Big Mart Sales Prediction Analysis

Objectives:

The aim of this project is to:

- Analyze and explore the data available on Big Mart to find out:
  - Which product people bought more low fat or regular fat?
  - Which product have more profitable the low fat or regular fat?
  - What kind of product people bought more?
  - What product have most profitable?
  - What is the most outlets size of big mart in our data?
  - What is the most profitable outlet size?
  - Which type of city (tire1, tire2, tire3) have most profitable?
  - Dose the display area of products effects on the sales?
  - Dose items with higher MRP (Maximum Retail Price) sold more?
  - Which outlet have the top sales?
- Create a model that can predict the sales per product for each store. To help Big Mart to understand the properties of products and stores which play a key role in increasing sales.

Design:

Big Mart is a big supermarket that sales 1559 products across 10 stores in different cities.

First, I download the data from Kaggle website and explore it to understand its features and data type. Then I do some data cleaning. After that I showed the data to answer the questions above and analyze target variable on features.

Second, I do some data preprocessing, then I implement a regression model to predict my target.

Data:

This dataset was taken from Kaggle website (BigMart Sales Data | Kaggle).

The dataset contains 8523 entries and 12 features,it is available as (.csv) file.

Algorithms:

1- Replace NAN values in Item_Weight with the mean value sense it is numerical.
2- Replace NAN values in Outlet_Size with the mode value sense it is categorical.
3- Replace the '0' values in Item_Visibility with the mean.
4- Replace Item_Fat_Content values (Low Fat,Regular ,LF ,reg ,low fat) to :Low Fat , Regular (for visualization)
5- Analyze and explore the data to answer the questions above.

6- Check the relationship between the target variable and others
7- Check the correlation between Numerical variables and target variable
8- Future engineering:
    a. Label encoding on: Item_Fat_Content, Outlet_Size, and Outlet_Location_Type.
    b. One hot encoder on: Item_Type and Outlet_Type.
9- Drop columns: Item_Identifier, Outlet_Identifier ,Outlet_Establishment_Year, Outlet_Type, Item_Type.
10- Implement a model to predict my target.

I used a **Regression Supervised Machine Learning algorithm**.

First, I split my data to (80% for train) / (20% for test) I fit the models on train set, and test on test set.
As you see these are the scores I got:

	Models	R^2
2	Degree 2 polynomial regression	0.595
1	Random Forest Regression	0.589
3	Lasso	0.563
0	Linear Regression	0.562

Second, I split my data to (80% for train / validation) / (20% for test) I fit the models on train set ,and test on validation and test set.

As you see these are the scores I got:

	Models	R^2
0	Degree 2 polynomial regression	0.612
2	Random Forest Regression	0.584
4	Linear Regression	0.574
1	Cross-Validation	0.558
3	Random Forest with Hyper Parameter Tuning	0.556
5	Ridge Regression	-7536.000

So, the best model was **degree 2 polynomial** with R^2 =0.612

Tools:

The work was done through ***Jupyter*** notebook using ***python***.

1- ***NumPy, pandas*** for discovering and cleaning the data.
2- ***matplotlib and seaborn*** for showing the data.
3- ***sklearn*** for building and training the model.