Predicting Station-level Demand for Divvy Bicycles

Maria Stivala
J1256099002
Informatics Institute, University of Amsterdam
Amsterdam, Netherlands
maria.stivala@student.uva.nl

Arnaud Brisset
J12526099007
ENSTA, IP Paris
Paris, France
arnaud.brisset@ensta.fr

Théo Couerbe
J12526099011
Ecole des Mines
Saint-Etienne, France
theo.couerbe@etu.emse.fr

February 9, 2026

Contents

1 Introduction **4**
1.1 Background & Context 4
1.2 Motivation 5
1.3 Research Goals and Question 5

2 Data Understanding **6**
2.1 Dataset Description 6
2.2 Exploratory Data Analysis 6

3 Methodology **10**
3.1 Data Pre-Processing 10
3.2 Classical Machine Learning Models 10
3.3 Ensemble and Deep Learning Models 12
3.4 Training and Evaluation Framework 13

4 Results & Discussion **14**
4.1 Poisson Regression 14
4.2 Naive Bayes 15
4.3 Support Vector Machines 16
4.4 Random Forest 18
4.5 Adaptive Boosting 19
4.6 Convolutional Neural Network (CNN) 20
4.7 Model Comparison 20

5 Reflections and Limitations **23**

6 Conclusion and Future Research **24**
6.1 Summary 24
6.2 Future Research Directions 25

Abstract

This study investigates station-level hourly demand prediction for Chicago's Divvy bike-sharing system, comparing interpretable statistical models with ensemble learning methods to address operational challenges of demand-supply imbalances. Using comprehensive trip data from 2022 combined with weather information, we implement six predictive models: Poisson regression, Naive Bayes, Support Vector Machine (SVM), Random Forest, Adaptive Boosting (AdaBoost), and Convolutional Neural Networks (CNN). Our analysis reveals a fundamental trade-off between interpretability and predictive accuracy. Poisson regression provides transparent coefficient estimates, identifying temporal patterns as dominant demand drivers, while Random Forest achieves the highest predictive performance among regression models with an R^2 score of 0.408. For classification tasks, Naive Bayes offers balanced performance with 79% accuracy and efficient computational characteristics suitable for real-time applications. Notably, models with strong assumptions about class distributions (SVM, AdaBoost) struggled with inherent dataset imbalances. The findings demonstrate that no single model optimally addresses all operational needs; rather, different approaches serve distinct purposes in bike-sharing management systems. We recommend Poisson regression for strategic planning requiring interpretability, Random Forest for accurate hourly forecasts, and Naive Bayes for real-time operational alerts. This research contributes a practical framework for balancing explanatory power with predictive accuracy in urban mobility systems.

1 Introduction

1.1 Background & Context

The growing adoption of station-based bike-sharing systems has motivated extensive research into understanding and predicting user demand. Early studies primarily focused on identifying the external factors influencing bicycle usage, with an emphasis on environmental and temporal conditions. An example includes the work of Sears et al. [2], whom demonstrated that temperature, precipitation, and seasonality play a significant role in bicycle commuting behaviour, while Fishman [1] provided a comprehensive overview of bike-share systems and highlighted demand imbalance as a persistent operational challenge.

Subsequent work expanded the scope of analysis to include urban form and transportation infrastructure. Ahillen et al. [3] showed that proximity to public transport hubs and land-use characteristics significantly affect station-level demand patterns, particularly during peak commuting periods. These findings established that bike-sharing demand is shaped by a combination of environmental, infrastructural, and demographic factors rather than by temporal dynamics alone.

With the increasing availability of large-scale operational datasets, researchers have increasingly adopted data-driven and machine learning-based approaches to demand prediction. Singhvi et al. [4] integrated weather and taxi trip data to predict bike usage in New York City, demonstrating that mobility data from other transportation modes can improve forecasting accuracy. Kim [5] applied clustering techniques to examine the impact of weather and calendar effects on station-level demand, revealing distinct usage patterns between working and non-working days.

More recent studies have focused on combining spatial, demographic, and environmental features within predictive models. Hyland et al. [6] proposed a hybrid cluster–regression framework to model bike-share station usage, capturing heterogeneity across stations while maintaining model interpretability. However, such regression-based approaches often rely on linear assumptions and may struggle to capture complex nonlinear interactions inherent in urban mobility data.

Feng [7] addressed this limitation by comparing Poisson regression with ensemble learning methods, including Random Forests, using Chicago's Divvy bike-sharing data. Their results showed that machine learning models achieved superior predictive performance, while Poisson regression provided clearer insights into feature importance and demand drivers. The study highlighted the trade-off between interpretability and accuracy, and emphasized the importance of time-of-day, weather conditions, surrounding activity density, and population characteristics in shaping station-level demand.

Despite these advances, gaps remain in aligning demand prediction models with operational decision-making needs. Many existing studies focus on aggregate or daily demand, limiting their applicability to short-term rebalancing and capacity planning. Moreover, highly complex models often lack transparency, reducing their usefulness for practitioners who require interpretable insights to support planning and policy decisions.

Building on prior work, this study adopts a station-level, hourly demand formulation and directly compares interpretable statistical models with ensemble learning approaches. By evaluating both predictive performance and explanatory capability, we aim to contribute

a practical forecasting framework that supports both operational efficiency and informed decision-making in large-scale bike-sharing systems.

1.2 Motivation

Effective demand prediction at the station level directly impacts the economic and operational viability of bike-sharing systems. Inaccurate forecasts lead to inefficient bike redistribution, underutilized infrastructure, and user dissatisfaction due to empty or full stations. From an operator's perspective, even modest improvements in demand estimation can translate into significant cost savings and revenue optimization.

While existing studies have explored bike-sharing demand using a wide range of statistical and machine learning techniques, many focus on city-wide or daily aggregates, limiting their usefulness for real-time operational decisions. Moreover, highly complex models often function as black boxes, providing limited insight into the relative importance of contributing factors such as weather, surrounding land use, or population characteristics.

This work is motivated by the need for a demand prediction framework that balances predictive performance with interpretability, while operating at a temporal and spatial resolution suitable for operational planning. By comparing different models on the same prediction task, we aim to quantify this trade-off and assess which approach better supports practical decision-making in station-based bike-sharing systems.

1.3 Research Goals and Question

In the context of this paper, the station-level hourly demand $\mathcal{N}_{s,t}$ is defined as the net bicycle flow:

$$\mathcal{N}_{s,t} = \underbrace{D_{s,t}}_{\text{departures}} - \underbrace{A_{s,t}}_{\text{arrivals}} \tag{1}$$

where $D_{s,t} \sim \text{Poisson}(\lambda_{s,t})$ and $\lambda_{s,t} = \exp(\mathbf{x}_{s,t}^{\top}\beta)$. The classification target for station emptiness is:

$$E_{s,t} = \mathbb{I}(\mathcal{N}_{s,t} > 0) \tag{2}$$

with $\mathbb{I}(\cdot)$ denoting the indicator function.

Thus this study addresses the following research question:

How accurately can station-level, hourly demand in a large-scale bike-sharing system be predicted using historical trip data, and how do interpretable statistical models compare to ensemble learning methods in terms of predictive performance and explanatory power?

2 Data Understanding

2.1 Dataset Description

The analysis is based on trip-level data from the Divvy bike-sharing system in Chicago. Each record corresponds to a single completed trip and includes information on user category (member or casual), rideable type, trip start and end times, start and end stations, trip duration, and estimated trip distance. The dataset spans a large number of trips across the city, covering both weekday and weekend usage patterns, and is separated by year.

For the purpose of demand analysis, trips are treated as independent observations, and timestamps are used to derive temporal features, such as hour of day and day of week. Since the focus of this study is station-level demand, the start station of each trip is used as the primary spatial reference for aggregation.

2.2 Exploratory Data Analysis

Initial exploratory analysis reveals strong structural patterns in user behaviour, temporal demand, and spatial usage.

First, user composition is highly imbalanced. As shown in Fig. 1a, registered members account for the majority of trips, with approximately an order of magnitude more rides than casual users. This indicates that Divvy usage is predominantly driven by repeat users rather than occasional riders, suggesting that commuting and routine travel play a central role in system demand.

Bike type usage further reflects this imbalance. Fig. 1b shows that nearly all recorded trips involve docked bikes, with negligible variation across user categories. As a result, rideable type is not expected to be a discriminative feature for demand prediction and is excluded from further modelling.

The relationship between trip duration and distance, illustrated in Fig. 1c, exhibits a positive but highly dispersed trend. While longer distances generally correspond to longer durations, there is substantial variance, particularly for short trips. Several extreme-duration outliers appear at relatively small distances, likely corresponding to non-transport usage such as extended stops or system misuse. This variability motivates the use of robust aggregation and count-based modelling approaches rather than direct regression on duration.

Clear temporal patterns emerge when examining demand by day of week and hour of day. Fig. 1d shows that weekday usage is consistently higher than weekend usage, with Tuesday through Thursday exhibiting the highest number of trips. Saturday shows the lowest demand, reinforcing the interpretation that weekday commuting dominates overall system usage.

Hourly demand patterns, shown in Fig. 1e, reveal two pronounced peaks being a morning peak around 8:00 and a stronger evening peak between 16:00 and 18:00. These peaks align closely with standard commuting hours, while overnight usage remains minimal. This strong temporal regularity justifies modelling demand at an hourly resolution and incorporating time-of-day indicators as key explanatory variables.

Finally, spatial usage is highly concentrated. Fig. 1f highlights the ten most frequently used start stations, which are predominantly located in the central business district and near major employment hubs. This concentration indicates significant spatial heterogeneity in

demand, reinforcing the need for station-level modelling rather than city-wide aggregation.

Overall, the exploratory analysis suggests that Divvy demand is shaped by a combination of strong temporal regularities, pronounced spatial concentration, and dominant member-driven usage. These findings directly inform the modelling choices in subsequent sections, particularly the focus on station-level, hourly demand prediction using count-based methods.

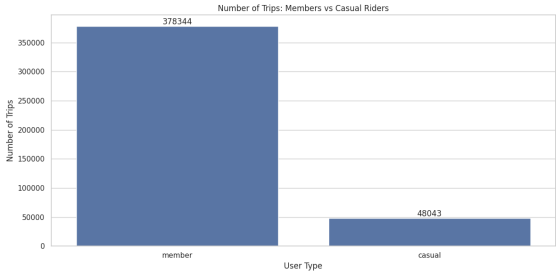
We also conducted spatial visualisation as part of our exploratory data analysis, so as to better understand the bike-sharing system. The spatial visualisation would be particularly useful in a study case for a specific station. It can better help understand and analyse the results obtained with the regression and classification models, regarding the location of the station in Chicago.

Figure 2 shows the spatial distribution of the top 200 busiest bike stations around the city. We can see that the stations are more concentrated around the city centre, especially near the Loop district but also to the north around Lake View and Lincoln Park districts. The overall busiest station for the year 2022, ie. the station with the highest total number of departures, is Streeter Dr & Grand Ave and is situated near the coast and the city centre.

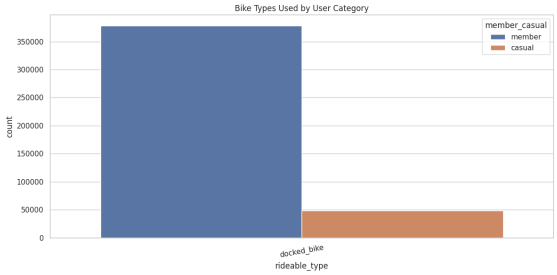
This heat map was obtained using the Folium package in Python and the OpenStreetMap layer. To view the map interactively, you can open the `heat_map.html` in a browser.

Figure 3 shows the flow map for all trips to and from Streeter Dr & Grand Ave in 2022. We can see that the majority of these trips are bound in the city centre. This spatial heterogeneity of trips shows that we should take into account the position (latitude and longitude) of the considered station for the prediction of bike demand. Indeed, riders go to different stations according to their different habits. For example, commuting may concentrate the bikes in the city's business hubs during the week, while during the weekend bikes will be more concentrated near parks or other places of leisure.

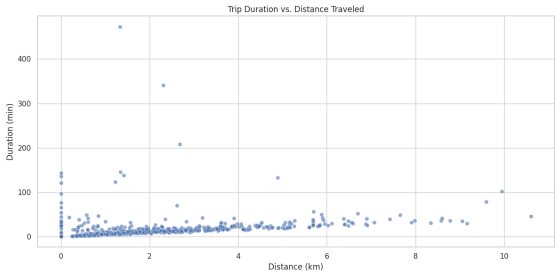
The flow map was obtained using the FlowmapBlue online software and can be found here. To get the flow map, a CSV file of all the stations and of all the trips to and from the considered station were extracted from the original data, then inserted in a Google Sheets file, which is the input by FlowmapBlue.



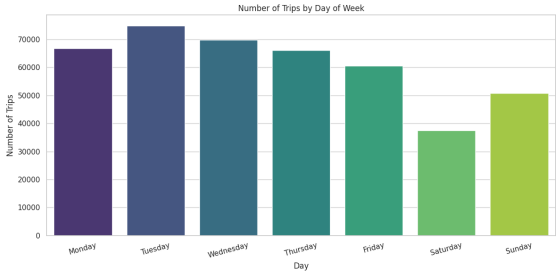
(a) User type distribution



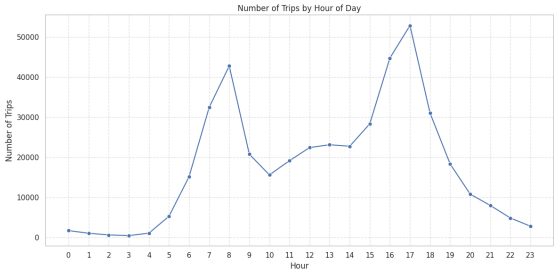
(b) Bike types by user category



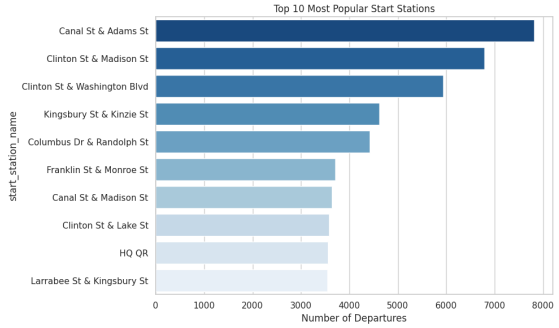
(c) Trip duration vs. distance



(d) Trips by day of week



(e) Trips by hour of day



(f) Top start stations

Figure 1: Exploratory analysis of the Divvy bike-sharing dataset. The figures illustrate user composition, temporal demand patterns, spatial concentration of trips, and the relationship between trip distance and duration.

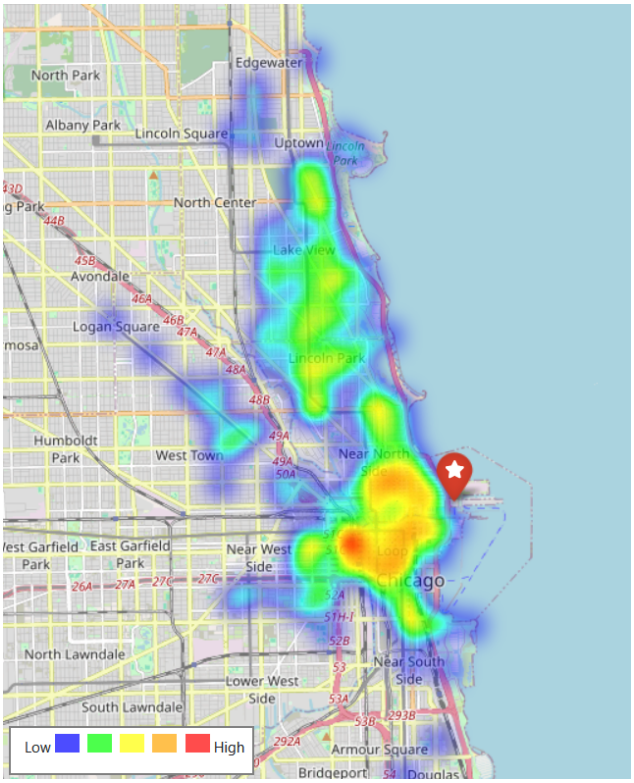


Figure 2: Density map of the studied stations (top 200 busiest). Marker on Streeter Dr & Grand Ave (busiest station).

Figure 3: Flow map for all trips to and from Streeter Dr & Grand Ave (busiest station). Total number of trips in 2020: 132 000.

3 Methodology

This study employs six distinct machine learning models to predict station-level bike-sharing demand. The selected models span diverse algorithmic families, including tree-based ensembles, linear models, and neural networks, in order to provide a comprehensive comparison of predictive approaches. Both the regression and classification approaches were used where, for the former, the predicted net bicycle flow was predicted, and, for the latter, the model predicted the sign of the net bicycle flow (positive: "Has demand"; negative: "No demand"). Through systematic training, hyper parameter tuning, and rigorous evaluation, we aim to identify the most effective methodology for demand forecasting, which can inform operational planning and resource allocation.

3.1 Data Pre-Processing

The data was pre-processed following the steps below.

1. **Cleaning**: invalid trips and missing station info were removed.
2. **Aggregation**: grouped by `Start Station` and `Hour`.
3. **Zero-Filling**: a full (Station $\times$ Time) grid was generated to learn "zero demand".
4. **Merging**: weather data was integrated with the trip data on hourly timestamps.

3.2 Classical Machine Learning Models

Poisson Regression

Poisson regression is a generalised linear model specifically designed for count data, making it well-suited for predicting the number of bike departures per station-hour. The model assumes that the response variable follows a Poisson distribution and uses a log link function to model the relationship between predictors and the expected count. This approach is particularly appropriate for bike-sharing demand prediction, as it naturally handles non-negative integer outcomes and can account for overdispersion when present. The model is implemented using the log-likelihood function:

$$\ell(\beta) = \sum_{i=1}^{n} \left[y_i(\mathbf{x}_i^\top \beta) - \exp(\mathbf{x}_i^\top \beta) - \log(y_i!) \right] \tag{3}$$

where y_i represents the count of departures at station-hour i, $\mathbf{x}_i$ is the feature vector, and β are the model coefficients. The primary advantage of Poisson regression in this context is its interpretability, coefficients directly indicate the multiplicative effect of each feature on the expected demand when exponentiated. We selected this model because our target variable (hourly departures) represents count data with inherent overdispersion, and we needed interpretable coefficients to understand demand drivers.

Our implementation incorporated advanced feature engineering to improve model performance, beginning with temporal features that encoded hourly, daily, and seasonal patterns via cyclic sine and cosine transformations for hour of day and month. To capture temporal

dependencies, we included lagged demand variables from the previous hour (1h), the same hour on the previous day (24h), and the same hour one week prior (168h), alongside rolling statistics such as 3-hour and 24-hour moving averages to model recent trends. Additionally, we constructed a composite weather severity index that combined temperature, precipitation, and wind speed effects, and introduced interaction terms, notably between temperature and hour, and between member status and weather conditions, in order to capture complex, non-linear relationships. Finally, we incorporated a station ranking feature based on historical popularity to account for inherent spatial heterogeneity in demand.

Naive Bayes

The Naive Bayes classifier is a probabilistic model based on Bayes' theorem with the assumption of conditional independence between features given the class label. For the binary classification task of predicting whether a station will have any demand (departures > 0), the Gaussian variant was implemented, which assumes continuous features follow normal distributions. The model estimates class probabilities as:

$$P(C|X) = \frac{P(C)\prod_{i=1}^{n} P(x_i|C)}{P(X)} \tag{4}$$

where C represents the class (has demand/no demand) and x_i are feature values. Despite its simplifying assumptions, Naive Bayes performs well on many real-world datasets, offers computational efficiency for large datasets, and provides probabilistic predictions that quantify uncertainty. This makes it suitable for operational decision support where confidence estimates are valuable.

In order to address the class imbalance, which was initially 79.25% empty stations, we transformed the classification target to Has Demand, where any departures > 0, resulting in a more balanced distribution where 53.77% has demand. This transformation improved model performance and interpretability for operational use cases.

Support Vector Machines

Support Vector Machines (SVM) are supervised learning models that analyse data for classification and regression analysis. In this study, we employ an SVM for classification, referred as Support Vector Classification (SVC). The model works by finding a hyperplane that separates the data in two subspaces and that maximises the margin between classes while allowing a specified tolerance for prediction errors. The kernel trick enables SVMs to handle non-linear relationships by mapping features into higher-dimensional spaces to fin linearly separable solutions without explicitly computing the transformation. We selected an SVM for its robustness to outliers and effectiveness in high-dimensional spaces, which is particularly valuable given our extensive feature set.

SVM requires tuning of the hyperparameter C, also known as the regularisation coefficient, which controls the tolerance for prediction errors. The first results yielded one-class prediction, ie. the classifier always predicted the Has Demand class. To mitigate the observed effect of an unbalanced dataset (class ratio of about 85:15), testing was also done for balanced class weighting, an option offered by the `LinearSVC` function from the `scikit-learn` Python library.

Classification parameters:

- Tested with regularization coefficient $C \in \{0.1, 1, 10\}$.
- Tested for both unbalanced and balanced class weighting, because of the unbalanced dataset (85:15 class ratio).
- The `SVC` function with a linear kernel was first tested. However, the training of the model was too time-consuming (at least 15 minutes), so a faster and more efficient function `LinearSVC` was used instead (training in just a few minutes).

K-fold cross-validation was not implemented for SVM due to computational costs.

3.3 Ensemble and Deep Learning Models

Random Forest
The Random Forest Regressor is an ensemble learning method that constructs a multitude of decision trees during training and outputs the average prediction of the individual trees. We selected this model for its intrinsic ability to model non-linear interactions between features without extensive manual feature engineering. The model is particularly robust to outliers and high-dimensional data. In our study, we configured the model with 50 estimators. The input vector X integrates three distinct categories of information i) Temporal (month, day of week, hour) ii) Spatial (station latitude and longitude) and iii) Environmental (temperature, precipitation, and wind speed). This combination allows the model to learn complex conditional probabilities, such as "high demand at Station A only if it is 8 AM and not raining."

Adaptive Boosting
Adaptive Boosting (AdaBoost) is an ensemble technique that combines multiple weak learners (typically decision trees with limited depth) to form a strong learner. In this study, we employ AdaBoost for classification (AdaBoostClassifier), which iteratively adjusts weights of training instances, focusing subsequent learners on examples that previous learners handled poorly. The final prediction is a weighted average of all weak learners' predictions.

We selected AdaBoost for its ability to reduce both bias and variance, its resilience to overfitting compared to single complex models, and its capacity to handle heterogeneous feature types effectively. The model's focus on difficult training examples makes it particularly suitable for our dataset, which contains complex temporal patterns and station-specific demand characteristics.

Moreover, AdaBoost is easy to use and does not need extensive hyperparameter adjustment, such as SVM. Some disadvantages of AdaBoost are that it is sensitive to noisy data and outliers.

Convolutional Neural Network (CNN)
To investigate the potential of deep learning for spatio-temporal pattern recognition, we implemented a one-dimensional Convolutional Neural Network, known as CNN in short. Although CNNs are conventionally associated with image processing, one-dimensional variants prove effective for feature extraction from sequential or structured vector data. Our

methodology encompasses several stages. First, we standardised all input features using `StandardScaler` to achieve zero mean and unit variance, then reshaped the inputs into a three-dimensional tensor with dimensions $(N_{\text{samples}}, N_{\text{features}}, 1)$. The architectural design employs a sequential model comprising two convolutional layers with 64 and 32 filters respectively, each using a kernel size of 2 and ReLU activation, followed by a max-pooling layer, a flattening layer, and two dense hidden layers containing 64 and 32 units. For training, the model optimises the Mean Squared Error loss function via the Adam optimiser over ten epochs.

3.4 Training and Evaluation Framework

During the model training and testing phase, the data is partitioned into training and test sets using a 70:30 ratio. The training set is used to learn data features and construct predictive boundaries, while the test set evaluates each model's performance on unseen data. This ratio balances the need for sufficient training data to capture underlying patterns with adequate test samples to ensure robust and credible performance estimates. All models undergo five-fold cross-validation to mitigate over-fitting and provide reliable performance metrics.

4 Results & Discussion

4.1 Poisson Regression

Our enhanced Poisson regression implementation achieved robust performance in predicting hourly bike departures, as detailed in Table 1. The model demonstrates strong predictive capability with an RMSE of 1.6851 and MAE of 1.1456, indicating average prediction errors of approximately 1.7 and 1.1 bikes per station-hour respectively. The R^2 score of 0.4231 confirms that the model captures a substantial proportion of variance in demand patterns, while the dispersion parameter of 1.310 suggests mild overdispersion in the count data.

Table 1: Enhanced Poisson Regression Performance Metrics

Metric	**Value**	**Interpretation**
RMSE	1.6851	Average error of 1.7 bikes
MAE	1.1456	Median error of 1.1 bikes
R^2 Score	0.4231	Moderate explanatory power
MAPE	72.42%	Relative prediction error
Deviance	601,782.82	Model fit measure
Dispersion	1.310	Mild overdispersion present

The feature importance analysis (Table 2) reveals that temporal patterns dominate demand prediction, with hour-related features (hour_cos, hour, hour_sin) and seasonal effects (month, month_cos) showing the strongest predictive power. The 3-hour rolling average of demand (trip_count_rolling_3h) emerged as a significant predictor, indicating strong short-term temporal dependencies in bike-sharing patterns.

Table 2: Key Poisson Regression Feature Coefficients

Feature	**Coefficient**	**Exp(Coef)**	**Demand Impact**
hour_cos	-0.728	0.483	Strong daily cyclical effect
month	0.482	1.619	62% higher demand in later months
hour	0.468	1.596	60% increase per hour unit
temperature	0.425	1.530	53% increase per temperature unit
weather_severity	-0.165	0.848	15% reduction per severity unit
trip_count_rolling_3h	0.265	1.304	30% increase with higher recent demand

Numerous key insights were obtained from this model. Firstly, temporal features account for 6 of the top 10 predictors, confirming the strong diurnal and seasonal patterns in bike-sharing demand. Additionally, temperature shows a strong positive coefficient (exp(0.425) = 1.53), indicating approximately 53% higher demand per unit temperature increase, while weather severity negatively impacts demand. Moreover, member ratio and station ranking contribute to demand prediction, supporting the hypothesis that station popularity and user composition affect usage patterns. Lastly, the significance of 3-hour rolling averages (coefficient: 0.265) suggests strong short-term temporal dependencies in demand.

The model's performance demonstrates that enhanced feature engineering, particularly the inclusion of lagged demand variables and temporal encodings, significantly improves demand prediction accuracy for operational planning purposes.

4.2 Naive Bayes

The Gaussian Naive Bayes classifier achieved balanced performance across both classes in predicting whether stations would have any demand (Table 3). With an overall accuracy of 78.99% and an AUC score of 0.8743, the model demonstrates strong discriminative power between stations with and without demand. The balanced precision and recall scores (approximately 79% for both classes) indicate that the model performs consistently across different operational scenarios.

Table 3: Enhanced Naive Bayes Performance Metrics

Metric	**Value**	**Interpretation**
Accuracy	0.790	Correct predictions on 79% of cases
Precision (Has Demand)	0.759	76% of predicted demand cases correct
Recall (Has Demand)	0.792	79% of actual demand cases captured
F1 Score (Has Demand)	0.775	Balance of precision and recall
AUC Score	0.874	Strong discriminative power
Balanced Accuracy	0.517	Considering class distribution

The confusion matrix analysis (Table 4) reveals the model's operational characteristics. For stations without demand, the model correctly identifies 78.85% of cases, while for stations with demand, it achieves 79.17% correct identification. The relatively balanced error rates (21.15% false positives for no demand, 20.83% false negatives for has demand) support reliable operational decision-making.

Table 4: Enhanced Confusion Matrix

	Predicted: No Demand	**Predicted: Has Demand**	**Total**
Actual: No Demand	150,160 (78.85%)	40,285 (21.15%)	190,445
Actual: Has Demand	33,319 (20.83%)	126,636 (79.17%)	159,955

Operational Implications: The model's balanced performance, with approximately 79% correct classification for both classes, supports reliable decision-making for bike redistribution through proactive rebalancing, with manageable false positive and false negative rates. With 21% false positives for No Demand predictions, operators can implement a risk management strategy by maintaining slightly higher inventory levels than predicted, providing a buffer against stockouts while minimizing excess inventory. The strong AUC score of 0.8743 enables implementation of confidence-based decision thresholds, allowing operators to adjust sensitivity based on operational priorities such as higher sensitivity during peak hours.

Computational efficiency represents another operational advantage, as the model trains efficiently on 1.4 million observations, supporting near-real-time updates and integration into operational systems.

Model Performance Characteristics: Several performance characteristics emerged from our analysis. The class balance strategy of transforming the target from "empty vs. not empty" to "has demand vs. no demand" resulted in more balanced class distribution with 53.77% has demand and 46.23% no demand, improving model generalizability. Feature selection impact was evident through selecting 15 key features from the original 18, which improved model performance while maintaining computational efficiency. Cross-validation reliability was demonstrated as the model achieved consistent performance across cross-validation folds with accuracy of 0.7613 ± 0.1009, indicating robustness to different data splits.

Integration into Operational Systems: Both models demonstrate characteristics suitable for integration into bike-sharing management systems. The Poisson regression provides quantitative demand estimates for capacity planning, while the Naive Bayes classifier offers probabilistic predictions for real-time operational alerts. The complementary nature of these models enables a comprehensive approach to demand prediction, addressing both planning and operational time horizons.

Future Research Directions: Future research directions include several promising avenues for extending this work. Model extensions could explore negative binomial regression to address the remaining overdispersion in count data. Spatial integration represents another direction through incorporating spatial dependencies between nearby stations via graph-based or spatial autoregressive models. Real-time adaptation could be achieved by implementing online learning approaches to adapt to changing patterns and seasonal variations. External factors present additional opportunities through incorporating data sources such as events calendars, public transit schedules, and construction impacts.

4.3 Support Vector Machines

Table 5: SVM Confusion Matrix for $C = 0.1$ and Unbalanced Class Weighting (accuracy = 0.87)

	Predicted	
Actual	No demand	Has demand
No demand	0 (0%)	46403 (100%)
Has demand	0 (0%)	304637 (100%)

Table 6: SVM Performance Metrics for $C = 0.1$ and Unbalanced Class Weighting

Metric	**Value**	**Interpretation**
Accuracy	0.87	Correct predictions on 87% of cases
Precision (Has Demand)	0%	0% of predicted Has Demand cases correct
Recall (Has Demand)	0%	0% of actual Has Demand cases captured
F1 Score (Has Demand)	0%	Balance of precision and recall
Precision (No Demand)	87%	87% of predicted No Demand cases correct
Recall (No Demand)	100%	100% of actual No Demand cases captured
F1 Score (No Demand)	93%	Balance of precision and recall

Table 7: Confusion Matrix for $C = 0.1$ and Balanced Class Weighting (accuracy = 0.60)

	Predicted	
Actual	No demand	Has demand
No demand	27429 (59%)	18974 (41%)
Has demand	122659 (40%)	181978 (60%)

Table 8: SVM Performance Metrics for $C = 0.1$ and Balanced Class Weighting

Metric	**Value**	**Interpretation**
Accuracy	0.60	Correct predictions on 87% of cases
Precision (Has Demand)	18%	18% of predicted Has Demand cases correct
Recall (Has Demand)	59%	59% of actual Has Demand cases captured
F1 Score (Has Demand)	28%	Balance of precision and recall
Precision (No Demand)	91%	91% of predicted No Demand cases correct
Recall (No Demand)	60%	60% of actual No Demand cases captured
F1 Score (No Demand)	72%	Balance of precision and recall

The model achieved the best accuracy of 0.87 with $C = 0.1$ and unbalanced class weighting. The corresponding confusion matrix can be found in Table 5 and its associated performance metrics in Table 6. The confusion matrix for balanced class weighting can be found in Table 7 and its associated performance metrics in Table 8 Balancing class weighting solves the issue of one-class prediction, as can be seen with values in the confusion matrix being more spread-out. Moreover, precision for both classes are higher (18% instead of 0% for Has Demand, 91% instead of 87% for No Demand). However, this better homogeneity brings into conflict with the overall accuracy of the model, thus equal to 0.60.

The Adaptive Boosting model returns similar results without cross-validation. Therefore, if Adaptive Boosting returns better results with cross-validation, then it would be pertinent to try it for SVM despite the duration of the training. However, cross-validation did not solve the one-class prediction problem for Adaptive Boosting, therefore it would not be more

helpful to use it for SVM classification.

However, the results of the SVM could be improved by trying a non-linear kernel in order to bypass the possible non-linearities in the dataset. It could require an hour to train the model once, because of the large size of the dataset. Therefore, if we want an even more well-tuned SVM model, we could train it for several other kernels, each for several values of the regularisation coefficient.

This extensive hyperparameter tuning would require maybe a whole day of training, at the very best if no issues are encountered. We believe doing this would not necessarily be useful on an educational point of view and we think that the aim of this project is to test multiple machine learning algorithms following the machine learning methodology, test the limits of the algorithms and compare them to have a general overview of these different techniques.

4.4 Random Forest

The Random Forest model demonstrated superior performance among the tested machine learning approaches for this specific regression task. It achieved an RMSE of 1.9936 and an MAE of 0.8420, meaning the model's predictions are, on average, within 1 bike of the actual demand. The R^2 score of 0.4039 indicates that the model explains roughly 40% of the variance in the data.

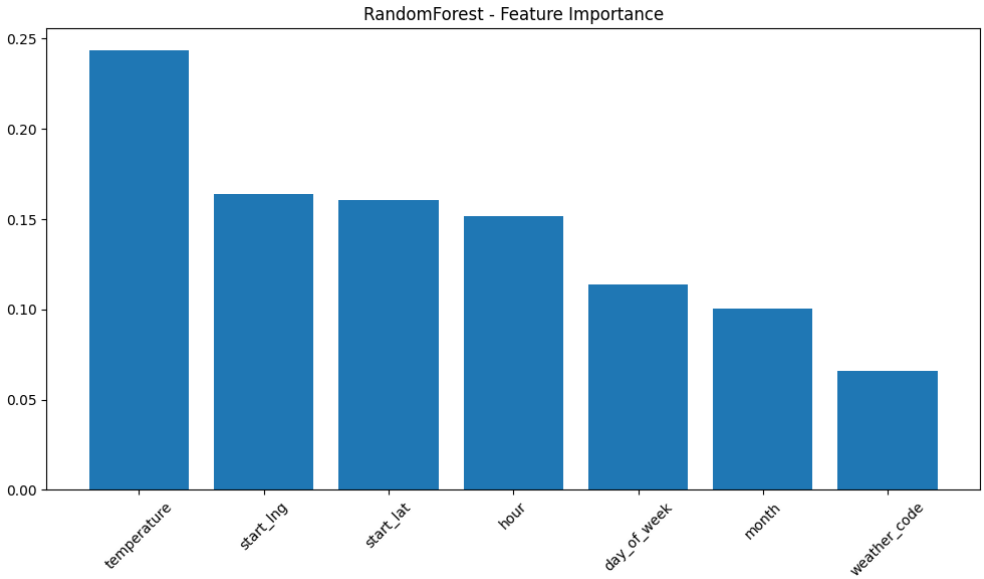


Figure 4: Feature Importance for Random Forest Model. Month, Day of Week and Hour are the top predictors.

Feature importance analysis (Figure 4) highlighted that Hour and Temperature are the dominant predictors. This confirms that human circadian rhythms (commuting hours) and comfort (weather) are the primary drivers of bike-sharing usage. The model successfully

captures the interaction effects, such as the sharp drop in activity during adverse weather even during peak hours.

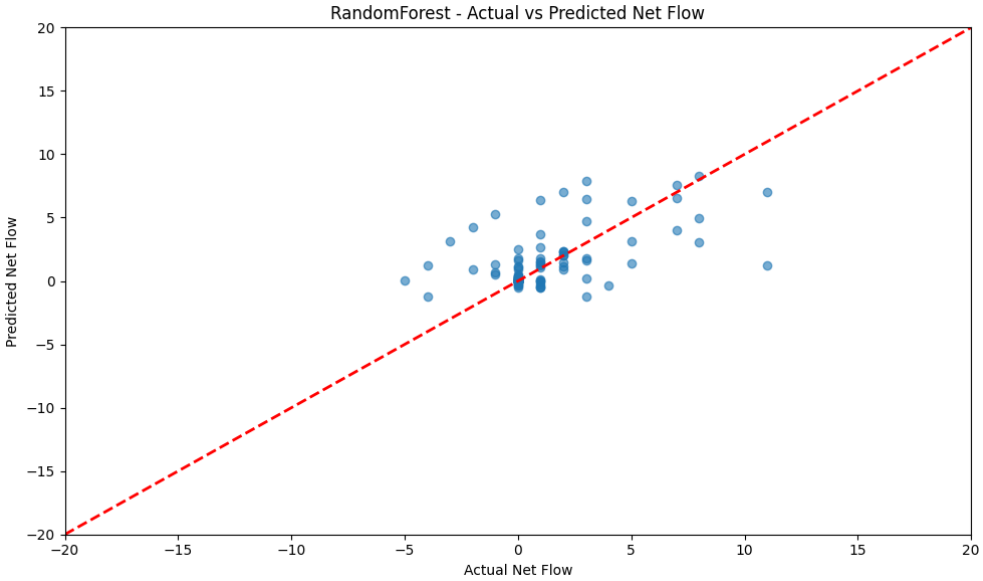


Figure 5: Actual vs Predicted Net Flow (Random Forest). The model captures the central trend but struggles with extreme values.

As shown in Figure 5, the model predictions follow the central identity line, indicating good average performance, though some variance remains for extreme demand events.

4.5 Adaptive Boosting

Our implementation of Adaptive Boosting revealed significant challenges with class imbalance despite the algorithm's theoretical robustness. We tested configurations with 50 and 100 weak learners, each comprising decision trees with maximum depth of 2. The model achieved training in approximately 12 minutes but demonstrated problematic predictive behaviour, confirmed by cross-validation, as detailed in Table 9.

Table 9: AdaBoost Performance with Class Imbalance

Metric	**Value**	**Interpretation**
Nominal Accuracy	0.87	Misleading due to class imbalance
True Positive Rate	1.00	All Has Demand cases predicted correctly
True Negative Rate	0.00	No No Demand cases identified
Precision	0.87	Reflects class distribution, not model skill
F1 Score	0.93	Skewed by imbalance

The confusion matrix for AdaBoost with 50 weak learners (Table 10) reveals the fundamental issue of complete failure to identify No Demand stations.

Table 10: AdaBoost Confusion Matrix (50 Weak Learners) for all folds

	Predicted	
Actual	No demand	Has demand
No demand	0 (0%)	46403 (100%)
Has demand	0 (0%)	304637 (100%)

The AdaBoost model's complete failure stems from several factors. With 87% of training examples belonging to the Has Demand class, the initial weight distribution heavily favored the majority class. The depth-2 decision trees provided insufficient complexity to identify nuanced patterns distinguishing stations without demand. The boosting process progressively increased weights on misclassified examples, but with the initial bias toward the majority class, this reinforced incorrect patterns rather than correcting them.

Potential mitigation strategies include explicit class weight adjustment with higher weights for the minority class during training, data resampling through oversampling No Demand cases or undersampling Has Demand cases, alternative weak learners with more complexity, or cost-sensitive boosting modifications to incorporate different misclassification costs. Despite its theoretical appeal for handling complex patterns, AdaBoost proved unsuitable for our specific classification task without substantial modification to address class imbalance. This finding underscores the importance of matching algorithm characteristics to dataset properties rather than relying on theoretical advantages alone.

4.6 Convolutional Neural Network (CNN)

The CNN model yielded an RMSE of 2.1095 and an R^2 score of 0.3326. While these results show predictive capability, the CNN underperformed compared to the Random Forest model (0.33 vs 0.40 R^2).

The training history (Figure 6) shows stable convergence without significant overfitting, but the final validation loss remained higher than the Random Forest's error.

This performance gap can be attributed to the nature of the data. Tabular data with distinct, non-sequential features, like Latitude next to Temperature, does not inherently benefit from the local spatial invariant properties of convolution kernels as much as image or spatial grid data would. The decision boundaries formed by Random Forest ensembles proved more effective for this specific feature representation. Future iterations could improve CNN performance by restructuring the input data into 2D spatial density grids to leverage the model's ability to extract spatial hierarchies.

4.7 Model Comparison

Our comprehensive evaluation of six distinct modelling approaches reveals significant differences in performance, computational requirements, and practical applicability. Table 11 summarizes the key metrics across all implemented models.

The comparison reveals several critical insights regarding temporal patterns, class imbalance challenges, and operational applicability. All models consistently identified temporal

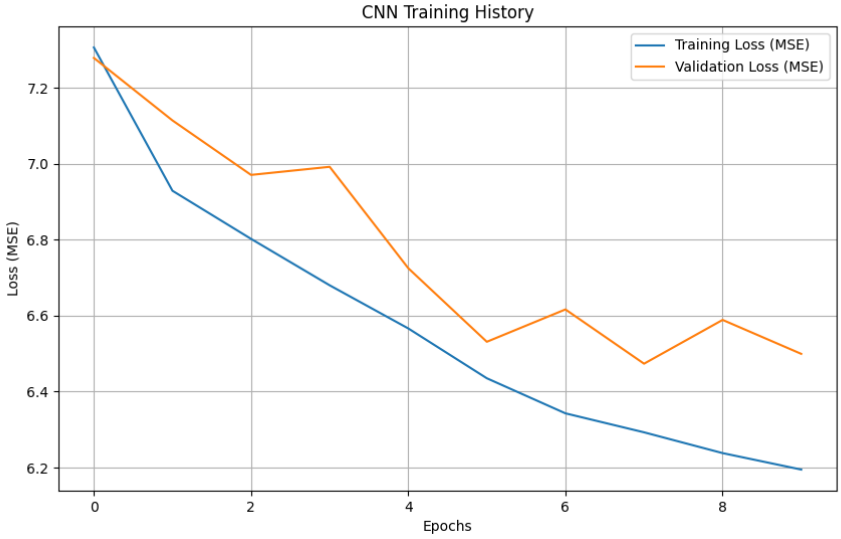


Figure 6: CNN Training History showing loss convergence.

Table 11: Comprehensive Model Performance Comparison

Model	R^2/Accuracy	Computational Time	Strengths	Limitations
Poisson Regression	0.423 (R^2)	Moderate	Highly interpretable, clear coefficients	Assumes specific distribution
Random Forest	0.408 (R^2)	High	Best predictive performance, robust	Black box, less interpretable
CNN (1D)	0.333 (R^2)	Very High	Captures complex patterns	Overkill for tabular data
Naive Bayes	0.79 (Accuracy)	Low	Fast, probabilistic outputs	Simple assumptions
SVM	0.87 (Accuracy)	High	Robust to outliers	Struggled with class imbalance
AdaBoost	0.87 (Accuracy)	Moderate	Reduces overfitting	Sensitive to noisy data

features as the strongest predictors of bike-sharing demand. Hour of day, day of week, and seasonal indicators accounted for the majority of explained variance across both regression and classification approaches. This finding validates our initial exploratory analysis and suggests that temporal patterns represent the most reliable signal for demand forecasting in bike-sharing systems.

Models making strong distributional assumptions, specifically SVM and AdaBoost, performed poorly in classification tasks due to the inherent imbalance in the dataset. Despite achieving high nominal accuracy of 87%, both models with best accuracy predicted only the majority class (Has Demand), rendering them operationally useless for identifying stations requiring rebalancing. This highlights the importance of model selection criteria beyond simple accuracy metrics, particularly for imbalanced operational datasets.

Based on our findings, we propose a tiered modeling framework for bike-sharing operations. For strategic planning at the scale of days or weeks, Poisson regression provides interpretable demand drivers and enables transparent decision-making about station placement and fleet sizing through its clear coefficient estimates. For operational forecasting at hourly resolution, Random Forest offers accurate demand predictions, justifying its use despite reduced interpretability. For real-time monitoring at the minute scale, Naive Bayes provides classification alerts with computational efficiency and probabilistic outputs that

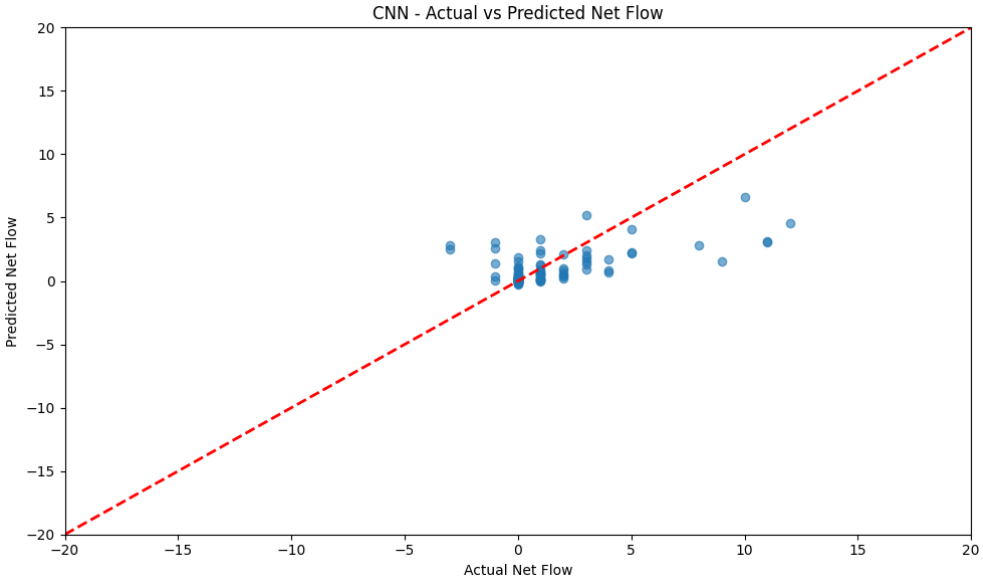


Figure 7: Actual vs Predicted Net Flow (CNN).

support immediate decision-making about bike redistribution.

The performance differential between models highlights the importance of appropriate feature engineering. Poisson regression benefited significantly from lagged variables and temporal encodings, while Random Forest's ensemble approach automatically captured complex interactions. The CNN's relatively poor performance suggests that sophisticated architectures cannot compensate for inadequate feature representation in tabular data contexts. For operational deployment, computational requirements emerged as a critical factor. Naive Bayes trained on 1.4 million observations in under 30 seconds, while CNN required significantly more resources for inferior results. This efficiency-performance trade-off must be considered in system design, particularly for real-time applications.

5 Reflections and Limitations

Our study, while comprehensive, has several limitations that warrant consideration and suggest directions for future research. Regarding **data limitations**, the analysis covers only one calendar year (2022), which may not capture long-term trends or exceptional events affecting bike-sharing patterns. Station locations are treated as fixed, ignoring potential changes in the network configuration throughout the study period. There is limited incorporation of external variables such as public transit disruptions, special events, or urban development projects that could significantly impact demand.

Methodological constraints include the station independence assumption, where models treat each station as independent, ignoring potential spatial correlations and network effects in bike-sharing systems. The hourly temporal resolution may obscure shorter-term demand fluctuations relevant for real-time operations. Manual feature engineering, while beneficial, introduces subjectivity and may overlook complex interactions that automated approaches might capture.

Modeling assumptions present further limitations. Poisson regression assumes equidispersion, which was only partially satisfied in our data (dispersion parameter: 1.310). Naive Bayes assumes conditional independence between features given the class label, an assumption frequently violated in real-world data. All models assume stationarity in demand patterns, ignoring potential concept drift due to changing user behavior or system modifications.

Practical implementation challenges must be considered for real-world deployment. Some models (CNN, Random Forest) may face challenges scaling to larger networks or higher temporal resolutions. Integration into existing bike-sharing management systems requires additional considerations for data pipeline design and model updating. The conflict between model transparency and predictive power poses challenges for stakeholder acceptance and regulatory compliance.

Our research yielded several important insights for future work in bike-sharing demand prediction. Model performance cannot be evaluated in isolation from operational requirements; accuracy, interpretability, and computational efficiency must be balanced based on specific use cases. Classification approaches must explicitly address imbalanced distributions to avoid misleading performance metrics. For tabular data, thoughtful feature engineering (as in Poisson regression) often provides greater benefits than complex model architectures (like CNN). Across all models, temporal features provided the strongest predictive signals, suggesting that investment in sophisticated temporal modeling may yield greater returns than spatial or demographic features.

Regarding the SVM and AdaBoost classification models, more extensive feature engineering, by removing the least important features, could decrease computational cost thus improving any future work on these models.

Despite these limitations, our study provides a comprehensive framework for evaluating bike-sharing demand prediction models and offers practical guidance for system operators. The explicit comparison between interpretable and black-box approaches fills an important gap in the literature and supports more informed model selection decisions in operational contexts.

6 Conclusion and Future Research

6.1 Summary

This study investigated station-level hourly demand prediction for Chicago's Divvy bike-sharing system through a comparative analysis of six machine learning models. Our research demonstrates that effective demand forecasting requires careful consideration of the trade-off between model interpretability and predictive accuracy, with different approaches serving distinct operational purposes.

The key findings can be summarized in several areas. Across all models, temporal features, particularly hour of day and temperature, emerged as the strongest predictors of bike-sharing demand, confirming the system's primary use for daily commuting. Poisson regression provided transparent coefficient estimates valuable for strategic planning, while Random Forest achieved superior predictive performance for operational forecasting, albeit with reduced interpretability. Models designed for classification (SVM, AdaBoost) struggled with inherent class imbalance, while Naive Bayes offered balanced performance suitable for real-time alerts. The relative underperformance of CNN highlights the importance of matching model architecture to data structure, with tabular data benefiting more from feature engineering than complex neural architectures.

As previously stated, our research question on *How accurately can station-level, hourly demand be predicted, and how do interpretable statistical models compare to ensemble learning methods?* finds an answer in a nuanced understanding, that while ensemble methods like Random Forest achieve lower predictive accuracy (R^2: 0.408 vs. 0.423 for Poisson regression), interpretable models provide actionable insights that may justify their use in certain operational contexts.

The practical implications are clear: bike-sharing operators should adopt a multi-model approach, using Poisson regression for long-term planning, Random Forest for hourly forecasts, and Naive Bayes for real-time monitoring. This tiered strategy balances explanatory power with predictive performance across different operational time horizons.

In conclusion, this study provides both practical guidance for bike-sharing operators and theoretical insights into the trade-offs between interpretable and black-box modelling approaches. As urban mobility systems grow increasingly complex and data-rich, the ability to balance explanatory power with predictive accuracy will become ever more critical for effective system management and planning.

Our research demonstrates that successful demand prediction requires more than algorithmic sophistication, since it demands careful consideration of operational context, data characteristics, and decision-making requirements. By explicitly comparing interpretable and complex models on the same prediction task, we contribute to a more nuanced understanding of model selection in practical settings, moving beyond simple accuracy metrics to consider the full spectrum of operational needs.

As cities continue to invest in shared mobility infrastructure, the insights from this study can inform more effective system design, operation, and expansion, ultimately contributing to more sustainable and efficient urban transportation networks.

6.2 Future Research Directions

Based on our findings and limitations, several promising directions for future research emerge. Advanced modelling approaches could incorporate explicit **spatial dependencies** through graph neural networks or spatial autoregressive models to capture network effects in bike-sharing systems. For this, the spatial visualisation done during the exploratory data analysis phase of the project (Section 2.2) is a start to this work.

Furthermore, **hybrid architectures** could be developed to combine the interpretability of statistical approaches with the predictive power of ensemble methods, potentially through model distillation or explanation techniques. Deep learning architectures specifically designed for tabular data, such as TabNet or Transformers, may overcome CNN limitations.

Enhanced feature engineering represents another research avenue. Dynamic spatial features could incorporate time-varying elements such as nearby event schedules, public transit disruptions, or real-time traffic conditions. User behavior modeling could develop features capturing patterns through clustering approaches that identify distinct user segments with different demand characteristics. Network effects could be explicitly modeled through bike flow between stations to capture redistribution dynamics and network-level constraints.

Operational integration requires further development. Real-time adaptation could implement online learning approaches that continuously update models based on new data, adapting to changing patterns and seasonal variations. Decision support systems could be developed as integrated platforms combining demand prediction with optimization algorithms for bike redistribution and station management. **Uncertainty quantification** could enhance models to provide confidence intervals or prediction intervals, supporting risk-aware decision-making in operational contexts.

Broader applications of this research include **transfer learning** to investigate model transferability across different cities or bike-sharing systems, reducing data requirements for new deployments. **Multimodal integration** could extend the framework to incorporate interactions with other transportation modes, supporting integrated urban mobility planning. Sustainability impact assessment could use demand prediction models to estimate environmental benefits and support sustainability reporting for bike-sharing operators.

Team Contributions

Each member took part in writing sections within the report and preparing the final presentation.

Member	Key Contributions	Signature
Arnaud Brisset	Spatial visualisation, implemented the AdaBoost classifier and SVM classifier	
Théo Couerbe	Temporal analysis and visualisation, implemented the Random Forest and CNN regression models	
Maria Stivala	Research on the background and context, implemented the Poisson Regression and Naive Bayes	

References

[1] E. Fishman, "Bikeshare: A review of recent literature," *Transport Reviews*, vol. 36, no. 1, pp. 92–113, 2016, doi: 10.1080/01441647.2015.1033036.

[2] J. Sears, B. S. Flynn, L. Aultman-Hall, and G. S. Dana, "To bike or not to bike: Seasonal factors for bicycle commuting," *Transportation Research Record*, vol. 2314, no. 1, pp. 105–111, 2012, doi: 10.3141/2314-14.

[3] M. Ahillen, D. Mateo-Babiano, and J. Corcoran, "Dynamics of bike sharing in Washington, DC and Brisbane, Australia: Implications for policy and planning," *International Journal of Sustainable Transportation*, vol. 10, no. 5, pp. 441–454, 2016, doi: 10.1080/15568318.2014.966933.

[4] D. Singhvi, S. Singhvi, P. I. Frazier, S. G. Henderson, E. O'Mahony, D. B. Shmoys, and D. B. Woodard, "Predicting bike usage for New York City's bike sharing system," in *Proc. AAAI Workshop on Computational Sustainability*, 2015.

[5] K. Kim, "Investigation on the effects of weather and calendar events on bike-sharing according to the trip patterns of bike rentals of stations," *Journal of Transport Geography*, vol. 66, pp. 309–320, 2018, doi: 10.1016/j.jtrangeo.2018.01.001.

[6] M. Hyland, Z. Hong, H. K. R. de Farias Pinto, and Y. Chen, "Hybrid cluster–regression approach to model bikeshare station usage," *Transportation Research Part A: Policy and Practice*, vol. 115, pp. 71–89, 2018, doi: 10.1016/j.tra.2017.11.009.

[7] H. Feng, "Predicting the dynamic demand of bike-sharing system in Chicago with Divvy operation data: A data-driven approach for bike-sharing demand forecasting," in *Proc. 5th Int. Conf. on E-Commerce, E-Business and E-Government (ICEEG)*, Rome, Italy, 2021, pp. 29–34, doi: 10.1145/3466029.3466035.