Big Mart Sales Prediction Using Machine Learning

Objective

The main objective of this project is to develop a predictive model to estimate the sales of various products across different stores in the Big Mart dataset. This helps the business understand patterns and make informed decisions regarding inventory, marketing, and strategy. The project goals include:

- Analyzing and preprocessing the Big Mart dataset.
- Building and training machine learning models.
- Evaluating model performance using appropriate metrics.
- Deploying the best-performing model using Streamlit to create an interactive web application.

Dataset Used

Dataset: Big Mart Sales Dataset
Source: https://www.kaggle.com/datasets/brijbhushannanda1979/bigmart-sales-data

Features in the Dataset:

- **Item_Identifier:** Unique ID for each product.
- **Item_Weight:** Weight of the product.
- **Item_Fat_Content:** Low Fat or Regular.
- **Item_Visibility:** The percentage of the product's visibility in the store.
- **Item_Type:** Category of the product.
- **Item_MRP:** Maximum Retail Price of the product.
- **Outlet_Identifier:** Unique ID for each store.
- **Outlet_Establishment_Year:** Year the store was established.
- **Outlet_Size:** Size of the store (Small/Medium/High).
- **Outlet_Location_Type:** Tier of the city (Tier 1/2/3).
- **Outlet_Type:** Grocery Store or Supermarket.

Target Variable:

- **Item_Outlet_Sales:** Sales of the product at a particular store.

Model Chosen

Several machine learning models were evaluated, including:

- **Linear Regression:** For baseline performance.
- **Decision Tree Regressor:** To capture non-linear relationships.
- **Random Forest Regressor:** An ensemble tree-based model.
- **XGBoost Regressor:** An optimized gradient boosting model known for speed and performance.

Selected Model:

XGBoost Regressor was chosen due to its superior performance in handling:

- High-dimensional and sparse data.
- Missing values internally.
- Non-linear relationships in data.

Its regularization capabilities also help reduce overfitting, making it well-suited for structured data like the Big Mart dataset.

Performance Metrics

As this is a regression problem, the following metrics were used:

- **R² Score (Coefficient of Determination):** Indicates how well the predictions approximate the actual values.
- **Mean Squared Error (MSE):** Measures the average squared difference between actual and predicted values.
- **Root Mean Squared Error (RMSE):** The square root of MSE, giving a more interpretable error value.

Reported Accuracy :

- **Training R² Score:** 0.62
- **Testing R² Score:** 0.60
- **RMSE on Test Set:** 1029.89

Challenges & Learnings

Challenges:

1. **Handling Categorical Features:**
    - Categorical columns like Item_Type, Outlet_Type, and Item_Fat_Content required proper encoding using Label Encoding and One-Hot Encoding.
2. **Feature Engineering:**
    - Created new features (e.g., Item_Visibility_MeanRatio, Outlet_Age) to help the model learn better patterns.
3. **Model Tuning:**
    - XGBoost has many hyperparameters, and optimizing them through Grid Search or Randomized Search was time-consuming but essential for boosting accuracy.
4. **Deployment:**
    - Integrating the trained XGBoost model with Streamlit required saving the model using joblib and loading it correctly for prediction in the web app.

Learnings:

1. **Boosting vs Bagging:**
    - XGBoost (boosting) generally outperformed Random Forest (bagging) on this dataset by correcting previous errors iteratively.
2. **Feature Importance:**
    - XGBoost provides built-in tools to visualize which features most influence the model.
3. **Deployment Practice:**
    - Developing a Streamlit interface allowed for real-world simulation and improved usability for non-technical users.
4. **Business Insights:**
    - Understanding how factors like MRP, Outlet Type, and Item Visibility affect sales can guide marketing and inventory planning.