GoogleCapstoneTT_Part4_DatVis

TT

01/02/2022

R Markdown

This is **fourth** part of my capstone project. Ref to previous parts.In the first part data have been collected and browsed. Therafter data was manipulated and processed. As as result data from various csv files has been made compatible. Thereafter data was merged into one large dataframe and exported to a big CSV file for further analysis. In the second part data was analysed in terms of compatibility, some column types were changed, data was cleaned and exported to new clean CSV file.

Loading required packages

```r
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.1     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```r
library(lubridate)
```

```
## 
## Attaching package: 'lubridate'
## The following objects are masked from 'package:base':
## 
##     date, intersect, setdiff, union
```

```
library(ggplot2)
library(dplyr)
```

Loading previously created dataframe:

```
bikedata <- read.csv("E:/Tomasz/CapstoneGoogle/capstone_bikedata_for_analysis.csv")
```

```
str(bikedata)
```

```
## 'data.frame':    3817572 obs. of  17 variables:
##  $ X                 : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ ride_id           : num  21742443 21742444 21742445 21742446 21742447 ...
##  $ started_at        : chr  "2019-01-01 00:04:37" "2019-01-01 00:08:13" "2019-01-01 00:13:
23" "2019-01-01 00:13:45" ...
##  $ ended_at          : chr  "2019-01-01 00:11:07" "2019-01-01 00:15:34" "2019-01-01 00:27:
12" "2019-01-01 00:43:28" ...
##  $ rideable_type     : int  2167 4386 1524 252 1170 2437 2708 2796 6205 3939 ...
##  $ tripduration      : int  390 441 829 1783 364 216 177 100 1727 336 ...
##  $ start_station_id  : int  199 44 15 123 173 98 98 211 150 268 ...
##  $ start_station_name: chr  "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave
& 18th St" "California Ave & Milwaukee Ave" ...
##  $ end_station_id    : int  84 624 644 176 35 49 49 142 148 141 ...
##  $ end_station_name  : chr  "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)"
"Western Ave & Fillmore St (*)" "Clark St & Elm St" ...
##  $ member_casual     : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ ride_length       : num  6.5 7.35 13.82 29.72 6.07 ...
##  $ date              : chr  "2019-01-01" "2019-01-01" "2019-01-01" "2019-01-01" ...
##  $ month             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ day               : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : int  2019 2019 2019 2019 2019 2019 2019 2019 2019 2019 ...
##  $ day_of_week       : chr  "Tuesday" "Tuesday" "Tuesday" "Tuesday" ...
```

After export CSV file some unwanted colums were added and type of columns “started_at” and “ended at” where changed from date to character. This needs to addresed with dropping columns and type change.

Change type for colums and droping column X:

```
bikedata <- bikedata %>%
  select(-c("X"))

str(bikedata)
```

```
## 'data.frame':    3817572 obs. of  16 variables:
##  $ ride_id           : num  21742443 21742444 21742445 21742446 21742447 ...
##  $ started_at        : chr  "2019-01-01 00:04:37" "2019-01-01 00:08:13" "2019-01-01 00:13:
23" "2019-01-01 00:13:45" ...
##  $ ended_at          : chr  "2019-01-01 00:11:07" "2019-01-01 00:15:34" "2019-01-01 00:27:
12" "2019-01-01 00:43:28" ...
##  $ rideable_type     : int  2167 4386 1524 252 1170 2437 2708 2796 6205 3939 ...
##  $ tripduration      : int  390 441 829 1783 364 216 177 100 1727 336 ...
##  $ start_station_id  : int  199 44 15 123 173 98 98 211 150 268 ...
##  $ start_station_name: chr  "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave
& 18th St" "California Ave & Milwaukee Ave" ...
##  $ end_station_id    : int  84 624 644 176 35 49 49 142 148 141 ...
##  $ end_station_name  : chr  "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)"
"Western Ave & Fillmore St (*)" "Clark St & Elm St" ...
##  $ member_casual     : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ ride_length       : num  6.5 7.35 13.82 29.72 6.07 ...
##  $ date              : chr  "2019-01-01" "2019-01-01" "2019-01-01" "2019-01-01" ...
##  $ month             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ day               : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : int  2019 2019 2019 2019 2019 2019 2019 2019 2019 2019 ...
##  $ day_of_week       : chr  "Tuesday" "Tuesday" "Tuesday" "Tuesday" ...
```

```
bikedata$started_at <- as_datetime(bikedata$started_at)
bikedata$ended_at <- as_datetime(bikedata$ended_at)

str(bikedata)
```

```
## 'data.frame':    3817572 obs. of  16 variables:
##  $ ride_id           : num  21742443 21742444 21742445 21742446 21742447 ...
##  $ started_at        : POSIXct, format: "2019-01-01 00:04:37" "2019-01-01 00:08:13" ...
##  $ ended_at          : POSIXct, format: "2019-01-01 00:11:07" "2019-01-01 00:15:34" ...
##  $ rideable_type     : int  2167 4386 1524 252 1170 2437 2708 2796 6205 3939 ...
##  $ tripduration      : int  390 441 829 1783 364 216 177 100 1727 336 ...
##  $ start_station_id  : int  199 44 15 123 173 98 98 211 150 268 ...
##  $ start_station_name: chr  "Wabash Ave & Grand Ave" "State St & Randolph St" "Racine Ave
& 18th St" "California Ave & Milwaukee Ave" ...
##  $ end_station_id    : int  84 624 644 176 35 49 49 142 148 141 ...
##  $ end_station_name  : chr  "Milwaukee Ave & Grand Ave" "Dearborn St & Van Buren St (*)"
"Western Ave & Fillmore St (*)" "Clark St & Elm St" ...
##  $ member_casual     : chr  "Subscriber" "Subscriber" "Subscriber" "Subscriber" ...
##  $ ride_length       : num  6.5 7.35 13.82 29.72 6.07 ...
##  $ date              : chr  "2019-01-01" "2019-01-01" "2019-01-01" "2019-01-01" ...
##  $ month             : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ day               : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ year              : int  2019 2019 2019 2019 2019 2019 2019 2019 2019 2019 ...
##  $ day_of_week       : chr  "Tuesday" "Tuesday" "Tuesday" "Tuesday" ...
```

Grouping data

```
dayweek_Plot <- bikedata %>%
  mutate(weekday = wday(started_at, label = TRUE)) %>%
  group_by(member_casual, weekday) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, weekday)
```

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

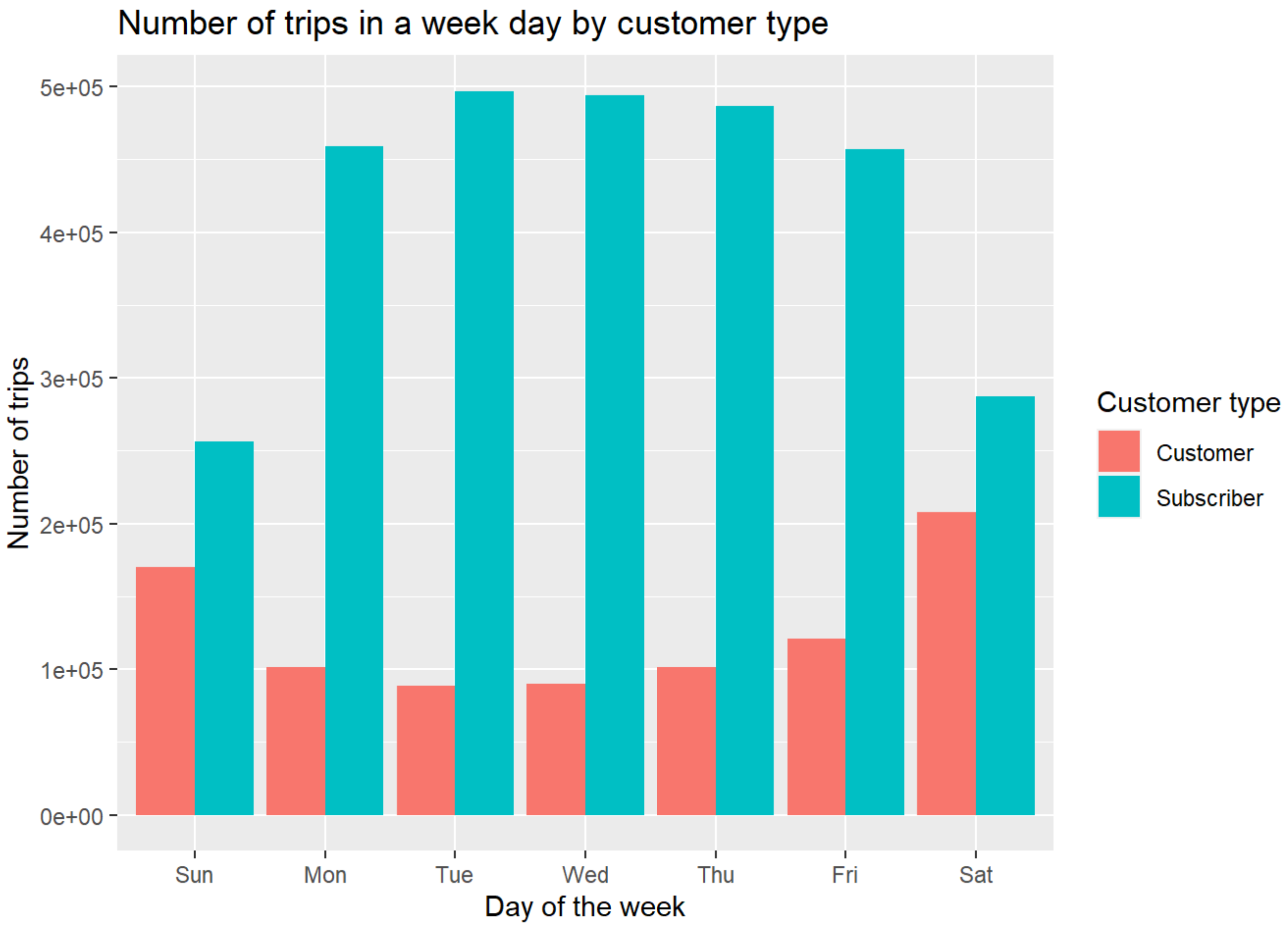
```
month_Plot <- bikedata %>%
  group_by(member_casual, month) %>%
  summarise(number_of_rides = n()
            ,average_duration = mean(ride_length)) %>%
  arrange(member_casual, month)
```

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

Below few basic visualisations will be created. At later stage of the capstone project they will be compared with similair visualisations in Tableau.

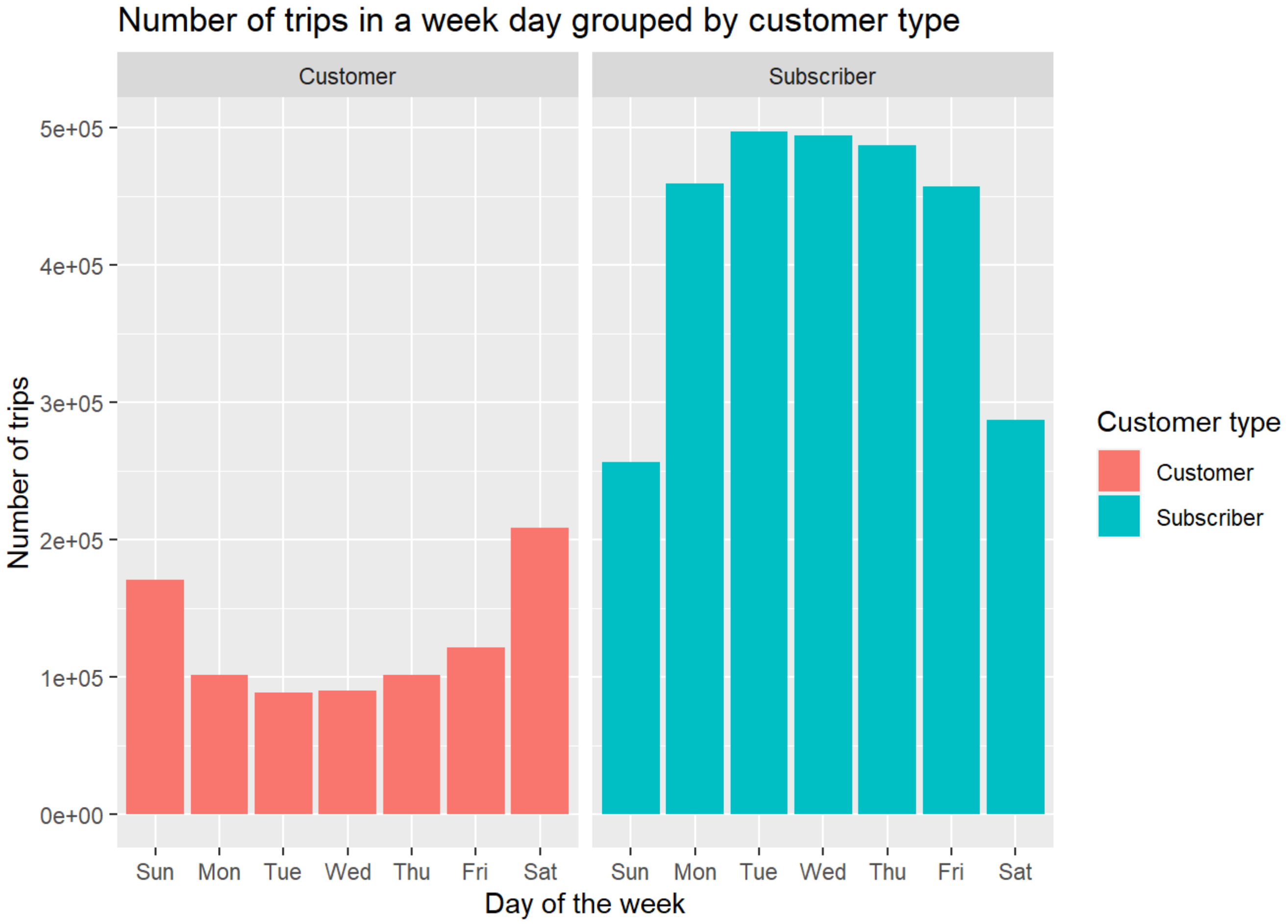
Plot 1 . Number of trips by week day by customer type

```
dayweek_Plot %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  labs(title = "Number of trips in a week day by customer type", x="Day of the week", y= "Num
ber of trips", fill= "Customer type") +
  geom_col(position = "dodge")
```



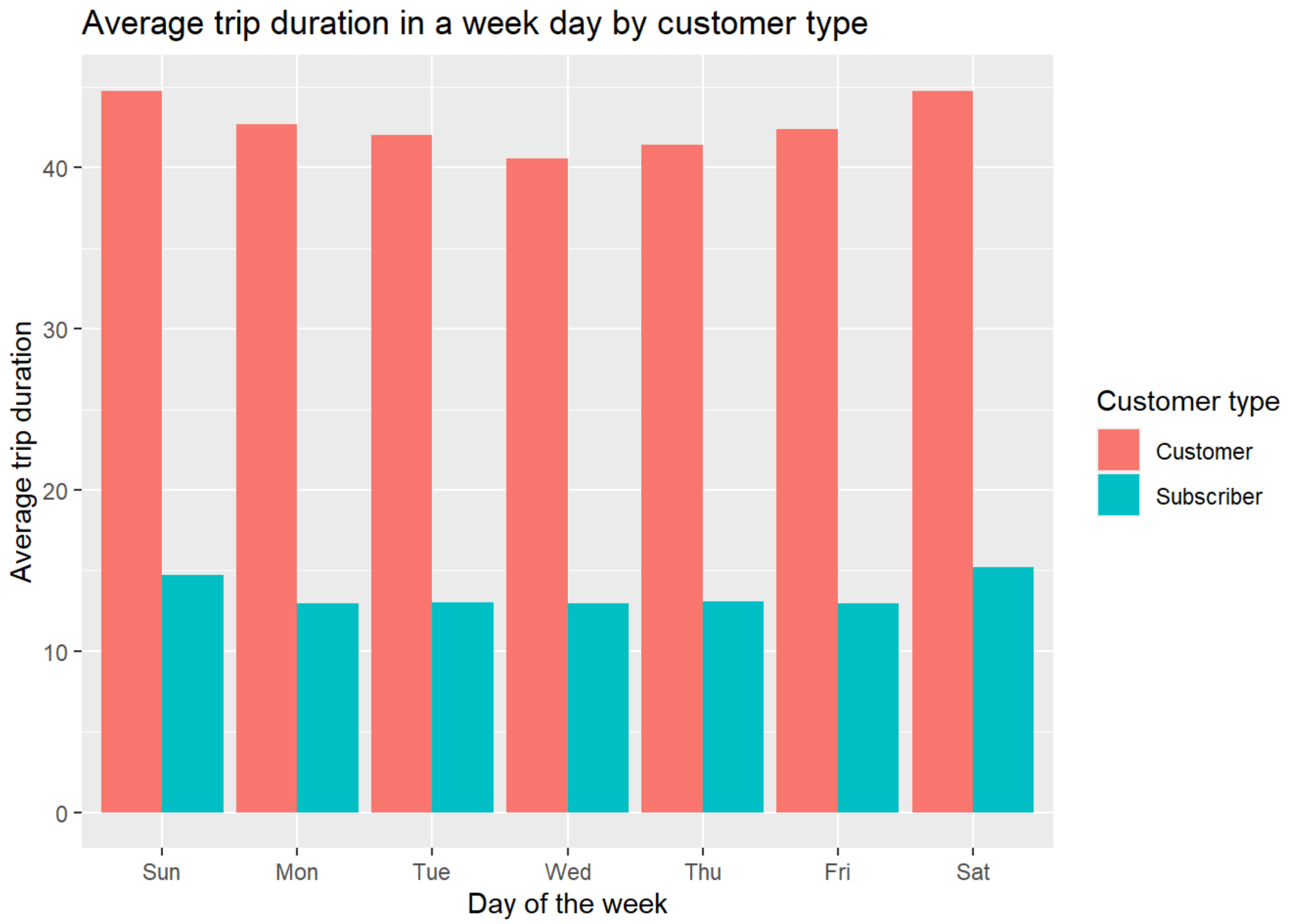
Plot 2 . Number of trips by week day by customer type with facet wrap (grouping by membership type)

```
dayweek_Plot %>%
  ggplot(aes(x = weekday, y = number_of_rides, fill = member_casual)) +
  labs(title = "Number of trips in a week day grouped by customer type", x="Day of the week",
y= "Number of trips", fill= "Customer type") +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual)
```



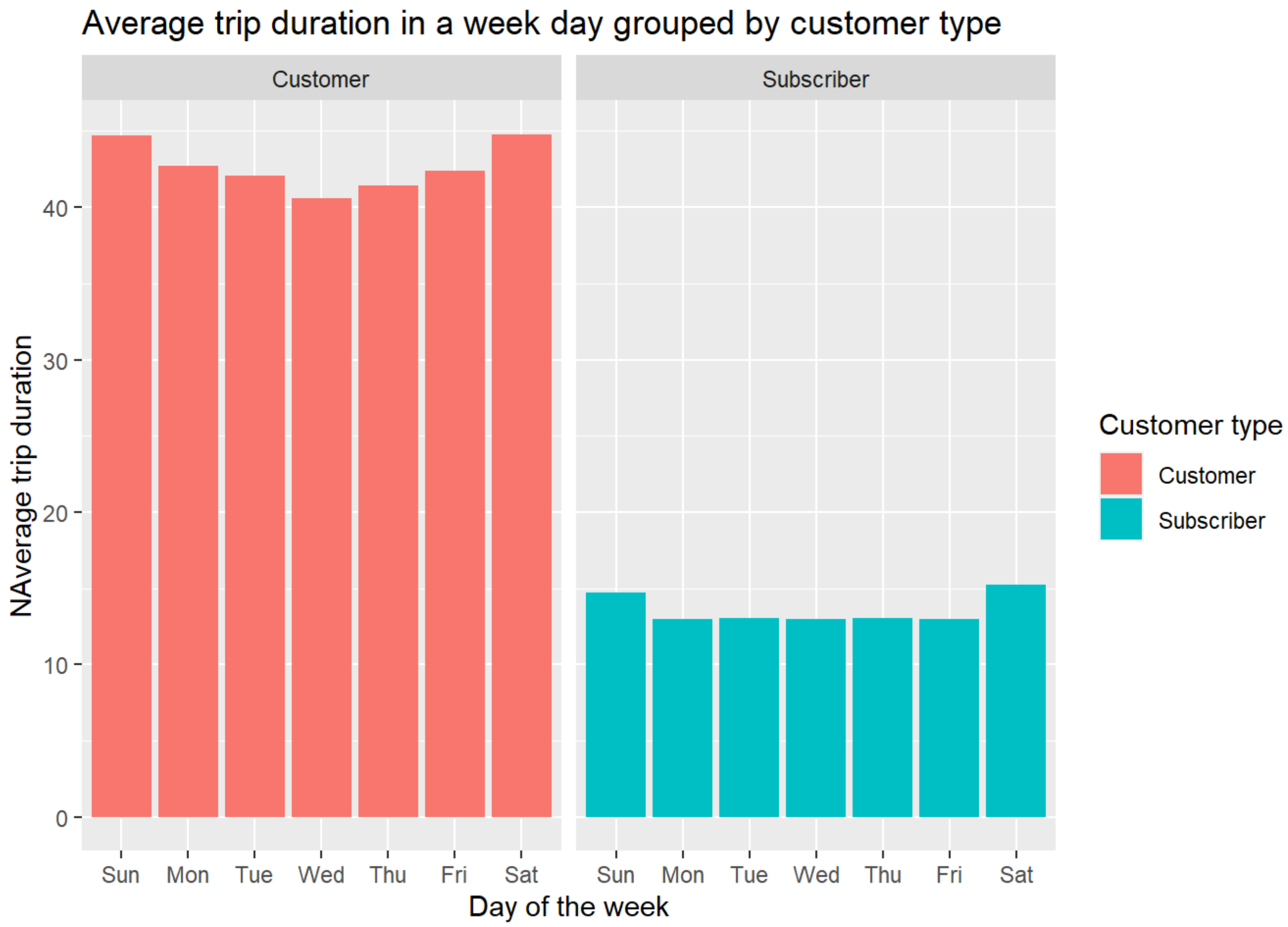
Plot 3 . Average trip duration by week day by customer

```
dayweek_Plot %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  labs(title = "Average trip duration in a week day by customer type", x="Day of the week", y
= "Average trip duration", fill= "Customer type") +
  geom_col(position = "dodge")
```



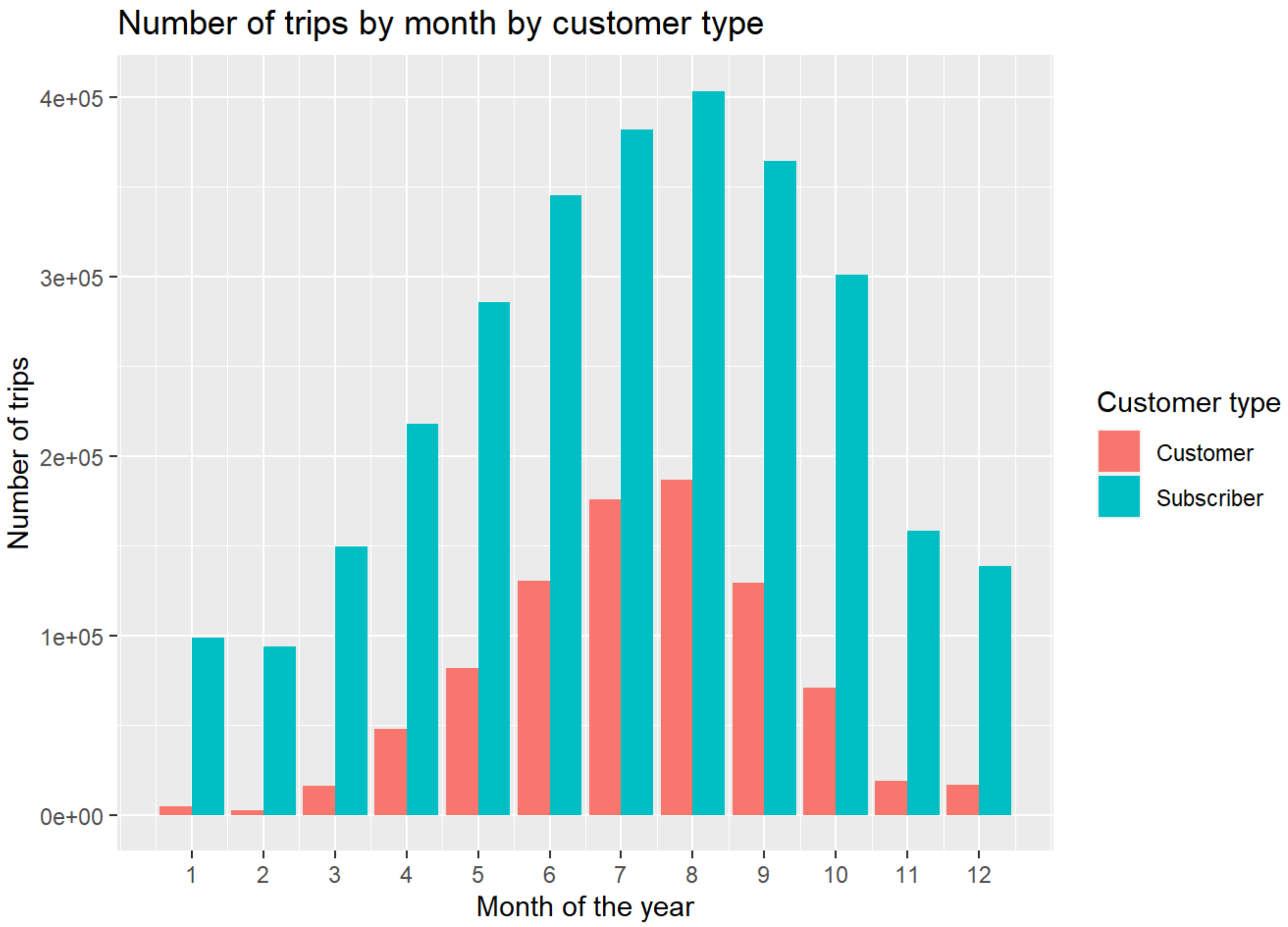
Plot 4 . Average trip duration by week day by customer type with facet wrap (grouping by membership type)

```
dayweek_Plot %>%
  ggplot(aes(x = weekday, y = average_duration, fill = member_casual)) +
  labs(title = "Average trip duration in a week day grouped by customer type", x="Day of the
 week", y= "NAverage trip duration", fill= "Customer type") +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual)
```



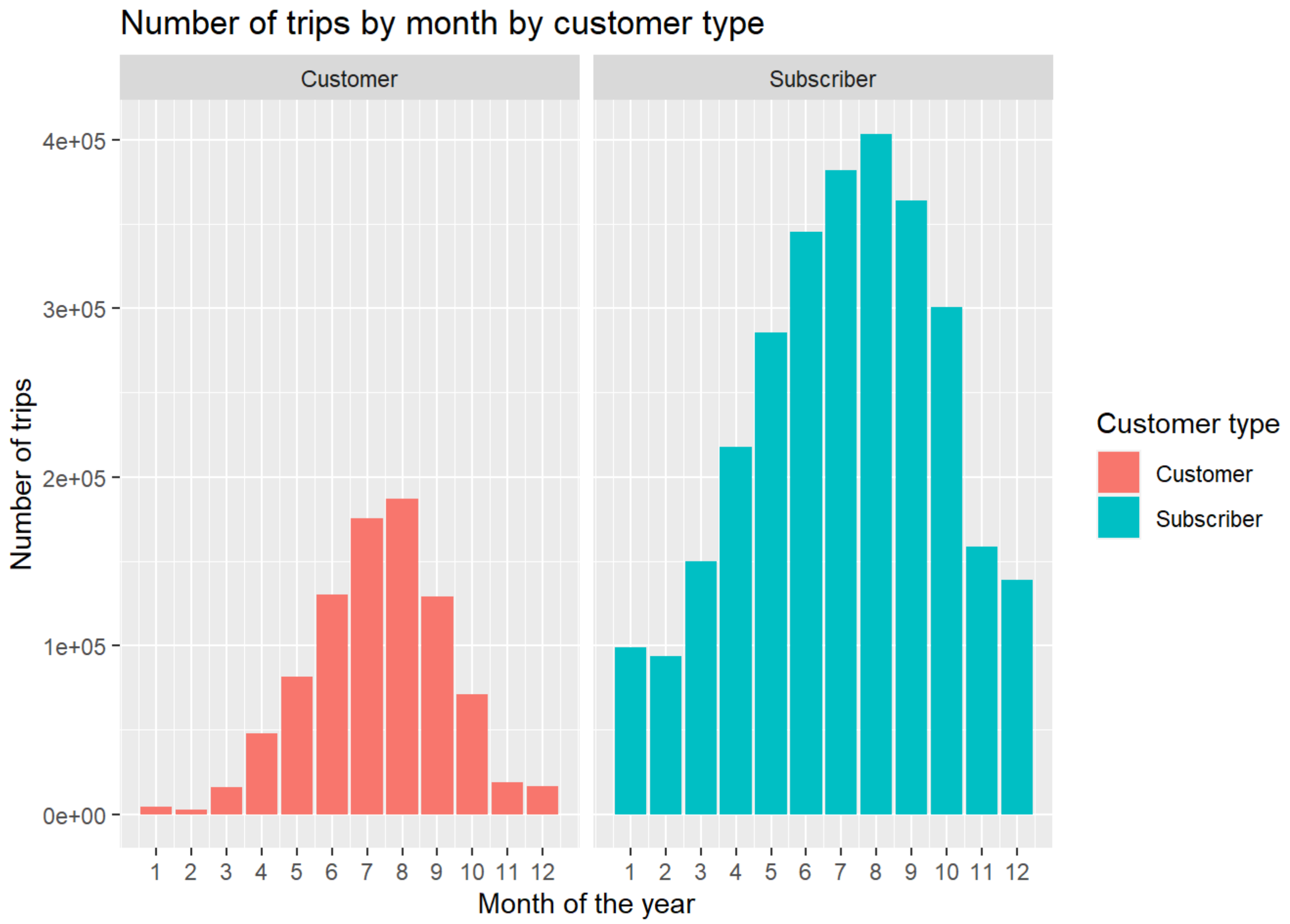
Plot 5. Number of trips by month by customer type

```
month_Plot %>%
  ggplot(aes(x = month, y = number_of_rides, fill = member_casual)) +
  labs(title = "Number of trips by month by customer type", x="Month", y= "Number of trips",
 fill= "Customer type")+
  geom_col(position = "dodge") +
  scale_x_continuous(name="Month of the year",  breaks = c(1, 2, 3, 4,5,6,7,8,9,10,11,12))
```



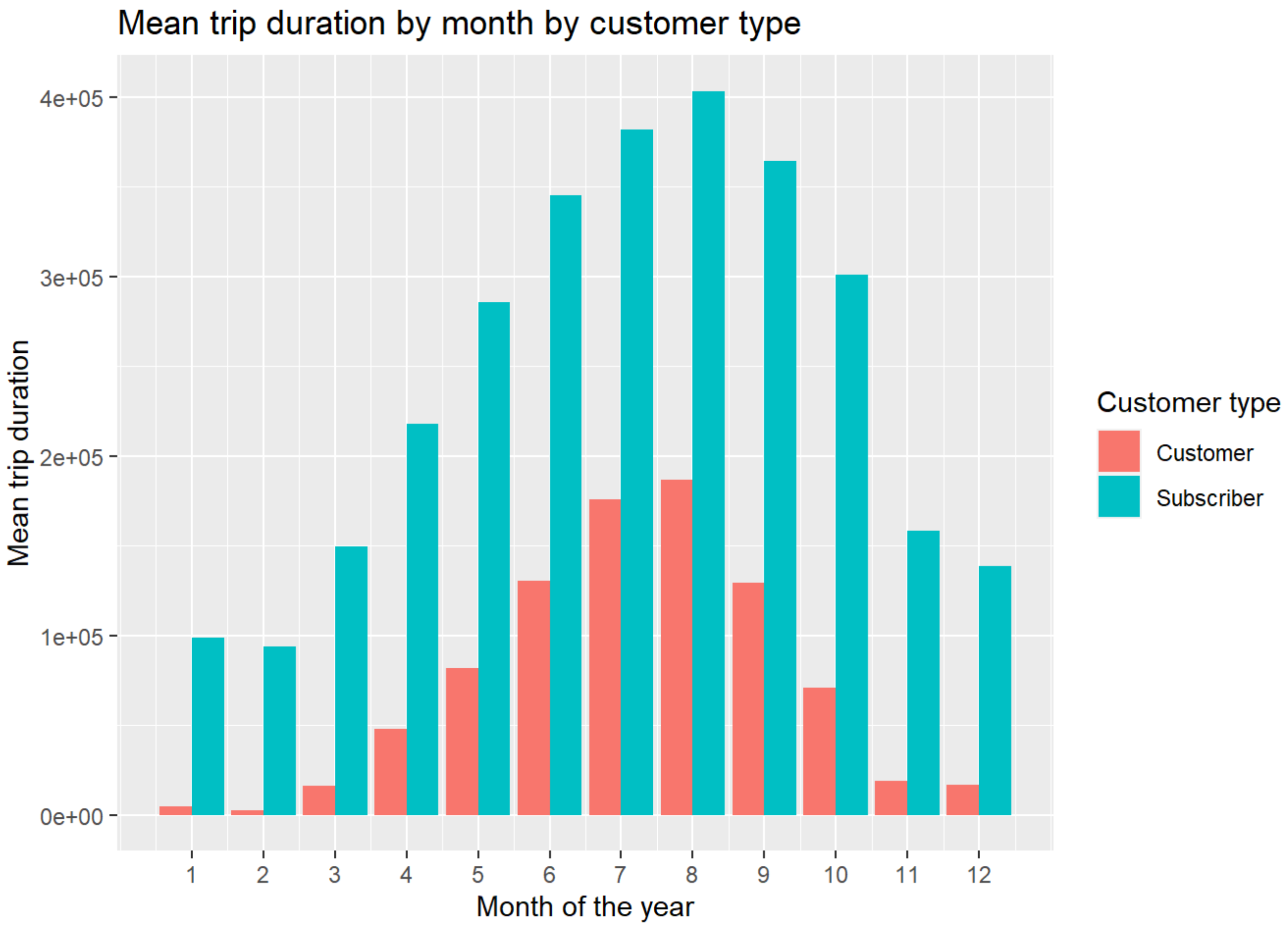
Plot 6.Number of trips by month by customer type with facet wrap (grouping by membership type)

```
month_Plot %>%
  ggplot(aes(x = month, y = number_of_rides, fill = member_casual)) +
  labs(title = "Number of trips by month by customer type", x="Day of the week", y= "Number o
f trips",  fill= "Customer type") +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual) +
  scale_x_continuous(name="Month of the year",  breaks = c(1, 2, 3, 4,5,6,7,8,9,10,11,12))
```



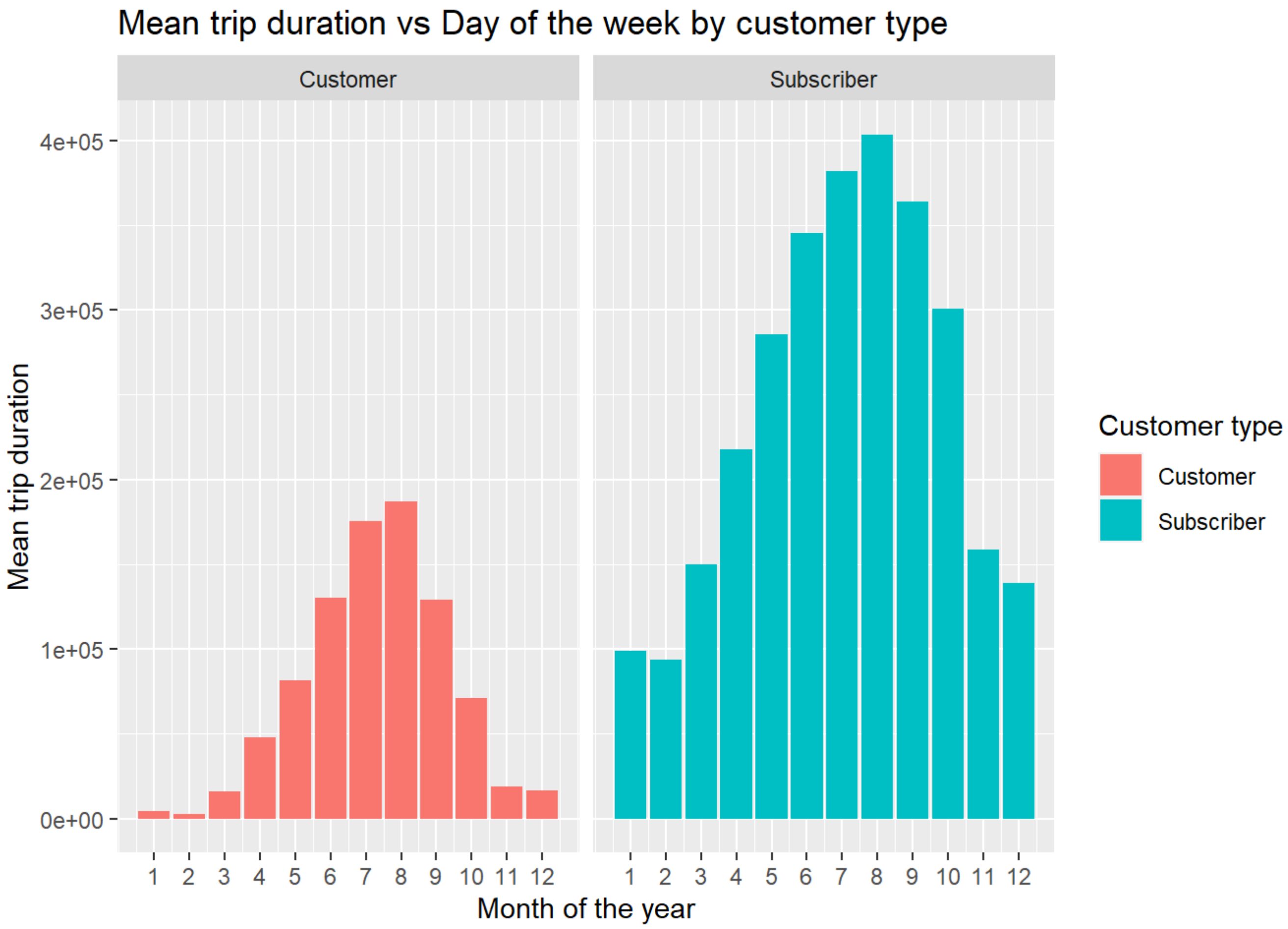
Plot 7. Mean trip duration by month by customer type

```
month_Plot %>%
  ggplot(aes(x = month, y = number_of_rides, fill = member_casual)) +
  labs(title = "Mean trip duration by month by customer type", x="Month", y= "Mean trip durat
ion", fill= "Customer type") +
  geom_col(position = "dodge") +
  scale_x_continuous(name="Month of the year",  breaks = c(1, 2, 3, 4,5,6,7,8,9,10,11,12))
```



Plot 8.Mean trip duration by month by customer type with facet wrap (grouping by membership type)

```
month_Plot %>%
  ggplot(aes(x = month, y = number_of_rides, fill = member_casual)) +
  labs(title = "Mean trip duration vs Day of the week by customer type", x="Day of the week",
y= "Mean trip duration",  fill= "Customer type") +
  geom_col(position = "dodge") +
  facet_wrap(~member_casual) +
  scale_x_continuous(name="Month of the year",  breaks = c(1, 2, 3, 4,5,6,7,8,9,10,11,12))
```


Mean trip duration vs Day of the week by customer type
Customer
Subscriber
Mean trip duration
4e+05
3e+05
2e+05
1e+05
0e+00
1
2
3
4
5
6
7
8
9
10
11
12
Month of the year
Customer type
Customer
Subscriber