Stat 437 HW2

Yu-Tung (Jenny), Cheng (ID:11678647)

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:

- Advanced Visualizations via ggplot2: adjusting legends, fonts, orientation, and math expressions
- Visualizing networks as graphs
- Interactive visualization

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 (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 4 different `month` 12, 1, 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 that you are required to use `dplyr` for this purpose.

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

```
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
```

```
library(nycflights13)
library(ggplot2)
library(tidyverse)
```

```
## -- Attaching core tidyverse packages ------------------- tidyverse 1.3.2.9000 --
## v forcats   1.0.0     v stringr   1.5.0
## v lubridate 1.8.0     v tibble    3.1.8
## v purrr     0.3.4     v tidyr     1.2.0
## v readr     2.1.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the ]8;;http://conflicted.r-lib.org/conflicted package]8;; to force all conflicts to become errors
```

```
head(flights)
```

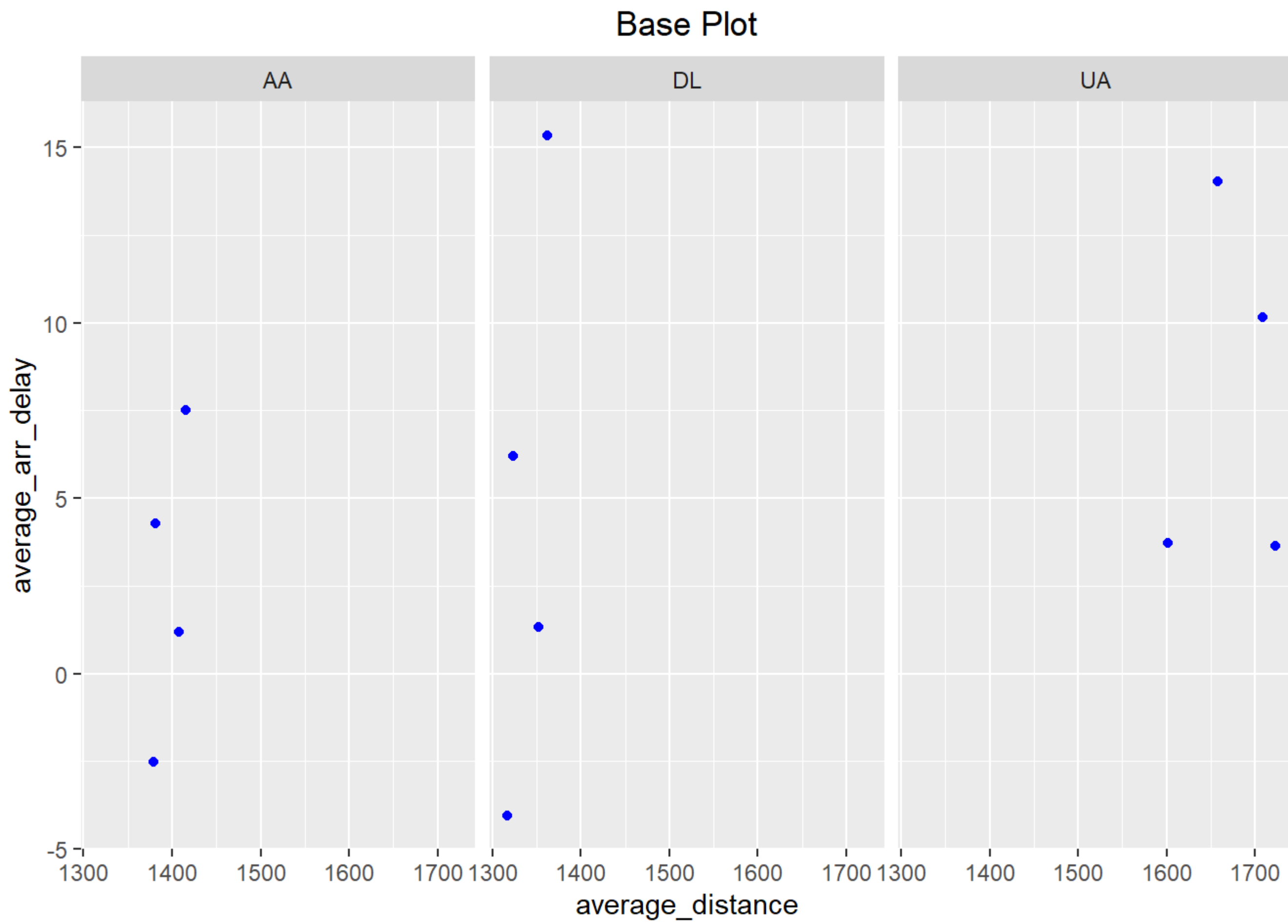
```
## # A tibble: 6 x 19
##    year month   day dep_time sched_dep~1 dep_d~2 arr_t~3 sched~4 arr_d~5 carrier
##   <int> <int> <int>    <int>       <int>   <dbl>   <int>   <int>   <dbl> <chr>
## 1  2013     1     1      517         515       2     830     819      11 UA
## 2  2013     1     1      533         529       4     850     830      20 UA
## 3  2013     1     1      542         540       2     923     850      33 AA
## 4  2013     1     1      544         545      -1    1004    1022     -18 B6
## 5  2013     1     1      554         600      -6     812     837     -25 DL
## 6  2013     1     1      554         558      -4     740     728      12 UA
## # ... with 9 more variables: flight <int>, tailnum <chr>, origin <chr>,
## #   dest <chr>, air_time <dbl>, distance <dbl>, hour <dbl>, minute <dbl>,
## #   time_hour <dttm>, and abbreviated variable names 1: sched_dep_time,
## #   2: dep_delay, 3: arr_time, 4: sched_arr_time, 5: arr_delay
```

```
##create a small data frame 'temp' from flights and select 6 months and 3 carriers from 'flights'

temp = flights %>% select(month, arr_delay,carrier, distance) %>% filter(month %in% c(12,1,7,8), carrier %in% c("UA", "AA",
"DL"), distance > 700)

#remove the missing value NA
temp = na.omit(temp)
myData = temp %>% group_by(carrier, month) %>% summarise_at(vars(arr_delay, distance), list(mean))

temp
```

```
## # A tibble: 42,562 x 4
##    month arr_delay carrier distance
##    <int>     <dbl> <chr>      <dbl>
##  1     1        11 UA          1400
##  2     1        20 UA          1416
##  3     1        33 AA          1089
##  4     1       -25 DL           762
##  5     1        12 UA           719
##  6     1         8 AA           733
##  7     1         7 UA          2475
##  8     1       -14 UA          2565
##  9     1        31 AA          1389
## 10     1        -8 UA          2227
## # ... with 42,552 more rows
```

(1.a) For each combination of the values of `carrier` and `month`, obtain the average `arr_delay` and obtain the average `distance`. Plot the average `arr_delay` against the average `distance`, use `carrier` as facet; add a title “Base plot” and center the title in the plot. This will be your base plot, say, as object `p`. Show the plot `p`.

```
p = ggplot(data = myData, aes(y= arr_delay, x = distance))+geom_point(col="blue")

p + facet_wrap(~carrier, nrow = 1)+labs(x= 'average_distance', y='average_arr_delay',title=('Base Plot'))+theme(plot.title=
element_text(hjust = 0.5), )
```



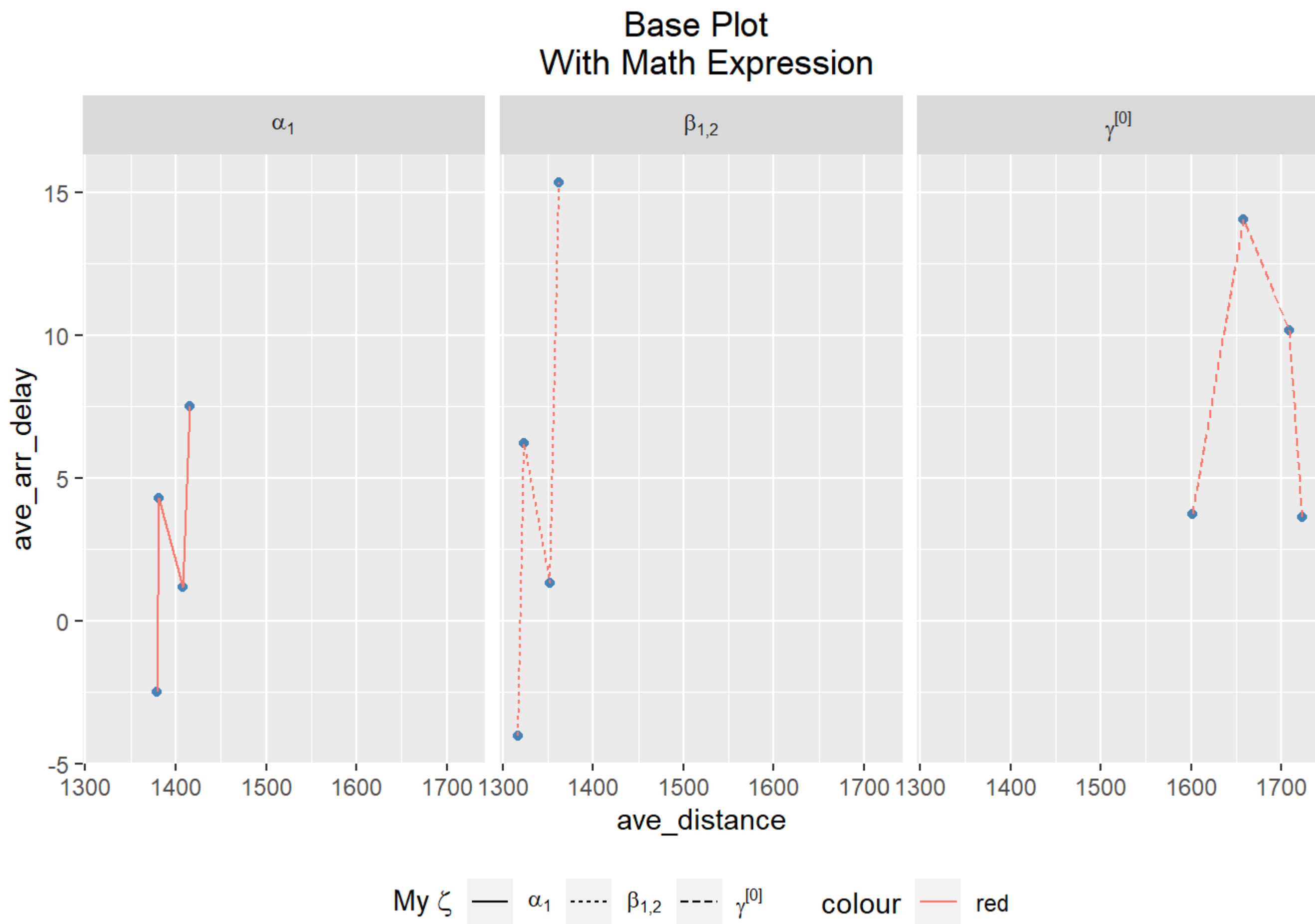

```
##Interpretation: In this Base Plot, it indicates that UA focuses on long-distance flights compared to AA and DL. DL relativ
ely shows the higher frequency of the average of arrival delay in the short-distance flights compared with AA.
```

(1.b) Modify `p` as follows to get a plot `p1` : connect the points for each `carrier` via one type of dashed line; code the 3 levels of `carrier` as α_1, $\beta_{1,2}$ and $\gamma^{[0]}$, and display them in the strip texts; change the legend title into "My ζ" (this legend is induced when you connect points for each `carrier` by a type of line), and put the legend in horizontal direction at the bottom of the plot; add a title "With math expressions" and center the title in the plot. Show the plot `p1` .

```
#map 3 levels to expressions
carrierStg = c(expression(alpha[1]), expression(beta['1,2']), expression(gamma^{'[0]'}))

#create variable DF (a factor) with levels “α1”, “β2” and “gamma”
myData$DF = factor(myData$carrier, labels = carrierStg)

#use 'slice' to check correctness of mapping

myData %>% select(distance,arr_delay,DF, carrier) %>% group_by(carrier) %>% slice(1)
```

```
## # A tibble: 3 x 4
## # Groups:   carrier [3]
##   distance arr_delay DF                          carrier
##      <dbl>     <dbl> <fct>                       <chr>
## 1    1408.      1.19 "alpha[1]"                  AA
## 2    1317.     -4.04 "beta[\"1,2\"]"             DL
## 3    1601.      3.72 "gamma^{\n    \"[0]\"\n}" UA
```

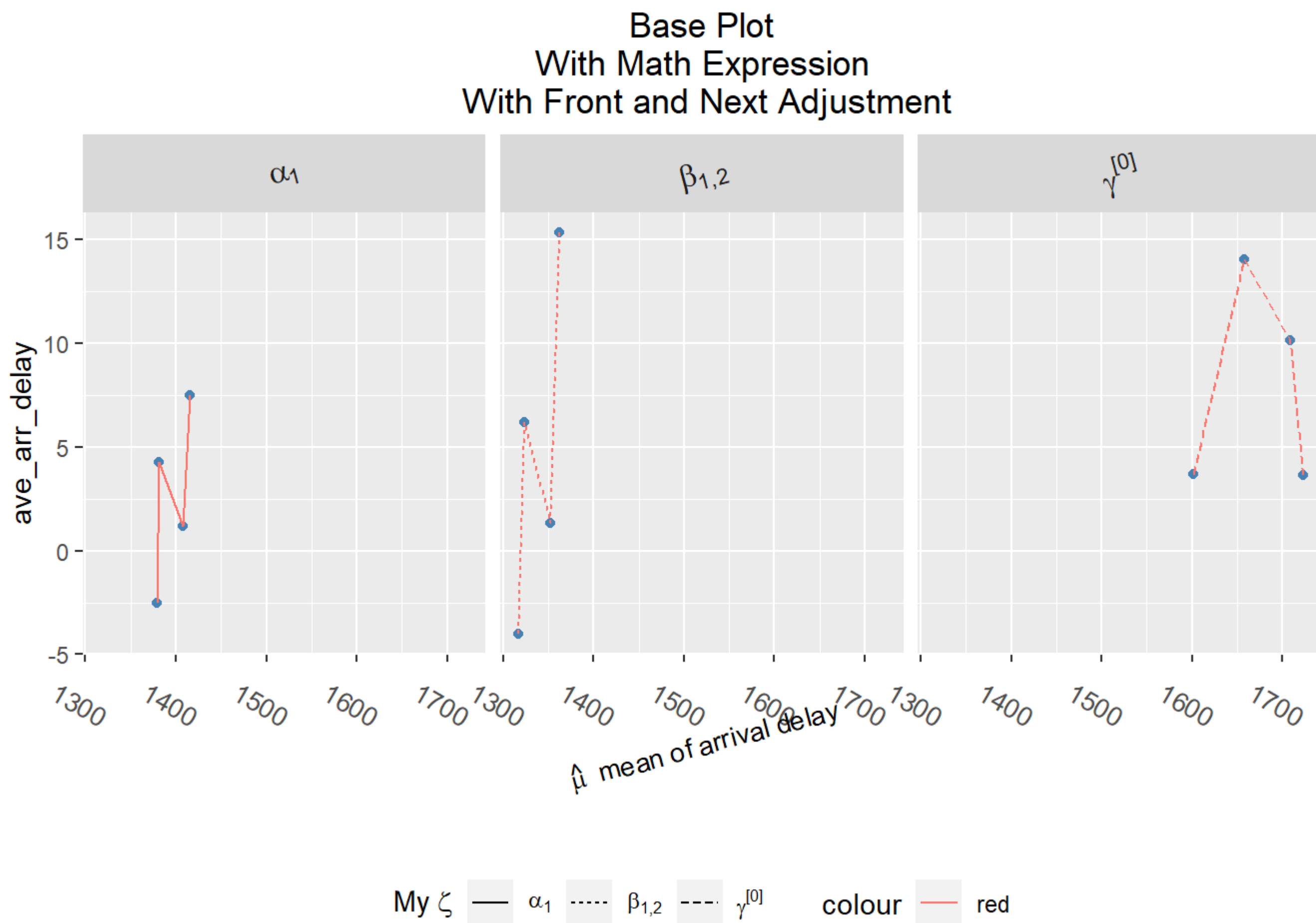
```
p1 = ggplot(myData, aes(distance, arr_delay))+geom_point(color = "steelblue")+ theme(plot.title = element_text(hjust=0.5))+g
eom_line(aes(linetype= myData$DF, color = "red"))+labs(linetype = expression(paste("My ", zeta, sep=""))) + scale_linetype_d
iscrete(labels =carrierStg)


p1 = p1+facet_wrap(~DF, nrow = 1, labeller = label_parsed) + labs(x='ave_distance', y='ave_arr_delay', title=('Base Plot \n
With Math Expression'))+theme(legend.position="bottom",legend.direction="horizontal")

print(p1)
```




```
##Interpretation: The graph shows the changes of the 3 levels of carriers to the Mathematical expressions which are AA as α1
with the solid line to connect the data points,  DL as β1,2 with the dash line to connect the data points, and UA as γ^[0] w
ith looser dash line to connect the data points. From the visualization, DL as β1,2 displays relatively huge gap between the
ir less frequent average of arrival delay and their most frequent data point. By this case, among the AA, DL and UA, DL(β1,
2) has comparatively highly unstable for the average flight punctuality rate.
```

(1.c) Modify `p1` as follows to get a plot `p2` : set the font size of strip text to be 12 and rotate the strip texts counterclockwise by 15 degrees; set the font size of the x-axis text to be 10 and rotate the x-axis text clockwise by 30 degrees; set the x-axis label as “$\hat{\mu}$ for mean arrival delay”; add a title “With front and text adjustments” and center the title in the plot. Show the plot `p2`

```
p2 = p1+ theme(axis.text.x = element_text(size= 10, angle = -30), axis.title.x= element_text(size=10, angle=15), strip.text= element_text(size=12, angle= 15))+labs(x = expression(paste(hat(mu), ' \t mean of arrival delay')), title= ('Base Plot\n With Math Expression \n With Front and Next Adjustment'))

print(p2)
```



```
##Interpretation: We set the font size of the strip text of α1, β1,2 and  γ^[0] with the counterclockwise rotate by 15 degrees. Besides that, by adding the mu^hat expression to express the mean of arrival delay with the font size of 10 and be rotated clockwise by 30 degrees.
```

Problem 2

This problem requires you to visualize the binary relationship between members of a karate club as an undirected graph. Please install the R library `igraphdata`, from which you can obtain the data set `karate` and work on it. Create a graph for `karate`. Once you obtain the graph, you will see that each vertex is annotated by a number or letter. What do the numbers or letters refer to? Do you see subgraphs of the graph? If so, what do these subgraphs mean?

```
#install package 'igraphdata' and 'igraph'
library(igraphdata)
library(igraph)
```

```
##
## Attaching package: 'igraph'
```

```
## The following objects are masked from 'package:lubridate':
##
##     %--%, union
```

```
## The following objects are masked from 'package:purrr':
##
##     compose, simplify
```

```
## The following object is masked from 'package:tidyr':
##
##     crossing
```

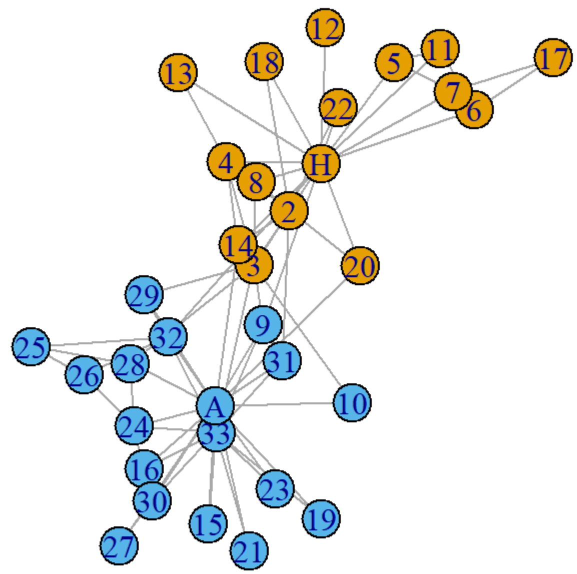
```
## The following object is masked from 'package:tibble':
##
##     as_data_frame
```

```
## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union
```

```
## The following objects are masked from 'package:stats':
##
##     decompose, spectrum
```

```
## The following object is masked from 'package:base':
##
##     union
```

```
data(karate)
plot(karate)
```




```
##Interpretation: This is a graphical representation of the social relationships among 34 individuals in a karate club. Each
sub-bubbles of numbers represent to the members in the karate club, and one of the two main bubbles 'A' refers to the karate
president John A. and another one with 'H' refers to the karate instructor.  The vertex attribute is regarded as the partici
pant's membership. However,  Actor 9 has more connections to 'H's group. Besides that, some sub-bubble individuals between
'H' and 'A' with the middle part usually show more complexity of overlapping relationships between them.
```

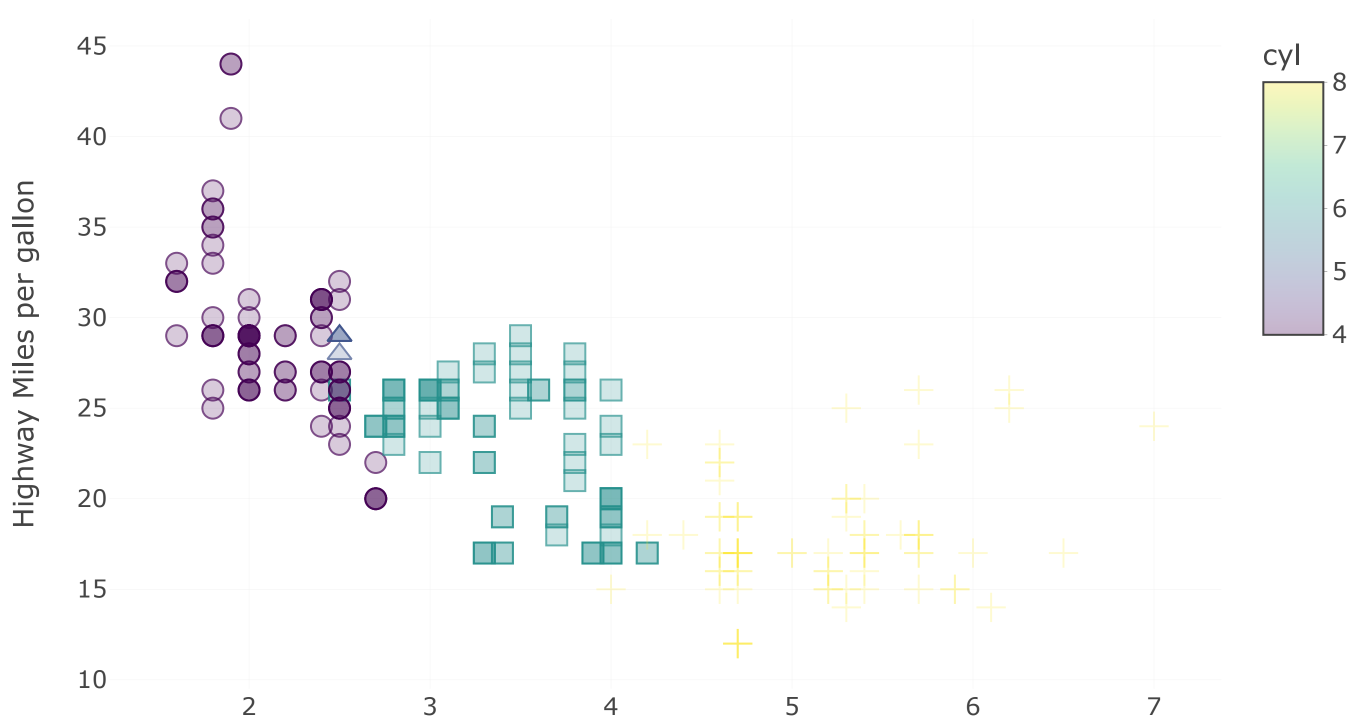
Problem 3

This problem requires to to create an interactive plot using `plotly` . If you want to display properly the plot in your HW answers, you may well need to set your HW document as an html file (instead of doc, docx or pdf file) when you compile your R codes.

Please use the `mpg` data set we have discussed in the lectures. Create an interactive, scatter plot between “highway miles per gallon” `hwy` (on the y-axis) and “engine displacement in litres” `displ` (on the x-axis) with the `color` aesthetic designated by “number of cylinders” `cyl` , and set the x-axis label as “engine displacement in litres” and y-axis label as “highway miles per gallon”. You need to check the object type for `cyl` and set it correctly when creating the plot. Add the title “# of cylinders” to the legend and adjust the vertical position of the legend, if you can. For the last, you may look through https://plotly.com/r/legend/ (https://plotly.com/r/legend/) for help.

```
library(plotly)
library(ggplot2)

plot_ly(data= mpg,x = ~displ, y= ~hwy, color = ~cyl, type = "scatter", symbol = ~cyl, width = 700, height = 400, size = 3, alpha = 0.3, text = ~cyl) %>%
      layout(xaxis = list(title="Engine Displacement in liters"), yaxis = list(title='Highway Miles per gallon'),
             legend=list(title=list(text='# of cylinders')))
```

```
## No scatter mode specifed:
##   Setting the mode to markers
##   Read more about this attribute -> https://plotly.com/r/reference/#scatter-mode
```



Engine Displacement in liters

```
##Interpretation: As the visualization shows that set of 4 cylinders has the least engine displacement in liters but it has
the largest amount consumption in highway miles per gallon compared to others with 5, 6 and 8 cylinders. However, the cars w
ith 8 cylinders, which are the indicated as round shape of data points, have the pretty high frequency of engine displacemen
t in liters with unstable ranges from different data. The 5 cylinder only has two data points from the graph in the triangul
ar shape of data points with technically fewer sample observations among others.
```