

# Class 5: Data Viz with ggplot

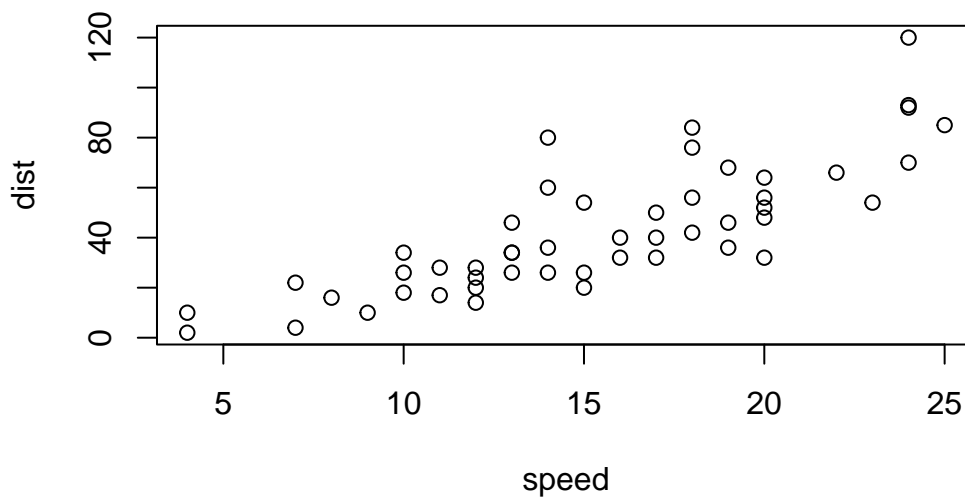
Safiya (PID:A18027139)

## Intro to ggplot

There are many graphics systems in R (ways to make plots and figures). These include “base” R plots. Today we will focus mostly on **ggplot2** package.

Let’s start with a plot of a simple in-built dataset called **cars**.

```
plot(cars)
```

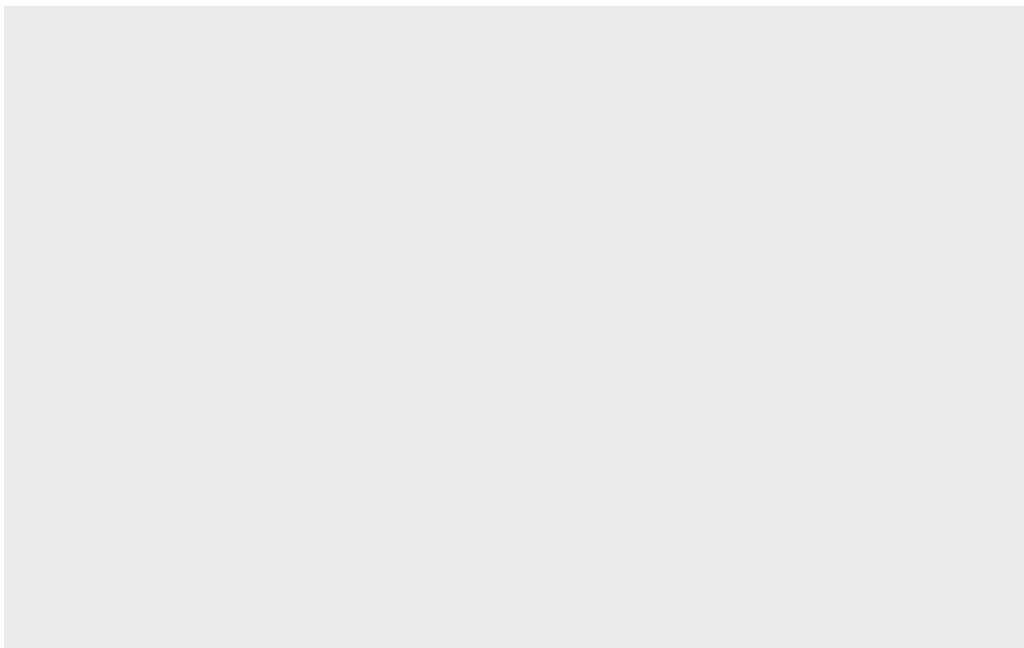


make this figure with **ggplot**. This package has to be installed on my computer. To install any R package I use the function `install.packages()`.

I will run ‘`install.packages(“ggplot2”)`’ in my R console not this quarto document.

Before I can use any functions from add-on packages I need to load the package from my “library()” with the `library(ggplot2)` call.

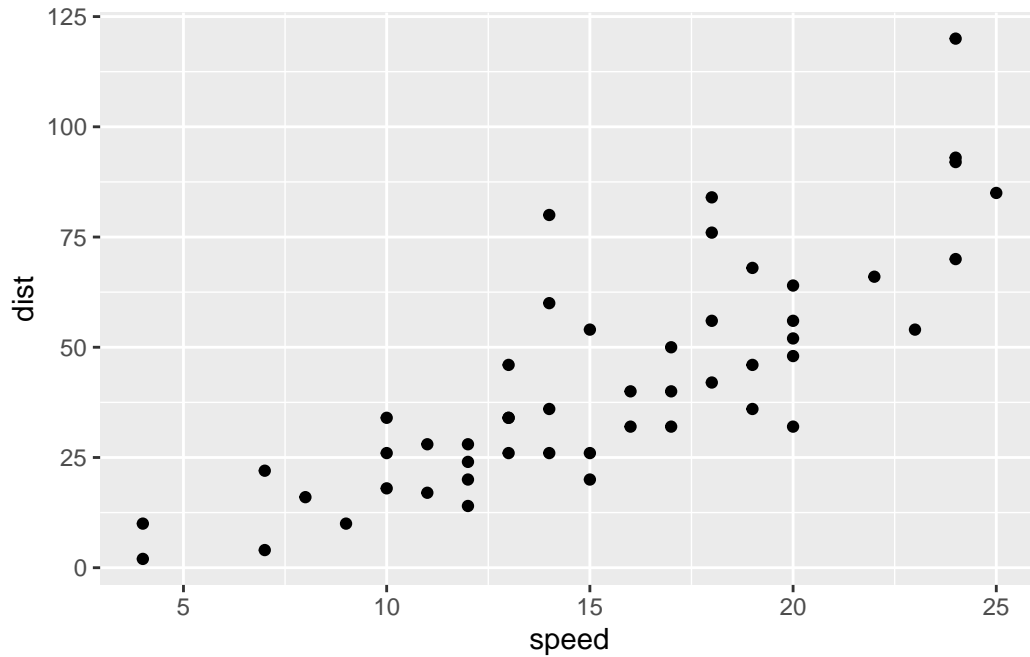
```
library(ggplot2)
ggplot(cars)
```



All ggplot figures have at least 3 things (called layers). These include:

-**data** (the input dataset I want to plot from), -**aes** (the aesthetic mapping of the data to my plot), -**geoms** (the `geom_point()`, `geom_line()` etc. that I want to draw).

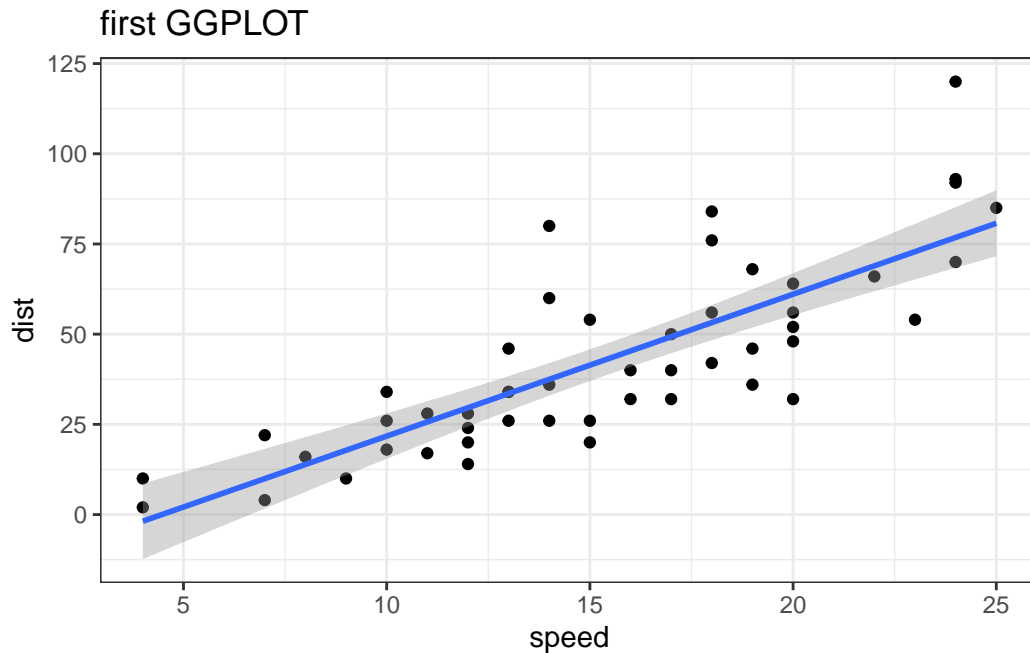
```
ggplot(cars) + aes(x=speed, y=dist) + geom_point()
```



Let's add a line to show the relationship here:

```
ggplot(cars) + aes(x=speed, y=dist) + geom_point() +  
  geom_smooth(method="lm") +  
  theme_bw() + labs(title="first GGPlot")
```

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



Which geometric layer should be used to create scatter plots in ggplot2?

Geom\_point

## gene expression figure

The code to read the dataset

```
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

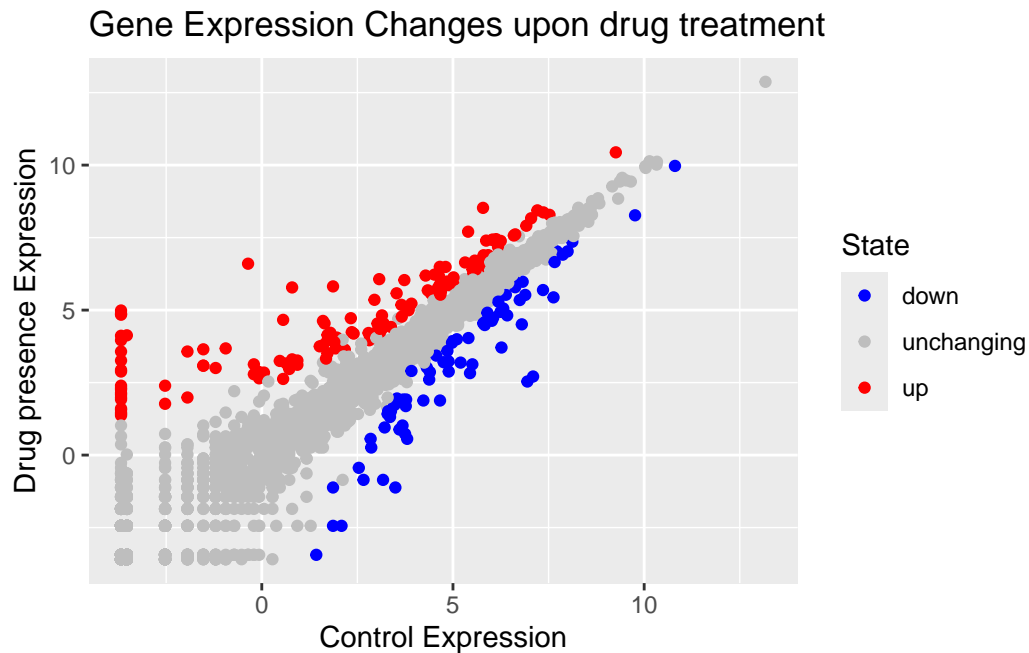
how many genes are in this dataset?

```
nrow(genes)
```

```
[1] 5196
```

First plot of this data set

```
ggplot(genes) +  
  aes(x=Condition1, y=Condition2, col=State) +  
  geom_point() + labs(title="Gene Expression Changes upon drug treatment",  
    x= "Control Expression",  
    y= "Drug presence Expression") +  
  theme_get() +  
  scale_colour_manual( values=c("blue","gray","red") )
```



Use the `table()` function on the `State` column of this data.frame to find out how many 'up' regulated genes there are. What is your answer?

```
round( table(genes$State)/nrow(genes), 4)
```

down	unchanging	up
0.0139	0.9617	0.0244

```
n.tot <- nrow(genes)
vals <- table(genes$State)

vals.percent <- vals/n.tot * 100
round(vals.percent, 2)
```

```
down  unchanging      up
1.39      96.17      2.44
```

Use the `colnames()` function and the `ncol()` function on the `genes` data frame to find out what the column names are (we will need these later) and how many columns there are. How many columns did you find?

```
colnames(genes)
```

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

## life Expectancy

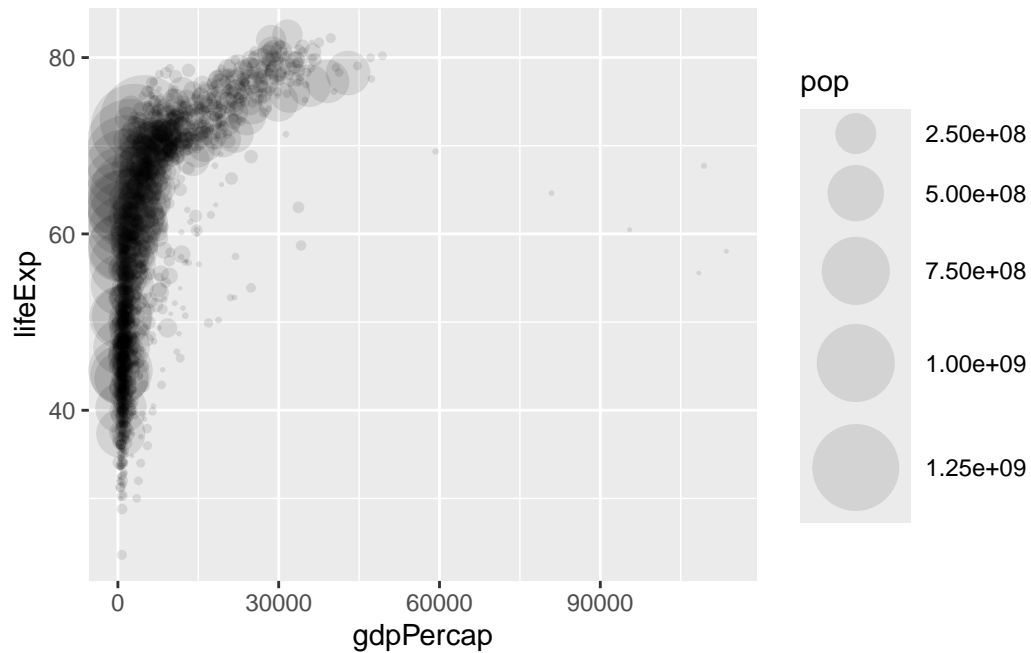
The code to read the dataset

```
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
```

```
ggplot(gapminder) +
  geom_point(aes(x= gdpPercap, y=lifeExp, size=pop), alpha=0.1) +
  scale_size_area(max_size = 15)
```



```
library(dplyr)
```

Attaching package: 'dplyr'

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

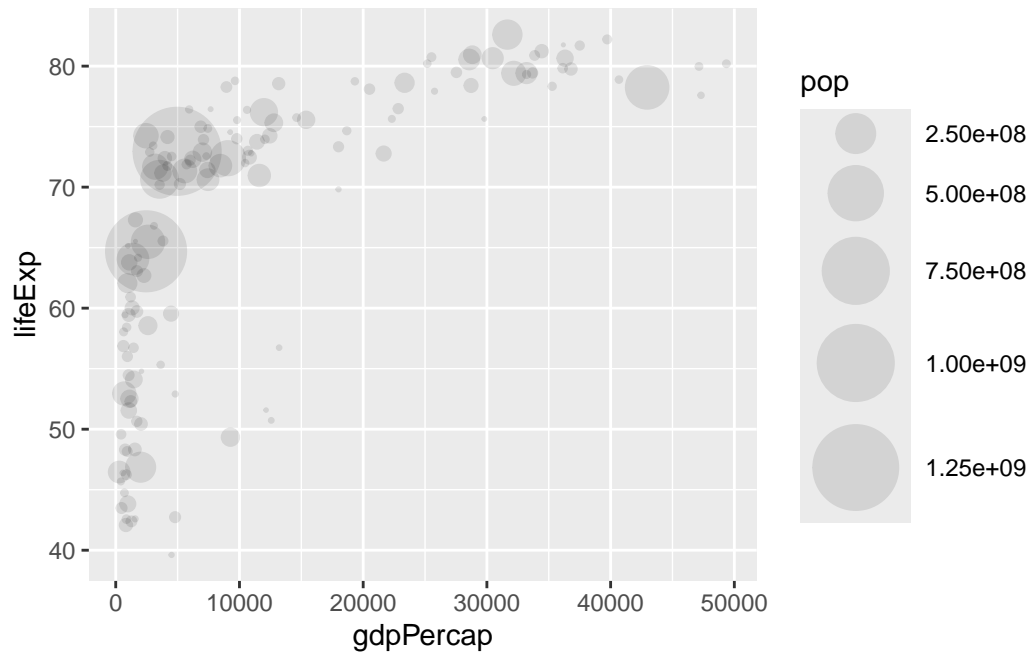
filter, lag

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

intersect, setdiff, setequal, union

```
gapmider_2007 <- filter(gapminder, year==2007)
```

```
ggplot(gapmider_2007) +  
  geom_point(aes(x= gdpPercap, y=lifeExp, size=pop), alpha=0.1) +  
  scale_size_area(max_size = 15)
```



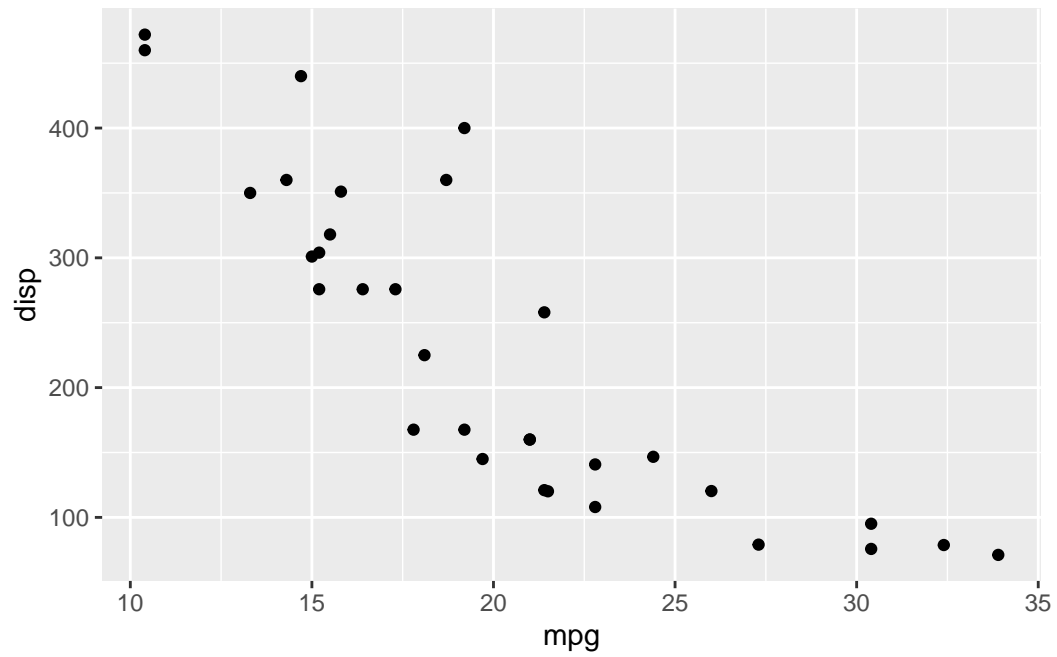
```
library(patchwork)
```

## Setup some example plots

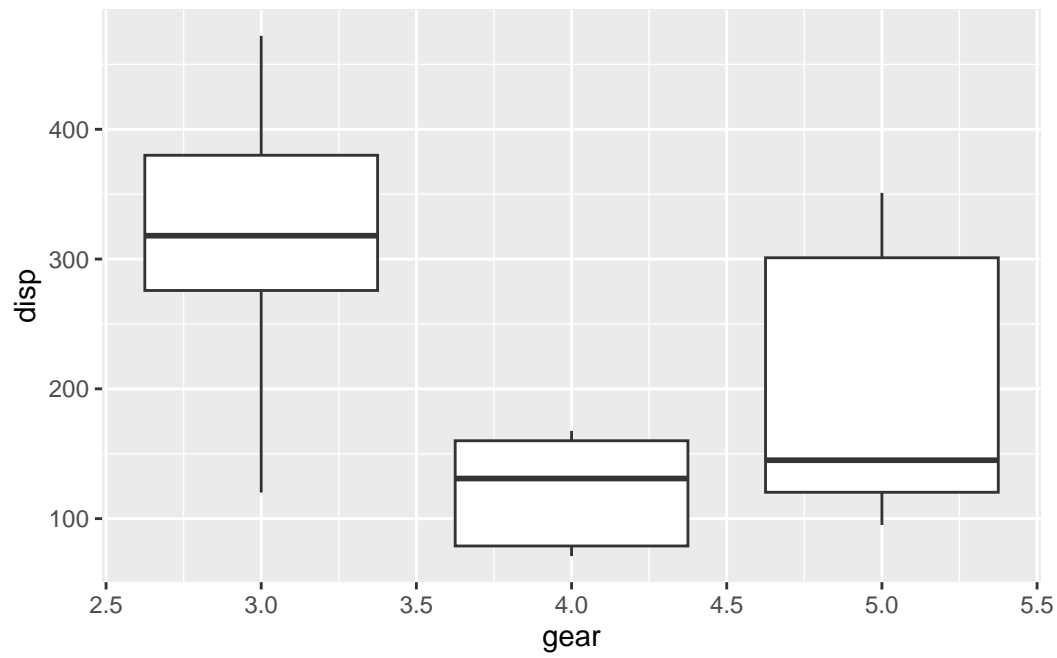
```
p1 <- ggplot(mtcars) + geom_point(aes(mpg, disp))
p2 <- ggplot(mtcars) + geom_boxplot(aes(gear, disp, group = gear))
p3 <- ggplot(mtcars) + geom_smooth(aes(disp, qsec))
p4 <- ggplot(mtcars) + geom_bar(aes(carb))
```

```
p1
```



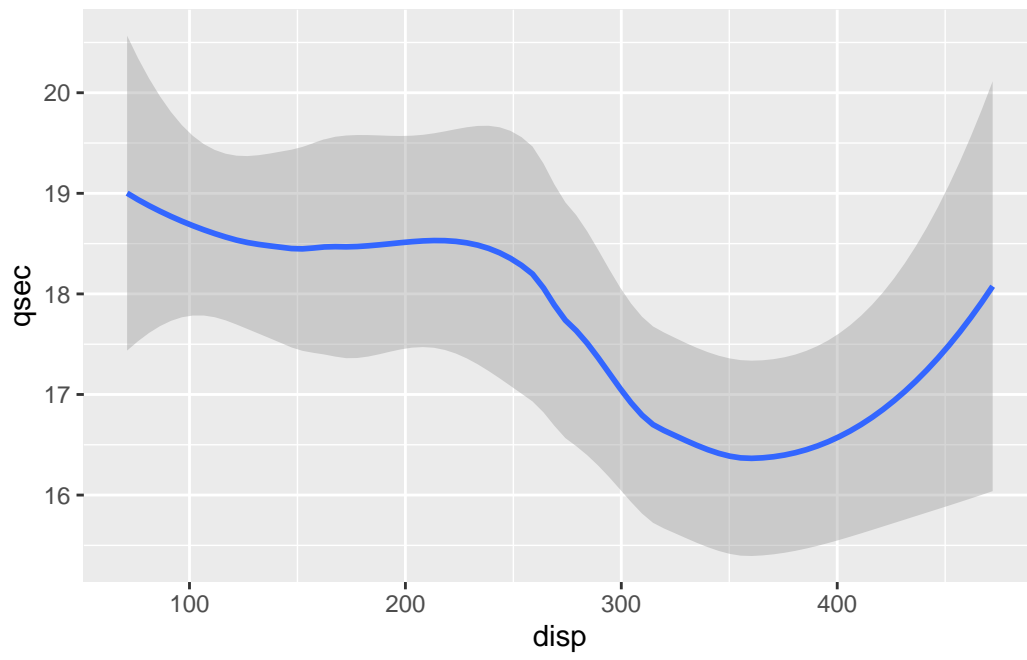


p2

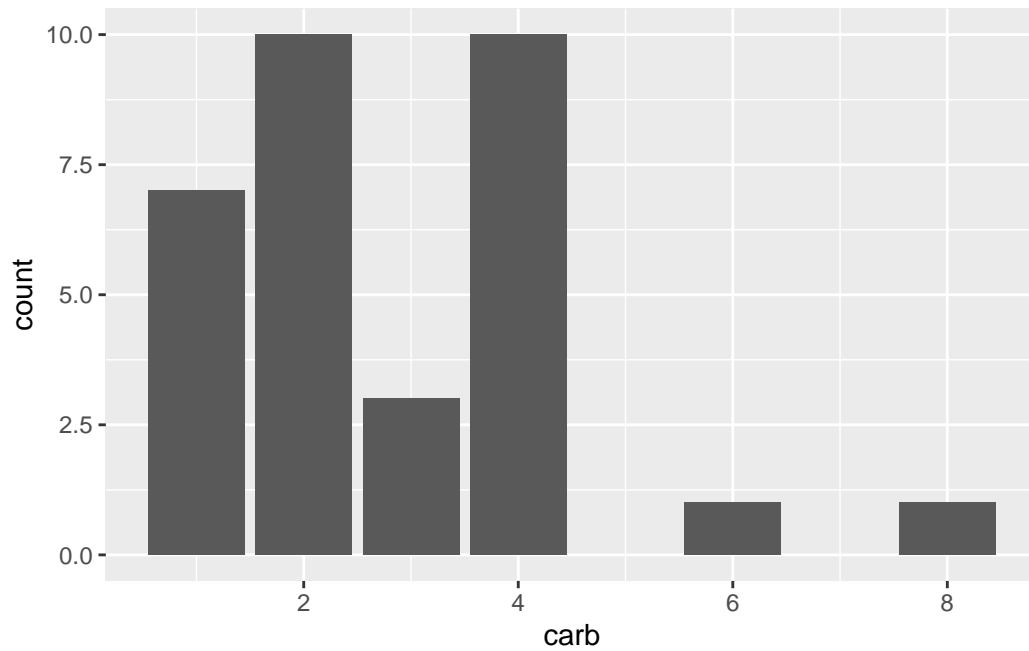


p3

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



p4



```
(p1 | p2 | p3) / p4
```

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

