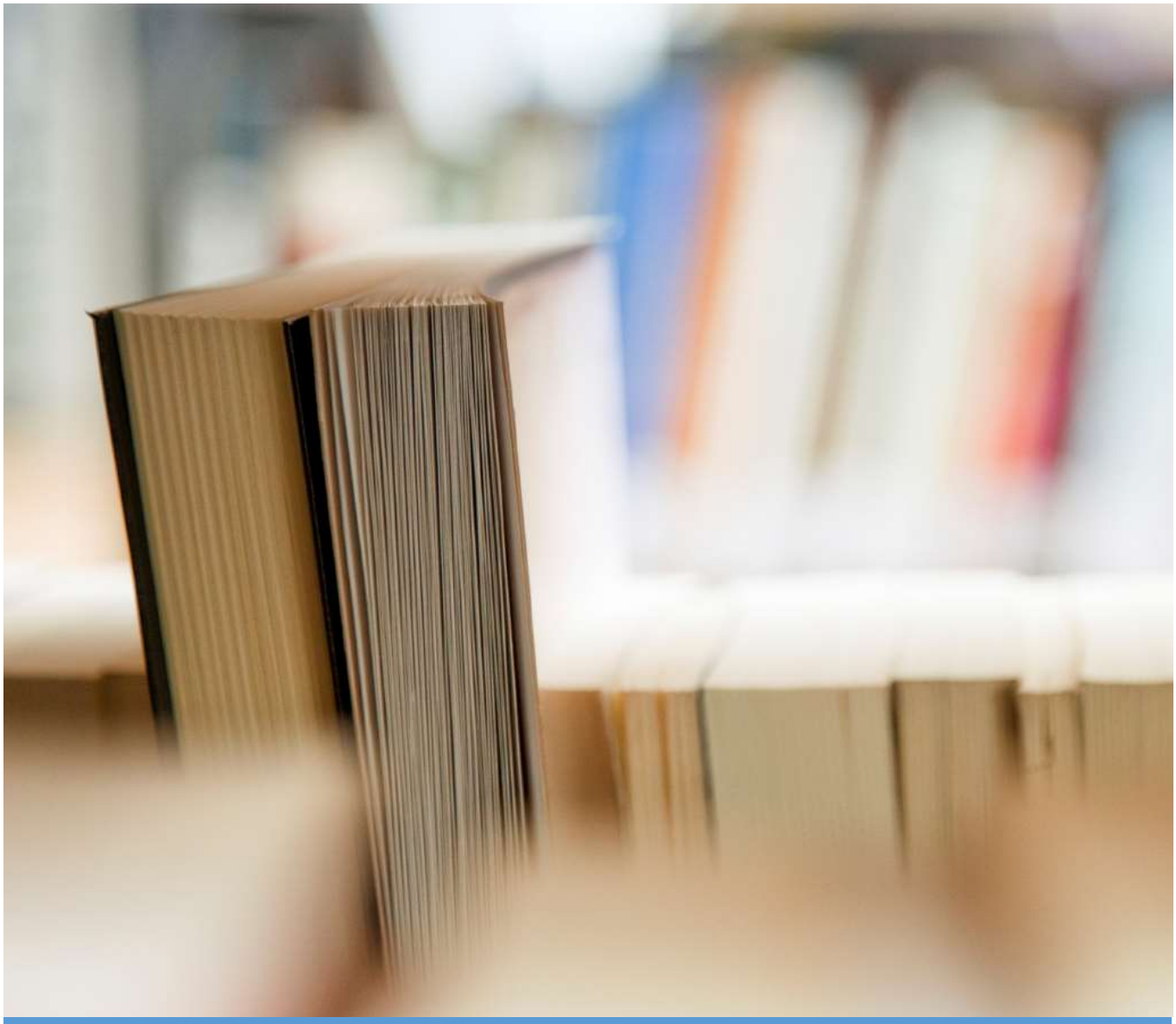


CA-2 DATA MINING

Application of Data mining Tools & Techniques.

10394423-Rohit Kamalakar Patil

4/22/19

M.Sc. Data Analytics.

Business Understanding:

1- Determining Business Objectives

A retail company wants to get insights about the customer purchase behaviour against various products of different categories. Provided data is sample of transactions made in retail store.

Identify the customer purchase behavior give advantages to the company in different line of business, such as marketing, customer care, or business development. As seen from provided data there is no analytical approach currently.

At that stage, the company want to build a machine learning model to predict the purchase amount of customer against various products which will help them to create personalized offer for customers against different products.

From provided dataset are there any clusters for consumers within the data is the main concern. Specifically, here the problem is a regression problem where we are trying to predict the dependent variable (the amount of purchase) with the help of the information contained in the other variables. Classification problem can also be settled in this dataset since several variables are categorical, and some other approaches could be "Predicting the age of the consumer" or even "Predict the category of goods bought".

2- Assessing the Current Situation

What sort of data are available for analysis? Are there any risk related to the project?
If we analyze the data, there is some information related to the customer demographics such as age group, sex, occupation and marital status. On the other hand, we have data on the city's size and how many years the customer has lived in it whereas on the product's side there is only information regarding the categories and the amount spent.

User ID	Product ID	Gender	Age	Occupati...	City C...	Stay In Current Cit...	Marital Status	Product Ca...	Product Cat...	Product C...	Purchase
1004033	P00019842	M	26-35	6	B	19595	0	5	null	null	6,967
1004033	P00219742	M	26-35	6	B	19595	0	6	11	16	16,110
1004034	P00051442	M	26-35	0	C	19595	1	8	17	null	9,770
1004034	P00112142	M	26-35	0	C	19595	1	1	2	14	15,751
1004034	P00215642	M	26-35	0	C	19595	1	8	null	null	6,033
1004034	P00227142	M	26-35	0	C	19595	1	5	null	null	3,528
1004034	P00346142	M	26-35	0	C	19595	1	1	15	null	15,641
1004035	P00237542	M	18-25	4	C	19595	1	1	15	16	19,525
1004035	P00260042	M	18-25	4	C	19595	1	5	8	null	8,806
1004036	P00013742	M	26-35	1	C	19177	1	5	null	null	7,046
1004036	P00211142	M	26-35	1	C	19177	1	5	null	null	5,150

Given data is showing the amount of spending in specific date which is on Black Friday. Therefore, making generalization about customer's general behavior would not be sufficient. In addition, given features might not be enough for making predictions.

3- Determine data mining goals

Project objective is detecting the customer purchase behavior based on purchases and product category. The goals at the end of the project are:

- How much customer will spend for specific product?
- Finding helpful insights for financial planning, inventory and human resource management, marketing and advertising
- Provide information about Customer, Store, Location level

Data Understanding:

The data understanding phase involves taking a closer look at the data available for mining. This step is critical in avoiding unexpected problems during the next phase--data preparation--which is generally defined as the longest part of a project.

1- Collecting Initial Data

This data is transactional data. We can look data with different perspectives. We could divide data with different levels such as, Customer level, Product level and Location level.

Customer Level Data: Provided data has clear demographics related with the customer, age, location, gender, marital status and occupation. With this information purchase behavior and target customers could be identified by customer level.

Product Level Data: Although detailed explanations are not provided about the product, product categories are given. Product level analysis could help to reach which product and

Product categories are profitable, are there any relationship between any product bundles.

Location Level Data: In this data set city level information is given. Which city is profitable, the density of customers in city level could be provide insight about the store locations.

The data is available at https://www.kaggle.com/mehdidag/black-friday.

2- Take a Quick look at Data Structure:

There are 537577 entries and 12 attributes in the dataset. Product_Category_2 and Product_Category_3 should has missing values because these attributes has less than 537577 entries. There are categorical and numeric fields.

```
In [40]: blackfriday.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 537577 entries, 0 to 537576
Data columns (total 12 columns):
User_ID                     537577 non-null int64
Product_ID                  537577 non-null object
Gender                      537577 non-null object
Age                         537577 non-null object
Occupation                  537577 non-null int64
City_Category               537577 non-null object
Stay_In_Current_City_Years  537577 non-null object
Marital_Status              537577 non-null int64
Product_Category_1          537577 non-null int64
Product_Category_2          370591 non-null float64
Product_Category_3          164278 non-null float64
Purchase                    537577 non-null int64
dtypes: float64(2), int64(5), object(5)
memory usage: 49.2+ MB
```

Data

Variable	Definition
User_ID	User ID
Product_ID	Product ID
Gender	Sex of User
Age	Age in bins
Occupation	Occupation (Masked)
City_Category	Category of the City (A,B,C)
Stay_In_Current_City_Years	Number of years stay in current city
Marital_Status	Marital Status
Product_Category_1	Product Category (Masked)
Product_Category_2	Product may belongs to other category also (Masked)
Product_Category_3	Product may belongs to other category also (Masked)
Purchase	Purchase Amount (Target Variable)

```
In [21]: blackfriday.describe()
```

Out[21]:

	User_ID	Occupation	Marital_Status	Product_Category_1	Product_Category_2	Product_Category_3	Purchase
count	5.375770e+05	537577.00000	537577.000000	537577.000000	370591.000000	164278.000000	537577.000000
mean	1.002992e+06	8.08271	0.408797	5.295546	9.842144	12.669840	9333.859853
std	1.714393e+03	6.52412	0.491612	3.750701	5.087259	4.124341	4981.022133
min	1.000001e+06	0.00000	0.000000	1.000000	2.000000	3.000000	185.000000
25%	1.001495e+06	2.00000	0.000000	1.000000	5.000000	9.000000	5866.000000
50%	1.003031e+06	7.00000	0.000000	5.000000	9.000000	14.000000	8062.000000
75%	1.004417e+06	14.00000	1.000000	8.000000	15.000000	16.000000	12073.000000
max	1.006040e+06	20.00000	1.000000	18.000000	18.000000	18.000000	23961.000000

3- Data Understanding and Exploration with Tableau:

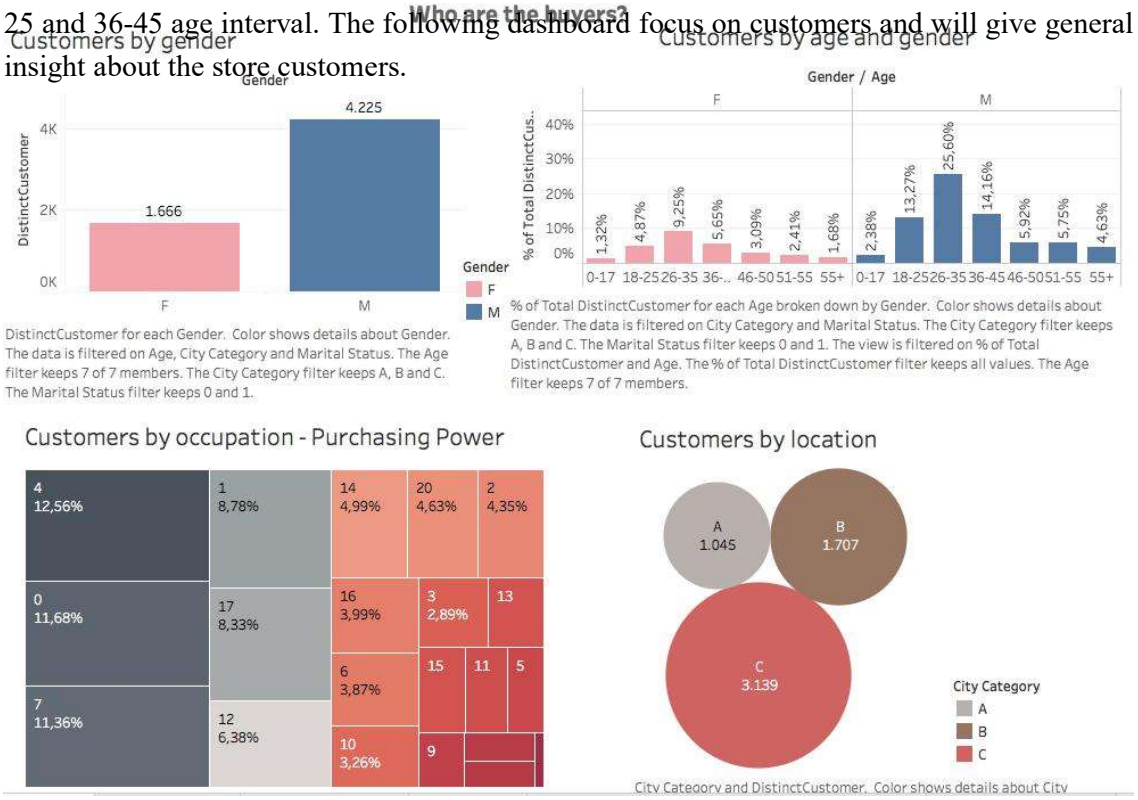
Who are my Customers?

"Number of Customers and Demographic Analysis"

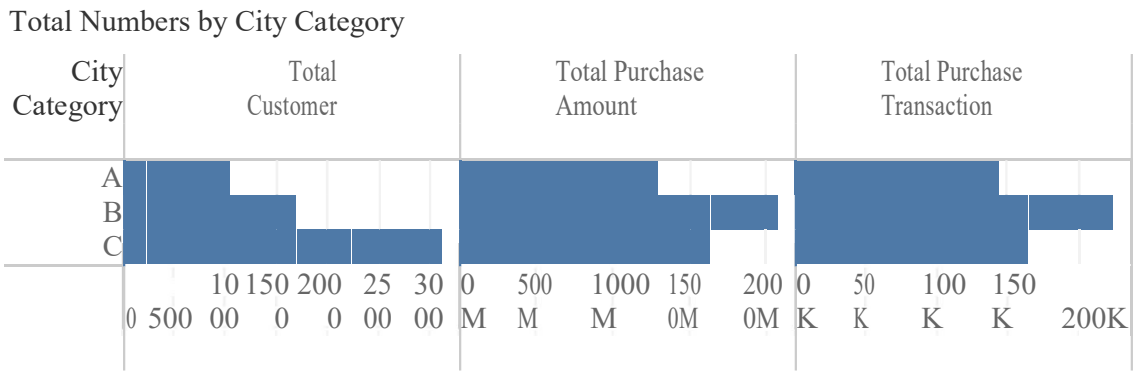
Gender	Marital Status	Age 0-17	18-25	26-35	36-45	46-50	51-55	55+	DistinctCustomer
F	0	78	217	320	202	49	49	32	32 , 924
	1		70	225	131	133	93	67	
M	0	140	608	924	503	107	87	101	
	1		174	584	331	242	252	172	

DistinctCustomer (color) broken down by Age vs. Gender and Marital Status. The data is filtered on City Category, which keeps A, B and C. The view is filtered on Marital Status and Age. The Marital Status filter keeps 0 and 1. The Age filter keeps 7 of 7 members.

This table shows count of the customers who has transaction in black Friday. Retail store has dominated by customers who are male and unmarried. Store has more customer between 18-25 and 36-45 age interval. The following dashboard focus on customers and will give general insight about the store customers.



Although the store customer data is well defined and does not have null value, some of the features are not labelled such as, occupation and location. In detail analysis, the occupation gives an idea about customer's purchase power. It's clear from the analysis that the store has more younger age male customers and they have occupation 4,0 and 7.



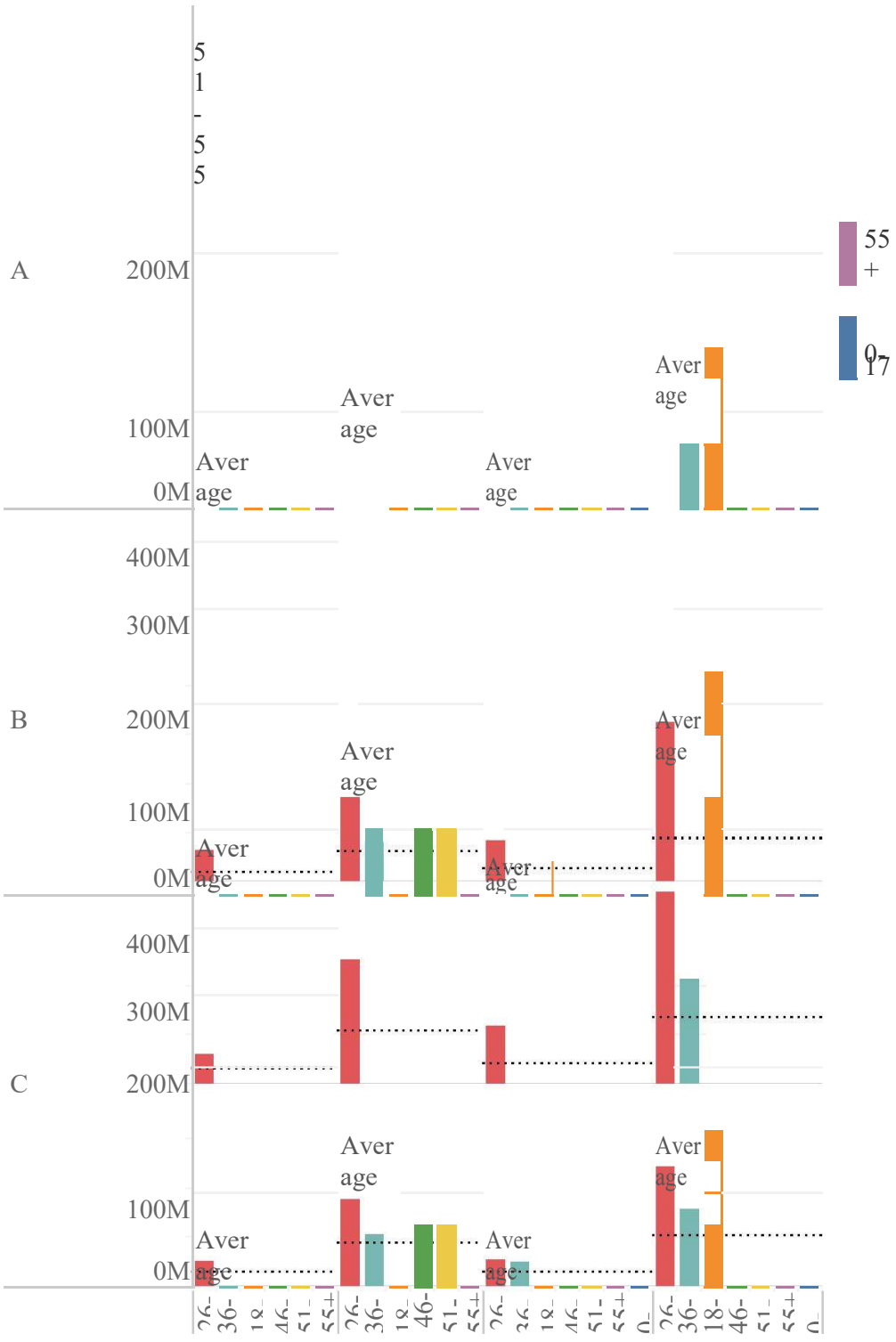
Total Customer, Total Purchase Amount and Total Purchase Transaction for each City Category.

Location might be another factor on sales volume. Firstly, examining the total volumes and numbers could be good for the deeper sales analysis. This graph only shows the total numbers of the store broken by city category.

When looking at the figures, although number of customers in city C are more than the city B, the purchase amount is differentiating. It seems city B gives more profit to the store.

Detail analysis for sales figures might give more idea about the customer purchase behavior.

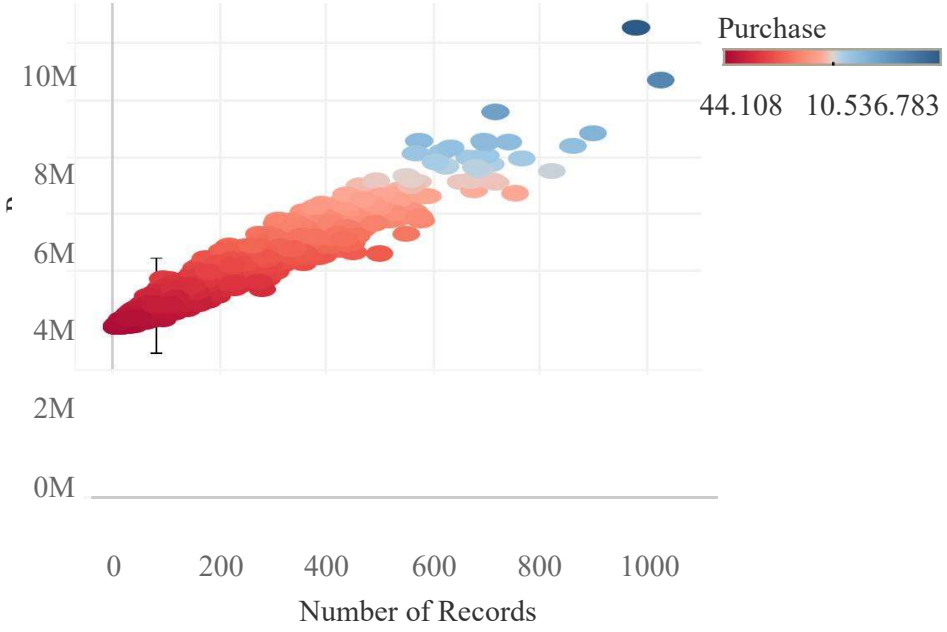
Who are the customers above the average spending? Who should be the target customers?

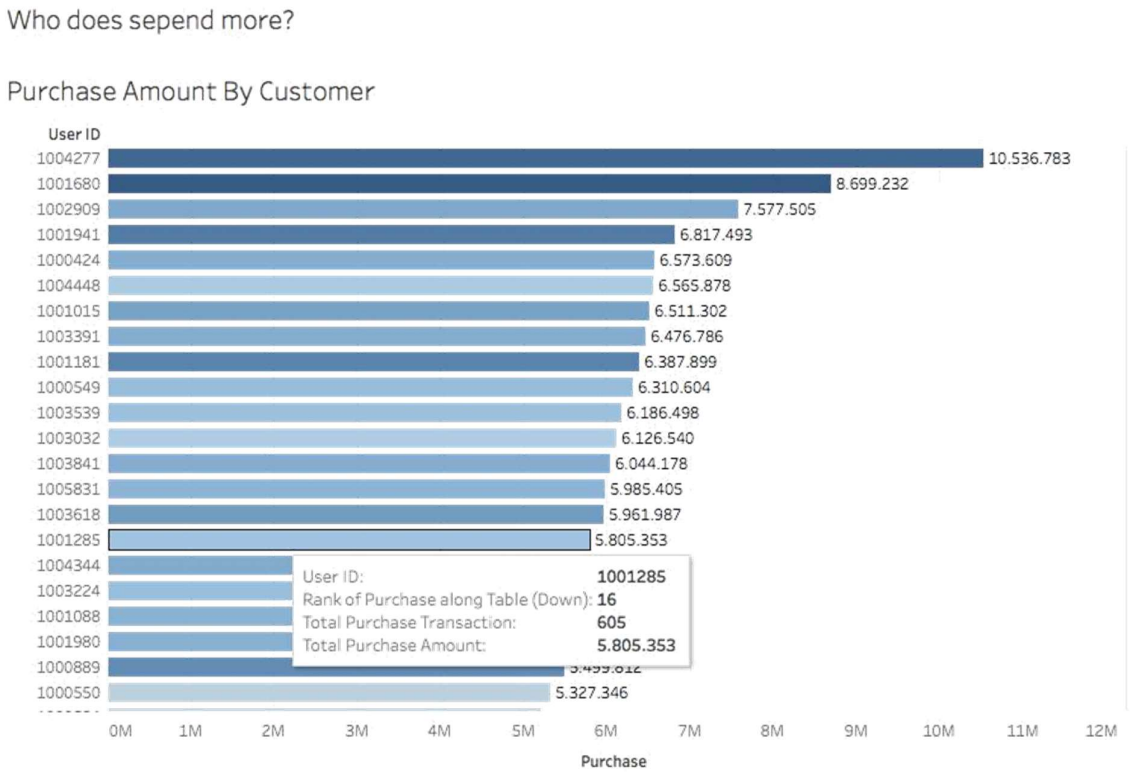

A
B
C
200M
100M
0M
400M
300M
Average
51-55
55+
0-17

This graph shows that, even male purchases are above the average, marital status has major effect on male customer's purchase. Who are these customers? With the scatter plot as below which could give better sense about the customer distribution, some of the customers has higher amounts on spending by age. Identifying these customers could be important for store. In that point, top customer analysis could be helpful.

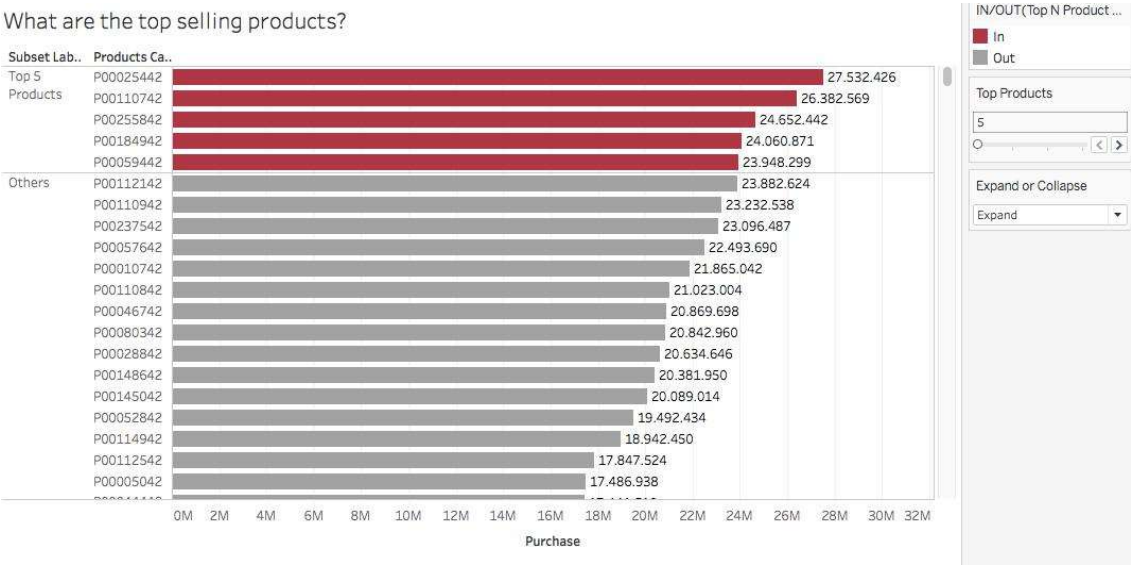
How does age effects Purchase?

Purchase Density by Age



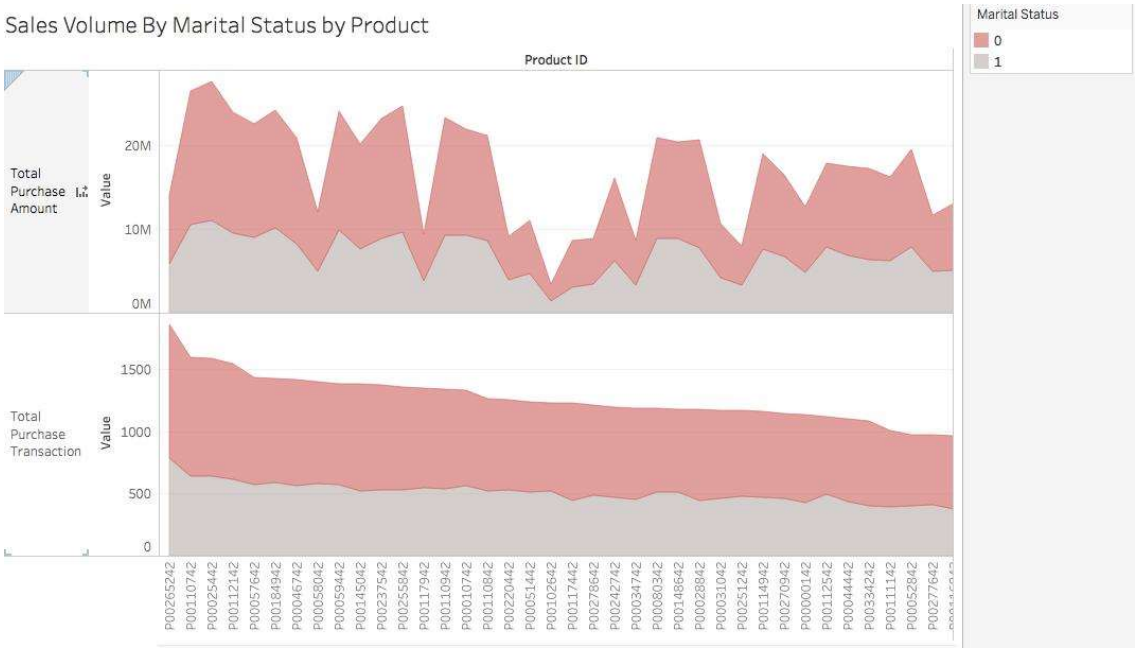


After examining the customer, it is time to get insight about the products. There are 3623 distinct products in this data set. Product categories has null values and not specific labeled which makes product analysis bit difficult. Top N analysis on product useful for finding of which product is most selling product?

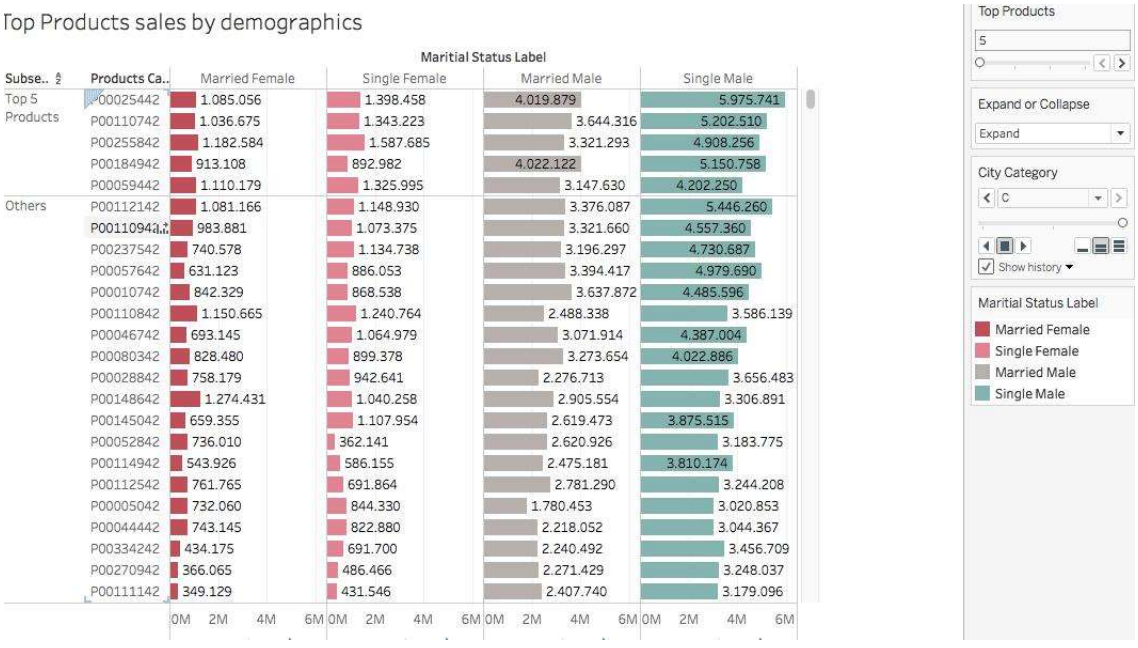


After examining the top products, the follow up question might be, what the relationships between this products and demographics are.

Former exploration gave idea about the marital status that has more effect on sales. Therefore, investigating the marital status and product relationship could be give a good insight.



This graph illustrates sales volumes broken down by marital status. Because there are many product, focusing on top N product and looking at products level of gender and marital status might give an idea about who are buying the top products.



This graph shows the distribution of the top selling 5 product broken down by marital status and gender and shows the differentiation between the city categories. Domination of male customers can be seen clearly in this graph. Male customers have also different behavior with regards to marital status.

Data Preparation, Modelling, Evaluation:

Data Preparation:

Firstly, after exploration of the data, missing values on product category 2 and product category 3 may cause a problem for modelling stage. Therefore, filling missing values is essential. Now we filled empty spaces with max value.

```
In [1]: import os
        import pandas as pd
        import numpy as np
        import seaborn as sns
        import matplotlib.pyplot as plt
        #To visualize the whole grid
        pd.options.display.max_columns = 999

In [2]: blackfriday=pd.read_csv("/Users/bilge/Downloads/BlackFriday.csv")

In [4]: blackfriday.isnull().sum()

Out[4]: User_ID                          0
        Product_ID                       0
        Gender                           0
        Age                              0
        Occupation                       0
        City_Category                    0
        Stay_In_Current_City_Years       0
        Marital_Status                   0
        Product_Category_1               0
        Product_Category_2          166986
        Product_Category_3          373299
        Purchase                         0
        dtype: int64

In [5]: #here we are making an array consisting of 2 columns namely Product_Category_2,Product_Category_3
        b = ['Product_Category_2','Product_Category_3']

In [7]: for i in b:
            exec("blackfriday.%s.fillna(blackfriday.%s.value_counts().idxmax(), inplace=True)" %(i,i))
```

Secondly, converting the categorical variables to the numerical variables would be easier for pandas library. Instead of ‘Male’ using 0’, ‘Female’ using 1 would be good for the algorithms or computations.

```
In [8]: from sklearn.preprocessing import LabelEncoder #import encoder from sklearn library
        LE = LabelEncoder()
        #Now we will encode the data into labels using label encoder for easy computing

In [11]: X = X.apply(LE.fit_transform)

In [12]: X.Gender = pd.to_numeric(X.Gender)
         X.Age = pd.to_numeric(X.Age)
         X.Occupation = pd.to_numeric(X.Occupation)
         X.City_Category = pd.to_numeric(X.City_Category)
         X.Stay_In_Current_City_Years = pd.to_numeric(X.Stay_In_Current_City_Years)
         X.Marital_Status = pd.to_numeric(X.Marital_Status)
         X.Product_Category_1 = pd.to_numeric(X.Product_Category_1)
         X.Product_Category_2 = pd.to_numeric(X.Product_Category_2)
         X.Product_Category_3 = pd.to_numeric(X.Product_Category_3)
```

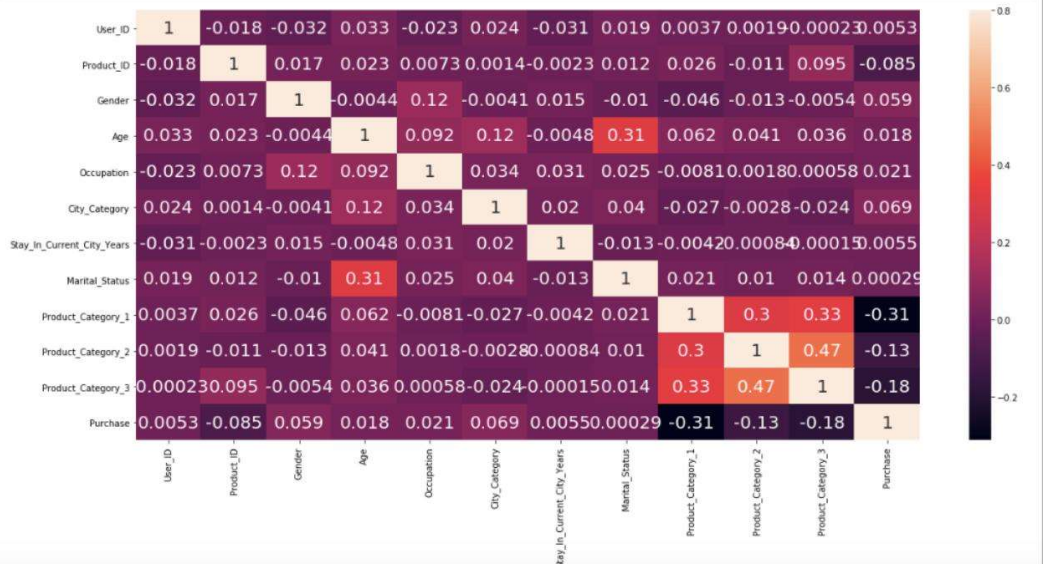
After converting categorical variables to numeric we can observe the correlation between the variables.

```
In [68]: corr = numeric_features.corr()
         print (corr['Purchase'].sort_values(ascending=False)[:12], '\n')
         print (corr['Purchase'].sort_values(ascending=False)[-12:])
```

```
Purchase                    1.000000
City_Category               0.068584
Gender                      0.059356
Occupation                  0.021112
Age                         0.018251
Stay_In_Current_City_Years  0.005527
User_ID                     0.005286
Marital_Status              0.000295
Product_ID                 -0.084920
Product_Category_2         -0.134950
Product_Category_3         -0.184587
Product_Category_1         -0.313649
Name: Purchase, dtype: float64

Purchase                    1.000000
City_Category               0.068584
Gender                      0.059356
Occupation                  0.021112
Age                         0.018251
Stay_In_Current_City_Years  0.005527
User_ID                     0.005286
Marital_Status              0.000295
Product_ID                 -0.084920
Product_Category_2         -0.134950
Product_Category_3         -0.184587
Product_Category_1         -0.313649
Name: Purchase, dtype: float64
```

```
In [25]: #correlation matrix
         f, ax = plt.subplots(figsize=(20, 9))
         sns.heatmap(corr, vmax=.8,annot_kws={'size': 20}, annot=True);
```



As it can be seen from the matrix, there is negative correlation between purchase and product categories.

Modelling:

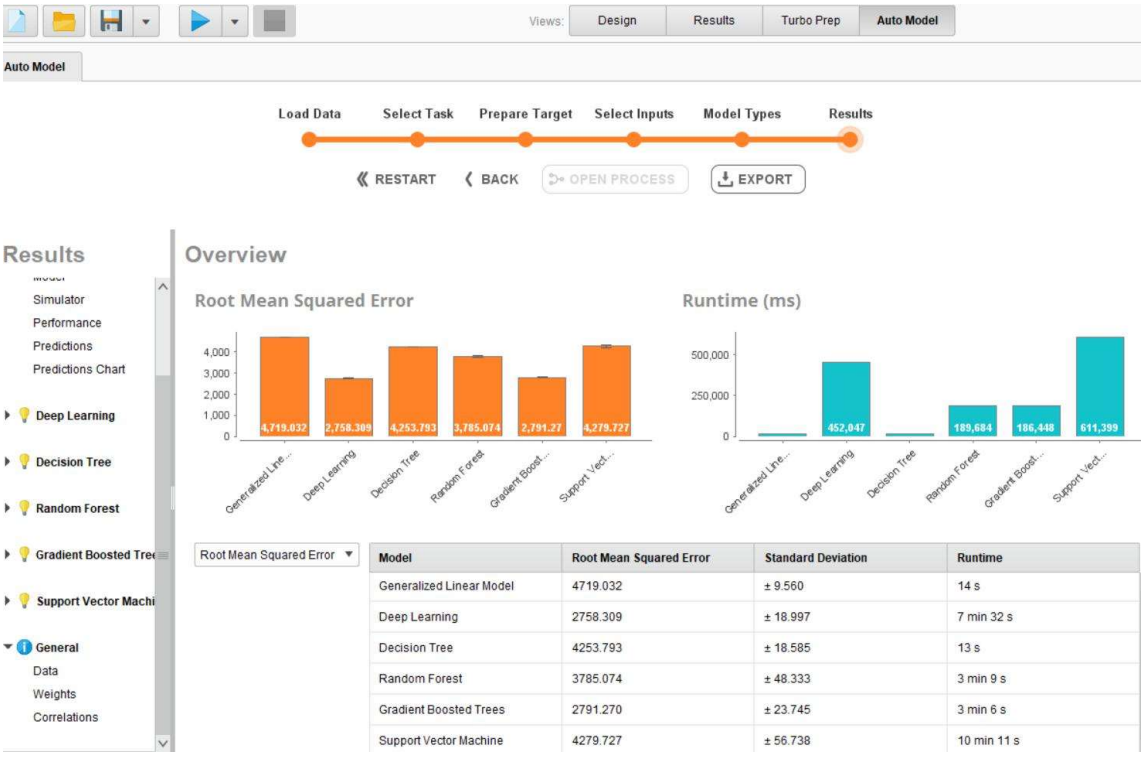
It is important to identify problem and select suitable methods for modelling stage. Algorithm selection and performance metric important for the beginning of the modelling stage. Here the data is labeled data because we already know customer spent on any product. The target variable is numeric which is purchase amount. As a

Result, we can say supervised learning and regression algorithms can be applied for this kind of problem. Usually for regression problems the typical performance measure is the Root Mean Square Error (RMSE). This function gives an idea of how much error the system makes in its predictions with higher weight for large errors.

Auto Model with RapidMiner and modelling with Python:

Before choosing the suitable model, it could be very useful to use rapidminer auto model properties. It gives idea about the models and their performances.

The result of the auto model shows Gradient Boosted Trees gives good accuracy and performance metrics. Although deep learning gave good accuracy, execution time of the model was not sufficient. As a conclusion, Decision tree, random forest, linear regression and Gradient Boosting is used for python modelling stage.



Model	Root Mean Squared Error	Standard Deviation	Runtime
Generalized Linear Model	4719.032	± 9.560	14 s
Deep Learning	2758.309	± 18.997	7 min 32 s
Decision Tree	4253.793	± 18.585	13 s
Random Forest	3785.074	± 48.333	3 min 9 s
Gradient Boosted Trees	2791.270	± 23.745	3 min 6 s
Support Vector Machine	4279.727	± 56.738	10 min 11 s

With python data is splitting to test and training set by using k-fold. After splitting the data to train and test set models are applied to the training set. After models fit the train set the score results are used for the test set.

Here, we have used K- Fold for splitting the data because k-fold minimizes variance applying average over k different fold (partitions). So the estimate performance of model becomes less sensitive to the partitioning of the data

PCA is used for the selection of the number of components as PCA is a technique for feature extraction. PCA combines input variable in a specific way later on dropping 'least significant' while retaining most valuable parts of all variables. Each of the new variable are all independent of one other which is beneficial because assumption of linear model require our independent variable should be independent of one another.
In our dataset there are 12 columns. We want to reduce the number of variables but aren't able to identify variables to completely remove from consideration. Hence we decided to go with PCA.

```
In [29]: from sklearn.model_selection import KFold
         kf = KFold(20)
         #Provides train/test indices to split data in train/test sets. Split dataset into k consecutive folds
         #Each fold is then used once as a validation while the k - 1 remaining folds form the training set.

In [30]: for a,b in kf.split(principalDf):
             X_train, X_test = Xs[a],Xs[b]
             y_train, y_test = Y[a],Y[b]

In [31]: from sklearn.linear_model import LinearRegression
         from sklearn.tree import DecisionTreeRegressor
         from sklearn.ensemble import RandomForestRegressor
         from sklearn.ensemble import GradientBoostingRegressor
```

```
from sklearn.linear_model import LinearRegression
from sklearn.tree import DecisionTreeRegressor
from sklearn.ensemble import RandomForestRegressor
from sklearn.ensemble import GradientBoostingRegressor
```

```
lr = LinearRegression()
dtr = DecisionTreeRegressor()
rfr = RandomForestRegressor()
gbr = GradientBoostingRegressor()
```

```
fit1 = lr.fit(X_train,y_train)#Here we fit training data to linear regressor
fit2 = dtr.fit(X_train,y_train)#Here we fit training data to Decision Tree Regressor
fit3 = rfr.fit(X_train,y_train)#Here we fit training data to Random Forest Regressor
fit4 = gbr.fit(X_train,y_train)#Here we fit training data to Gradient Boosting Regressor
```

Model Evaluation and Results:

```
In [34]: print("Accuracy Score of Linear regression on train set",fit1.score(X_train,y_train)*100)
         print("Accuracy Score of Decision Tree on train set",fit2.score(X_train,y_train)*100)
         print("Accuracy Score of Random Forests on train set",fit3.score(X_train,y_train)*100)
         print("Accuracy Score of Gradient Boosting on train set",fit4.score(X_train,y_train)*100)

Accuracy Score of Linear regression on train set 26.253751627358746
Accuracy Score of Decision Tree on train set 95.96374999341674
Accuracy Score of Random Forests on train set 91.15995471072081
Accuracy Score of Gradient Boosting on train set 67.57835499055898

In [35]: print("Accuracy Score of Linear regression on test set",fit1.score(X_test,y_test)*100)
         print("Accuracy Score of Decision Tree on test set",fit2.score(X_test,y_test)*100)
         print("Accuracy Score of Random Forests on test set",fit3.score(X_test,y_test)*100)
         print("Accuracy Score of Gradient Boosting on testset",fit4.score(X_test,y_test)*100)

Accuracy Score of Linear regression on test set 27.29085156153851
Accuracy Score of Decision Tree on test set 43.66549391861273
Accuracy Score of Random Forests on test set 63.02687901755775
Accuracy Score of Gradient Boosting on testset 66.65426409069882
```

As seen from the result, linear regression model gave very low accuracy result in both training and test data set. Although Decision tree gave high accuracy on training data set, it performed very less on test data. When the model performs well on training dataset and performs bad on test data, is called overfitting. Here is the decision tree memorized the train set therefore it gave very less accuracy for unseen data. Same situation, overfitting occurs for Random Forest model.

Gradient Boosting gave accurate and stable results for both test and train set. As a conclusion, we chose Gradient Boosting as a final model for deployment stage.

If more features would be added to dataset such as unit prices, date time data, etc., the model accuracy might be increased. Provided that detailed product tree also could be helpful for prediction of the customer purchase.

Deployment and Insights:

From the previous modelling stage results, Gradient Boosting gives stable result on both train and test data, this model has been chosen for deployment.

For further and detailed analysis, time based transactional data for calculation of the loyal customers and churn analysis is required for the calculations.

Market basket analysis could also be applied for this data set but detailed product tree should be provided for analysis. Hourly purchase transactions are needed for determination of the trends and might give insights about possible human resource management.

Patterns in this data shows that behavior for customers over the age of 45 shifting dramatically. These results may be useful for planning and making marketing decisions.

Customers dominated by male, may be making some marketing campaigns could be an idea for increasing female customers.

Location level analysis shows purchase amount changing based on city. Product stock analysis may be done based on this information.

Bibliography/References:

1- Kaggle (2019) 'Black Friday A study of sales through customer behaviours', Kaggle, 2019 [Online] Available at: https://www.kaggle.com/mehdidag/black-friday (Accessed: 4 April 2019)

2- Black Friday(2019) 'How much will customer spend', Medium, 2019[Online] Available at: https://medium.com/diogo-menezes-borges/project-3-analytics-vidhya-hackaton-black-friday-f6c6bf3da86f (Accessed: 4 April 2019)

3- DataCamp(2019) 'Machine Learning Black Friday Dataset', datacamp, 2019[Online] Available at : https://www.datacamp.com/community/tutorials/ml-black-friday-dataset Accessed (12 April 2019)