EED-363: Applied Machine Learning Project

Vinayak Mehrotra- 1710110391- vm897@snu.edu.in
Sachin Mavi- 1710110285- sm790@snu.edu.in

Big Mart Sales Prediction

Shiv Nadar University

INTRODUCTION

This report describes our implementation of the Big Mart Sales Prediction System. Big Mart is a big supermarket chain and this project predicts sales for their products across different stores. Big Mart has collected data of 1559 products in 10 different stores for the year 2013 in order to identify which product and store plays a pivotal role in their profit generation.

Description of the Problem

Before selecting the algorithms and the performance metrics, it is important to understand the question. We are dealing here with a regression problem, because the target which is Item Outlet Sales is a continuous variable and not a discrete problem.

1. Supervised Learning: The data is provided with data labels along with the target variable at hand.
2. Plain Batch Learning: This is not a time-series data and new data can be incorporated easily without changing the data much.
3. Performance Measure: Being a regression problem where we fit a line, Root Mean Square Error (RMSE) is an appropriate measure for our problem.

Description of the Dataset

1. The dataset is already divided into training and test dataset, with the training data containing 8523 examples and 11 features and 1 target variable.

Variable	Description
Item_Identifier	Unique product ID
Item_Weight	Weight of product
Item_Fat_Content	Whether the product is low fat or not
Item_Visibility	The % of total display area of all products in a store allocated to the particular product
Item_Type	The category to which the product belongs
Item_MRP	Maximum Retail Price (list price) of the product
Outlet_Identifier	Unique store ID
Outlet_Establishment_Year	The year in which store was established
Outlet_Size	The size of the store in terms of ground area covered
Outlet_Location_Type	The type of city in which the store is located
Outlet_Type	Whether the outlet is just a grocery store or some sort of supermarket
Item_Outlet_Sales	Sales of the product in the particulat store. This is the outcome variable to be predicted.

Index	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
count	7060	8523	8523	8523	8523
mean	12.8576	0.066132	140.993	1997.83	2181.29
std	4.64346	0.0515978	62.2751	8.37176	1706.5
min	4.555	0	31.29	1985	33.29
25%	8.77375	0.0269895	93.8265	1987	834.247
50%	12.6	0.0539309	143.013	1999	1794.33
75%	16.85	0.0945853	185.644	2004	3101.3
max	21.35	0.328391	266.888	2009	13087

2. The dataset contains 5 numeric features (including one target) and 7 categorical features.

Index	0
Item_Weight	float64
Item_Visibility	float64
Item_MRP	float64
Outlet_Establishment_Year	int64
Item_Outlet_Sales	float64

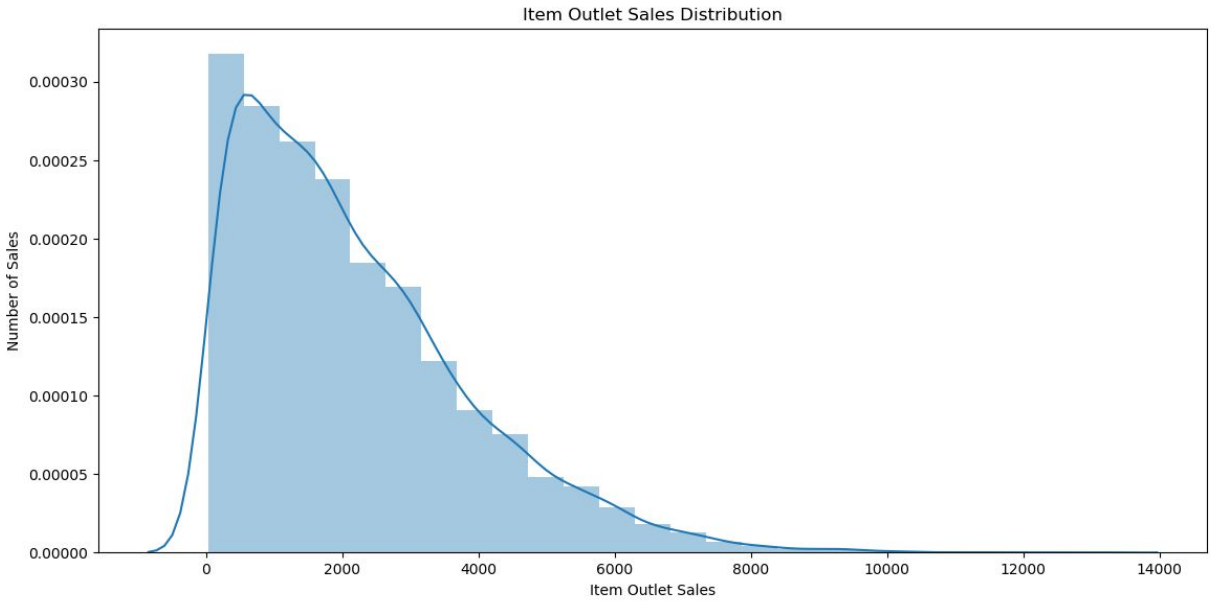
Index	Item_Identifier	Item_Fat_Content	Item_Type	Outlet_Identifier	Outlet_Size	Outlet_Location_Type	Outlet_Type
0	FDA15	Low Fat	Dairy	OUT049	Medium	Tier 1	Supermarket Type1
1	DRC01	Regular	Soft Drinks	OUT018	Medium	Tier 3	Supermarket Type2
2	FDN15	Low Fat	Meat	OUT049	Medium	Tier 1	Supermarket Type1
3	FDX07	Regular	Fruits and Vegetables	OUT010	nan	Tier 3	Grocery Store
4	NCD19	Low Fat	Household	OUT013	High	Tier 3	Supermarket Type1
5	FDP36	Regular	Baking Goods	OUT018	Medium	Tier 3	Supermarket Type2
6	FDO10	Regular	Snack Foods	OUT013	High	Tier 3	Supermarket Type1
7	FDP10	Low Fat	Snack Foods	OUT027	Medium	Tier 3	Supermarket Type3
8	FDH17	Regular	Frozen Foods	OUT045	nan	Tier 2	Supermarket Type1

3. The dataset is not clean, in the sense that there are some missing values, some redundancies in the dataset, which we will fix and then fit models in it.

```
<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
```

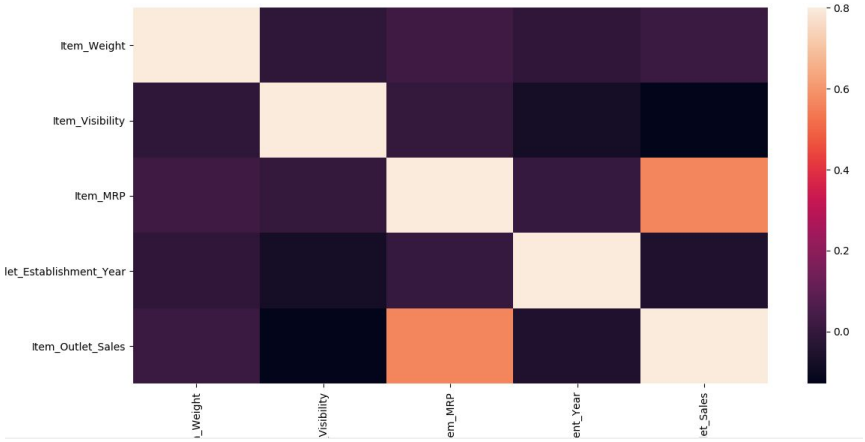
Complete Analysis of the Dataset

1. Histogram displaying the distribution of the target variable.

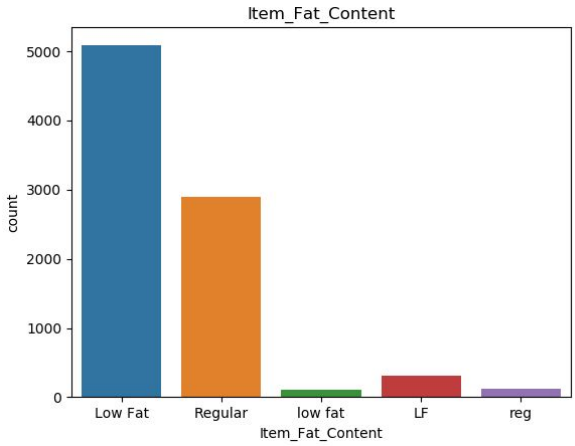


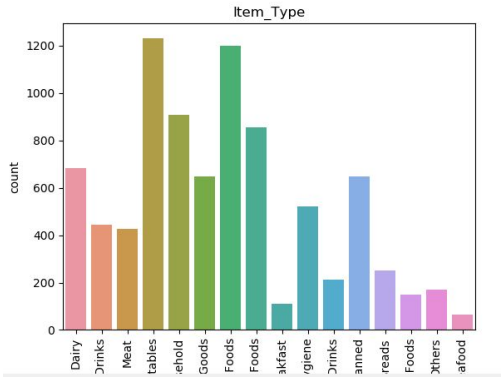
2. Correlation of the numeric features with the target and a heatmap

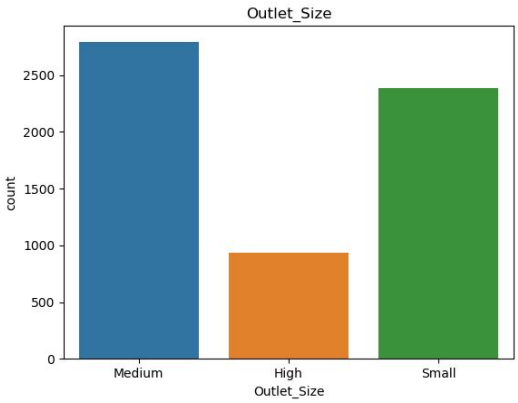
Index	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
Item_Weight	1	-0.0140477	0.0271412	-0.0115883	0.0141227
Item_Visibility	-0.0140477	1	-0.00131485	-0.0748335	-0.128625
Item_MRP	0.0271412	-0.00131485	1	0.00501992	0.567574
Outlet_Establishment_Year	-0.0115883	-0.0748335	0.00501992	1	-0.049135
Item_Outlet_Sales	0.0141227	-0.128625	0.567574	-0.049135	1

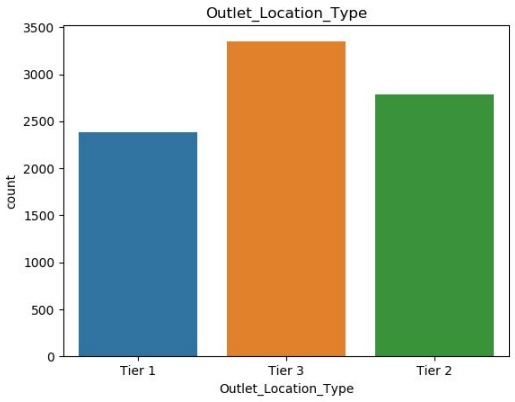


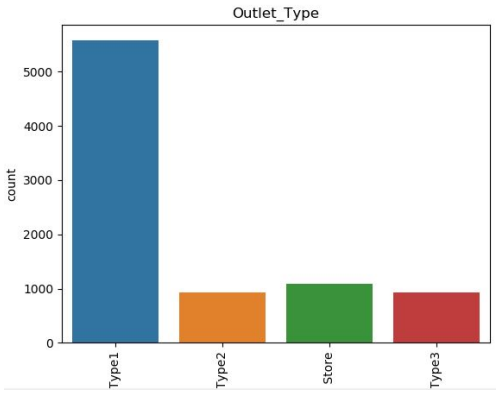
3. Univariate Analysis of categorical features











4. Summary of Univariate analysis

· Upon univariate analysis, we observed that the feature Item_Fat_Content had redundancies in its categories, i.e., Low Fat was written in three different ways, which had to be corrected

· We observed that the feature Item_Identifier indicates three different kinds of items broadly, so it had to be made that way.

· We observed that in the dataset, Item_Visibility feature has some values as 0, which implies that the store has no such product, so we treated it as a missing value and used the 'mean' strategy for the imputer.

· Features Item_Weight and Outlet_Size contain some values as NaN.

· There are 1559 unique items in a single store, which had to be kept in mind.

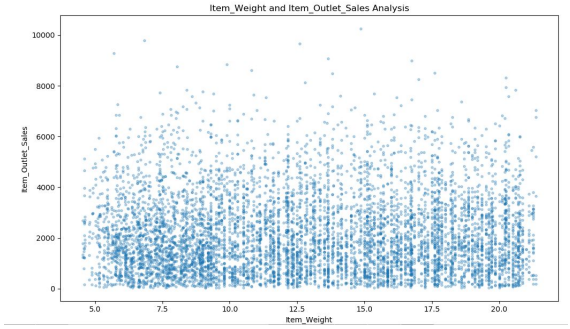
· The Item_Type feature has different scattered categories, so we can create a new variable which makes this classification more streamlined, so that the encoding is reduced.

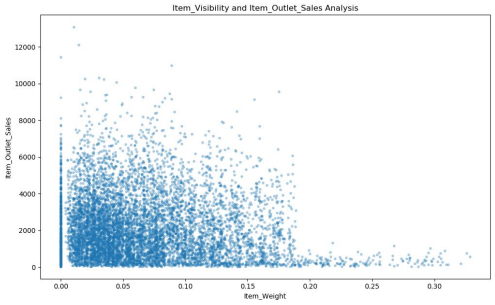
· Feature Outlet_Establishment_Year has years which are not understood by the model as they should be. So, we have to convert them in a way that is more intuitive for the models.

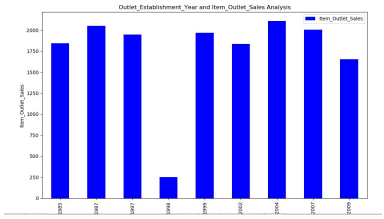
· Using Dummy Encoding for the categorical variables, we have to keep in mind that we take n-1 columns out of the n dummy encoded columns for a categorical feature, to prevent dummy variable trap which brings in Multicollinearity.

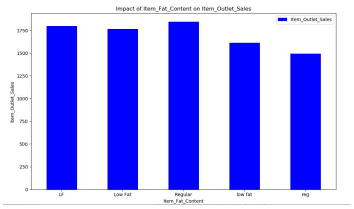
· We also observed that certain products were Non-Consumable and we had to replace their values in the Item_Fat_Content feature with Non-Edible.

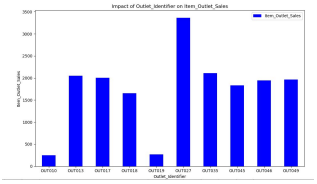
5. Bivariate Analysis of categorical features

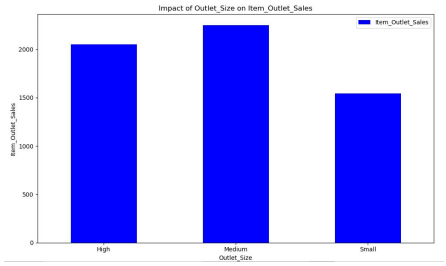


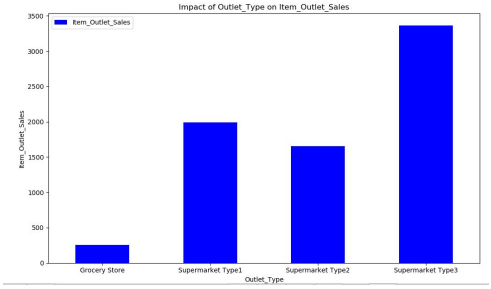


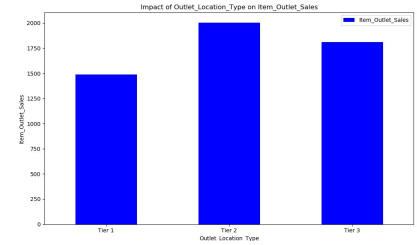












Index	OUT010	OUT013	OUT017	OUT018	OUT019	OUT027	OUT035	OUT045	OUT046	OUT049
Outlet_Type	Grocery Store	Supermarket Type1	Supermarket Type1	Supermarket Type2	Grocery Store	Supermarket Type3	Supermarket Type1	Supermarket Type1	Supermarket Type1	Supermarket Type1

Index	Grocery Store	Supermarket Type1	Supermarket Type2	Supermarket Type3
Outlet_Location_Type	Tier 3	Tier 2	Tier 3	Tier 3

6. Summary of Bivariate Analysis

Item_Weight and Item_Outlet_Sales Analysis: we found out that the correlation was low with the help of a heatmap

Item_Visibility and Item_Outlet_Sales Analysis: Initial guess based on intuition is that the products that are kept in front will make the sales go high and increase the profit

Many Products have Visibility = 0

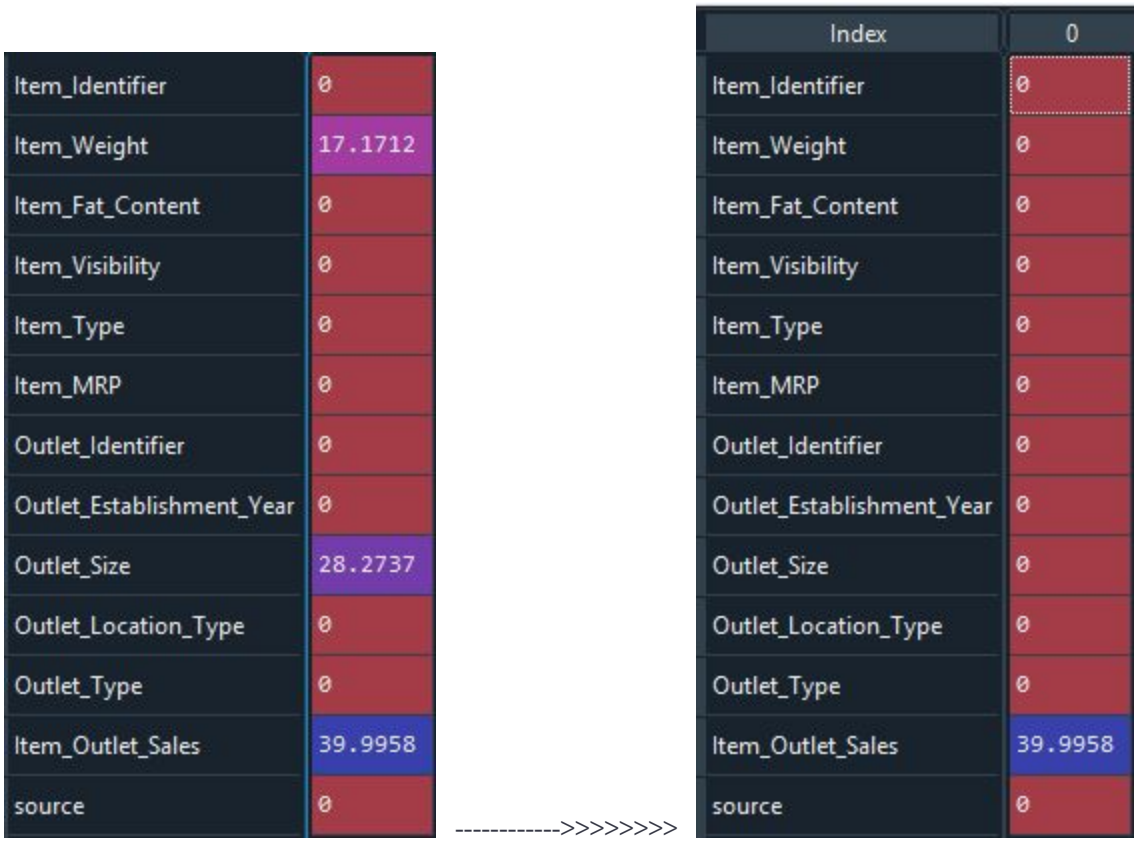
The data shows a trend that eliminates our hypothesis, which can be due to the fact that important products that control the profit do not need substantial visibility, they are just in demand

Year is not related to the target, year 1998 has low sales which may be due to the fact that few stores may have opened in that year

low fat products seem to have higher sales than regular products

It is visible that Grocery Stores have less sales, maybe because why will someone go for grocery store and then to a different store when there are big stores having everything available under one roof. Supermarket type 3 has higher sales than Supermarket type 1.

7. Checking the percentage of null values in features



8. After imputing the missing values and encoding the categorical features:

```
Data columns (total 33 columns):
 #   Column                      Non-Null Count  Dtype
---  ------                      --------------  -----
 0   Item_Identifier             8523 non-null   object
 1   Item_Weight                 8523 non-null   float6
 2   Item_Visibility             8523 non-null   float6
 3   Item_MRP                    8523 non-null   float6
 4   Outlet_Identifier           8523 non-null   object
 5   Item_Outlet_Sales           8523 non-null   float6
 6   Outlet_Years                8523 non-null   int64
 7   Item_Fat_Content_0          8523 non-null   uint8
 8   Item_Fat_Content_1          8523 non-null   uint8
 9   Item_Fat_Content_2          8523 non-null   uint8
 10  Outlet_Location_Type_0      8523 non-null   uint8
 11  Outlet_Location_Type_1      8523 non-null   uint8
 12  Outlet_Location_Type_2      8523 non-null   uint8
 13  Outlet_Size_0               8523 non-null   uint8
 14  Outlet_Size_1               8523 non-null   uint8
 15  Outlet_Size_2               8523 non-null   uint8
 16  Outlet_Type_0               8523 non-null   uint8
 17  Outlet_Type_1               8523 non-null   uint8
 18  Outlet_Type_2               8523 non-null   uint8
 19  Outlet_Type_3               8523 non-null   uint8
 20  Item_Type_Combined_0        8523 non-null   uint8
 21  Item_Type_Combined_1        8523 non-null   uint8
 22  Item_Type_Combined_2        8523 non-null   uint8
 23  Outlet_0                    8523 non-null   uint8
 24  Outlet_1                    8523 non-null   uint8
 25  Outlet_2                    8523 non-null   uint8
 26  Outlet_3                    8523 non-null   uint8
 27  Outlet_4                    8523 non-null   uint8
 28  Outlet_5                    8523 non-null   uint8
 29  Outlet_6                    8523 non-null   uint8
 30  Outlet_7                    8523 non-null   uint8
 31  Outlet_8                    8523 non-null   uint8
 32  Outlet_9                    8523 non-null   uint8
dtypes: float64(4), int64(1), object(2), uint8(26)
```

Model Building

The Performance Measure for Regression problems is RMSE:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i-1}^{n}\left(\left|\mathrm{x}_{\mathrm{predict}} - \mathrm{x}_{\mathrm{actual}}\right|\right)^2}$$

This has to be minimized.

1. Multiple Linear Regression: A model which create a linear relationship between the dependent variable and one or more independent variable, mathematically linear regression is defined as:

$$y = w^T x$$

   Where y is the target variable and x are the independent variables

2. Ridge Regression: This method fits the training data with a bigger bias, so that the variance is less because of the tradeoff between bias and variance. This ensures that the fitted line has coefficients which are less sensitive to the data points. The cost function for this method is defined as:

$$min\left(|(Y - X(\theta)|)^2 + \lambda \|\theta\|^2\right)$$

Where lambda is the penalty term, higher its value bigger will be the penalty. Controlled in the code by the parameter alpha.

3. PCA: We have used Principal Component Analysis for Feature Extraction to see if we get better performance metrics. But before this, feature scaling had to be performed. We obtained a variance ratio for all principal components for which a Scree Plot can be made:

0	0.179419
1	0.134807
2	0.129997
3	0.126102
4	0.0957349
5	0.0672351
6	0.0538451
7	0.051503
8	0.0436192
9	0.0366068
10	0.0229473
11	0.0168713
12	0.0147598
13	0.014167
14	0.0123859
15	5.12536e-33
16	3.16673e-33
17	2.30885e-33

4. Cross-Validation: We performed a 20 fold Cross Validation for gauging the performance of our models

Results

1. Multiple Linear Regression

```
Model Report
RMSE : 1127.4397896331782
CV Score: Mean-1128.6604942911842 | Std-43.46889549208598 | Min-
1075.0950737053818 | Max-1210.1833665555916
```

2. Ridge Regression

```
Model Report
RMSE : 1128.50978732565
CV Score: Mean-1129.6546534034967 | Std-44.74134707905992 | Min-
1075.6032944593242 | Max-1217.1588675967935
```

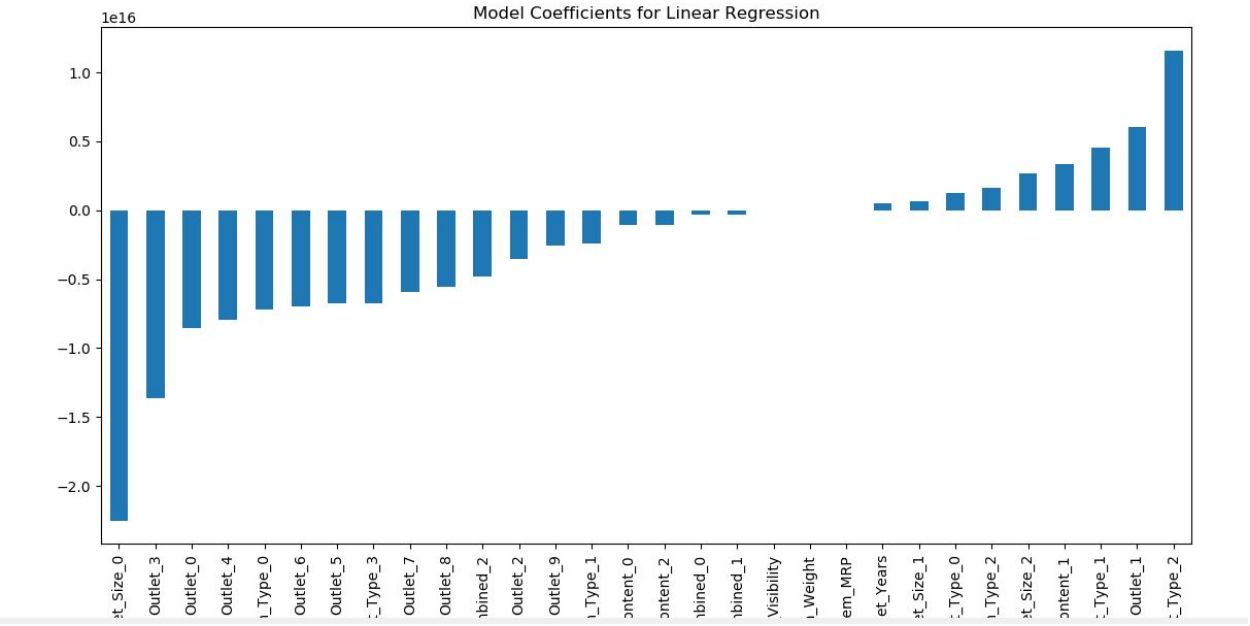
3. Linear Regression after PCA

```
Model Report
RMSE : 1127.4133151169722
CV Score: Mean-1128.6843512359603 | Std-43.50147217333108 | Min-
1074.7591385193184 | Max-1210.3636899596106
```

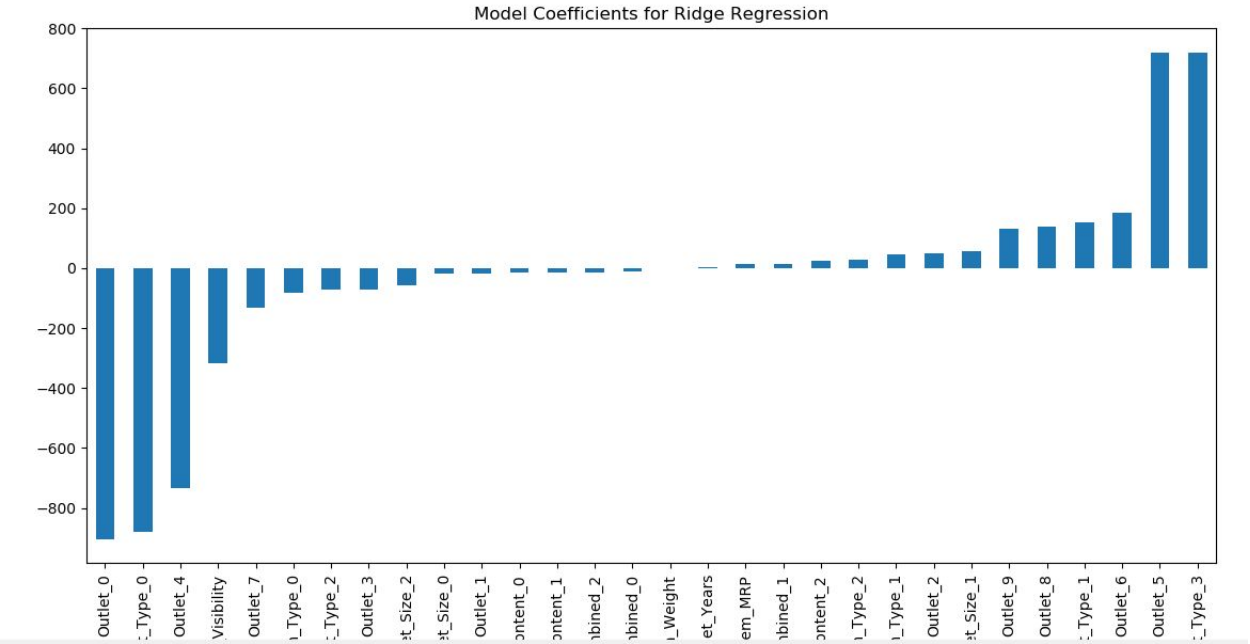
4. Ridge Regression after PCA

```
Model Report
RMSE : 1127.4133153866837
CV Score: Mean-1128.6842919403025 | Std-43.50170483752371 | Min-
1074.759418787128 | Max-1210.3647348398556
```

5. Model Coefficients for Linear Regression



6. Model Coefficients for Ridge Regression



Conclusions

1. PCA with lesser components resulted in a worse performance because of the fact that some important components were discarded.
2. Feature Extraction resulted in a tiny bit better result, but not a very good result.
3. Linear Regression after PCA gave the best result with the least RMSE.

Scope Of Improvement

1. Taking into consideration the value of Adjusted R squared would improve feature selection as it changes when new features are added or removed in backward selection.
2. Hyperparameter Tuning of the parameters of our model using Cross Validation or other methods would result in better accuracy.
3. Using models like Decision Trees and XGBoost and other methods of Ensembling would result in way better accuracy.

   We worked on this project whole-heartedly and did what we could given the circumstances. These improvements are not just written for the sake of writing, we will learn these methods and continue with the project. It as a very good learning experience.

References

1. https://www.researchgate.net/publication/336530068_A_Comparative_Study_of_Big_Mart_Sales_Prediction
2. https://medium.com/diogo-menezes-borges/project-1-bigmart-sale-prediction-fdc04f07dc1e