NYC Taxi Fare Prediction: Machine Learning and Data Analysis

Iachim Cristian
Serbicean Alexandru
Universidade da Madeira
2238224@student.uma.pt 2236924@student.uma.pt

March 22, 2025

Abstract

To be completed

Contents

1 Introduction **2**
1.1 Objectives 2

2 Data Analysis and Preprocessing **2**
2.1 Dataset Overview 2
2.2 Data Cleaning 2
2.3 Feature Engineering 2
2.4 Time-Based Features 3
2.5 Trip-Related Features 3
2.6 Indicators for Trip Patterns 3
2.7 Fare-Based Feature 4
2.8 Impact of Feature Engineering 4

1 Introduction

Predicting taxi fares is crucial for optimizing pricing models, improving ride-sharing services, and enhancing urban mobility. This project leverages machine learning algorithms to classify NYC taxi fares into high and low categories. The dataset, sourced from Kaggle, contains ride details such as trip distance, pickup/dropoff time, passenger count, and fare amount.

1.1 Objectives

- Develop machine learning models to predict fare amounts.
- Compare the performance of different algorithms.
- Investigate potential overfitting in ensemble models.
- Deploy a trained model as an API for real-world applications.

2 Data Analysis and Preprocessing

2.1 Dataset Overview

The dataset consists of NYC taxi trips from 2019, obtained from Kaggle. It includes:

- **Numerical Features:** Trip distance, fare amount, trip duration, etc.
- **Categorical Features:** Payment type, vendor ID, store-and-forward flag.
- **Datetime Features:** Pickup and dropoff timestamps.

2.2 Data Cleaning

The preprocessing steps include:

1. Handling missing values.
2. Encoding categorical variables (payment type, vendor ID).
3. Standardizing numerical features using `StandardScaler`.
4. Removing outliers using the **IQR method**.

2.3 Feature Engineering

Feature engineering plays a crucial role in enhancing the predictive power of machine learning models by deriving new, meaningful features from raw data. In this project, we extracted several time-based and trip-related features to improve model performance. These features provide additional insights into trip patterns, customer behavior, and fare variations.

2.4 Time-Based Features

Understanding the temporal dynamics of taxi trips is essential, as factors such as rush hours, weekends, and nighttime conditions can significantly influence fare prices and trip durations.

- **Pickup Hour** (`pickup_hour`) Extracted from the `tpep_pickup_datetime` column, this feature represents the hour of the day when the trip started. It helps capture daily patterns such as peak and off-peak hours.

- **Pickup Day** (`pickup_day`) Extracted from the `tpep_pickup_datetime` column to indicate the day of the month when the trip took place. This is useful for identifying fare fluctuations over the course of a month.

- **Pickup Weekday** (`pickup_weekday`) Represents the day of the week (0 = Monday, 6 = Sunday). This feature helps determine the impact of weekdays versus weekends on taxi fares.

- **Pickup Month** (`pickup_month`) Extracted from the `tpep_pickup_datetime` column to indicate the month of the trip. This is useful for analyzing seasonal effects on taxi fares.

2.5 Trip-Related Features

To capture the dynamics of taxi trips, several features were engineered to measure trip duration, speed, and fare characteristics.

- **Trip Duration** (`trip_duration`) Calculated as the time difference between `tpep_dropoff_datet` and `tpep_pickup_datetime`, converted into minutes.

  $$\text{trip\_duration} = \frac{\text{tpep\_dropoff\_datetime} - \text{tpep\_pickup\_datetime}}{60} \tag{1}$$

  This feature is essential for estimating fare amounts, as longer trips generally cost more.

- **Speed in Miles per Hour** (`speed_mph`) Estimated as the total trip distance divided by the trip duration (converted into hours).

  $$\text{speed\_mph} = \frac{\text{trip\_distance}}{\frac{\text{trip\_duration}}{60} + 1e^{-6}} \tag{2}$$

  This feature is useful for detecting outliers (e.g., unrealistically high speeds might indicate data anomalies).

2.6 Indicators for Trip Patterns

Binary indicators were created to represent whether a trip occurred during specific conditions such as weekends, rush hours, or nighttime.

- **Weekend Indicator (`is_weekend`)** Set to **1** if the trip occurred on **Saturday (5) or Sunday (6)**, otherwise **0**. This helps analyze fare trends during weekends versus weekdays.
- **Rush Hour Indicator (`is_rush_hour`)** Set to **1** if the trip occurred during typical rush hours (**7-9 AM** or **5-7 PM**), otherwise **0**. This feature helps capture fare surges during high-traffic periods.
- **Nighttime Indicator (`is_night`)** Set to **1** if the trip took place between **midnight (00:00) and 5 AM**, otherwise **0**. Nighttime trips often have different fare structures due to lower traffic but increased demand for taxis.

2.7 Fare-Based Feature

To formulate the problem as a classification task, we introduced a binary target variable.

- **High-Fare Classification (`high_fare`)** Set to **1** if the `fare_amount` is greater than the median fare, otherwise **0**. This feature helps convert the regression problem (predicting fare amounts) into a **binary classification task** (high-fare vs. low-fare).

2.8 Impact of Feature Engineering

The introduction of these features significantly improves model performance by providing:

- **More granularity in temporal patterns** (e.g., rush hour vs. non-rush hour trips).
- **Better trip duration and speed insights** (useful for estimating realistic fare amounts).
- **Fare classification for improved decision-making** (e.g., predicting whether a trip is expensive based on known features).

These engineered features contribute to a higher predictive accuracy and help refine the machine learning models used in this project.