**Mini Project Report**

**WHOLESALE CLUSTERING ANALYSIS**

**Submitted by**

Yushant Tyagi T194062

**Under Supervision of**

Prof. Jayvant H. Devare

Miss. Shobha Mourya



**School of Computer Engineering and Technology**

MIT ACADEMY OF ENGINEERING

November 2019

**Acknowledgments**

I owe my deep gratitude to our project guide Jayvant Devare Sir and Shobha Ma’am who took keen interest on our project work and guided us all along, till the completion of our project work by providing all the necessary information for developing a good system. I would also like to thank all my classmates and teachers who helped us in putting together the entire project and motivated us to keep on improving and achieving the best of the results. This would not have been possible without your efforts.

November 2019 Yushant Tyagi T194062

**Contents**

**ACKNOWLEDGEMENTS** **1**

**CONTENTS** **2**

**1. INTRODUCTION** **3**

OVERVIEW 3

BACKGROUND AND MOTIVATION 3

OBJECTIVE 4

METHODOLOGY 4

**2. DATA SET** **5**

DESCRIPTION 6

NUMBER OF ATTRIBUTES USED 6

**3. SYSTEM WORKING** **10**

BLOCK DIAGRAM……………………………………………………………………………………………………….......**10**

SAMPLE CODE……………………………………………………………………………………………10

SCREENSHOT RESULT 6

**4. CONCLUSION** **12**

**5. FUTURE WORK**  **14**

**6. REFERENCES** **16**

1. **INTRODUCTION**

**OVERVIEW**

We know how difficult it is now a days to deal with data. But thanks to the new mining techniques which helps us to handle data very smoothly.

In this project, we have used the most famous and important technique of data mining which is clustering.

Clustering is something which helps us find trends in the data and draw conclusions from it. The project has been implemented in R and has used inbuilt R libraries which are used for plotting graphs.

Clustering has the following features:

* **Scalability** − We need highly scalable clustering algorithms to deal with large databases.
* **Ability to deal with different kinds of attributes** − Algorithms should be capable to be applied on any kind of data such as interval-based (numerical) data, categorical, and binary data.
* **Discovery of clusters with attribute shape** − The clustering algorithm should be capable of detecting clusters of arbitrary shape. They should not be bounded to only distance measures that tend to find spherical cluster of small sizes.
* **High dimensionality** − The clustering algorithm should not only be able to handle low-dimensional data but also the high dimensional space.

The data consists of the number of sales of different products of a retailer who sells those products in market.

**BACKGROUND AND MOTIVATION**

The motivation of this project is the curiosity we have for handling big data.

Bigger the data, better we can apply the available techniques to it to grab some useful information from it.

Some points that need to be kept in mind are:

* A cluster of data objects can be treated as one group.
* While doing cluster analysis, we first partition the set of data into groups based on data similarity and then assign the labels to the groups.
* The main advantage of clustering over classification is that, it is adaptable to changes and helps single out useful features that distinguish different groups.
* Making Regression models and predicting the data using the model.

Also, R has an [amazing variety](http://wiki.math.yorku.ca/index.php/R:_Cluster_analysis)of functions for [cluster analysis](http://cran.cnr.berkeley.edu/web/views/Cluster.html). In this section, I will describe three of the many approaches: hierarchical agglomerative, partitioning, and model based. While there are no best solutions for the problem of determining the number of clusters to extract, several approaches are given below.

**K-means**clustering is the most popular partitioning method. It requires the analyst to specify the number of clusters to extract. A plot of the within groups sum of squares by number of clusters extracted can help determine the appropriate number of clusters. The analyst looks for a bend in the plot similar to a screen test in factor analysis.

Multiple regression is an extension of linear regression into relationship between more than two variables. In simple linear relation we have one predictor and one response variable, but in multiple regression we have more than one predictor variable and one response variable.

We create the regression model using the **lm()** function in R. The model determines the value of the coefficients using the input data. Next, we can predict the value of the response variable for a given set of predictor variables using these coefficients

**OBJECTIVE**

The objective of this project is to grab some useful data in the form of trends. Trends are something which tells us the similarity in data and how can they be formed in groups based on their similarity.

We all know that, data analysis is a process of inspecting, [cleansing](https://en.wikipedia.org/wiki/Data_cleansing), [transforming](https://en.wikipedia.org/wiki/Data_transformation) and [modeling](https://en.wikipedia.org/wiki/Data_modeling) [data](https://en.wikipedia.org/wiki/Data) with the goal of discovering useful information, informing conclusion and supporting decision-making. Data analysis has multiple facets and approaches, encompassing diverse techniques under a variety of names, and is used in different business, science, and social science domains. In today's business world, data analysis plays a role in making decisions more scientific and helping businesses operate more effectively.

Mathematical formulas or models called [algorithms](https://en.wikipedia.org/wiki/Algorithms) may be applied to the data to identify relationships among the variables, such as [correlation](https://en.wikipedia.org/wiki/Correlation_and_dependence) or [causation](https://en.wikipedia.org/wiki/Causality). In general terms, models may be developed to evaluate a particular variable in the data based on other variable(s) in the data, with some residual error depending on model accuracy (i.e., Data = Model + Error)

The dataset, which consists of thousands of rows can give a lot of clusters. These clusters will then help us decide our next step in wholesale market. This will help the retailer to know what and how much to sell next.

Studying the Correlation between the various dimensions and making a multiple regression model. Predicting the values and obtaining the model with the highest possible accuracy rate. Showing these relations through various plots.

**METHODOLOGY**

The following steps were applied to achieve our goal. These steps can also be found in the form of comments in the code.

1. Prepare the data for analysis. Remove the missing value and remove “Channel” and “Region” columns because they are not useful for clustering.
2. Standardize the variables.
3. Determine the number of clusters.
4. The correct choice of k is often ambiguous, but from the above plot, I am going to try my cluster analysis with 6 clusters.
5. Fit the model and print out the cluster means.
6. Plotting the results.

### Outlier detection with K-Means

### the data are partitioned into k groups by assigning them to the closest cluster centres, as follows

### Then calculate the distance between each object and its cluster centre, then pick those with largest distances as outliers and print out outliers’ IDs.

1. These are the outliers. Let me make it more meaningful.
2. Making some conclusions by plotting variables together.
3. Prepare the training and test data set using split function
4. Use the training set to train the model which checks the Channel value dependent on the other dimensions
5. Predict the values using the generated model using the test set
6. Count the correctly matching predictions and calculate the accuracy rate if the model.
7. **DATASET EXPLORATION**

The dataset has been taken from <http://archive.ics.uci.edu/ml/machine-learning-databases/00292/>

This data has 7 attributes. Each attribute describes certain feature of the dataset.

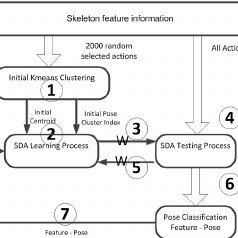
**Attribute Information:**

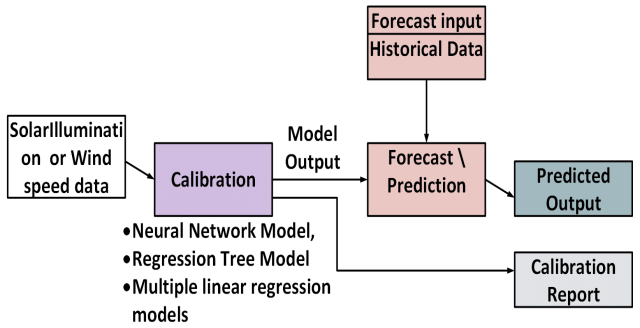
1) FRESH: annual spending (m.u.) on fresh products (Continuous);  
2) MILK: annual spending (m.u.) on milk products (Continuous);  
3) GROCERY: annual spending (m.u.)on grocery products (Continuous);  
4) FROZEN: annual spending (m.u.)on frozen products (Continuous)  
5) DETERGENTS\_PAPER: annual spending (m.u.) on detergents and paper products (Continuous)  
6) DELICATESSEN: annual spending (m.u.)on and delicatessen products (Continuous);  
7) CHANNEL: customersâ€™ Channel - Horeca (Hotel/Restaurant/CafÃ©) or Retail channel (Nominal)  
8) REGION: customersâ€™ Region â€“ Lisnon, Oporto or Other (Nominal)  
Descriptive Statistics:

(Minimum, Maximum, Mean, Std. Deviation)  
FRESH ( 3, 112151, 12000.30, 12647.329)  
MILK (55, 73498, 5796.27, 7380.377)  
GROCERY (3, 92780, 7951.28, 9503.163)  
FROZEN (25, 60869, 3071.93, 4854.673)  
DETERGENTS\_PAPER (3, 40827, 2881.49, 4767.854)  
DELICATESSEN (3, 47943, 1524.87, 2820.106)

REGION Frequency  
Lisbon 77  
Oporto 47  
Other Region 316  
Total 440  
  
CHANNEL Frequency  
Horeca 298  
Retail 142  
Total 440

**Block Diagram: -**





Data

**SYSTEM WORKING**

**CODE**

library(corrplot)

# Clustering

library(cluster)

library(ggplot2)

library(factoextra)

library(car)

library(dplyr)

customer <- read.csv('Wholesale.csv')

head(customer)

str(customer)

View(customer)

customer1<- customer

customer1<- na.omit(customer1)

customer1$Channel <- NULL

customer1$Region <- NULL

#Standardize the variables.

customer1 <- scale(customer1)

corrmatrix <- cor(customer)

corrplot(corrmatrix, method = 'number')

#Determine the number of clusters.

wss <- (nrow(customer1)-1)\*sum(apply(customer1,2,var))

for (i in 2:15) wss[i] <- sum(kmeans(customer1,

centers=i)$withinss)

plot(1:15, wss, type="b", xlab="Number of Clusters",

ylab="Within groups sum of squares")

#Fit the model and print out the cluster means.

fit <- kmeans(customer1, 6) # fit the model

aggregate(customer1,by=list(fit$cluster),FUN=mean) # get cluster means

customer1 <- data.frame(customer1, fit$cluster) #append cluster assignment

#Plotting the results.

library(cluster)

clusplot(customer1, fit$cluster , color=TRUE, shade=TRUE, labels=2, lines=0)

#Outlier detection with K-Means

customer2 <- customer[, 3:8]

kmeans.result <- kmeans(customer2, centers=6)

kmeans.result$centers

#Then calculate the distance between each object and its cluster center,

#then pick those with largest distances as outliers and print out outliers’ IDs.

kmeans.result$cluster # print out cluster IDs

centers <- kmeans.result$centers[kmeans.result$cluster, ]

distances <- sqrt(rowSums((customer2 - centers)^2)) # calculate distances

outliers <- order(distances, decreasing=T)[1:5] # pick up top 5 distances

print(outliers)

#These are the outliers. Let us make it more meaningful.

print(customer2[outliers,])

#Hierarchical Clustering

idx <- sample(1:dim(customer)[1], 40)

customerSample <- customer[idx,]

customerSample$Region <- NULL

customerSample$Channel <- NULL

#There are a wide range of hierarchical clustering methods,

#I heard Ward’s method is a good appraoch, so try it out.

d <- dist(customerSample, method = "euclidean") # distance matrix

fit <- hclust(d, method="ward.D")

plot(fit) # display dendogram

groups <- cutree(fit, k=6) # cut tree into 6 clusters

rect.hclust(fit, k=6, border="red") # draw dendogram with red borders around the 6 clusters

count\_fresh=sum(customer$Fresh)

count\_Milk=sum(customer$Milk)

count\_gro=sum(customer$Grocery)

count\_fro=sum(customer$Frozen)

count\_Det=sum(customer$Detergents\_Paper)

count\_del=sum(customer$Delicassen)

data=c(count\_fresh,count\_Milk,count\_gro,count\_fro,count\_Det,count\_del)

vals=c(1,2,3,4,5,6)

data\_vals=data.frame(data,vals)

ggplot(data\_vals,aes(x=vals,y=data))+geom\_line()

salesfor1 <- customer %>%

filter(Channel==1)

salesfor2 <- customer %>%

filter(Channel==2)

c1=count(salesfor1)

c2=count(salesfor2)

max(salesfor1)

max(salesfor2)

a=max(salesfor1$Fresh)

b=max(salesfor2$Fresh)

c=max(salesfor1$Milk)

d=max(salesfor2$Milk)

e=max(salesfor1$Detergents\_Paper)

f=max(salesfor2$Detergents\_Paper)

g=max(salesfor1$Delicassen)

h=max(salesfor2$Delicassen)

i=max(salesfor1$Frozen)

j=max(salesfor2$Frozen)

k=max(salesfor1$Grocery)

l=max(salesfor2$Grocery)

vals1= c(a,c,e,g,i,k)

vals2=c(b,d,f,h,j,l)

frames\_dat=c(vals1,vals2)

chanel=c(1,2)

customer <- customer %>%

mutate(sum\_sale=((Milk+Frozen+Fresh+Detergents\_Paper+Grocery+Delicassen)))

barplot(frames\_dat,col = c("Blue","Red"),beside = TRUE,legend.text = c("1","2"),main = "Channel Based Expenditure",xlab="Expenditure",ylab="Objects",las=1)

pairs(customer, col=customer$Channel)

plot(customer)

cor(customer)

scatterplot( Fresh ~ Frozen|Channel, data=customer,xlab="Fresh", ylab="Frozen",main="Scatter Plot: Horeca(Hotel/Restaurant/Cafe) vs. Retail Channel (Store)",smoother=T,lty=1, lwd=2, by.groups=T)

ggplot(customer, aes(x = Region , y = sum\_sale ,col=factor(Channel)) ) + geom\_point()

library(caTools)

#install.packages('caTools')

set.seed(123)

split=sample.split(customer$Channel, SplitRatio = 0.8)

training\_set=subset(customer,split==TRUE)

test\_set=subset(customer,split==FALSE)

prediction\_model <- lm(Channel~Milk+Frozen+Detergents\_Paper+Grocery+Fresh+Delicassen , data=training\_set)

plot(prediction\_model)

predicted <- predict(prediction\_model,newdata = test\_set,interval = "prediction")

final\_frame=data.frame(predicted,test\_set)

correct\_guess = correct\_guess %>%

mutate(avg\_predict\_channel=(lwr+upr)/2)

correct\_guess= final\_frame %>%

filter(((lwr+upr)/2 < 0.7& Channel==0)|((lwr+upr)/2 >0.7&Channel==1))

accuracy=count(correct\_guess)

total=count(test\_set)

accuracy\_rate=accuracy/total

print("Accurarcy Rate of The prediction model is =")

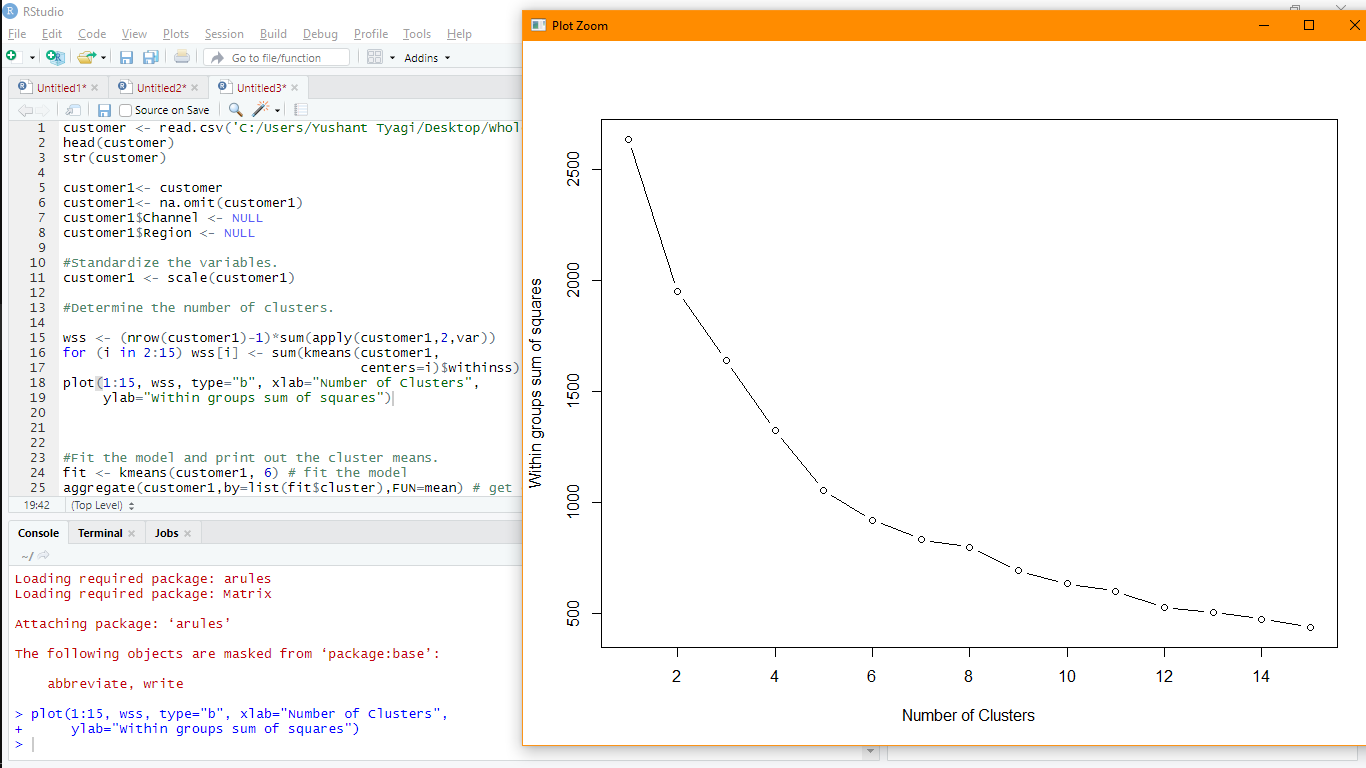
accuracy\_rate

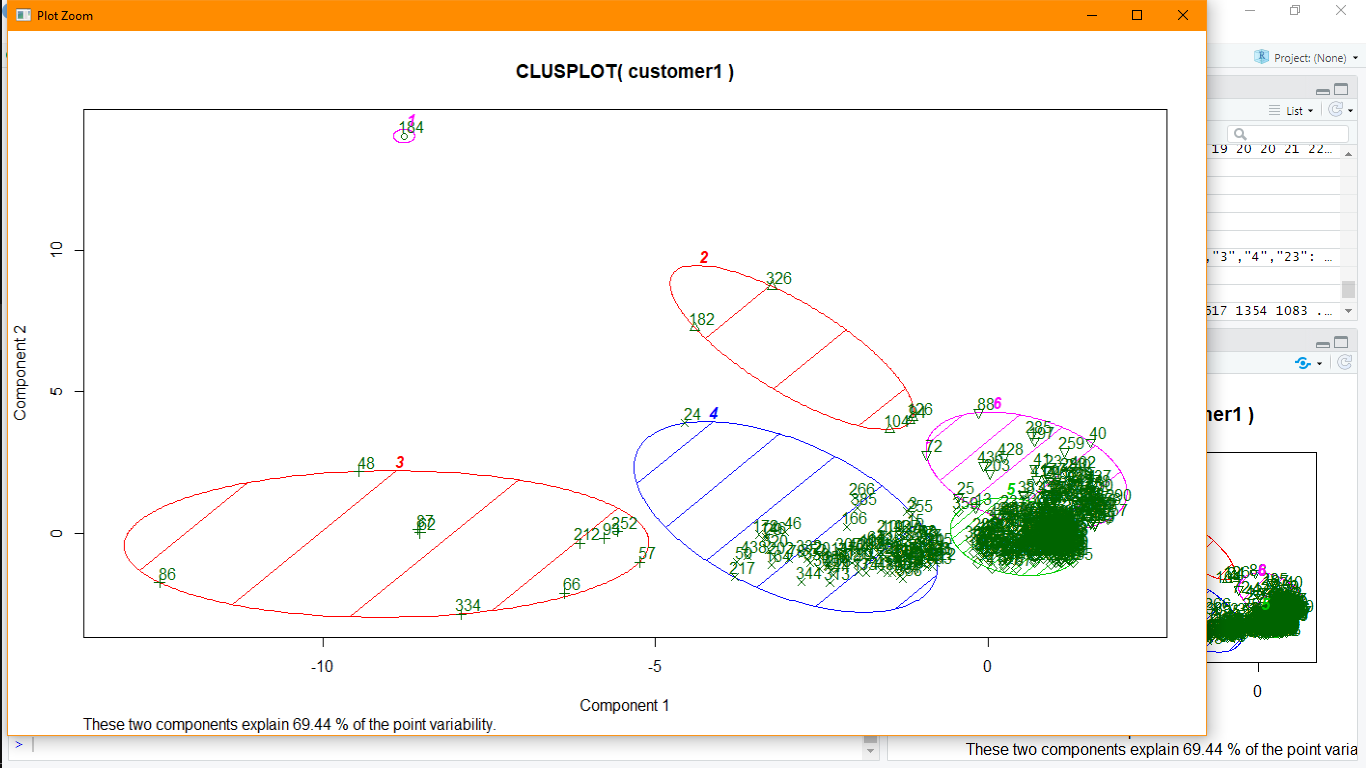
# the model is mostly successful in predicting the value of channel

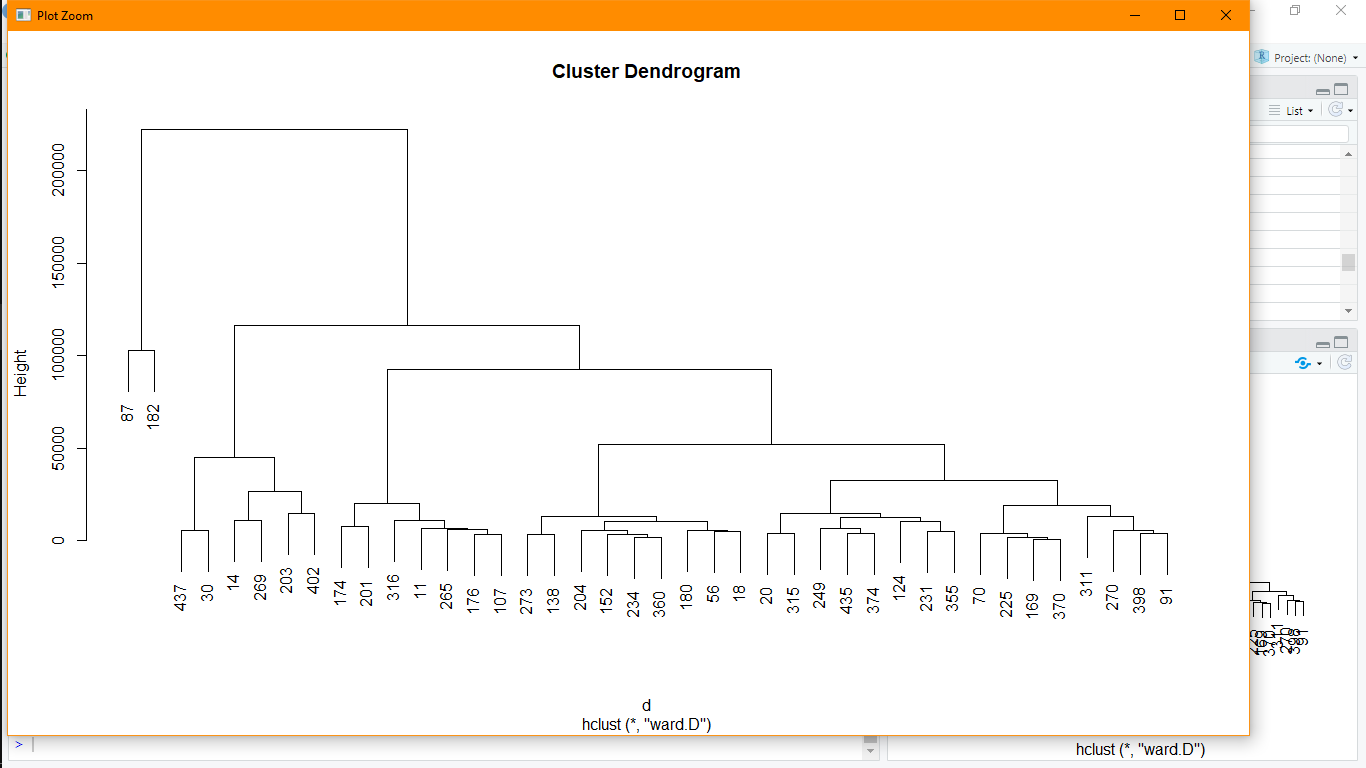
final\_model\_obtained=data.frame(correct\_guess$avg\_predict\_channel,correct\_guess$Channel)

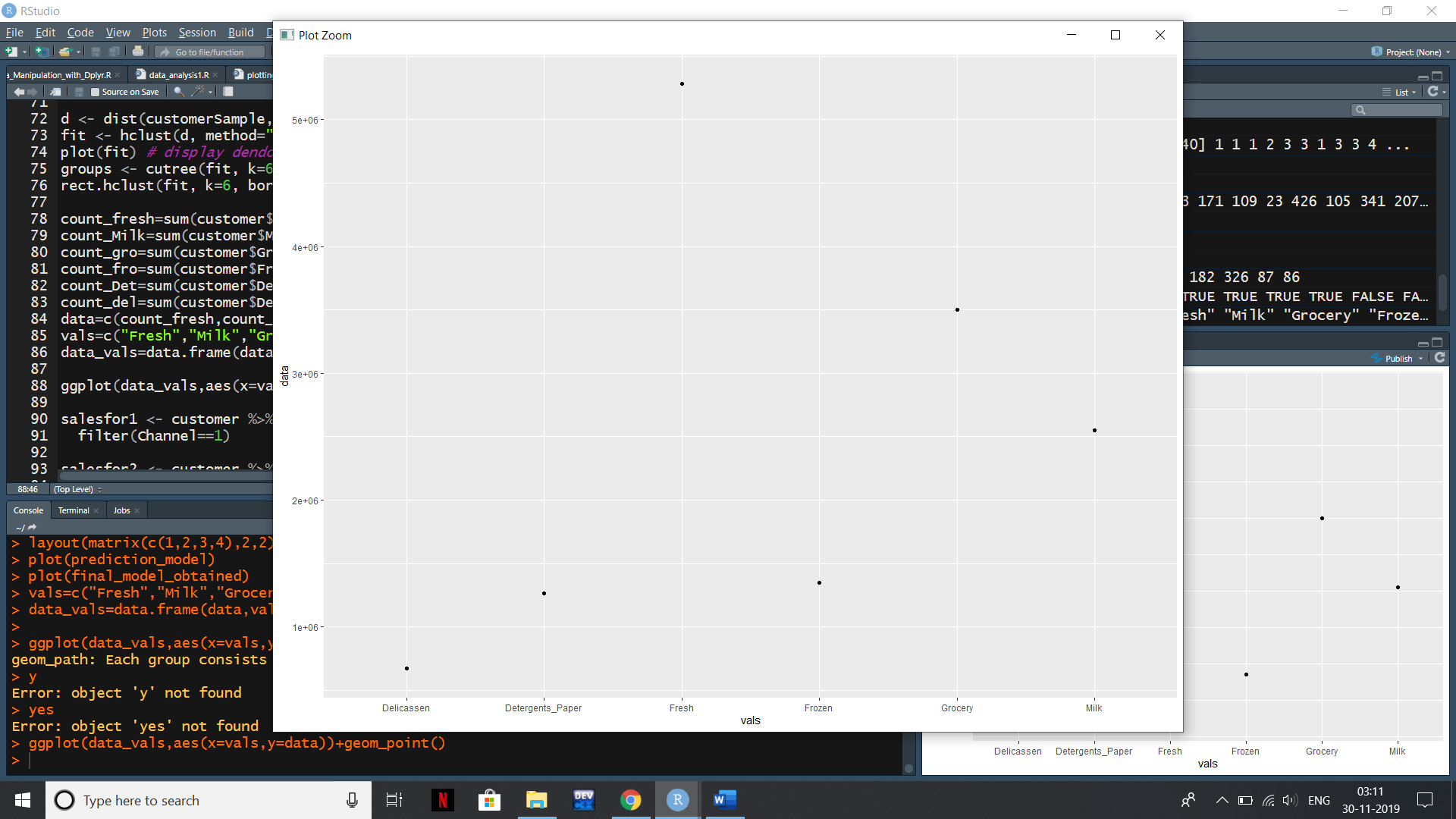
plot(final\_model\_obtained)

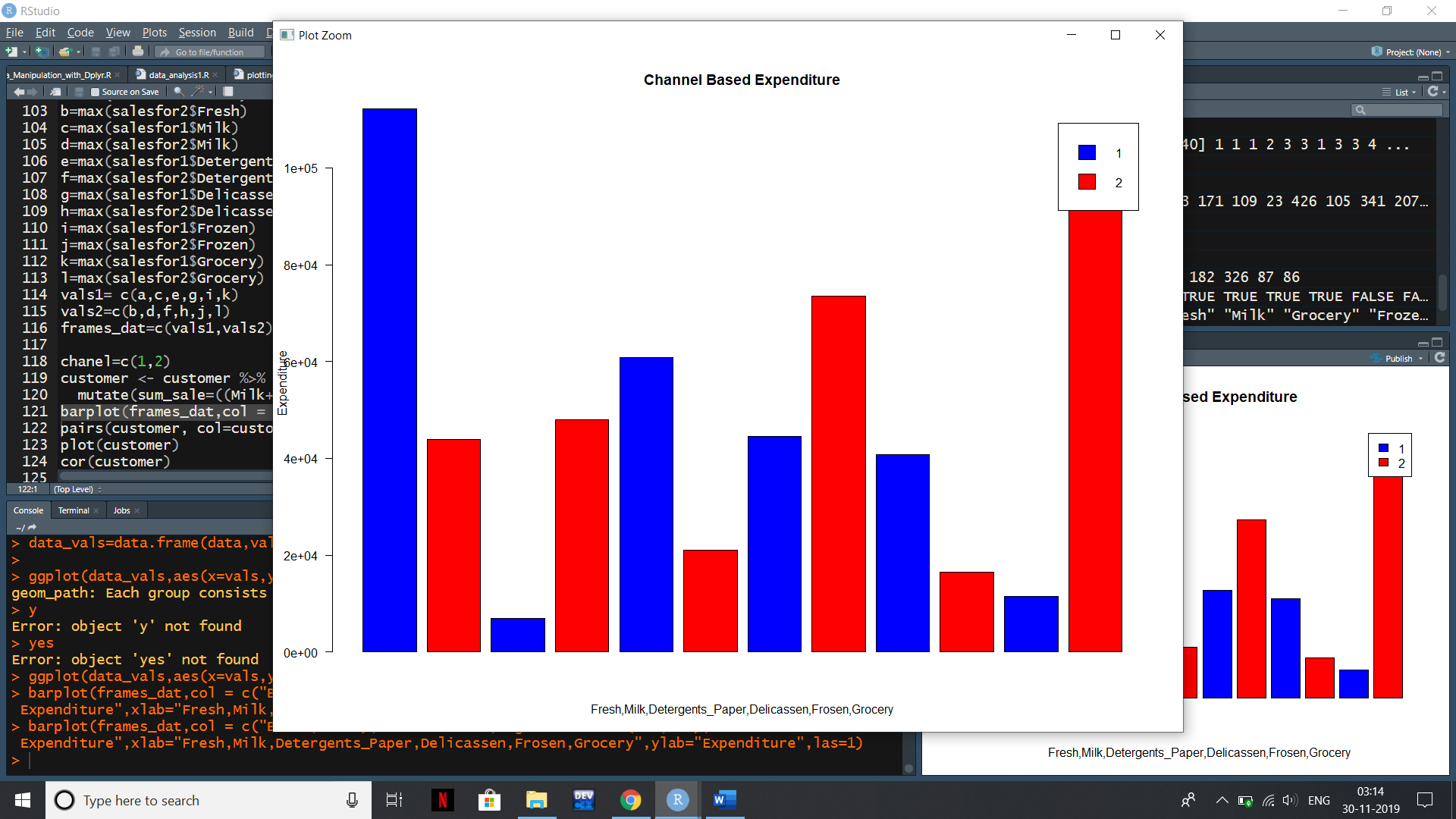
**SCREENSHOTS**

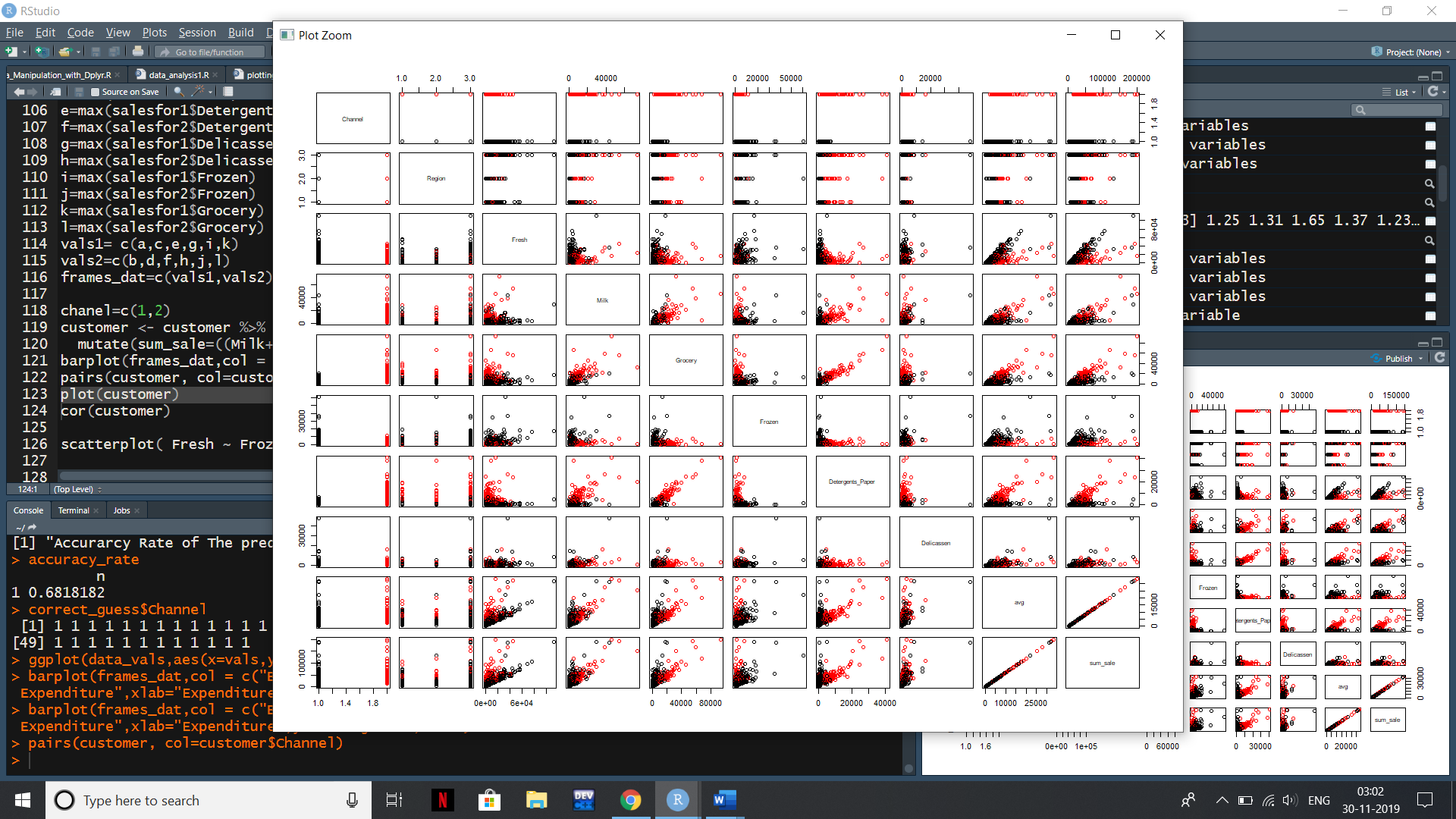


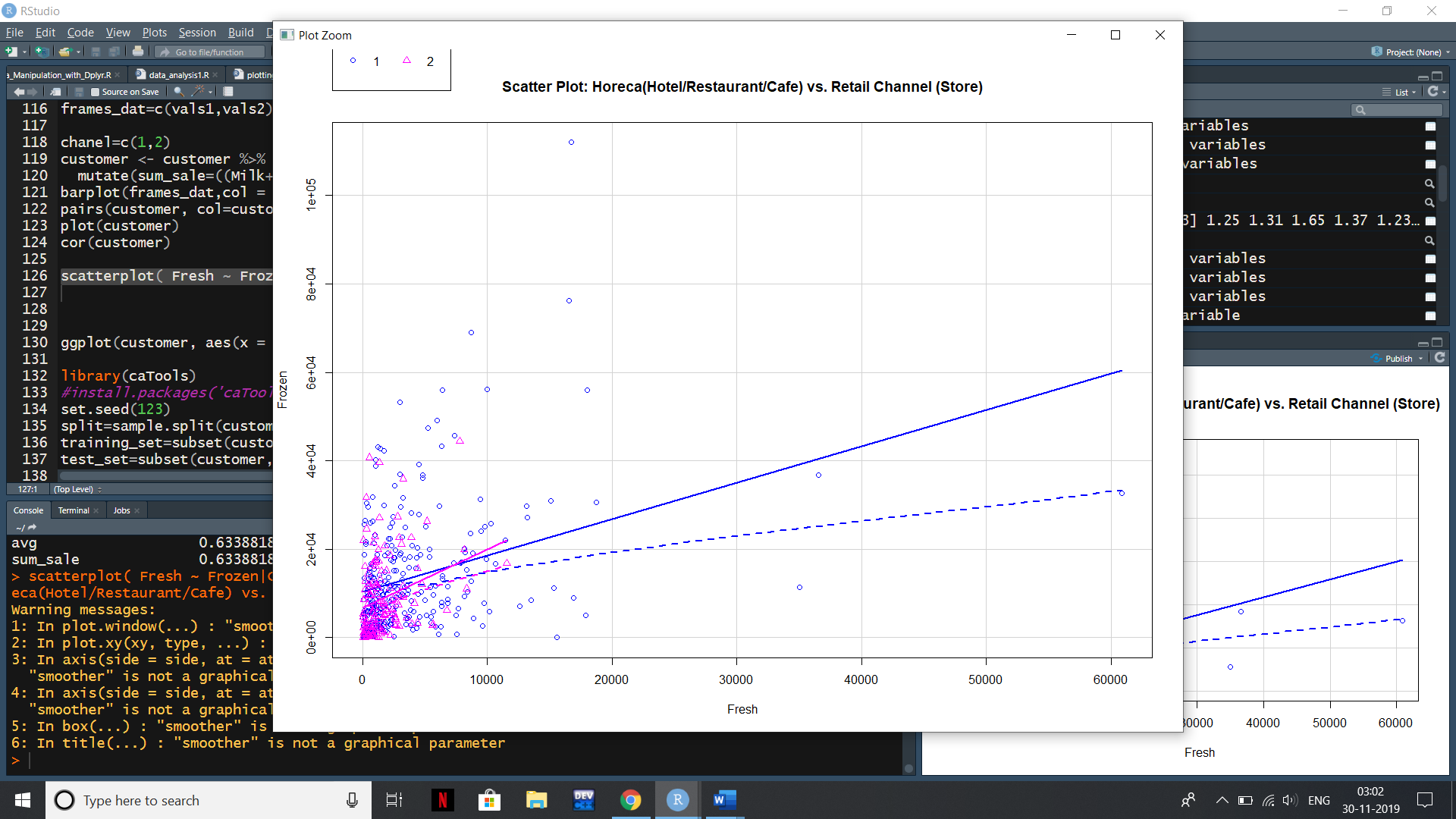


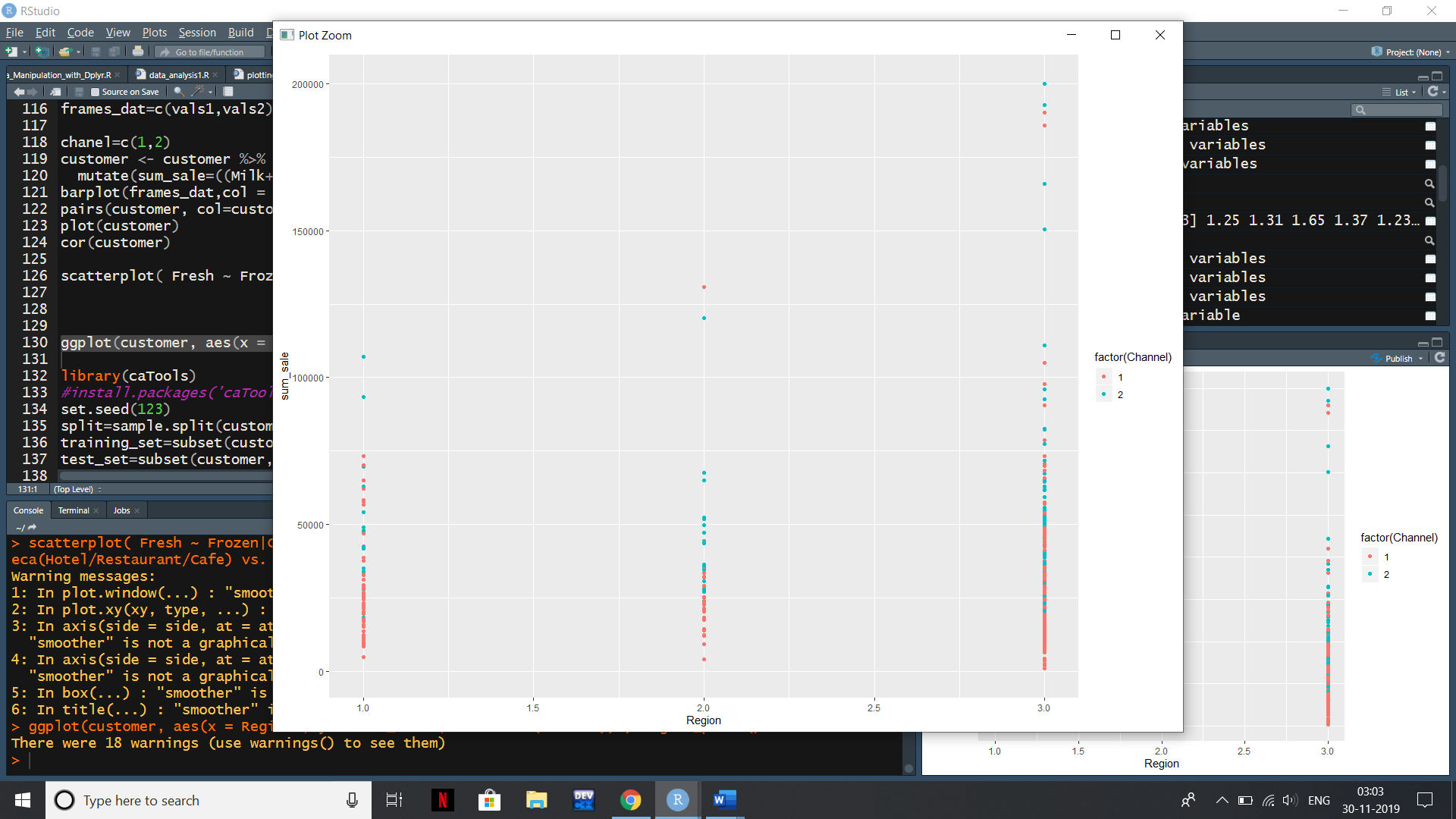


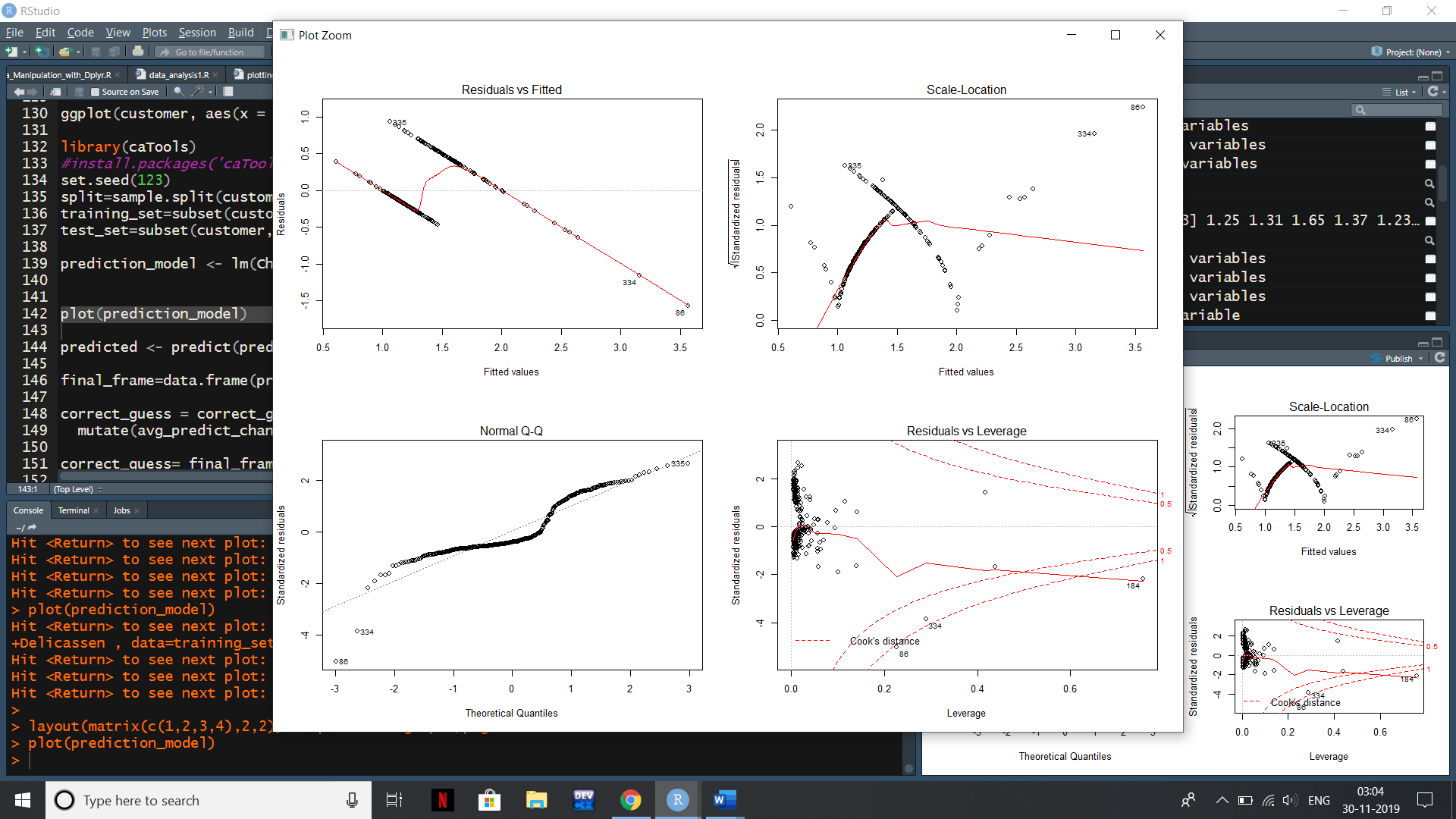


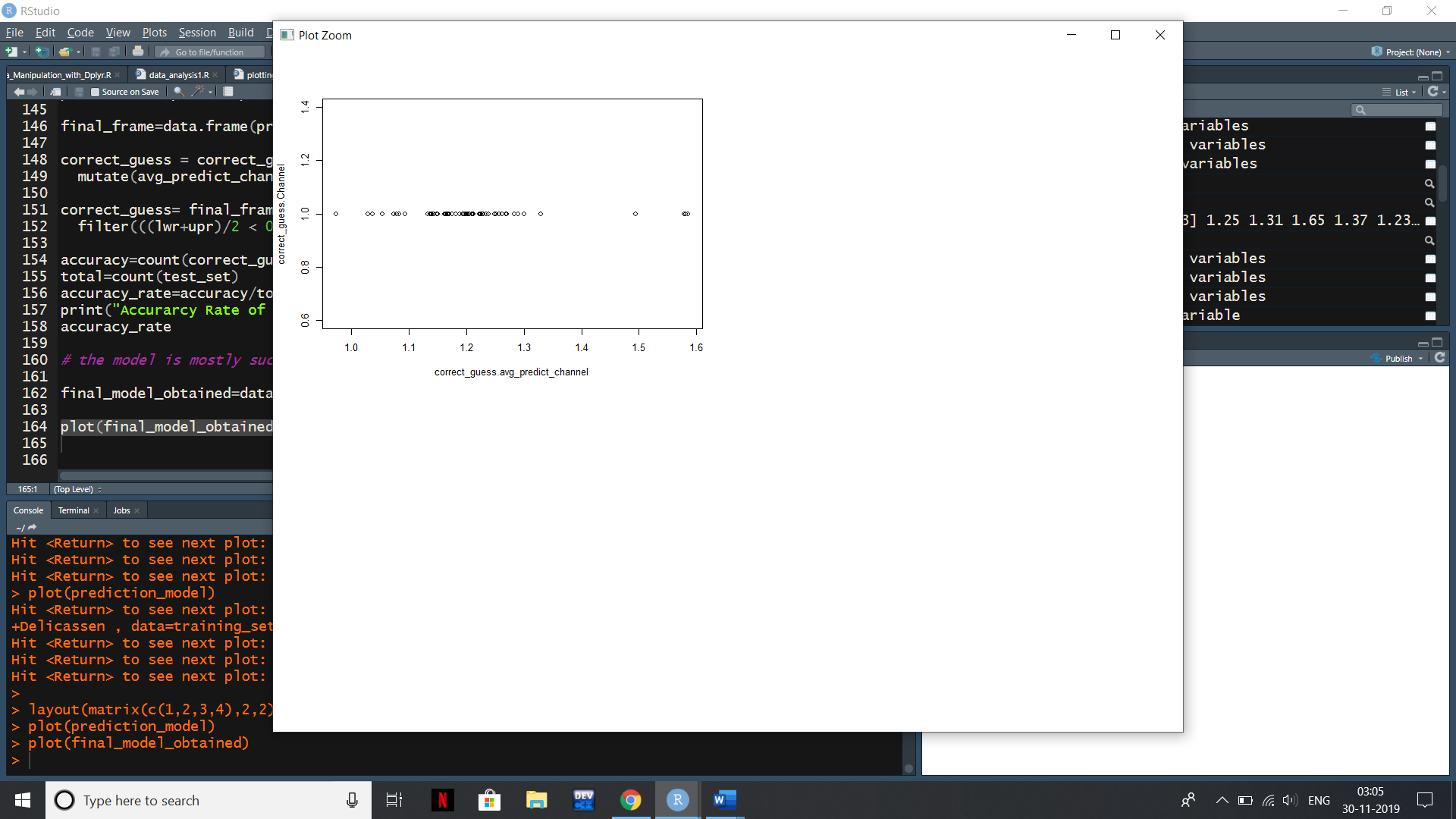












**CONCLUSION AND FUTURE SCOPE**

**4.1 CONCLUSION**

* Easier Way to group similar data together and helps getting a in brief look of big data
* Helps in predicting values of dimension using independent dimensions and form a model for

this purpose.

* 1. **Future scope**
* The Prediction model has an accuracy rate of 0.68 which can be increased by

targeted regression models and adding more data in the training set.

* The Clustering helps us in analyzing the type of data we are dealing with in brief and

make decision making easier and efficient. But we can decrease the outliers present in the model.

**REFERENCES**

1. <https://www.statmethods.net/graphs/scatterplot.html>
2. <https://www.tutorialspoint.com/r/r_multiple_regression.htm>
3. <https://www.geeksforgeeks.org/clustering-in-machine-learning/>
4. <https://www.tutorialspoint.com/r/index.htm>