

Class 5: Data Viz by ggplot

Chen (PID=A59026768)

2024-01-24

Graphics systems in R

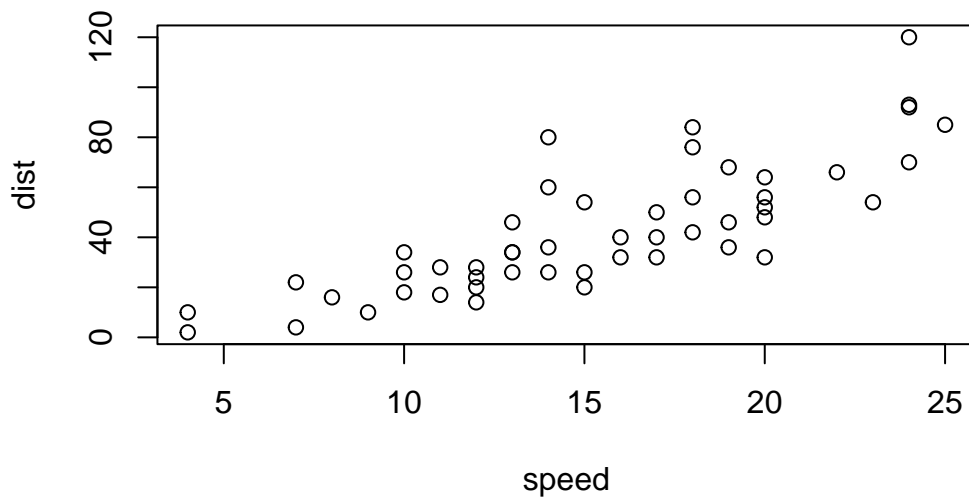
There are many graphics systems for R. These include so called “**base R**” and those in add-on packages like `ggplot2`.

```
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

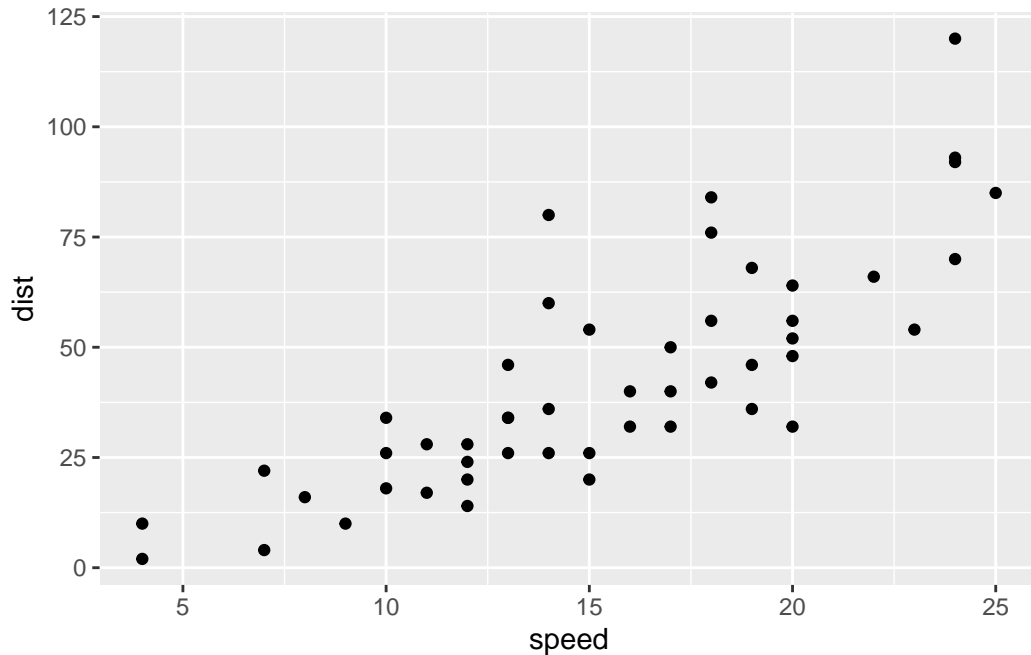
```
plot(cars)
```



How can we make this with `ggplot`? This is an add-on package and I first need to install it on my computer. This install is to a one time only deal. To install packages use `install.packages()` function. To use it we need to load up the package from our library use `library()`

```
#do it in console
#install.packages("ggplot2")
library(ggplot2)

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



Using ggplot is not as straightforward as base R plot for basic plots. I have some more typing to do. Every ggplot has at least 3 things (layers): -data (data.frame) -aes (how the data map to the plot) -geoms (think of this as the type of plot, e.g. points, lines ...etc)

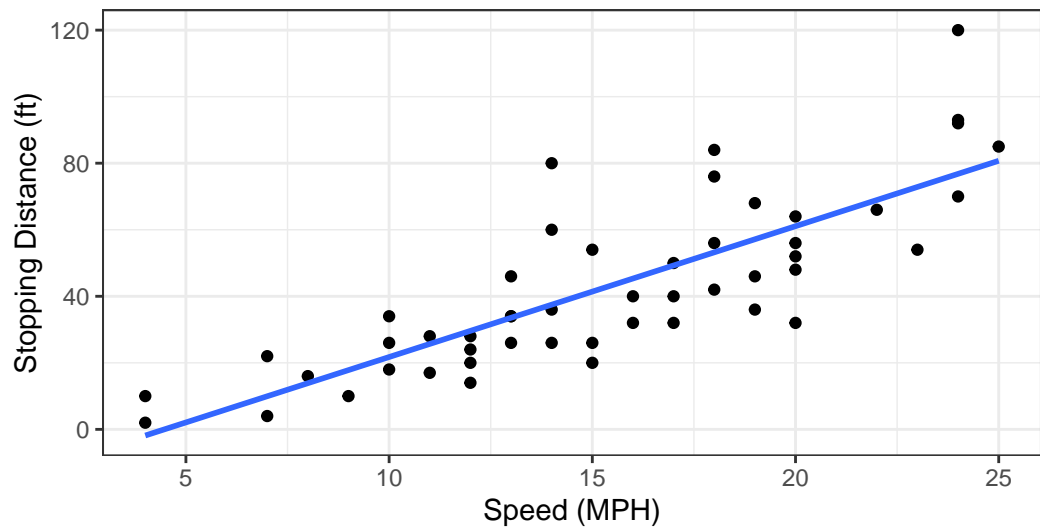
Here ggplot is verbose- i.e. U had more typing to do. However, I can add more layers.

```
ggplot(cars)+
  aes(speed, dist)+
  geom_point()+
  #geom_path()+
  geom_smooth(method= "lm", se = FALSE)+
  labs(title="Speed and Stopping Distances of Cars",
        x="Speed (MPH)",
        y="Stopping Distance (ft)",
        subtitle = "Your informative subtitle text here",
        caption="Dataset: 'cars'")+
  theme_bw()
```

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

Speed and Stopping Distances of Cars

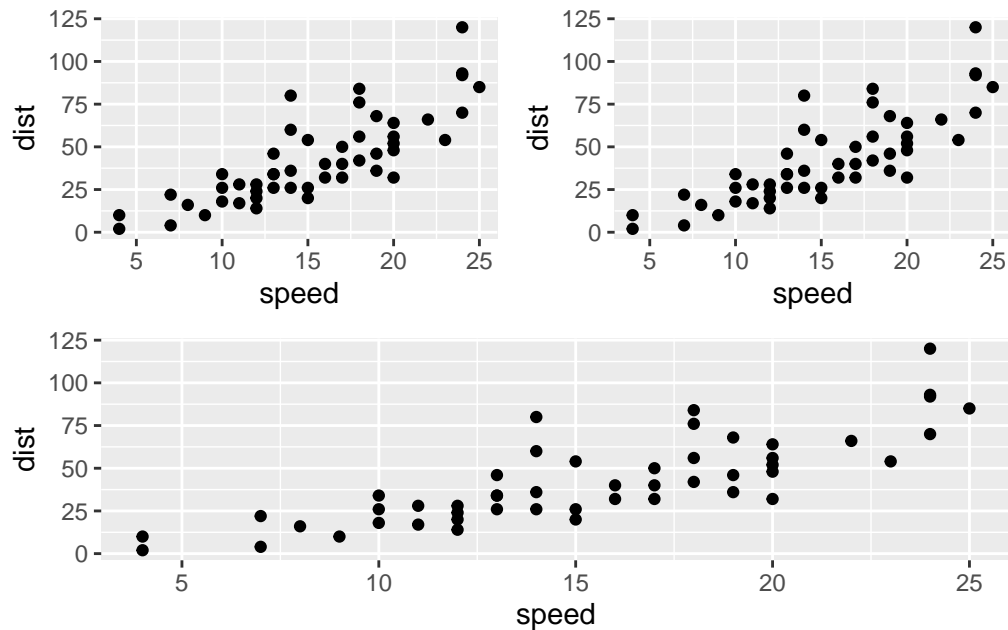
Your informative subtitle text here



Dataset: 'cars'

```
p <- ggplot(cars)+  
  aes(speed, dist)+  
  geom_point()
```

```
library(patchwork)  
(p | p)/p
```



```
url <- "https://bioboot.github.io/bimm143_S20/class-material/up_down_expression.txt"
genes <- read.delim(url)
head(genes, 7)
```

	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
7	ABCA7	3.4484220	3.8266509	unchanging

Use the `nrow()` function to find out how many genes are in this dataset. What is your answer?

```
colnames(genes)
```

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

```
ncol(genes)
```

```
[1] 4
```

```
n <- nrow(genes)
t <- table(genes$State)
n
```

```
[1] 5196
```

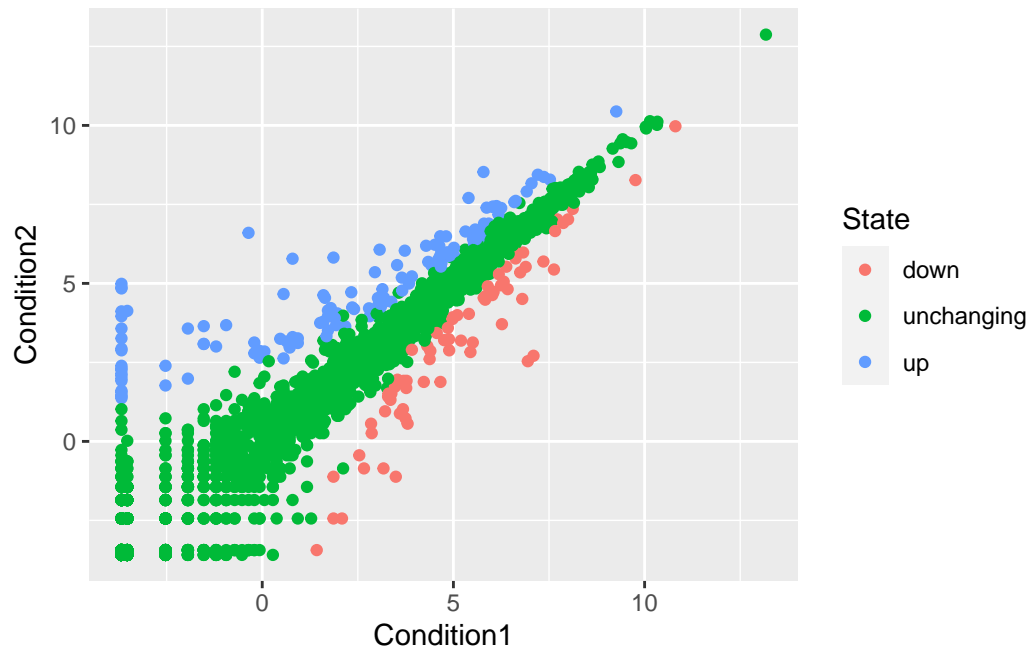
```
t
```

down	unchanging	up
72	4997	127

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

down	unchanging	up
1.39	96.17	2.44

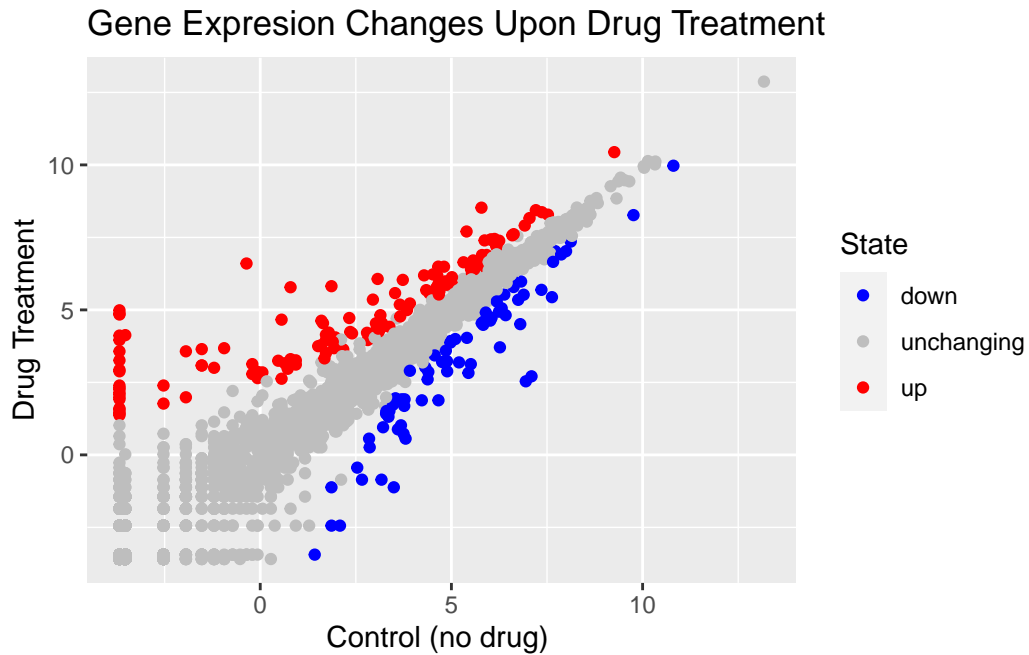
```
ggplot(genes)+
  #if you want ur color depends on your data, u have to argue the color at aes, instead of
  aes(x = Condition1, y = Condition2, col = State)+
  geom_point()
```



```
p <- ggplot(genes)+
  aes(x = Condition1, y = Condition2, col=State, name=Gene)+
  geom_point()+
  scale_colour_manual( values=c("blue","gray","red") )

p2 <-p+ labs(title="Gene Expression Changes Upon Drug Treatment",
             x="Control (no drug) ",
             y="Drug Treatment")

p2
```

There is an interactive version wotj `plotly` so that you can pinpoint any point on the figure.

```
library(plotly)
```

Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

`last_plot`

The following object is masked from 'package:stats':

`filter`

The following object is masked from 'package:graphics':

`layout`

```
#ggplotly(p2)
```

When library(plotly), there are some messages says:

Attaching package: 'plotly'

The following object is masked from 'package:ggplot2':

last_plot

The following object is masked from 'package:stats':

filter

The following object is masked from 'package:graphics':

layout

In this case,

```
#use "|" instead of "/"
```

```
library(plotly)
```

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

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

```
library(ggplot2)
```

```
library(dplyr)
```

Attaching package: 'dplyr'

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

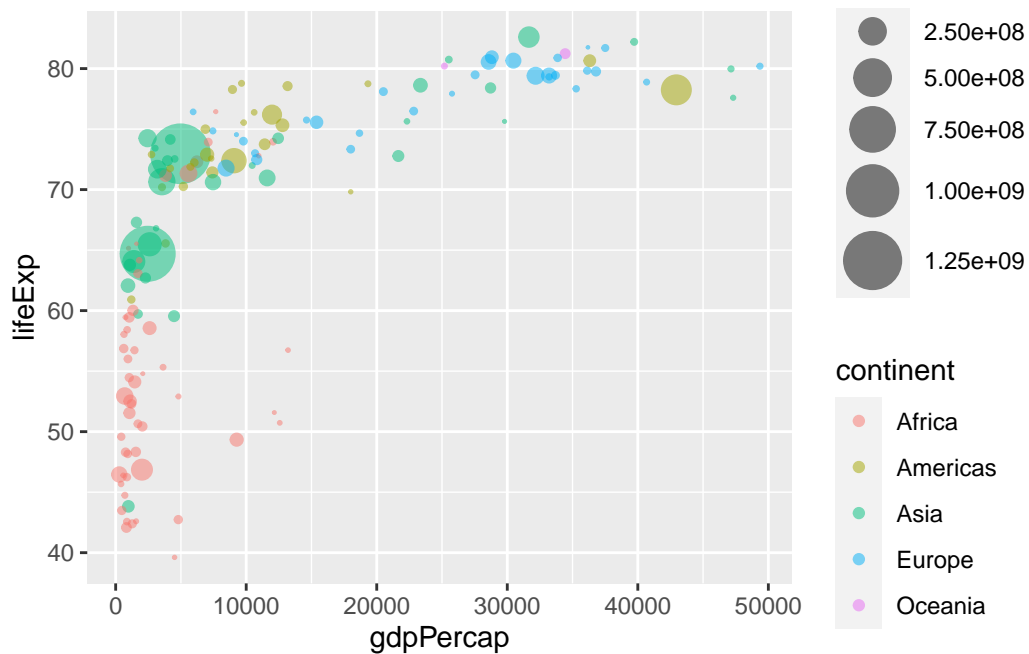
filter, lag

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

intersect, setdiff, setequal, union

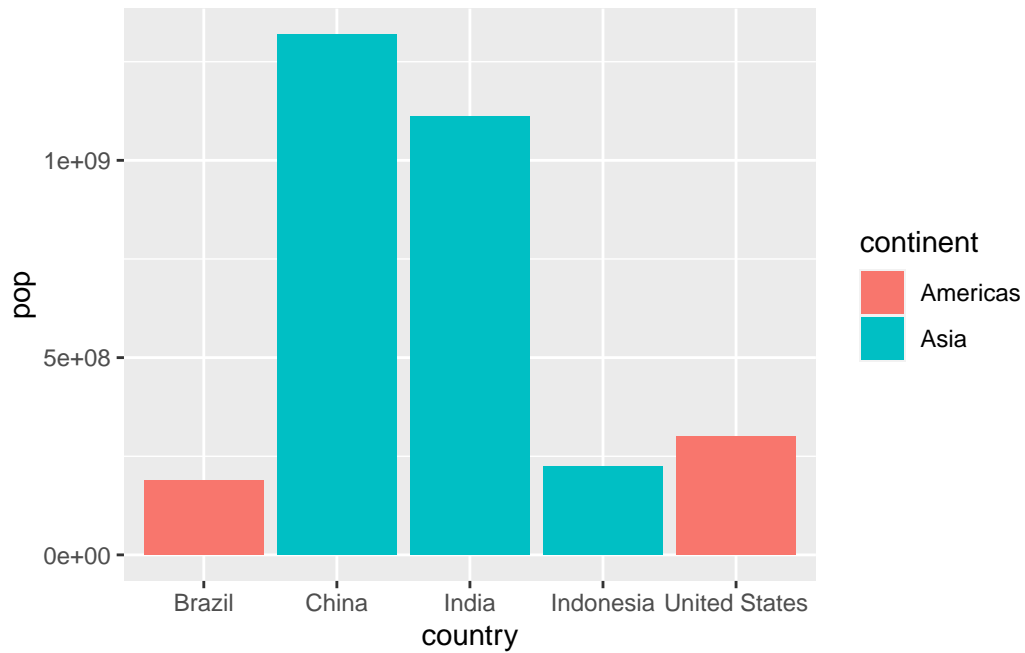
```
gapminder_2007 <- gapminder %>% filter(year==2007)
```

```
ggplot(gapminder_2007)+
  geom_point(aes(x = gdpPercap, y = lifeExp, col = continent, size = pop), alpha = 0.5)+
  scale_size_area(max_size = 10)
```



```
gapminder_top5 <- gapminder %>% filter(year==2007) %>% arrange(desc(pop)) %>%
  top_n(5, pop)
#gapminder_top5
```

```
ggplot(gapminder_top5) +
  geom_col(aes(x=country, y= pop, fill = continent))
```



```
ggplot(gapminder_top5)+
  geom_col(aes(x=reorder(country, -pop), y=pop, fill=gdpPercap))
```

