My Toronto Bikeshare Journey: Unraveling Patterns and Predictions using Machine Learning

Imagine you manage a bustling network of bikes spread across Toronto. You want to keep things running smoothly – making sure there are enough bikes where people need them, when they need them. You also want to know who's using your bikes – annual members or casual riders – so you can tailor your services better.

This project aims to help achieve these goals by:

1. **Forecasting Station Demand:** The goal is to predict how many bike trips will originate from a specific station during a given time. It's like predicting the number of people expected to attend a concert to ensure adequate staffing and entry points.
2. **Classifying User Types:** The project seeks to identify whether a rider is an annual member, or a casual user based on their trip characteristics. Imagine a detective piecing together clues to solve a case; in this scenario, the clues are trip duration, start and end stations, and time of day.
3. **Clustering Stations:** The idea is to group similar stations together. Picture drawing circles on a map to connect stations that exhibit similar patterns, like high usage during rush hour or weekends.

These insights would help make bike sharing in Toronto more efficient and enjoyable for everyone.

About the Dataset and its Sources

Our data journey starts with information freely available from the City of Toronto. It's like opening two treasure chests, each containing different but valuable clues:

1. **Ridership Data:** This chest holds the records of each bike trip – like a diary documenting every adventure, noting how long the trip was, when and where it started and ended, and whether the rider was an annual member or a casual user. This data is of year 2018.

   Source: City of Toronto Open Data Portal - Bike Share Toronto Ridership Data [1]

2. **Station Information Data:** This chest contains details about each bike station – like an address book listing each station's name, location coordinates (latitude and longitude), and capacity. It's like having a map of all the bike stations with vital information about each location.

   Source: Bike Share Toronto GBFS Feed - Station Information [2]

Exploring the Data Landscape

By combining these datasets, we can piece together the puzzle of bike sharing in Toronto.

Ridership Data:

- **trip_id**: A unique identifier for each bike trip, like a serial number.
- **trip_duration_seconds:** The total time of the bike trip in seconds.
- **from_station_id:** The unique ID of the station where the trip started.
- **trip_start_time:** The date and time when the trip began.
- **from_station_name:** The name of the station where the trip started.
- **trip_stop_time:** The date and time when the trip ended.
- **to_station_id:** The unique ID of the station where the trip ended.
- **to_station_name:** The name of the station where the trip ended.
- **user_type:** Indicates whether the rider was an "Annual Member" or a "Casual Member".

Station Information Data:

- **station_id:** A unique ID number for each bike station.
- **name:** The name of the station, like "Union Station" or "Yonge and Dundas."
- **lat and lon:** These numbers represent the station's location on a map, like GPS coordinates. 'lat' tells you how far north or south the station is, and 'lon' tells you how far east or west it is.

Challenges faced in Adding Locations to Ridership Data:

Imagine you have two lists of bike stations, but one is from a few years ago. Some station names might have changed since then, and some stations might be missing from the older list.

- **Station Name Updates:** I had to update the old station names to match the new names, making sure both lists were talking about the same locations.
- **Filling in the Blanks:** For stations with missing addresses, I used Google Maps to find their locations and added that information to the old list.

By addressing these challenges, I created a more accurate and complete dataset, enabling a better understanding of bike-sharing patterns in Toronto. This type of data cleaning is often necessary when working with real-world datasets that change over time.

Unveiling the Data's Secrets

With a refined and location-enriched dataset, I embarked on an Exploratory Data Analysis (EDA) journey, seeking hidden patterns and insights by first understanding the dataset's structure and basic characteristics through data profiling.

I performed a series of data profiling steps:

1. **Counting:** I first counted the total number of letters and pages in the package to get an idea of its size. I found 1,922,955 trips in the year 2018.
2. **Identifying Types:** I checked whether the letters were handwritten, typed, or contained pictures to understand the different formats. In my case, it was data types of features.
3. **Finding Unique Features:** I looked for unique elements like stamps, postmarks, or return addresses to categorize the letters. Here the unique value counts of each feature.
4. **Spotting Missing Information:** I scanned for any missing pages or incomplete addresses that might make the letters difficult to understand. Luckily there were no null values.

These data profiling steps provided a solid foundation for my subsequent EDA and modeling endeavors. With a better understanding of the dataset's structures and characteristics, I was ready to uncover deeper insights.

Crafting Meaningful Features

With the data cleansed and explored, I turned my attention to feature engineering—the art of transforming raw data into insightful variables. Like a data alchemist, I sought to extract hidden knowledge and refine existing features into more potent tools for my models.

Taming the Time Dimension

First, I tackled the unruly nature of time. Recognizing that different quarters of data had different date formats, I standardized them into a consistent form. This ensured a smooth and accurate analysis across all time periods.

Next, I dissected the *'trip_start_time'* into its essential components: minute, hour, day of the week, and month. This allowed me to capture the nuances of when trips occurred. I further categorized days and months, setting their natural order to reflect the cyclical nature of time. To highlight weekend trips, I created an '*is_weekend*' column, a simple yet powerful indicator.

For clarity and model compatibility, I converted trip durations from seconds to minutes. This made the data more interpretable and user-friendly.

Rush Hour Insights

To capture the ebb and flow of city life, I created a '*rush_hour*' category. This feature identified periods of high demand, categorizing trips into "morning," "evening," and "off-peak" based on their start time.

Encoding for Clarity

With these new temporal features in place, I applied encoding techniques to represent categorical variables effectively. I mapped user types, days of the week, and months to numerical values, allowing models to understand these categories. I then used binary encoding to represent '*user_type*' and '*rush_hour*' as a series of 0s and 1s, a language models can readily understand.

To capture the circular nature of time, I implemented cyclic encoding for hour, day of the week, and month. This encoding reflects the fact that 11 PM is closer to midnight (0 AM) than it is to 10 PM.

Preparing for the Future

Finally, I converted Boolean (True/False) values to numerical (1/0) format for compatibility with certain models. As a final touch, I created an inverse cyclic encoding function to decode my encoded features back into their original form for future analysis and interpretation.

Through this feature engineering journey, I transformed raw data into a set of refined and meaningful features, ready to fuel insightful models and unlock the secrets of the Toronto Bikeshare system.

Predicting Bike Demand - The SDFM Takes Center Stage

With my feature-engineered dataset ready, I turned my attention to building a predictive model to forecast bike demand at each station. This model, dubbed **the Station Demand Forecasting Model (SDFM),** aimed to accurately predict the total number of trips originating from a given station during a specific time period.

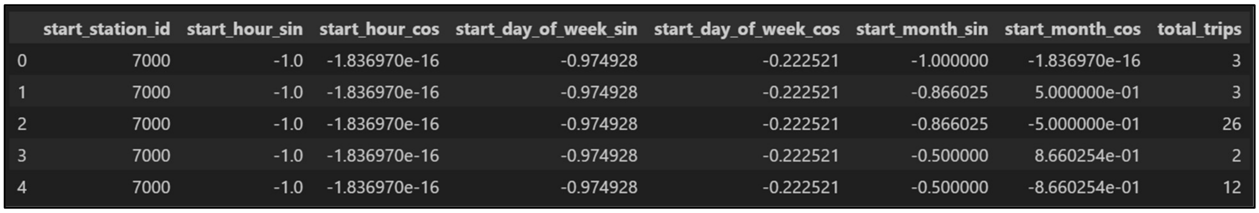
Preparing the Data for Prediction

Before unleashing the power of machine learning algorithms, I carefully prepared the data for model training.

This involved two crucial steps:

1. **Target Encoding:** To effectively represent categorical features like *'start_hour',* *'start_day_of_week'*, and '*start_month'* I applied target encoding. This technique replaces each category with the count of the target variable (total trips) for that category, providing a numerical representation that models can easily understand.
2. **Splitting the Data:** I divided the dataset into two parts: a training set and a testing set. The training set would be used to teach the model the relationships between features and total trips, while the testing set would be used to evaluate the model's performance on unseen data, ensuring its ability to generalize to new situations.

With these preparatory steps complete, I was ready to train my SDFM and unleash its predictive power on the Toronto Bikeshare data.



	start_station_id	start_hour_sin	start_hour_cos	start_day_of_week_sin	start_day_of_week_cos	start_month_sin	start_month_cos	total_trips
0	7000	-1.0	-1.836970e-16	-0.974928	-0.222521	-1.000000	-1.836970e-16	3
1	7000	-1.0	-1.836970e-16	-0.974928	-0.222521	-0.866025	5.000000e-01	3
2	7000	-1.0	-1.836970e-16	-0.974928	-0.222521	-0.866025	-5.000000e-01	26
3	7000	-1.0	-1.836970e-16	-0.974928	-0.222521	-0.500000	8.660254e-01	2
4	7000	-1.0	-1.836970e-16	-0.974928	-0.222521	-0.500000	-8.660254e-01	12

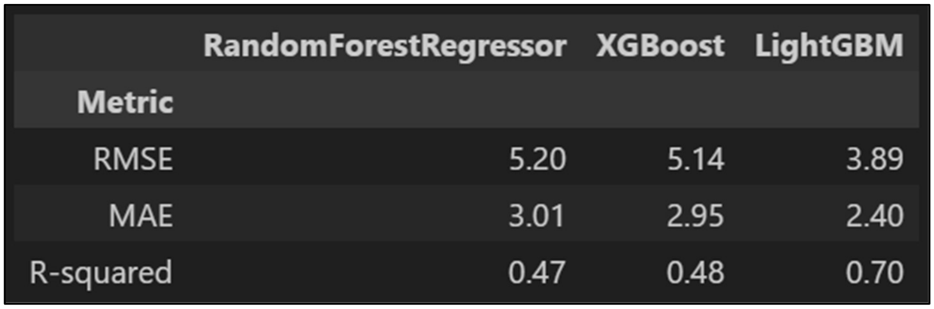
Figure 1. Sample SDFM dataset

With the data prepared, I was ready to train and evaluate three powerful machine learning models**: Random Forest Regressor, XGBoost,** and **LightGBM**. Each model brought its unique strengths to the table, offering a diverse approach to forecasting bike demand.

Training and Evaluation

I meticulously trained each model on the training data, allowing them to learn the relationships between features and total trips. To assess their performance, I used a variety of metrics:

- **RMSE (Root Mean Squared Error):** This metric measured the average difference between the predicted and actual total trips, providing a sense of overall accuracy. Lower RMSE values indicated better performance.
- **MAE (Mean Absolute Error):** Similar to RMSE, MAE also measured prediction errors, but without squaring them. This provided a more direct interpretation of the average prediction error.
- **R-squared:** This metric represented the proportion of variance in the total trips explained by the model, indicating the model's goodness of fit. Higher R-squared values signified better performance.



	RandomForestRegressor	XGBoost	LightGBM
Metric			
RMSE	5.20	5.14	3.89
MAE	3.01	2.95	2.40
R-squared	0.47	0.48	0.70

Figure 2. Comparison of Performance metrics of Station Demand Forecasting Models

Visualizing Predictions

To further assess the models' performance, I created a scatter plot comparing the actual total trips to the predicted total trips for each model. This visualization provided a visual representation of the model's accuracy and ability to capture the overall trend of bike demand.

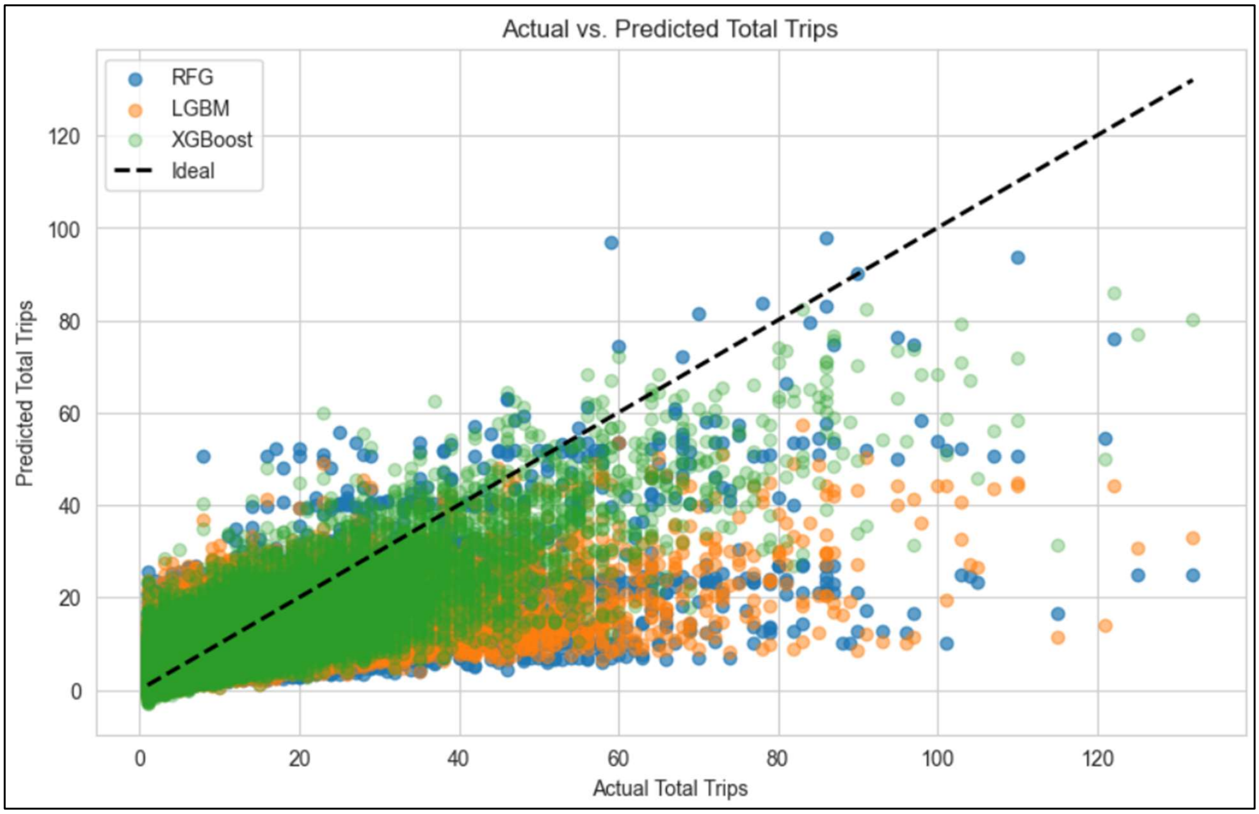
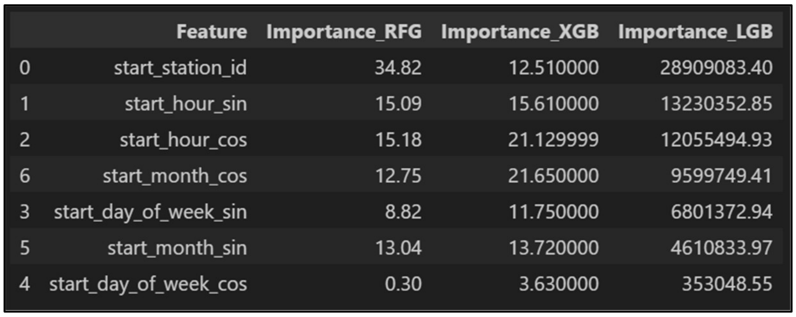


Figure 3. Predictions of SDFM

Feature Importance

Finally, I investigated the importance of each feature in the models' predictions. This helped me understand which factors most significantly influenced bike demand.



	Feature	Importance_RFG	Importance_XGB	Importance_LGB
0	start_station_id	34.82	12.510000	28909083.40
1	start_hour_sin	15.09	15.610000	13230352.85
2	start_hour_cos	15.18	21.129999	12055494.93
6	start_month_cos	12.75	21.650000	9599749.41
3	start_day_of_week_sin	8.82	11.750000	6801372.94
5	start_month_sin	13.04	13.720000	4610833.97
4	start_day_of_week_cos	0.30	3.630000	353048.55

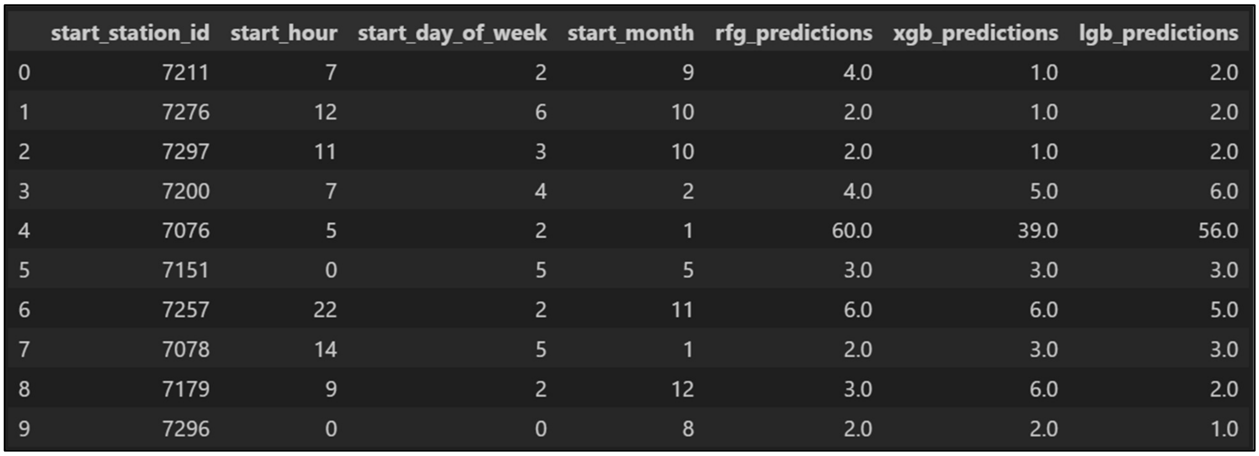
Figure 4. Feature Importance of SDFM

Conclusion

Overall, **LightGBM** appears to be the most suitable model for forecasting station demand in the Toronto Bikeshare system due to its superior performance across all evaluated metrics. LightGBM's accuracy and ability to capture data variance make it the preferred choice.

Making Predictions

At first, I randomly generated 10 sample data, and did Inverse Cyclic Encoding to make it in readable format for us.



	start_station_id	start_hour	start_day_of_week	start_month	rfg_predictions	xgb_predictions	lgb_predictions
0	7211	7	2	9	4.0	1.0	2.0
1	7276	12	6	10	2.0	1.0	2.0
2	7297	11	3	10	2.0	1.0	2.0
3	7200	7	4	2	4.0	5.0	6.0
4	7076	5	2	1	60.0	39.0	56.0
5	7151	0	5	5	3.0	3.0	3.0
6	7257	22	2	11	6.0	6.0	5.0
7	7078	14	5	1	2.0	3.0	3.0
8	7179	9	2	12	3.0	6.0	2.0
9	7296	0	0	8	2.0	2.0	1.0

Figure 5. SDFM Prediction dataset

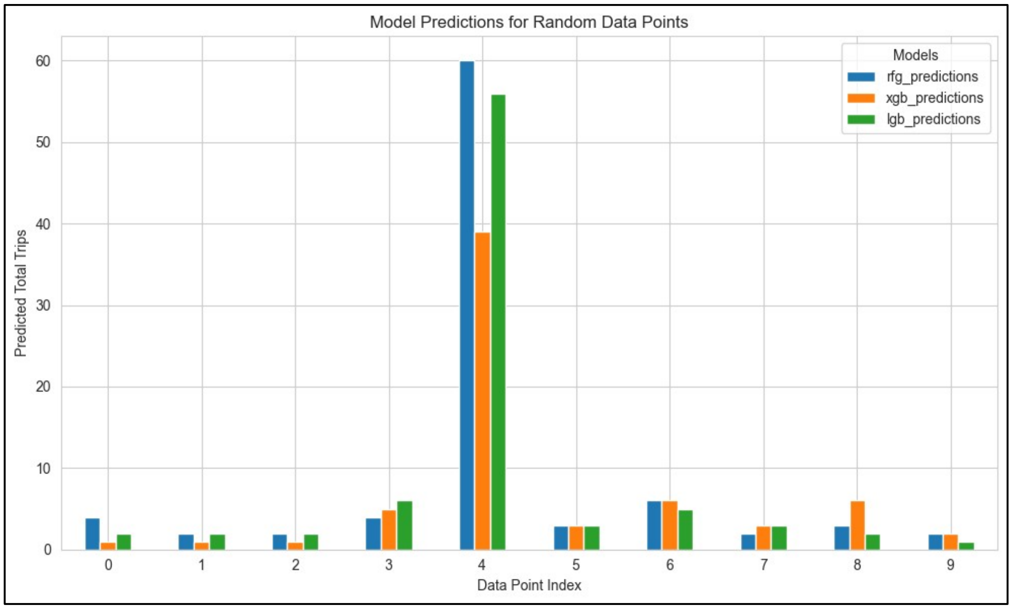


Figure 6. Output of SDFM Prediction dataset

Unveiling Station Groups - The Power of K-Means

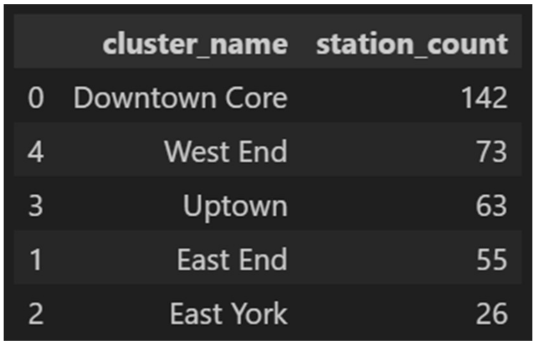
After successfully forecasting station demand, I shifted my focus to understanding the spatial relationships between stations. This involved grouping stations with similar characteristics using the K-Means clustering algorithm. This was like drawing constellations in the night sky, connecting stars (bike stations) to form meaningful patterns.

Creating Station Clusters

I applied K-Means clustering to the station data, aiming to create 5 distinct clusters based on station usage patterns and geographical proximity. These clusters were intuitively named: **"West End", "Downtown Core", "Uptown", "East End", and "East York".**

Cluster Size and Distribution

To gain an overview of the clusters, I calculated the number of stations belonging to each group. This provided insights into the relative size and distribution of clusters across the city.

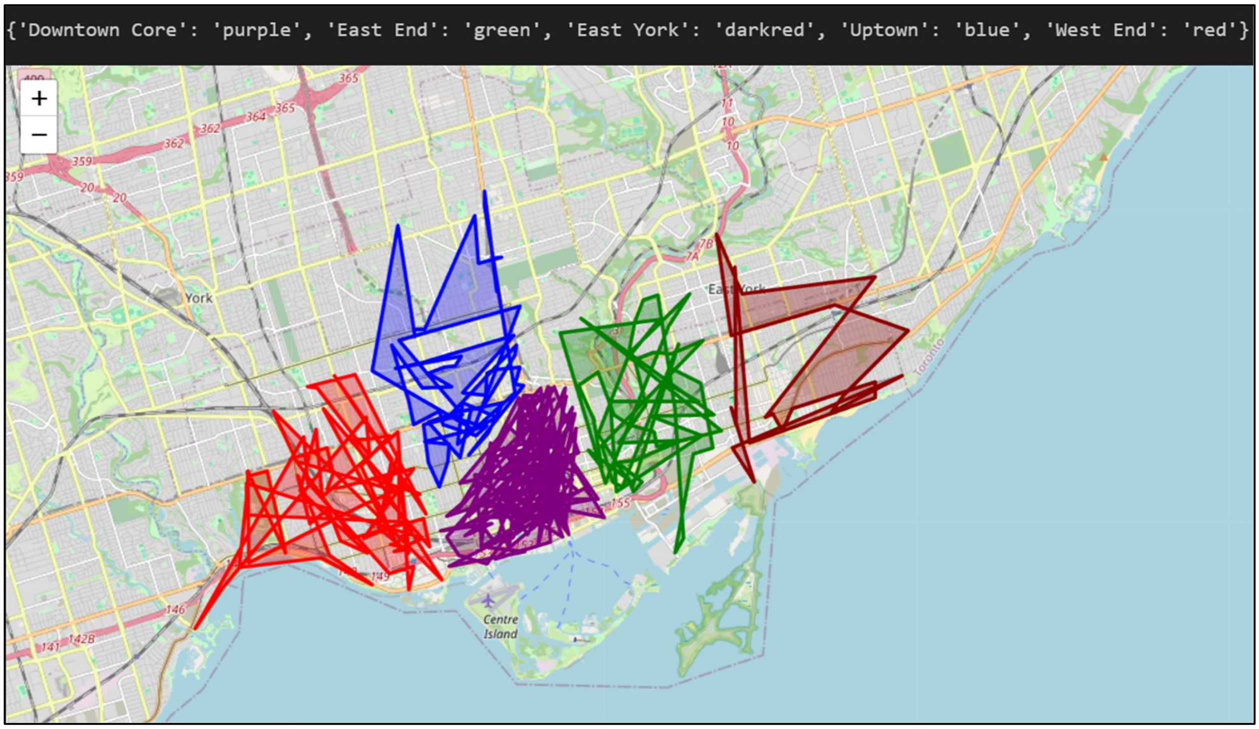


	cluster_name	station_count
0	Downtown Core	142
4	West End	73
3	Uptown	63
1	East End	55
2	East York	26

Figure 7. Cluster-wise total stations

Visualizing Clusters

I visualized the clusters using Folium, a powerful library for creating interactive maps. Despite challenges in creating perfect polygons, I managed to generate a cluster map that highlighted the geographic boundaries of each cluster.



```
{'Downtown Core': 'purple', 'East End': 'green', 'East York': 'darkred', 'Uptown': 'blue', 'West End': 'red'}
```

Figure 8. Clusters plotted on map of Toronto

To provide a more detailed view, I also created a map displaying all bike stations, color-coded according to their assigned cluster. This allowed for easy identification of stations belonging to the same group and revealed interesting spatial patterns.

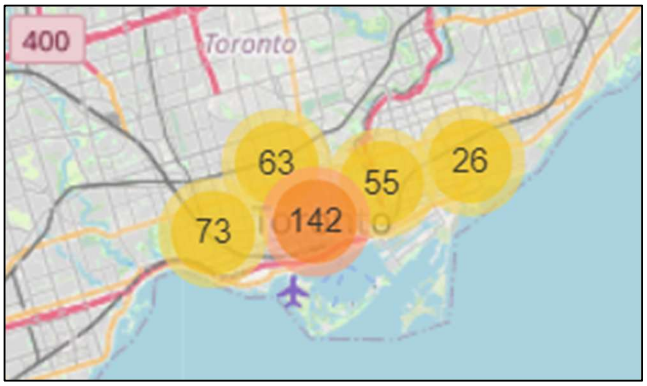


Figure 9. Stations plotted cluster-wise on map of Toronto

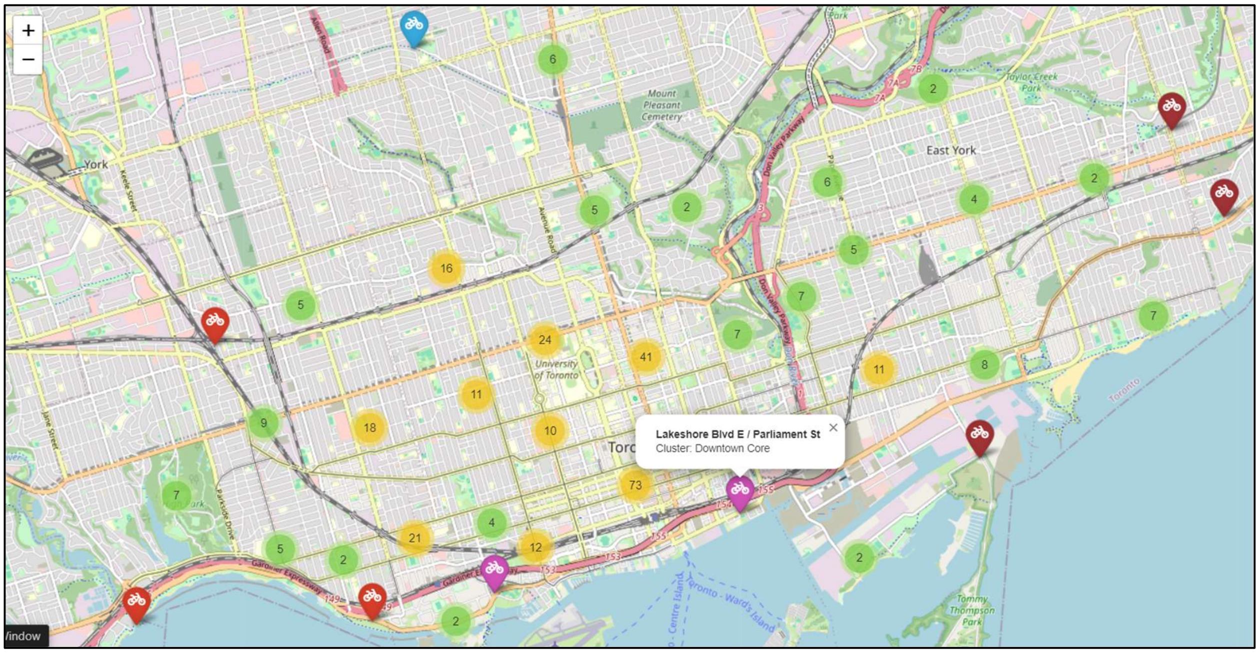


Figure 10. Stations plotted cluster-wise on map of Toronto Zoomed

Through station clustering, I gained a valuable spatial perspective on the Toronto Bikeshare system. This understanding can be used to optimize bike rebalancing efforts, strategically locate new stations, and improve overall system efficiency.

Classifying Users - Unraveling Rider Identities

With a better understanding of station demand and spatial relationships, I turned my attention to predicting user types – annual members versus casual members. This involved building a classification model that could accurately categorize riders based on their trip characteristics.

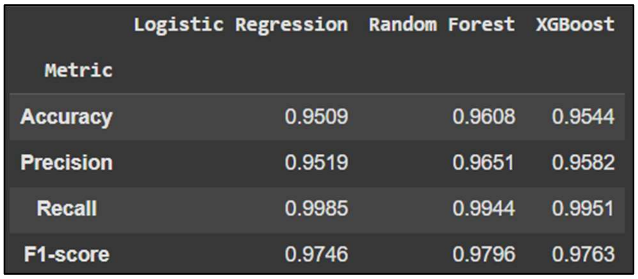
Preparing the Data and Training Models

I began by splitting the trips data into training and testing sets, mirroring the approach used for the SDFM. This ensured a robust evaluation of the model's performance on unseen data.

Next, I trained three popular classification algorithms: Logistic Regression, Random Forest Classifier, and XGBoost Classifier. Each model brought its unique approach to understanding and predicting user types.

Model Comparison and Selection

To evaluate the models' performance, I used four key metrics: Accuracy, Precision, Recall, and F1-score. These metrics provided a comprehensive view of the models' ability to correctly classify users:



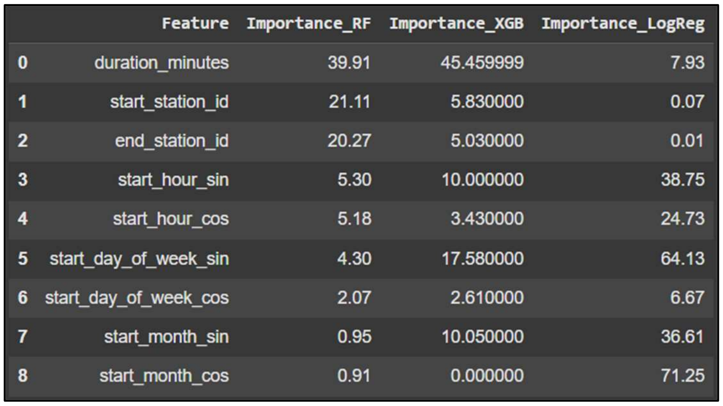
	Logistic Regression	Random Forest	XGBoost
Metric			
Accuracy	0.9509	0.9608	0.9544
Precision	0.9519	0.9651	0.9582
Recall	0.9985	0.9944	0.9951
F1-score	0.9746	0.9796	0.9763

Figure 11. Comparison of Performance metrics of User Classification Models

Based on the performance metrics, Random Forest emerged as the most effective model for user classification. It achieved the highest accuracy, precision, and F1-score, indicating its superior ability to correctly identify both annual and casual members. Random Forest's overall performance made it the preferred choice for this task.

Feature Importance and Predictions

To gain further insights, I analyzed the feature importance scores for each model, revealing the factors that most strongly influenced user type predictions.



	Feature	Importance_RF	Importance_XGB	Importance_LogReg
0	duration_minutes	39.91	45.459999	7.93
1	start_station_id	21.11	5.830000	0.07
2	end_station_id	20.27	5.030000	0.01
3	start_hour_sin	5.30	10.000000	38.75
4	start_hour_cos	5.18	3.430000	24.73
5	start_day_of_week_sin	4.30	17.580000	64.13
6	start_day_of_week_cos	2.07	2.610000	6.67
7	start_month_sin	0.95	10.050000	36.61
8	start_month_cos	0.91	0.000000	71.25

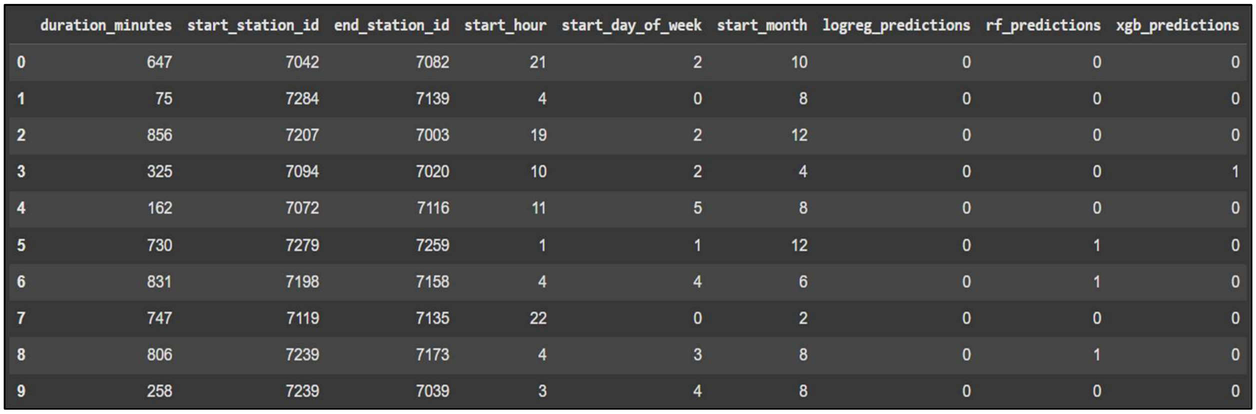
Figure 12. Feature Importance of User Classification Models

Conclusion

Similarly, the user classification model analyzed trip characteristics to identify patterns associated with different user types, enabling accurate predictions. The Random Forest model emerged as the most skilled "detective," correctly identifying users with the highest accuracy and precision.

Making Predictions

At first, I randomly generated 10 sample data, and did Inverse Cyclic Encoding to make it in readable format for us.



	duration_minutes	start_station_id	end_station_id	start_hour	start_day_of_week	start_month	logreg_predictions	rf_predictions	xgb_predictions
0	647	7042	7082	21	2	10	0	0	0
1	75	7284	7139	4	0	8	0	0	0
2	856	7207	7003	19	2	12	0	0	0
3	325	7094	7020	10	2	4	0	0	1
4	162	7072	7116	11	5	8	0	0	0
5	730	7279	7259	1	1	12	0	1	0
6	831	7198	7158	4	4	6	0	1	0
7	747	7119	7135	22	0	2	0	0	0
8	806	7239	7173	4	3	8	0	1	0
9	258	7239	7039	3	4	8	0	0	0

Figure 13. Sample dataset and Output of User Classification Prediction dataset

Insights and Impact

Through this comprehensive data science project, I embarked on a fascinating journey into the world of Toronto's bike-sharing system. By combining data analysis, feature engineering, and machine learning, I successfully achieved the project's core objectives: predicting station demand, classifying user types, and clustering stations.

Key Findings and Achievements

- **Station Demand Forecasting:** The LightGBM model emerged as the most accurate predictor of station demand, offering valuable insights for managing bike availability and optimizing resource allocation.
- **User Classification:** The Random Forest model demonstrated exceptional performance in distinguishing between annual and casual members, providing insights into user behavior and informing targeted marketing strategies.
- **Station Clustering:** The K-Means algorithm revealed meaningful spatial patterns, grouping stations with similar characteristics and enabling strategic planning for system expansion and bike rebalancing.

Impact and Recommendations

The insights gained from this project have the potential to significantly enhance the efficiency and user experience of the Toronto Bikeshare system. By leveraging the predictive power of the SDFM, operators can anticipate demand fluctuations and proactively redistribute bikes to meet user needs. The user classification model can inform targeted marketing campaigns, encouraging casual members to become annual subscribers and increasing overall system usage. The station clustering analysis can guide strategic decision-making regarding station placement, capacity planning, and bike rebalancing strategies.

Future Directions

While this project provided valuable insights, there are several avenues for further exploration:

- **Real-time Demand Prediction:** Integrating real-time data feeds and implementing dynamic prediction models could further enhance the accuracy and responsiveness of station demand forecasting.
- **Incorporating External Factors:** Investigating the impact of external factors, such as weather conditions and special events, on bike demand could refine the predictive models and enhance their accuracy.

Concluding Thoughts

This project has not only deepened my understanding of data science techniques but has also provided valuable insights into the dynamics of urban bike-sharing systems. By harnessing the power of data, I believe we can transform transportation networks, promoting sustainable mobility and enhancing the quality of life in our cities.

References

1. City of Toronto Open Data Portal - Bike Share Toronto Ridership Data. https://open.toronto.ca/dataset/bike-share-toronto-ridership-data/
2. Bike Share Toronto GBFS Feed - Station Information. https://tor.publicbikesystem.net/ube/gbfs/v1/en/station_information

Source Code

https://colab.research.google.com/drive/1YTBqm4u_E2tzTSzuJh1cw4vuZ3r7GpqV

GitHub Repo

https://github.com/mohitrathod7/Bike-Share-Toronto/tree/main/Modeling