Homework01

##Machine Learning Main Ideas

###Question 1. In supervised learning, the machine using the data which is labeled. It means some data is already tagged with the correct answer. It can be compared to learning which takes place in the presence of a supervisor or a teacher. We wish to fit a model that relates the response to the predictors, with the aim of accurately predicting the response for future observations (prediction) or better understanding the relationship between the response and the predictors (inference). Examples for this case are linear/logistic regression and SVM.

On the other hand, unsupervised learning is a machine learning technique, where it does not need to supervise the model. Instead, it need to allow the model to work on its own to discover information. It mainly deals with the unlabeled data. We can seek to understand the relationships between the variables or between the observations. Examples are KNN and PCA.

The main difference is that data has correct answer or not, which is same as response variable y.

###Question 2. Those models are used whether the data is continuous or categorical(quantitative or qualitative).

If the former, we would use regression model. It will predict exact value of response variable y, such as IQ, height or price. If the latter, we would classification model. It will predict the group of the observation, such as gender, country or color.

However, we can also use continuous variable as categorical by grouping. For example, less than IQ 100 are group 1, over are group 2.

###Question 4. Descriptive models: It is focused on a model which visually emphasize a trend in data. It could show line, scatter plot and hitogram.

Inferential models: It is focused to describe relationship between outcome and predictors, and figure out which features are important.

Predictive models: The main purpose is to predict Y with minimum reducible error. It will figure out what combination of features could explain and predict Y well.

###Question 5. 1) Those could be call as parametric and non-parametric. Mechanistic assume a parametric form for function ‘f’. It does not need to match true unknown ‘f’. It an add parameters to be more flexibility.

Empirically-driven has no assumption about ‘f’. It requires a larger number of observation. It is relatively more flexible by default.

However, both have an issue of over-fitting if the model has too many features.

2. I think mechanistic model is easier to understand. This is because, it can use even a few features, which means easy to interpret the model. If there are many features in the model, it is hard to clarify the meaning of function ‘f’. Therefore, it might be hard to understand empirically-driven model which requires a larger number of observations.

3. Over-fitting means low variance and high bias. Thus, Empirically-driven model, which is much more flexible, has low variance and high bias. The reason why is the high flexibility indicates the possibility of over-fitting. On the other hand, mechanistic model such as linear/logistic model are less flexible, so they have high variance and low bias.

###Question 6. 1) Predictive. The main purpose it to predict y, which is voted candidates.

2. Inferential. This is because, we want to know about the significant of features such as contact with the candidate.

##Exploratory Data Analysis

###Exercise 1.

```
install.packages("ggplot2", repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/d3/ym7c6dt10vj79qz8tpjm58z80000gn/T//RtmpsV1sgd/downloaded_packages
```

```
library(ggplot2)
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'tibble'
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'pillar'
```

```
install.packages("tidyverse", repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/d3/ym7c6dt10vj79qz8tpjm58z80000gn/T//RtmpsV1sgd/downloaded_packages
```

```
library(tidyverse)
```

```
## Warning: replacing previous import 'lifecycle::last_warnings' by
## 'rlang::last_warnings' when loading 'hms'
```

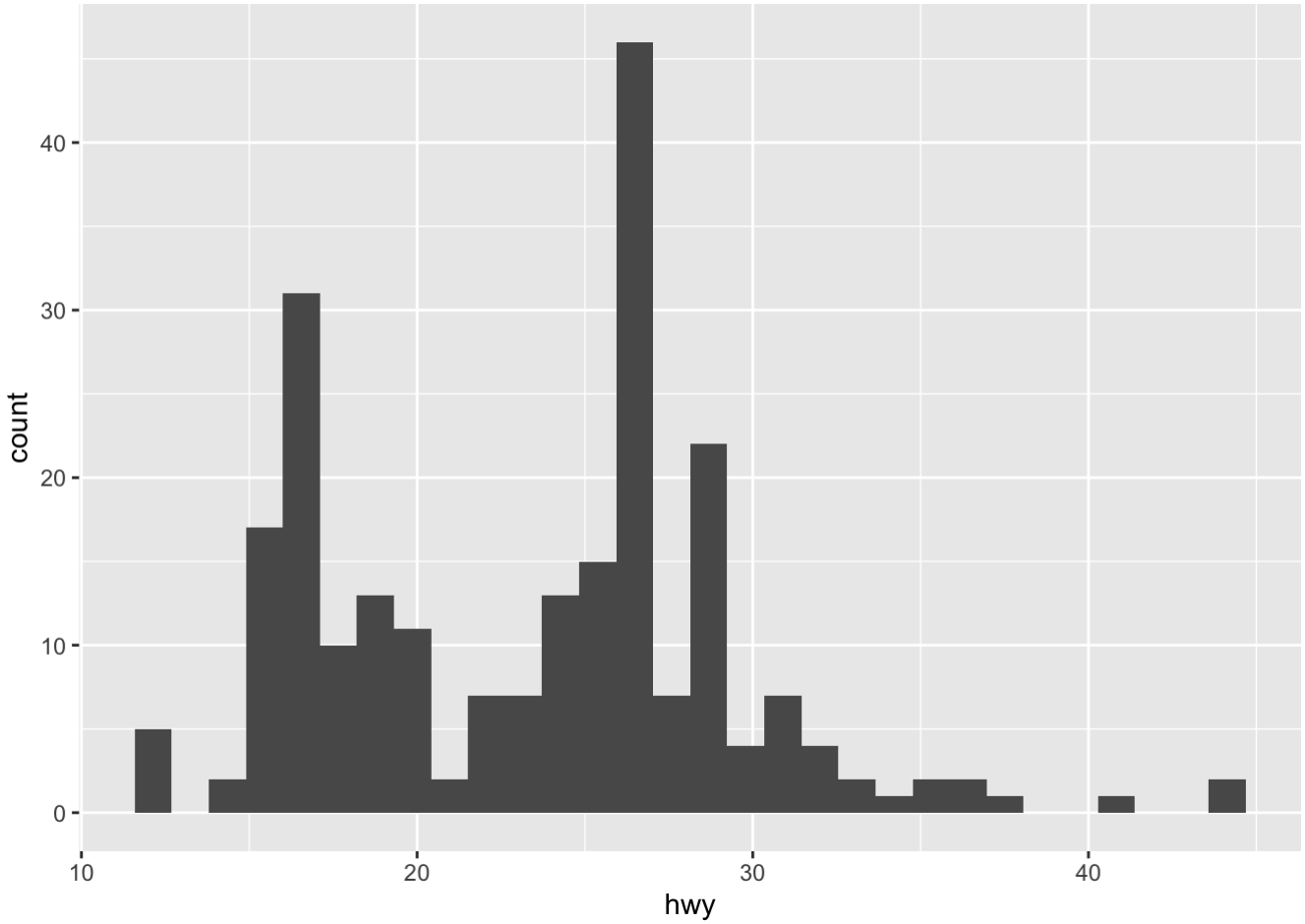
```
## ── Attaching packages ─────────────────────────────────────── tidyv
erse 1.3.1 ──
```

```
## ✓ tibble  3.1.0     ✓ dplyr   1.0.6
## ✓ tidyr   1.1.3     ✓ stringr 1.4.0
## ✓ readr   1.4.0     ✓ forcats 0.5.1
## ✓ purrr   0.3.4
```

```
## ── Conflicts ──────────────────────────────────────────────────────── tidyverse_
conflicts() ──
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
ggplot(mpg, aes(x = hwy)) +
        geom_histogram()
```

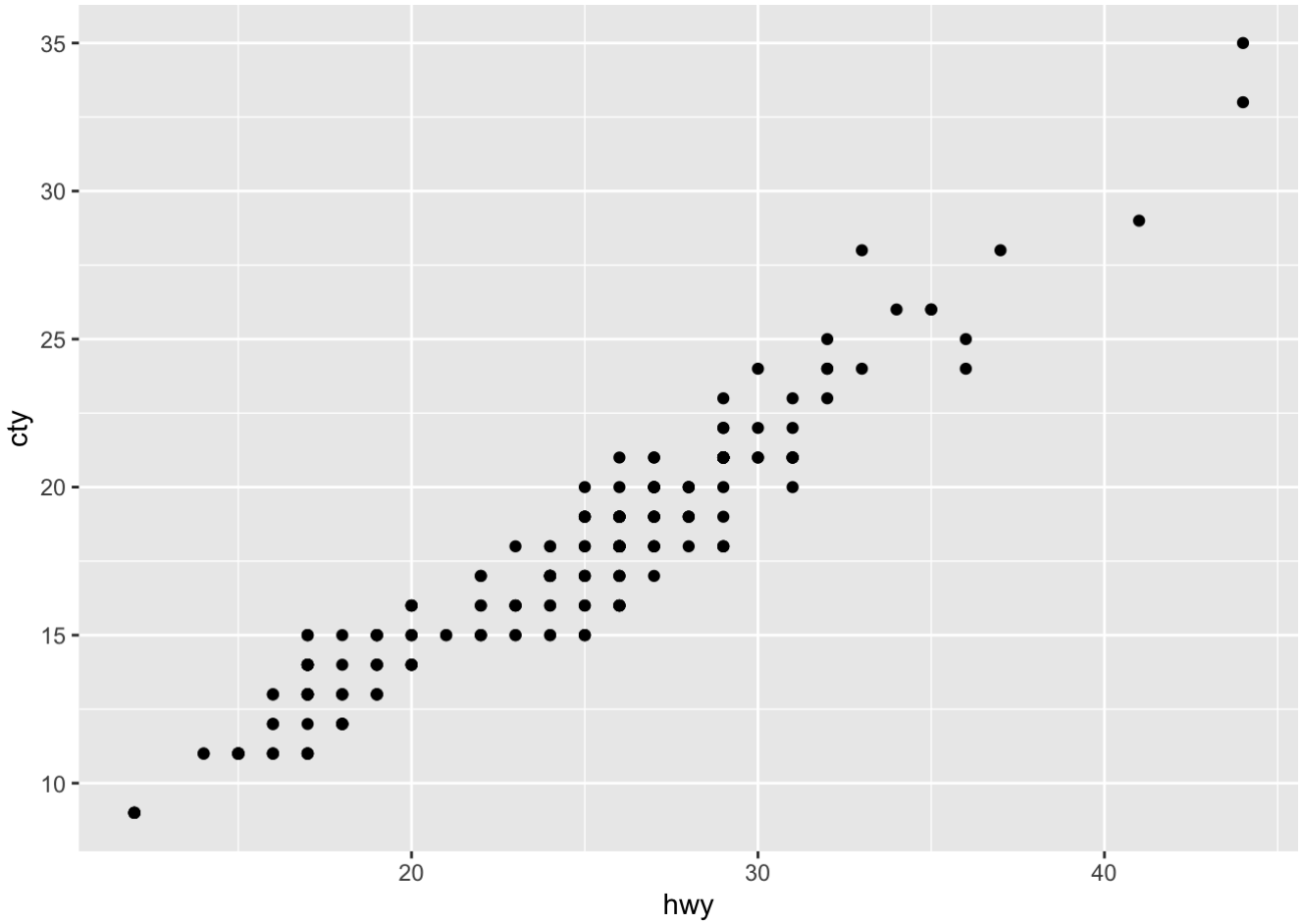
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



Most cars can travel less than 30 miles. It is distributed around 16/17 and 26/27 - bimodal.

###Exercise 2.

```
ggplot(mpg, aes(x = hwy, y = cty)) +
        geom_point()
```



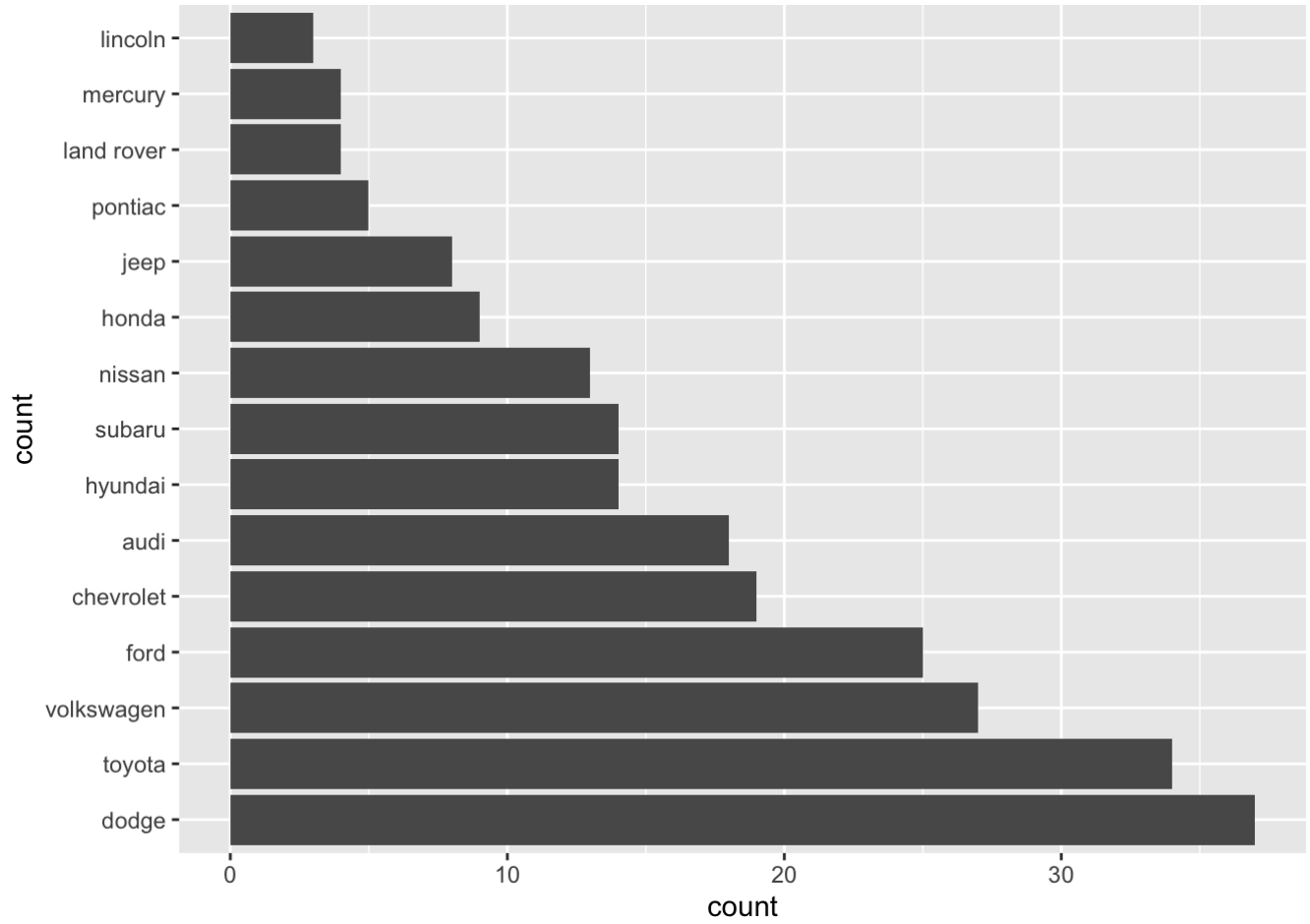
Intuitively, they have positive corelation, because it runs fine whether on the cty or the hwy. Moreover, most cars are travel less than 23 cty, similar result as in the previous question.

###Exercise 3.

```
install.packages("forcats", repos = "http://cran.us.r-project.org")
```

```
##
## The downloaded binary packages are in
##  /var/folders/d3/ym7c6dt10vj79qz8tpjm58z80000gn/T//RtmpsV1sgd/downloaded_packages
```

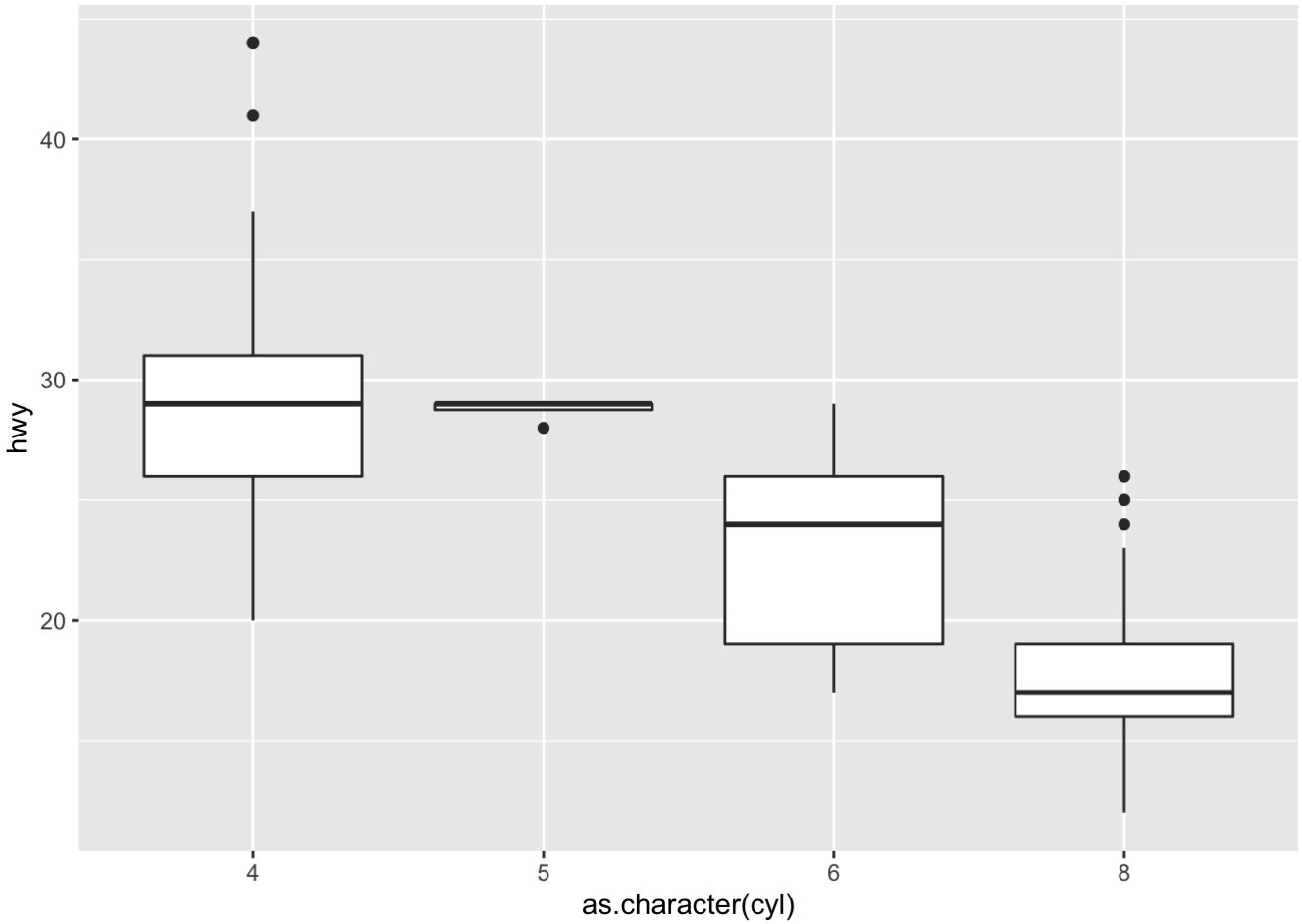
```
library(forcats)
library(ggplot2)
library(tidyverse)

mpg %>%
  ggplot(aes(x = fct_infreq(manufacturer))) +
  geom_bar() + coord_flip() +
  labs(x = "count")
```



Dodge produced the most cars and Lincoln produced the least cars.

###Exercise 4.

```
ggplot(mpg, aes(x = as.character(cyl), y = hwy)) +
        geom_boxplot()
```



If a car have less cyl, it could travel more. If a car have more cyl, it could travel less. The lowest and highest value of cyl have outliers.

###Exercise 5.

```
install.packages("corrplot", repos = "http://cran.us.r-project.org")
```

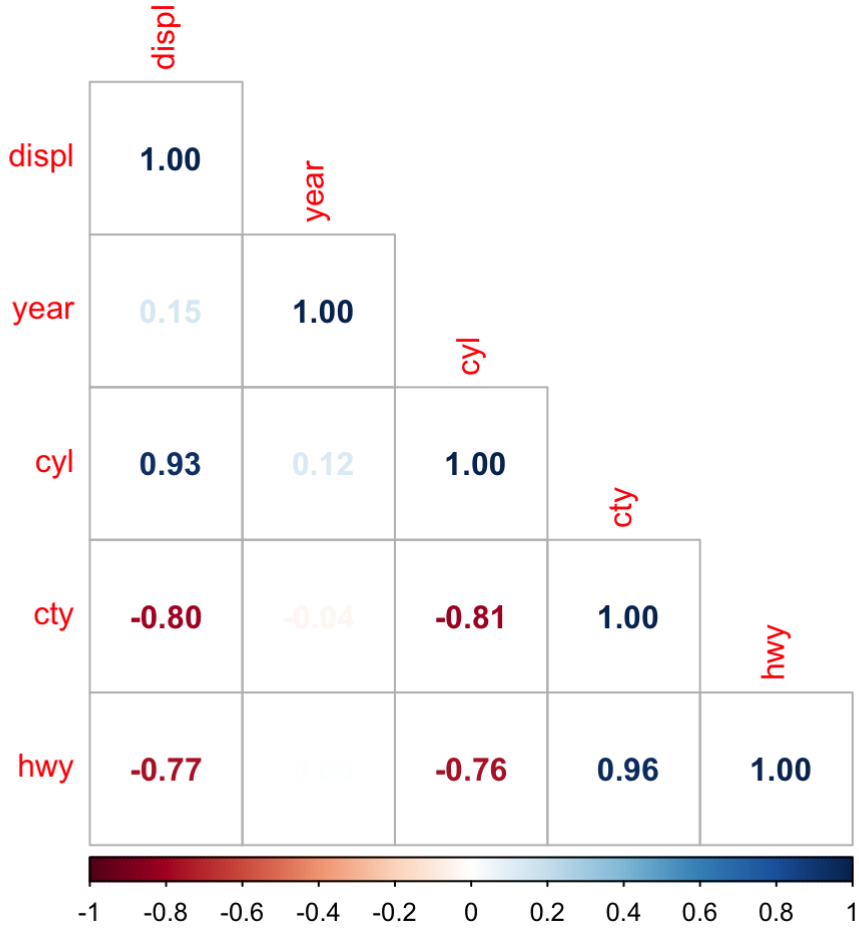
```
##
##   There is a binary version available but the source version is later:
##          binary source needs_compilation
## corrplot   0.89   0.92             FALSE
```

```
## installing the source package 'corrplot'
```

```
library(corrplot)
```

```
## corrplot 0.92 loaded
```

```
M = cor(mpg %>% dplyr::select(where(is.numeric)))
corrplot(M, method = "number", type = "lower")
```



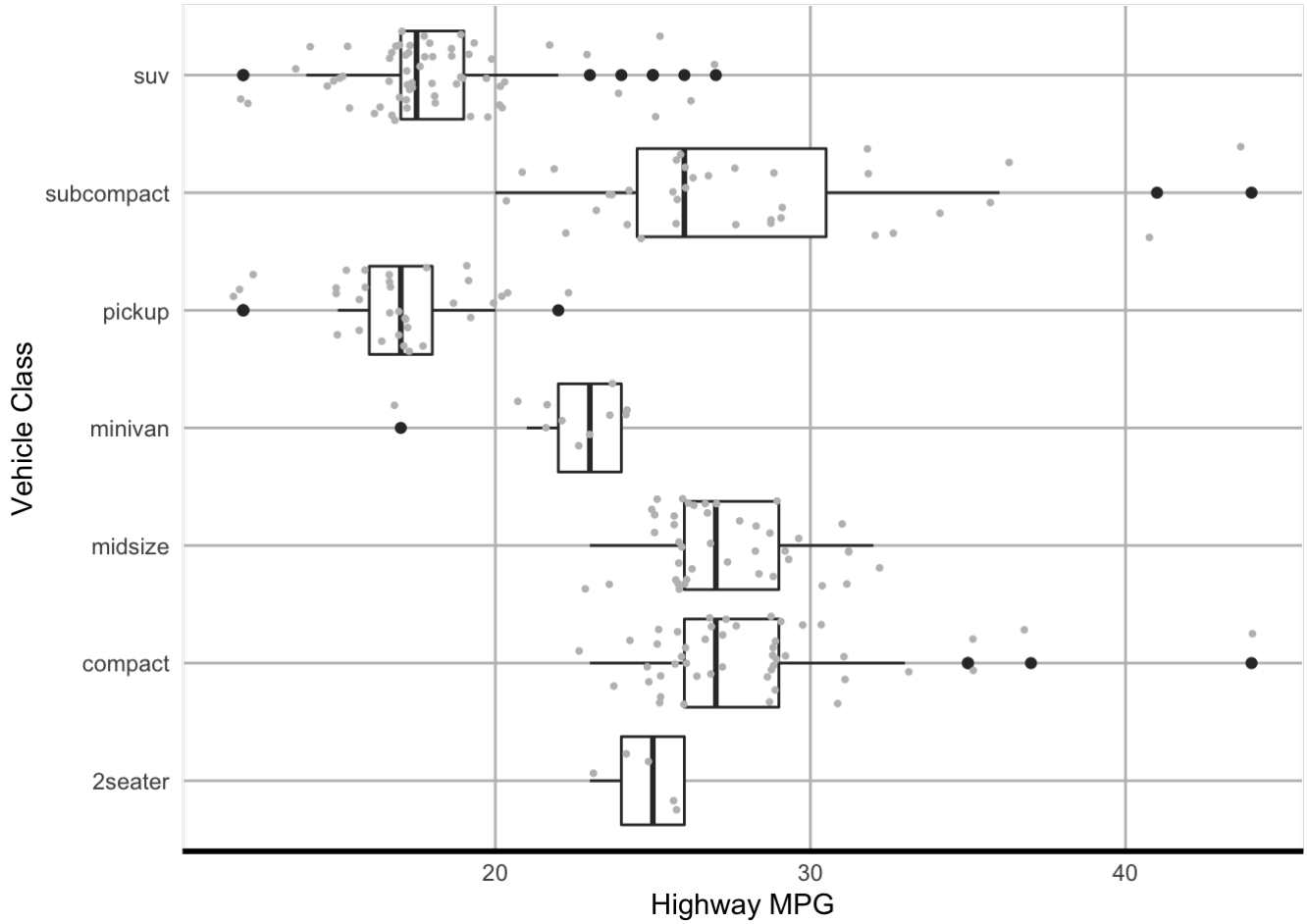
hwy-cty, cyl-displ are strongly positively correlated and year-displ, cyl-year are weakly correlated. On the other hand, cty-displ, hwy-displ, cty-cyl, hwy-cyl are strongly negatively correlated and cty-year are weakly correlate. hwy-year seems independent. These are expected results.

###Exercise 6.

```
install.packages('ggthemes', repos = "http://cran.us.r-project.org")
```

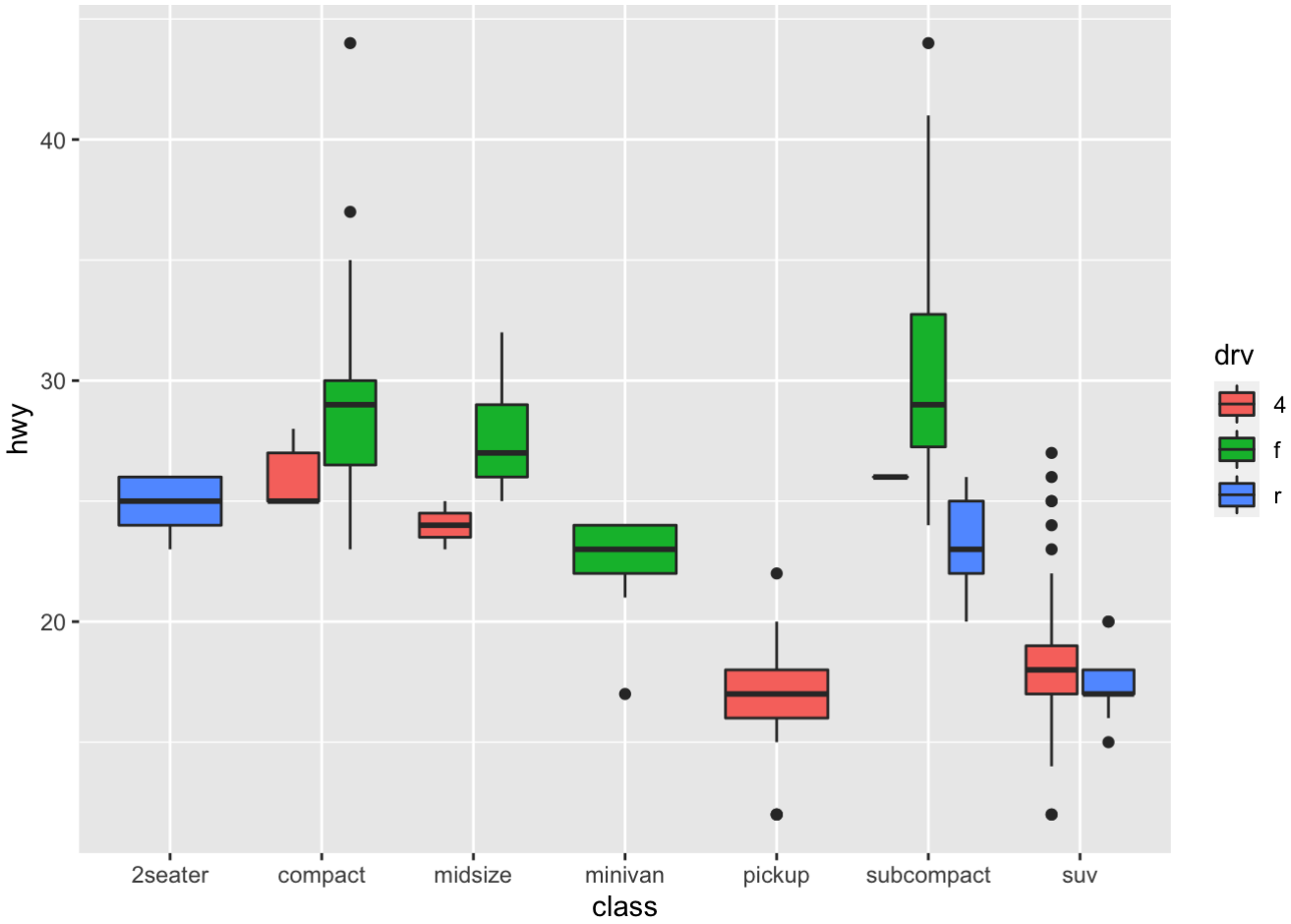
```
##
## The downloaded binary packages are in
##  /var/folders/d3/ym7c6dt10vj79qz8tpjm58z80000gn/T//RtmpsV1sgd/downloaded_packages
```

```
library(ggthemes)

ggplot(mpg, aes(x = hwy, y = class)) +
        geom_boxplot() +
        labs(x = 'Highway MPG', y = 'Vehicle Class') +
        geom_jitter(color = "grey", size = 0.7, alpha = 1) +
        theme(axis.ticks = element_blank()) +
        theme(axis.line.x = element_line(size = 1, colour = "black")) +
        theme(panel.border = element_rect(colour = 'white', fill = NA)) +
        theme(panel.background = element_rect(fill = "white", colour = "grey50")) +
        theme(panel.grid.major = element_line(colour = 'grey'))
```



###Exercise 7.

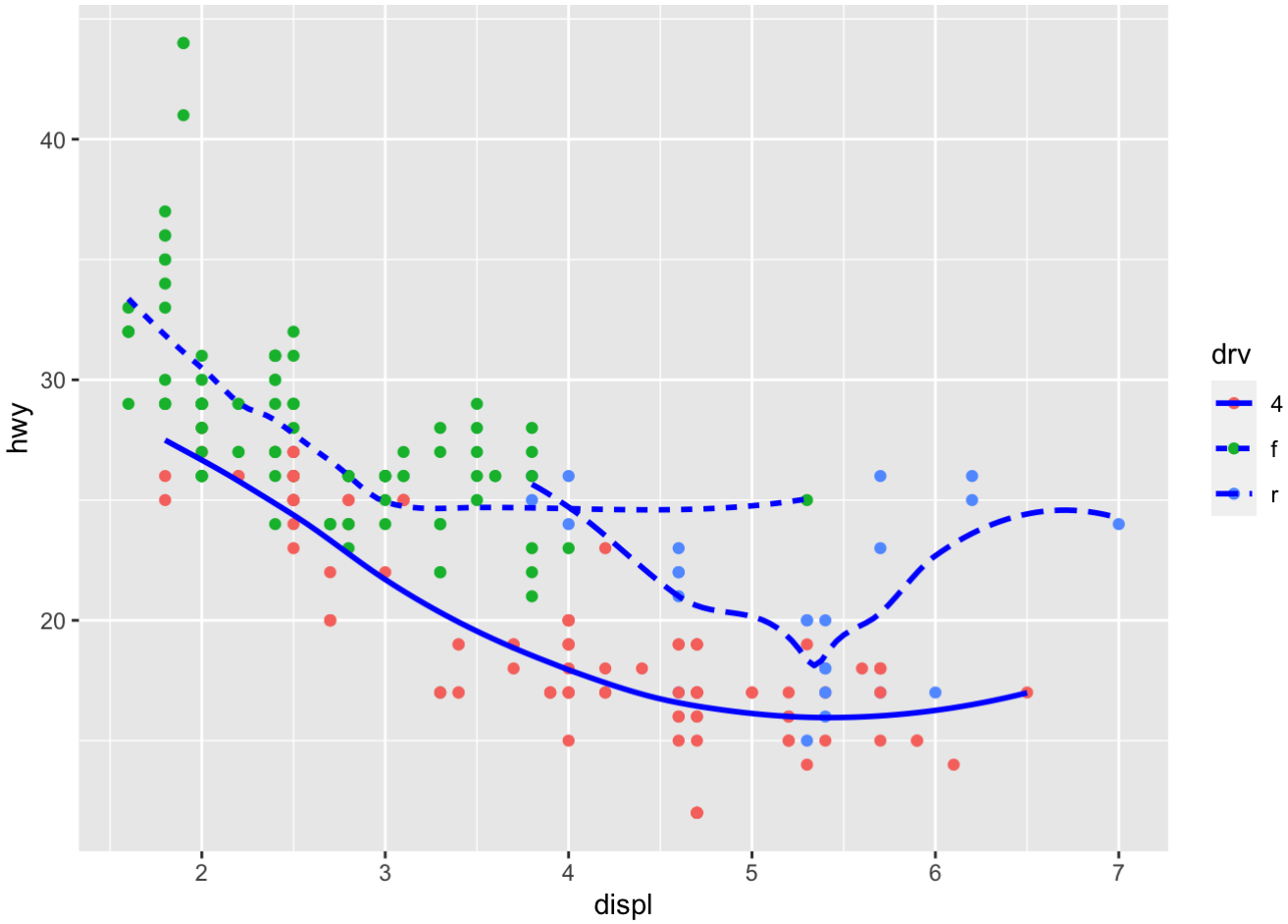
```
ggplot(mpg, aes(x = class, y = hwy, fill = drv)) +
        geom_boxplot()
```



###Exercise 8.

```
ggplot(mpg, aes(x = displ, y = hwy, color = drv)) +
        geom_point() +
        geom_smooth(aes(linetype = drv), color = 'blue', se = FALSE)
```

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


hwy
40
30
20
2
3
4
5
6
7
displ
drv
4
f
r