Homework-3

Ritwik Bhriguvanshi

2/19/2020

```
knitr::opts_chunk$set(echo = TRUE)

library("tidyverse")

## -- Attaching packages ---------------------------------------------------
---------- tidyverse 1.3.0 --

## v ggplot2 3.2.1     v purrr   0.3.3
## v tibble  2.1.3     v dplyr   0.8.3
## v tidyr   1.0.0     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.4.0

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

library("dplyr")
library("ggplot2")
library(readxl)

sheets <- excel_sheets("online_retail_II.xlsx")
sheet1 <- read_excel("online_retail_II.xlsx", sheet = sheets[1])
sheet2 <- read_excel("online_retail_II.xlsx", sheet = sheets[2])

data1 <- rbind(sheet1, sheet2)
str(data1)

## Classes 'tbl_df', 'tbl' and 'data.frame':    1067371 obs. of  8 variables:
##  $ Invoice    : chr  "489434" "489434" "489434" "489434" ...
##  $ StockCode  : chr  "85048" "79323P" "79323W" "22041" ...
##  $ Description: chr  "15CM CHRISTMAS GLASS BALL 20 LIGHTS" "PINK CHERRY
LIGHTS" "WHITE CHERRY LIGHTS" "RECORD FRAME 7\" SINGLE SIZE" ...
##  $ Quantity   : num  12 12 12 48 24 24 24 10 12 12 ...
##  $ InvoiceDate: POSIXct, format: "2009-12-01 07:45:00" "2009-12-01
07:45:00" ...
##  $ Price      : num  6.95 6.75 6.75 2.1 1.25 1.65 1.25 5.95 2.55 3.75 ...
##  $ Customer ID: num  13085 13085 13085 13085 13085 ...
##  $ Country    : chr  "United Kingdom" "United Kingdom" "United Kingdom"
"United Kingdom" ...

dim(data1)
```

```
## [1] 1067371       8

## Cleaning Date in the Table

f <- '%Y/%m/%d'
data1$InvoiceDate <- as.Date(data1$InvoiceDate, format = f)
head(data1)      ## Check if the Date format has been updated

## # A tibble: 6 x 8
##   Invoice StockCode Description Quantity InvoiceDate Price `Customer ID`
##   <chr>   <chr>     <chr>          <dbl> <date>      <dbl>         <dbl>
## 1 489434  85048     15CM CHRIS~       12 2009-12-01   6.95         13085
## 2 489434  79323P    PINK CHERR~       12 2009-12-01   6.75         13085
## 3 489434  79323W    WHITE CHER~       12 2009-12-01   6.75         13085
## 4 489434  22041     "RECORD FR~       48 2009-12-01   2.1          13085
## 5 489434  21232     STRAWBERRY~       24 2009-12-01   1.25         13085
## 6 489434  22064     PINK DOUGH~       24 2009-12-01   1.65         13085
## # ... with 1 more variable: Country <chr>

## Removing Duplicate rows
library(dplyr)
data1 <- distinct(data1)
dim(data1)     ## Check new dimension of data

## [1] 1033034       8

## Remove cancelled Orders
## Two way:
data1 <- data1[nchar(data1$Invoice) == 6, ]  ## nchar for Invoice will be 7
for cancelled order

## AND
data1 <- data1[data1$Quantity > 0,]   ## Quantity should be positive

## Remove StockCode with less than 5 characters
## These are unusual product items like POST, which we do not want to focus
on
data1 <- data1[nchar(data1$StockCode) >= 5,]

## Changing column name to make to remove space
colnames(data1)[7] <- 'CustomerID'


## Adding Year, Month, and Date Columns in the Dataframe

library(tidyverse)
library(lubridate)

##
## Attaching package: 'lubridate'
```

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

data1 = data1 %>%
  mutate(InvoiceDate = ymd(InvoiceDate)) %>%
  mutate_at(vars(InvoiceDate), funs(year, month, day))

## Warning: funs() is soft deprecated as of dplyr 0.8.0
## Please use a list of either functions or lambdas:
##
##   # Simple named list:
##   list(mean = mean, median = median)
##
##   # Auto named with `tibble::lst()`:
##   tibble::lst(mean, median)
##
##   # Using lambdas
##   list(~ mean(., trim = .2), ~ median(., na.rm = TRUE))
## This warning is displayed once per session.

TotalSales <- data1$Quantity * data1$Price

## Joining a new column for Sales in the Existing Dataframe

data1 <- cbind(data1, TotalSales)

## Dropping missing values from customer ID
q1a <- drop_na(data1, "CustomerID")
dim(data1)      ## Check new dimension of data

## [1] 1006097      12

library(dplyr)

q1a <- q1a %>%
  group_by(year, month, CustomerID) %>%
  summarize(., Count = n(),SumQuantity = sum(Quantity),TotalSpending =
sum(TotalSales))

q1a <- q1a %>%
  group_by(CustomerID) %>%
  summarize(., AverageSpending = mean(TotalSpending)) %>%
  arrange(desc(AverageSpending))

## Filtering out top 20 Average Spending
q1a <- q1a %>%
top_n(20, AverageSpending)

q1a
```

```
## # A tibble: 20 x 2
##    CustomerID AverageSpending
##         <dbl>           <dbl>
##   1     16446          84236.
##   2     15098          39916.
##   3     18102          25260.
##   4     15749          22267.
##   5     14646          21070.
##   6     17450          17485.
##   7     12346          15511.
##   8     14156          12650.
##   9     16000          12394.
## 10      13687          11881.
## 11      14911          10913.
## 12      18052          10877.
## 13      14096          10652.
## 14      14028          10396.
## 15      12415          10288.
## 16      12590           9341.
## 17      14088           9148.
## 18      12357           8719.
## 19      13902           8506.
## 20      18139           8438.
```

```
## Filtering top 20 customer IDs by putting it back in the original data

q1aNewData <- data1 %>%
  select(CustomerID,Invoice,year, month, Quantity, TotalSales, Country) %>%
  filter(., CustomerID %in% c(16446,15098,18102,15749,
                              14646,17450,12346,14156,
                              16000, 13687,14911,18052,
                              14096,14028,12415 ,12590 ,
                              14088, 12357,13902,18139)) %>%
  group_by(.,CustomerID) %>%
  summarize(., Count = n(), TotalQuantity = sum(Quantity), AmountSpent =
sum(TotalSales)) %>%
  arrange(desc(AmountSpent))
```

Response: After cleaning the data and removing the rows which we do not want to include in our analysis, the following customerID were generated with the highest Average Spending per month. After finding these Cutsomer IDs, I filtered the original data (stored as data1) with these customer IDs.

Significant characteristics: Most of these Customer IDs are wholesalers who have an high frequency of occurence. Some of these are regular customers who constantly buy products from this retail store as they have a very high amount of quantity bought and amount spent. Examples of these customer IDS are: **18102**, **14646**, **14156**, **14911**

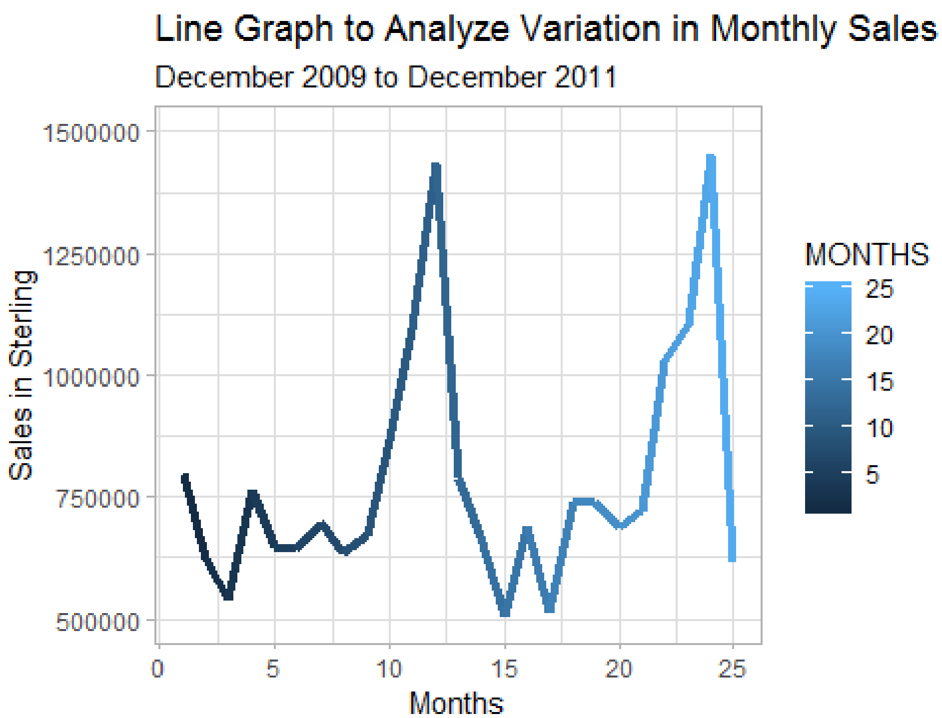
However, few of the high spending customers have a very low frequency count. For example, CustomerID- **16446** has a frequency count of only 3, but the total amount spent is

168472, which is the sixth highest. This is the reason why this customer has the highest average monthly sales.

There are also a few customers who buy products in less quantity, but the products are expensive, making them a part of customers with high spendings. An example of this type can be Customer ID - **15098** has bought total quantity of 121, but has a spending of **39916**

```
## We are using the data1 dataset because we will use the sales for customer
IDs with NA

q1b <- data1 %>%
  select(Invoice, StockCode, Description, Quantity, Invoice, Price,
CustomerID, year, month, day,          TotalSales) %>%
  group_by(.,year, month) %>%
  summarize(., Count = n(), QuantitySold = sum(Quantity), Sales =
sum(TotalSales))

q1b

## # A tibble: 25 x 5
## # Groups:   year [3]
##     year month Count QuantitySold   Sales
##    <dbl> <dbl> <int>        <dbl>   <dbl>
##  1  2009    12 43495       444025 798217.
##  2  2010     1 30322       395033 621353.
##  3  2010     2 27896       391603 537942.
##  4  2010     3 39724       529332 761749.
##  5  2010     4 32761       385120 646519.
##  6  2010     5 33407       422586 643655.
##  7  2010     6 38366       413148 697412.
##  8  2010     7 32006       357758 633139.
##  9  2010     8 32130       521324 674238.
## 10  2010     9 40496       592122 869322.
## # ... with 15 more rows

## Plot to see the monthly change in sales
library(ggplot2)

MONTHS <- 1:25   ## Create Dummy Variable for 25 months


mygraph1 <- ggplot() + geom_line(aes(y = Sales, x = MONTHS, colour = MONTHS),
size = 1.5,
          data = q1b, stat="identity") + geom_smooth(method = lm)

mygraph1 <- mygraph1 +
            theme_light() +
            ylim(500000,1500000)+
            labs(
            x = "Months",
```

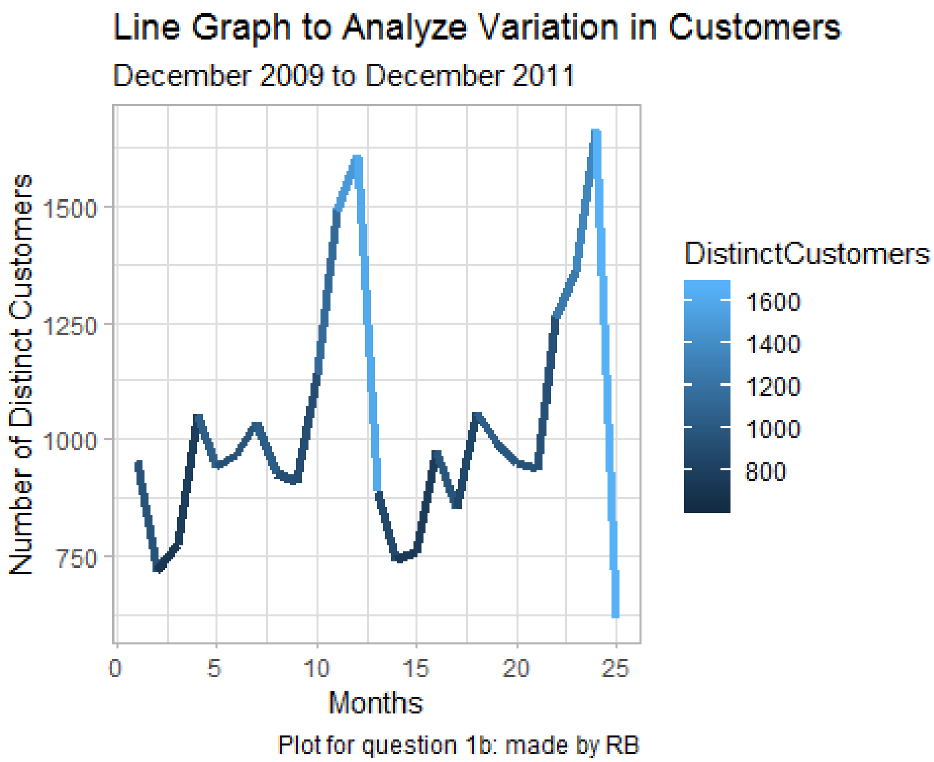
```
            y = "Sales in Sterling",
            title = "Line Graph to Analyze Variation in Monthly Sales",
            subtitle = "December 2009 to December 2011",
            caption = "Plot for question 1b: made by RB")

mygraph1
```



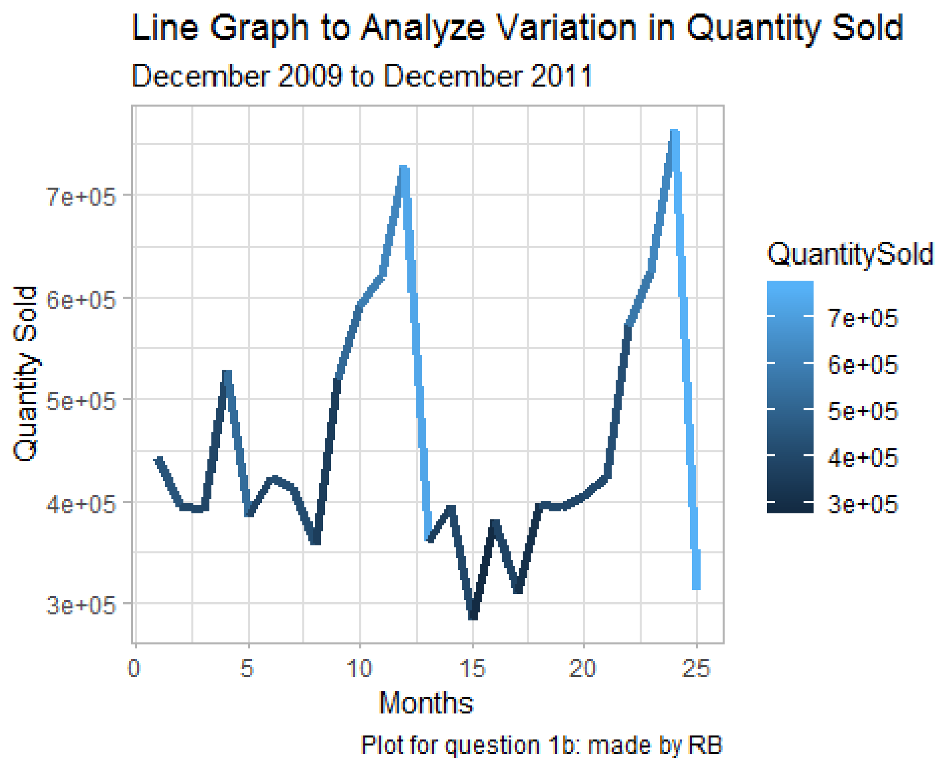
Plot for question 1b: made by RB

```
## Number of Customers by each Month

q1b <- data1 %>%
  select(Invoice, StockCode, Description, Quantity, Price, CustomerID, year,
month, day,TotalSales) %>%
  group_by(.,year, month) %>%
  summarize(., Count = n(), QuantitySold = sum(Quantity), Sales =
sum(TotalSales), DistinctCustomers = n_distinct(CustomerID))

mygraph2 <- ggplot() + geom_line(aes(y = DistinctCustomers, x = MONTHS,
colour = DistinctCustomers), size = 1.5,
          data = q1b, stat="identity") + geom_smooth(method = lm)

mygraph2 <- mygraph2 +
            theme_light() +
            labs(
            x = "Months",
            y = "Number of Distinct Customers",
            title = "Line Graph to Analyze Variation in Customers",
```

```
            subtitle = "December 2009 to December 2011",
            caption = "Plot for question 1b: made by RB")

mygraph2
```




```
## Graph for Quantity Sold

mygraph3 <- ggplot() + geom_line(aes(y = QuantitySold, x = MONTHS, colour =
QuantitySold), size = 1.5,
          data = q1b, stat="identity") + geom_smooth(method = lm)

mygraph3 <- mygraph3 +
            theme_light() +
            labs(
            x = "Months",
            y = "Quantity Sold",
            title = "Line Graph to Analyze Variation in Quantity Sold",
            subtitle = "December 2009 to December 2011",
            caption = "Plot for question 1b: made by RB")

mygraph3
```



Response:

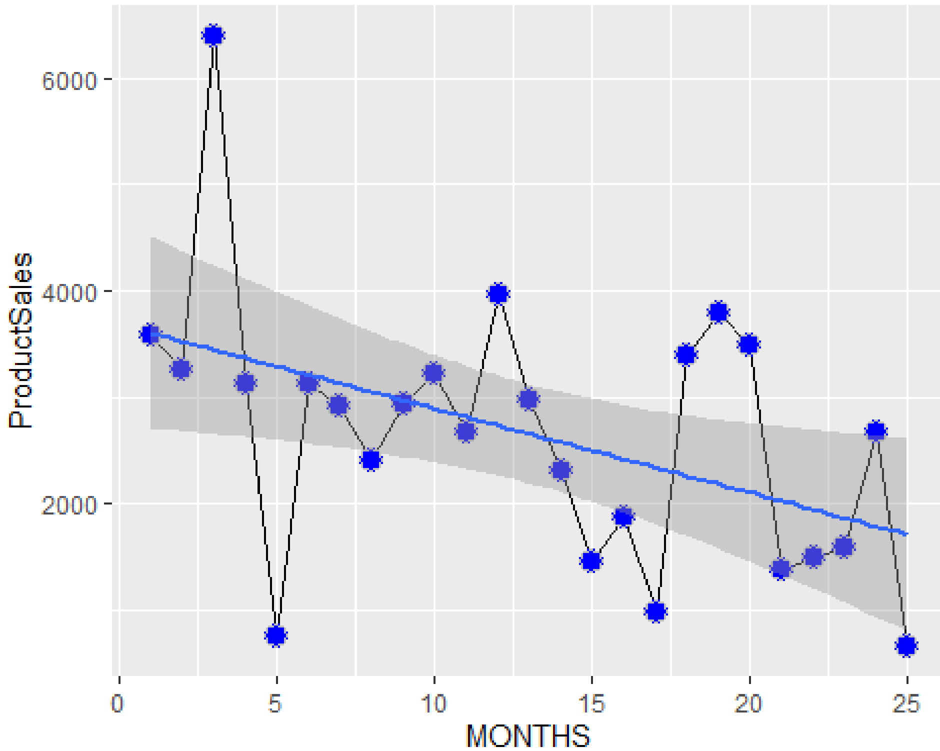
I created a dummy variable to store the data as a spread of 25 months starting from the following dates: Month 1 – December 2009, and goes on to Month 25 – December 2011

The three graphs show two peaks during the 25 months of data provided. The peaks are during the month of October and November which suggests that customers are trying to stock up to prepare for the Holiday season (Thanksgiving and Christmas Holidays). The sales remain pretty much constant for the rets of the months.

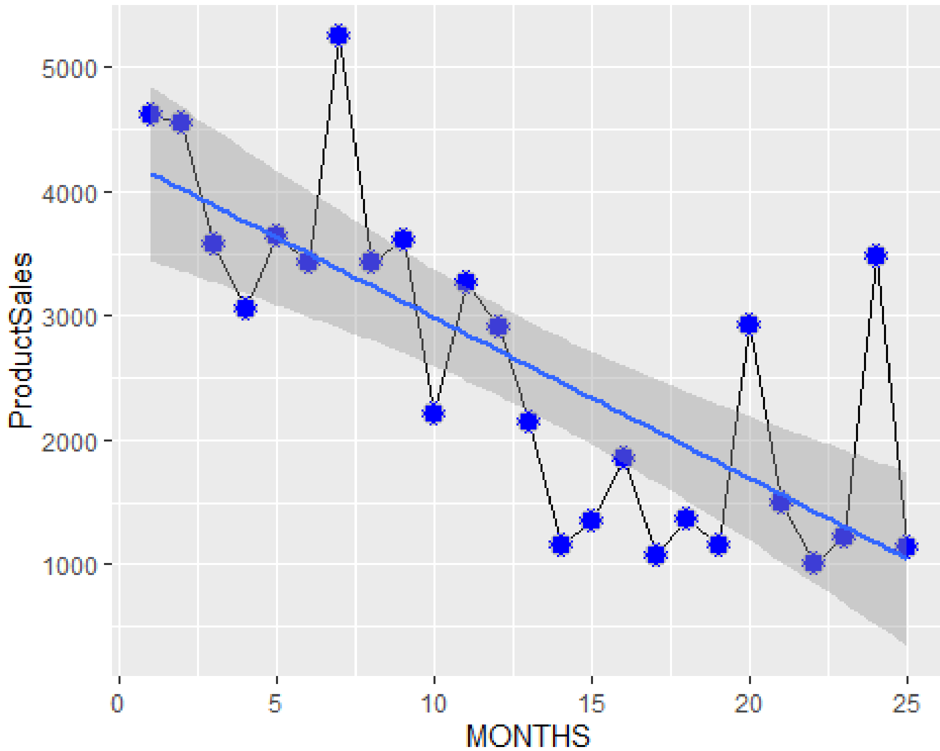
Graph2 depicts number of distinct customers shopping during the entirety of 25 months and Graph3 depicts the total quantity of products being sold over 25 months. We can infer that these two graphs have the pea around the same time. Hence, both the reasons- quanity of products and number of customers attribute to variation in sales.

```
products <- data1 %>%
  group_by(StockCode) %>%
  summarize(ProductSales = sum(TotalSales))

## Finding 25 customers with the most sales
Top_Products <- head(products[order(products$ProductSales, decreasing =
T),],25)

## Subsetting

q1c <- data1 %>%
  filter(StockCode %in% Top_Products$StockCode) %>%
  group_by(StockCode, year, month) %>%
```

```r
  summarize(ProductSales = sum(TotalSales))

## Adding a column to help form a for looop
q1c$Sequence <- seq(1, nrow(q1c))

## LOOP to find significant P-values

Pvalues <- c()
for (i in 1:nrow(q1c)){
  model <- (lm(ProductSales ~ Sequence, data = q1c[q1c$StockCode ==
as.character(q1c$StockCode[i]),]))
  if(nrow(summary(model)$coefficients) == 2) {
  Pvalues[i] <- (summary(model)$coefficients[2,4])
  }
}

(significant <- which(Pvalues <= 0.05))

##   [1]   1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17
##  [18]  18  19  20  21  22  23  24  25  51  52  53  54  55  56  57  58  59
##  [35]  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75 101
##  [52] 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118
##  [69] 119 120 121 122 123 124 125 196 197 198 199 200 201 202 203 204 205
##  [86] 206 207 208 209 210 211 212 213 214 215 216 217 218 219 220 343 344
## [103] 345 346 347 348 349 350 351 352 353 354 355 356 357 358 359 360 361
## [120] 362 363 364 365 366 367 368 369 370 371 372 373 374 375 376 377 378
## [137] 379 380 381 382 383 384 385 386 387 388 389 390 391 392 517 518 519
## [154] 520 521 522 523 524 525 526 527 528 529 530 531 532 533 534 535 536
## [171] 537 538 539 540 541

(unique(q1c[significant,]$StockCode))     ## Find the significant products
with variation

## [1] "20685"  "21137"  "21843"  "22197"  "48138"  "79321"  "85123A"

mygraph4 <- q1c[q1c$StockCode == '20685',] %>%
  ggplot(aes(y = ProductSales, x = MONTHS)) +
  geom_line(color = "black") +
  geom_point(shape = 21, color = 'gray', fill = "blue", size = 4)+
  geom_smooth(method = "lm")
mygraph4
```




```
mygraph5 <- q1c[q1c$StockCode == '21843',] %>%
  ggplot(aes(y = ProductSales, x = MONTHS)) +
  geom_line(color = "black") +
  geom_point(shape = 21, color = 'gray', fill = "blue", size = 4)+
  geom_smooth(method = "lm")
mygraph5
```



Response: ProductID with variation in sales:

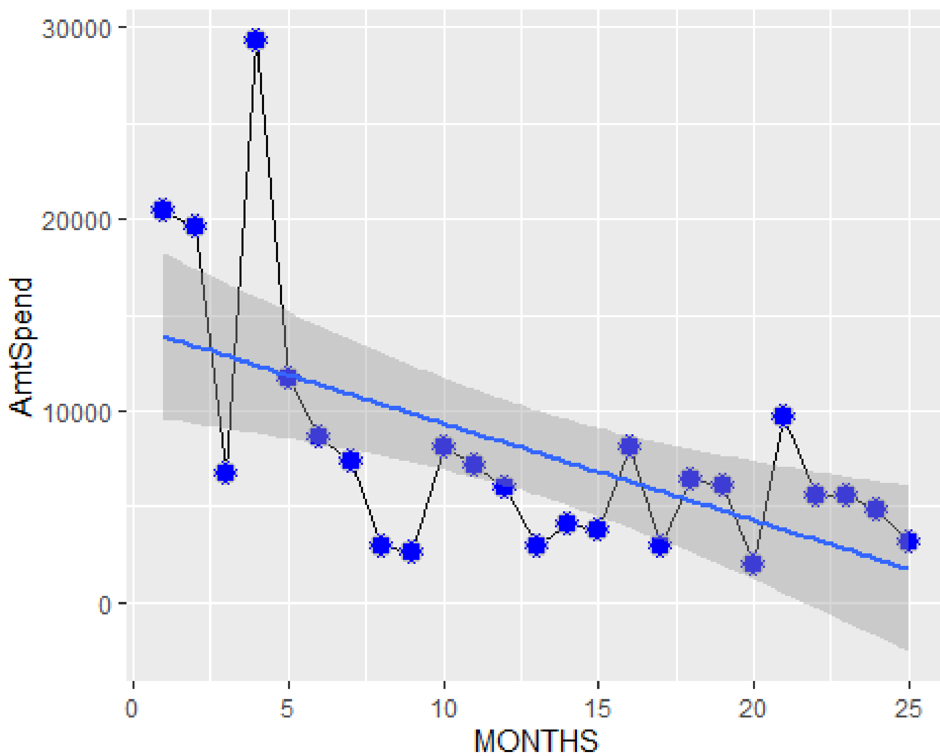
“20685” “21137” “21843” “22197” “48138” “79321” “85123A”

I subsetted the products after narrowing it on total sales. I then found out the product with 25 highest sales. After grouping these products on month and year, and finding significant products, there were seven different product IDs with monthly variation in sales.

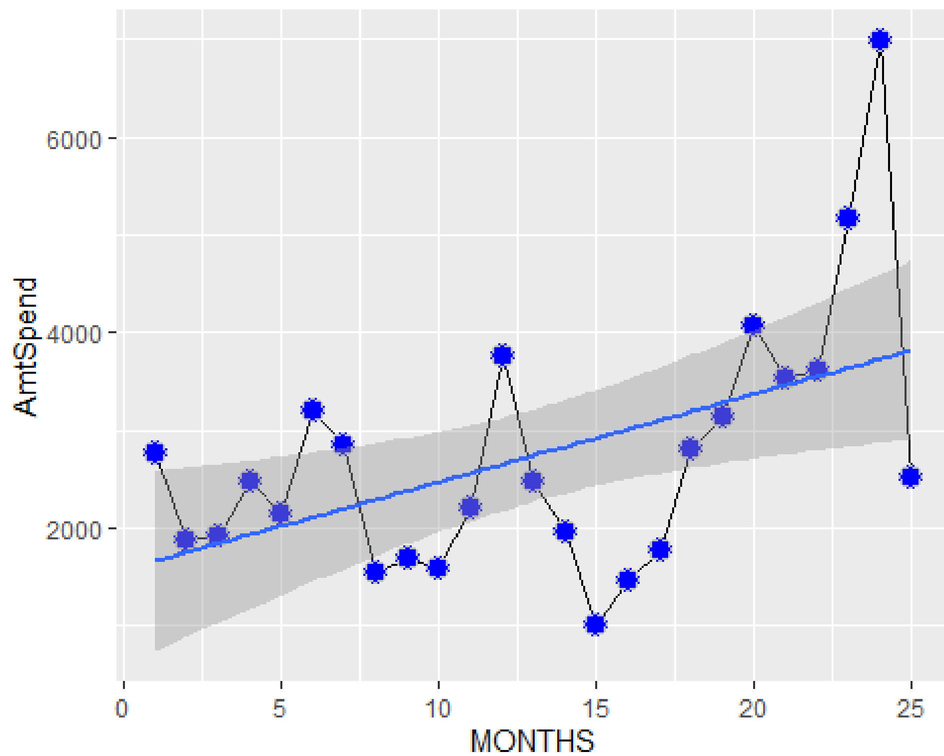
Graph4 and Graph5 shows the variation in sales of the first two listed products.

```
customers <- drop_na(data1, "CustomerID")

customers <- customers %>%
  group_by(CustomerID) %>%
  summarize(CustomerSpending = sum(TotalSales))

## Finding 25 customers with the most sales
Top_Customers <- head(customers[order(customers$CustomerSpending, decreasing
= T),],25)

## Subsetting

q1d <- data1 %>%
  filter(CustomerID %in% Top_Customers$CustomerID) %>%
  group_by(CustomerID, year, month) %>%
  summarize(AmtSpend = sum(TotalSales))
```

```r
## Adding a column to help form a for looop
q1d$Sequence <- seq(1, nrow(q1d))

## LOOP

Pvalues <- c()
for (i in 1:nrow(q1d)){
  model <- lm(AmtSpend ~ Sequence, data = q1d[q1d$CustomerID
==as.character(q1d$CustomerID[i]),])
  Pvalues[i] <- summary(model)$coefficients[2,4]
}

summary(Pvalues)

##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.     NA's 
## 0.002525 0.163346 0.272222 0.331407 0.468746 0.888850        2

(significant <- which(Pvalues <= 0.05))

##  [1]  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80
## [18]  81  82  83  84  85  86  87  88 425 426 427 428 429 430 431 432 433
## [35] 434 435 436 437 438 439 440 441 442 443 444 445 446 447 448 449

(unique(q1d[significant,]$CustomerID))

## [1] 13694 17841

library(ggplot2)

mygraph6 <- q1d[q1d$CustomerID == '13694',] %>%
  ggplot(aes(y = AmtSpend, x = MONTHS)) +
  geom_line(color = "black") +
  geom_point(shape = 21, color = 'gray', fill = "blue", size = 4)+
  geom_smooth(method = "lm")
mygraph6
```




```
mygraph7 <- q1d[q1d$CustomerID == '17841',] %>%
  ggplot(aes(y = AmtSpend, x = MONTHS)) +
  geom_line(color = "black") +
  geom_point(shape = 21, color = 'gray', fill = 'blue', size = 4)+
  geom_smooth(method = "lm")
mygraph7
```



Response: CustomerID with variation in sales:

'13694' '17841'

I subsetted the products after narrowing it on total sales. I then found out the product with 25 highest sales. After grouping these products on month and year, and finding significant products, there were two different product IDs with monthly variation in sales.

Graph6 and Graph7 shows the variation in sales of the first two listed products.

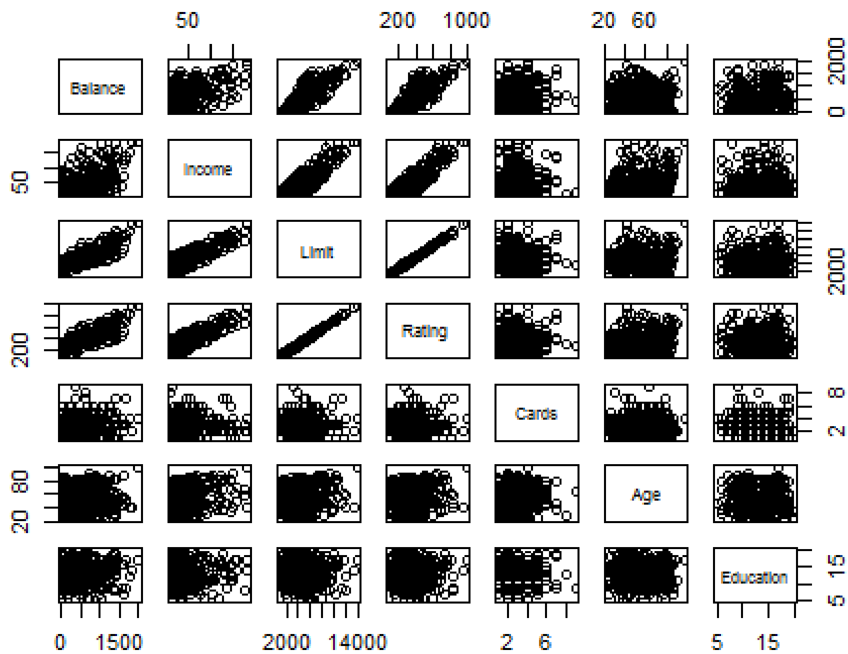
```
r <- read.csv("Credit_homework3.csv",header = TRUE)
head(r)
```

```
##   ID  Income Limit Rating Cards Age Education Gender Student Married
## 1  1  14.891  3606    283     2  34        11   Male      No     Yes
## 2  2 106.025  6645    483     3  82        15 Female     Yes     Yes
## 3  3 104.593  7075    514     4  71        11   Male      No      No
## 4  4 148.924  9504    681     3  36        11 Female      No      No
## 5  5  55.882  4897    357     2  68        16   Male      No     Yes
## 6  6  80.180  8047    569     4  77        10   Male      No      No
##   Ethnicity Balance
## 1 Caucasian     333
## 2     Asian     903
## 3     Asian     580
## 4     Asian     964
## 5 Caucasian     331
## 6 Caucasian    1151
```

```
str(r)

## 'data.frame':    400 obs. of  12 variables:
##  $ ID       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Income   : num  14.9 106 104.6 148.9 55.9 ...
##  $ Limit    : int  3606 6645 7075 9504 4897 8047 3388 7114 3300 6819 ...
##  $ Rating   : int  283 483 514 681 357 569 259 512 266 491 ...
##  $ Cards    : int  2 3 4 3 2 4 2 2 5 3 ...
##  $ Age      : int  34 82 71 36 68 77 37 87 66 41 ...
##  $ Education: int  11 15 11 11 16 10 12 9 13 19 ...
##  $ Gender   : Factor w/ 2 levels " Male","Female": 1 2 1 2 1 1 2 1 2 2 ...
##  $ Student  : Factor w/ 2 levels "No","Yes": 1 2 1 1 1 1 1 1 1 2 ...
##  $ Married  : Factor w/ 2 levels "No","Yes": 2 2 1 1 2 1 1 1 1 2 ...
##  $ Ethnicity: Factor w/ 3 levels "African American",..: 3 2 2 2 3 3 1 2 3
1 ...
##  $ Balance  : int  333 903 580 964 331 1151 203 872 279 1350 ...

library(ggplot2)
library(leaps)
library(GGally)

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

##
## Attaching package: 'GGally'

## The following object is masked from 'package:dplyr':
##
##     nasa

library(lmtest)

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

library(car)

## Loading required package: carData

##
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':
##
##     recode
```

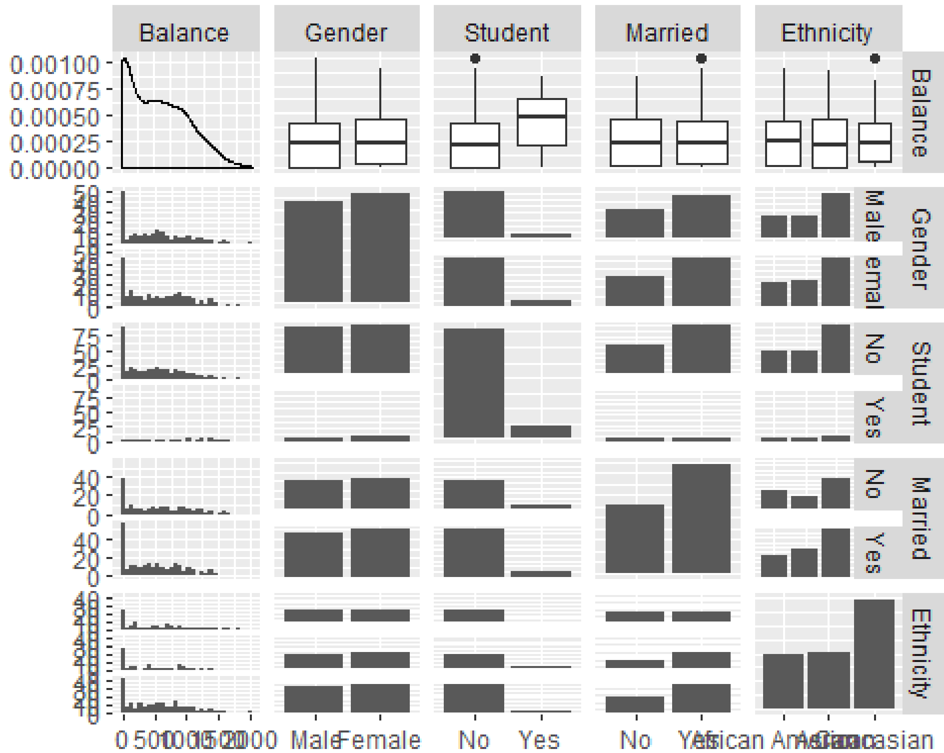
```
## The following object is masked from 'package:purrr':
##
##     some
```

```r
## Relationship of Balance with numerical data
pairs(r[,c("Balance" ,'Income','Limit', 'Rating', 'Cards', 'Age',
'Education')])
```




```r
## Relationship of Balance with categorical data
library(GGally)
ggpairs(r[,c("Balance" ,'Gender','Student', 'Married', 'Ethnicity')])
```

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



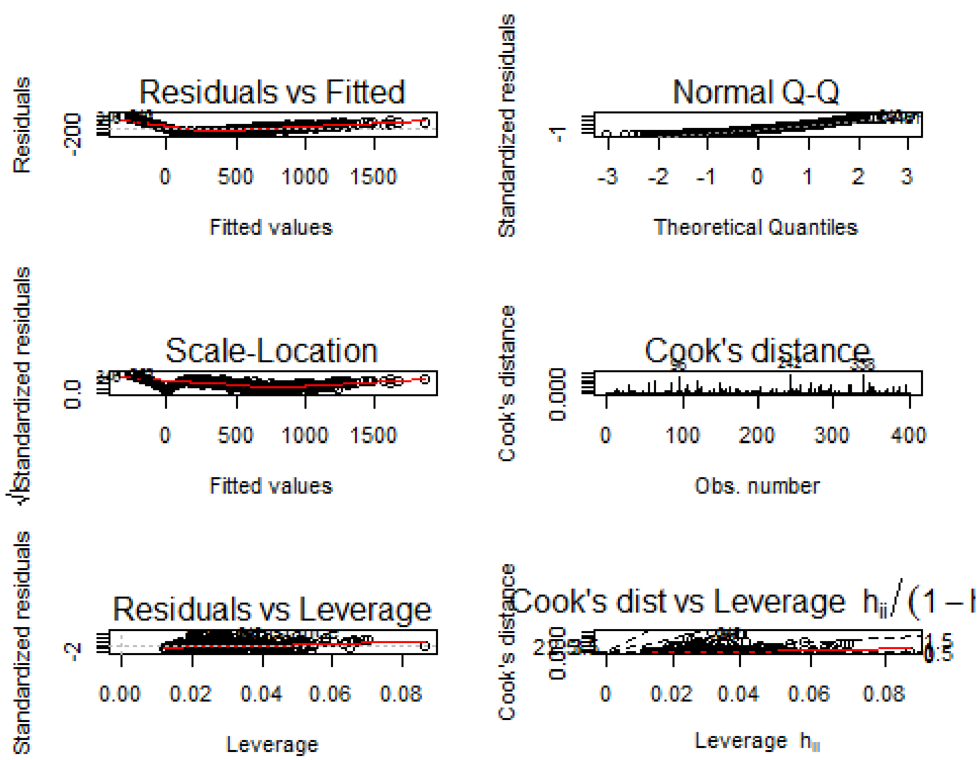
Response: From the pair-wise matrix, we can infer that Ratings and Credit limit are two important factors that have an impact on balance and posses positive linear association with balance. The status of being a also has a significant impact on balance. The level of education variable can be used to explain the credit card balance to a certain extent.

For categorical variables- married, gender, and ethnicity does not effect the balance since the median values for all the different types are in the same range.

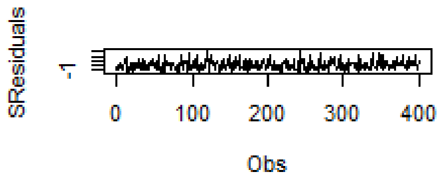
```
MLR <- lm(Balance ~ Income + Limit + Rating + Cards + Age + Education +
Gender + Student + Married + Ethnicity ,data=r)
summary(MLR)

##
## Call:
## lm(formula = Balance ~ Income + Limit + Rating + Cards + Age +
##     Education + Gender + Student + Married + Ethnicity, data = r)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -161.64  -77.70  -13.49   53.98  318.20
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -479.20787   35.77394 -13.395  < 2e-16 ***
## Income                -7.80310    0.23423 -33.314  < 2e-16 ***
## Limit                  0.19091    0.03278   5.824 1.21e-08 ***
## Rating                 1.13653    0.49089   2.315   0.0211 *
```

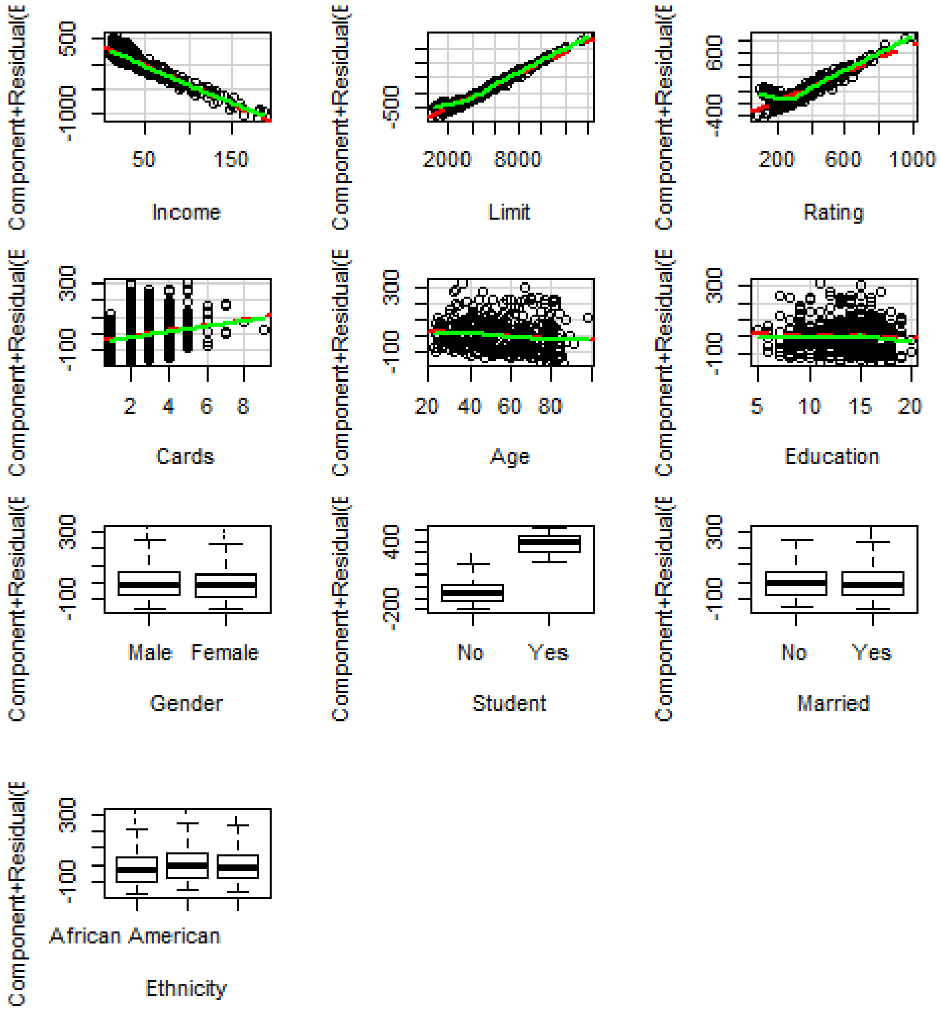
```
## Cards                17.72448    4.34103   4.083 5.40e-05 ***
## Age                  -0.61391    0.29399  -2.088   0.0374 *
## Education            -1.09886    1.59795  -0.688   0.4921
## GenderFemale        -10.65325    9.91400  -1.075   0.2832
## StudentYes          425.74736   16.72258  25.459  < 2e-16 ***
## MarriedYes           -8.53390   10.36287  -0.824   0.4107
## EthnicityAsian       16.80418   14.11906   1.190   0.2347
## EthnicityCaucasian   10.10703   12.20992   0.828   0.4083
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 98.79 on 388 degrees of freedom
## Multiple R-squared:  0.9551, Adjusted R-squared:  0.9538
## F-statistic: 750.3 on 11 and 388 DF,  p-value: < 2.2e-16
```

```r
SResiduals <- rstandard(MLR)   ## Standardized Residuals

par (mfrow=c(3,2))
plot (MLR,c(1,2,3,4,5,6))
```

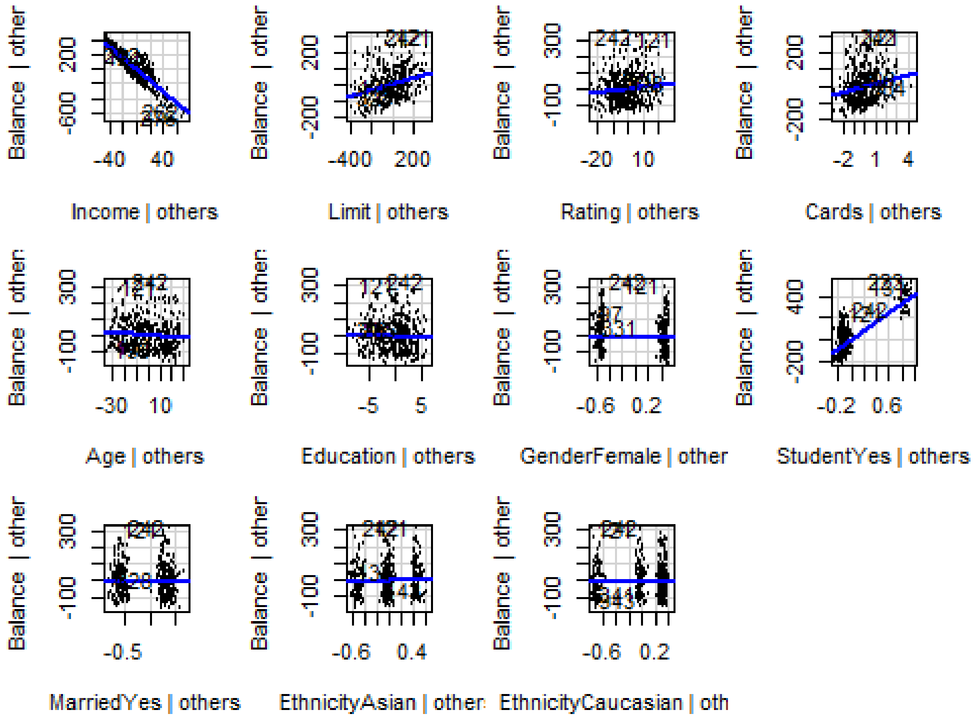



```r
plot.ts(SResiduals,xlab="Obs")


crPlots(MLR,main="",cex=0.5, id.n = 3, id.cex = 0.8,
col.lines=c("red","green"))
```


SResiduals
-1
0
100
200
300
400
Obs




```
avPlots(MLR,cex=0.5,id.n = 3,id.cex = 0.8,main="", layout =c(3,4))

## Warning in plot.window(...): "id.n" is not a graphical parameter
```




```
## Multicollinearity

x <- model.matrix(MLR)[,-1]
corr_x<-cor(x);round(corr_x,3)

##                    Income  Limit Rating  Cards    Age Education
## Income              1.000  0.792  0.791 -0.018  0.175    -0.028
## Limit               0.792  1.000  0.997  0.010  0.101    -0.024
## Rating              0.791  0.997  1.000  0.053  0.103    -0.030
## Cards              -0.018  0.010  0.053  1.000  0.043    -0.051
## Age                 0.175  0.101  0.103  0.043  1.000     0.004
## Education          -0.028 -0.024 -0.030 -0.051  0.004     1.000
## GenderFemale       -0.011  0.009  0.009 -0.023  0.004    -0.005
## StudentYes          0.020 -0.006 -0.002 -0.026 -0.030     0.072
## MarriedYes          0.036  0.031  0.037 -0.010 -0.073     0.049
## EthnicityAsian     -0.017 -0.032 -0.036  0.006 -0.060     0.030
## EthnicityCaucasian -0.020 -0.003 -0.001 -0.006 -0.001    -0.038
##                    GenderFemale StudentYes MarriedYes EthnicityAsian
## Income                   -0.011      0.020      0.036         -0.017
## Limit                     0.009     -0.006      0.031         -0.032
## Rating                    0.009     -0.002      0.037         -0.036
## Cards                    -0.023     -0.026     -0.010          0.006
## Age                       0.004     -0.030     -0.073         -0.060
## Education                -0.005      0.072      0.049          0.030
## GenderFemale              1.000      0.055      0.012          0.025
## StudentYes                0.055      1.000     -0.077          0.054
## MarriedYes                0.012     -0.077      1.000          0.089
```

```
## EthnicityAsian            0.025      0.054      0.089           1.000
## EthnicityCaucasian       -0.010     -0.048      0.011          -0.582
##                    EthnicityCaucasian
## Income                         -0.020
## Limit                          -0.003
## Rating                         -0.001
## Cards                          -0.006
## Age                            -0.001
## Education                      -0.038
## GenderFemale                   -0.010
## StudentYes                     -0.048
## MarriedYes                      0.011
## EthnicityAsian                 -0.582
## EthnicityCaucasian              1.000
```

```
vif(MLR)
```

```
##                 GVIF Df GVIF^(1/(2*Df))
## Income      2.786182  1        1.669186
## Limit     234.028100  1       15.297977
## Rating    235.848259  1       15.357352
## Cards       1.448690  1        1.203615
## Age         1.051410  1        1.025383
## Education   1.019588  1        1.009747
## Gender      1.005849  1        1.002920
## Student     1.031517  1        1.015636
## Married     1.044638  1        1.022075
## Ethnicity   1.032231  2        1.007962
```

```
## Box - Cox Transformation

b <- summary(powerTransform (cbind(Income,Limit,Rating,Cards,Age,Education,
(Balance+0.01))~1, data=r))
b
```

```
## bcPower Transformations to Multinormality
##           Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Income       0.3500        0.33       0.2423       0.4576
## Limit        0.6938        0.69       0.6269       0.7607
## Rating       0.6139        0.61       0.5341       0.6936
## Cards        0.3666        0.50       0.1997       0.5335
## Age          0.7968        1.00       0.4814       1.1121
## Education    1.4595        1.46       1.0781       1.8408
##              0.3956        0.40       0.3643       0.4269
##
## Likelihood ratio test that transformation parameters are equal to 0
##  (all log transformations)
##                                      LRT df       pval
## LR test, lambda = (0 0 0 0 0 0 0) 1069.341  7 < 2.22e-16
##
## Likelihood ratio test that no transformations are needed
```

```
##                                         LRT df       pval
## LR test, lambda = (1 1 1 1 1 1 1) 1124.191  7 < 2.22e-16
```

```
round(b$result,2)
```

```
##             Est Power Rounded Pwr Wald Lwr Bnd Wald Upr Bnd
## Income           0.35        0.33         0.24         0.46
## Limit            0.69        0.69         0.63         0.76
## Rating           0.61        0.61         0.53         0.69
## Cards            0.37        0.50         0.20         0.53
## Age              0.80        1.00         0.48         1.11
## Education        1.46        1.46         1.08         1.84
##                  0.40        0.40         0.36         0.43
```

```
r$Income2 <- r$Income^0.33
r$Limit2 <- r$Limit^0.69
r$Rating2 <- r$Rating^0.61
r$Cards2 <- r$Cards^0.50
r$Age2 <- r$Age^1
r$Education2 <- r$Education^1.46
```

```
head(r)
```

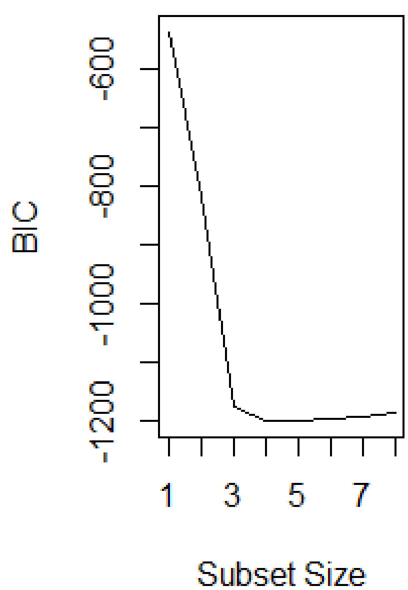
```
##   ID  Income Limit Rating Cards Age Education Gender Student Married
## 1  1  14.891  3606    283     2  34        11   Male      No     Yes
## 2  2 106.025  6645    483     3  82        15 Female     Yes     Yes
## 3  3 104.593  7075    514     4  71        11   Male      No      No
## 4  4 148.924  9504    681     3  36        11 Female      No      No
## 5  5  55.882  4897    357     2  68        16   Male      No     Yes
## 6  6  80.180  8047    569     4  77        10   Male      No      No
##   Ethnicity Balance  Income2   Limit2  Rating2   Cards2 Age2 Education2
## 1 Caucasian     333 2.438175 284.6740 31.30327 1.414214   34   33.14617
## 2     Asian     903 4.659987 434.0318 43.37183 1.732051   82   52.13066
## 3     Asian     580 4.639123 453.2223 45.04924 2.000000   71   33.14617
## 4     Asian     964 5.212887 555.5913 53.48349 1.732051   36   33.14617
## 5 Caucasian     331 3.772244 351.6024 36.06846 1.414214   68   57.28160
## 6 Caucasian    1151 4.249539 495.3218 47.93122 2.000000   77   28.84032
```

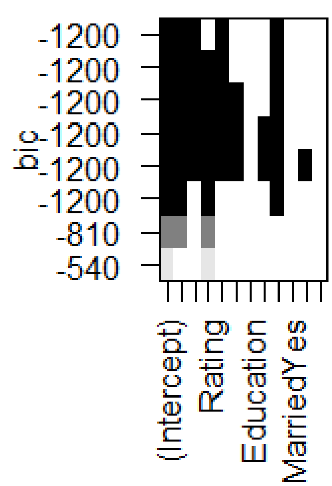
Response: There is evidence of non-constant variance, as fitted and residuals show non-linear relationship. The errors are not normally distributed (in the qq-plot), as it has long tail. The variable which has non-linear relationship with Balance are Rating and Limit.

There is existence of multi-collinearity as large coefficients are present in the correlation matrix.

```
## Variable Selection

VS <- regsubsets(Balance ~ Income + Limit + Rating + Cards + Age + Education
+ Gender + Student + Married + Ethnicity ,data = r)

VSSummary <- summary(VS)
```

```
names(summary(VS))

## [1] "which"  "rsq"    "rss"    "adjr2"  "cp"     "bic"    "outmat" "obj"

c("BIC" = which.min(VSSummary$bic))

## BIC
##   4

min(VSSummary$bic)

## [1] -1198.053

par(mfrow=c(1,2))
plot(VSSummary$bic, xlab="Subset Size", ylab="BIC", type='l')

plot(VS, ylab = "BIC")
```





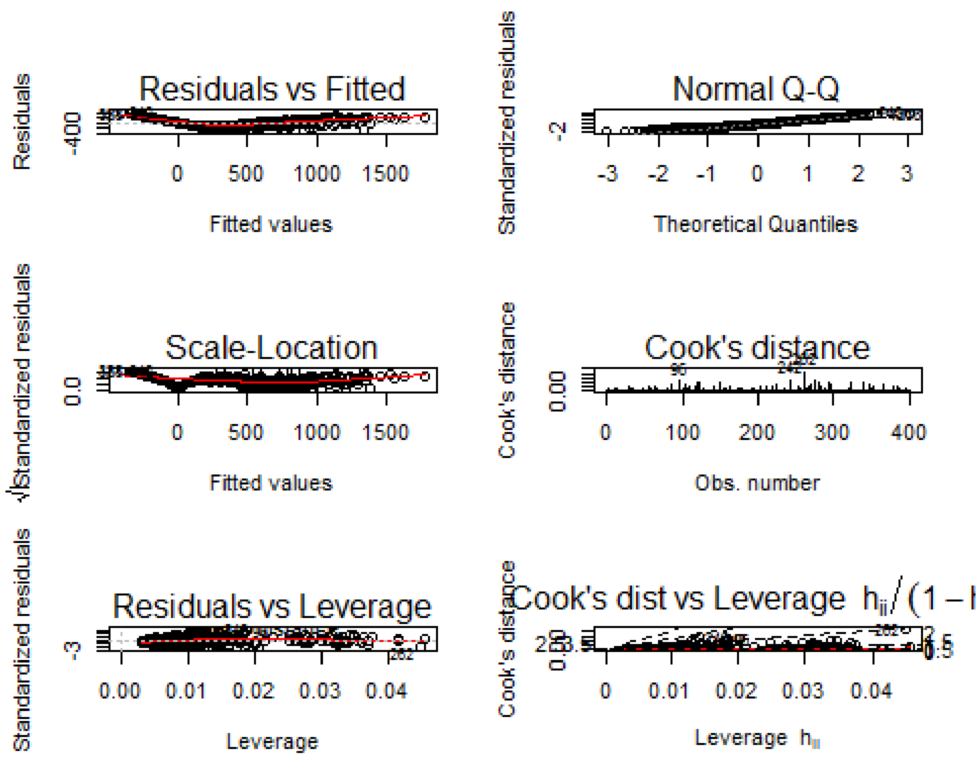

```
coef(VS,4)  ## Income, Limit, Cards, Student are the best predictor
variables.

##   (Intercept)        Income         Limit         Cards    StudentYes
## -499.7272117    -7.8392288     0.2666445    23.1753794   429.6064203
```

Response: Using the best group of variables for the model and the BIC criterion, we pick the model that incorporates Income, Limit, Cards, and Student Status. As long as the BIC and complexity is same, the method of subset selection is appropriate

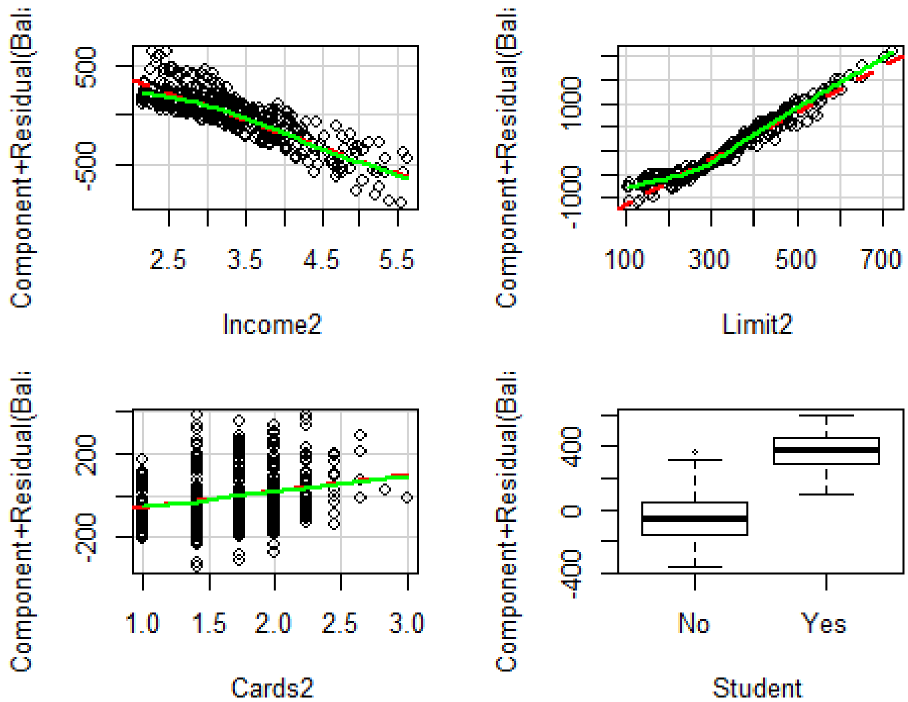
```r
FinalModel <- lm (Balance ~ Income2 + Limit2 + Cards2 + Student , data = r)
summary(FinalModel)
```

```
##
## Call:
## lm(formula = Balance ~ Income2 + Limit2 + Cards2 + Student, data = r)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -323.32 -108.41  -15.66   92.74  403.80
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -380.28588   41.58973  -9.144  < 2e-16 ***
## Income2     -268.40244   12.77157 -21.016  < 2e-16 ***
## Limit2         4.84357    0.08793  55.087  < 2e-16 ***
## Cards2        77.38931   16.96134   4.563 6.75e-06 ***
## StudentYes   421.31065   22.45242  18.765  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 134.6 on 395 degrees of freedom
## Multiple R-squared:  0.9151, Adjusted R-squared:  0.9143
## F-statistic:  1065 on 4 and 395 DF,  p-value: < 2.2e-16
```

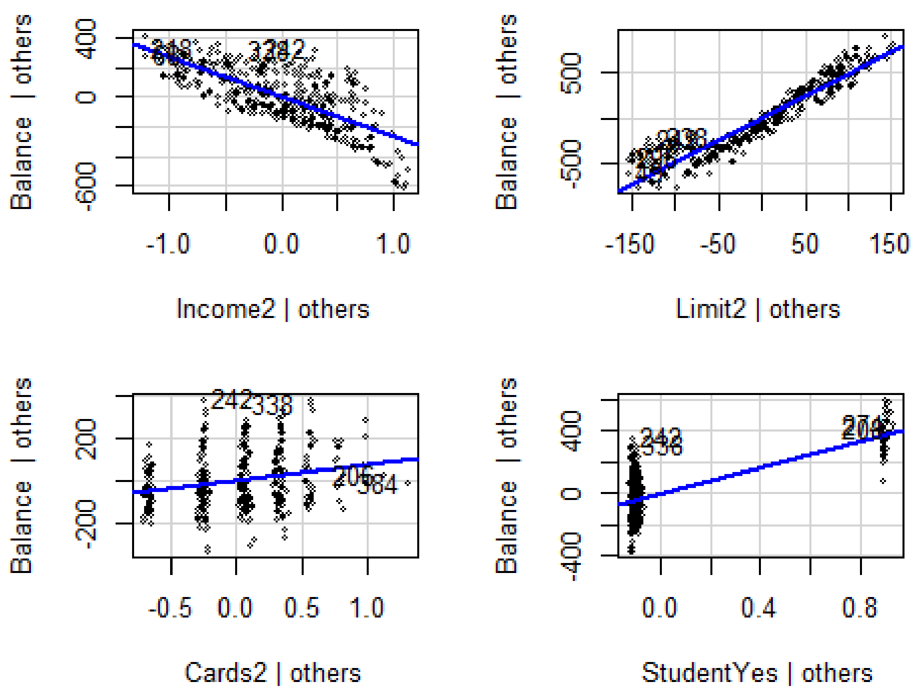
```r
Residuals <- rstandard(FinalModel)      ## Standardized Residuals
par(mfrow=c(3,2))
plot(FinalModel,c(1,2,3,4,5,6))
```




```
## CR Plot for Final Model

crPlots(FinalModel,main="",cex=0.5, id.n = 3, id.cex = 0.8,
col.lines=c("red","green"))
```




```
avPlots(FinalModel,cex=0.5,id.n = 3,id.cex = 0.8,main="")
```



```r
## VIF for Final Model
vif(FinalModel)

##  Income2   Limit2   Cards2  Student
## 2.231393 2.228436 1.003026 1.001482

## Bootstrapping

set.seed(123)
Model <- lm(Balance ~ Income2 + Limit2 + Cards2 + Student, data= r) # Best
predictors


Residuals <- residuals(Model) # Residuals from observed data
Predicted <- fitted(Model) # Predicted values from observed data

nb <- 4000 # Number of Bootstrapping samples
coefmat <- matrix(0,nb,5)
```

Response: Violation exists in the model even after transformation of variables. Some of the violations are: normality and constant variance are not existant. The VIF of coefficients still indicate that they are tolerable and stable, but is still not appropriate.

```r
set.seed(533)
coefmat <- matrix(0,nb,5)
for(i in 1:nb) # Repeat the process with a loop
{
boot_y <- Predicted + sample(Residuals, rep=TRUE) # generated predicted value
bMod <- update(Model, boot_y ~ .) # fitting model with bootstrap data
coefmat[i,] <- coef(bMod) # Store estimates through a loop
}
colnames(coefmat) <- c("Intercept","Income2","Limit2","Cards2","Student")
coefmat <- data.frame(coefmat)
cbind(t(apply(coefmat,2,function(x)
quantile(x,c(0.025,0.975)))),confint(Model))

##                  2.5%       97.5%       2.5 %      97.5 %
## Intercept -462.856970 -300.829522 -462.050784 -298.520969
## Income2   -292.180388 -243.300156 -293.511194 -243.293687
## Limit2       4.671733    5.010254    4.670709    5.016428
## Cards2      44.538053  110.890472   44.043514  110.735100
## Student    378.000832  465.958762  377.169465  465.451837
```

Response:

From the bootstrap estimation of confidence interval.

95% confidence that effect of 1 unit change income2 will be have an effect lying in the interval (-292.18,-243.30) on Balance2.

95% confidence that effect of 1 unit change Limit2 will be have an effect lying in the interval (4.67,5.01) on Balance2.

95% confidence that effect of 1 unit change cards2 will be have an effect lying in the interval (44.54,110.89) on Balance2.

95% confidence that effect of 1 unit change student2 will be have an effect lying in the interval (378.00,466.69) on Balance2.