International Journal of Science and Research (IJSR)
ISSN: 2319-7064
SJIF (2022): 7.942

A Mathematical Viewpoint on Regression Modelling of Big Data Sales Analysis using Python

Dr. Vivek Parkash

Assistant Professor of Mathematics, Dyal Singh College, Karnal - 132001 (Haryana), India
Email: *lethal007[at]hotmail.com*

Abstract: ***Mathematics is an integral part of data science. Data science utilizes mathematical background, as the machine learning models and algorithms require mathematics to get valuable insights from data. A consistent analysis of the sales is very crucial for the growth of any business. A detailed analysis of the sales, revenue and performance of a business helps it to unearth new opportunities and locate the problem areas of the business and therefore bringing new dimensions of opportunities leading to multidimensional growth prospects of the business. In this paper, I wish to analyze a case of big data analysis - the Big Mart sales analysis using Decision Tree,*** *XG Boost,* ***Linear and Random Forest machine learning Regression models and find out the correlation between various factors reflecting and influencing the sales and revenue of the business.***

Keywords: Economics, Revenue, Regression, Label Encoder, Visibility, BigMart, Attributes

1. Introduction and works cited

In this paper, I propose to analyze the sales data of numerous establishments of Big Mart since its inception in 1985, using regression models in Python, Pandas, Linear Regression, Label Encoder, sea born and Matplotlib. The imported data contains sales record of various items as dairy, soft drinks, meat, fruits and vegetables, households, baking soda, snacks foods, frozen foods, hard drinks, breads, health and hygiene, baking goods and canned in Tier1, Tier2 and Tier3 supermarkets of BigMart. Julian Vasilev and Maria Kehajova (2017) have done the sales analysis using Rectangle method. Myint Myint Yee (2018) has given a model on Improving Sales Analysis in Retail Sale using Data Mining Algorithm with Divide and Conquer Method. J. Eardley-Simpson (1974) developed a model to analyze the sales with the aim to enable marketing people to analyze information available to them. Aditi Chaudhary (2022) proposed a study on analysing the impact of marketing on the sales performance of the company. Shridhar Mashalkar (2022) emphasized on the use of Data Modelling, Management and Automation in Salesforce. Nayana R, Chaithanya G, Meghana T, Narahari K S, Sushma M (2022) designed a Predictive Analysis for Big Mart Sales using Machine Learning Algorithms. Anurag Bejju (2016) has discussed the Sales Analysis of E - Commerce Websites using Data Mining Techniques. Kiran Singh and Rakhi Wajgi discussed Data analysis and virtualization of sales data of shopping websites.

2. Methodology

Importing libraries and data
First we need to import various libraries like pandas, Linear Regression, Numpy, Label Encoder, matplotlib and sea born etc. (Figure - 1).

The. csv data file is downloaded from Kaggle and pandas are used to read the data frame.

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

```
df1 = train_mart_data = pd.read_csv('train.csv')
df2 = test_mart_data = pd.read_csv('test.csv')
```

Figure - 1: Import various libraries and data sets

After reading the train data frame, we see that the train data of big mart sales data consists of training data 8523 rows and 12 columns whereas the test mart data consists of 5681 rows and 11 columns. Various attributes of the mart data with their non - null count are shown below (Figure - 2, 3, 4).

```
train_mart_data.head(400)
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size
0	FDA15	9.30	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium
1	DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium
2	FDN15	17.50	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium
3	FDX07	19.20	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN
4	NCD19	8.93	Low Fat	0.000000	Household	53.8614	OUT013	1987	High
...	...	...	...	...	...	...	...	...	...
395	FDQ56	6.59	Low Fat	0.105761	Fruits and Vegetables	84.8908	OUT049	1999	Medium

Figure 2: Train Mart data

```
test_mart_data.head()
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size
0	FDW58	20.750000	Low Fat	0.007565	Snack Foods	107.8622	OUT049	1999	Medium
1	FDW14	8.300000	reg	0.038428	Dairy	87.3198	OUT017	2007	Medium
2	NCN55	14.600000	Low Fat	0.099575	Others	241.7538	OUT010	1998	Medium
3	FDQ58	7.315000	Low Fat	0.015388	Snack Foods	155.0340	OUT017	2007	Medium
4	FDY38	12.857645	Regular	0.118599	Dairy	234.2300	OUT027	1985	Medium

Figure 3: Test Mart data

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8523 entries, 0 to 8522
Data columns (total 12 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   Item_Identifier            8523 non-null   object
 1   Item_Weight                7060 non-null   float64
 2   Item_Fat_Content           8523 non-null   object
 3   Item_Visibility            8523 non-null   float64
 4   Item_Type                  8523 non-null   object
 5   Item_MRP                   8523 non-null   float64
 6   Outlet_Identifier          8523 non-null   object
 7   Outlet_Establishment_Year  8523 non-null   int64
 8   Outlet_Size                6113 non-null   object
 9   Outlet_Location_Type       8523 non-null   object
 10  Outlet_Type                8523 non-null   object
 11  Item_Outlet_Sales          8523 non-null   float64
dtypes: float64(4), int64(1), object(7)
memory usage: 799.2+ KB
```

Figure 4: Various attributes in the Mart data

Data Cleaning Process

Datasets that we imported for developing economic models a re equipped with a number of missing values of attributes. T o locate the missing values in train mart data, we use *isnull* a nd *sum* function and analyze that the two columns, represent ed by attributes *Item_Weight* and *Outlet_Size*, we find that th ese two attributes have one thousand four hundred sixty three and two thousand four hundred ten missing values respective ly (Figure - 5). Similarly applying the same two functions for locating missing values in test mart data, we notice that *Item _Weight* and *Outlet_Size* have respectively nine hundred sev enty - six and one thousand six hundred six missing values (F igure - 6).

```
Item_Identifier                 0
Item_Weight                  1463
Item_Fat_Content                0
Item_Visibility                 0
Item_Type                       0
Item_MRP                        0
Outlet_Identifier               0
Outlet_Establishment_Year       0
Outlet_Size                  2410
Outlet_Location_Type            0
Outlet_Type                     0
Item_Outlet_Sales               0
dtype: int64
```

Figure 5: Attributes showing missing train data

```
Item_Identifier                 0
Item_Weight                   976
Item_Fat_Content                0
Item_Visibility                 0
Item_Type                       0
Item_MRP                        0
Outlet_Identifier               0
Outlet_Establishment_Year       0
Outlet_Size                  1606
Outlet_Location_Type            0
Outlet_Type                     0
dtype: int64
```

Figure 6: Attributes showing missing values in test mart dat a

Now, we assign 0 to these missing values using mean and mode function. The data having attributes assigned 0 is shown in Figure - 7.

```
Item_Identifier              0
Item_Weight                  0
Item_Fat_Content             0
Item_Visibility              0
Item_Type                    0
Item_MRP                     0
Outlet_Identifier            0
Outlet_Establishment_Year    0
Outlet_Size                  0
Outlet_Location_Type         0
Outlet_Type                  0
Item_Outlet_Sales            0
dtype: int64
```

```
df2.isnull().sum()
```

```
Item_Identifier              0
Item_Weight                  0
Item_Fat_Content             0
Item_Visibility              0
Item_Type                    0
Item_MRP                     0
Outlet_Identifier            0
Outlet_Establishment_Year    0
Outlet_Size                  0
Outlet_Location_Type         0
Outlet_Type                  0
```

Figure 7: Assigning 0 to the missing values

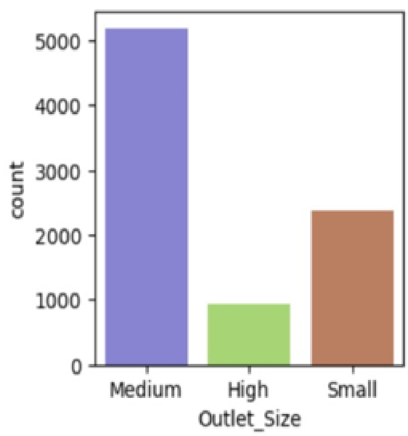

Figure 8: Showing number of different outlet

Paper ID: SR23109085518 DOI: https://dx.doi.org/10.21275/SR23109085518

Let us have a look at the relationship between the attribute *outlet_size* and their count. Figure - 8 reveals that the Big Mart has the highest number of medium sized outlets spread across different regions and the outlets of high size are least in number. There are around 5000 plus outlets of medium size whereas the number of high sized outlets is around 1000. Figure - 9 shows year wise percentage of the outlet establishments. From this pie chart, it is visible that BigMart started its journey in 1985 initially having 17% of the total outlets. The Highest number of the outlets were opened in 1985 itself and the least being 7% in 1998. In other years, the percentage of outlets opened have remained uniform. It also follows from the pie chart that BigMart has continuously grown in popularity and amid high demand of its products; BigMart has opened new outlets at uniform pace.

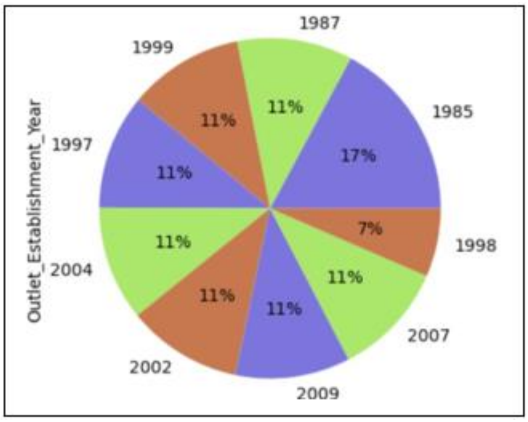

Figure 9: Year wise percentage

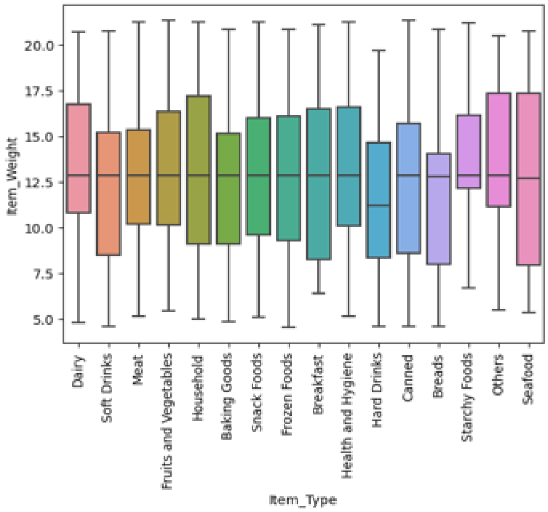

Figure 10: Item weight vs Item Type

Next analysis given in Figure - 10 shows the comparison between the attributes *Item_type* and *Item_weight*. The plot of the weights of various attributes reflects that the weights of all the items lies in the range 9.0 to 17.5. The length of the bar shows that the item named seafood has the highest weight and the item starchy foods has the least weight.

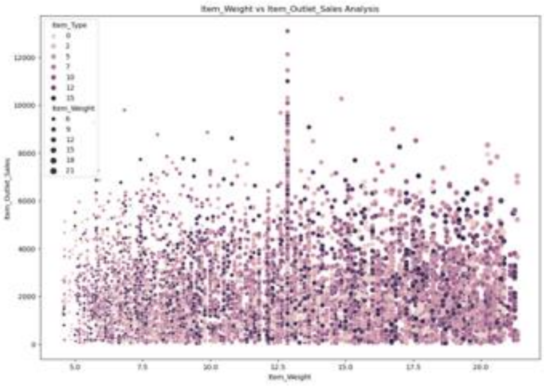

Figure 11: Item weight vs Item sales

The scatter diagram shown in Figure - 11, 12 reflects the relation between the attributes *Item_weight* vs. *item sales* and Item visibility vs. maximum retail price.

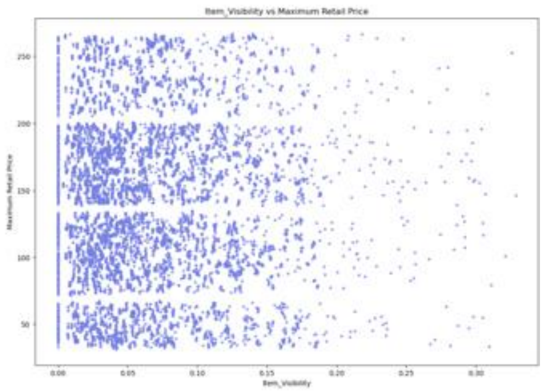

Figure 12: Item visibility and maximum retail price.

By looking at the figure, we can say that low fat content item is having the lowest sales as compared to the regular item fat content which is having maximum sales.

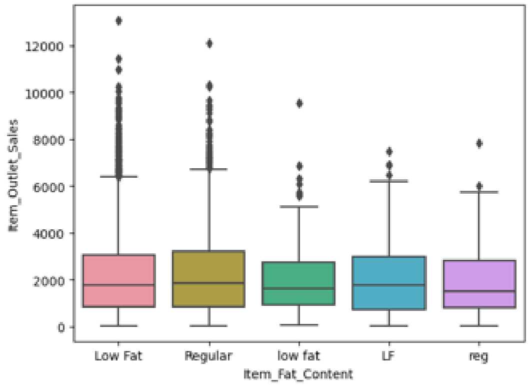

Figure 13: Item fat content vs Item Outlet sales

Paper ID: SR23109085518 DOI: https://dx.doi.org/10.21275/SR23109085518

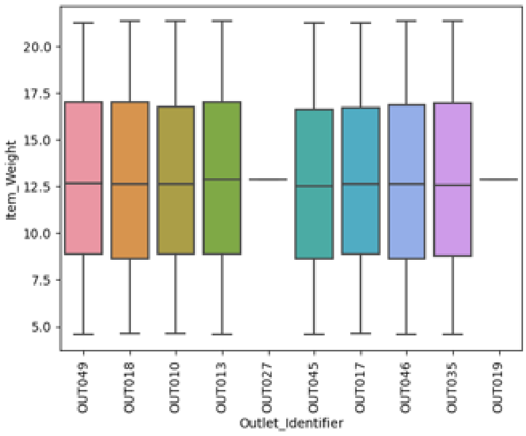

Figure 14: Item Identifiers vs Item Weight

Figure - 14 describes the comparison between *outlet identifier* and *item weight*. It reveals that the weights of outlet identifiers are almost identical.

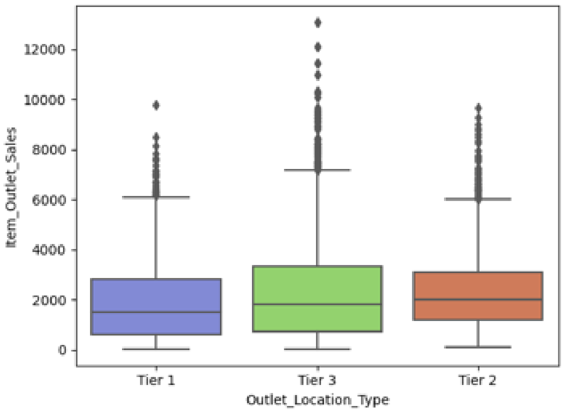

Figure 15: Outlet location type vs item outlet sales

The Figure - 15 shows the comparison between *Outlet type* and *Item Outlet sales*. The figure reveals that Tier 3 *outlet type* are having maximum sales where as Tier2 type outlets deliver minimum sales. This gives an idea that Tier 3 type outlets have major contribution to BigMart sales.

Figure - 16 depicted below shows the sales share of different items. The table data given in Figure - 16 reveals the item outlet sales for different items namely Dairy, soft drinks, meat, fruits and vegetables, households, baking goods, snacks foods, frozen foods, drinks, canned, breads, starchy foods, others, seafood etc. The plot shows that the item seafood has the least sale share of 1 percent among all items and the item snack foods along with fruits and vegetables has the highest sales share of 14 percent. The sales of all other items are more or less in the range 1 - 14%.

```
Fruits and Vegetables    1232
Snack Foods              1200
Household                 910
Frozen Foods              856
Dairy                     682
Canned                    649
Baking Goods              648
Health and Hygiene        520
Soft Drinks               445
Meat                      425
Breads                    251
Hard Drinks               214
Others                    169
Starchy Foods             148
Breakfast                 110
Seafood                    64
Name: Item_Type, dtype: int64
```

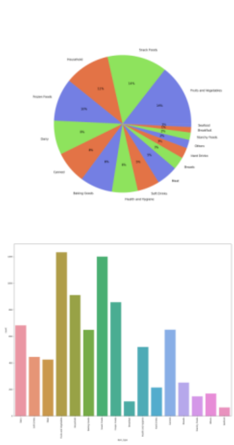

Figure 16: Sale shares of different items

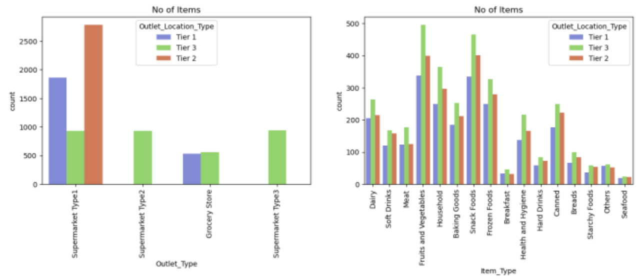

Figure 17: Sales of outlet type and item type

Figure - 17 displays the *outlet type* vs. their count and *item type* vs. their count. From the figure, we conclude that supermarket type 1 is located in tier1, tier2 and tier3 locations, supermarket type 2 is only at tier3 location, supermarket type 3 at tier3 location and grocery store at tier1 and tier3 locations. Also we get an idea that supermarket type 1 has maximum number of outlets in tier2 location whereas least number of outlets at tier3 locations. Moreover, BigMart has higher number of supermarket type1 outlets. Figure - 18 also shows that the item Fruits and vegetables have the maximum food count among all items and this comes on the back of tier3 locations the most. Second to fruits and vegetables is the item snack food. Similar to the fruits and vegetables, the maximum sales of the snack food come from tier 3 locations. We also conclude from Figure - 17 that tier1 locations are the least contributors to the BigMart sales and tier3 locations are the biggest contributors.

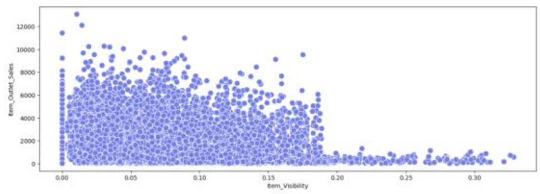

Figure 18: Item visibility vs item outlet sales

Figure - 18 displays the relationship between *item visibility* and the item *outlet sales*. The scatter diagram reflects that the items having low visibility are the largest contributors to the outlet sales whereas the sales of items having greater visibility declines dramatically. The figure tells that the items having visibility in the range 0 - 0.18 are preferred the most.

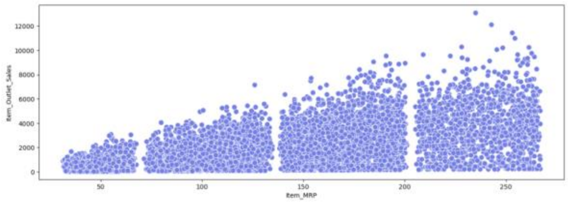

Figure 19: Item MRP vs Item outlet sales

Figure - 19 shows the relationship between *item MRP* and *item outlet sales*. The scatter plot data tells that the buyers prefer buying those items, which have MRP in the range 165 - 250, and such items are the largest contributors to the outlet sales. The low - ticket items are not that much preferred by the consumers.

Now we come to the modeling part and the results concluded by different models used for calculation and analysis of data. Four regression models have been used namely Decision Tree Regression, Linear Regression, XG Boost Regression and Random Forest Regresson. Figures - 20, 21, 22, 23, 24 and Table 01 show the results calculated by all four - regression models and the accuracy rate of the regression models.

Comparison of Different Economic Models and Results:

a) Decision Tree – Regression

```
from sklearn.tree import DecisionTreeRegressor

tree = DecisionTreeRegressor(max_depth=15, min_samples_leaf=100)

tree.fit(xvar_train,yvar_train)
tree_pred = tree.predict(xvar_test)

tree_pred

array([1385.21468855, 3751.63993554, 1459.84807152, ..., 3142.5178233 ,
       3363.12993231, 4170.64740602])

tree_accuracy = round(tree.score(xvar_train,yvar_train)*100)
tree_accuracy

62
```

Figure 20: Decision Tree Regression

b) Linear - Regression

```
lr_pred

array([2528.05383995, 3625.30926251, 1158.79233766, ..., 2111.72941652,
       4598.25557121, 3455.17062917])

lr_accuracy = round(lr.score(xvar_train,yvar_train)*100)
lr_accuracy

51
```

Figure 21: Linear Regression

c) XG Boost - Regression

```
yvar_pred = model.predict(xvar_test)
yvar_pred

array([2230.6816, 3892.4814, 1478.3667, ..., 2878.7039, 3031.0427,
       3835.7886], dtype=float32)

model.score(xvar_train, yvar_train)*100

68.90574487616622
```

Figure 22: XG Boost Regression

d) Random Forest - Regression

```
from sklearn.ensemble import RandomForestRegressor

rf = RandomForestRegressor(n_estimators=500,max_depth=8,min_samples_leaf=30,n_jobs=6)

rf.fit(xvar_train,yvar_train)

rf_accuracy = round(rf.score(xvar_train,yvar_train)*100)

rf_accuracy

64
```

Figure - 23: Random Forest Regression

Table 1: Accuracy rate in percentage terms

S. No.	Model	Accuracy Rate %
1	Decision Tree Regression	62
2	Linear Regression	51
3	XG Boost Regression	69
4	Random Forest Regression	64

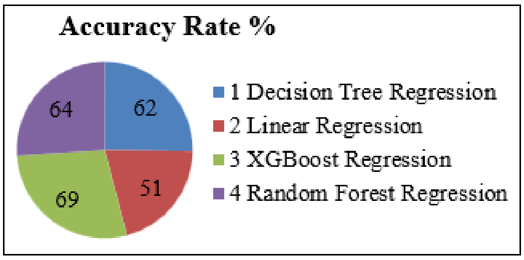

Figure 24: Pie chart showing accuracy rate percent

3. Conclusion

The accuracy percentages of Decision Tree Regression, Linear Regression, XG Boost Regression and Random Forest Regression are obtained in the above table. We conclude that XG Boost Regression model shows the maximum accuracy and hence predicts the data most accurately followed by Random Forest Regression model. So, the sales data has been successfully analyzed using the regression models.

References

[1] Vasilev, Julian & Kehajova, Maria. (2017). Sales analysis by the rectangle method. Leonardo Electronic Journal of Practices and Technologies.2017.149 - 160.
[2] Myint Myint Yee. (2018). Improving Sales Analysis in Retail Sale using Data Mining Algorithm with Divide and Conquer Method, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 07, Issue 07 (July 2018).
[3] Eardley-Simpson, J. (1974). "Sales analysis: a visual approach", European Journal of Marketing, Vol.8 No.1, pp.57 - 74. https: //doi. org/10.1108/EUM0000000005077.
[4] aditi choudhary (2022). A study on analysing the impact of marketing on the sales performance of the company. International Journal of Advance Research And Innovative Ideas In Education, 8 (2), 504 - 511.
[5] Shridhar Mashalkar, Vineeth R. (2022). Data Modelling, Management and Automation in Salesforce, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 11, Issue 09 (September 2022).
[6] Nayana R, Chaithanya G, Meghana T, Narahari K S, Sushma M. (2022). Predictive Analysis for Big Mart

Sales using Machine Learning Algorithms, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) RTCSIT – 2022 (Volume 10 – Issue 12).
[7] Bejju, Anurag. (2016). Sales Analysis of E - Commerce Websites using Data Mining Techniques. International Journal of Computer Applications.133.36 - 40.10.5120/ijca2016907812.
[8] K. Singh and R. Wajgi, (2016). "Data analysis and visualization of sales data, " World Conference on Futuristic Trends in Research and Innovation for Social Welfare (Startup Conclave), Coimbatore, India, 2016, pp.1 - 6, doi: 10.1109/STARTUP.2016.7583967.