Stat 437 HW1

John Salmon (011745357)

General rule

Please show your work and submit your computer codes in order to get points. Providing correct answers without supporting details does not receive full credits. This HW covers:

- The basics of `dplyr`
- Creating scatter plot using `ggplot2`
- Elementary Visualizations (via ggplot2): density plot, histogram, boxplot, barplot, pie chart
- Advanced Visualizations via ggplot2: faceting, annotation

For an assignment or project, you DO NOT have to submit your answers or reports using typesetting software. However, your answers must be well organized and well legible for grading. Please upload your answers in a document to the course space. Specifically, if you are not able to knit a .Rmd/.rmd file into an output file such as a .pdf, .doc, .docx or .html file that contains your codes, outputs from your codes, your interpretations on the outputs, and your answers in text (possibly with math expressions), please organize your codes, their outputs and your answers in a document in the format given below:

```
Problem or task or question ...
Codes ...
Outputs ...
Your interpretations ...
```

It is absolutely not OK to just submit your codes only. This will result in a considerable loss of points on your assignments or projects.

Problem 1

Please refer to the NYC flight data `nycflights13` that has been discussed in the lecture notes and whose manual can be found at https://cran.r-project.org/web/packages/nycflights13/index.html. We will use `flights`, a tibble from `nycflights13`.

You are interested in looking into the average `arr_delay` for 6 different `month` 12, 1, 2, 6, 7 and 8, for 3 different `carrier` “UA”, “AA” and “DL”, and for `distance` that are greater than 700 miles, since you suspect that colder months and longer distances may result in longer average arrival delays. Note that you need to extract observations from `flights` and obtain the needed sample means for `arr_delay`, and that you are required to use `dplyr` for this purpose.

```r
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
```

```r
library(nycflights13)
flights1 = flights %>%
    filter(month %in% c(12, 1, 2, 6, 7, 8), carrier %in% c("UA",
        "AA", "DL"), distance > 700)  #extract observations
na.omit(flights1)
```

```
## # A tibble: 62,188 x 19
##     year month   day dep_time sched_dep_time dep_delay arr_time sched_arr_time
##    <int> <int> <int>    <int>          <int>     <dbl>    <int>          <int>
##  1  2013     1     1      517            515         2      830            819
##  2  2013     1     1      533            529         4      850            830
##  3  2013     1     1      542            540         2      923            850
##  4  2013     1     1      554            600        -6      812            837
##  5  2013     1     1      554            558        -4      740            728
##  6  2013     1     1      558            600        -2      753            745
##  7  2013     1     1      558            600        -2      924            917
##  8  2013     1     1      558            600        -2      923            937
##  9  2013     1     1      559            600        -1      941            910
## 10  2013     1     1      559            600        -1      854            902
## # i 62,178 more rows
## # i 11 more variables: arr_delay <dbl>, carrier <chr>, flight <int>,
## #   tailnum <chr>, origin <chr>, dest <chr>, air_time <dbl>, distance <dbl>,
## #   hour <dbl>, minute <dbl>, time_hour <dttm>
```

The following tasks and questions are based on the extracted observations.

(1.a) In a single plot, create a density plot for `arr_delay` for each of the 6 months with `color` aesthetic designated by `month`. Note that you need to convert `month` into a factor in order to create the plot. What can you say about the average `arr_delay` across the 6 months?

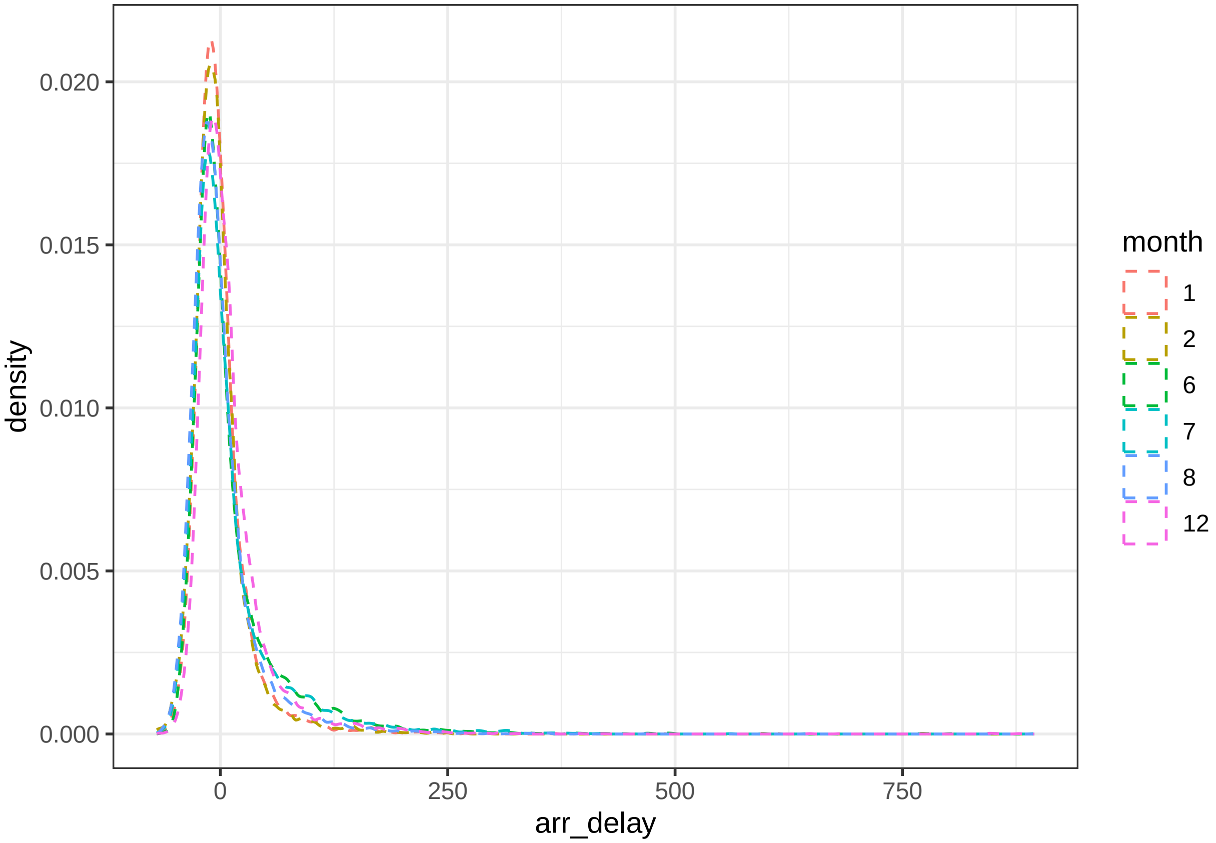
```
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.3.2
```

```
flights1 = flights1 %>%
    dplyr::mutate_at("month", as.factor)

arr_delay.density_plot = ggplot(flights1, aes(x = arr_delay,
    color = month, na.rm = TRUE)) + geom_density(linetype = "dashed") +
    theme_bw()
arr_delay.density_plot
```

```
## Warning: Removed 1307 rows containing non-finite outside the scale range
## (`stat_density()`).
```



Based on this density plot, month 12 appears to have a slightly higher occurrence of arrival delays compared to the other months. This is inferred by examining the peak (or mode) which is slightly greater 0 on the x axis. The other months all have similar modes centered much closer to 0 although the frequency (or number of values in each mode) differs between months. As for the average, there is no precice information on the mean arrival delay that can be gathered from a density plot.

(1.b) In a single plot, create a boxplot for `arr_delay` for each of the 3 carriers. What can you say about the average `arr_delay` for the 3 carriers?

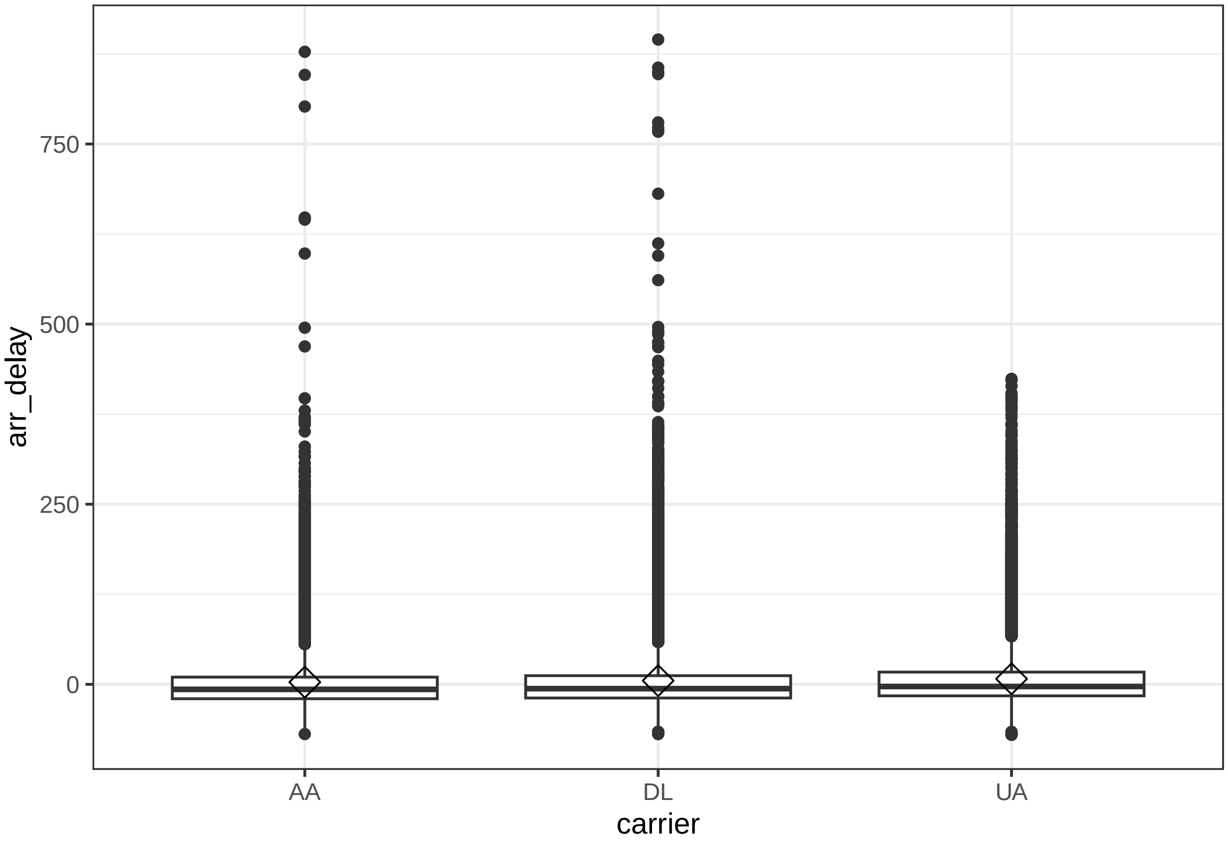
```
arr_delay.box_plot = ggplot(flights1, aes(x = carrier, y = arr_delay)) +
    geom_boxplot() + theme_bw() + stat_summary(fun.y = mean,
    geom = "Point", shape = 23, size = 4)
```

```
## Warning: The ‘fun.y‘ argument of ‘stat_summary()‘ is deprecated as of ggplot2 3.3.0.
## i Please use the ‘fun‘ argument instead.
## This warning is displayed once every 8 hours.
## Call ‘lifecycle::last_lifecycle_warnings()‘ to see where this warning was
## generated.
```

```
arr_delay.box_plot
```

```
## Warning: Removed 1307 rows containing non-finite outside the scale range
## (‘stat_boxplot()‘).
```

```
## Warning: Removed 1307 rows containing non-finite outside the scale range
## (‘stat_summary()‘).
```



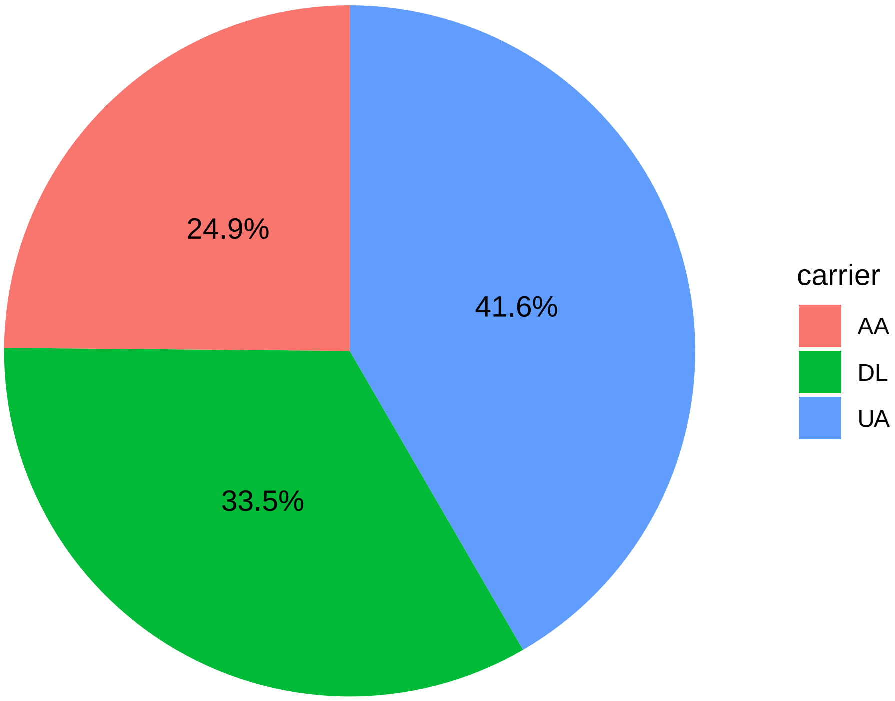
Based on the box plots, it can be said that the mean arrival delay of American Airlines and Delta are both very similar for the 6 specified winter months. While the mean arrival delay of United Airlines appears to be ever-so-slightly higher than the other two airlines. This suggests that United has more delay on average than American and Delta for these months.

(1.c) Create a pie chart for the 3 carriers where the percentages are the proportions of observations for each carrier and where percentages are superimposed on the sectors of the pie chart disc.

```
library(scales)
# get percentages for bar(pie) chart
flights2 = flights1 %>%
    group_by(carrier) %>%
    count() %>%
    ungroup() %>%
    mutate(percentage = n/sum(n)) %>%
    arrange(desc(carrier))
flights2$labels <- scales::percent(flights2$percentage)
flights2
```

```
## # A tibble: 3 x 4
##   carrier     n percentage labels
##   <chr>   <int>      <dbl> <chr>
## 1 UA       26437       0.416 41.6%
## 2 DL       21272       0.335 33.5%
## 3 AA       15786       0.249 24.9%
```

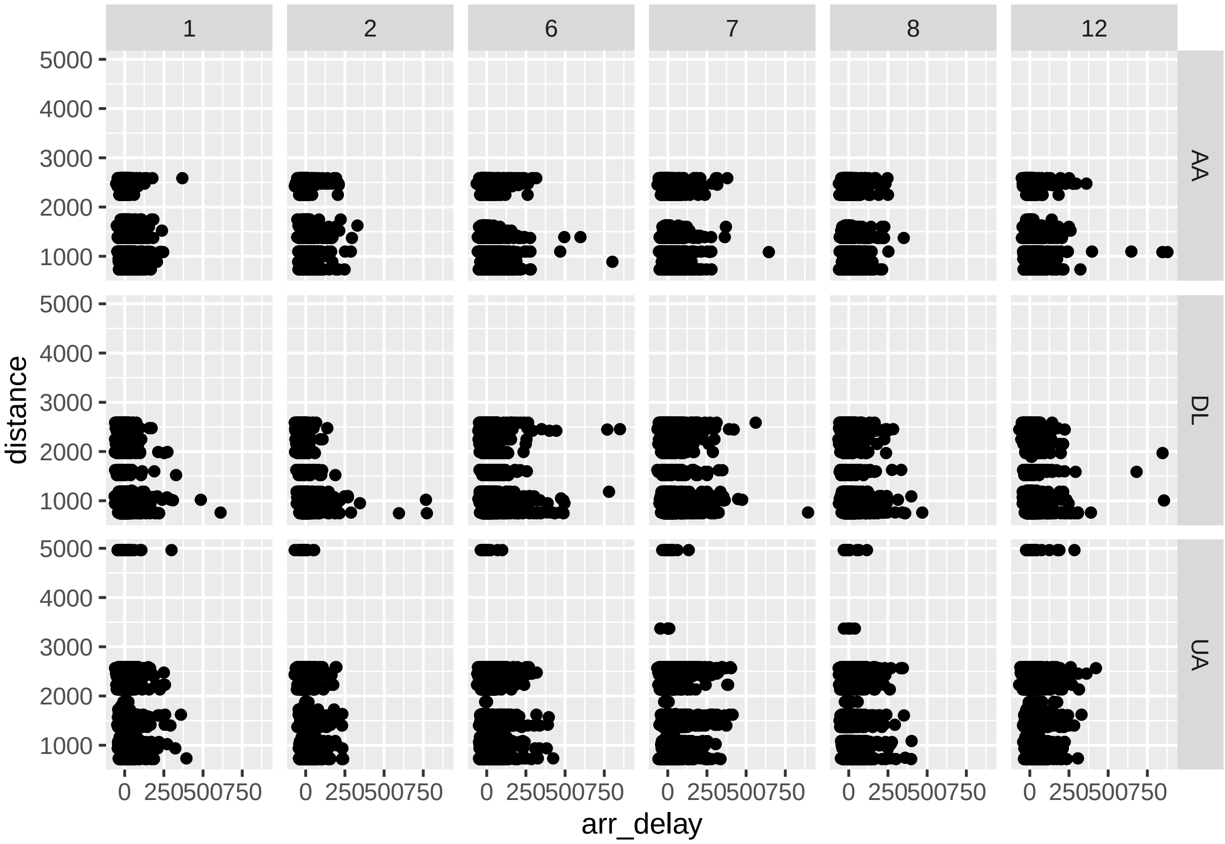
```
# construct pie chart
arr_delay.pie = ggplot(flights2) + geom_bar(aes(x = "", y = percentage,
    fill = carrier), stat = "identity", width = 1) + coord_polar("y",
    start = 0) + theme_void() + geom_text(aes(x = 1, y = cumsum(percentage) -
    percentage/2, label = labels))
arr_delay.pie
```



(1.d) Plot `arr_delay` against `distance` with `facet_grid` designated by `month` and `carrier`.

```
plot_template = ggplot(data = flights1) + geom_point(aes(x = arr_delay,
    y = distance))
arr_delay.facet = plot_template + facet_grid(carrier ~ month,
    scale = "fixed")
arr_delay.facet
```

```
## Warning: Removed 1307 rows containing missing values or values outside the scale rang
## (‘geom_point()‘).
```



(1.e) For each feasible combination of values of `month` and `carrier`, compute the sample average of `arr_delay` and save them into the variable `mean_arr_delay`, and compute the sample average of `distance` and save these averages into the variable `mean_distance`. Plot `month` against `mean_arr_delay` with `shape` designated by `carrier` and `color` by `mean_distance` and annotate each point by its associated `carrier` name.

```
summary_stats = flights1 %>%
    group_by(month, carrier) %>%
    summarise(mean_arr_delay = mean(arr_delay), mean_distance = mean(distance))
```

```
## 'summarise()' has grouped output by 'month'. You can override using the
## '.groups' argument.
```

```
summary_stats
```

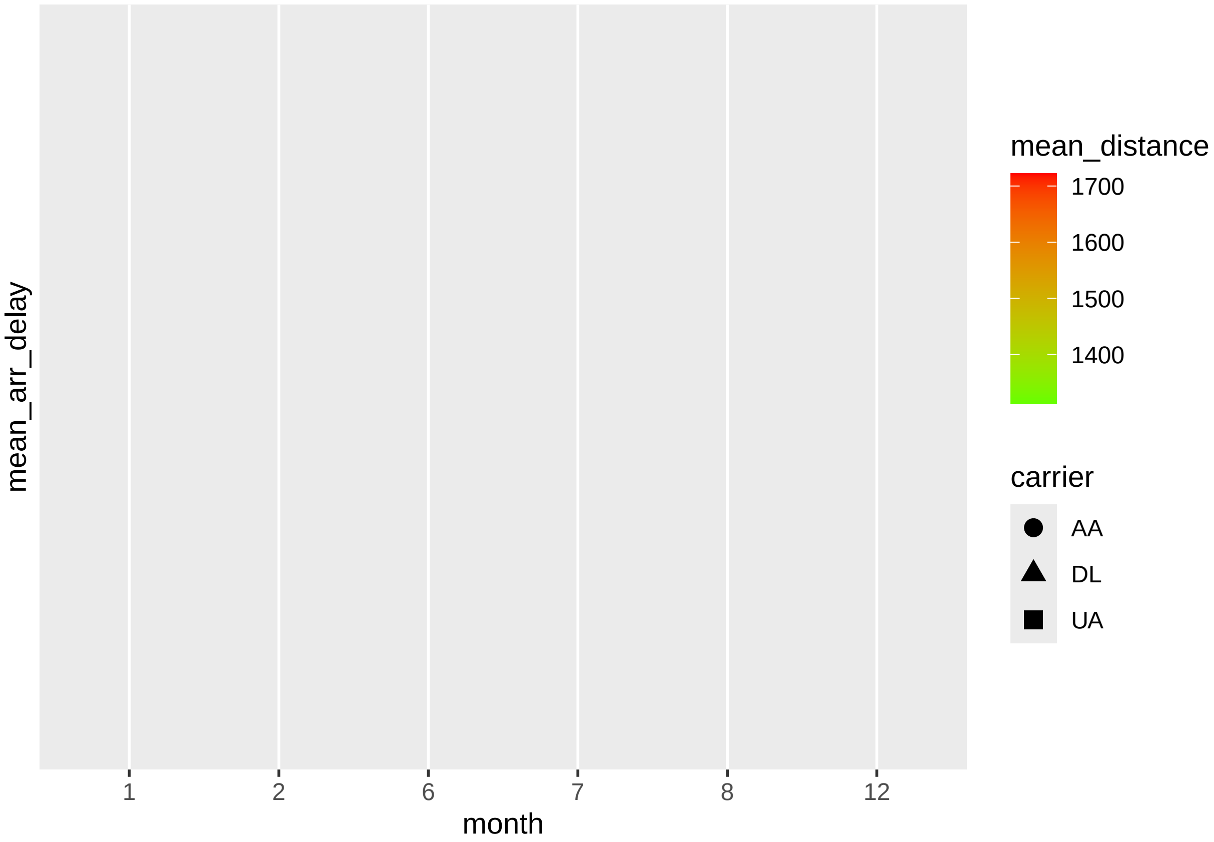
```
## # A tibble: 18 x 4
## # Groups:   month [6]
##    month carrier mean_arr_delay mean_distance
##    <fct> <chr>            <dbl>         <dbl>
##  1 1     AA                  NA         1404.
##  2 1     DL                  NA         1314.
##  3 1     UA                  NA         1598.
##  4 2     AA                  NA         1404.
##  5 2     DL                  NA         1312.
##  6 2     UA                  NA         1569.
##  7 6     AA                  NA         1382.
##  8 6     DL                  NA         1353.
##  9 6     UA                  NA         1693.
## 10 7     AA                  NA         1376.
## 11 7     DL                  NA         1357.
## 12 7     UA                  NA         1708.
## 13 8     AA                  NA         1378.
## 14 8     DL                  NA         1352.
## 15 8     UA                  NA         1722.
## 16 12    AA                  NA         1412.
## 17 12    DL                  NA         1324.
## 18 12    UA                  NA         1655.
```

```
arr_delay.summary_stats_plot = ggplot(data = summary_stats, aes(x = month,
    y = mean_arr_delay, shape = carrier, color = mean_distance)) +
    geom_point(size = 3) + geom_text(aes(label = carrier), hjust = -0.5,
    vjust = 1, size = 3) + scale_color_gradient(low = "#66FF00",
    high = "#FF0000")

arr_delay.summary_stats_plot
```

```
## Warning: Removed 18 rows containing missing values or values outside the scale range
## (`geom_point()`).
```

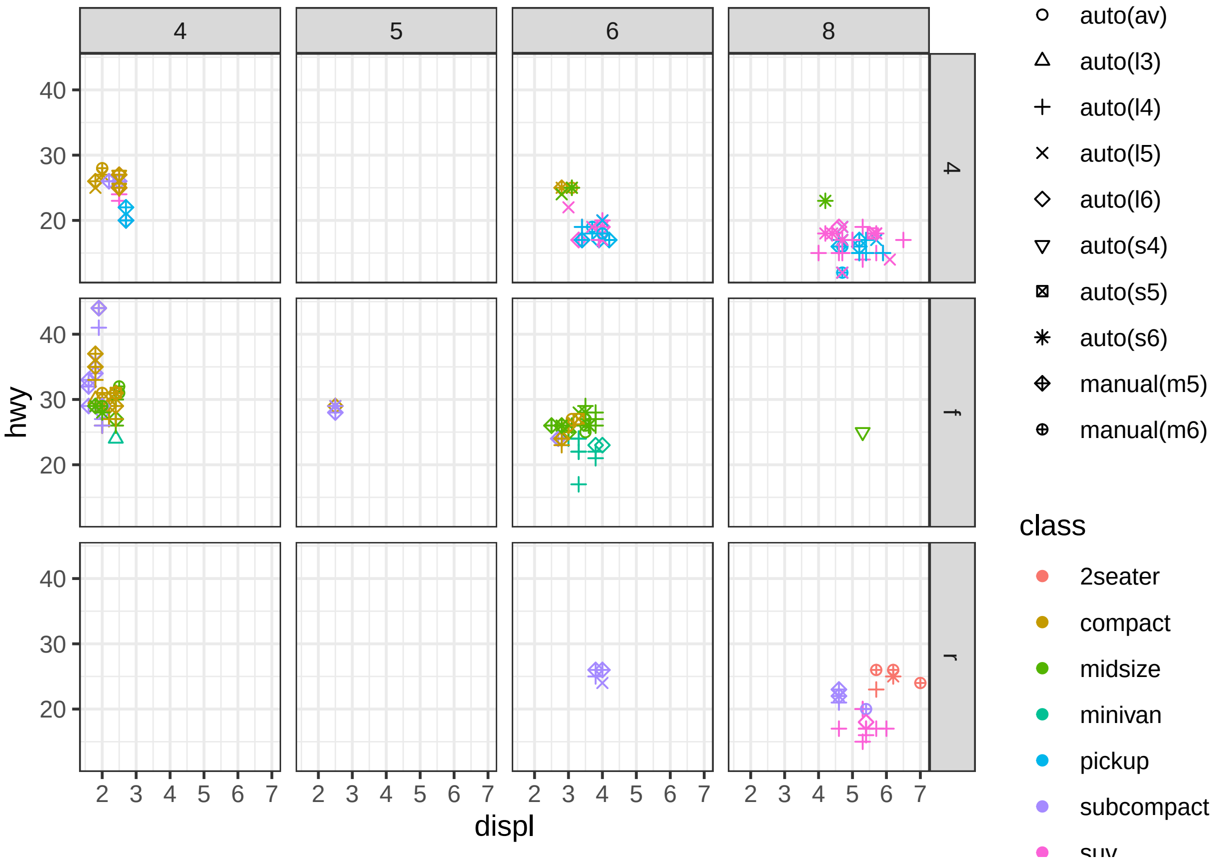
```
## Warning: Removed 18 rows containing missing values or values outside the scale range
## (`geom_text()`).
```



Problem 2

Please refer to the data set `mpg` that is available from the `ggplot2` package. Plot `displ` against `hwy` with faceting by `drv` and `cyl`, `color` disgnated by `class`, and `shape` by `trans`. This illustrates visualization with 4 factors.

```
mpg1 = na.omit(mpg)
mpg.fourfactorplot = ggplot(mpg1, aes(x = displ, y = hwy)) +
    theme_bw() + geom_point(aes(colour = class, shape = trans)) +
    scale_shape_manual(values = 1:length(unique(mpg1$trans))) +
    facet_grid(drv ~ cyl)
mpg.fourfactorplot
```



4
5
6
8
4
f
r
hwy
40
30
20
displ
2
3
4
5
6
7
auto(av)
auto(l3)
auto(l4)
auto(l5)
auto(l6)
auto(s4)
auto(s5)
auto(s6)
manual(m5)
manual(m6)
class
2seater
compact
midsize
minivan
pickup
subcompact
suv