Concept Drift and Predicting Duration of Divvy Bike Trips

Jackie Glasheen, Kathryn Link-Oberstar, Jennifer Yeaton
University of Chicago

Abstract

We explored the concept of distribution drift in Divvy bike ridership data, spanning from 2014 through 2019. Utilizing a random sample of one million trips (approximately 5% of the dataset), we examine trip duration trends, revealing a notable shift in distribution patterns between 2014-2017 and 2018-2019, especially during summer months. Our analysis incorporates various machine learning models including K-Nearest Neighbors (KNN), Random Forest, and Multi-Layer Perceptrons (MLP) to predict trip durations. The models are evaluated using Mean Squared Error (MSE), Mean Absolute Error (MAE), and Root Mean Squared Error (RMSE), with adjustments for recent trends to address the observed distribution drift. The MLP model demonstrates superior performance, suggesting its effectiveness in handling high-dimensional data and adapting to non-linear patterns in the presence of distribution drift. Our findings highlight the importance of accounting for temporal changes in data distributions when developing predictive models.

1 Introduction And Problem Definition

Distribution drift can arise when the underlying distribution of data used to train a model changes over time. There are primarily two types of drift: *Covariate (feature) drift*, where the distribution of the input data (features) changes, and *Concept Drift*, where the relationship between the input data and the output variable (the concept) alters. Both types can significantly affect the performance of a machine learning model. Our focus in this prediction task will primarily be on Concept Drift.

We examine distribution drift within the framework of a trip duration forecasting model for Divvy bikes in Chicago. Demand and usage forecasting are crucial for managing efficient bike-sharing systems and other transportation infrastructures. Our preliminary analysis of the trip duration dataset revealed signs of distribution drift, potentially reflecting changes in user behavior over time. To address this, we engaged in feature engineering, model selection, and parameter tuning, aiming to develop a robust model capable of handling distribution drift and effectively forecasting trip duration on new data.

2 Literature Review

Research addressing distribution drift is substantial. In Cummings et al. (2023), authors propose a statistical test designed to identify violations of the assumption that data are independent and identically distributed (IID). The violation of the IID assumption suggests adjacent examples in a data set exhibit more similar values. The proposed method involves constructing a k-nearest neighbor (KNN) model based on feature values and then evaluating the indices of the data points. They then determine if there is a statistically significant difference between the distributions of index distances between k-nearest neighbors compared to the index distances between arbitrary data point pairs.

37th Conference on Neural Information Processing Systems (NeurIPS 2023).

Agrahari et al. (2022) studied concept drift in the context of online learning, where data streams are continuously updated. Hidden data distribution drift will reduce the accuracy of the learning algorithms, often going undetected. Unlike traditional models that evaluate distribution drift on stationary data, Agrahari addresses the fact that in many modern contexts, models use live time series streams. Agrahari introduces survey categorizations and use-cases of concept drift detectors for real time data that allows classifiers to automatically adapt to drift over time.

Xiang et al., in their 2023 publication, present an in-depth literature review on neural networks' effectiveness in handling distribution drift. They explore how different learning models address various types of concept drift. For instance, Discriminative Learning, which concentrates on class distinction, is adept at handling abrupt drift but less effective with incremental and gradual Drifts.

Beyond distribution drift, other relevant literature includes work relating to transportation management algorithms. Karami et al., for example, evaluates ARIMA, Kalman filtering, random walk, KNN, and deep learning as they pertain to controlling and forecasting car traffic.

3 Analysis: Data Exploration

The full data set is comprised over 21 million observations, encompassing all trips on Divvy bikes from mid-2013 to 2019. For a more manageable analysis, we reduced the data set to a random sample of 1 million trips from this period, approximately 5% of the total data.

After sampling the data, processing the data, and removing extreme records, we investigated the trip duration, both by origin station and total trips over time. While the trip duration trends over time are varied at the station level, in aggregate, we see likely evidence of distribution drift over the years observed. In particular, while the trip duration averages follow a similar pattern from 2014 to 2017, the average duration changes significantly in 2018 and 2019.

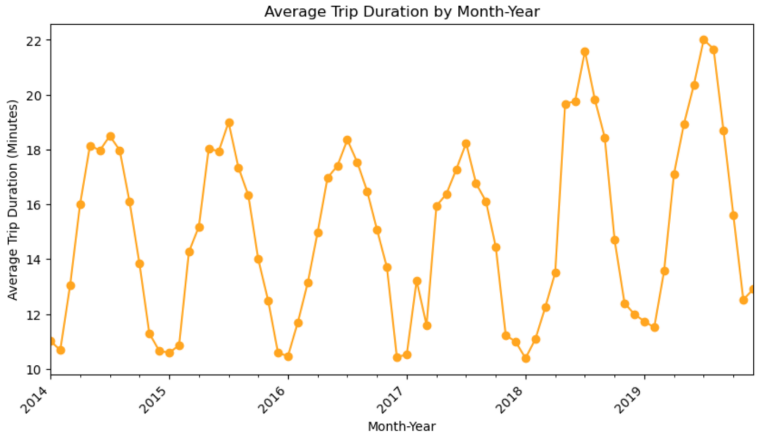

Figure 1: Average Trip Duration By Month - Year

This trend is further illuminated by visualizing the averages plotted with each year as a separate series. We find that the trip duration trend abruptly changes in the summer months (April - September) of 2018 and 2019 relative to the prior years.

The significance of this distribution drift is that any model forecasting trip duration is likely to exhibit error if trained on data from the years 2014-2017, because the distribution has since shifted and the earlier patterns may lead to underestimating future trip length.

4 Analysis: Model Building

Data Cleaning
Feature Engineering To prepare our data for modeling, we executed basic data cleaning and feature engineering:

- Created distinct columns for month, day of the week, hour, and year.

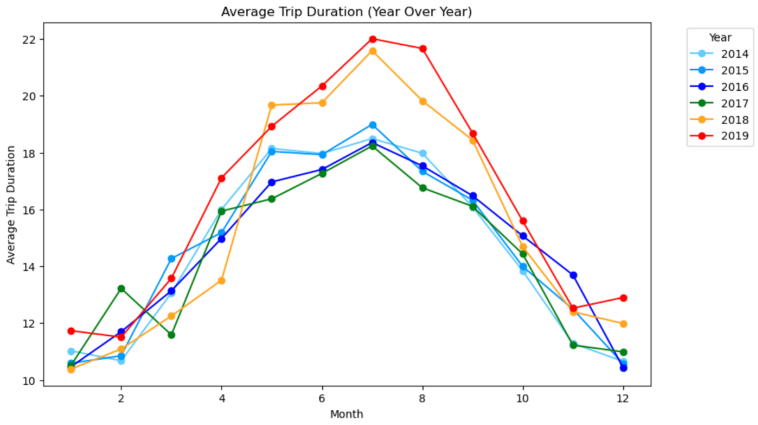

Figure 2: Average Trip Duration (Year Over Year)

- Recoded user type (subscriber or non-subscriber) and gender as binary variables.
- Converted trip duration from seconds to minutes for better interpretability.
- Excluded outliers, defined as trips exceeding 10 hours.

Model-Specific Adjustments Some models required tailored adjustments:

- KNN: Model cannot process missing values, so we dropped feature for birth year and a small number of rows with missing data from latitude and longitude.
- Multi-Layer Perceptron: We rescaled features to unit variance.

Model Training

The goal was to use the models to predict trip duration in minutes. We trained each model using the following inputs: user type (subscriber or not), user gender, user birth year (except for KNN), latitude and longitude of the starting station, and the year, day of the week, hour, and month of the ride.

Model Parameter Tuning

We explored various models across three different types: KNN, Random Forest, and Multi-Layer Perceptron (Neural Network), all of which are recognized in literature for their potential in handling distribution drift. We assessed our models using the following metrics:

- *Mean Squared Error (MSE)*: The average of squared differences between predicted and actual values. It gives more weight to larger errors due to squaring each term. (Note: MSE can be disproportionately high in cases with many outliers).
- *Mean Absolute Error (MAE)*: The average of absolute differences between predicted and actual values. Unlike MSE, it treats all errors uniformly.
- *Root Mean Squared Error (RMSE)*: The square root of the MSE, aligning the error metric with the target variable's units. It offers greater interpretability and lessens some of the penalties for large errors inherent in MSE.

In fine-tuning the parameters of these models, we employed several strategies to enhance performance and counter concept drift:

- *Weighted Loss Function*: Substituting MSE or Huber loss with a weighted loss function places greater emphasis on recent data, thus making the model more adaptive to new trends and patterns, a crucial aspect when dealing with concept drift.
- *Sample Weights*: Oversampling recent data augments the representation of newer trends in training, directing the model to concentrate more on current patterns.
- *Decay Factor*: Applying a decay factor to older data diminishes their influence, making the model less dependent on outdated trends.

- *Further feature selection*: Selecting a subset of the features we decided to include in our models and testing how different subsets of features performed.

Additionally, we experimented with the *Huber Loss function* in lieu of MSE, as it is better equipped to handle outliers and abrupt changes in data distributions, common occurrences in concept drift.

Table 1: Summary of Model Performance

Model	MSE	MAE	RMSE
KNN: Train/test on all years (Baseline)	31,461.20	12.78	177.37
KNN: Train on 2014-2017; Test on 2019	104,101.07	15.41	322.64
KNN: Train on 2015-2017; Test on 2019	104,115.76	15.42 3	322.67
KNN: Train on 2014-2018; Test on 2019	104,675.61	17.09	323.54
KNN: Train on 2015-2018; Test on 2019	104,858.53	17.34	323.82
KNN: Train on 2018; Test on 2019	105,619.38	19.18	324.99
Random Forest	560.90	11.43	23.68
Random Forest - Weighted Sample	563.05	11.45	23.73
MLP (Baseline - MSE)	549.12	12.14	23.4
MLP - Huber Loss	586.28	11.33	24.2
MLP - Weighted Loss Function	554.78	11.78	23.6
MLP - Sample Weights (Oversample newer observations)	597.10	11.34	24.44
MLP - Decay Factor (decay factor to reduce influence of older data)	585.11	11.33	24.19

5 Analysis: Discussion

The Multi-Layer Perceptron model and Random Forest performed comparably, both surpassing KNN in minimizing error:

KNN: The KNN model faced challenges across various data splits. Its heavy reliance on distance metrics to identify nearest neighbors becomes less effective in high-dimensional spaces, a phenomenon often referred to as the "curse of dimensionality." In attempts to better accommodate concept drift, we used smaller segments of more recent data. However, this approach might have led to overfitting, reducing the model's generalizability in our test data.

Random Forest: As an ensemble learning method, Random Forest combines multiple decision tree classifiers to make predictions. This model significantly outperformed KNN, likely owing to its proficiency in managing high-dimensional data. Unlike KNN, Random Forest is generally more robust to outliers and can handle complex linear relationships effectively.

Multi-Layer Perceptron (Neural Network): The Multi-Layer Perceptron also outperformed KNN and showed similar efficacy to Random Forest. Our strategy included using MSE as a baseline and experimenting with Huber Loss to reduce the impact of outliers. The MLP's robustness in processing high-dimensional data and its proficiency in managing non-linear patterns likely contributed to its effective performance. Its inherent adaptability and flexibility, courtesy of its hidden layers, might explain its enhanced capacity to manage data with distribution drift compared to KNN.

6 Conclusions

In our study of trip duration prediction in a dataset that exhibited distribution drift, the Multi-Layer Perceptron and Random Forest models showed similar effectiveness and outperformed KNN. The better performance of the Multi-Layer Perceptron and Random Forest models can be attributed to their superior capabilities in handling high-dimensional data and concept drift. Modifying the models to prioritize more recent data yielded mixed results. For the Random Forest model, these adjustments slightly deteriorated performance across all metrics. For the Multi-Layer Perceptron, applying a weighted loss function marginally improved MSE and MAE but was slightly less effective on RMSE compared to the baseline. Future research could explore the integration of real-time data feeds and continuous learning models to further enhance the adaptability of predictive systems in rapidly changing environments like city bike-share programs.

References

[1] Agrahari, S. Singh, A. K (2022). Concept drift detection in data stream mining: A literature review. *Journal of King Saud University-Computer and Information Sciences*, 34 (10):9523–9540.

[2] Cummings, J. Snorrason, E. & Muller, J (2023) Detecting Dataset Drift and Non-IID Sampling via k-Nearest Neighbors. arXiv:2305.15696v1.

[3] Karami, Z. & Kashef, R Smart transportation planning: data, models, and algorithms (2020) Transport. Eng., 2 , Article 100013, 10.1016/j.treng.2020.100013.

[4] Xiang, Q. Zi, L. Cong, X. & Wang, Y. (2023) Concept drift adaptation methods under the deep learning framework: A literature review. Applied Sciences, 13(11), 6515.