

# lab05: Data vis with ggplot

Karina Cardenas | PID: A16742606

## Table of contents

<b>Background</b>	<b>1</b>
Plotting in Base . . . . .	2
Importing ggplot . . . . .	2
<b>Gene expression</b>	<b>8</b>
Color scatterplot . . . . .	9
<b>Using different geoms</b>	<b>10</b>
Panel of plots . . . . .	14
<b>Life Expectancy by country plot</b>	<b>15</b>
Gapminder plot . . . . .	16

## Background

There are many graphic systems available in R. These include “base” R and tons of add-on packages like **ggplot2**.

Let’s compare “base” and **ggplot2** briefly: We can use some example data that is built-in with R called `cars`:

```
head(cars)
```

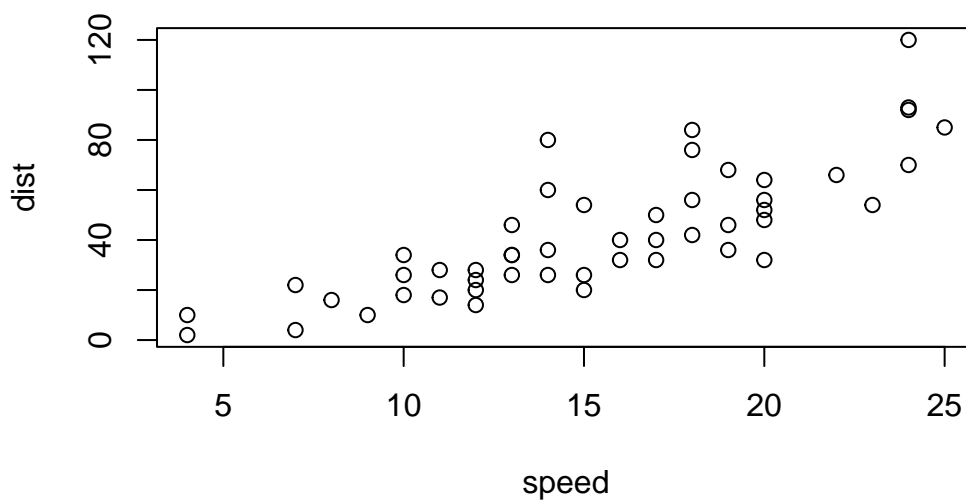
```
      speed dist
1         4    2
2         4   10
3         7    4
4         7   22
```

5	8	16
6	9	10

## Plotting in Base

In base R I can just call `plot()`

```
plot(cars)
```



## Importing ggplot

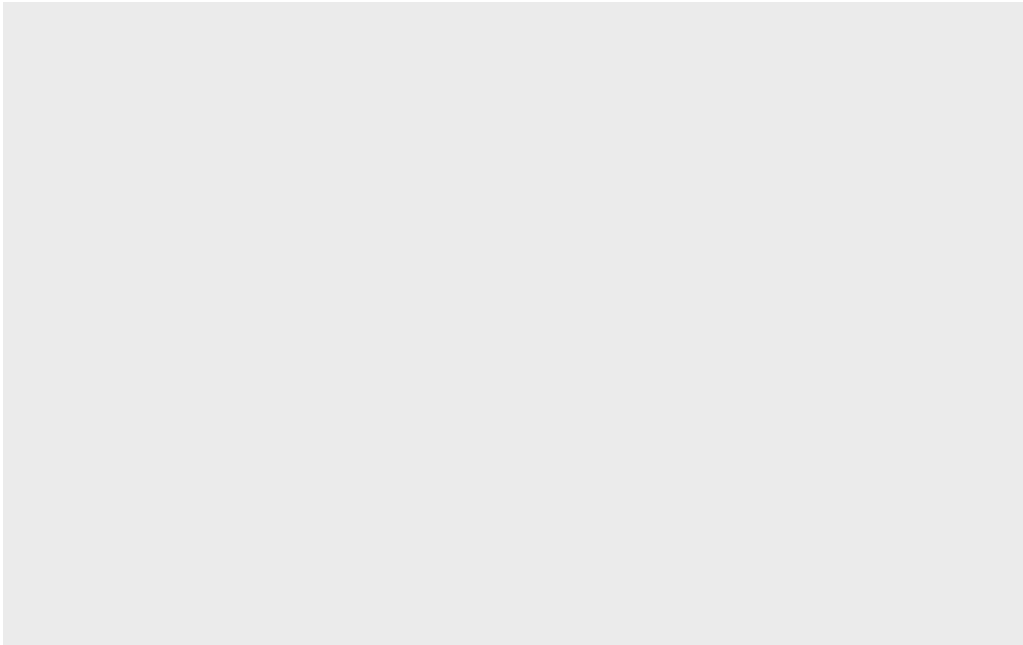
How we can do this with **ggplot2**

First we need to install the package. We do this `install.packages("ggplot2")`. I only need to do this once and then it will be available on my computer from then on.

Key point: I only install packages in the R console not within quarto docs or R scripts.

Before using any add-on package, loading it up with a call to `library()`

```
library(ggplot2)
ggplot(cars)
```

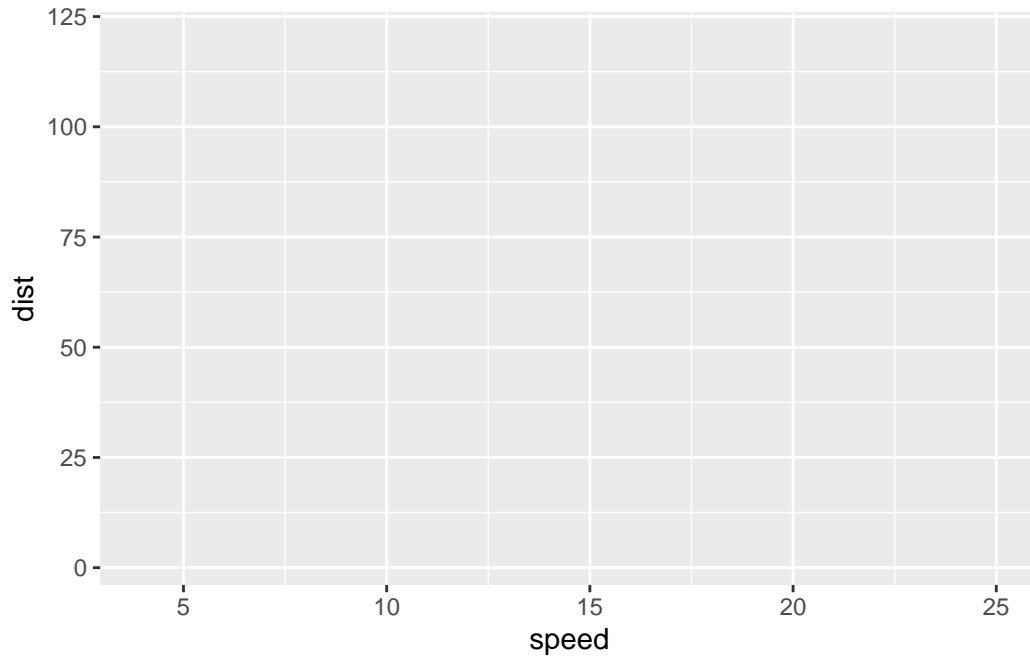


Every ggplot has at least 3 things:

- the **data** (in our case `cars`)
- the **aesthetics** (how the data map to the plot)
- the **geoms** that determine how the plot is drawn (lines, points, column, boxplots, densities, etc)

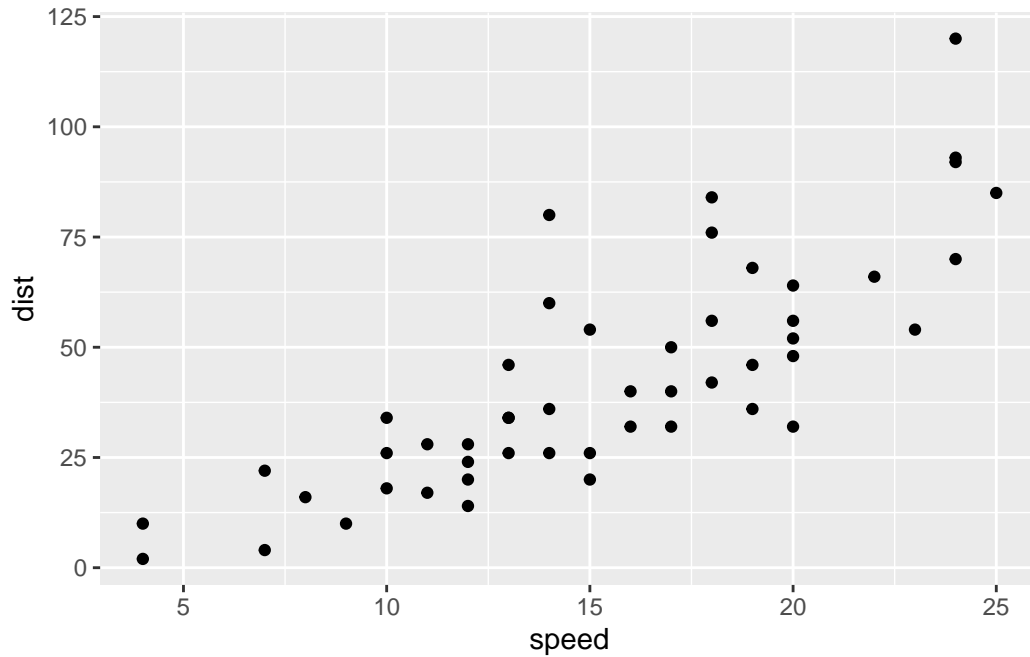
Two of three elements: introduced the aesthetics such as specifying the x and y axes.

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



All three elements: For “simple” plots, ggplot is much more verbose than base R but the defaults are nicer and for complicated plots it becomes much more efficient and structured.

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



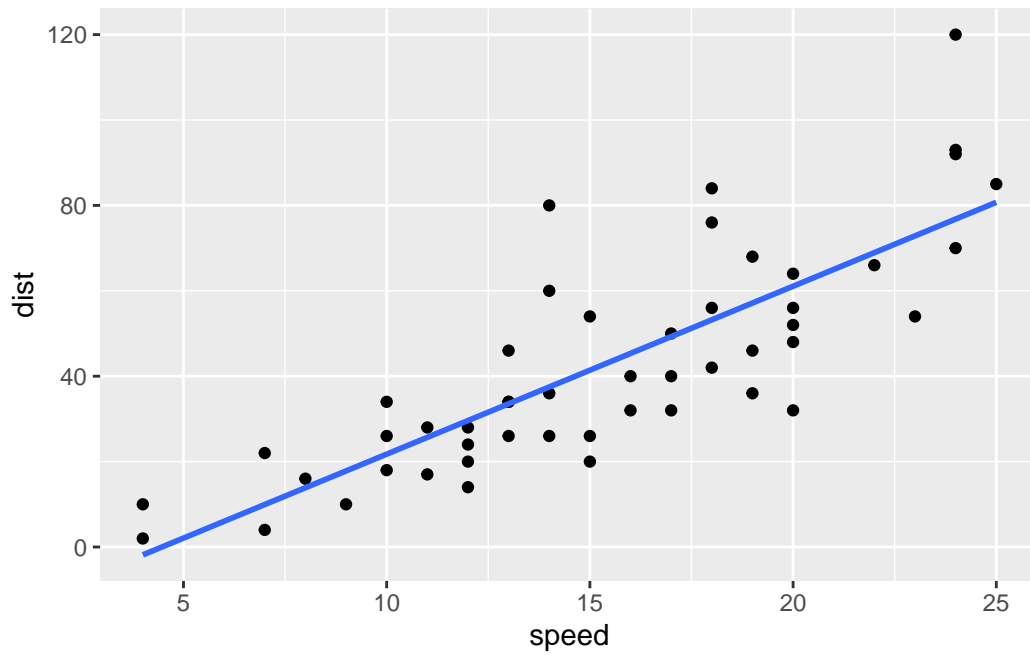
Q. Add a line to show the relationship of speed and stopping distance (i.e add another “layer”)

```
p <- ggplot(cars) +  
  aes(x = speed, y = dist) +  
  geom_point() +  
  geom_smooth(se = FALSE, method = "lm")
```

I can always save any ggplot object (i.e plot) and then use it later for adding more layers.

```
p
```

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

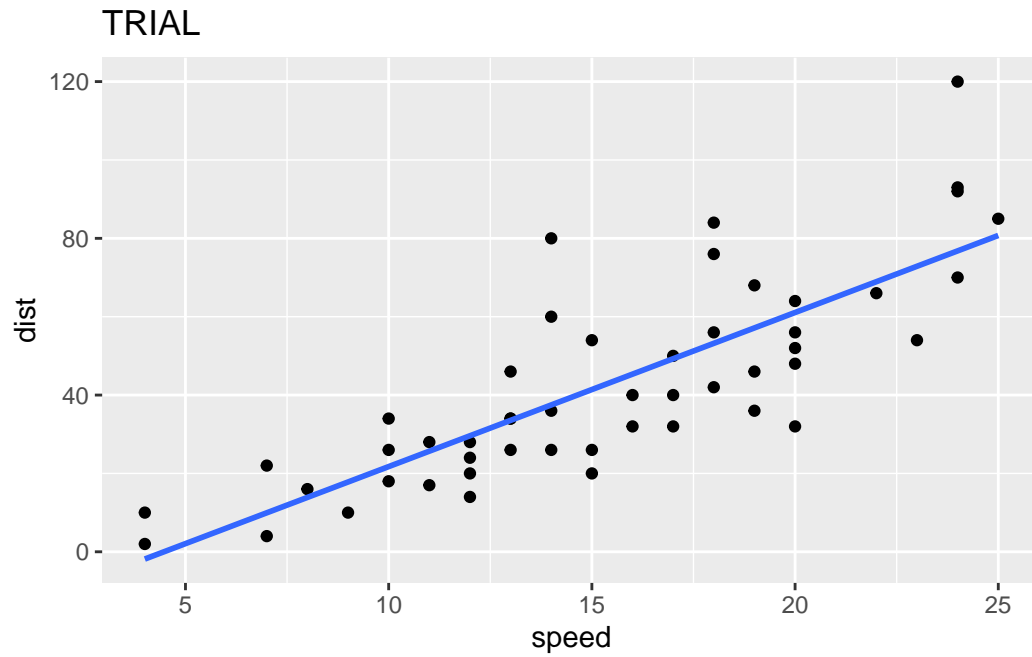


Q. Add a title and subtitle to the plot

Option 1

```
p + ggtitle("TRIAL")
```

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



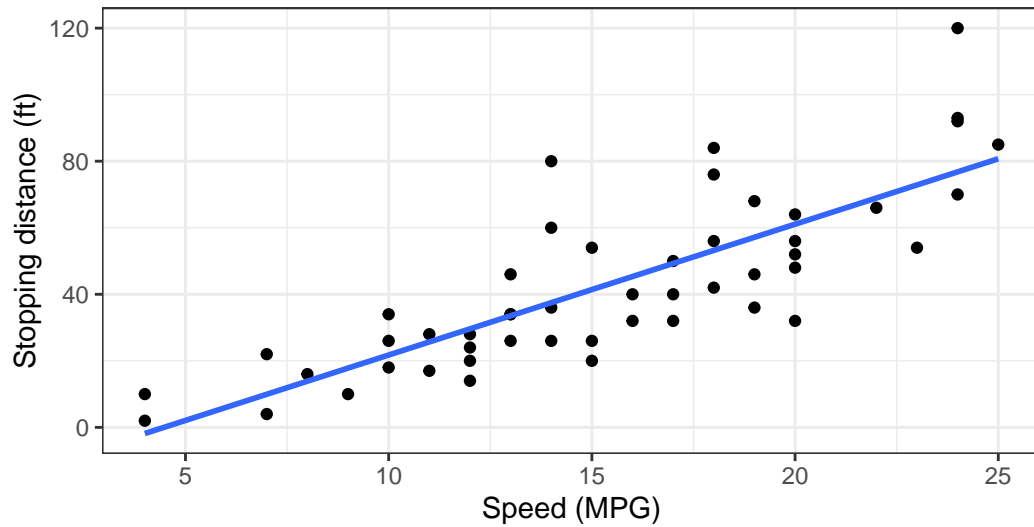
Option 2

```
p + labs(title = "My first ggplot",  
         subtitle = "stopping distance of old cars",  
         caption = "BIMM 143",  
         x = "Speed (MPG)",  
         y = "Stopping distance (ft)") +  
  theme_bw()
```

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

## My first ggplot

stopping distance of old cars



BIMM 143

## Gene expression

Read input data into 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?

```
nrow(genes)
```

```
[1] 5196
```



Q. How many columns are in the dataset?

```
ncol(genes)
```

```
[1] 4
```

Q. What are the column names?

```
colnames(genes)
```

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

Q. How many up and down regulated genes are there?

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

down	unchanging	up
72	4997	127

Q. What fraction of total genes are upregulated?

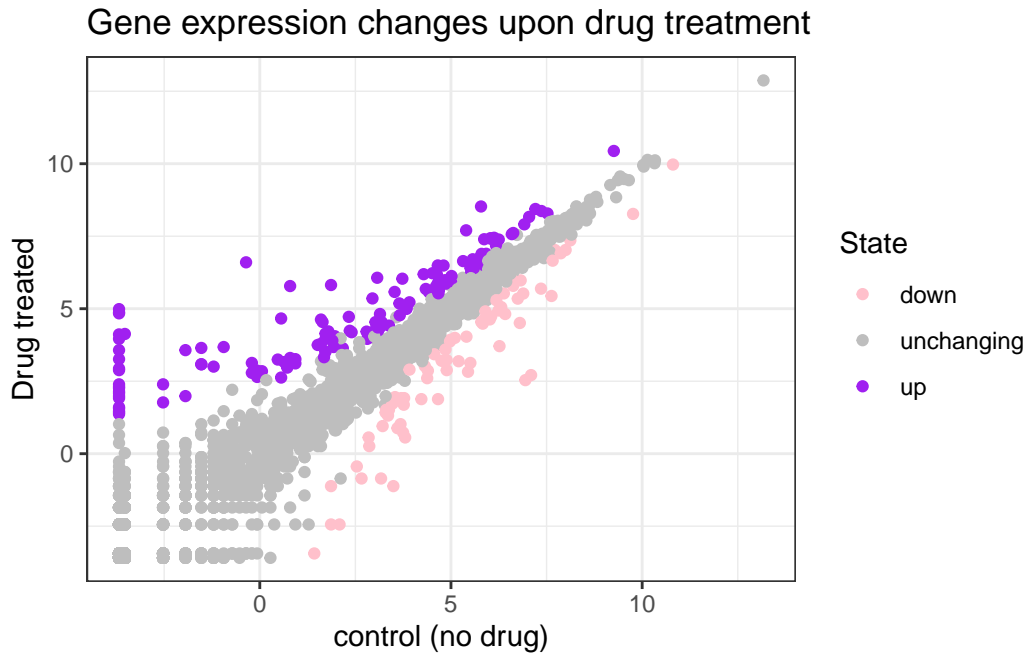
```
round( table(genes$State)/nrow(genes) * 100, 2 )
```

down	unchanging	up
1.39	96.17	2.44

Q. Make a first plot of this data

## Color scatterplot

```
ggplot(genes) +
  aes(x = Condition1, y = Condition2, col = State) +
  scale_color_manual( values = c("pink", "grey","purple")) +
  geom_point() +
  labs(title = "Gene expression changes upon drug treatment",
       x = "control (no drug)",
       y = "Drug treated") +
  theme_bw()
```



## Using different geoms

Let's plot some aspects of the in-built `mtcars` dataset.

```
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

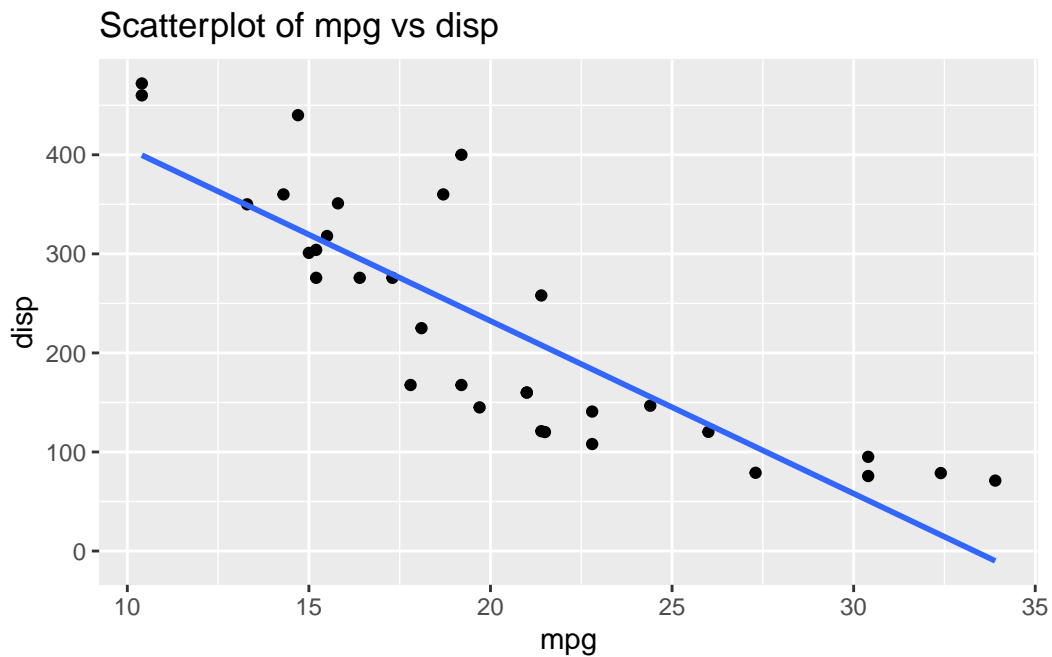
Hornet Sportabout	18.7	8	360	175	3.15	3.440	17.02	0	0	3	2
Valiant	18.1	6	225	105	2.76	3.460	20.22	1	0	3	1

Q. Scatterplot of mpg vs disp

```
p1 <- ggplot(mtcars) +
  aes(x = mpg, y = disp) +
  geom_point() +
  geom_smooth(se = FALSE, method = "lm") +
  labs(title = "Scatterplot of mpg vs disp", x = "mpg", y = "disp")

p1
```

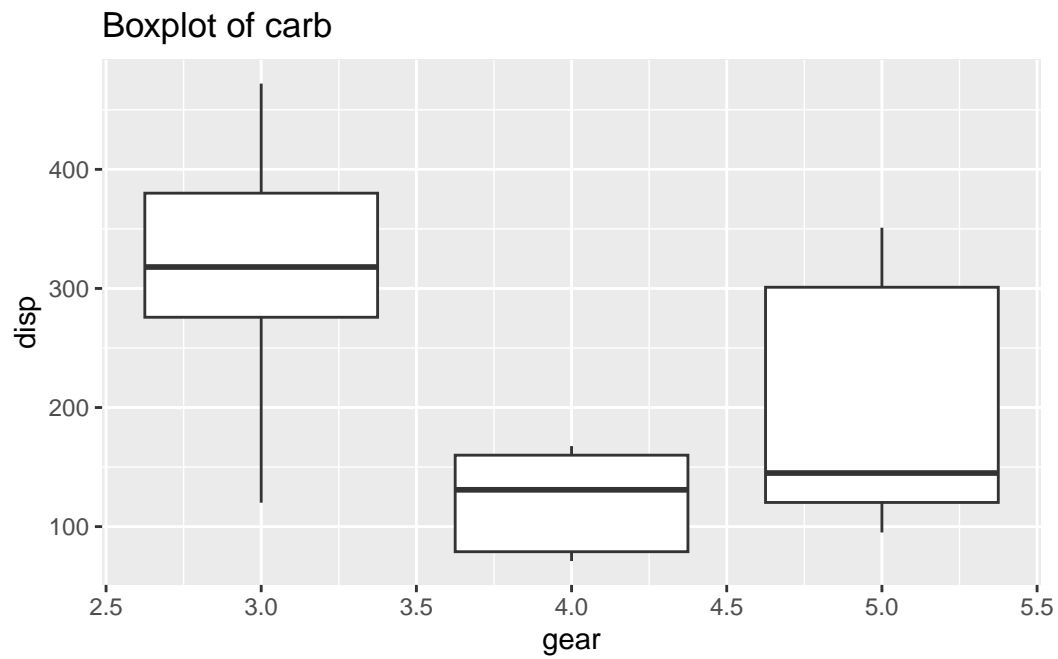
`geom\_smooth()` using formula = 'y ~ x'



Q. Boxplot of gear vs disp

```
p2 <- ggplot(mtcars) +
  aes(x = gear, y = disp, group = gear) +
  geom_boxplot() +
  labs(title = "Boxplot of carb", x = "gear", y = "disp")

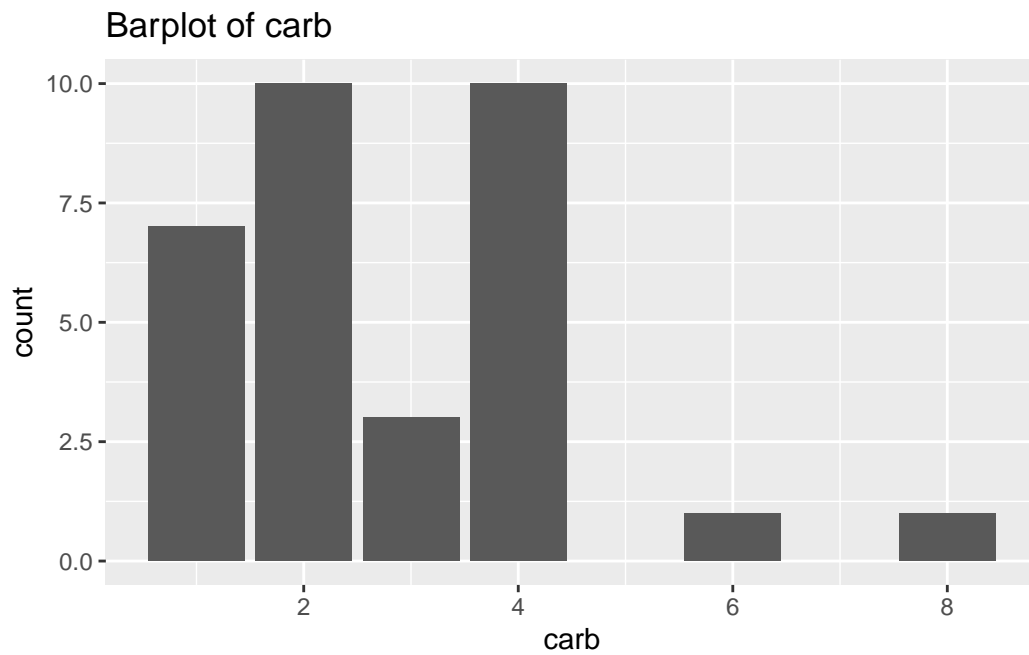
p2
```



Q. Barplot of carb

```
p3 <- ggplot(mtcars) +  
  aes(x = carb) +  
  geom_bar() +  
  labs(title = "Barplot of carb")
```

p3

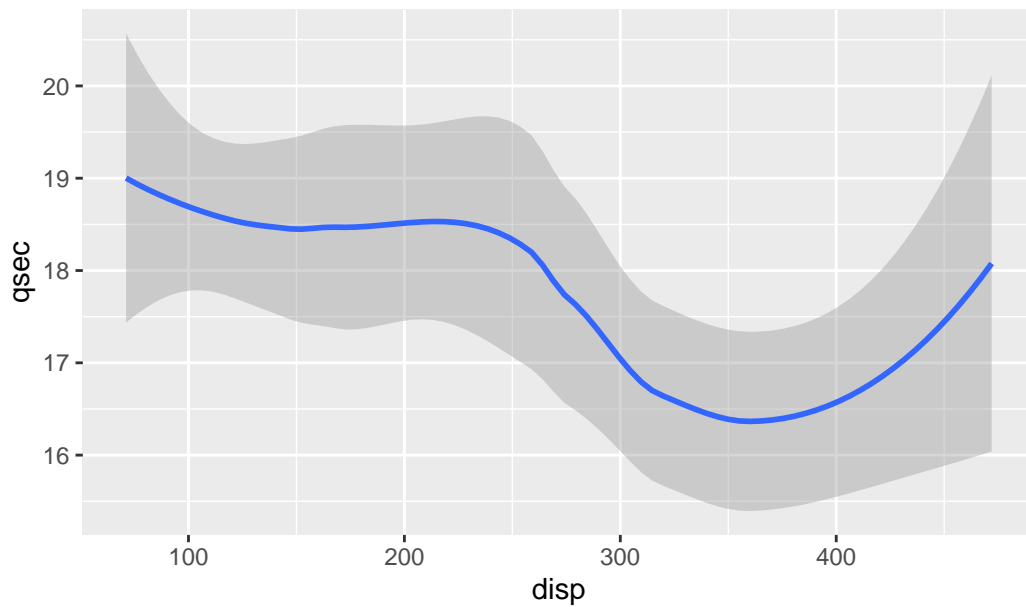


Q. Smooth of disp vs qsec

```
p4 <- ggplot(mtcars) +  
  aes ( x = disp, y = qsec) +  
  geom_smooth() +  
  labs(title = "Smoothplot of disp vs qsec", x = "disp", y = "qsec")  
  
p4
```

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

Smoothplot of disp vs qsec



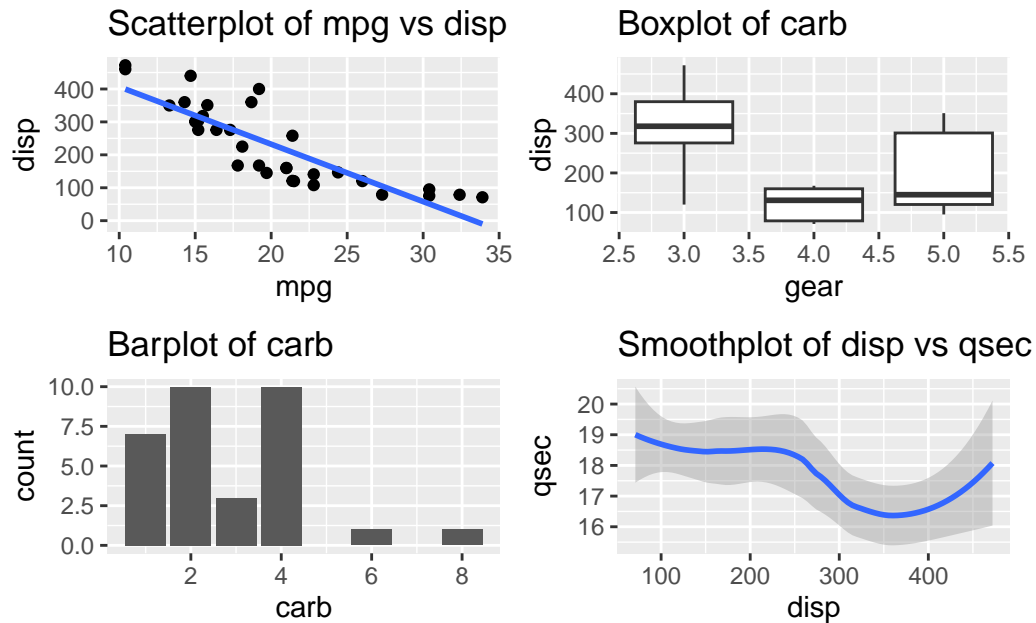
## Panel of plots

I want to combine all these plots into one figure with multiple pannels.

We can use the **patchwork** package to do this.

```
library(patchwork)
(p1 + p2 + p3 + p4)
```

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



```
ggsave(filename = "myplot.png", width = 10, height = 10 )
```

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

## Life Expectancy by country plot

Importing/Reading the file

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

	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

5	Afghanistan	Asia	1972	36.088	13079460	739.9811
6	Afghanistan	Asia	1977	38.438	14880372	786.1134

Q. How many countries are in this dataset?

```
length(table(gapminder$country))
```

```
[1] 142
```

## Gapminder plot

```
ggplot(gapminder) +
  aes(x = gdpPercap, y = lifeExp, col = continent, size = pop) +
  geom_point(alpha = 0.3) +
  facet_wrap(~continent) +
  theme_bw()
```

