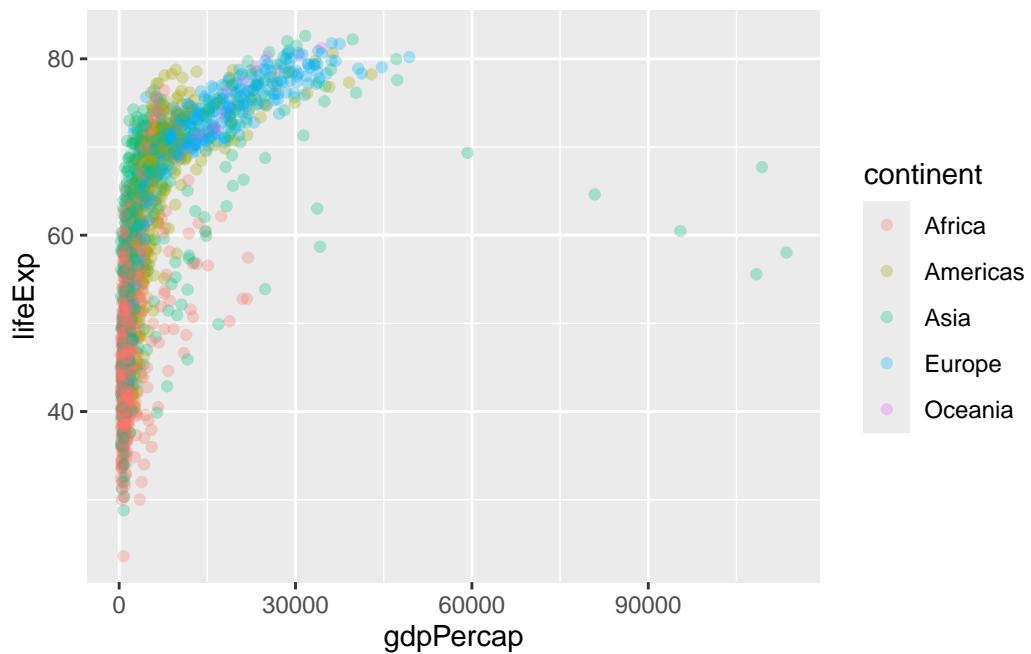
```
intersect, setdiff, setequal, union
```

```
url <- "https://raw.githubusercontent.com/jennybc/gapminder/master/inst/extdata/gapminder.csv"
gapminder <- read.delim(url)
head(gapminder, 4)
```

	country	continent	year	lifeExp	pop	gdpPerCap
1	Afghanistan	Asia	1952	28.801	8425333	779.4453
2	Afghanistan	Asia	1957	30.332	9240934	820.8530
3	Afghanistan	Asia	1962	31.997	10267083	853.1007
4	Afghanistan	Asia	1967	34.020	11537966	836.1971

Version 1 of the ggplot

```
ggplot(gapminder) +
  aes(gdpPerCap, lifeExp, col = continent) +
  geom_point(alpha = 0.3)
```



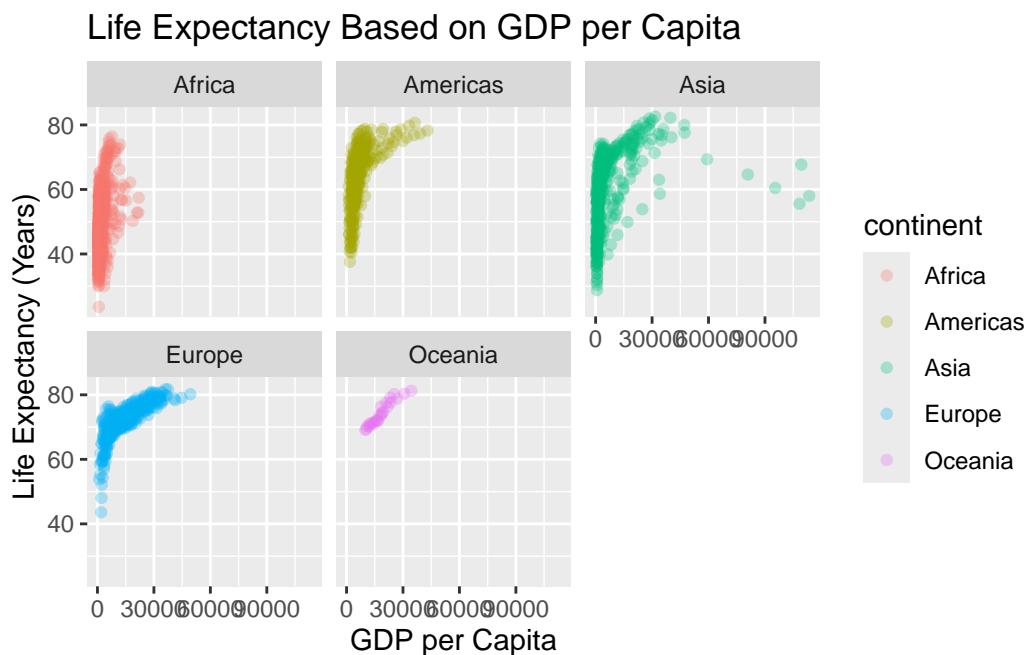
Now, let's “facet” (i.e. make a separate plot) by continent with `facet_wrap`

```
ggplot(gapminder) +
  aes(gdpPerCap, lifeExp, col = continent) +
```

```

geom_point(alpha = 0.3) +
facet_wrap(~continent) +
labs(title = "Life Expectancy Based on GDP per Capita",
x = "GDP per Capita",
y = "Life Expectancy (Years)")

```



How big is this gapminder data? How do we separate the data by year?  
> **N.B.** Hint, we will use the `filter` function from the `dplyr` package

```

library(dplyr)
nrow(gapminder)

```

[1] 1704

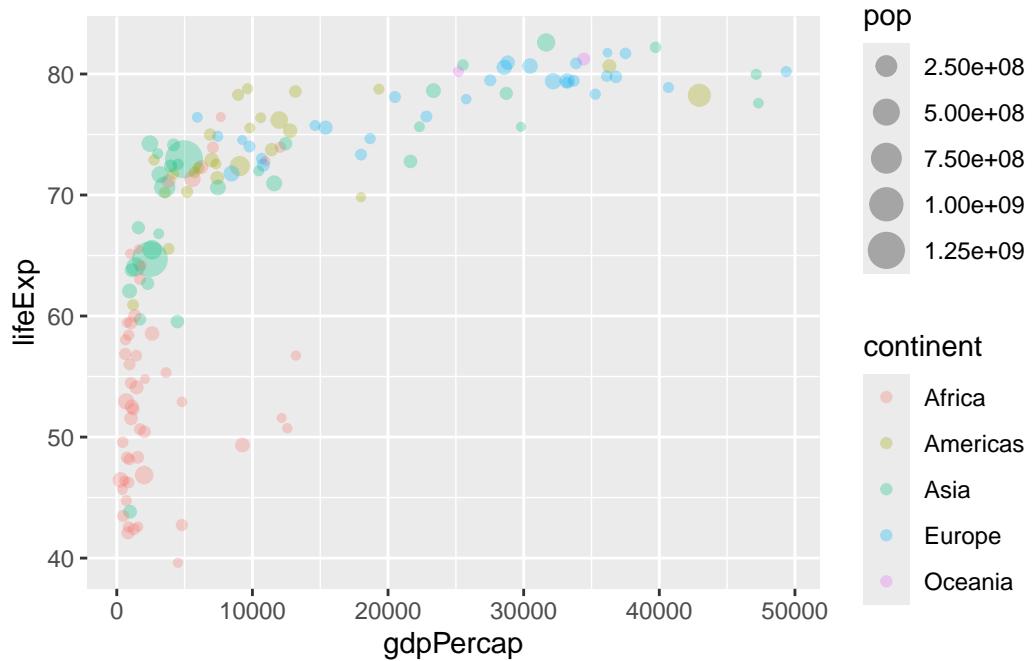
Here, I will select the data only from 2007 that also includes the size of the population using `size`.

```

gapminder_2007 <- filter(gapminder, year == 2007)
ggplot(gapminder_2007) +

```

```
aes(gdpPercap, lifeExp, col = continent, size = pop) +
geom_point(alpha = 0.3)
```



Here, I will filter even further by countries.

```
gapminder_2007_Ireland <- filter(gapminder, year == 2007, country == "Ireland")
gapminder_2007_Ireland
```

	country	continent	year	lifeExp	pop	gdpPercap
1	Ireland	Europe	2007	78.885	4109086	40676

```
filter(gapminder, year == 2007, country == "United States")
```

	country	continent	year	lifeExp	pop	gdpPercap
1	United States	Americas	2007	78.242	301139947	42951.65

Q. Make a plot comparing 1977 and 2007 for all countries

```

gapminder_compare <- filter(gapminder, year %in% c(1977, 2007))
ggplot(gapminder_compare) +
  aes(gdpPercap, lifeExp, col = continent) +
  geom_point(alpha = 0.) +
  facet_wrap(~year) +
  labs(title = "GDP per Capita to Life Expectancy in 1977 and 2007",
       x = "GDP per Capita",
       y = "Life Expectancy (Year)")

```

