Detailed Project Report

Stores Sales Prediction

Written By	Shaik Imran Fazil Bandi Lokesh O Dinesh Kumar Shaik Chetansha
Version	1.0
Date	07-02-2023

Document Change Control Record

Version	Date	Author	Comments

Reviews

Version	Date	Reviewer	Comments

Approval Status:

Version	Review date	Reviewed By	Approved By	Comments

Index

1. Introduction

1.1 Abstract

A class of methods known as "machine learning" enables software applications to predict outcomes more accurately without having to be explicitly coded. Building models and using algorithms that can take input data and use statistical analysis to predict an output while updating results as new data becomes available is the fundamental tenet of machine learning. These models can be used in many contexts and taught to meet management expectations so that precise actions can be done to meet the organization's goal. In order to forecast the sales of various things and comprehend the influences of various elements on the sales of the items, the instance of Big Mart, a one-stop shopping mall, has been examined in this paper. examining multiple dataset components High levels of accuracy are produced from the data collected for Big Mart and the method used to develop a predictive model, and these observations can be used to inform decisions about how to increase sales.

1.2 Machine Learning

The amount of data available is growing daily, and this massive volume of unprocessed data must be carefully evaluated in order to produce outcomes that meet the current standards for being highly useful and finely pure. It is accurate to claim that Machine Learning (ML) is evolving at a rapid rate, just as Artificial Intelligence (AI) has over the previous 20 years. ML is a significant pillar of the IT industry and, as a result, a significant, if typically unnoticed, aspect of our lives. Data is incredibly useful in current aspects, therefore as technology advances, so will the analysis and interpretation of data to produce effective results.

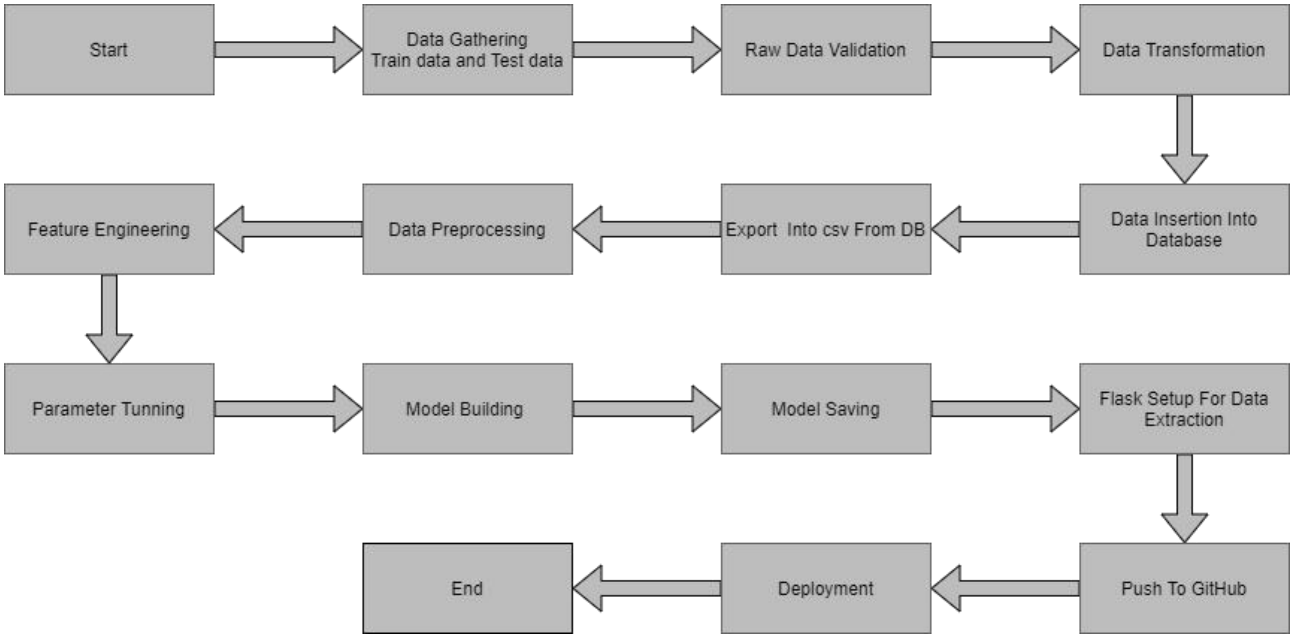
Both supervised and unsupervised types of tasks are dealt with in machine learning, and often a classification-type problem serves as a source for knowledge acquisition. The main focus is on developing a system self-efficient so that it can perform computations and analysis to produce much more accurate and precise results. It creates resources and uses regression to make precise predictions about the future. Data can be transformed into knowledge by applying statistical and probabilistic algorithms. Sampling distributions are used as a conceptual foundation for statistical inference.
ML can take many different forms. First, numerous ML applications and the kinds of data they work with are explored in this study. The problem statement that is the focus of this work is then codified.

1.3 Problem Statement

"To understand Big Mart sales in order to determine what impact specific qualities of an item play and how they affect their sales."
A predictive model that calculates the sales of each item for each store can be constructed to assist Big Mart in achieving this objective. Moreover, the most important variables that can boost their sales and what product or store qualities might be altered.

2. Architecture:

The project was completed using the following workflow.



2.1 Data gathering:

Bigmart sales data can be seen at https://www.kaggle.com/brijbhushannanda1979.
Data in.csv format is used to store Train and Test data.

2.2 Raw Data Validation:

Before moving on with any operation, multiple sorts of validation must be performed on the loaded data. Validations include ensuring that all of the columns have a standard deviation of zero and ensuring that no columns have any complete missing data. These are necessary because the qualities that include them are useless. It won't have any impact on an item's sales at the relevant stores.
For example, if an attribute has zero standard deviation, all of its values are the same and its mean is zero. This suggest that regardless of whether sales go up or down, the attribute will remain the same. Similar to this, it serves no purpose to take any property into account when operating if all of its values are missing. Increasing the likelihood of the dimensionality curse is needless.

2.3 Data Transformation

It is necessary to alter the data before sending it to the database so that it may be inserted into the system without difficulty. Here, the characteristics "Item Weight" and "Outlet Type" include the missing values. They are therefore filled out with the relevant data types in both the train set and the test set.

2.5 New Feature Generation

We can construct new mrp categories as mrp bins by deriving new item categories from item types.

2.6 Data preprocessing

All the steps necessary before delivering the data to be used in model construction are carried out

during data preparation. The 'Item Visibility' characteristics, for example, have certain values equal to 0, which is inappropriate given that if the item is available on the market, how can its visibility be 0? In its place, the average value of the item visibility for the relevant "Item Identifier" category has been used. A new feature called "Outlet years" was added, which subtracts the provided establishment year from the current year. With the addition of a new "Item Type" attribute, the types of the items are now indicated by the first two characters of the item identifier. The mapping of "fat content" is then carried out based on "low," "regular," and "non-edible."

2.7 Feature Engineering:

It was discovered after preprocessing that certain of the attributes are not crucial to the item sales for the specific retailer. Therefore, their qualities are dropped. To turn the categorical data into numerical features, even one hot encoding is carried out.

2.8 Parameter tunning:

With the help of Randomized searchCV, parameters are tuned. Four algorithms—Linear Regression, Gradient Boost, Random Forest, and XGBoost regressor—are employed to solve this problem. All four of these algorithms' parameters are adjusted and sent to the model.

2.9 Model building:

Data set is passed into all four models, Linear Regression and Random Forest, after executing all of the preparation processes mentioned above, scaling, and hyper parameter tuning. The results show that the Random Forest Regressor performs best, with the lowest RMSE value (781.64) and the highest R2 score (0.55). Therefore, "Random Forest" did well in this problem.

2.10 Model saving:

The model is then stored in.sav format using the pickle library.

2.11 Git Hub

A push to the GitHub repository will include the whole project directory.

2.12 Deployment:

The Heroku cloud platform was used to deploy the project from the set up cloud environment.
Link to the application: https://bigmartsalesprediction.herokuapp.com

3. Data set description

The data scientists at Big Mart gathered sales information from 10 of their stores, each of which had 1559 unique products as of the data collection. These stores were spread out across different locales. With the help of all the data, it can be deduced what function specific attributes of an item serve and how they impact sales. The dataset appears as follows:

```
data=pd.read_csv("/content/Train.csv")
data.head()
```

	Item_Identifier	Item_Weight	Item_Fat_Content	Item_Visibility	Item_Type	Item_MRP	Outlet_Identifier	Outlet_Establishment_Year	Outlet_Size	Outlet_Location_Type	Outlet_Type	Item_Outlet_S
0	FDA15	9.30	Low Fat	0.016047	Dairy	249.8092	OUT049	1999	Medium	Tier 1	Supermarket Type1	3735.
1	DRC01	5.92	Regular	0.019278	Soft Drinks	48.2692	OUT018	2009	Medium	Tier 3	Supermarket Type2	443.
2	FDN15	17.50	Low Fat	0.016760	Meat	141.6180	OUT049	1999	Medium	Tier 1	Supermarket Type1	2097.
3	FDX07	19.20	Regular	0.000000	Fruits and Vegetables	182.0950	OUT010	1998	NaN	Tier 3	Grocery Store	732.
4	NCD19	8.93	Low Fat	0.000000	Household	53.8614	OUT013	1987	High	Tier 3	Supermarket Type1	994.

As seen in Fig., the data collection includes a variety of data kinds, including integer, float, and object.

```
[ ] data.info()
```

```
<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
```

There may be several kinds of underlying patterns in the raw data, which can potentially provide insights into the issue and in-depth knowledge about the subject of interest. However, attention should be exercised when dealing with data because it could include null values, redundant values, or different sorts of ambiguity, which necessitates pre-processing of the data. Therefore, a dataset should be investigated as thoroughly as feasible.

The following table illustrates various statistically significant factors for numerical properties, including mean, standard deviation, median, count of values, maximum value, etc.

```
[ ] data.describe()
```

	Item_Weight	Item_Visibility	Item_MRP	Outlet_Establishment_Year	Item_Outlet_Sales
count	7060.000000	8523.000000	8523.000000	8523.000000	8523.000000
mean	12.857645	0.066132	140.992782	1997.831867	2181.288914
std	4.643456	0.051598	62.275067	8.371760	1706.499616
min	4.555000	0.000000	31.290000	1985.000000	33.290000
25%	8.773750	0.026989	93.826500	1987.000000	834.247400
50%	12.600000	0.053931	143.012800	1999.000000	1794.331000
75%	16.850000	0.094585	185.643700	2004.000000	3101.296400
max	21.350000	0.328391	266.888400	2009.000000	13086.964800

To ensure that analysis and model fitting are accurate, preprocessing of this dataset entails performing analysis on the independent variables, such as checking for null values in each column and then replacing or filling them with supported relevant data types. Some of the representations created using Pandas tools, which provide information on model values for categorical columns and variable counts for numerical columns, are displayed above. Deciding which value to prioritise for further investigation activities and analysis depends on the maximum and minimum values in numerical columns as well as their percentile values for the median. During the model building process, data types from various columns are also employed for label processing and one-hot encoding.

4. Implementation and Results

This section discusses the programming language, libraries, implementation platform, data modelling, and the observations and outcomes that resulted from it.

4.1 Implementation Platform and Language

Python is a general-purpose, interpreted-high level language that is frequently used in modern times to solve domain problems rather than handle system complexity. For programming, it is also known as the "batteries included language." It has a number of libraries utilised for scientific research and inquiries as well as a number of libraries from other parties to facilitate effective

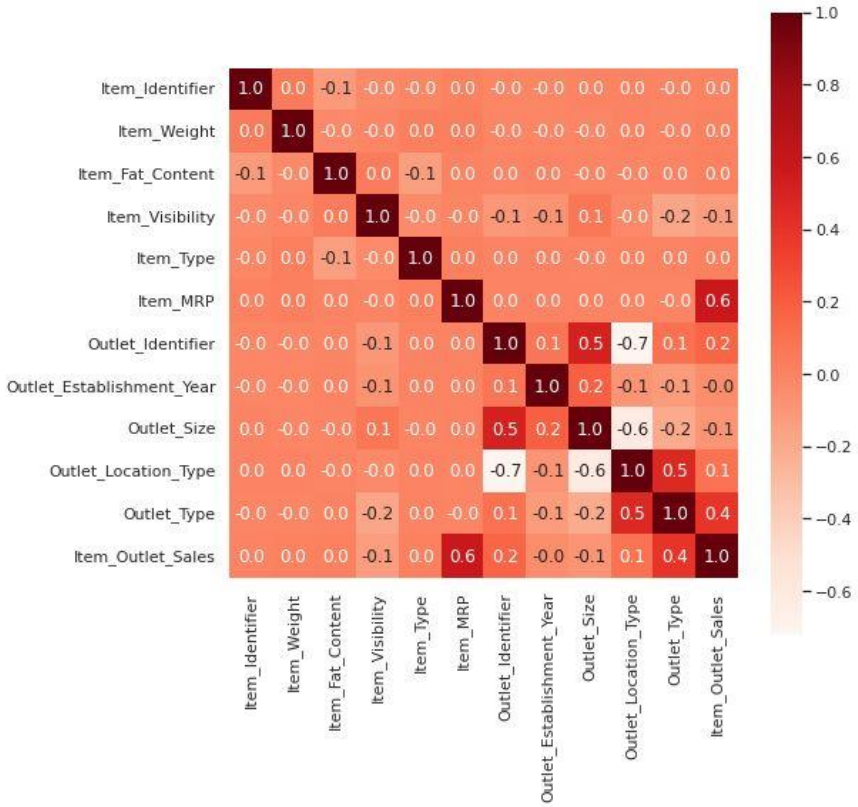
issue solutions.

The Python libraries Numpy and Matplotlib have been utilised in this work for scientific computation and 2D visualisation, respectively. Additionally, the Python Pandas tool has been used to conduct data analysis. To complete jobs by assembling the random forest approach, utilise the random forest regressor. Jupyter Notebook has been utilised as a development environment because it excels at "literate programming," where human-friendly code is interspersed within code blocks.

4.2 Correlation

```
[ ]  corr = data.corr()
     plt.figure(figsize=(8,8))
     sns.heatmap(corr,cbar=True,square=True,fmt='.1f',annot=True,cmap='Reds')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x7f9ffeb1bb50>
```



- The relationship between item visibility and the dependent variables item outlet sales and grocery store outlet type is essentially nonexistent. This suggests that product visibility has no impact on sales, which runs counter to the common belief that "greater visibility equals more sales."
- The maximum retail price (Item MRP) is positively connected with sales at an outlet, suggesting that the price set by the outlet affects sales.
- The differences in MRP that different retailers charge rely on their specific sales.

4.3 Metrics for Data Modelling

- The correlation between two variables R2 (R-squared) is a statistic that gauges how well a model fits the data, or how closely the predictions of regression come close to the actual data points. The value 1 of R2 indicates that regression predictions fully match the real data points, and higher values of R2 show stronger model successes in terms of prediction coupled with accuracy. The application of improved R2 measurements produces even better outcomes. The dataset's target column's logarithmic values turn out to be important for the prediction procedure. Therefore, it may be argued that improved results can be determined by adjusting the columns used in the prediction. Incorporating correction may have also been accomplished by calculating the square root of a column. Additionally, because the target variable's square root tends to have a normal distribution, it makes the dataset and target variable easier to visualise.
- An essential metric during the estimating phase is the measuring of error. For measuring the correctness of continuous variables, the terms Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) are frequently employed. It may be claimed that both MAE and RMSE can be used to express the average model prediction error in terms of the variable of interest. The MAE, which gives equal weight to each individual difference, is the average of the absolute disparities between forecast and actual observation over the test sample. The term "RMSE" refers to the square root of the average of the squared discrepancies between the prediction and the actual observation. R2 is a relative measure of fit, whereas RMSE is an absolute value. RMSE aids in calculating the average error of the variable and is also a quadratic grading scheme. Better model fitting results from low RMSE values acquired for linear or multiple regression.
- Regarding the findings of this study, it can be concluded that there isn't much of a difference between our train and test sample because the metric RMSE ratio is determined to be identical to the ratio of train to test sample. RMSE is a good indicator, along with measuring precision and other necessary characteristics, of how well responses are anticipated by our model, and the outcomes can be deduced from them. By incorporating outlier detection and high leverage points into future data exploration,

a significant improvement could be accomplished. Combining numerous low-dimensional, theoretically simpler sub-models that are simple to verify by subject matter experts is another strategy that can be used. This technique is known as ensemble learning.

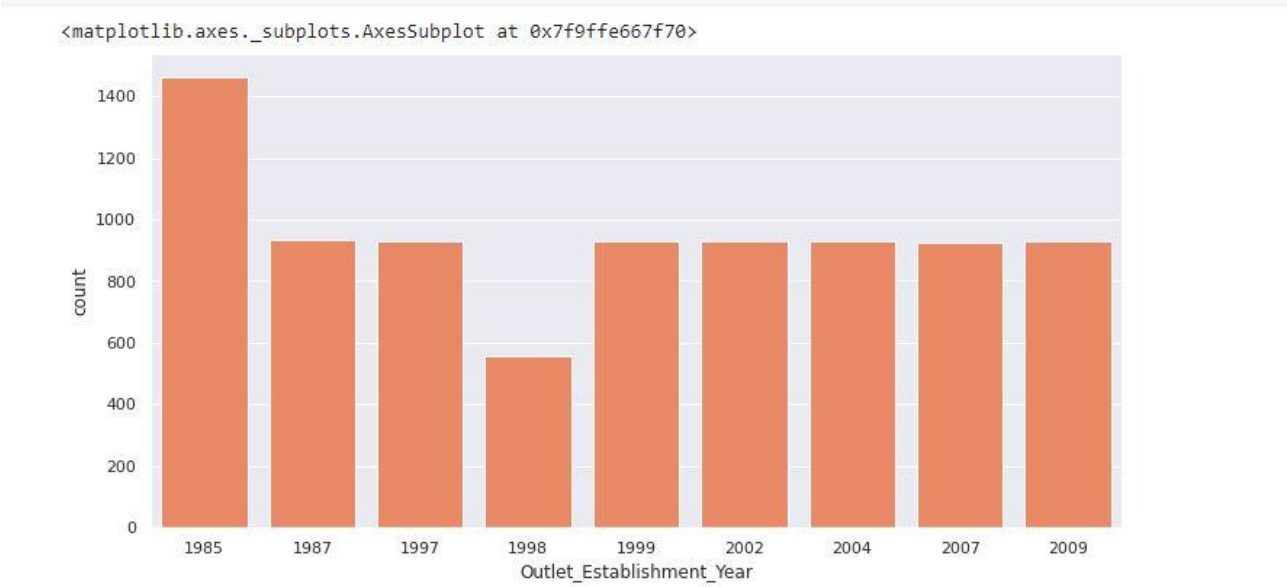
4.4 Prediction results

- The biggest location did not result in the most sales. The site at OUT027, a Supermarket Type3 with a size that was identified in our dataset as medium, generated the highest sales. You may say that this outlet performed significantly better than any other outlet site in the dataset under consideration for any size.
- Item Outlet Sales, the target variable, has a median value of 3364.95 for the OUT027 location. A median value of 2109.25 was found at the site (OUT035) with the second-highest median score.
- For the Gradient boost model, adjusted R-squared and R-squared values are higher than usual. Additionally, compared to other models with the greatest CV score, its RMSE value is low. As a result, the gradient boost model is more accurate and fits the data better.

5. Conclusion

The fundamentals of machine learning and the related data processing and modelling techniques have been covered in this project, followed by an application of these concepts to the problem of predicting sales in several Big Mart shopping complexes. The predicted results after implementation indicate the relationship between the many factors taken into account and how a specific site of a medium size recorded the best sales, implying that additional retail locations should adopt a similar strategy for increased sales.

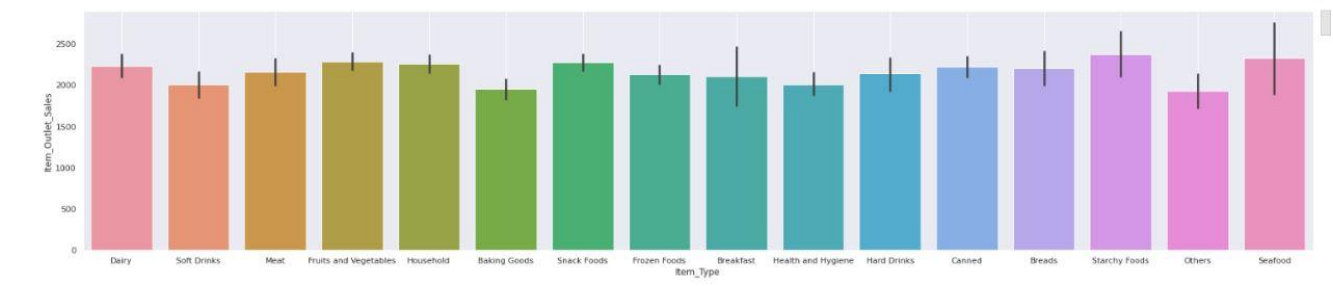
Additionally, it may be determined that more sites should be upgraded to Tier 3 under the "Supermarket Type 3" outlet type in order to boost product sales at Big Mart. This approach can help any one-stop-shopping mall, like Big Mart, by predicting how many of its products will sell in the future at various locations.

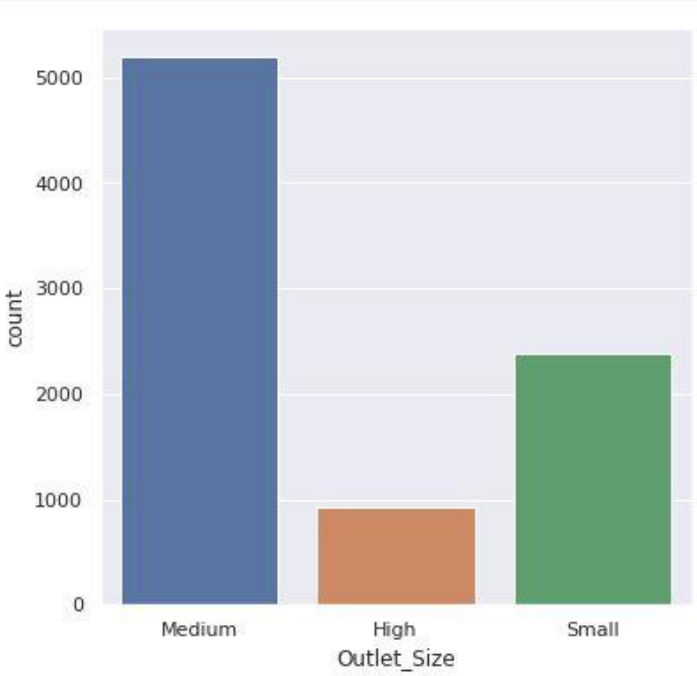


6. Future Scope

To increase the originality and success of this sales prediction, many instances parameters and other elements can be used. With more factors being employed, accuracy—which is crucial in prediction-based systems—can be considerably improved. Additionally, understanding the operation of the sub-models can boost system productivity. For increased usability, the project can be further developed in a web-based application or in any device supported by an in-built intelligence thanks to the Internet of Things (IoT). In order to develop more exact results that are closer to actual world circumstances, many stakeholders involved with sales information can also contribute more inputs to aid in hypothesis formulation.

The old approaches could be observed to have a greater and more beneficial impact on the overall development of a corporation's tasks when paired with efficient data mining methods and features. One of the key benefits is that the regression outputs are more expressive and, to a certain extent, more intelligible. Additionally, variations can be added to the suggested strategy to boost its adaptability at a crucial point in the regression model-building process. Additional experiments are required for accurate resource efficiency measures in order to properly assess and optimise.





7. Q & A:

Q1) What is the data's source?

Ans. he client supplies the following URL:
https://www.kaggle.com/brijbhushannanda1979/bigmart-sales-data

Q 2) What kind of data was it?

Ans. The information included both numerical and categorical values.

Q 3) What was the exact process you used for this project?

Ans. Regarding this, see the section on architecture.

Q 4) What should you do with incompatible files or files that failed the file validation after it is complete?

Ans. These files are transferred to the Achieve Folder, and we eliminated the faulty data folder after sharing a list of these files with the customer.

Q 5) What methods did you employ for pre-processing the data?

- • Eliminating undesirable traits
- • Visualizing the relationship between independent variables and the outcomes variables Distribution of continuous values can be checked and modified.
- • Eliminating outliers
- • Cleaning up data and imputing if there are any null values.
- • Transforming numerical values from category data.

- Data scaling

Q 6) How training was done or what models were used?

- • We used clustering over fit to separate the data into clusters before dividing the training and validation sets.
- The training and validation data were split up according to cluster.
- The data used for training and validation were scaled.
- Linear regression, Gradient boost, Random forest, and XGBoost were among the algorithms utilised.

Q 7)How was prediction made?

Ans. The client shares the testing files. To get the forecast, we feed its data into the best model we have kept in pickle format.

Q 8) Where was the model used?

Ans. We launch the model on the Heroku platform once it is complete. This model is an online application that allows users to enter data, which are then extracted in the backend and used to make predictions for the user.