IML Hackathon 2024

HU.BER–Optimizing Public Transportation Routes

Task 1: Predicting Passenger Boardings at Bus Stops

In this task, our objective was to predict the number of passengers boarding a bus at a given stop. We explored several regression models and techniques to improve the prediction accuracy, using **Mean Squared Error (MSE)** as the evaluation metric.

Initial Approach: Linear Regression

We began with a simple Linear Regression model with non-negative constraints to prevent predicting negative values. However, despite the constraints, the model exhibited underfitting, as the training data size increased but the model's ability to generalize remained poor, reflected in high MSE (3.17) and low R² (0.29) scores.

Improving the Model: Poisson Regression

Recognizing that passenger counts are non-negative integers, we implemented Poisson Regression, a model specifically designed for count data. After scaling the features using StandardScaler, the model achieved a Mean Squared Error (MSE) of 3.02 and an R² score of 0.53, indicating moderate performance. While this approach provided more realistic predictions than the baseline, it still struggled to fully capture the complexity of the data, particularly on larger datasets. Despite improvements in prediction accuracy, the model's ability to generalize remained limited.

Advanced Model: XGBRegressor with Log Transformation

We switched to XGBoost, a powerful gradient boosting model that excels at handling large datasets and complex relationships. To enhance performance, we applied a log transformation to the target variable, ensuring that predictions remained non-negative. Regularization terms (alpha and lambda) were introduced to penalize large coefficients and help control overfitting. This approach led to significant improvements, achieving a Mean Squared Error (MSE) of 0.09 and an R² score of 0.63, the best results across all models tested.

Handling Outliers

To address underfitting and improve model performance, we implemented an outlier detection and handling mechanism using the Interquartile Range (IQR). Outliers in features such as time intervals (arrival vs. closing time) were capped, reducing their impact on the model's learning process. This helped smooth the prediction errors, which was evident from the residual plots where the residuals became more tightly distributed around zero.

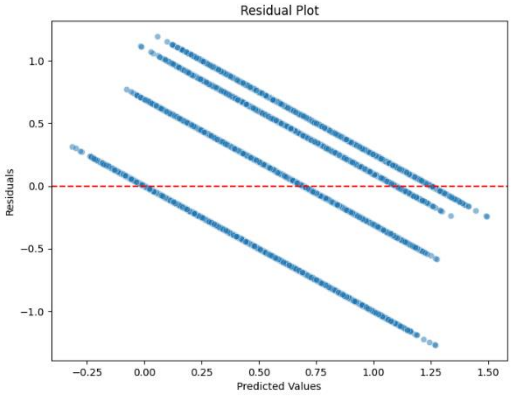
Results and Visualizations

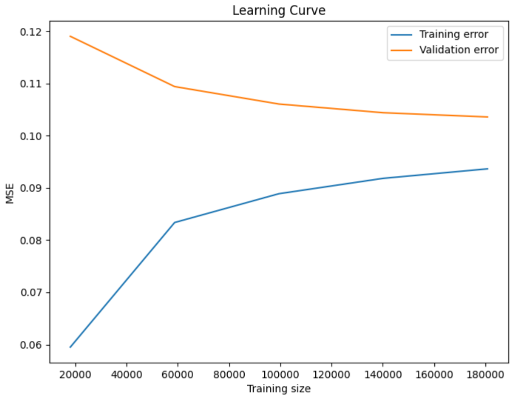
The final model showed consistent performance across different training sizes, as illustrated by the learning curves. The training and validation errors converged, indicating that the model was neither underfitting nor overfitting.

- **Training MSE:** 0.09
- **Training R²:** 0.63
- **Cross-validation MSE:** Mean of ~0.10 across folds

Visualizations:

1. **Residual Plot:** The residuals in the final XGBRegressor model show a more uniform distribution around zero, confirming that the model captured the underlying trends without being biased by extreme values or outliers.
2. **Learning Curve:** The training and validation curves converged smoothly, showing that the model improved as the dataset size increased, and the validation error gradually decreased.
3.





Task 2: Predicting Trip Duration

In this task, our objective was to predict the duration of a single bus trip, from its first station to its last station. Each trip_unique_id represents a complete bus trip. Based on the information from the stops along the trip, we aimed to predict the arrival time at the last stop, with predictions expressed in minutes.

Initial Approach: Linear Regression

We initially applied a simple **Linear Regression** model to predict trip duration. Despite the model's simplicity, it encountered significant challenges due to the complexity of the data, as indicated by high error metrics. This model produced a **Mean Squared Error (MSE) of 114.58** and an **R² score of 0.70** on the training set, along with erratic cross-validation MSE values, which demonstrated limited generalization. Linear Regression was unable to capture the non-linear relationships in the data, leading to underfitting.

Improving the Model: XGBRegressor

To better handle the data complexity, we transitioned to **XGBRegressor**, a gradient boosting model that can manage non-linear relationships more effectively. This model provided an improvement, achieving an **MSE of 77.42** and an **R² score of 0.80** on the training data. However, the cross-validation MSE values remained high, ranging from 118.53 to 151.07, indicating the model was still struggling to generalize well to unseen data.

Advanced Model: RandomForestRegressor

Recognizing the need for a model that could better capture complex patterns without overfitting, we implemented a **RandomForestRegressor**. Initially, this model reduced the **training MSE to 26.81** and raised the **R² score to 0.93**. However, the cross-validation MSE values, ranging from 125.98 to 155.42, suggested that the model was still overfitting to some degree.

Enhanced Feature Engineering and Final Model

To address the overfitting and further improve the model's generalization, we enhanced the preprocessing pipeline with additional feature engineering steps:

- **Log Transformation** of skewed features (total_passengers_up and num_stations) to reduce the impact of extreme values.
- **Interaction Term** between total_passengers_up and num_stations, capturing the cumulative effect of these features.
- **Time-Based Features**, including hour and day_of_week, to account for potential variations in trip duration related to the time of day and day of the week.

These feature engineering improvements resulted in a significant enhancement in performance.

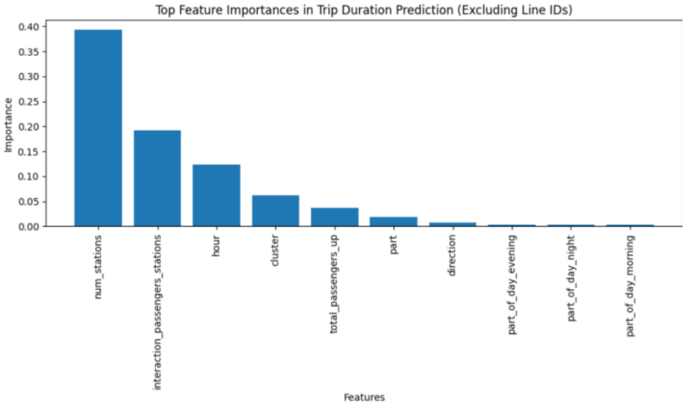
Results and Visualizations

The **RandomForestRegressor** final model demonstrated strong generalization, with the cross-validation MSE values closely aligned with the training MSE, indicating minimal overfitting. The addition of engineered features allowed the model to capture complex patterns more effectively, resulting in robust performance across different data splits.

- **Training MSE**: 13.36
- **Training R²**: 0.96
- **Cross-Validation MSE**: Average ~97 across folds

Visualizations:

1. **Feature Importance Plot**: The top features influencing trip duration are **num_stations**, **interaction_passengers_stations**, and **hour**, highlighting the impact of trip structure and timing.


Top Feature Importances in Trip Duration Prediction (Excluding Line IDs)
Importance
num_stations
interaction_passengers_stations
hour
cluster
total_passengers_up
part
direction
part_of_day_evening
part_of_day_night
part_of_day_morning
Features