**SALES PREDICTION AND ANALYSIS USING MACHINE LEARNING TECHNIQUES**

*Report submitted to the SASTRA Deemed to be University*

*as the requirement for the course*

### CSE300 / INT300 - MINI PROJECT

*Submitted by*

**Pinnama Reddy Shanmugha Reddy**

**(Reg. No.: 124015168, IT)**

**Kothuri Leela Balaji**

**(Reg. No.: 124003418,CSE)**

**Dornadula Venkata Sai Hemanth**

**(Reg. No.: 124003415, CSE)**

### MAY 2023



### SCHOOL OF COMPUTING

**THANJAVUR, TAMIL NADU, INDIA – 613 401**



**SCHOOL OF COMPUTING**

**THANJAVUR – 613 401**

# Bonafide Certificate

This is to certify that the report titled “**SALES PREDICTION AND ANALYSIS USING MACHINE LEARNING TECHNIQUES**” submitted as a requirement for the course, CSE300 / INT300: MINI PROJECT for B.Tech. is a bonafide record of the work done by **Mr. Pinnama Reddy Shanmugha Reddy (Reg. No.: 124015168, IT) , Mr. Kothuri Leela Balaji (Reg. No.: 124003418,CSE), Mr. Dornadula Venkata Sai Hemanth (Reg. No.: 124003415, CSE)** during the academic year 2022-23, in the School of Computing, under my supervision.

**Signature of Project Supervisor : **

**Name with Affiliation : Dr. Sujatha M** , Asst. Professor-II, SOC

**Date : 12/05/20223**

Mini Project *Viva voce* held on /05/2023

#### Examiner 1 Examiner 2

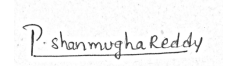


**SCHOOL OF COMPUTING**

**THANJAVUR – 613 401**

#### Declaration

We declare that the report titled " **SALES PREDICTION AND ANALYSIS USING MACHINE LEARNING TECHNIQUES** " submitted by us is an original work done by us under the guidance of **Dr.Sujatha M** , **Asst. Professor-II /SoC, School of COMPUTING, SASTRA** Deemed to be University during the sixth semester of the academic year 2022-23, in the School of Computing. The work is original and wherever we have used materials from other sources, we have given due credit and cited them in the text of the report. This report has not formed the basis for the award of any degree, diploma, associate-ship, fellowship or other similar title to any candidate of any University.

**Signature of the candidate: **

****



**Name of the candidate :** Pinnama Reddy Shanmugha Reddy (Reg. No.: 124015168, IT)

Kothuri Leela Balaji (Reg. No.: 124003418,CSE)

Dornadula Venkata Sai Hemanth (Reg. No.: 124003415, CSE)

**Date :** 12/05/2023

**Acknowledgements**

We would like to thank our Honorable Chancellor **Prof. R. Sethuraman** for providing us with an opportunity and the necessary infrastructure for carrying out this project as a part of our curriculum.

We would like to thank our Honorable Vice-Chancellor **Dr. S. Vaidhyasubramaniam** and **Dr. S. Swaminathan,** Dean**,** Planning & Development, for the encouragement and strategic support at every step of our college life.

We extend our sincere thanks to **Dr. R. Chandramouli,** Registrar, SASTRA Deemed to be University for providing the opportunity to pursue this project.

We extend our heartfelt thanks to **Dr. A. Umamakeswari**, Dean, School of Computing, **Dr. V. S. Shankar Sriram**, Associate Dean, Department of Computer

Science and Engineering and **Dr. R. Muthaiah**, Associate Dean, Department of Information Technology and Information & Communication Technology for their motivation and support offered in materializing this project.

Our guide **Dr. Sujatha M** , Asst. Professor-II, School of Computing was the driving force behind this whole idea from the start. His deep insight in the field and invaluable suggestions helped us in making progress throughout our project work. We also thank the project review panel members for their valuable comments and insights which made this project better.

We would like to extend our gratitude to all the teaching and non-teaching faculties of the School of Computing who have either directly or indirectly helped us in the completion of the project.

We gratefully acknowledge all the contributions and encouragement from our family and friends resulting in the successful completion of this project.

# 

**List of Figures**

|  |  |  |
| --- | --- | --- |
| **Figure. No.** | **Title** | **Page. No.** |
| 3.1 | Work flow | 5 |
| 5.1.1 | Distribution Plot for Item Outlet Sales | 17 |
| 5.1.2 | Count Plot for Item Type | 17 |
| 5.1.3 | Categorical Plot Between Item Fat Content & Outlet sales | 18 |
| 5.1.4 | Scatter Plot Between Item MRP & Outlet sales | 18 |
| 5.2.1 | Feature Selection Graph for linear regression | 19 |
| 5.2.2 | Comparision of Actual & Predicted Values for linear regression | 19 |
| 5.3.1 | Feature Selection Graph for XGBoost regression | 20 |
| 5.3.2 | Comparision of Actual & Predicted Values for XGBoost regression | 20 |
| 5.4.1 | Feature Selection Graph for Ridge regression | 21 |
| 5.4.1 | Comparision of Actual & Predicted Values for Ridge regression | 21 |
| 5.5.1 | Feature Selection Graph for Random Forest regression | 22 |
| 5.5.2 | Comparision of Actual & Predicted Values for Random Forest regression | 22 |
| 5.6.1 | Feature Selection Graph for lasso regression | 23 |
| 5.6.2 | Comparision of Actual & Predicted Values for Random Lasso regression | 23 |
| 5.7.1 | Home Page | 24 |
| 5.7.2 | Data Analysis | 24 |
| 5.7.3 | Prediction with different algorithms | 25 |
| 5.7.4 | Random Forest Regression | 25 |
| 5.7.5 | Big Mart Sales Prediction | 26 |
| 5.7.6 | Result & Data Analysis Report | 26 |

**List of Tables**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Title** | **Page. No.** |
| 1.2 | Literature Survey | 3 |
| 3.1 | The metrics comparison for two machine learning models | 7 |

# 

**Abbreviations**

**XG Boost** Extreme gradient boost

**RSME** Root Squared Mean Error

# 

**Abstract**

Sales of different retailers for their respective products will vary over the time . Sometimes it may also result in huge loss for them. So to overcome this problem, analyze their historical data and status of current sales properly to predict the future sales. Sales forecast help businesses to make better decisions based on future revenue. Also it helps them to take necessary steps to increase their growth as the competition between various stores is increasing rapidly. It is vital to forecast future demand for each product at different stores for the customers because rivalry between various stores is intensifying quickly . Sales forecasting is essential for a business house to enable it to produce required quantity at right time. Predicting the sales manually gets increasingly difficult as the volume of product grows dramatically. Therefore to predict the sales various machine learning algorithms such as Linear regression, Random forest regression , XGBoost regression , Ridge regression are used in this project and comparing them to get best results. The data set will consists of various product attributes. Since various algorithms are being used efficiency and accuracy of sales prediction is improved compared to single model predictive approaches.

**KEYWORDS**: Linear regression, Random forest regression , XGBoost regression , Ridge regression

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S.No** | **Title** | **Page No.** |
| 1 | Acknowledgements | i |
| 2 | List of Figures | ii |
| 3 | List of Tables | iii |
| 4 | Abbrevations | iv |
| 5 | Abstract | v |
| 6 | Summary of the base paper | 1 |
| 7 | Merits and Demerits of the base paper | 4 |
| 8 | Methodology And Work Plan | 5 |
| 9 | Source Code | 8 |
| 10 | SnapShots | 17 |
| 11 | Conclusion And Future Plans | 27 |
| 12 | References | 28 |
| 13 | Appendix | 29 |

**CHAPTER 1**

# SUMMARY OF THE BASE PAPER

**Title:** Exploratory Data analysis and sales forecasting of bigmart dataset using supervised and ANN algorithms

**Journal:** Measurement : sensors

**Authors:** T.K. Thivakaran , M Ramesh

#### Published: 2022

**1.1 Introduction :**

#### Sales prediction is an essential task for business for planning their operations and to make decisions. Since huge data of sales is available businesses can make use of machine learning techniques to understand their sales and predict future sales. By analzying past sales businesses can get a valuable insights so that they can understand customer behaviour , trends of sales , and other factors that affect sales. They all can be used to make changes in our business strategies that leads to better profit. There exists many approaches for analyzing sales. Due to advancement in technology analyzing huge data also became easy.

To analyze sales Machine learning approach is used. Because it’s easy to implement and it produces accurate forecast results . The main benefit of Machine learning algorithms is that they can process vast amount of data where the humans can’t do it. We can assure Faster processing because machine learning algorithms can process data much faster than humans and allows to generate forecasts much quickly. It’s also easily adaptable to changing market conditions and new trends. So that we can understand and adjust the sales strategies without much lag.

For predicting the sales using machine learning we have followed a series of steps before applying to machine learning algorithms such as Data pre-processing, Data analysis , Train Test split .In most cases Data cleaning, data transformation and data reduction are involved In pre-processing . After that we apply this data to Machine learning algorithms . Here we have used Linear regression , Random Forest regression , XGBoost regression , Ridge regression , Lasso regression and Artificial neural networks. All these algorithms are trained to create a final model based on the data.

The Data processing starts with identifying null values in all attributes of data set. After identifying those null values, we will fill them either by median or mode according to attribute. Now the data is cleaned. After that we will generate various graphs on that data, so that we can analyse how each attribute is distributed. This completes data analysis part. After that we have make the data fit in to algorithms that is we have to convert all categorical attributes into numerical attributes. For that we used target encoding that replaces that attribute with the mean of the target attribute.

After that this data is applied to various algorithms and model is evaluated based on r2 score, mean absolute error & mean squared error. Then we will compare the results of all models and will select the best model for our model with high accuracy. So in this way the best model is selected for our problem.

**1.2 literature Survey :**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **S.No** | **Title** | **Authors** | **year** | **Workdone** |
| **1** | Applied linear statistical Model | Kutner, Nachtsheim,neter | 2013 | To anticipate sales, researchers employed a generic linear strategy, a decision tree approach, and a good gradient approach. |
| **2** | Defining architecture components of the Big data ecosystem | Yuridemchenko ,ceesdelaat, peter membrey | 2014 | The linear regression method was organized into structured data. The next is to use machine learning techniques to model data for predictions, with a predicted accuracy of 84%. |
| **3** | The Data analysis process : 5 steps to better decision marketing | Sandeep Udmale, Vijay Sambhe | 2015 | Forecasted sales using linear regression and the XG booster algorithm, which included data gathering and translation into processed data. In the end, they were able to anticipate which model would yield the best results. |
| **4** | Forecasting of sales by using fusion of machine learning techniques | Mohit Gurnani,Yogesh Korke,Sunil Bhirud | 2017 | Composite models produce better results than individual models. Decomposition methods are considerably superior to hybrid mechanisms. |

**CHAPTER 2**

# MERITS AND DEMERITS OF THE BASE PAPER

**2.1 Merits :**

* Exploratory data analysis is applied on Bigmart data which is an essential step for understanding data . It’s also helped for selecting appropriate model. We are able to identify patterns and other anomalies because of this approach.
* The comparision of performance of different machine learning models enabled the businesses to choose appropriate model for their needs.
* The findings are relevant and applicable to business because they make use of dataset of real world from a retail business.
* The sales forecasting accuracy is improved than the traditional statistical models . Because they make use of decision trees , random forests , regression which are supervised machine learning algorithms.
* It provides practical insights in to the data preprocessing , feature selection forecasting sales and feature selection which are applications of machine learning.

**2.2 Demerits :**

* The findings may not be generalised to all other businesses who are having some different sales patterns because dataset is from a single retail business.
* There is no detailed comparision with statistical models or other approaches because it mainly focuses on different machine learning algorithms comparision.
* Privacy concerns or bias in the data is not present because it didn’t addressed ethical considerations of the machine learning algorithms.
* The accuracy of the sales forecast is limited because external factors are not considered for ex : competition or economy change
* Detailed explanation of machine learning algorithms is not present. The readers without technical background may found it difficult to grasp the methodology

**CHAPTER 3**

## METHODOLY AND WORKPLAN

#### 3.1 Project Phases

**3.1.1 DATA COLLECTION:**

In this project, to build the predictive model and find out the sales of each product we must construct the various machine learning models. The dataset includes total of 11 features in the dataset. This dataset was collected from Kaggle.

**3.1.2 DATA PREPROCESSING:**

Then the collected dataset was pre-processed by carrying out various techniques. In these

operations carried out is listed below:

1. Filling Missing values

2. Target Encoding

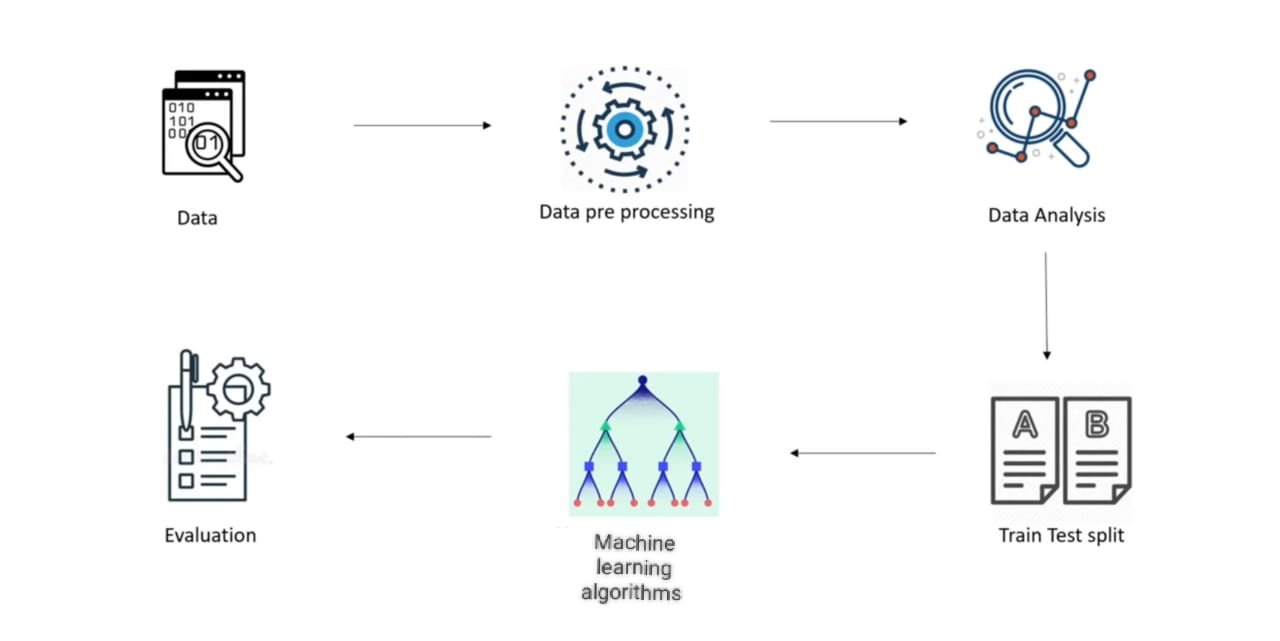


Fig 3.1 Workflow

**3.1.3 DATA ANALYSIS:**

Data analysis helps in understanding the characteristics of the data, identifying patterns and

relationships, and insights from data. In these operations carried out is listed below:

1. Univariate Analysis

2. Bivariate Analysis

**3.1.4: Train Test Split**

Train-test split is a common technique used in machine learning to evaluate the performance of a model. The most common split ratios are 80:20, 70:30 or 60:40 for the training and test sets respectively. The best split where error is less will be considered as best train-test split. Once the split is done, the model is trained on the training set and then evaluated on the test set.

**3.1.5 MACHINE LEARNING TECHNIQUES:**

The machine learning models are constructed after the features are extracted from the dataset. The models which are used are:

1.Linear Regression

2.Random Forest Regression

3. Lasso Regression

4.XGBooster Regressor

5.Ridge Regression

6.Artifical Neural Network

The model is trained with the cleaned dataset of different features, and then it is passed on to the next phases.

**3.1.5 EVALUATION:**

The trained model is then tested with the testing data and different evaluation metrics are calculated i.e., mean absolute error, mean squared error, R2 Score, root mean squared error.

The XG Booster Regressor was the best out of all the predictive models.

Table 3.1 The metrics comparison for two machine learning models

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.NO | Model Name | R2 score | Mean Absolute Error | Mean squared Error | RSME |
| 1 | Linear Regression | 0.61433 | 768.6087 | 1036142.446 | 1017.91 |
| 2 | Random Forest | 0.66057 | 675.7036 | 913110.1739 | 955.56 |
| 3 | XG Booster | 0.67015 | 701.9810 | 1019908.817 | 1009.9 |
| 4 | Lasso Regression | 0.61449 | 768.5132 | 1035702.570 | 1017.6 |
| 5 | Ridge Regression | 0.62899 | 789.8099 | 1147177.058 | 1071.6 |
| 6 | ANN algorithm | 0.66642 | 676.7329 | 966.0210429 | 966.02 |

**CHAPTER 4**

**SOURCE CODE**

**4.1 Data Pre-processing**

**4.1.1 Filling Missing Values**

import numpy as np

import pandas as pd

import matplotlib.pyplot as plt

import seaborn as sns

df = pd.read\_csv('Train.csv')

df.head()

# checking for missing values

df.isnull().sum()

df['Item\_Weight'].fillna(df['Item\_Weight'].mean(), inplace=True)

mode\_of\_Outlet\_size = df.pivot\_table(values='Outlet\_Size', columns='Outlet\_Type', aggfunc=(lambda x: x.mode()[0]))

miss\_values = df['Outlet\_Size'].isnull()

df.loc[miss\_values, 'Outlet\_Size'] = df.loc[miss\_values,'Outlet\_Type'].apply(lambda x: mode\_of\_Outlet\_size[x])

**4.1.2 Target Encoding**

import category\_encoders as ce

encoder=ce.TargetEncoder(cols=['Item\_Identifier','Item\_Fat\_Content','Item\_Type','Outlet\_Identifier','Outlet\_Size','Outlet\_Location\_Type','Outlet\_Type'] )

df = encoder.fit\_transform(df,df['Item\_Outlet\_Sales'])

df.head()

**4.2 Data Analysis**

**4.2.1 Univariate Analysis**

df.Item\_Fat\_Content.value\_counts().plot(kind = "bar")

sns.countplot(x = "Item\_Type", data = df)

plt.xticks(rotation = 90)

plt.show()

df.Outlet\_Type.value\_counts().plot(kind = "bar");

plt.figure(figsize=(5,5))

plt.show()

**4.2.2 Bivariate Analysis**

sns.catplot(x="Item\_Fat\_Content",y= "Item\_Outlet\_Sales",kind="boxen",data=df)

plt.scatter(df.Item\_MRP, df.Item\_Outlet\_Sales, color = "hotpink")

**4.3 Train-Test Split**

# Importing the train test split function

from sklearn.model\_selection import train\_test\_split

train\_x,test\_x,train\_y,test\_y = train\_test\_split(x,y, random\_state = 56,test\_size=0.4)

**4.4 Machine Learning Techniques**

**4.4.1 Linear Regression**

from sklearn.linear\_model import LinearRegression as LR

**#** Creating instance of Linear Regresssion

lr = LR(normalize = True)

# Fitting the model

lr.fit(train\_x, train\_y)

# Predicting over the Test Set and calculating error

test\_predict = lr.predict(test\_x)

from sklearn.metrics import mean\_absolute\_error as mae

from sklearn.metrics import mean\_squared\_error,r2\_score

print('R2\_score' ,r2\_score(test\_predict,test\_y))

print('Test Mean Absolute Error ', k )

print('Test Mean Squared Error',mean\_squared\_error(test\_predict,test\_y))

**4.4.2 Artificial Neural Network**

from keras.models import Sequential

from keras.layers import Dense

model = Sequential()

model.add(Dense(units=64, activation='relu', input\_dim=X\_train.shape[1]))

model.add(Dense(units=32, activation='relu'))

model.add(Dense(units=1, activation='linear'))

**# Model Tuning**

model.compile(optimizer='adam',loss='mean\_absolute\_error',metrics=['mean\_absolute\_error'])

model.summary()

y\_pred = model.predict(X\_test)

from sklearn.metrics import mean\_absolute\_error as mae

from sklearn.metrics import mean\_squared\_error,r2\_score

print('R2\_score' ,r2\_score(test\_predict,test\_y))

print('Test Mean Absolute Error ', k )

print('Test Mean Squared Error',mean\_squared\_error(test\_predict,test\_y))

**4.4.3 XGBoost Regression :**

param\_grid = {

'max\_depth': [3, 4, 5],

'learning\_rate': [0.01, 0.05, 0.1],

'n\_estimators': [100, 500, 1000],

'objective': ['reg:squarederror', 'reg:linear', 'reg:gamma']

}

from xgboost import XGBRegressor

xgb\_model = XGBRegressor()

from sklearn.model\_selection import GridSearchCV

grid\_search = GridSearchCV(estimator=xgb\_model, param\_grid=param\_grid, cv=5, n\_jobs=-1, verbose=2)

X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(X,Y,test\_size=0.3,random\_state=2)

print(X.shape,X\_train.shape,X\_test.shape)

grid\_search.fit(X\_train,Y\_train)

best\_params = grid\_search.best\_params\_

best\_score = grid\_search.best\_score\_

xgb\_model = XGBRegressor(\*\*best\_params)

xgb\_model.fit(X\_train, Y\_train)

test\_pred = xgb\_model.predict(X\_test)

r2\_test = metrics.r2\_score(Y\_test,test\_pred)

print("r2 score : ")

print(r2\_test)

mean\_absolute\_test = metrics.mean\_absolute\_error(Y\_test,test\_pred)

print("Mean absolute Error :")

print(mean\_absolute\_test)

mean\_squared\_test = metrics.mean\_squared\_error(Y\_test,test\_pred)

print("Mean squared Error :")

print(mean\_squared\_test)

**4.4.4 Ridge Regression :**

from sklearn.linear\_model import Ridge

from sklearn.model\_selection import GridSearchCV

ridge\_reg = Ridge()

X\_train,X\_test,Y\_train,Y\_test = train\_test\_split(X,Y,test\_size=0.3,random\_state=2)

ridge\_reg.fit(X\_train,Y\_train)

test\_pred = ridge\_reg.predict(X\_test)

print(X.shape,X\_train.shape,X\_test.shape)

r2\_test = metrics.r2\_score(Y\_test,test\_pred)

print("r2 score : ")

print(r2\_test)

mean\_absolute\_test = metrics.mean\_absolute\_error(Y\_test,test\_pred)

print("Mean absolute Error :")

print(mean\_absolute\_test)

mean\_squared\_test = metrics.mean\_squared\_error(Y\_test,test\_pred)

print("Mean squared Error :")

print(mean\_squared\_test)

**4.4.5 Random Forest Regression :**

from sklearn.ensemble import RandomForestRegressor

from sklearn.model\_selection import GridSearchCV

param\_grid = {

'bootstrap': [True],

'max\_depth': [5, 7, 9, 11],

'max\_features': [5, 7, 9, 3, 11],

'min\_samples\_leaf': [10, 50, 100],

'n\_estimators': [200, 400, 600, 300] }

rf = RandomForestRegressor()

grid\_search = GridSearchCV(estimator = rf, param\_grid = param\_grid, cv = 5, n\_jobs = 4, verbose = 2)

rid\_search.fit(X\_train[model\_cols],Y\_train)

grid\_search.best\_params\_

rf=RandomForestRegressor(n\_estimators=400,max\_depth=7,n\_jobs=4,min\_samples\_leaf=10,max\_features=5)

rf.fit(X\_train[model\_cols],Y\_train)

Y\_pred\_rf= rf.predict(X\_test[model\_cols])

print("r2 score :")

print(r2\_score(Y\_test,Y\_pred\_rf))

print("mean absolute error :")

print(mean\_absolute\_error(Y\_test,Y\_pred\_rf))

print("mean square error :")

print(mean\_squared\_error(Y\_test,Y\_pred\_rf))

**4.4.6 Lasso Regression :**

from sklearn.linear\_model import Lasso

from sklearn.model\_selection import GridSearchCV

param\_grid = {'alpha': [0.001, 0.01, 0.1, 1, 10]}

grid\_search = GridSearchCV(lasso, param\_grid=param\_grid, cv=5)

grid\_search.fit(X\_train[model\_cols], Y\_train)

print("Best Parameter: ", grid\_search.best\_params\_)

lasso = Lasso(alpha=10)

lasso.fit(X\_train[model\_cols], Y\_train)

Y\_pred = lasso.predict(X\_test[model\_cols])

print("r2 score :")

print(r2\_score(Y\_test,Y\_pred))

print("mean absolute error :")

print(mean\_absolute\_error(Y\_test,Y\_pred))

print("mean square error :")

print(mean\_squared\_error(Y\_test,Y\_pred))

**Feature Selection & Actual vs Predicted :**

import shap

shap.initjs()

explainer = shap.LinearExplainer(ridge\_reg,X\_test)

shap\_values = explainer.shap\_values(X\_test)

shap.summary\_plot(shap\_values, X\_test, plot\_type="bar")

import category\_encoders as ce

cat\_cols = ['Item\_Identifier','Item\_Fat\_Content','Item\_Type','Outlet\_Identifier','Outlet\_Size','Outlet\_Location\_Type','Outlet\_Type']

encoder = ce.TargetEncoder(cols=cat\_cols)

bd\_ = encoder.fit\_transform(bd[cat\_cols], bd['Item\_Outlet\_Sales'])

bd\_.columns = [col+'enc' for col in bd\_.columns]

bd = pd.concat([bd,bd\_], axis=1)

bd

column = 'Outlet\_Type'

k = X\_test.groupby([column])[['actual','pred']].sum().reset\_index()

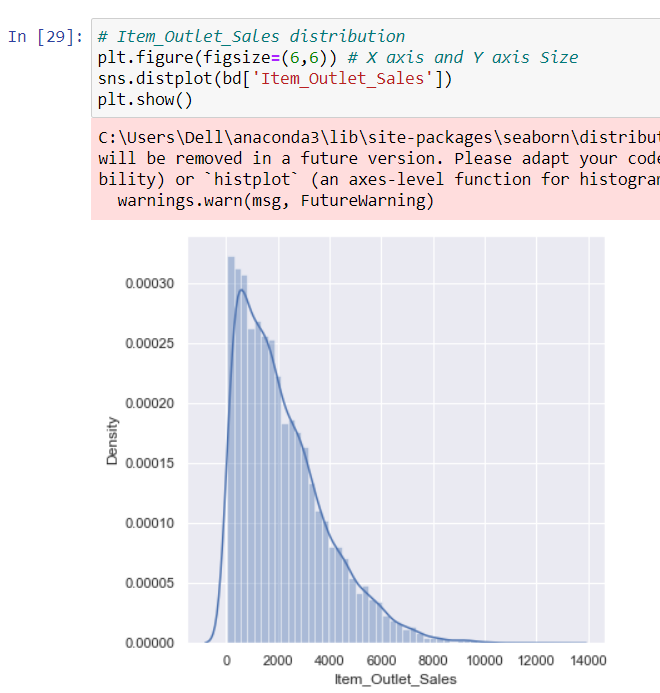
k = pd.melt(k, id\_vars=column, var\_name="type\_of\_value", value\_name="sales")

sns.factorplot(x=column, y='sales', hue='type\_of\_value', data=k, kind='bar', height=8.27, aspect=11.7/8.27)

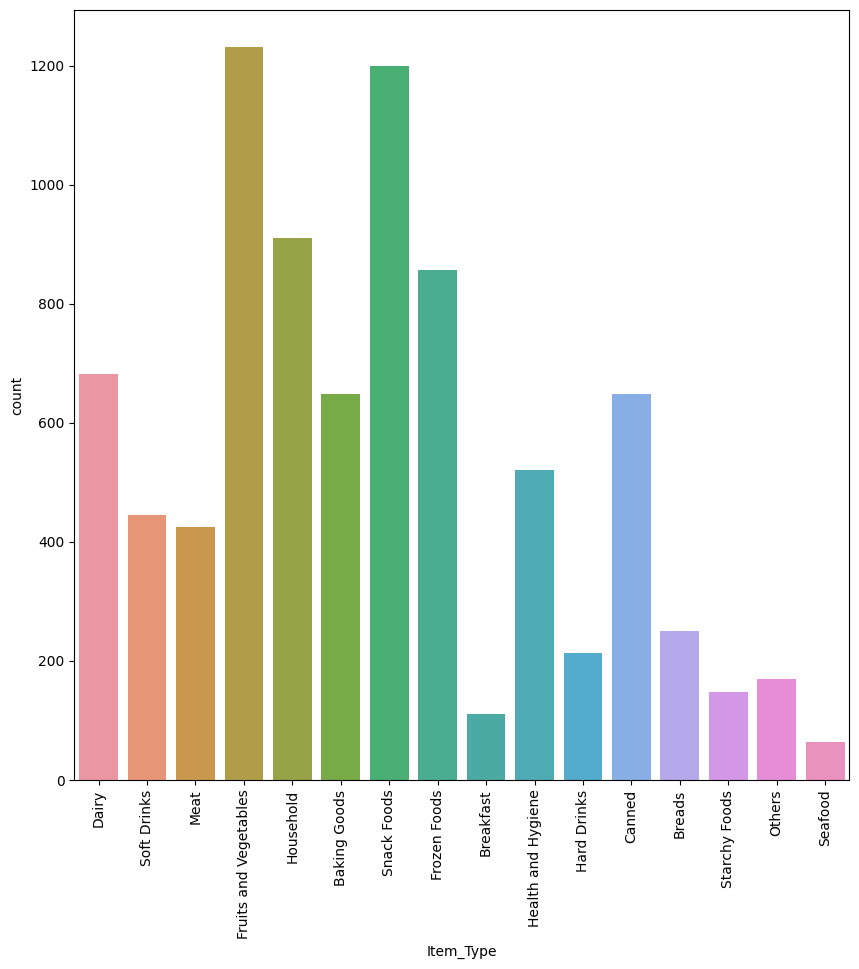
**CHAPTER 5**

# SNAPSHOTS

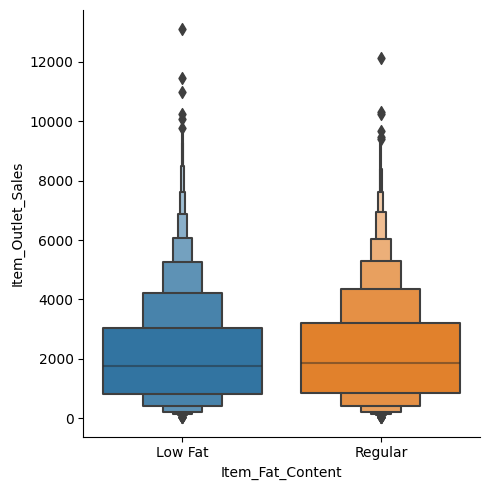
**5.1 Data Analysis**



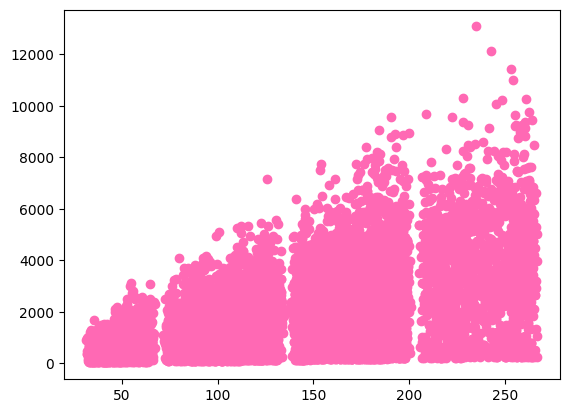
5.1.1 Distribution Plot for Item Outlet Sales



5.1.2 Count Plot for Item Type



5.1.3 Categorical Plot Between Item Fat Content & Outlet sales



5.1.4 Scatter Plot Between Item MRP & Outlet sales

**5.2 Linear Regression :**

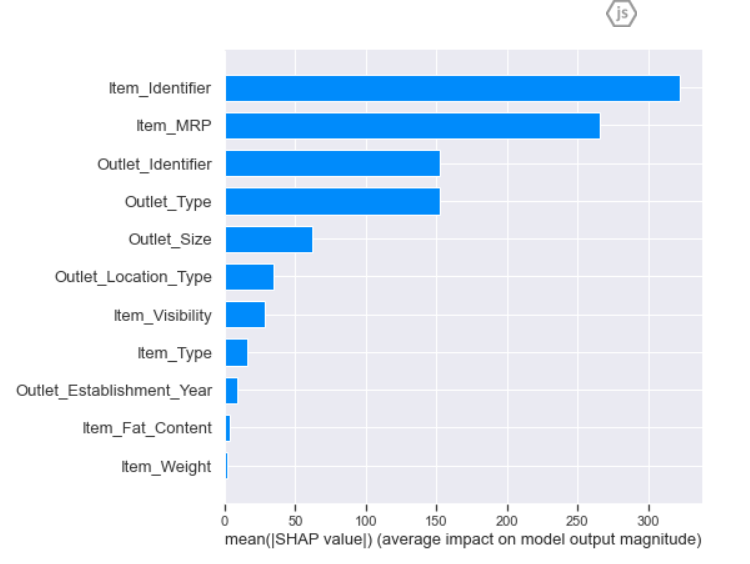


Fig 5.2.1 Feature Selection Graph

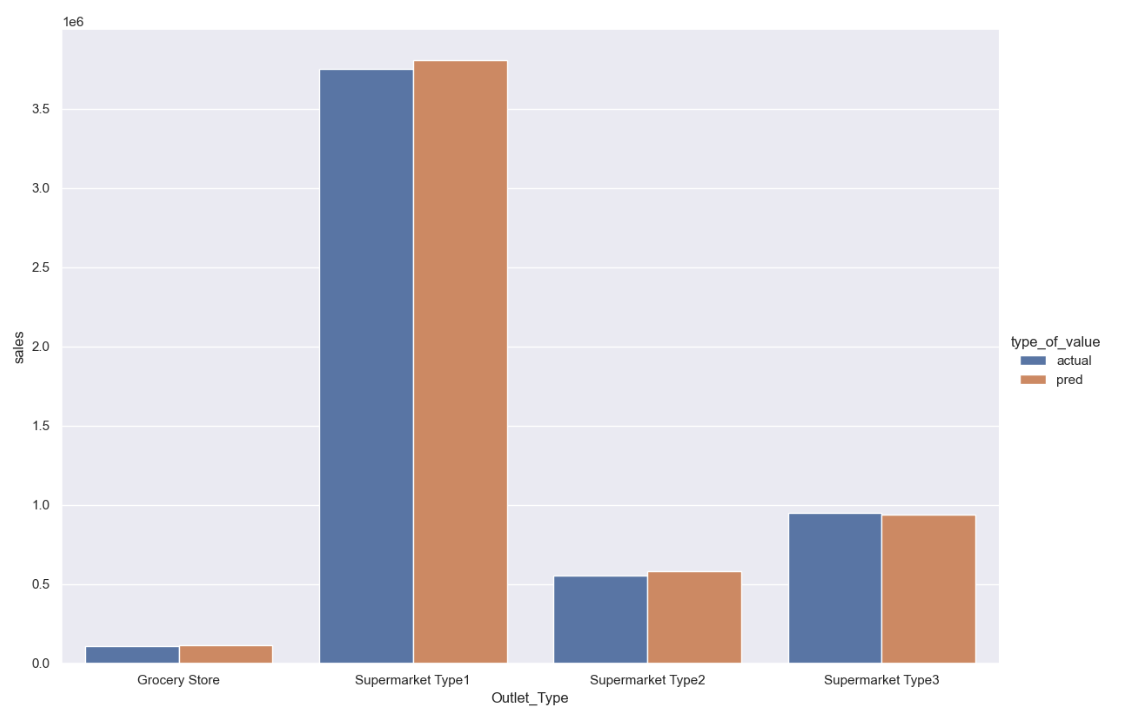


Fig 5.2.2 Comparision of Actual & Predicted Values

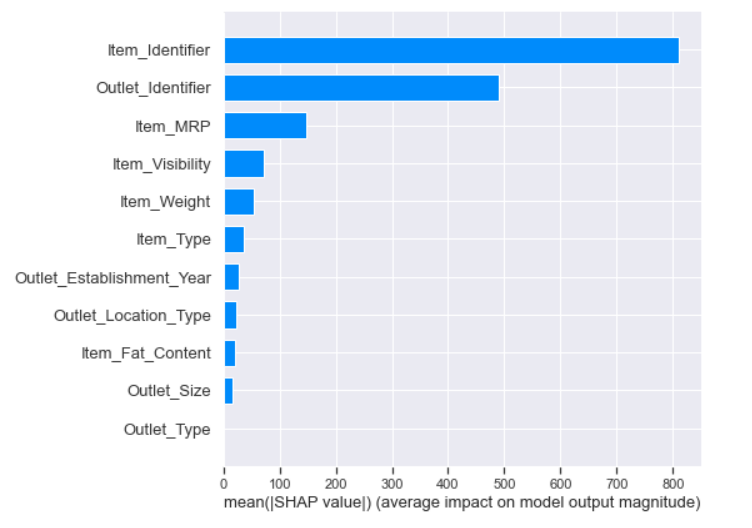
* 1. **XGBoost Regression : **

Fig 5.3.1 Feature Selection Graph

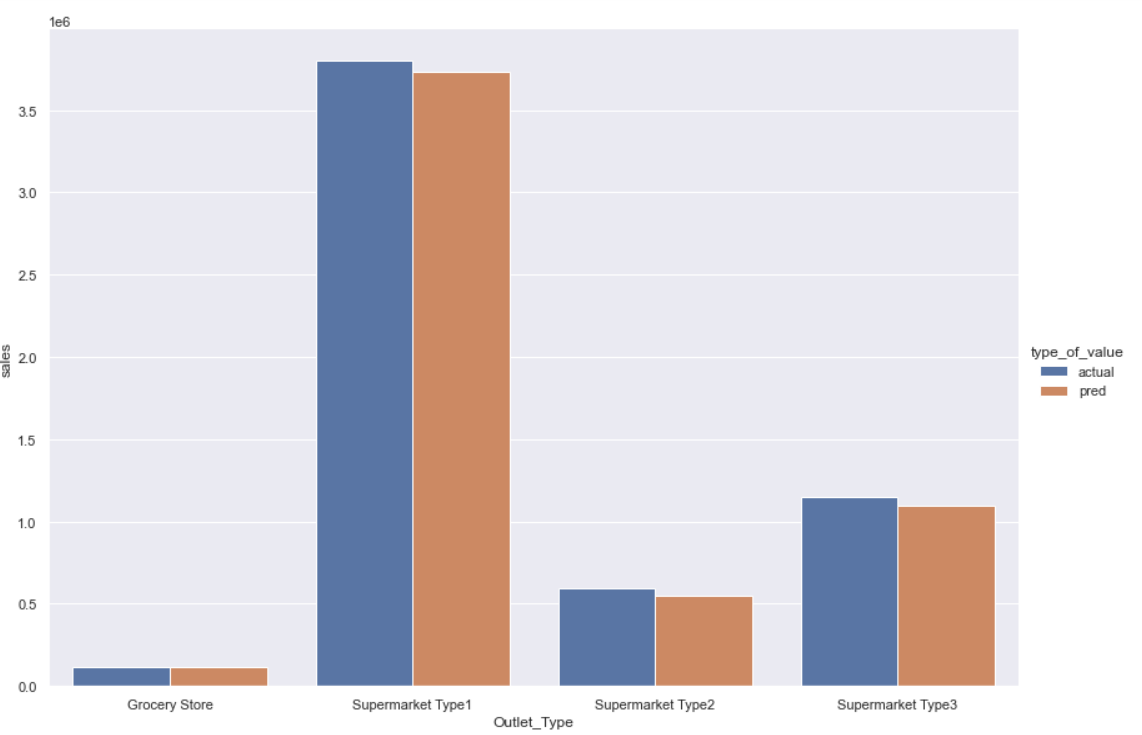


Fig 5.3.2 Comparision of Actual & Predicted Values

* 1. **Ridge Regression :**

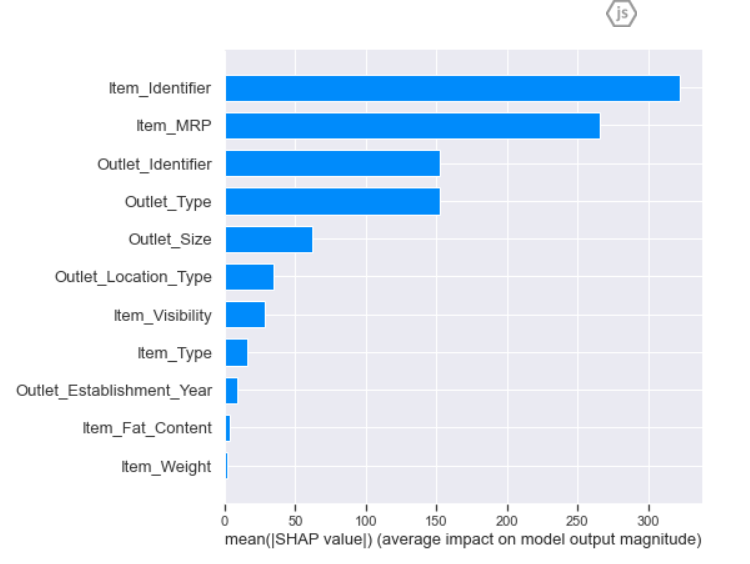
****

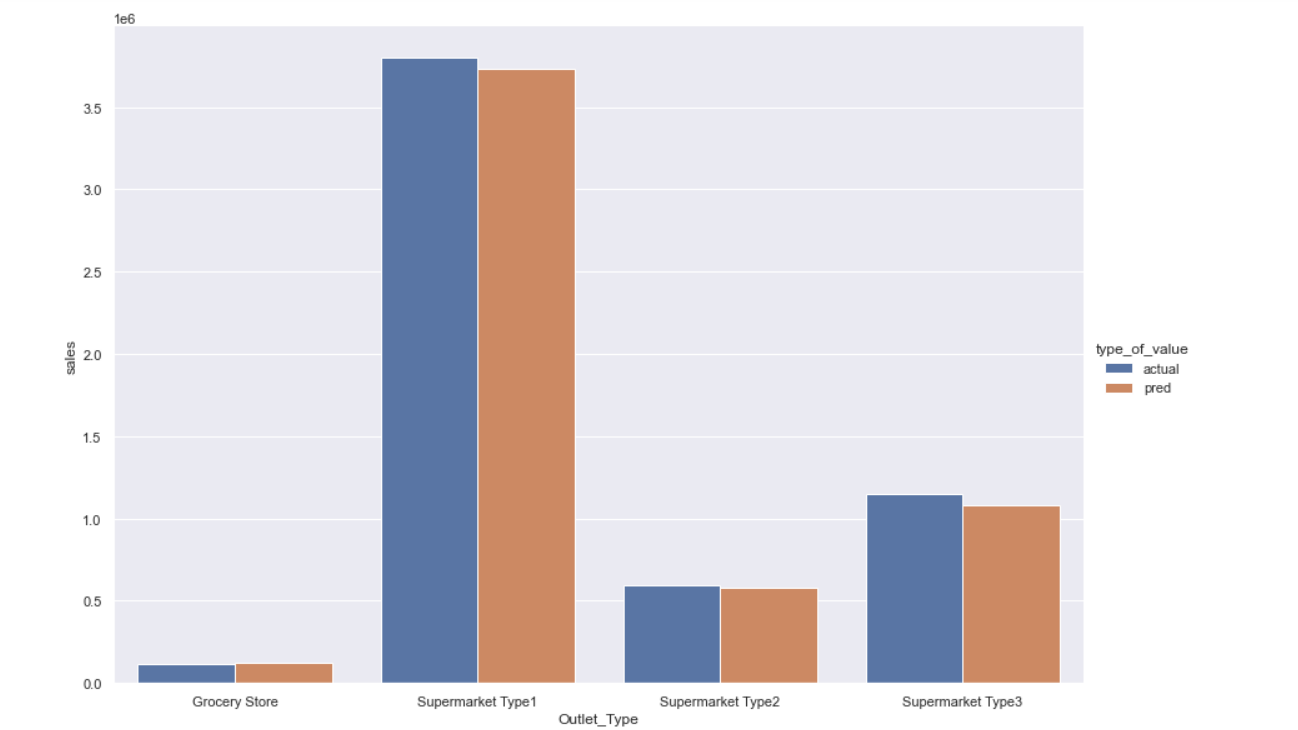
Fig 5.4.1 Feature Selection Graph

Fig 5.4.2 Comparision of Actual & Predicted Values

* 1. **Random Forest Regression :**

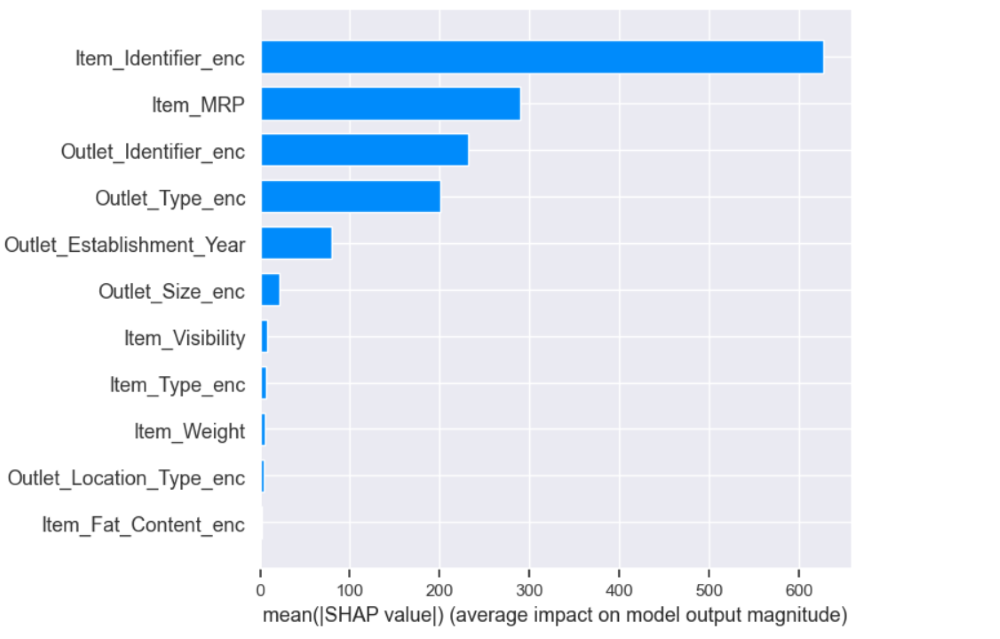


Fig 5.5.1 Feature Selection Graph

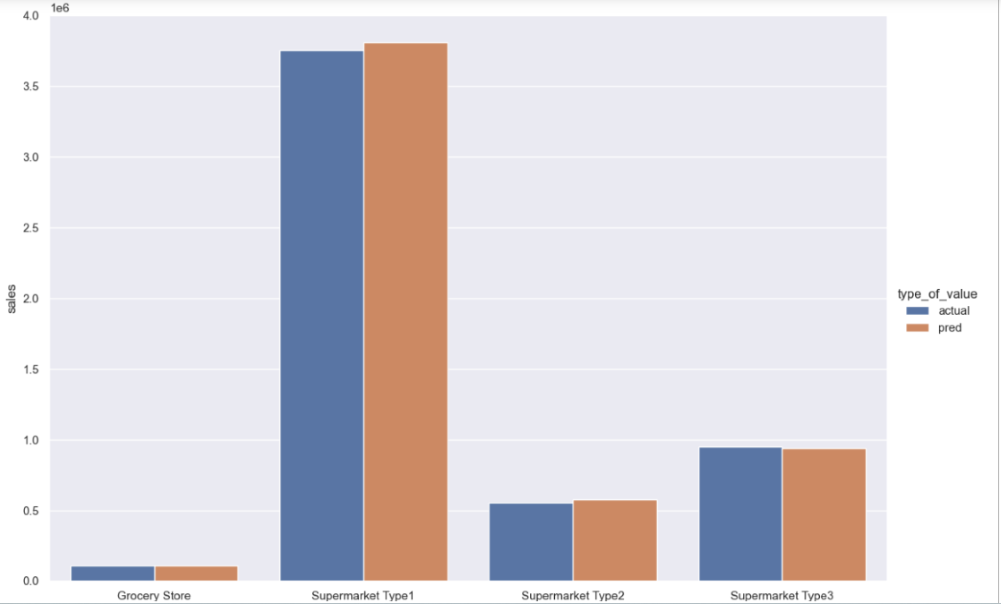


Fig 5.5.2 Comparision of Actual & Predicted Values

* 1. **Lasso Regression :**

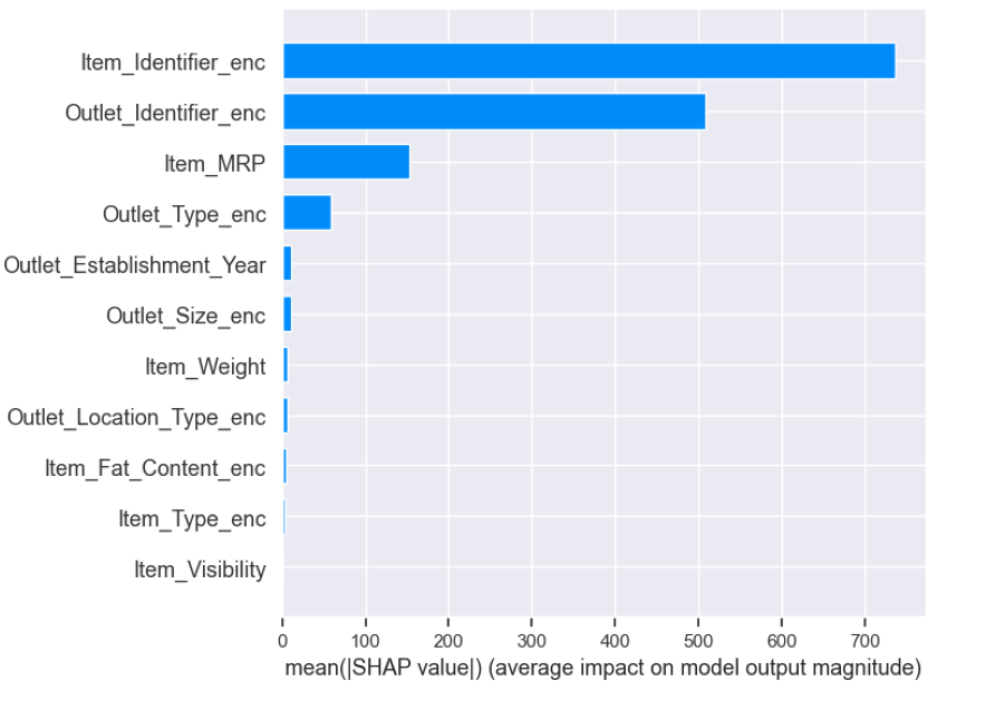


Fig 5.6.1 Feature Selection Graph

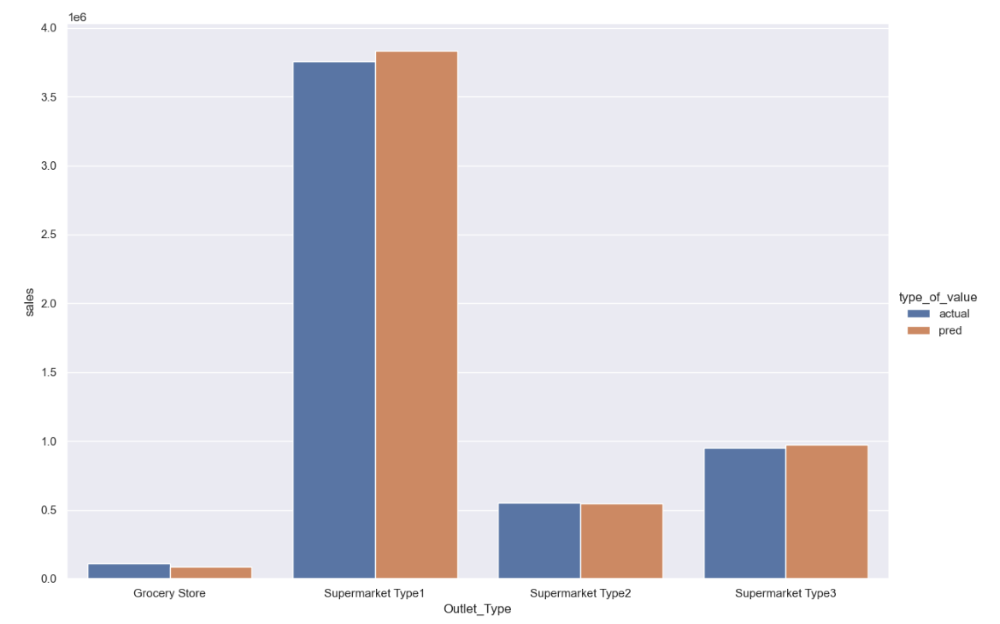


Fig 5.6.2 Comparision of Actual & Predicted Values

* 1. **Web Frame :**

****

Fig 5.7.1 Home Page

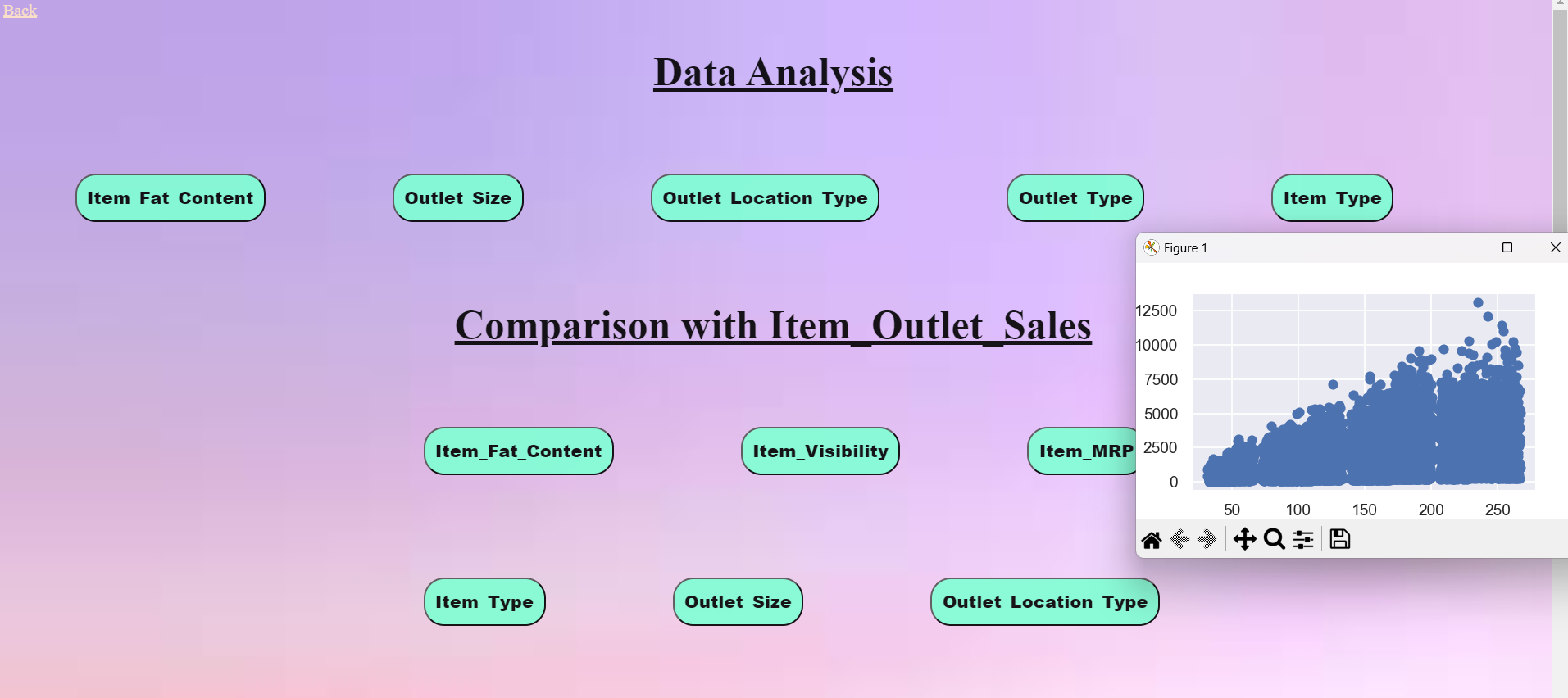
****

Fig 5.7.2 Data Analysis

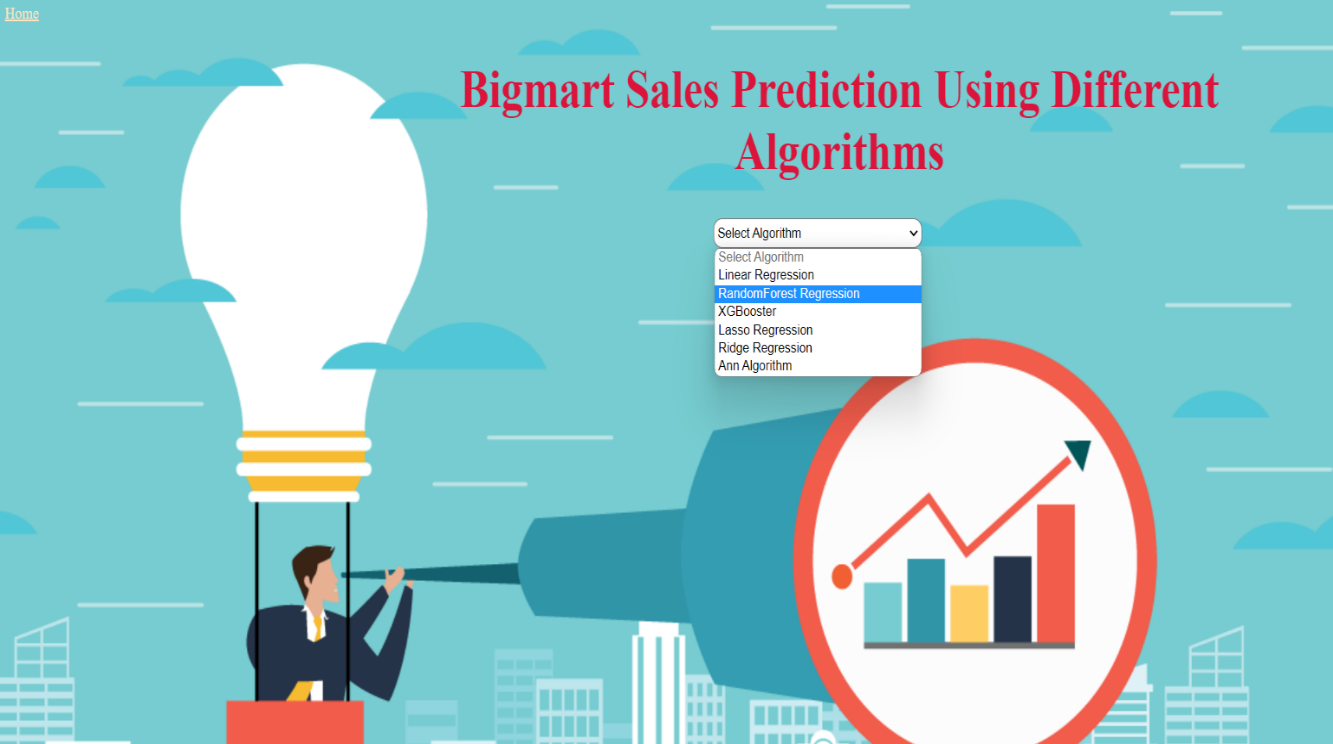


Fig 5.7.3 Prediction with different algorithms

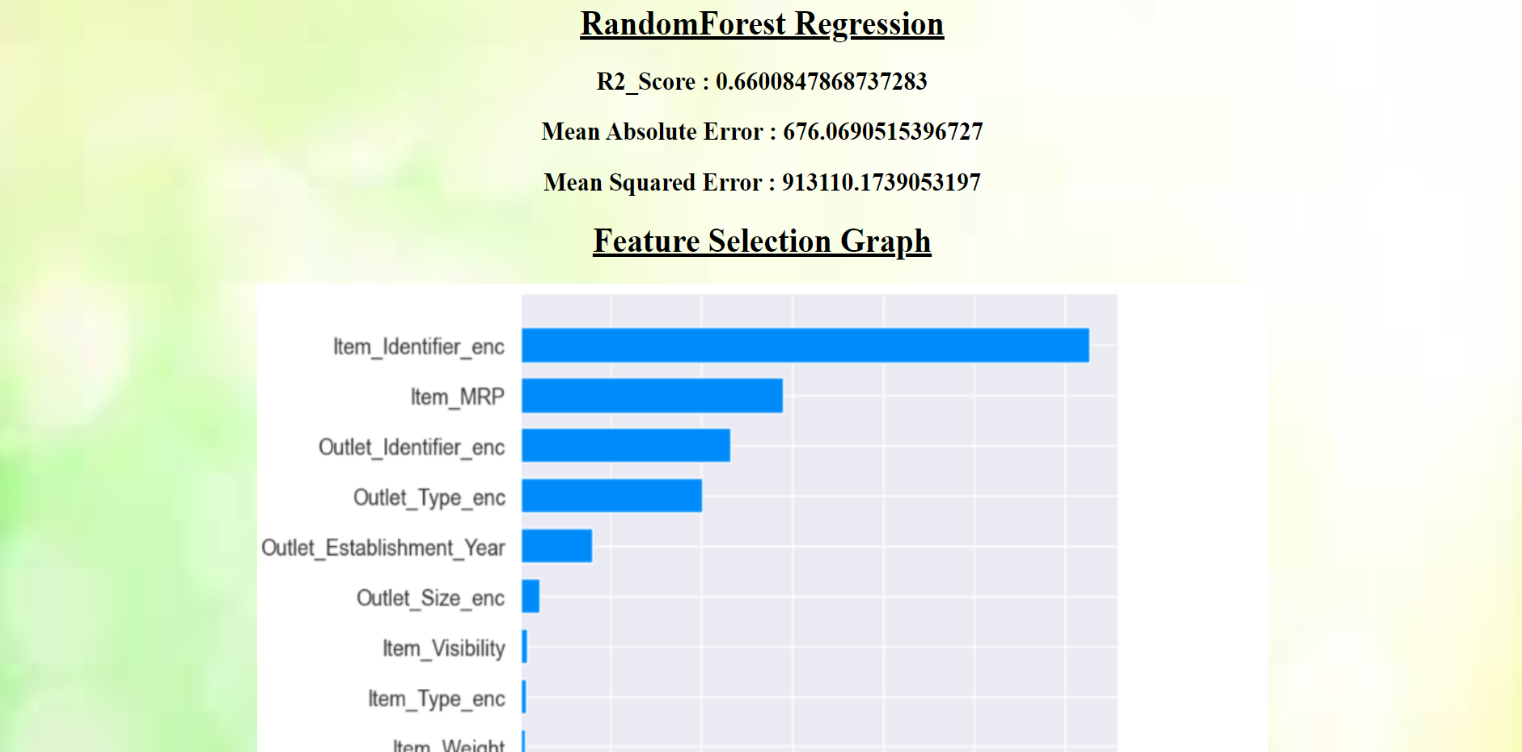


Fig 5.7.4 Random Forest Regression

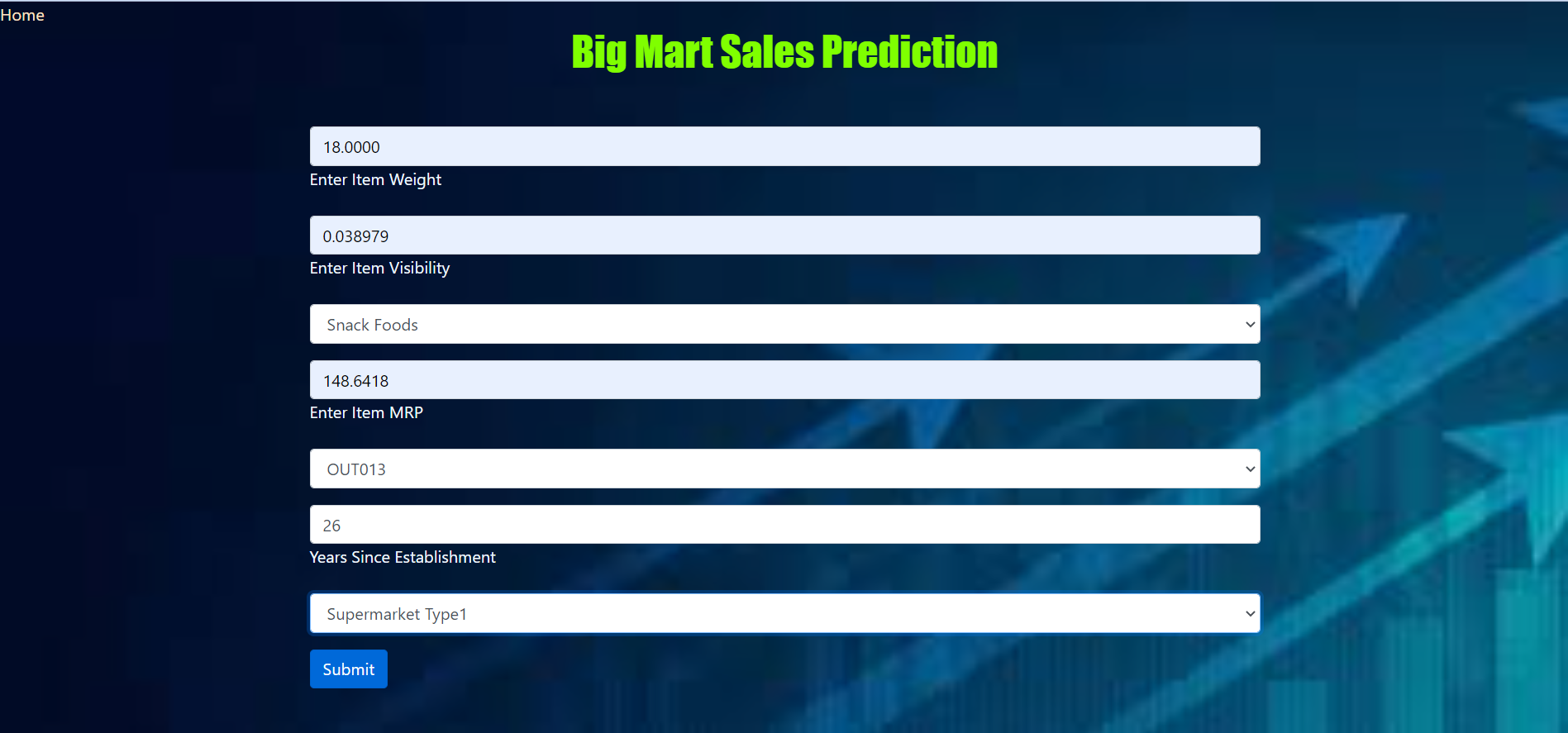


Fig 5.7.5 Big Mart Sales Prediction



Fig 5.7.6 Result & Data Analysis Report

**CHAPTER 6**

## Conclusion and Future Plans

## 6.1 Conclusion:

## In conclusion, by projecting sales and comprehending the elements that affect sales performance, machine learning algorithms are used in bigmart to maximise profitability, increase operational efficiency, and enhance decision-making processes.BigMart can manage supply chains, organise promotions, optimise inventory levels, set fair prices, and maximise profitability by comprehending demand for various products.BigMart can evaluate the necessity of marketing efforts, pinpoint areas for improvement, and make data-driven decisions to improve overall performance by comparing forecasted sales figures with actual sales data.Data analysis is additional factor to know more about the attributes.BigMart may acquire insights into consumer behaviour and market trends and modify business plans by finding underlying patterns and trends for various products.Overall, the BigMart sales forecast project using machine learning algorithms serves as a valuable tool for improving decision-making processes, increasing operational efficiency, and maximising profitability.

**5.2 Future Works:**

* Accuracy of the model can be improved by adding the data of outlet\_sales for more years.
* To make the process much convinient we can use it in the bigmart stores and make it available as an application.
* To maintain the quality of the sales projections, regular monitoring and review of the model's accuracy, performance indicators, and any biases are crucial.

**CHAPTER 7**

## References

[1] Kutner, Nachtsheim, Neter, L (Eds.), Applied Linear Statistical Models, fifth ed., Mc Graw Hill India, 2013.

[2] Yuri Demchenko, Cees de Laat, Peter Membrey, Defining Architecture Components of the Big Data Ecosystem, 2014.

[3] Big Sky, The data analysis process: 5 steps to better decision making, in: https ://www.bigskyassociates.com/blog/bid/372186/The-Data-Analysis-Process-5-Ste ps-To-Better-DecisionMaking. [4] Dataaspirant, HOW the random forest algorithm works in machine learning. htt ps://dataaspirant.com/2017/05/22/random-forest-algorithm-machine-learing/.

[5] Mohit Gurnani, Yogesh Korke, Prachi Shah, Sandeep Udmale, Vijay Sambhe, Sunil Bhirud, Forecasting of sales by using fusion of machine learning techniques, in: 2017 International Conference on Data Management, Analytics and Innovation (ICDMAI), IEEE, October 2017.

[6] J. Armstrong, Sales Forecasting”, SSRN Electronic Journal, July 2008.

[7] Samaneh Beheshti-Kashi, Hamid Reza Karimi, Klaus-Dieter Thoben, Michael Lütjen, “A survey on retail sales forecasting and prediction in fashion markets”, Sys. Sci. Control Eng.: Open Access J.. 3. 154-161. 10.1080/ 21642583.2014.999389.

[8] Gopalakrishnan T, RiteshChoudhary,Sarada Prasad, “Prediction of sales value in online shopping using LinearRegression”,20174th International Conference on Computing Communication and Automation (ICCCA). [11] Megahed, Peifeng Yin and Hamid Reza Motahari Nezhad, “An optimization approach to services SalesForecasting in A multi staged sales pipeline”, 2016 IEEE International Conference on Services Computing

**CHAPTER-8**

**APPENDIX**

**Title:** Exploratory Data analysis and sales forecasting of bigmart dataset using supervised and ANN algorithms

**Journal:** Measurement : sensors

**Authors:** T.K. Thivakaran , M Ramesh

#### Published: 2022

**Link:** https://www.sciencedirect.com/science/article/pii/S2665917422000228

**ORIGINALITY REPORT /PLAGARISM CHECK :**

