CS 725- Foundations of Machine Learning: Credit Card Reward Maximization

Datasets:

credit_card_transactions (from kaggle)
cleaned_dataset_for_model (preprocessed)

Code: dataset_cleaning.ipynb

Preprocessing Steps:

1. Handled missing values.

2. Encoded categorical variables using (Label Encoding).

3. Standardized numerical variables using (tandardScaler).

Key Features:

Categorical Features: - `first`, `last`, `category`.

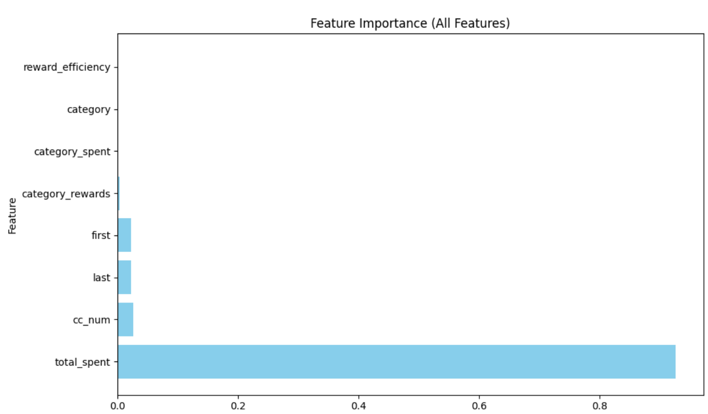
Numerical Features: - `cc_num`, `category_spent`, `reward_efficiency`, `total_spent`.

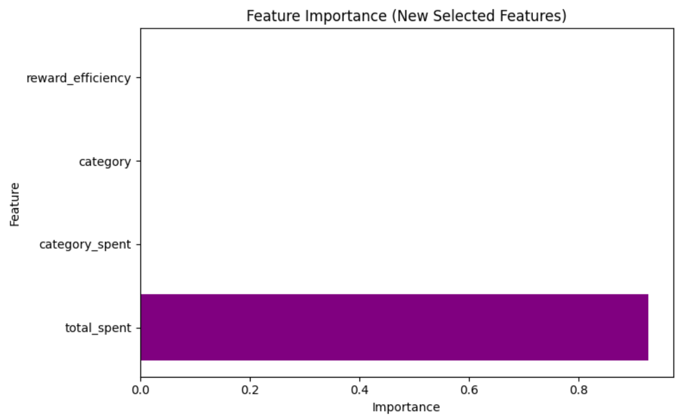
Target Variable: - `total_rewards` (predicted rewards for users).

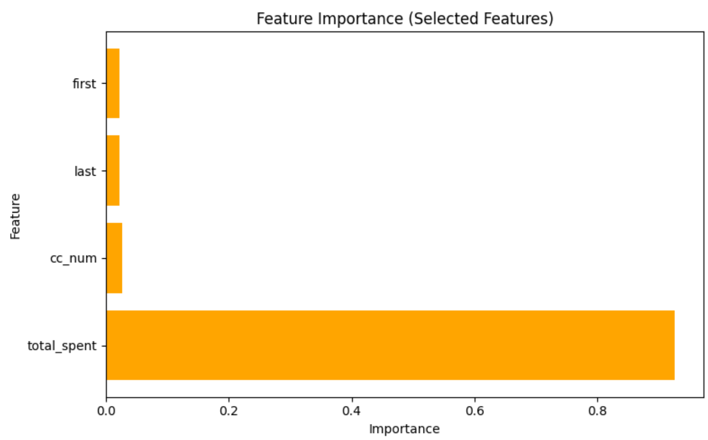
Feature Importance:

most impactful features influencing the model's predictions.

Code: feature_importance.ipynb







Model Training:

Code: model_training.ipynb

Features= ['first', 'last', 'cc_num', 'category', 'category_spent', 'reward_efficiency', 'total_spent']

Target= data['total_rewards']

Split: Divided the cleaned dataset into training (80%) and validation/test (20%) sets.

Model Selection: Random Forest Regressor, LightGBM, XGBoost.

Each model was trained on the prepared training set.

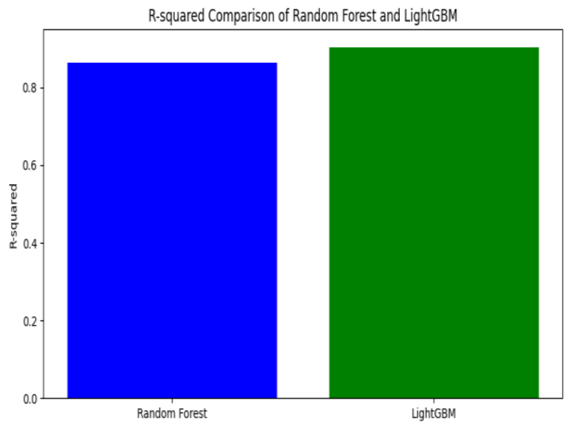
Computed predictions on the validation/test set.

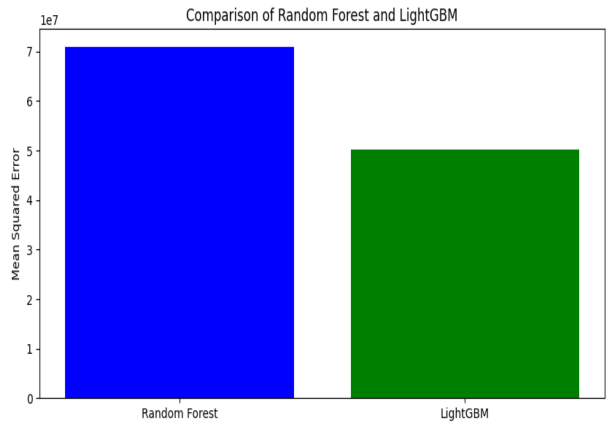
Cross-validation reduced the risk of overfitting and confirmed that the models generalized well.

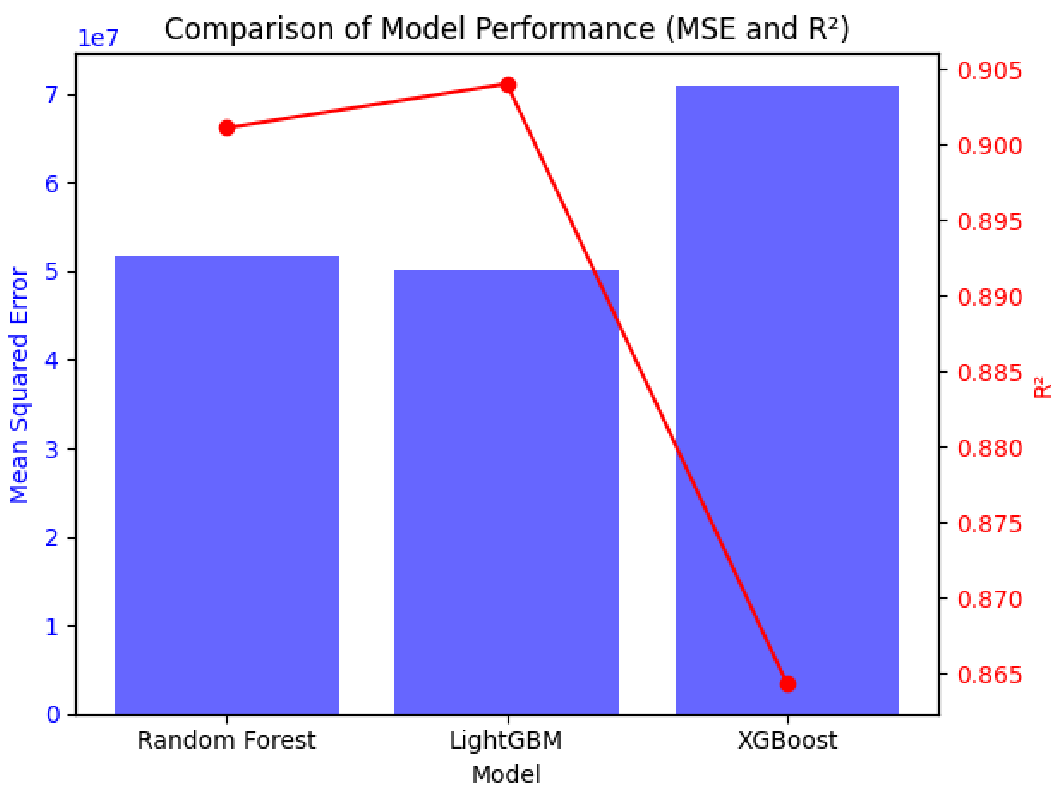
Hyperparameter tuning significantly improved LightGBM's performance, leading to the lowest MSE and highest R^2 values.

Evaluation Metrics: Compared model performance using the following metrics:

Mean Squared Error (MSE), R^2 (Coefficient of Determination)







LightGBM was the optimal model for predicting credit card rewards due to its balance of speed and accuracy.