BigMart Sales Prediction.

* Problem Statement:

Sales Prediction for Big Mart Outlets:

The data collected in 2013 annual sales for 1559 products across 10 stores in different cities.

Also, certain attributes of each product and store have been defined. The aim is to build a predictive model and predict the sales of each product at a particular outlet.

Using this model, BigMart will try to understand the properties of products and stores, which play a crucial role in increasing sales and developing better business strategies.

* Technical Stack:
  + Language: Python
  + Libraries: Pandas, NumPy, Matplotlib, Sklearn
* Dataset:

|  |  |
| --- | --- |
| Variable | Definition |
| Item\_Identifier | Unique Products ID |
| Item\_Weight | Weight of product |
| Item\_Fat\_Content | Whether the product is low fat or regular. |
| Item\_Visibility | The % of the total display area of all products in a store allocated to each product. |
| Item\_Type | The category to which the product belongs. |
| Item\_MRP | Maximum retail price of the product. |
| Outlet\_Identifier | Unique store ID. |
| Outlet\_Establishment\_Year | The year in which the store was established. |
| Outlet\_Size | The size of the store in terms of group 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 particular store. This is the outcome variable to be predicted. |

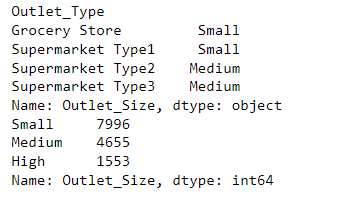
* Problem Statements [EDA]:
  + Graphical user interface, text

    Description automatically generatedIs ‘Item\_Fat\_Content’ might affect the production of the sales?
  + ‘Item\_Type’ column may vary.

Graphical user interface, text

Description automatically generated with medium confidence

* + There are four categories in ‘Item\_Fat\_Content’, which might be changed.
  + Is the ‘Outlet\_Type’ column important for the model building?



* + ‘Item\_Identifier’ is the column with so many different categories we can see, but do we need that many categories in our dataset
  + How did "Outlet\_Establishment\_Year" affect the production of the sales?

Graphical user interface, application, Teams

Description automatically generated

* + could the Size of the store influence the item sales at a particular store?
  + Is the Location of the Outlet important for the Item Outlet Sales?
  + How “Item\_weight” can affect sales?
* Required libraries:
  + Pandas, NumPy, Matplotlib, Seaborn, Sklearn
* Data Pre-processing:
  + Finding the null values.
  + Checking the unique values.
  + Sanity Check & Frequency check for the columns which have categorical values.
    - "Item\_Fat\_Content": combine names for the different categories,
    - "Item\_Type" & "Outlet\_Size": handling missing values from the two different columns.,
    - ‘Item\_Weight’: where we will handle missing values by using aggregate and groupby functions.
    - “Item\_Identifier”: we can merge the category by using replace function from one column only.
* Feature Engineering:
  + In the "Outlet\_Establishment\_Year" column we will drop the column but before that, we will also create a new column that will about mention how many years old the shop is.
  + Converting all the zero values to mean in the “Item\_VIsiblity” column.
  + Creating a correlation table for the difference between numerical values.
* Outliers:
  + As we will concatenate the train and test data, we should predict the “Item\_Outlet\_Sales” column from the training dataset as test data already have that column. So, the outlier will be fixed by predicting the sales.

Chart, bar chart

Description automatically generated

* Graphs and Plots:
  + Using scatter plot and histogram plot we should analyze the Sales of the product and get some idea of how sales price (dependant value) can work with the independent values like item type, location, or fat content of the product.

Chart, scatter chart

Description automatically generated

* + After plotting to analyze the product sales, we should apply correlation or heat map, as the correlation between variables indicates that as one variable changes in value, the other variable tends to change in a specific direction. Understanding that relationship is useful because we can use one variable's value to predict the other variables' value.

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

Chart, histogram

Description automatically generated

* + In our case, it should be sales as the relationship between the independent variable and the dependent variable is positive or negative or moderate positive or negative.
  + Correlation can measure only int or float values which can range from -1 to 1. And for that, we should convert string values into int or Float values with the help of ‘Label Encoding’ as independent variables are categorical values.

A picture containing chart

Description automatically generated

* Model Selection:
  + Now, we will separate data into test & train with the ratio of 80% & 20%.
  + 80% data for the training dataset and 20% will be test data for the basic model selection.
  + For the model selection will import some modules from the sklearn libraries.
  + Here, we will use the Cross Validation Score model as a basic\_model.
  + We get the MSE Score and R2 score from the basic model selection.

Graphical user interface, text, application, email

Description automatically generated

* Standardization:
  + We will Standardize the model before training the dataset because the independent variable is a simple method to reduce multicollinearity that is produced by higher-order terms.
  + Standardizing the independent variable can also help to determine which variable is the most important.
  + For Standardization will use the “StandardScaler” module.

Text

Description automatically generated

* Robust Scaler:
  + Robust Scaler removes the median and scales the data according to the quantile range. IQR (Interquartile range) between the 1st quartile (25th quartile) and the 3rd quartile (75th quantile). but here our range will be 0.1 and 0.8.
  + This Scaler also handles the Outliers and their Scales according to the quantile range.
  + For the Robust Scaler we will Normalize our data using ‘MinMaxScaler’.

Text

Description automatically generated

* Best Model:
  + For getting the best model we will compare all models using RMSE Score (Root Mean Squared Error).
  + RMSE is the standard deviation of the *residuals* (prediction Error).
  + It will tell how concentrated the data is around the *line of best fit*.
  + Using metrics, we get the Intercept and coefficient values.
  + We are getting MAE Score, MSE Score, RMSE Score, and R2 Score for the model.

A picture containing scatter chart

Description automatically generated

* + With the rest robust normalization, we will predict the final sales of the product.

Table

Description automatically generated