Big Market Sales Prediction

Data Analysis and Prediction Documentation

Problem Statement

The task is to predict the sales of various items across 10 different stores using historical data. The dataset consists of **1559 unique items** sold at **10 outlets**, making this a **supervised regression** problem with tabular data.

Data Characteristics

- **Total Unique Products**: 1559
- **Total Outlets**: 10
- **Most Frequent Product Occurrence**: Only 10 times
- **Observation**: High sparsity per product making it unsuitable for deep learning or high-variance models without risk of overfitting.

Approach and Methodology

1. Exploratory Data Analysis (EDA)

Conducted in 1.Exploratory Data Analysis-4.ipynb

- Identified key patterns across product categories, outlet types, item visibility, and MRP.
- Observed sparsity in product representation informing the need for robust, regularized models.

2. Feature Engineering

- We Iteratively created domain-specific features such as:
    - Mean_LogSale_per_Item, Mean_LogSale_per_Outlet, Item_Profile, Sales_to_MRP
    - Median sales across item-outlet, item-profile clusters
    - Deviation from city-tier MRPs and profile frequency bins
- Features evaluated using **SHAP values** and Recursive Feature Elimination (RFE).
- Final model trained on about 30 features, selected after rigorous SHAP inspection and cross-validation.

3. Modeling

Given the dataset size and feature characteristics, deep models were ruled out. Instead, we opted for:

- **XGBoost**: primary learner due to handling of sparse tabular data and regularization.
- **LightGBM**: faster, histogram-based boosting tree alternative.
- **Ridge Regression**: used as a meta-learner in stacked models.
- Neural networks / Deep learning: avoided due to insufficient data scale.

4. Validation Strategy

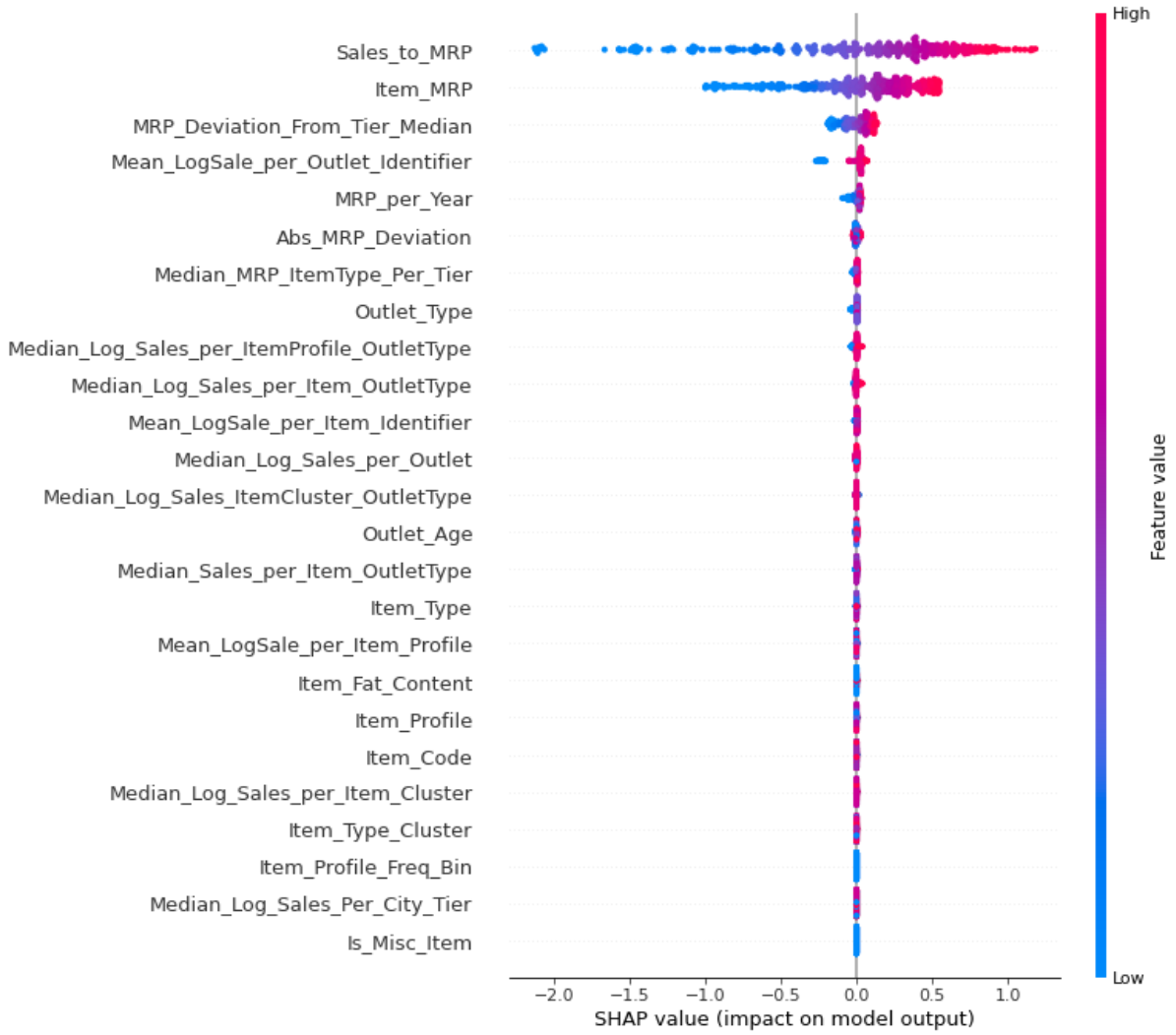
- Started with **Stratified Shuffle Split** using engineered frequency-based clusters
- Moved to **K-Fold Cross Validation** (k=5) to better capture variance in sales
- Applied **RandomizedSearchCV** for tuning XGBoost and LightGBM
- Evaluation Metric: **RMSE on inverse log-transformed sales**

Model Training

Implemented in:

- 2.Model_Training_XGB-4.ipynb: Local training, stratified sampling, SHAP analysis
- 2.Model_Training_XGB-4-Colab.ipynb: Colab-based training with GPU + KFold CV + RandomSearchCV
- 2_Model_Training_LightGBM_4_Colab: LGBM training
- All generated predictions are saved as:
- xgb_1_submission.csv
- xgb_2_submission.csv
- xgb_3_submission.csv
- xgb_4_submission.csv
- xgb_4_RCV_submission.csv
- xgb_final_stratified_submission_colab.csv
- xgb_kfold_submission_colab.csv
- ensemble_submission_1.csv
- lgbm_submission_colab.csv

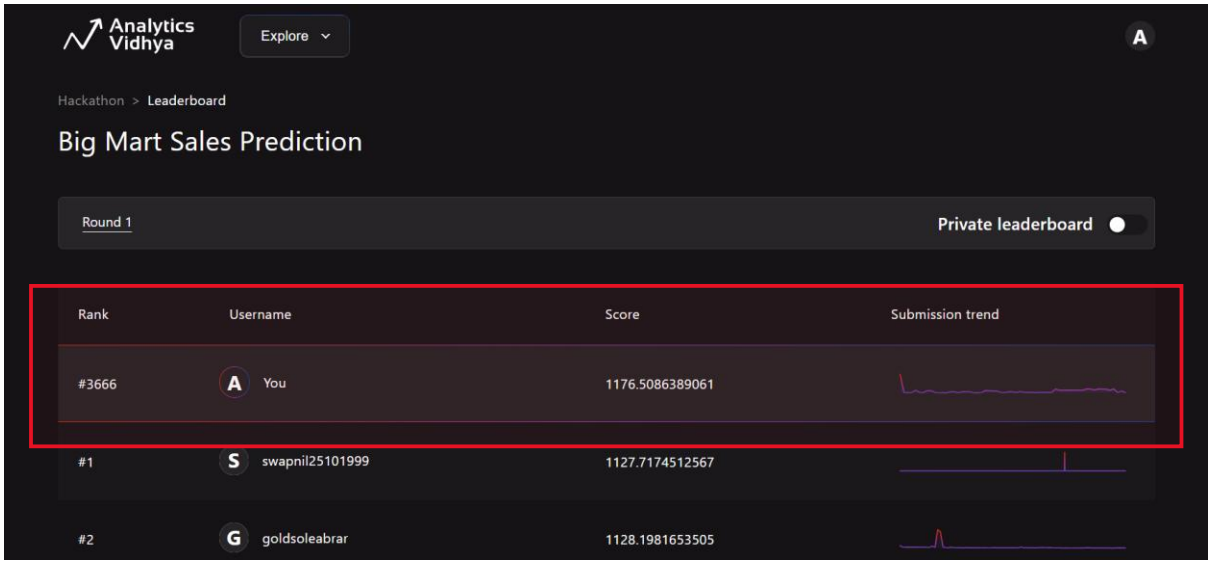
SHAP Analysis



- Top 5 features:
    - Sales_to_MRP
    - Item_MRP
    - MRP_Deviation_From_Tier_Median
    - Mean_LogSale_per_Outlet_Identifier
    - MRP_per_Year
- Several engineered features showed minor SHAP impact but were retained for potential interactions.
- SHAP summary plot revealed a long tail of low-impact features investigated using RFE pruning.

Conclusion:

- Traditional models, especially XGBoost was well-suited for this low-data problem.
- Strong feature engineering combined with SHAP-driven pruning improved model explainability.
- K-Fold CV and RandomSearchCV contributed to robust validation but gains saturated near **1176.5086389061 RMSE**.
- Stack ensemble showed potential, but further marginal improvement might require external data, feature embeddings, or boosting diversity.



Although the solution did not surpass the top leaderboard score, we effectively navigated the limitations of a sparse and imbalanced dataset through rigorous exploratory data analysis, thoughtful feature engineering, and the strategic use of robust models like XGBoost and LightGBM. Our approach was guided by interpretability using SHAP values, validated through K-Fold cross-validation, and reinforced by ensemble modelling techniques.

The best model achieved a leaderboard RMSE of **1176.50**, placing it competitively close to the top-performing solution. With further access to richer features (such as external demand drivers), there remains strong potential to close the performance gap even further.