Customer Segmentation at MiniMall

MSBA 310 – Applied Statistical Analysis

Prepared for:

Dr. Imad Bou-Hamad

Prepared by:

Samer Haidar

December 9, 2020

Abstract

In light of the economic crisis that is striking Lebanon accompanied by the COVID-19 pandemic, Lebanese businesses are struggling to maintain their customer base and their competitive positions due to the decreased purchasing power of consumers and the Lebanese Lira devaluation. In this study, we suggest to MiniMall, a Lebanese supermarket, a method to better understand their customers by classifying them into light and heavy ones. Classifying customers would in return lead to more efficient advertising and marketing plans tailored to those different segments. We developed three predictive models, two of which are logistic classification models, and the third is a tree classification model. Our target variable in those models is the customer label (light or heavy). The target variable is predicted depending on eight independent variables (Number of products bought, number of visits, etc.). Accordingly, we tested our models and chose the optimal one. Additionally, we developed a product recommendation system that recommends products to customers based on an algorithm that detects similarities in purchasing behavior between customers.

Keywords: supermarket, correlation, heavy customer, light customer predictors, logistic classifier, tree classifier, customer-based collaborative filtering, recommendation system

Introduction and Literature Review

Nowadays, we live in a globalized world where competition between businesses is at its highest; therefore, any business that wants to survive in this environment is bound to seek a competitive advantage. This is especially essential in those days in the light of the COVID-19 pandemic which has caused a major economic shock. Looking closer at the country we live in, Lebanon, the financial, economic, and political crises are increasing the burden on businesses. In addition to the COVID-19 pandemic, Lebanese businesses are suffering from inflation, increased operating costs, and most importantly a decreasing consumer purchasing power. Having said that, our aim in this study is to identify possible business insights to MiniMall, a Lebanese supermarket located in Nabatieh, to assist them in maintaining a competitive edge in their corresponding industry. MiniMall is a three-story hypermarket with a total area of 1,000 m^2. The supermarket offers a wide variety of high-quality food products as well as household and other convenience products. It is visited daily by hundreds of customers who come to shop from a wide array of products for their everyday needs. MiniMall like any other supermarket has different types of customers, so it's essential for them to segment their customers to better initiate targeted marketing and advertising campaigns tailored to different customer segments. Besides, given that data is easily attainable through the POS system, MiniMall should also leverage the data at-hand to develop a product recommendation system to recommend products to customers based on similarities in their shopping behavior.

Data analysis and modeling can help markets in assessing the effectiveness of their range to keep the targeted customers satisfied. According to the article "The Role of Big Data and Predictive Analytics in Retailing" (Bradlow, 2017), markets are able to track customers and link transactions. Loyalty programs are considered the most effective way of tracking customers where the ability to manage customer data is important to predict customer behavior. In addition to that, information about products will help in producing a product information matrix targeting customer product similarities and thereby recommending products based on these similarities. Therefore, data retrieved on customers and products purchased can be well addressed to classify customers. The article "Classifying and Understanding Prospective Customers via Heterogeneity of Supermarket Stores" (Tanaka, 2017), assisted us in understanding customers and classifying the high quality of them for us to utilize their experience and offer loyalty cards. In this research, the method RFM is used where R stands for Recency that denotes coming to store, F for frequency which shows the frequency of visits, and M for Monetary which represents total amount purchased. Good customers are classified using logistic regression. An advantage of logistic regression includes its ability to quantitatively understand how much the input of the explanatory variable to the model contributes to the targeted variable. The explanatory variables are the indicator of the RFM where a model is constructed for all customers inputting the explanatory variables. The evaluation of the model is then carried through accuracy using the F-value and precision. The RFM index identifies the good customers. The research was able to present the benefits of the logistic method in identifying the customers and products that generate the most store sales. On the other hand, the research paper on "Large Scale Product Recommendation of Supermarket Ware Based on Customer Behavior Analysis" (Andreas Kanavos, 2018) proposes another method that aims on targeting customer's classification based on their behavior and recommending new products that they are more likely to purchase. The research uses Map Reduce Programming Environment to process the dataset as well as Spark/Hadoop. Given the supermarket dataset, they aim on predicting whether a customer will purchase an item or not. The data consists of eight fields Customer Id, Product Category, ProductId, shop, Number of items, distance from each supermarket, and price. The implementation process involves data cleaning and categorizing customers into three categories (A, B, C) that correspond to the average money paid regularly. Customer behavior is extracted by analyzing the prediction model and the information on customers' behavior. The total amount purchased by customers is then categorized and 1-FoldCross Validation is used to evaluate the training at test data. The chosen classifiers are evaluated using True positive rate, False Positive rate well as F-measure (Kanavos, 2018). These research papers provide us with a foundation of how to proceed and implement the statistical analysis on a supermarket's transactions to be able to classify customers and recommend products.

Problem Description

It is well known that marketing and advertising activities are crucial for the exposure and profitability of retail stores. For this purpose, our objective in this study is to classify customers according to their long-term importance to the profitability of MiniMall. We aim to classify customers as light and heavy ones in order for the supermarket to be able to initiate more efficient advertising and marketing campaigns in an attempt to maintain existing customers, acquire new ones, and to increase their share from their customers' wallets. We will test different models which are built using several predictive techniques such as Logistic classifier and Classification Trees (CART), and choose the model that best fits our data, and that possesses the highest predictive accuracy. In this study, we will also build a product recommendation system by developing a user-user similarity matrix. The purpose of this system is to detect similarities in consumers' purchasing behaviors and recommend products to customers accordingly.

Data Description

We got from MiniMall supermarket the transactions done in September and October 2019. The data consists of each item sale done. We manipulated and cleaned these datasets until we reached the "Customer Data" dataset which is used for classifying customers whether they're considered to be heavy customers or light ones. To clean from outliers, we eliminated all data points where the total spent was > 2,000,000LL (33 data points) which were restaurants, cafés, and humanitarian organizations. As shown in Fig. 1, we noticed a decrease in sales in days 45-60 (Oct 17-Oct 29) due to the protests during the Lebanese Revolution.

After that, to classify the customer between being a heavy or light customer, we took into consideration the 75th percentile of the total spent (353,541LL) and frequency of visits (6 visits) (check Table 1). Accordingly, if a customer exceeded the previously mentioned percentiles, he would be classified as a heavy customer with Label = 1, else they would be classified as a light customer with Label = 0. As shown in Table 2, the 1788 heavy customers contributed with 747,938,651LL to the supermarket while the 1431 light customers contributed to 83,196,914LL (check Tables C1, C2 in Appendix C).

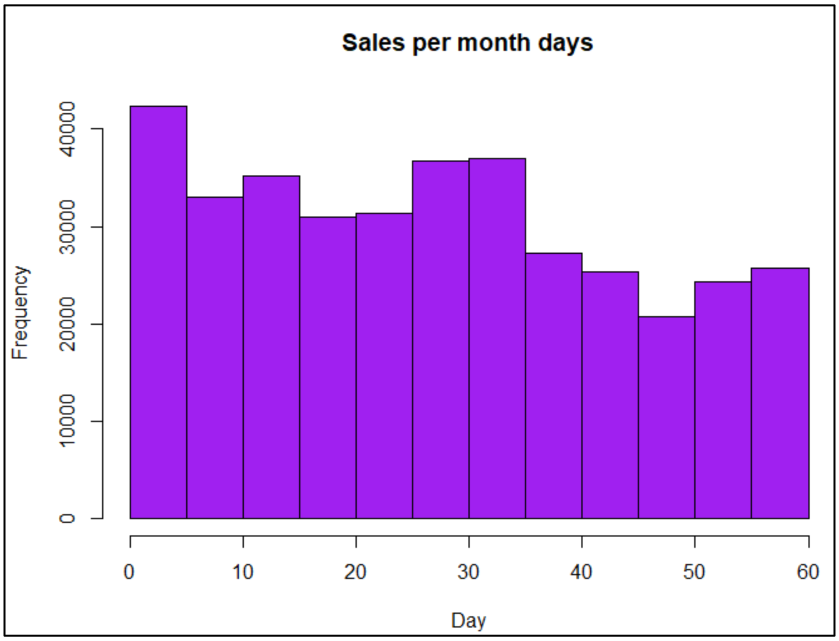

Fig. 1. Histogram for the frequency of daily sales in September and October 2019

Table 1. *Quartiles of Total Spent and Frequency Variables*

	Minimum	Q1	Median	Mean	Q3	Max
Total Spent	500	59,169	159,280	258,197	353,541	1,945,533
Frequency	1	1	3	4.725	6	52

Table 2. *Statistics for the Total Spent Variable to each Cluster*

Customer	Sum	Min	Max	Mean
Light	83,196,914	500	166,564	58,139
Heavy	747,938,651	113,626	1,945,533	418,310

Next, we manipulated and cleaned the September data set until we reach the "September Customer Data" dataset. To examine the relationship between the independent quantitative variables, we created a correlation heat map to check the correlation between the different independent variables, check Fig. 2. It can be noticed that the Total Amount Spent and the Number of Products Bought are highly correlated (correlation = 0.89). In addition, the Average Amount Spent and the Minimum, and Maximum Amounts Spent also exhibit some form of correlation (correlation = 0.91 and 0.86 respectively). Other than that, all other independent variables didn't seem to show a high correlation.

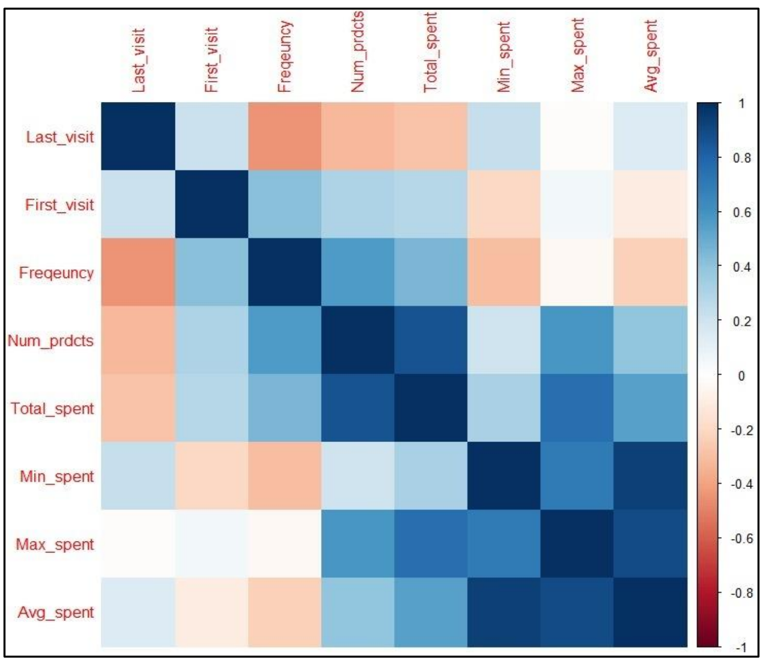

Fig. 2. Correlation heat map of quantitative variables

In addition to that, to examine the relationship, if there is any association, between some of our independent qualitative variables (Average spent, minimum spent, frequency of visits, and the number of products bought), and our dependent categorical variable (Label), we created side-by-side boxplots that you can see in Fig. 3. It is obvious that there is a significant association between the category of the customers (Light Customers represented by '0' and Heavy Customers represented by '1') and each of the other independent variables.

Next, to prepare for our models, we added the customer labels (1 if the customer is Heavy and 0 for light customers) "September Customer Data" to finally have "September Customer Data with Labels" dataset which will be used to train and test our model on. (check Fig B2 in Appendix B for the flow of datasets and dataset's details). To perform our analysis, we split the "Customer September Data with Labels" dataset into training and validation data, 70% and 30% respectively. Our main aim is to try to suggest ways to classify customer labels (heavy or light) beforehand and suggest ways to deal with heavy customers. That's why, we tried several classifiers: logistic classifier and classification tree, to find the optimal one that predicts customer classification. These models will take into consideration: customer's average spending, frequency, number of products bought, the minimum amount spent, the maximum amount spent, the day of the first visit, the day of the last visit, and whether the customer benefits from a discount or no.

Fig. 3. Boxplots of Customer Segment with respect to Average Spent, Number of Visits, Number of Products Bought and Maximum Spent

Results and discussion

After checking the two models, we found that the logistic classifier had a high multi-collinearity for some of its variables (check Table 3), so we added another logistic classifier excluding the average spent variable since as discussed previously and shown in Fig 2, it is highly correlated with maximum spent and minimum spent. Concerning the classification tree, check Table D1 in Appendix D where we chose the tree (check Fig. D4 and Table 4 in Appendix D) with 5 nodes and an x-error = 0.36339. Next, after comparing the three models (check Table 4 and Fig 4), even though the first logistic classifier had a better accuracy rate, we picked the second logistic classifier since the first had a variable, the average spent, in it causes multi-collinearity which might lead to an inaccurate model (check Table 3 for VIFs).

Table 3. *VIFs for variables in both logistic classifiers*

Model	Number of Products	Last Visit	First Visit	Average Spent	Discount	Min Spent	Max Spent	Frequency
Logistic Classifier	1.863	2.545	2.357	206.765	1.186	82.196	40.392	3.329
Modified Logistic Classifier	1.832	2.512	2.321	-	1.174	5.282	4.595	2.989

Table 4. *Model performance evaluation for the three models*

Model	Sensitivity	Specificity	FPPV	FNPV	Accuracy Rate	Error Rate
Logistic Regression	88.5	81.5	12.7	16.7	85.65	14.35
Modified Logistic Regression	88.11	81.81	12.6	17.17	85.52	14.48
Regression Tree	89	76.06	14.43	21.21	83.64	17.23

$$\begin{aligned}\text{Label} = {} & -3.78 + 1.71 \times 10^{-2}(\text{Number of Products}) - 3.7 \times 10^{-2}(\text{Last Visit}) \\ & + 2.34 \times 10^{-2}(\text{First Visit}) - 2.15(\text{Discount}) \\ & - 1.068 \times 10^{-5}(\text{Minimum Spent}) + 4.55 \times 10^{-5}(\text{Maximum Spent}) \\ & + 0.34(\text{Frequency})\end{aligned}$$

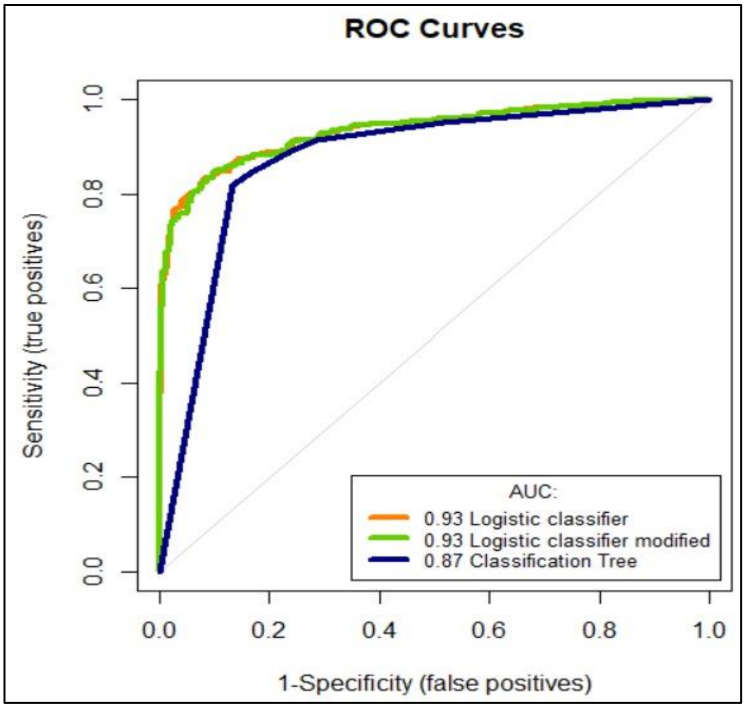

Fig. 4. ROC curves of all three models

With the number of products, Last Visit, Discount, Minimum Spent, Maximum Spent, Frequency being the significant predictors (check Table 5). Analyzing these significant variables, as shown in Table 5, the number of products, first visit, maximum spent, and frequency increase the chances of having a heavy customer. On the other hand, last visit, discount, and minimum spent variables decrease the chance of having a heavy customer. Digging deeper, with every increase in days of last visits, i.e. the longer it takes for a customer to re-visit, the percentage of them being a heavy customer decreases by 3.6%. On the other hand, increasing the number of products by one, with all other coefficients being constant, would increase the odds of having a heavy customer by 2%, so for example, we the number of products increased by 20, the odds of having a heavy customer would increase the odds of having a heavy customer by 40%. Concerning frequency, with every extra visit increase, the odds of having a heavy customer increases by 40%. That's why our recommendation will be based on increasing the number of products bought and the frequency of visits. This model is 85.52% accurate. Besides, the area under the ROC curve (check Fig. 4) = 0.93 which is considered to be good.

Table 5. *Model's Coefficients Statistics*

Variables	Estimate	Odds	Std. Error	z-value	Pr(>|z|)
Y-intercept	-3.784	0.023	2.702×10^{-1}	-14.006	$< 2 \times 10^{-16}$
Number of Products	1.705×10^{-2}	1.017	6.088×10^{-3}	2.800	0.00511
Last Visit	-3.695×10^{-2}	0.964	1.417×10^{-2}	-2.608	0.00911s
First Visit	2.339×10^{-2}	1.0234	1.477×10^{-2}	1.616	0.10609
Discount	-2.461	0.0853	7.125×10^{-1}	-3.012	0.00259
Minimum Spent	-1.068×10^{-5}	0.999	4.309×10^{-6}	-2.477	0.01323
Maximum Spent	4.554×10^{-5}	1.00004	4.526×10^{-6}	10.060	$< 2 \times 10^{-16}$
Frequency	3.435×10^{-1}	1.405	6.790×10^{-2}	5.059	4.21×10^{-7}

In addition to that, using Table 4 we calculated:

- Error rate = 14.48%. Approximately 14% of the customers were wrongly classified.
- Sensitivity = 88.11%. Approximately 88% of heavy customers are correctly classified.
- Specificity = 81.81%. Approximately 82% of light customers are correctly classified.
- False-positive predictive value (FPPV) = 12.6%. Approximately 13% of customers were predicted to be heavy but they are actually light customers.
- False Negative predictive value (FNPV) =17.17%. Approximately 17% of the customers were predicted to be light customers but they are actually heavy customers.

For the customers classified as heavy, we used the customer-based collaborative filtering. To do that, we first started by constructing the customer-product similarity binary matrix where each row is a unique customer, and each column is a unique product. A value of 1 in the matrix at the location (i,j) indicates that customer (i) purchased product (j), whereas a 0 indicates that no purchase was made. Next, to derive the customer-customer similarity matrix, we will compute distances between customers based on the cosine similarity measure which is the cosine of the angle between two vectors (customers), i.e. cosine of the transpose of the customer-product matrix. Finally, we recommend products. We took as an example customer of ID=23. As shown in Table 6, he has similar purchasing behavior as customers with ID 741, which has the highest cosine value in row 23, 0.4264. We then suggest to the customer of ID 23 items bought by the customer of ID 741 that weren't bought by him (check Fig. 5). To get these recommendations, we used the recommender function which we developed on R (check Appendix A).

Table 6. *Maximum Cosines for Customers Similar to Customer of ID 23*

CustomerID	23	741	70	760	2531	1961
23	1	0.426	0.329	0.301	0.273	0.213

```
> recommender(23,741)
 [1] "WATER GALLON 10 Liter"                     "PEPSI DIET 500ML"
 [3] "YOUNIS SMALL WHITE BREAD"                  "COW MEAT"
 [5] "HERBAL ESSENCES BODY ENVY 700 ML -10%"     "WOODEN BAKERY BURGER BUN*6 450G"
 [7] "LESIEUR MAYO 710-30%"                      "CHEDDAR SAUCE"
 [9] "AMERICANA CHICKEN ZINGER"                  "HAAGEN DAZS VANILLA&CREAM 500ML"
[11] "PRESIDENT CHEESE 8PCS"                     "PEPSI 1.25L DIET"
[13] "YOUNIS BIG WHITE BREAD"                    "HARVEST CHICK PEAS 1KG"
[15] "ABU KASS 2KG"                              "CALIFORNI GARDEN TUNA OIL185G*3
[17] "SAJ ADSHIT MARKOUK"                        "KLIM MILK 750G"
[19] "CHTOURA FOUL PLASTENIAN 400G"              "PANZANI SP 500GR 2+1 FREE"
[21] "XTRA KETCHUP 340G"                         "TAGAZIEH BEEF 340G"
[23] "BONELESS CHICKEN BREAST"                   "CHICKEN WINGS"
[25] "CHICKEN DABBOUS"                           "BONJUS CREAMOLAIT 400GR"
```

Fig. 5. *List of suggested items to the customer of ID 23*

Conclusion and recommendation

Given the bad economic situation in Lebanon, businesses have to adapt and find ways to stay up and running. One of the ways to do that is to focus on the already heavy customer and push those on the edge to become heavy customers. As we have already established, the main variables that affect the customer classification, whether they are heavy or light, are the number of products purchased, and the frequency of visits. By recommending products to customers using specialized notification for customers with similar purchasing behaviors, we open the door to selling more products and consequently more visits to the store. For future work and research, we could also work on the Product-Product similarity matrix and place products that possess similar characteristics next to each other on the supermarket shelves. Thus, we would induce more impulse purchases by customers. In addition, to that, this will increase convenience for customers who will find the products that they usually buy together next to other on shelves.

References

Kanavos, A., Iakovou, S. A., Sioutas, S., & Tampakas, V. (2018, May 9). *Large Scale Product Recommendation of Supermarket Ware Based on Customer Behaviour Analysis*. https://www.mdpi.com/2504-2289/2/2/11.

Bradlow, E. T., Gangwar, M., Kopalle, P., & Voleti, S. (2017, March 20). *The Role of Big Data and Predictive Analytics in Retailing*. https://www.sciencedirect.com/science/article/abs/pii/S0022435916300835.

Kanavos, A., Iakovou, S. A., Sioutas, S., & Tampakas, V. (2018, May 9). *Large Scale Product Recommendation of Supermarket Ware Based on Customer Behaviour Analysis*. https://www.mdpi.com/2504-2289/2/2/11.

Appendix A

R code

```r
#### Libraries ####
library("readr")
library("dplyr")
library("corrplot")
library("car")
library("pROC")
library("PresenceAbsence")
library("coop")
library("stringr")
library("rpart")
library("rpart.plot")
library("patchwork")
library("ggplot2")

#####SEPTEMBER DATA####
#reading data
sept <- read.csv("RptStockMovAmount2.csv")
dim(sept)
colnames(sept)
sept <- sept[-c(2,4,7,9,10,11,13,14,16,20,22,25,27,30,31)]

#changing column names
for (i in 1:length(sept[9,])){
  colnames(sept)[i] <- as.character(sept[9,i])
}
colnames(sept)

#Dealing with empty cells
str(sept)
sept$Type <- as.character(sept$Type)
unique(sept$Type)
sept <- subset(sept,(Type=='POS')) #since all "clean transactions"  have POS as type
dim(sept)

#removing unneccesarry columns
sept <- sept[-c(3,6,9,14,15,16)]
sept$Month <- "September"
head(sept,1)
colnames(sept)[3] <- "Customer"
colnames(sept)
dim(sept)
```

```r
#Changing column types
sept$Date <-as.numeric(gsub('/','',str_sub(as.character(sept$Date),1,2)))
sept$Total <- as.numeric(gsub(',','',as.character(sept$Total)))
sept$Customer <- as.character(sept$Customer)
sept$Item <- as.character(sept$Item)
sept$Qty. <- as.numeric(gsub(',','',as.character(sept$Qty.)))
sept$C.Qty <- as.numeric(gsub(',','',as.character(sept$C.Qty)))
sept$U.Price <- as.numeric(gsub(',','',as.character(sept$U.Price)))
sept$T.Price <- as.numeric(gsub(',','',as.character(sept$T.Price)))
sept$Discount <- as.numeric(gsub(',','',as.character(sept$Discount)))
sept$T.Nbr. <- as.numeric(gsub(',','',as.character(sept$T.Nbr.)))

hist(sept$Date,xlab="Day",main="Month Days", col='blue')

#####OCTOBER DATA####
#cleaning data
oct <- read.csv("RptStockMovAmount.csv")
dim(oct)
head(oct)
oct <- na.omit(oct)
dim(oct)
oct <- oct[,-3]
str(oct)
colnames(oct)[3]<-"Customer"
colnames(oct)

#fixing column type
oct$Date <-as.numeric(gsub('/','',str_sub(as.character(oct$Date),1,2)))
oct$Total <- as.numeric(gsub(',','',as.character(oct$Total)))
oct$Customer <- as.character(oct$Customer)
oct$Item <- as.character(oct$Item)
oct$Qty. <- as.numeric(gsub(',','',as.character(oct$Qty.)))
oct$C.Qty <- as.numeric(gsub(',','',as.character(oct$C.Qty)))
oct$U.Price <- as.numeric(gsub(',','',as.character(oct$U.Price)))
oct$T.Price <- as.numeric(gsub(',','',as.character(oct$T.Price)))
oct$Discount <- as.numeric(gsub(',','',as.character(oct$Discount)))
oct$Month <- "October"
oct$Date <- oct$Date +30 # to take into consideration the september 30 days
summary(oct$Date)

hist(oct$Date,xlab="Day",main="Month Days", col='blue')
```

```r
#####JOINING SEPT AND OCT####
str(sept)
str(oct)
colnames(sept)
colnames(oct)
data<- bind_rows(sept,oct)
data <- data[,-6]
dim(data)
colnames(data)

hist(data$Date,xlab="Day",main="Sales per month days", col='blue')

#####CREATING CUSTOMER DATA FOR BOTH MONTHS####

convert_to_customer_data <- function(data){
  #days last visit
  recency <- aggregate(Date~Customer, data = data, max)
  recency[,2]<- max(data[,1])-recency[,2]

  #first visit
  first_visit <- aggregate(Date~Customer, data = data, min)
  first_visit[,2]<- max(data[,1])-first_visit[,2]

  #number of times the customer came per month
  frequency <- aggregate(Date~Customer, data = data, unique)
  for ( i in 1:nrow(frequency)){
    frequency$freq[i] <- length(frequency$Date[[i]])
  }

  #If the customer benefits from a discount or not
  cust_discount <-aggregate(Discount~Customer, data = data, sum)
  cust_discount$discount <- ifelse(cust_discount[,2]>0,1,0)
  #Total spent per customer
  cust_totalpurch <-aggregate(Total~Customer, data = data, sum)
  #number of products bought by customer
  cust_number_of_items<- aggregate(Item~Customer, data = data, length)
  #Get the minimum and maximum amount amount
  df <- summarise(group_by(data,Date,Customer),total=sum(Total))
  min_amount <- aggregate(df$total~as.factor(df$Customer), data = df,min)
  max_amount <- aggregate(df$total~as.factor(df$Customer), data = df,max)

  #forming customer data dataframe
  customer_data <-data.frame(recency,first_visit[,2],frequency[,3],cust_discount[,3],cust_number_of_items[,2],cust_totalpurch[,2],min_amount[,2],
                             max_amount[,2])
  customer_data$Average_purchase <- customer_data$cust_totalpurch...2./customer_data$frequency...3.
  colnames(customer_data) <-
  c("CustomerID","Last_visit","First_visit","Freqeuncy","Discount","Number_of_products","Total_spent","Min_spent","Max_spent","Average_spent")
  return(customer_data)
}
```

```r
customer_data <- convert_to_customer_data(data)
colnames(customer_data)
head(customer_data)

#####CHECKING FOR OUTLIERS CUSTOMER DATA####

#CHECKING FOR OUTLIERS

#total spent
boxplot(customer_data$Total_spent, col = "red", main = "Boxplot of Total Amount Spent")
summary(customer_data$Total_spent)
dim(customer_data)
customer_data <- subset(customer_data, Total_spent < 2000000)
summary(customer_data$Total_spent)
dim(customer_data)
boxplot(customer_data$Total_spent, col = "red", main = "Boxplot of Total Amount Spent")

#Frequency
boxplot(customer_data$Freqeuncy,col = "red", main = "Boxplot of Frequency of Visits")
summary(customer_data$Freqeuncy)


# Creating the customer labels

for (i in 1:nrow(customer_data)){
  if(customer_data$Total_spent[i] >= 356141  || customer_data$Freqeuncy[i] >= 6){
    customer_data$Label[i] <- 1 #both
  }
  else {
    customer_data$Label[i] <- 0
  }
}

customer_labels <- data.frame(customer_data$CustomerID, customer_data$Label)
colnames(customer_labels) <- c("CustomerID", "Label")
head(customer_labels)
table(customer_labels[,2])
```

```r
#####CREATING CUSTOMER DATA FOR ONLY SEPTEMBER####
dim(sept)
colnames(sept)

customer_data_sept <- convert_to_customer_data(sept)
head(customer_data_sept)
dim(customer_data_sept)
colnames(customer_data_sept)


#####CHECKING FOR OUTLIERS IN CUSTOMER SEPTEMBER DATA####
#total spent
boxplot(customer_data_sept$Total_spent, col = "red", main = "Boxplot of Total Amount Spent")
summary(customer_data_sept$Total_spent)
dim(customer_data_sept)
customer_data_sept <- subset(customer_data_sept, Total_spent < 2000000)
summary(customer_data_sept$Total_spent)
dim(customer_data_sept)
boxplot(customer_data_sept$Total_spent, col = "red", main = "Boxplot of Total Amount Spent")

#Frequency
boxplot(customer_data_sept$Freqeuncy,col = "red", main = "Boxplot of Frequency of Visits")
summary(customer_data_sept$Freqeuncy)


#####LEFT JOINING CUSTOMER SEPTEMBER AND LABELS DATA####
data_sept_labels <-  merge(x = customer_data_sept, y = customer_labels, by = "CustomerID", all.x = TRUE)
dim(data_sept_labels)
head(data_sept_labels)
colnames(data_sept_labels)
data_sept_labels <- na.omit(data_sept_labels)
dim(data_sept_labels)

data_sept_labels <- data_sept_labels[,-1] #deleting customer ID model

#####Visualization####
cor_data = data.frame(data_sept_labels[,c(1,2,3,5,6,7,8,9)])
correlation = cor(cor_data)
corrplot(correlation, method = "color")

x1 = ggplot(data_labels,
            aes(x = Label,
                y = Average_spent)) +
  geom_boxplot(color = "darkorange4", fill = "darkorange2") +
  labs(title = "Average Spent by Customer Segment", x = "Customer Segment",
       y = "Average Spending")
```

```r
x2 = ggplot(data_labels,
            aes(x = Label,
                y = Freqeuncy)) +
  geom_boxplot(color = "mediumorchid4", fill = "mediumorchid2") +
  labs(title = "Number of Visits by Customer Segment", x = "Customer Segment",
       y = "Number of Visits")

x3 = ggplot(data_labels,
            aes(x = Label,
                y = Number_of_products)) +
  geom_boxplot(color = "seagreen4", fill = "seagreen2") +
  labs(title = "Number of Products Bought by Customer Segment", x = "Customer Segment",
       y = "Number of Products Bought")

x4 = ggplot(data_labels,
            aes(x = Label,
                y = Max_spent)) +
  geom_boxplot(color = "tomato4", fill = "tomato2") +
  labs(title = "Maximum Spent by Customer Segment", x = "Customer Segment",
       y = "Maximum Spent")

(x1 | x2) / (x3 | x4)
```

```r
#####SPLITTING DATA####
str(data_sept_labels)
data_sept_labels$Discount <- as.factor(data_sept_labels$Discount)
data_sept_labels$Label <- as.factor(data_sept_labels$Label)
set.seed(1000)
split <- sample(1:2, nrow(data_sept_labels), replace = TRUE, prob=c(0.7, 0.3))
training_data <- data_sept_labels[split==1, ]
validation_data <- data_sept_labels[split==2, ]


#####LOGISTIC MODEL#####

colnames(training_data)

lc <- glm(Label~Number_of_products+Last_visit+First_visit+Average_spent
          +Discount+Min_spent+Max_spent+Freqeuncy, data = training_data,
          family = "binomial")
summary(lc)

lc_pred <- predict(lc, validation_data,type = "response")

#Check for multi-collinearity

vif(lc)


lc2 <- glm(Label~Number_of_products+Last_visit+First_visit
           +Discount+Min_spent+Max_spent+Freqeuncy, data = training_data,
           family = "binomial")
summary(lc2)

lc2_pred <- predict(lc2, validation_data,type = "response")

vif(lc2)
```

```r
#####REGRESSION TREE####

class_tree <- rpart(Label ~ Last_visit + First_visit + Freqeuncy + Discount
                    + Number_of_products
                    + Min_spent + Max_spent + Average_spent, data = training_data,
                      control = rpart.control(cp = 0.0001))
printcp(class_tree)

rpart.plot(class_tree,type=4,extra=2,
           main="Customer Segmentation")

rpart(formula = Label ~ ., data = training_data, control = rpart.control(cp = 0.0001))
class_tree$cptable

bestcp=class_tree$cptable[which.min(class_tree$cptable[,"xerror"]),"CP"]

tree.pruned=prune(class_tree, cp = bestcp)

x11()
rpart.plot(tree.pruned,type=4,extra=2,
           main="Customer Segmentation")


tree_pred=predict(tree.pruned, newdata=validation_data,type="prob")

tree_pred=tree_pred[,2]
```

```r
#####Evaluating Models ####

acc_measures <- function(p) {
  act_pred=data.frame(ID=1:nrow(validation_data),1*(validation_data$Label== 1),p)
  # create data frame for actual and predicted but ID must be there
  conf_mat=cmx(act_pred)#  creates a confusion matrix
  total_acc= pcc(conf_mat) # Overall accuracy

  sens=sensitivity(conf_mat)# to obtain snesitivty

  spec=specificity(conf_mat)# to obtain specificity
  auc <- auc(act_pred)
  x=c(total_acc,sens,spec,auc)

  #names(vec)= c("Total Accuracy", "Sensitivity", "Specificity","AUC")
  return(x)
}


acc_measures(lc_pred)
acc_measures(lc2_pred)
acc_measures(tree_pred)

act_pred_mult=data.frame(ID=1:nrow(validation_data),1*(validation_data$Label== 1),lc_pred,lc2_pred,tree_pred)

x11()
auc.roc.plot(act_pred_mult,col=c("navyblue","darkorange1","green4"),line.type = 1, lwd = 2,
             threshold = 1001,  main="ROC Curves",legend.text=c("Logistic classifier",
                                                  "Logistic classifier modified", "Classification Tree"),

             )
```

```r
#####Customer Recommendations####

cid <- sort(customer_data$CustomerID)
head(cid)
length(cid)

item <- sort(unique(data$Item))
head(item)
length(item)

user_item <- data.frame(cid)
#code below takes few mins
for (j in 1:length(item)){
  spec_cid <- select(filter(data, Item == item[j] ), Customer)
  col1 <- rep(0, length(cid))
  for (i in 1:nrow(spec_cid)){
    if (spec_cid[i,] %in% cid) {
      col1[match(spec_cid[i,],cid)] = 1
    }
  }
  user_item <- data.frame(user_item,col1)
  colnames(user_item)[j+1] <- item[j]
}
dim(user_item)
```

```r
#Collaborative filtering
#User-based Collaborative Filtering
user_item <-user_item[,-1]# to delete id column
#User-User Similarity Matrix
cos_matrix <- cosine(t(user_item))
dim(cos_matrix) # 3218 x 3218 matrix ==> checks
cos_matrix <- data.frame(cos_matrix)
colnames(cos_matrix) <- 1:length(cid)

#heavy customer ids
xx <- customer_labels
xx$CustomerID <- 1:nrow(customer_labels)
heavy_customers_ids <- subset(xx, Label==1)$CustomerID
length(heavy_customers_ids)
head(heavy_customers_ids)
heavy_customers_ids[6]

#Making recommendations for Customer 23
sort(cos_matrix[23,], decreasing = TRUE)[1:6]

#customers if IDs:741, 70, 760 2531, 1961 are the most similar to customerID = 23

#suggestion products for customer of ID 23 with that of ID 741

recommender <- function(customer_id,similar_customer_id){
  items_of_customer <-unique(subset(data, Customer==sort(unique(data$Customer))[customer_id])$Item)
  potential_suggestions <- unique(subset(data,
                                 Customer==sort(unique(data$Customer))[similar_customer_id])$Item)

  recommedations <- c()
  for (i in 1:length(potential_suggestions)){
    if (!(potential_suggestions[i] %in% items_of_customer)) {
      recommedations <- c(recommedations, potential_suggestions[i])
    }
  }
  return(recommedations)
}
unique(subset(data, Customer==sort(unique(data$Customer))[23])$Item)
unique(subset(data, Customer==sort(unique(data$Customer))
              [741])$Item)
recommender(23,741)
```

Appendix B

Datasets

September data: Raw data of MiniMall customer transactions for September 2019 check Table B1 for details

October data: Raw data of MiniMall customer transactions for October 2019 check Table B1 for details

Data: All MiniMall customer transactions for September and October 2019 check Table B1 for details

Customer data: Data for customer information for September and October 2019, check Table B2 for details

Customer September data: Data for customer information for September 2019, check Table B2 for details

Labels: Customer labels (1 if heavy 0 if light) dataset check Table B3 for details

Customer September data with labels: Data for customer information for September 2019 with labels

Table B1: *September data, October data, and data datasets descriptions*

Column Name	**Description**
Date	Day of the month starting 1 (Sept 1) and ends in 59 (October 29)
T.Nbr.	Item barcode
CustomerID	Customer ID

Item	Item Name
Qty.	Quantity bought of the item
U.Price	Item's price
T.Price	Quantity*Price
Discount	Discount Amount
Total	Total price - discount
Month	Either September or October

Table B2: *Customer data, September customer data, datasets descriptions*

Column Name	**Description**
Customer ID	Customer ID
Last_Visit	Days since the customer last came to the supermarket
First_Visit	Days since the customer first came to the supermarket
Frequency	Number of times the customer came to the supermarket
Discount	1 if the customer benefits from a discount 0 otherwise

Number_of_Products	Total number of products bought by the customer
Total_Spent	Total amount spent by the customer
Min_Spent	Minimum amount spent in customer visits
Max_Spent	Maximum amount spent in customer visits
Average_Spent	The average amount spent for the customer in their visits

Table B3: *Label dataset description*

Column Name	**Description**
Customer ID	Customer ID
Label	1 if heavy, 0 otherwise

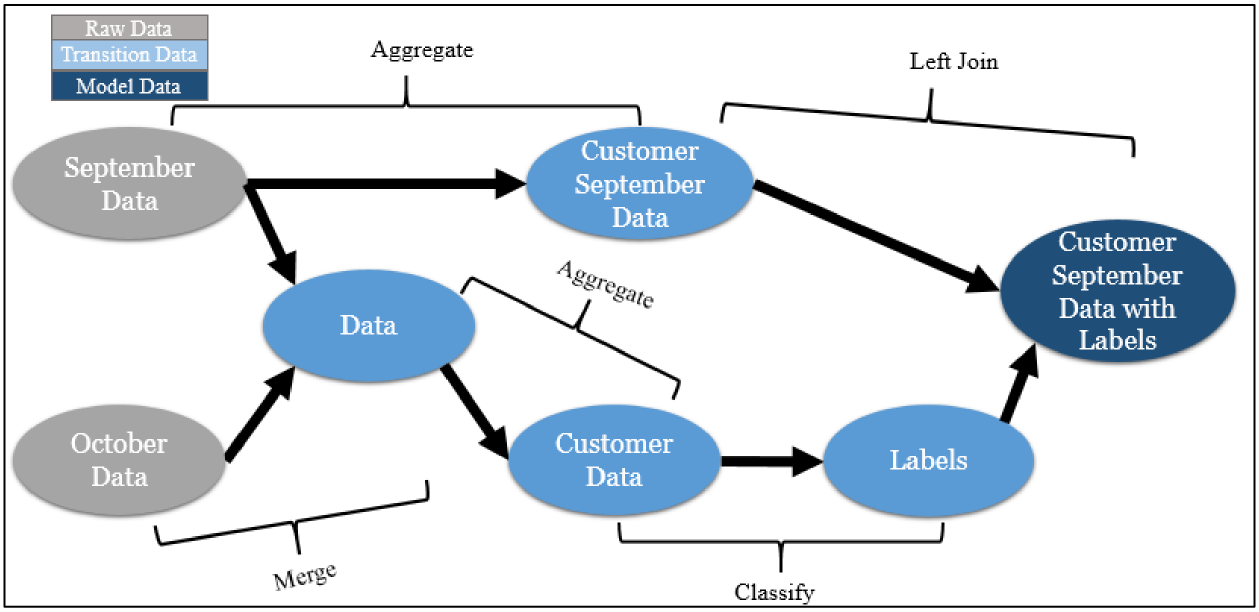

Fig. B1. *Data manipulation flow to reach customer September data with labels*

Appendix C

Customer Label Statistics

Table C1. *Statistics for the Average Spent Variable to each Cluster*

Customer	Min	Max	Mean
Light	500	166,563	37,586
Heavy	5,329	828,463	112,818

Table C2. *Statistics for the Frequency Variable to each Cluster*

Customer	Sum	Min	Max	Mean
Light	2,968	1	23	2.07
Heavy	12,241	1	52	6.84

Appendix D

Classification Tree

Table D1. *Statistics for classification tree*

CP	Number of splits	Rel error	X error	X stdv
0.57738896	0	1.00000	1.00000	0.028460
0.06191117	1	0.42261	0.43472	0.021996
0.01076716	2	0.36070	0.39569	0.021181
0.00583221	5	0.32840	0.36339	0.020453
0.00403769	8	0.31090	0.36878	0.020578
0.00302826	9	0.30686	0.37012	0.020609
0.00269179	13	0.29475	0.36339	0.020453
0.00235532	17	0.28264	0.36608	0.020516
0.00224316	23	0.26649	0.36878	0.020578
0.00168237	29	0.25303	0.36743	0.020547
0.00134590	31	0.24899	0.37281	0.020671
0.00100942	35	0.24226	0.37281	0.020671
0.00100942	40	0.23553	0.39973	0.021269
0.00089726	44	0.23149	0.40781	0.021442
0.00044863	47	0.22880	0.40646	0.021413
0.00010000	50	0.22746	0.41723	0.021639

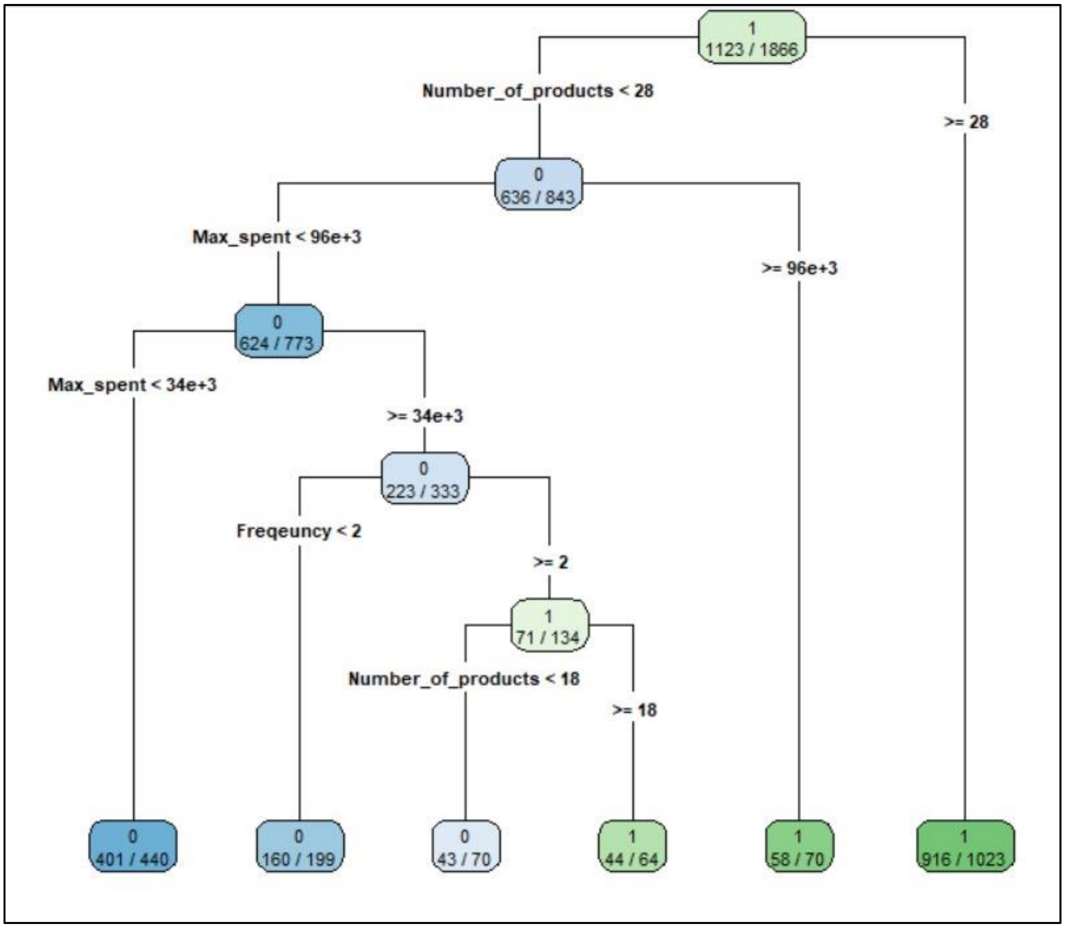

Fig. D1. Classification Tree