ANZ Virtual Internship Report

This file outlines a sample response to task 1 & 2 using Rstudio.

Load the libraries used for the tasks

```
library(stringr)
library(lubridate)
library(tidyverse)
library(modelr)
library(sp)
library(leaflet)
library(geosphere)
library(knitr)
library(rpart)
```

Task 1 Exploratory data analysis

1.1 Load the transaction dataset

```
df = read.csv("data/DSynth_Output_100c_3m_v3.csv")
```

1.2 Data preparation

The dataset contains 12043 transactions for 100 customers who have one bank account each. Trasactional period is from 01/08/2018 - 31/10/2018 (92 days duration). The data entries are unique and have consistent formats for analysis. For each record/row, information is complete for majority of columns. Some columns contain missing data (blank or NA cells), which is likely due to the nature of transaction. (i.e. merchants are not involved for InterBank transfers or Salary payments) It is also noticed that there is only 91 unique dates in the dataset, suggesting the transaction records for one day are missing (turned out to be 2018-08-16).

The range of each feature should also be examined which shows that there is one customer that resides outside Australia.

```
# examine the summary of the dataset
summary(df)
str(df)

# change the format of date column
df$date<- as.Date(df$date,format = "%d/%m/%Y")

# the dateset only contain records for 91 days, one day is missing
DateRange <- seq(min(df$date), max(df$date), by = 1)
DateRange[!DateRange %in% df$date]  # 2018-08-16 transactions are missing

# derive weekday and hour data of each transaction
df$extraction = as.character(df$extraction)
df$hour = hour(as.POSIXct(substr(df$extraction,12,19),format="%H:%M:%S"))
df$weekday = weekdays(df$date)

# confirm the one -to -one link of account_id and customer_id
df %>% select(account,customer_id) %>%
  unique() %>%
  nrow()

# split customer & merchant lat_long into individual columns for analysis
dfloc = df[,c("long_lat","merchant_long_lat")]
dfloc<- dfloc %>% separate("long_lat", c("c_long", "c_lat"),sep=' ')
dfloc<- dfloc %>% separate("merchant_long_lat", c("m_long", "m_lat"),sep=' ')
dfloc<- data.frame(sapply(dfloc, as.numeric))
df <- cbind(df,dfloc)

# check the range of customer location
# filtering out transactions for those who don't reside in Australia
# Reference: http://www.ga.gov.au/scientific-topics/national-location-information/dimensions/con
tinental-extremities

df_temp <- df %>%
  filter (!(c_long >113 & c_long <154 & c_lat > (-44) & c_lat < (-10)))
length(unique(df_temp$customer_id))
```

Location infomation suggested there is one customer who resides outside Australia. However, all his/her transaction histories occured within AU thus these records are included for further analysis.

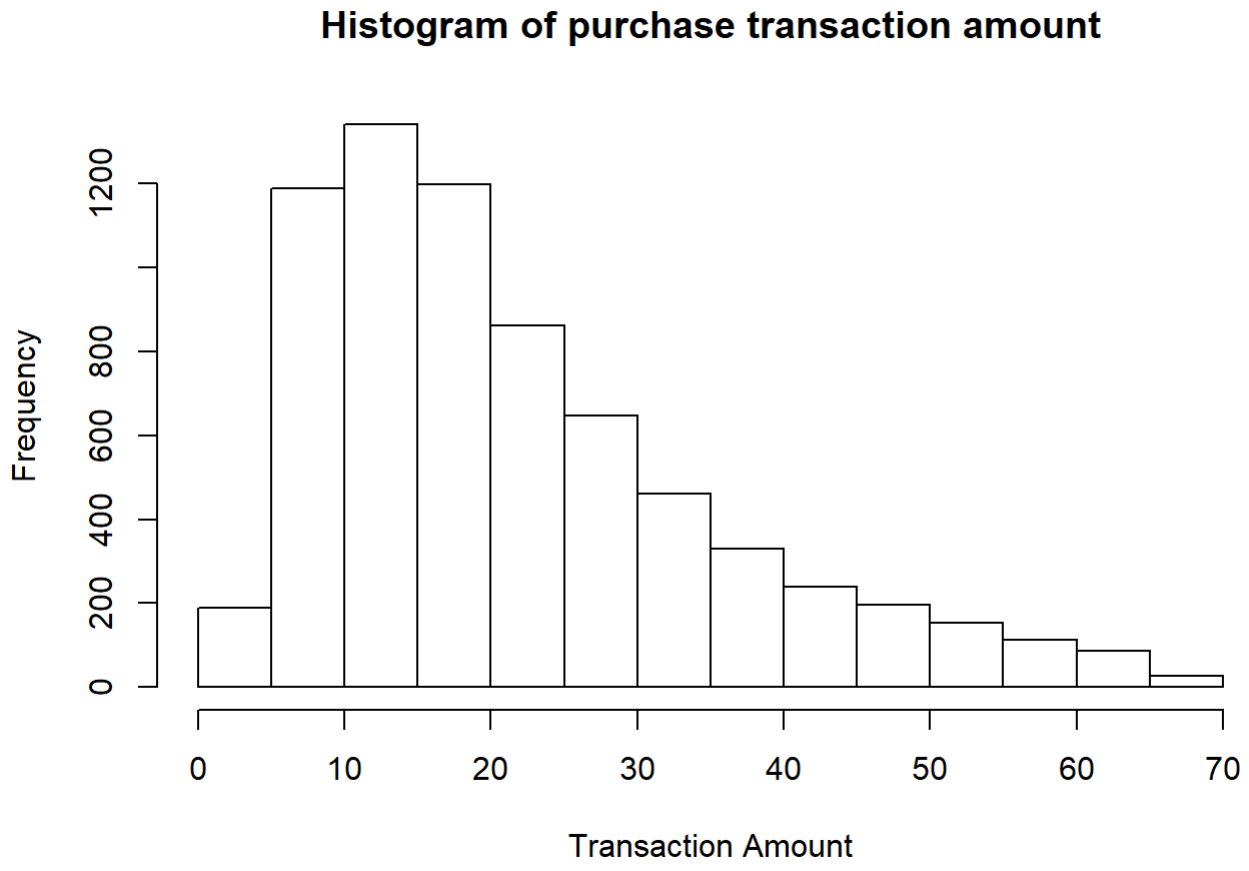
```
# check the distribution of missing values
apply(df, 2, function(x) sum(is.na(x)| x == ''))
# check the number of unique values for each column
apply(df, 2, function(x) length(unique(x)))
```

1.3 Gather some interesting overall insights about the data.

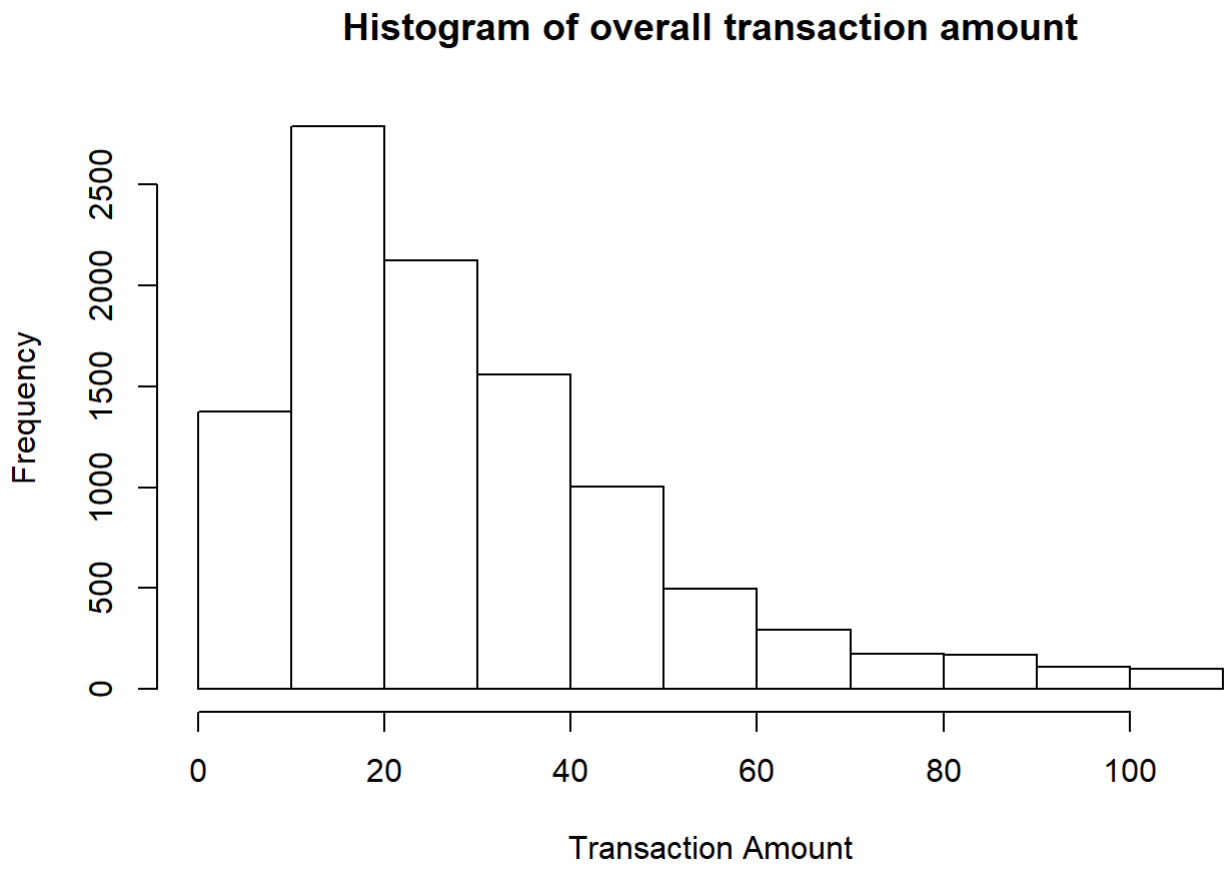
Given transactions include purchanse and other types - salary, payment etc, it is probably worthwhile to filter these out from overall dataset and have a closer look.

```
# filtering out purchase transactions only
# assuming purchase transactions must be associated with a merchant (have a merchant Id)
df_temp <- df %>% filter(merchant_id != '' )
# it turned out that is equivilent to excluding following categories of transactions
df_csmp <- df %>%filter(!(txn_description %in% c('PAY/SALARY',"INTER BANK", "PHONE BANK","PAYMEN
T")))

summary(df_csmp)
```

```
# visualise the distribution of transaction amount
hist(df_csmp$amount[!df_csmp$amount %in% boxplot.stats(df_csmp$amount)$out],   #exclude outliers
     xlab= 'Transaction Amount', main = 'Histogram of purchase transaction amount')
```

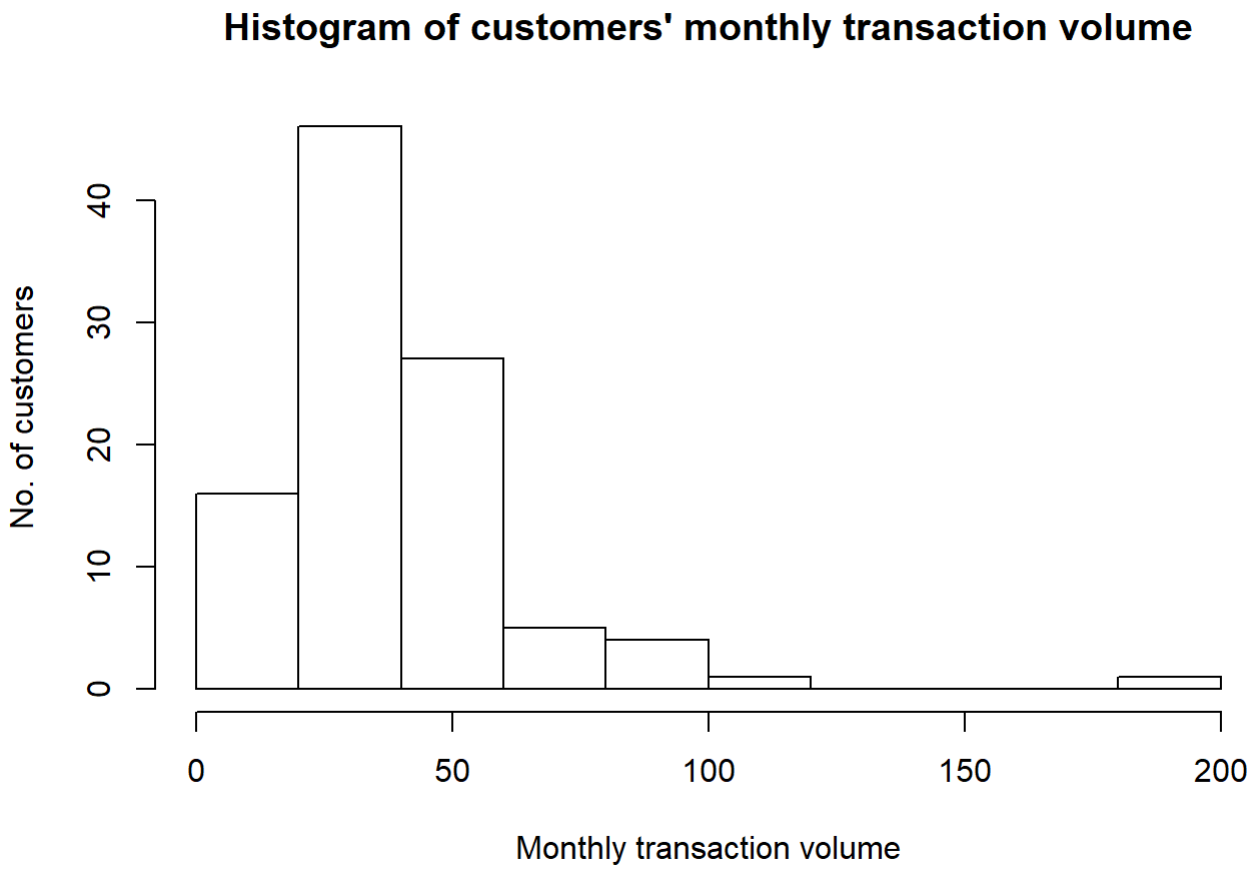



```
hist(df$amount[!df$amount %in% boxplot.stats(df$amount)$out],   #exclude outliers
     xlab= 'Transaction Amount',main = 'Histogram of overall transaction amount')
```



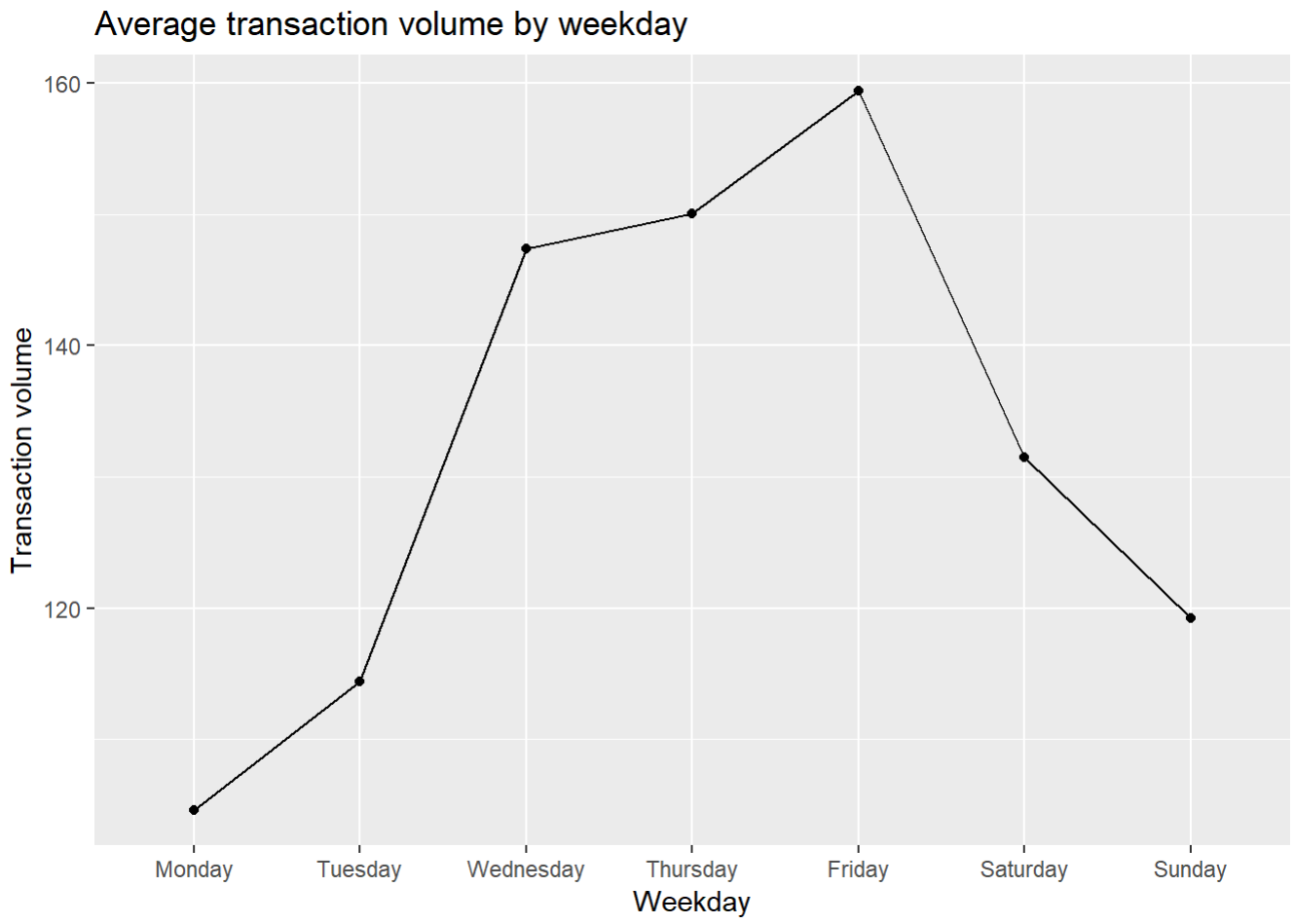
Visualise customers'average monthly transaction volume.

```
df2 <- df %>%
  group_by(customer_id) %>%
  summarise(mon_avg_vol = round(n()/3,0))

hist(df2$mon_avg_vol,
     xlab= 'Monthly transaction volume', ylab='No. of customers', main = "Histogram of customer
s' monthly transaction volume")
```

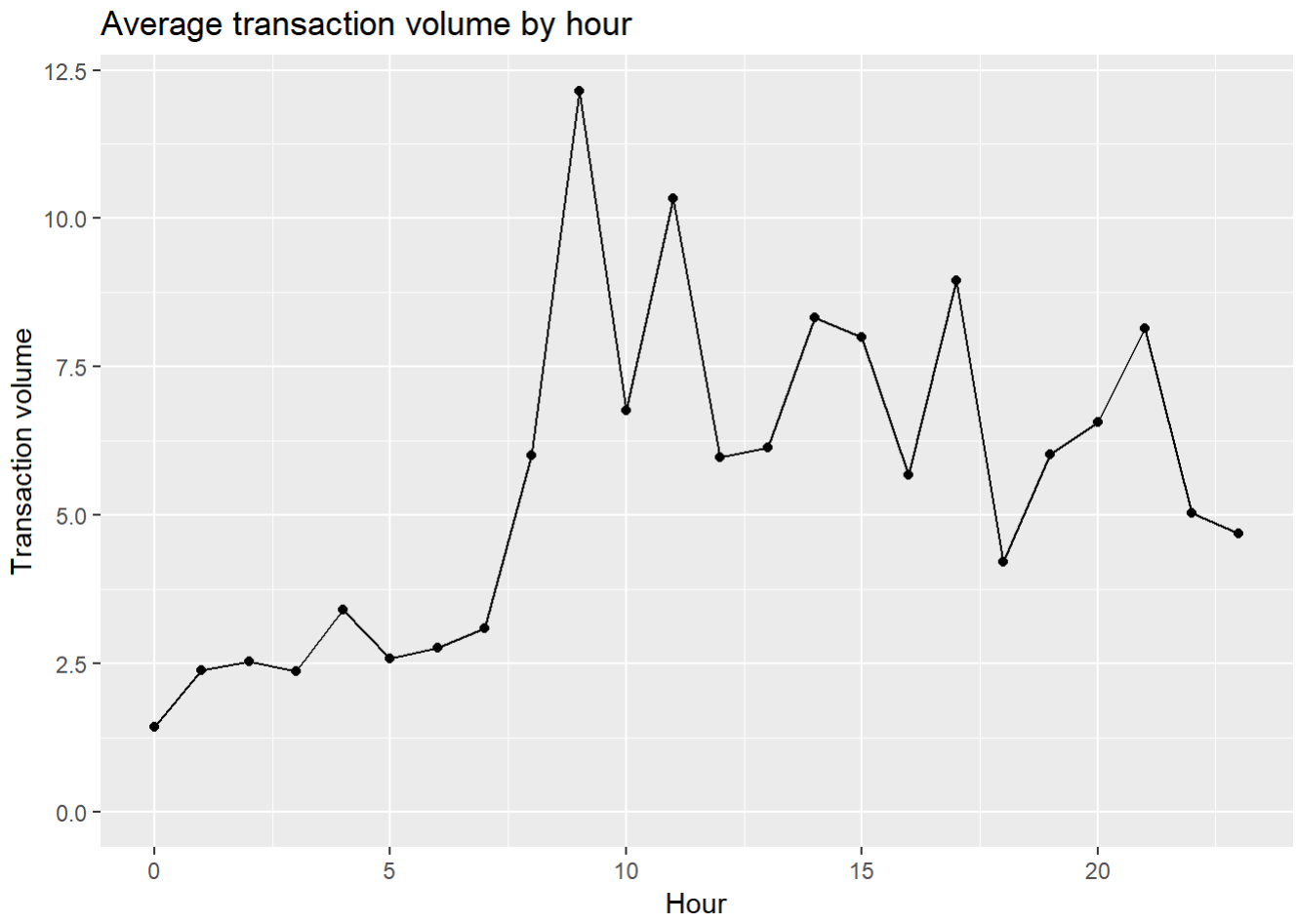


1.4 Segment the dataset by transaction date and time.

```
# Visualise transaction volume over an average week.

df3 <- df %>%
  select(date,weekday) %>%
  group_by(date,weekday) %>%
  summarise(daily_avg_vol = n()) %>%
  group_by(weekday) %>%
  summarise(avg_vol=mean(daily_avg_vol,na.rm=TRUE ))
df3$weekday <- factor(df3$weekday, levels=c( "Monday","Tuesday","Wednesday",
                                  "Thursday","Friday","Saturday","Sunday"))

ggplot(df3,aes(x=weekday, y=avg_vol)) +geom_point()+geom_line(aes(group = 1))+
     ggtitle('Average transaction volume by weekday') +
  labs(x='Weekday',y='Transaction volume')
```



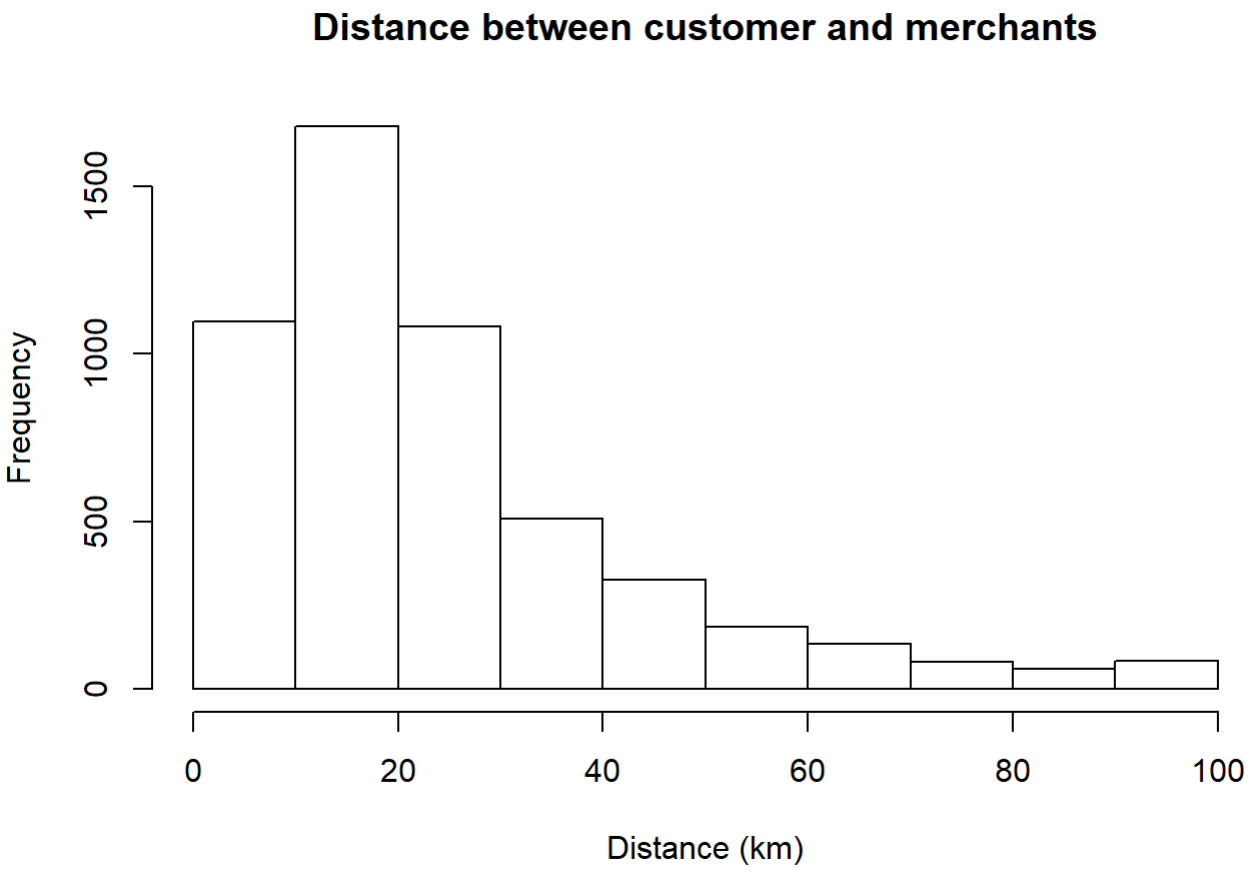

```
# visualize transaction volume over an average week.

df4 <- df %>%
  select(date,hour) %>%
  group_by(date,hour) %>%
  summarize(trans_vol=n()) %>%
  group_by(hour) %>%
  summarize(trans_vol_per_hr = mean(trans_vol,na.rm=TRUE))

 ggplot(df4,aes(x=hour,y=trans_vol_per_hr))+geom_point()+geom_line(aes(group = 1))+
  ggtitle('Average transaction volume by hour') +
  labs(x='Hour',y='Transaction volume') + expand_limits( y = 0)
```



1.5 challenge: exploring location information

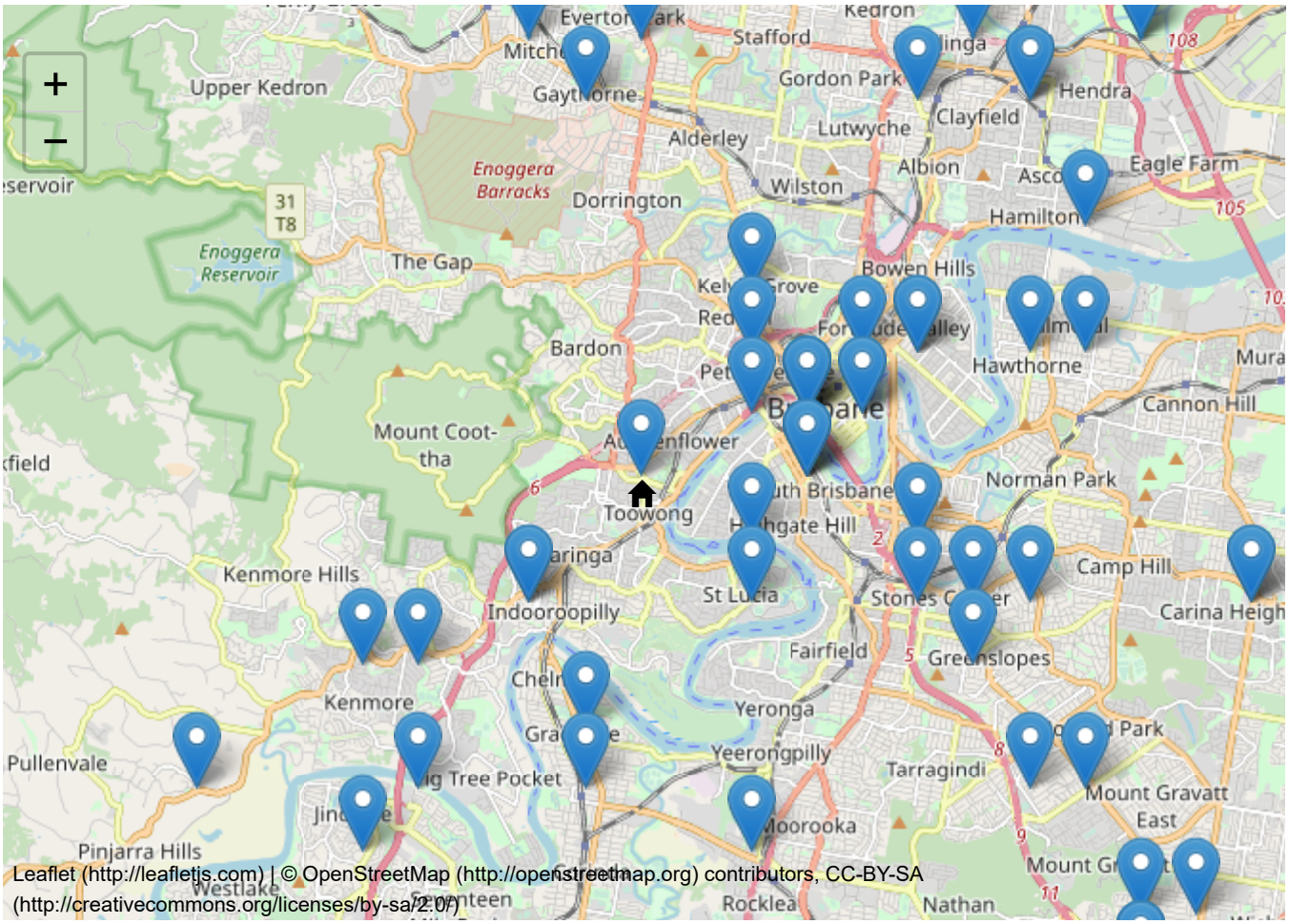
We could firstly see the distribution of distance between a customer and the merchange he/she trades with.

```
# exclude the single foreign customer whose location information was incorrectly stored (i.e lat
itude 573)

df_temp <- df_csmp %>%
  filter (c_long >113 & c_long <154 & c_lat > (-44) & c_lat < (-10))

dfloc = df_temp [,c("c_long", "c_lat","m_long", "m_lat")]
dfloc<- data.frame(sapply(dfloc, as.numeric))

dfloc$dst <- distHaversine(dfloc[, 1:2], dfloc[, 3:4]) / 1000

hist(dfloc$dst[dfloc$dst<100], main = "Distance between customer and merchants",xlab= 'Distance
 (km)' )
```



To validate, we could further plot the location of the customer and the merchants he/she trades with on a map.

```
merch_dist <- function (id ){

  ### This function takes in a customer Id and plot the location of the customer and all
  ### merchants he/she have traded with.

  cus_icon<- makeAwesomeIcon(icon = 'home', markerColor = 'green')

  l = subset (df_csmp[,c("customer_id","m_long","m_lat")], customer_id == id)
  l <- l[c("m_long","m_lat")]

  cus_loc <- unique(subset (df_csmp[,c("customer_id","long_lat")], customer_id == id))
  cus_loc <- cus_loc %>% separate("long_lat", c("long", "lat"),sep=' ')

  df_t = data.frame(longtitude = as.numeric(l$m_long), latitude = as.numeric(l$m_lat))

  coordinates(df_t) <- ~longtitude+latitude
  leaflet(df_t) %>% addMarkers() %>% addTiles() %>%
    addAwesomeMarkers(
    lng=as.numeric(cus_loc$long), lat=as.numeric(cus_loc$lat),
    icon = cus_icon)
}

merch_dist(id ='CUS-51506836' )
```


Upper Kedron
Enoggera Barracks
Enoggera Reservoir
The Gap
Dorrington
Alderley
Gaythorne
Stafford
Gordon Park
Lutwyche
Wilston
Albion
Clayfield
Hendra
Eagle Farm
Hamilton
Bowen Hills
Hawthorne
Cannon Hill
Norman Park
Camp Hill
Carina Heights
Bardon
Mount Coot-tha
Toowong
Indooroopilly
Kenmore Hills
Kenmore
Pullenvale
Pinjarra Hills
St Lucia
Fairfield
Greenslopes
Yeronga
Yeerongpilly
Tarragindi
Mount Gravatt East
Moorooka
Rocklea
Nathan
Leaflet (http://leafletjs.com) | © OpenStreetMap (http://openstreetmap.org) contributors, CC-BY-SA (http://creativecommons.org/licenses/by-sa/2.0/)