**Retail Store Case Study Report for the Black Friday sale**

**Group No**.: Group 18

**Student Names**: Isha Pote, Akhilesh Ghadge

Executive Summary: The dataset contains data from a retail store that would like to understand the purchasing habits of their customers so that they can offer a personalized list of products that would interest their customers. We impute the missing data and perform EDA. We use machine learning in order to build a prediction model that will predict a customers purchase amount. A list of machine learning models is compiled and built using tenfold repeated cross validation with three repeats. These models include a glm model, a glmnet model, a linear regression model, a GBM model and a treebag model. The model that produced the best median RMSE was the gbm model. This model produced a median RMSE of 3002.142. In order to increase sales, there are two possible solutions. First to market the products to the part of the demographic that doesn’t shop at the given retail store. The second to market products to the part of the demographic that buys the products based on their purchase history and future purchase predictions.

# Background and Introduction

A retail store would like to understand the purchasing habits of its customers so that they can offer customized list of products that would interest those customers.

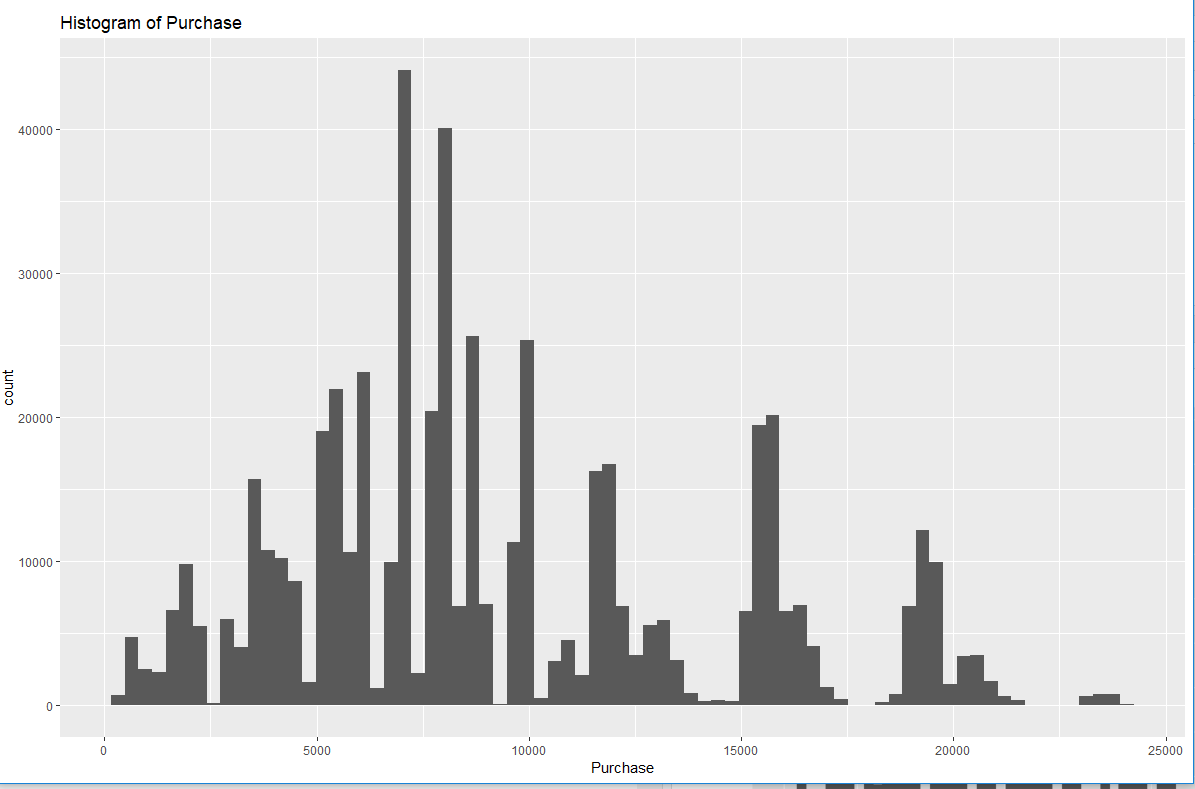
The report will explore a month’s worth of sales data from the store. The variable of interest in this report is amount purchased in the last month. The data also contains the demographic information including age, gender, occupation, city, category, stay in current city and marital status. Additionally, the dataset also contains product information including ID and different product category information.

Using this information, a machine learning model will be built that will predict purchase amount based on the customer’s demographics and the categories of the product and the store will have information they need to offer customers the products they need.

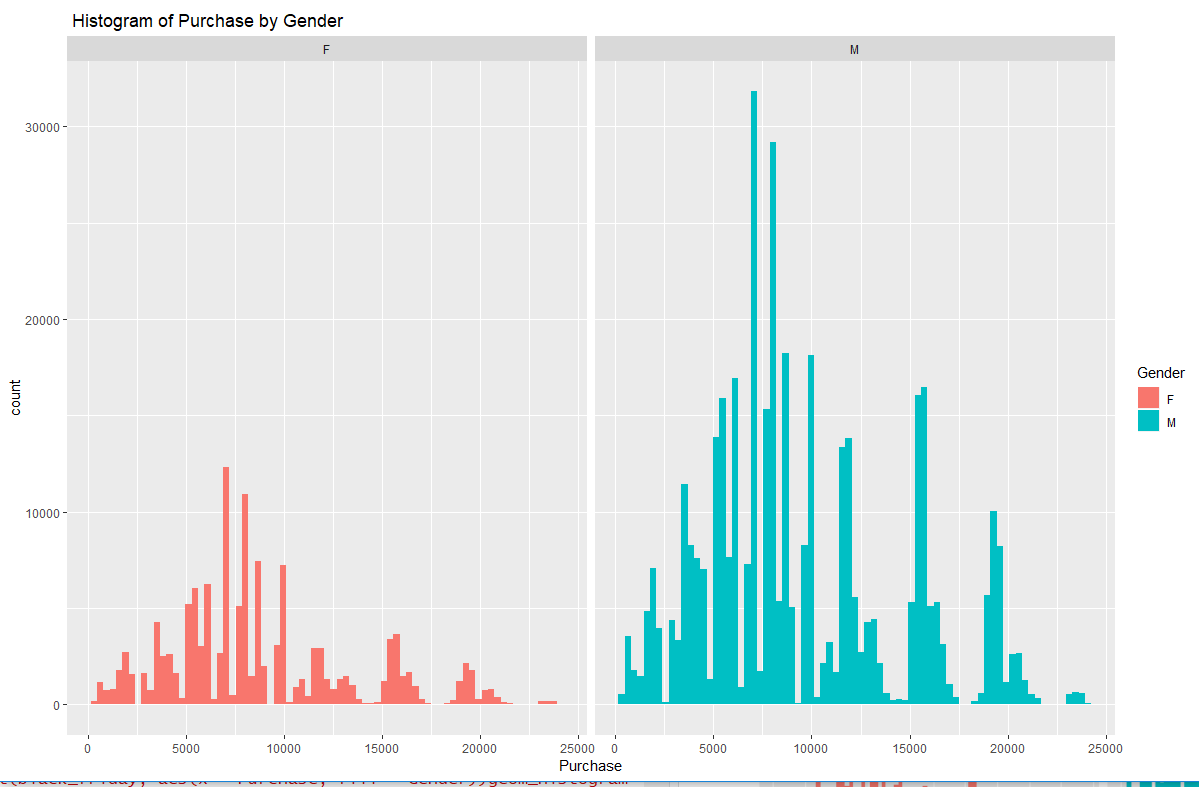
1. **Data Exploration and Visualization**

The first step in exploring the data is loading the required libraries. The dataframe has 12 variables and 550068 observations. Inspecting the data reveals that every variable other than Purchase is a categorical variable. The columns that require changing to factor variables Marital\_Status, Occupation, User\_ID, Product\_Category\_1, Product\_Category\_2 and Product\_Category\_3.

Another problem with data is that Product\_Category\_2 and Product\_Category\_3 have missing values. These columns contain categorical data, so it won’t be appropriate to use median or kNN imputation. All the missing values are imputed as 0.



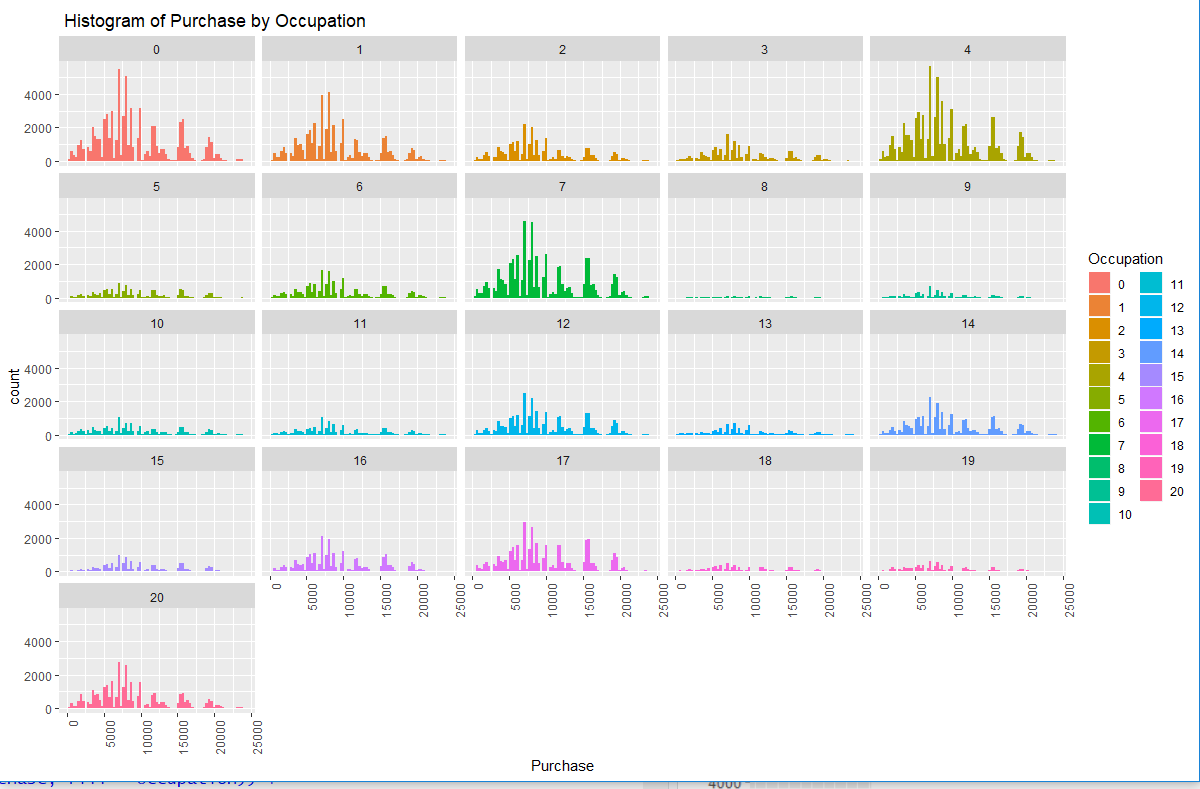
The target variable Purchase has an almost Gaussian distribution. A histogram of the purchase variable shows a unimodal curve that has a positive skew which explains why the mean of purchase amount variable is larger than the median of purchase amount variable.



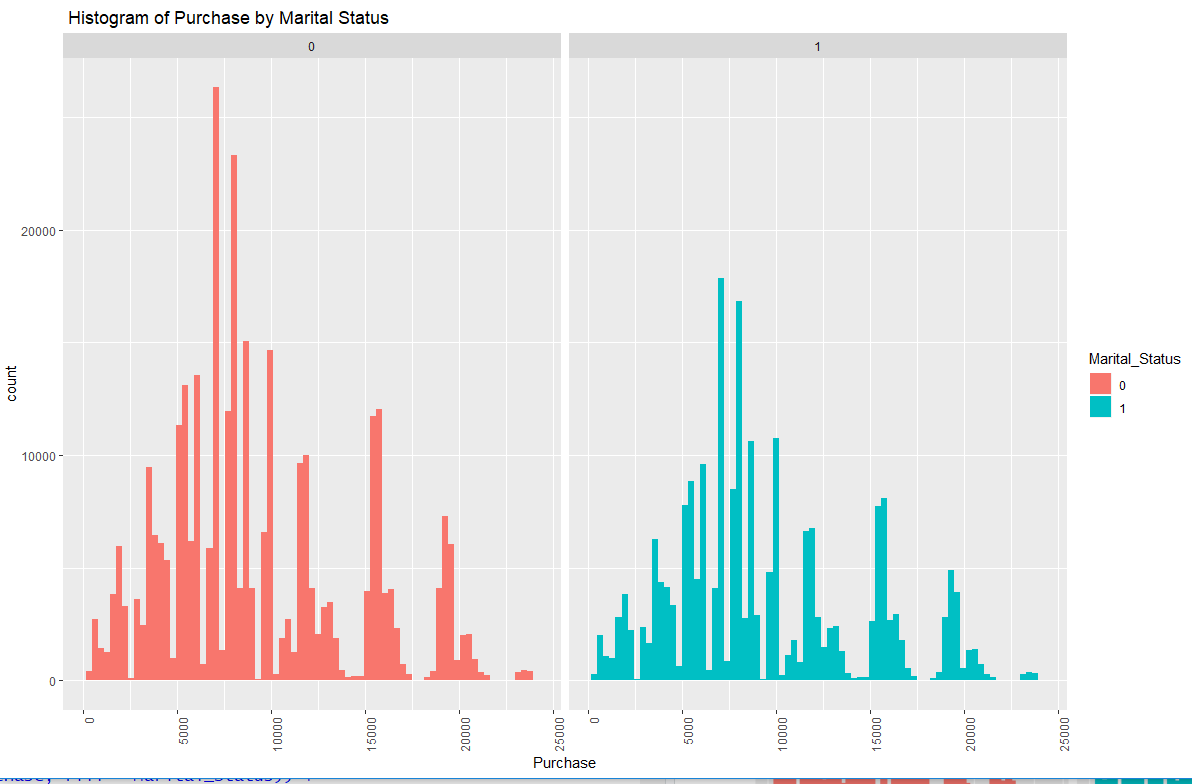
On average the men spend more money on purchase than women. This last conclusion is more reasonable since the percentage of male buyers is higher than female buyers. Further, men shopped for more products than women.



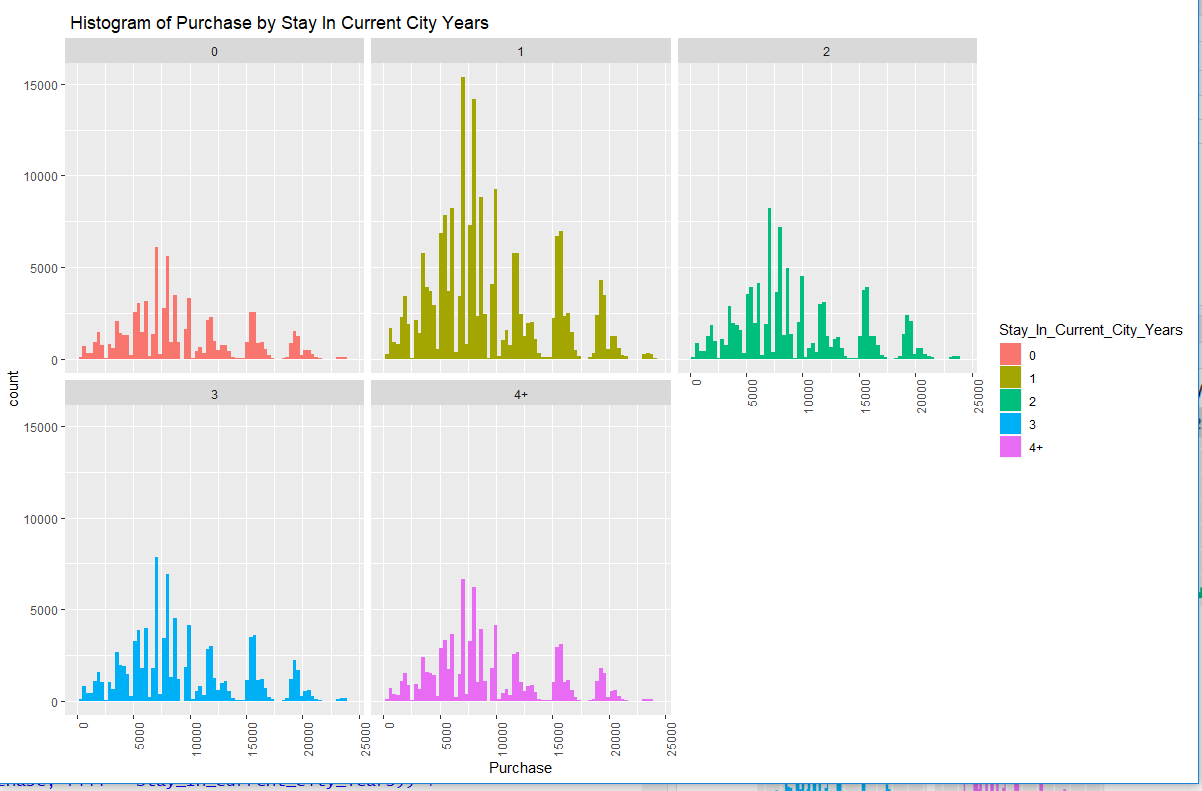
Curiously, on average customer with more than 50 years old are the ones who spent the most.



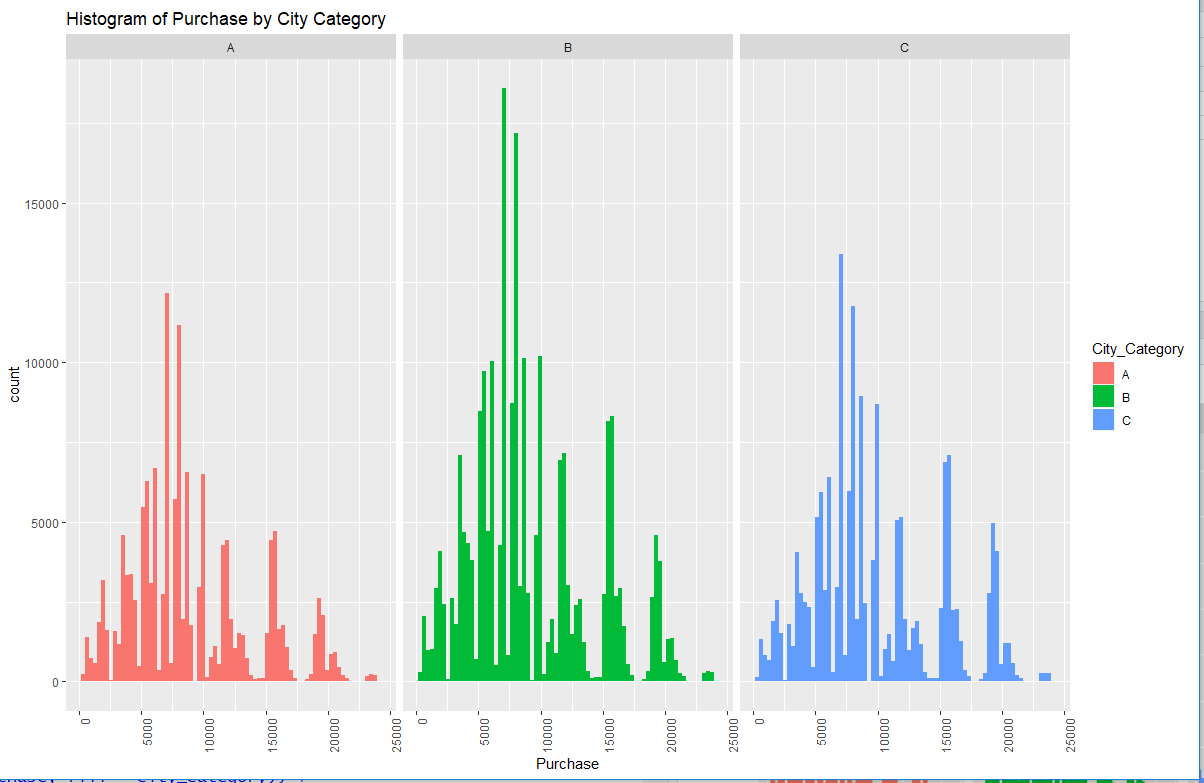
There are some occupations which have higher representations, but the amount each user spends on average is the same for all occupations.



On average an individual customer tends to spend the same amount independently if his/her is married or not.



The longest someone is living in that city the less prone they are to buy new things. Hence, if someone is new in town and needs a great number of new things for their house that they’ll take advantage of the low prices in Black Friday to purchase all the things needed.



We see that city type ‘B’ had the highest number of purchases registered. However, the city whose buyers spend the most is city type ‘C’.

After fully cleaning the data the cleaned dataset is examined. It reveals that the median purchase amount for the last month was $8,047. In addition, it shows that the mean was $9,264. The minimum purchase amount was $12 and the maximum purchase amount was $23,961.

# Data Preparation and Preprocessing

The columns with categorical variables are changed to numeric variables. These include Marital\_Status, Occupation, User\_ID, Product\_Category2, Product\_Category\_3. The columns Product\_Category\_2 and Product\_Category\_3 have missing values. We impute the nan values to 0.

After imputing the nan values, we check the spread of Product\_Category\_1, Product\_Category\_2, Product\_Category\_3. Then we calculate the standard deviation of purchase. Then we find the final summary of the data.

After fully cleaning the data the cleaned dataset is examined. The median purchase amount for the last month was $8,047. Additionally, the mean was $9,264. The minimum purchase amount was $12 and the maximum purchase amount was $23,961.

# Data Mining Techniques and Implementation

* This report uses machine learning in order to build a prediction model that will predict a customer’s purchase amount. The first step to building this model is removing the User\_ID and Product\_ID columns as these variables have zero variance. Each Product\_ID and each User\_ID will be particular to the customer or product.
* After these near zero variance variables were removed, a sample was selected from the data. This sample was selected because this dataset is large. Fortunately, a large sample of the data should be enough to represent the data accurately and to build a machine learning model. The sample that was selected was selected randomly. The sample size 10% of the data. Once the sample is selected, the next step in building the model can be performed.
* Next the sampled data was partitioned. 70% of the sample was selected randomly to be the train set. The train set is the data that will be used to build the algorithm. The other 30% of the sample was assigned to the test set. The test set will be used to test the accuracy of the model.
* Once the sample is partitioned, a list of machine learning algorithms using different methods is compiled. These algorithms were built using ten-fold repeated cross-validation with three repeats. The models created include a glm model, a glmnet model, a linear regression model, a GBM model and a treebag model.
* Next, three ensemble models were created. These ensemble models included a linear regression model, a glmnet model and a gbm model. Each model also used ten-fold cross-validation repeated three times. All three of these models proved to be better predictors than any of the other models alone.

**Data Exploration and Visualization**

# Data Preparation and Preprocessing

Imputing values , converting categorical variables to binary, splitting data into train and test set.

**Machine Learning Algorithms**

**Training algorithm**

a glm model, a glmnet model, a linear regression model, a GBM model and a treebag model.

**Testing**

linear regression model, a glmnet model and a gbm model.

Choose the best model.

V. Performance Evaluation

* Once the sample is partitioned, a list of machine learning algorithms using different methods is compiled. These algorithms were built using ten-fold repeated cross-validation with three repeats. The model that produced the best median RMSE was the gbm model. This model produced a median RMSE of 3002.142. Using this model would produce fairly accurate predictions. However, creating an ensemble model using these models should produce even greater accuracy.
* Next, three ensemble models were created. These ensemble models included a linear regression model, a glmnet model and a gbm model. Each model also used ten-fold cross-validation repeated three times. All three of these models proved to be better predictors than any of the other models alone. The gbm model produced an RMSE of 2980.54. The linear model produced an RMSE of 2993.22 and the glmnet model produced an RMSE of 2993.751.
* Once the different models were used to make predictions, it is discovered the the glmnet stack produced the best predictions. This model produced an RMSE of 3085.436. Therefore, it was the model used to make the final predictions for the testing dataset.

# VI. Discussion and Recommendation

There are a few conclusions that can be made using the analysis in this paper.

* The first conclusion is that even when broken down into different demographics, the median purchase made by customers does not fluctuate much. It didn’t matter if the group was male, female, young, old, married or unmarried, the median purchase by the customers hovered around $8000.
* However, some groups were more present than others. Males shopped more than females. The marital status 0 shopped more than the marital status 1. Unfortunately, which label mean married, and which label means unmarried is unknown. Also, customers between the ages of 18 and 45 shopped the most. The age range 26-35 had the highest turnout. Additionally, people who only lived in their city for a year shopped a lot.

There are two different ways that the retail store could increase their sales.

* First, to advertise to groups that do not shop much. Further research has to be done in order to come up with a targeted marketing campaign.
* The other option will be to target the customers that shop often offer customized list of products that would interest those customers.

Finally, the models were tested to find the model that makes the best predictions. When analyzing the data from the testing model, it is revealed that the Product\_Category\_1, Product\_Category\_2 and Product\_Category\_3 variables have new levels. These new variables will present a problem when making predictions. Therefore, the original models needed to be revisited and these variables were left as numeric variables after 0 was imputed for the missing values.

The glmnet stack produced the best predictions. This model produced an RMSE of 3085.436. Therefore, it was the model used to make the final predictions for the testing dataset.

# VII. Summary

The dataset contains data from ABC Private Limited that would like to understand the purchasing habits of their customers so that they can offer a personalized list of products that would interest their customers.

We impute the missing data and perform EDA. Then we split the data into train data and test data. We use machine learning in order to build a prediction model that will predict a customers purchase amount. A list of machine learning models is compiled and built using tenfold repeated cross validation with three repeats. These models include a glm model, a glmnet model, a linear regression model, a GBM model and a treebag model. The model that produced the best median RMSE was the gbm model. This model produced a median RMSE of 3002.142.

We then create three ensemble models which include a linear regression model, a glmnet model and a gbm model. Each model uses ten-fold cross-validation repeated three times. The gbm model produced an RMSE of 2980.54. The linear model produced an RMSE of 2993.22 and the glmnet model produced an RMSE of 2993.751. The glmnet stack produced the best results so this model was used to produce the final predictions for the training dataset.

# Appendix: R Code for use case study

install.packages("dplyr")

library(dplyr)

library(ggplot2)

install.packages("caret")

library(caret)

install.packages("caretEnsemble")

library(caretEnsemble)

library(VIM)

install.packages("gridExtra")

library(gridExtra)

install.packages("glmnet")

library(glmnet)

#loading data

black\_friday <-read.csv("BlackFriday.csv")

#Previewing Data

head(black\_friday)

**Output-**

User\_ID Product\_ID Gender Age Occupation City\_Category Stay\_In\_Current\_City\_Years

1 1000001 P00069042 F 0-17 10 A 2

2 1000001 P00248942 F 0-17 10 A 2

3 1000001 P00087842 F 0-17 10 A 2

4 1000001 P00085442 F 0-17 10 A 2

5 1000002 P00285442 M 55+ 16 C 4+

6 1000003 P00193542 M 26-35 15 A 3

Marital\_Status Product\_Category\_1 Product\_Category\_2 Product\_Category\_3 Purchase

1 0 3 NA NA 8370

2 0 1 6 14 15200

3 0 12 NA NA 1422

4 0 12 14 NA 1057

5 0 8 NA NA 7969

6 0 1 2 NA 15227

#Understanding the structure of data

str(black\_friday)

**Output-**

'data.frame': 537577 obs. of 12 variables:

$ User\_ID : int 1000001 1000001 1000001 1000001 1000002 1000003 1000004 1000004 1000004 1000005 ...

$ Product\_ID : Factor w/ 3623 levels "P00000142","P00000242",..: 671 2375 851 827 2733 1830 1744 3319 3597 2630 ...

$ Gender : Factor w/ 2 levels "F","M": 1 1 1 1 2 2 2 2 2 2 ...

$ Age : Factor w/ 7 levels "0-17","18-25",..: 1 1 1 1 7 3 5 5 5 3 ...

$ Occupation : int 10 10 10 10 16 15 7 7 7 20 ...

$ City\_Category : Factor w/ 3 levels "A","B","C": 1 1 1 1 3 1 2 2 2 1 ...

$ Stay\_In\_Current\_City\_Years: Factor w/ 5 levels "0","1","2","3",..: 3 3 3 3 5 4 3 3 3 2 ...

$ Marital\_Status : int 0 0 0 0 0 0 1 1 1 1 ...

$ Product\_Category\_1 : int 3 1 12 12 8 1 1 1 1 8 ...

$ Product\_Category\_2 : int NA 6 NA 14 NA 2 8 15 16 NA ...

$ Product\_Category\_3 : int NA 14 NA NA NA NA 17 NA NA NA ...

$ Purchase : int 8370 15200 1422 1057 7969 15227 19215 15854 15686 7871 ...

#summarising data

summary(black\_friday)

**Output-**

User\_ID Product\_ID Gender Age Occupation City\_Category

Min. :1000001 P00265242: 1858 F:132197 0-17 : 14707 Min. : 0.000 A:144638

1st Qu.:1001495 P00110742: 1591 M:405380 18-25: 97634 1st Qu.: 2.000 B:226493

Median :1003031 P00025442: 1586 26-35:214690 Median : 7.000 C:166446

Mean :1002992 P00112142: 1539 36-45:107499 Mean : 8.083

3rd Qu.:1004417 P00057642: 1430 46-50: 44526 3rd Qu.:14.000

Max. :1006040 P00184942: 1424 51-55: 37618 Max. :20.000

(Other) :528149 55+ : 20903

Stay\_In\_Current\_City\_Years Marital\_Status Product\_Category\_1 Product\_Category\_2

0 : 72725 Min. :0.0000 Min. : 1.000 Min. : 2.00

1 :189192 1st Qu.:0.0000 1st Qu.: 1.000 1st Qu.: 5.00

2 : 99459 Median :0.0000 Median : 5.000 Median : 9.00

3 : 93312 Mean :0.4088 Mean : 5.296 Mean : 9.84

4+: 82889 3rd Qu.:1.0000 3rd Qu.: 8.000 3rd Qu.:15.00

Max. :1.0000 Max. :18.000 Max. :18.00

NA's :166986

Product\_Category\_3 Purchase

Min. : 3.0 Min. : 185

1st Qu.: 9.0 1st Qu.: 5866

Median :14.0 Median : 8062

Mean :12.7 Mean : 9334

3rd Qu.:16.0 3rd Qu.:12073

Max. :18.0 Max. :23961

NA's :373299

#Changing Numeric Variables to Categorical Variables

black\_friday$Marital\_Status <- factor(black\_friday$Marital\_Status)

black\_friday$Occupation <- factor(black\_friday$Occupation)

black\_friday$User\_ID <- factor(black\_friday$User\_ID)

black\_friday$Product\_Category\_2 <- factor(black\_friday$Product\_Category\_2)

black\_friday$Product\_Category\_3 <- factor(black\_friday$Product\_Category\_3)

#new summary of data

summary(black\_friday)

**Output-**

User\_ID Product\_ID Gender Age Occupation City\_Category

1001680: 1025 P00265242: 1858 F:132197 0-17 : 14707 4 : 70862 A:144638

1004277: 978 P00110742: 1591 M:405380 18-25: 97634 0 : 68120 B:226493

1001941: 898 P00025442: 1586 26-35:214690 7 : 57806 C:166446

1001181: 861 P00112142: 1539 36-45:107499 1 : 45971

1000889: 822 P00057642: 1430 46-50: 44526 17 : 39090

1003618: 766 P00184942: 1424 51-55: 37618 20 : 32910

(Other):532227 (Other) :528149 55+ : 20903 (Other):222818

Stay\_In\_Current\_City\_Years Marital\_Status Product\_Category\_1 Product\_Category\_2

0 : 72725 0:317817 Min. : 1.000 8 : 63058

1 :189192 1:219760 1st Qu.: 1.000 14 : 54158

2 : 99459 Median : 5.000 2 : 48481

3 : 93312 Mean : 5.296 16 : 42602

4+: 82889 3rd Qu.: 8.000 15 : 37317

Max. :18.000 (Other):124975

NA's :166986

Product\_Category\_3 Purchase

16 : 32148 Min. : 185

15 : 27611 1st Qu.: 5866

14 : 18121 Median : 8062

17 : 16449 Mean : 9334

5 : 16380 3rd Qu.:12073

(Other): 53569 Max. :23961

NA's :373299

#spread of Product categories

table(black\_friday$Product\_Category\_1)

**Output-**

1 2 3 4 5 6 7 8 9 10 11 12 13

138353 23499 19849 11567 148592 20164 3668 112132 404 5032 23960 3875 5440

14 15 16 17 18

1500 6203 9697 567 3075

table(black\_friday$Product\_Category\_2)

**Output-**

2 3 4 5 6 7 8 9 10 11 12 13 14 15 16

48481 2835 25225 25874 16251 615 63058 5591 2991 13945 5419 10369 54158 37317 42602

17 18

13130 2730

table(black\_friday$Product\_Category\_3)

**Output-**

3 4 5 6 8 9 10 11 12 13 14 15 16 17 18

600 1840 16380 4818 12384 11414 1698 1773 9094 5385 18121 27611 32148 16449 4563

#imputing 0 for missing values in Product\_Category\_2 & Product\_category\_3

black\_friday$Product\_Category\_2 <- as.numeric(black\_friday$Product\_Category\_2)

black\_friday[is.na(black\_friday$Product\_Category\_2), "Product\_Category\_2"] <- 0

black\_friday$Product\_Category\_3 <- as.numeric(black\_friday$Product\_Category\_3)

black\_friday[is.na(black\_friday$Product\_Category\_3), "Product\_Category\_3"] <- 0

#standard deviation of Purchase

sd(black\_friday$Purchase)

**Output-**

[1] 4981.022

#final summary of data

summary(black\_friday)

**Output-**

User\_ID Product\_ID Gender Age Occupation City\_Category

1001680: 1025 P00265242: 1858 F:132197 0-17 : 14707 4 : 70862 A:144638

1004277: 978 P00110742: 1591 M:405380 18-25: 97634 0 : 68120 B:226493

1001941: 898 P00025442: 1586 26-35:214690 7 : 57806 C:166446

1001181: 861 P00112142: 1539 36-45:107499 1 : 45971

1000889: 822 P00057642: 1430 46-50: 44526 17 : 39090

1003618: 766 P00184942: 1424 51-55: 37618 20 : 32910

(Other):532227 (Other) :528149 55+ : 20903 (Other):222818

Stay\_In\_Current\_City\_Years Marital\_Status Product\_Category\_1 Product\_Category\_2

0 : 72725 0:317817 Min. : 1.000 Min. : 0.000

1 :189192 1:219760 1st Qu.: 1.000 1st Qu.: 0.000

2 : 99459 Median : 5.000 Median : 4.000

3 : 93312 Mean : 5.296 Mean : 6.096

4+: 82889 3rd Qu.: 8.000 3rd Qu.:13.000

Max. :18.000 Max. :17.000

Product\_Category\_3 Purchase

Min. : 0.000 Min. : 185

1st Qu.: 0.000 1st Qu.: 5866

Median : 0.000 Median : 8062

Mean : 2.999 Mean : 9334

3rd Qu.: 5.000 3rd Qu.:12073

Max. :15.000 Max. :23961

#Histogram of Purchase Column

ggplot(black\_friday, aes(x = Purchase)) +

geom\_histogram(bins = 75) +

labs(title= "Histogram of Purchase")

#Histogram of Purchase column vs Gender

ggplot(black\_friday, aes(x = Purchase, fill = Gender)) +

geom\_histogram(bins = 75) +

facet\_grid(. ~ Gender) +

labs(title= " Histogram of Purchase by Gender")

#Histogram of Purchase vs Age

ggplot(black\_friday, aes(x = Purchase, fill = Age)) +

geom\_histogram(bins = 75) +

facet\_wrap(~ Age) +

labs(title= "Histogram Purchase by Age") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

#Histogram of Purchase vs occupation

ggplot(black\_friday, aes(x = Purchase, fill = Occupation)) +

geom\_histogram(bins = 75) +

facet\_wrap(~ Occupation) +

labs(title= " Histogram of Purchase by Occupation") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

#Histogram of Purchase vs Marital Status

ggplot(black\_friday, aes(x = Purchase, fill = Marital\_Status)) +

geom\_histogram(bins = 75) +

facet\_wrap(~ Marital\_Status) +

labs(title= " Histogram of Purchase by Marital Status") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

#Histogram of Purchase vs stay in current city

ggplot(black\_friday, aes(x = Purchase, fill = Stay\_In\_Current\_City\_Years)) +

geom\_histogram(bins = 75) +

facet\_wrap(~ Stay\_In\_Current\_City\_Years) +

labs(title= " Histogram of Purchase by Stay In Current City Years") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

# Histogram of purchase vs city

ggplot(black\_friday, aes(x = Purchase, fill = City\_Category)) +

geom\_histogram(bins = 75) +

facet\_wrap(~ City\_Category) +

labs(title= "Histogram of Purchase by City Category") +

theme(axis.text.x = element\_text(angle = 90, hjust = 1))

# removing Nearzerovariables

bfm <- black\_friday %>%

select(-User\_ID, -Product\_ID)

#New Summary of data

summary(bfm)

**Output-**

Gender Age Occupation City\_Category Stay\_In\_Current\_City\_Years

F:132197 0-17 : 14707 4 : 70862 A:144638 0 : 72725

M:405380 18-25: 97634 0 : 68120 B:226493 1 :189192

26-35:214690 7 : 57806 C:166446 2 : 99459

36-45:107499 1 : 45971 3 : 93312

46-50: 44526 17 : 39090 4+: 82889

51-55: 37618 20 : 32910

55+ : 20903 (Other):222818

Marital\_Status Product\_Category\_1 Product\_Category\_2 Product\_Category\_3 Purchase

0:317817 Min. : 1.000 Min. : 0.000 Min. : 0.000 Min. : 185

1:219760 1st Qu.: 1.000 1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 5866

Median : 5.000 Median : 4.000 Median : 0.000 Median : 8062

Mean : 5.296 Mean : 6.096 Mean : 2.999 Mean : 9334

3rd Qu.: 8.000 3rd Qu.:13.000 3rd Qu.: 5.000 3rd Qu.:12073

Max. :18.000 Max. :17.000 Max. :15.000 Max. :23961

#Sampling data

set.seed(366284)

bf\_sample <- createDataPartition(y = bfm$Purchase,

p = 0.1, list=FALSE)

bf\_sample <- bfm[bf\_sample, ]

#Summary of data after sampling

summary(bf\_sample)

**Output-**

Gender Age Occupation City\_Category Stay\_In\_Current\_City\_Years

F:13231 0-17 : 1461 4 : 7206 A:14170 0 : 7371

M:40528 18-25: 9854 0 : 6830 B:22924 1 :18909

26-35:21456 7 : 5811 C:16665 2 : 9883

36-45:10743 1 : 4543 3 : 9283

46-50: 4587 17 : 3753 4+: 8313

51-55: 3676 20 : 3194

55+ : 1982 (Other):22422

Marital\_Status Product\_Category\_1 Product\_Category\_2 Product\_Category\_3 Purchase

0:31860 Min. : 1.000 Min. : 0.000 Min. : 0.000 Min. : 187

1:21899 1st Qu.: 1.000 1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 5866

Median : 5.000 Median : 4.000 Median : 0.000 Median : 8062

Mean : 5.275 Mean : 6.083 Mean : 2.998 Mean : 9322

3rd Qu.: 8.000 3rd Qu.:13.000 3rd Qu.: 5.000 3rd Qu.:12073

Max. :18.000 Max. :17.000 Max. :15.000 Max. :23961

#Partitiong data

inTrain <- createDataPartition(y = bf\_sample$Purchase,

p = 0.7, list=FALSE)

train <- bf\_sample[inTrain, ]

test <- bf\_sample[-inTrain, ]

#Caretlists

control <- trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE, classProbs = TRUE)

algorithmList <- c('glm', 'glmnet', 'lm', 'treebag', 'gbm')

models <- caretList(Purchase ~ ., train, trControl = control, methodList = algorithmList)

#Testing Models Predictive Accuracy

results <- resamples(models)

summary(results)

**Output-**

Call:

summary.resamples(object = results)

Models: glm, glmnet, lm, treebag, gbm

Number of resamples: 30

MAE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

glm 3485.460 3519.948 3549.570 3545.918 3564.789 3632.974 0

glmnet 3482.270 3518.128 3550.788 3545.017 3563.229 3630.118 0

lm 3485.460 3519.948 3549.570 3545.918 3564.789 3632.974 0

treebag 2325.749 2349.966 2362.541 2365.365 2380.083 2412.726 0

gbm 2251.340 2270.762 2285.390 2284.204 2295.808 2334.849 0

RMSE

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

glm 4538.443 4567.629 4619.134 4616.301 4636.954 4776.629 0

glmnet 4538.934 4569.171 4618.632 4616.074 4636.011 4775.831 0

lm 4538.443 4567.629 4619.134 4616.301 4636.954 4776.629 0

treebag 3014.020 3060.813 3071.443 3075.707 3091.464 3139.114 0

gbm 2937.196 2970.188 2988.407 2989.460 3001.792 3056.735 0

Rsquared

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's

glm 0.1075230 0.1240033 0.1349418 0.1326547 0.1400529 0.1547559 0

glmnet 0.1074201 0.1243869 0.1351358 0.1327726 0.1407449 0.1551233 0

lm 0.1075230 0.1240033 0.1349418 0.1326547 0.1400529 0.1547559 0

treebag 0.5969730 0.6098720 0.6145313 0.6148684 0.6204068 0.6389329 0

gbm 0.6174413 0.6345896 0.6375504 0.6373552 0.6420593 0.6611509 0

#Building Ensembles

stack\_glmnet <- caretStack(models, method = "glmnet", trControl = trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE))

stack\_glmnet

**Output-**

A glmnet ensemble of 2 base models: glm, glmnet, lm, treebag, gbm

Ensemble results:

glmnet

112896 samples

5 predictor

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 3 times)

Summary of sample sizes: 101605, 101607, 101606, 101608, 101607, 101607, ...

Resampling results across tuning parameters:

alpha lambda RMSE Rsquared MAE

0.10 7.91216 2983.289 0.6376127 2272.499

0.10 79.12160 2989.662 0.6361456 2287.714

0.10 791.21601 3024.374 0.6322548 2324.677

0.55 7.91216 2983.326 0.6376066 2272.668

0.55 79.12160 2987.479 0.6367646 2283.953

0.55 791.21601 3052.332 0.6336604 2340.524

1.00 7.91216 2983.396 0.6375926 2273.063

1.00 79.12160 2985.634 0.6372954 2278.291

1.00 791.21601 3087.687 0.6372954 2359.101

RMSE was used to select the optimal model using the smallest value.

The final values used for the model were alpha = 0.1 and lambda = 7.91216.

#testing model

predictions\_glmnet <- predict(stack\_glmnet, test)

error <- predictions\_glmnet - test$Purchase

#calculation rmse

sqrt(mean(error^2))

**Output-**

[1] 3046.695

#Linear Regresson emsemble

stack\_lm <- caretStack(models, method = "lm", trControl = trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE))

stack\_lm

**Output-**

A lm ensemble of 2 base models: glm, glmnet, lm, treebag, gbm

Ensemble results:

Linear Regression

112896 samples

5 predictor

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 3 times)

Summary of sample sizes: 101607, 101607, 101606, 101605, 101606, 101607, ...

Resampling results:

RMSE Rsquared MAE

2982.541 0.6378372 2270.224

Tuning parameter 'intercept' was held constant at a value of TRUE

#testing model by prediction

predictions\_lm <- predict(stack\_lm, test)

error <- predictions\_lm - test$Purchase

sqrt(mean(error^2))

**Output-**

[1] 3045.509

#GBM Ensemble

stack\_gbm <- caretStack(models, method = "gbm", trControl = trainControl(method = "repeatedcv", number = 10, repeats = 3, savePredictions = TRUE))

stack\_gbm

**Output-**

A gbm ensemble of 2 base models: glm, glmnet, lm, treebag, gbm

Ensemble results:

Stochastic Gradient Boosting

112896 samples

5 predictor

No pre-processing

Resampling: Cross-Validated (10 fold, repeated 3 times)

Summary of sample sizes: 101605, 101607, 101608, 101607, 101605, 101606, ...

Resampling results across tuning parameters:

interaction.depth n.trees RMSE Rsquared MAE

1 50 3004.177 0.6365599 2313.764

1 100 2974.843 0.6397946 2262.229

1 150 2973.575 0.6400149 2259.403

2 50 2976.227 0.6396986 2266.802

2 100 2971.281 0.6405699 2257.859

2 150 2969.886 0.6408985 2255.887

3 50 2972.973 0.6402641 2261.679

3 100 2969.574 0.6409731 2256.350

3 150 2968.344 0.6412601 2254.258

Tuning parameter 'shrinkage' was held constant at a value of 0.1

Tuning

parameter 'n.minobsinnode' was held constant at a value of 10

RMSE was used to select the optimal model using the smallest value.

The final values used for the model were n.trees = 150, interaction.depth = 3, shrinkage =

0.1 and n.minobsinnode = 10.

#testing model by gbm

predictions\_gbm <- predict(stack\_gbm, test)

error <- predictions\_gbm - test$Purchase

sqrt(mean(error^2))

**Output-**

[1] 3029.206

#Importing testing data

testing <- read.csv("BlackFriday.csv")

#Converting Data

testing$Marital\_Status <- factor(testing$Marital\_Status)

testing$Occupation <- factor(testing$Occupation)

testing$User\_ID <- factor(testing$User\_ID)

#Imputing 0 for missing values in Product\_Category\_2,Product\_Category\_3

testing$Product\_Category\_2 <- as.numeric(testing$Product\_Category\_2)

testing[is.na(testing$Product\_Category\_2), "Product\_Category\_2"] <- 0

testing$Product\_Category\_3 <- as.numeric(testing$Product\_Category\_3)

testing[is.na(testing$Product\_Category\_3), "Product\_Category\_3"] <- 0

#Removing nonzero values

testing\_sub <- testing %>%

select(-User\_ID, -Product\_ID)

summary(testing\_sub)

**Output-**

Gender Age Occupation City\_Category Stay\_In\_Current\_City\_Years

F:132197 0-17 : 14707 4 : 70862 A:144638 0 : 72725

M:405380 18-25: 97634 0 : 68120 B:226493 1 :189192

26-35:214690 7 : 57806 C:166446 2 : 99459

36-45:107499 1 : 45971 3 : 93312

46-50: 44526 17 : 39090 4+: 82889

51-55: 37618 20 : 32910

55+ : 20903 (Other):222818

Marital\_Status Product\_Category\_1 Product\_Category\_2 Product\_Category\_3 Purchase

0:317817 Min. : 1.000 Min. : 0.000 Min. : 0.000 Min. : 185

1:219760 1st Qu.: 1.000 1st Qu.: 0.000 1st Qu.: 0.000 1st Qu.: 5866

Median : 5.000 Median : 5.000 Median : 0.000 Median : 8062

Mean : 5.296 Mean : 6.785 Mean : 3.872 Mean : 9334

3rd Qu.: 8.000 3rd Qu.:14.000 3rd Qu.: 8.000 3rd Qu.:12073

Max. :18.000 Max. :18.000 Max. :18.000 Max. :23961

#Final testing

testing\_predictions\_glmnet <- predict(stack\_glmnet, testing\_sub)

testing$Purchase <- testing\_predictions\_glmnet

submission\_glmnet <- testing[, c("User\_ID", "Product\_ID", "Purchase")]

dim(submission\_glmnet)

**Output-**

[1] 537577 3

head(submission\_glmnet)

**Output-**

User\_ID Product\_ID Purchase

1 1000001 P00069042 9871.534

2 1000001 P00248942 14974.791

3 1000001 P00087842 1267.045

4 1000001 P00085442 1077.683

5 1000002 P00285442 7871.378

6 1000003 P00193542 13070.109

write.csv(submission\_glmnet, "black\_friday\_predictions.csv",

row.names = FALSE)

**Output-**

****

**https://drive.google.com/open?id=1ByLQ0vBFv7VZxmHxCwwlFrQ8kz7J0ks7**

1. **Citations**

Kolassa, Stephan. "Evaluating predictive count data distributions in retail sales forecasting." *International Journal of Forecasting* 32.3 (2016): 788-803.

Sun, Zhan-Li, et al. "Sales forecasting using extreme learning machine with applications in fashion retailing." *Decision Support Systems* 46.1 (2008): 411-419.

Cumby, Chad, et al. "Predicting customer shopping lists from point-of-sale purchase data." *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2004.

Ragg, Thomas, et al. "Bayesian learning for sales rate prediction for thousands of retailers." *Neurocomputing* 43.1-4 (2002): 127-144.

Sagaert, Yves R., et al. "Tactical sales forecasting using a very large set of macroeconomic indicators." *European Journal of Operational Research* 264.2 (2018): 558-569.

Chu, Ching-Wu, and Guoqiang Peter Zhang. "A comparative study of linear and nonlinear models for aggregate retail sales forecasting." *International Journal of production economics* 86.3 (2003): 217-231.

Ramasubramanian, Karthik, and Abhishek Singh. "Machine Learning Theory and Practices." *Machine Learning Using R*. Apress, Berkeley, CA, 2017. 219-424.