Navigating Transit Delays: Predicting TTC Journey Disruptions

Umid Ghimire
AIML Department
Lambton College
c0888241

Sonam Sherpa
AIML Department
Lambton College
c0888896

Rabin Khadka
AIML Department
Lambton College
c0885373

Ankit Bista
AIML Department
Lambton College
c0883640

Ashok Shrestha
AIML Department
Lambton College
c0882940

***Abstract*—This project aims to create a machine learning model to forecast transit delays in the Toronto Transit Commission (TTC) public transportation system. By analyzing historical data on incidents, routes, times, and other relevant features, the project attempts to estimate delays for effective resource allocation, better planning, and enhanced service quality. The most efficient model is investigated using different machine learning algorithms, such as time series analysis, regression, and classification.**

***Index Terms*—ttc, delay, public transport, algorithms**

I. Introduction

Public transport is the most efficient commuting method within Toronto, and transit delays can significantly impact people's lives and society. The TTC comprises buses and subways, and a delay of a few seconds can cause a delay in transfer, which results in a catastrophic delay. By developing an accurate and efficient model, we aim to enable transportation authorities to make informed decisions, mitigate disruptions, and enhance overall commuter experiences. Due to various factors, the timing of trains and buses can be affected; sometimes, there might be road brokerage, some accidents etc. Those reasons can be major problems for many people. To understand the delays of trains, we can use different machine learning models on previous data, which find out some meaningful information and patterns. Using these patterns, we can estimate the timing of trains in the future. This kind of solution would be a great help for all of us to get to our destined places. The main objective is to predict Toronto Transit Commission (TTC) service delays. Since it directly affects the everyday lives of millions of commuters, delay prediction is an essential component of public transportation systems. The paper aims to improve the TTC's effectiveness, dependability, and overall user experience to become a more appealing and reliable method of transportation for the neighbourhood.

II. Related Work

Airline flight delay prediction (Tang, 2021) implements five different classification algorithms to identify the delay pattern for the flights coming in and out of JFK Airport, New York. The dataset used in this study was collected between November 2019 and December 2020 and had 23 feature sets and 28820 rows. Among five different classification algorithms, the Decision tree classifier outperforms all with accuracy and precision of 0.977 and 0.977, respectively. The paper recommends using SMOTE technique to resolve the imbalance and improve the prediction. Another similar project is Train delay prediction with machine learning and Deep learning technique (Obulesu, 2023). For this project, the researchers have implemented CNN, Random Forest, KNN and Logistic Regression. CNN has the best performance at 83% percent, closely followed [1] by Random Forest at 76%. The authors suggest implementing the NLP technique in social media posts to improve the accuracy of the prediction further. Predicting Delays in the Supply Chain with the Use of Machine Learning (Al-Saghir, 2022) is another project that helps to understand how delays in shipment are related to different groups of attributes directly or indirectly. The author implemented CRISP-DM as the primary approach for data mining and analysis. Among the four classification models used, Logistic Regression has the highest accuracy of 75%. The author suggests adding new features to improve the overall accuracy of the model. Similarly, a paper on flight delay prediction using machine learning techniques on Eygptian flights (Mohamed et al., 2018) proposes using a Decision tree classifier to predict the delays with higher accuracy. Among the four decision tree classifiers used, REPTree shows the highest accuracy of 80.3%, and the paper also extends to the apriori association technique to extract valuable information regarding flight delays.

III. Materials & Methods

The different techniques and steps were followed; and considered to building a pipeline of different methods. Here are some methods that we follow:

A. Data Extraction & Preparation

Data preparation is the process of preparing data for analysis or modelling by cleaning, organizing, and changing it. In this step of the data preparation process, we altered the dataset's ”time” column. Initially, the 'time' column carried time values in the form of hours and minutes. This format was modified to show time as the total number of minutes since midnight. This transformation makes it easier for us to work with time consistently and numerically, facilitating the analysis and comparison of time-related data. We simplify the data and make it appropriate for various analytical and modelling tasks that take time into consideration by converting the time to minutes. In our dataset, various columns included categories, such as the route number, the day of the week, the location, the kind of event, and the kind of delay. Because algorithms enjoy numbers, we transformed these categories into numbers, but we did so in a way that gave each category a unique number. In this manner, the algorithms can handle these columns more readily when we use them for tasks like making predictions or data analysis. Preparing the data for the following steps in the analysis or modelling process is like assigning each category a unique code that the algorithm can decode.

1) Relation between attributes: The Kendall rank correlation coefficients (a non-parametric measure) provide valuable information about the ordinal relationships between categorical attributes. Notably, the coefficient between ”route” and ”location” is 0.167, indicating a moderately positive correlation. This suggests that certain routes tend to be associated with specific locations, which might be useful for route planning and optimization. On the other hand, the negative correlation coefficients between ”route” and ”time,” ”route” and ”vehicle,” and ”route” and ”station delay” (-0.042, -0.046, -0.035, respectively) are relatively weak, suggesting that the route and these attributes are not strongly related in a monotonic manner.

The point-biserial correlation coefficient measures the strength and direction of the relationship between a binary (dichotomous) attribute and a continuous attribute. For instance, the point-biserial correlation between ”min delay” and ”incident” is -0.138, indicating a negative correlation. This suggests that certain types of incidents may be associated with lower delays, which might warrant further investigation into incident management practices.

Cramér’s V, used to assess the association between categorical attributes, reveals a strong association between ”delay type” and ”location” (V = 0.593) and ”delay type” and ”incident” (V = 0.477). These high values suggest that the distribution of ”delay type” varies significantly across different locations and types of incidents. This finding highlights the importance of considering location and incident type when analyzing delay types, which could be crucial for improving response strategies.

B. Statistical analysis

We observe the summary statistics for the dataset attributes. The ”Min Delay” represents the average delay in minutes, ranging from a minimum of 7 minutes to a maximum of 996 minutes, with a mean of approximately 189 minutes. The ”Min Gap” refers to the time gap between vehicles, and it has a mean of about 20 minutes, with a range from 0 to 999 minutes, suggesting some extreme values or data anomalies. The ”Vehicle” attribute has a relatively large range, ranging from 0 to 93561, likely representing unique identifiers for vehicles. The ”month” and ”day” attributes indicate the month and day when the delays occurred, with a mean month of approximately 6 and a mean day of around 16. The ”station delay” is binary, taking values of 0 or 1, possibly indicating whether a delay occurred at a particular station (1) or not (0).

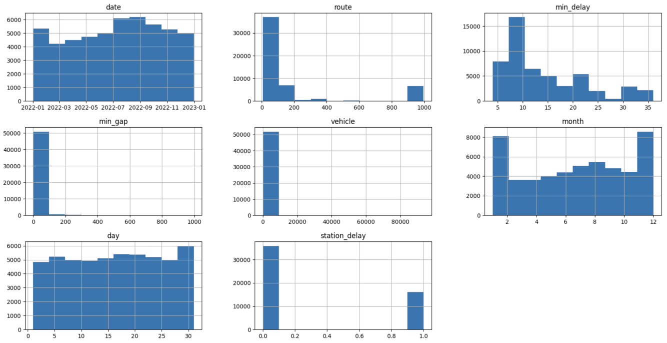
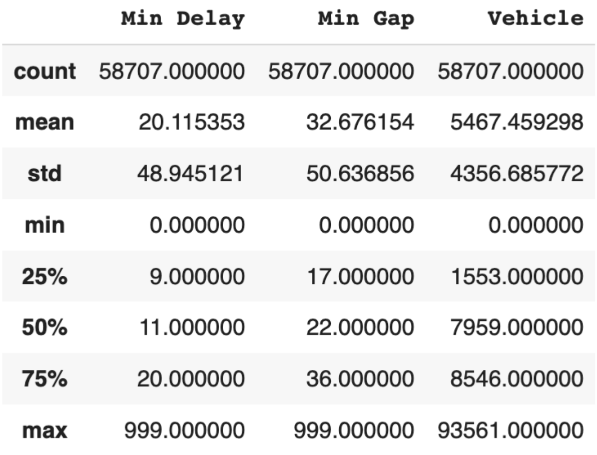


Fig. 1: Histogram Plot

The dataset statistics indicate a notable amount of variability, with relatively high standard deviations for ”Min Delay,” ”Min Gap,” and ”Vehicle,” suggesting a wide spread of values. The presence of the maximum values of 999 for ”Min Gap” and ”Vehicle” may require further investigation to determine if they are valid or if they represent some form of missing or outlier data. The ”Route” attribute, not included in the summary statistics, likely represents the different public transportation routes, but more details about the distribution of route values and the context of this data are needed for a comprehensive understanding. Additionally, the ”station delay” binary attribute shows that delays occurred at about 30 percent of the stations, on average. Further analysis and data preprocessing, such as handling missing values, outlier detection, and context-specific interpretation of the attributes, would be essential before using this data for any modelling or analysis.



	Min Delay	Min Gap	Vehicle
count	58707.000000	58707.000000	58707.000000
mean	20.115353	32.676154	5467.459298
std	48.945121	50.636856	4356.685772
min	0.000000	0.000000	0.000000
25%	9.000000	17.000000	1553.000000
50%	11.000000	22.000000	7959.000000
75%	20.000000	36.000000	8546.000000
max	999.000000	999.000000	93561.000000

Fig. 2: Statistics

C. Data Visualization

Data visualization is the process of transforming complex data into visual formats like charts and graphs to make it easier to understand and to show insights that raw data could conceal. It is an effective tool for communicating, assisting with the comprehension of patterns, trends, and connections within the data, ultimately facilitating well-informed decisions and compelling presentations.

We observed different visualizations during our project. It helped us to understand the relationship between different attributes in the datasets.

1) Number of bus date count: We analyzed bus date counts through two visualizations. The first is a KDE histogram combo highlighting frequency distribution, while the second is a box plot indicating central tendency and outliers. We can observe that the distribution of buses over time and very few outliers may be present in the dataset.

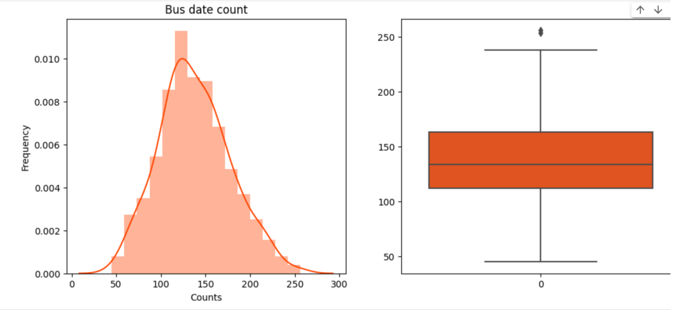


Fig. 3: Number of bus date count

2) The time values count and its frequency: We investigated bus time distribution using two graphical representations. The initial plot consists of a KDE histogram combination, showcasing the distribution of time frequencies. This is followed by a box plot that provides insights into the central tendencies and potential outliers in the time data. The figures, displayed side by side, offer a comprehensive view of the distribution characteristics. Properly labelled axes and the title 'Time value counts in the bus' enhance the clarity and interpretation of the visualizations.

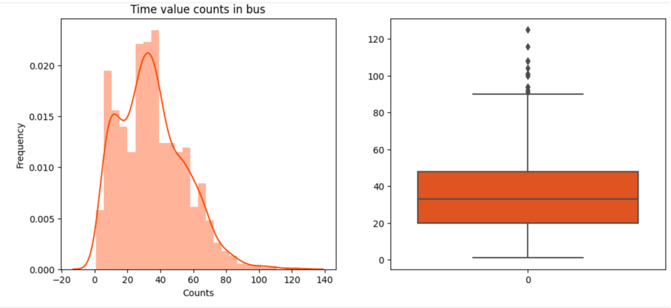


Fig. 4: The time values count and its frequency

3) Average delay time by month in 2022: We observed the average delay time per month in 2022 using a line plot. This concise representation showcases potential trends or patterns in delay durations over the course of the year.

We can observe that the 2nd (February), followed by the 4th (A 10th (October) months, has the highest number of delays, whereas the 7th (July) month has the least number of delays. It can be predicted that months right after winters have more number of delays than Summers. People usually enjoy holidays and visit different places; therefore, summers have fewer delays than any other season.

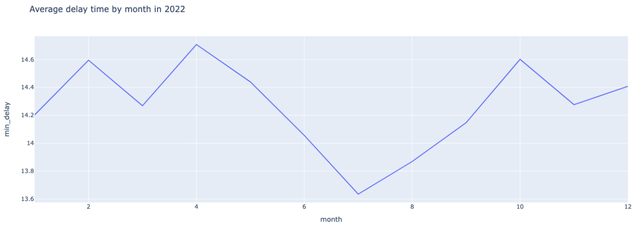


Fig. 5: Average delay time by month in 2022

4) Incident Count: We delved into incident counts for different incident types through a bar chart. The x-axis denotes incident types, while the y-axis displays the respective count values.This visualization method effectively communicates the frequency of each incident type, allowing for easy comparison and understanding.

We can observe that, the incidents or reason behind the delays are mostly due to Operations-Operator followed by Mechanical Incidents. The Least number of delays are due the Cleaning -Disinfection and Late Entering Service Incidents. We can predict that the cleaning and entering happens when the bus is no longer in service during the day.

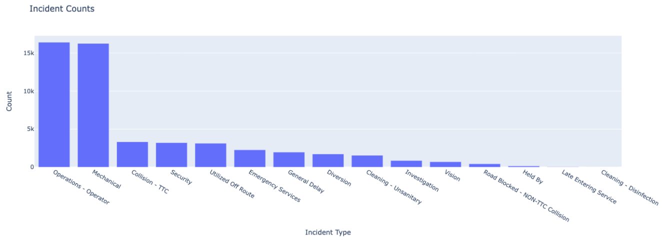


Fig. 6: Incident Count

5) Incidents by Incident Type: We explored incident distribution by incident type using a pie chart. Each slice of the chart corresponds to a specific incident type, with the size of the slice representing the proportion of incidents belonging to that type. The chart's title, "Incidents by Incident Type," concisely conveys its focus. The visualization provides a visual breakdown of incident occurrences, aiding quick comprehension.

We can observe that the most significant portion of the delays, i.e. 31 percent, are due to Operations-Operator and Mechanical Incidents.

6) Distribution of Delay Types: We examined delay types using a bar chart. The horizontal axis represents the delay types, while the vertical axis signifies their respective counts.

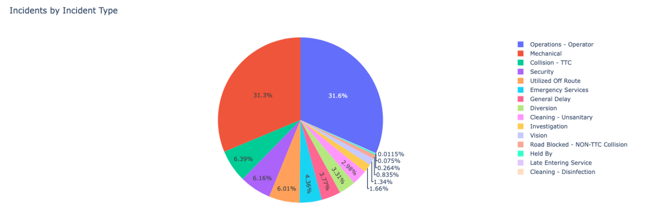


Fig. 7: Incidents by Incident Type

The visualization of delay types reveals interesting insights. Notably, the distribution demonstrates that the occurrence of ”Medium” delays is relatively high, indicating a significant frequency of such delays. In contrast, ”Long” delays appear to be comparatively rare, reflecting a lower frequency. Moreover, the occurrence of ”Small” delays is almost equivalent to that of ”Medium”. This visual representation effectively highlights the varying frequencies of different delay types within the dataset.

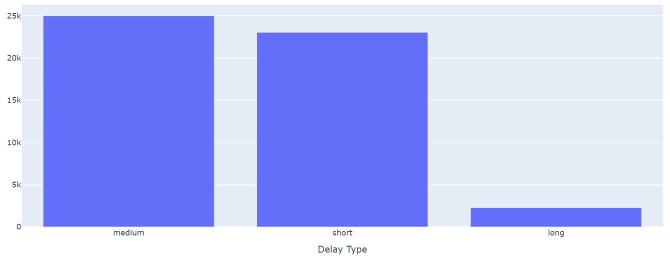


Fig. 8: Distribution of Delay Types

7) Delay Duration vs Time: We conducted a focused analysis of ”Long” delays by correlating them with the hour of occurrence. The resulting scatter plot presents a clear relationship between delay duration and the hour of the day. Each point on the plot represents a ”Long” delay incident, with the x-axis representing the hour and the y-axis depicting the delay duration. The colour-coded points provide additional information regarding the type of incident associated with each delay. The title ”Delay Duration vs. Time” succinctly captures the plot's purpose, and the axes labels effectively communicate the examined variables.

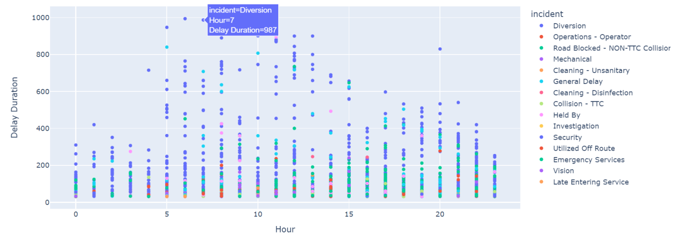


Fig. 9: Delay Duration vs Time

8) Occurrence of Min Delay by Month: We examined the occurrences of minimum delays based on their relationship with the month and incident type. Our approach involved grouping the data by month, minimum delay duration, and incident type, leading to a comprehensive dataset.

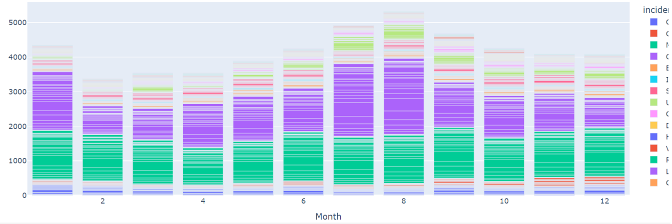


Fig. 10: Occurrence of Min Delay by Month

9) Min Delay VS Min Gap: We constructed a scatter plot using the min delay and 'map' variables. Each marker on the plot corresponds to a data point, where the x-coordinate represents the 'min delay' and the y-coordinate signifies the 'mingap'. The size of the markers is set at 10 for optimal visibility. The colour of each marker is indicative of the 'min delay' value, with a colour bar on the side indicating the value ranges. The plot's title, ”Min Delay vs. Min Gap,” concisely summarizes the content, while the x and y-axis titles clarify the plotted variables. The resulting visualization provides an intuitive comparison between the two variables' relationships. We can observe that the number of min delays and min gap are almost the same and are highly correlated.

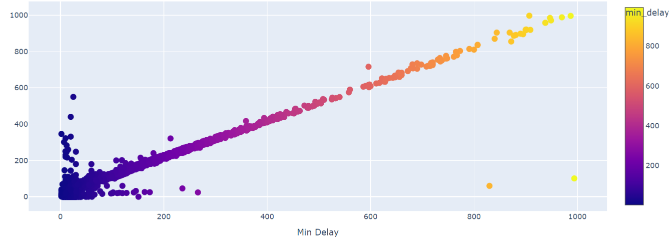


Fig. 11: Min Delay VS Min Gap

10) Incident By Day: We examined incident occurrences based on the days of the week through a bar chart. The x-axis represents different days, while the y-axis displays the corresponding incident counts. The title ”Incidents by Day” summarizes the visualization's intent, and proper labels for the x and y-axes, ”Day” and ”Count” respectively, ensure clarity. The chart offers a clear overview of incident distribution across the days of the week, enhancing our understanding of weekly patterns.

We can observe that most delays are caused on Fridays, followed by Thursdays and Wednesdays. The least number of delays are on Sundays, as the number of bus services is quite less and probably TTC bus go through the mechanical services.

D. Data Modeling

The solution to the problem can be achieved by predicting the delay before the delay occurs. This solution can give

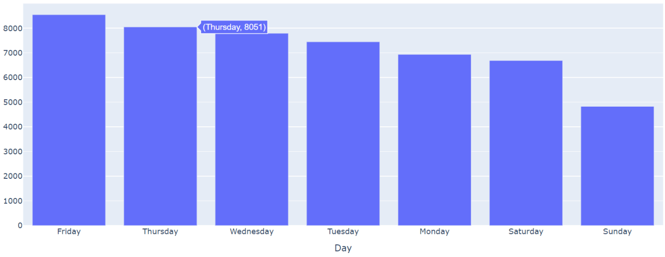


Fig. 12: Incident By Day

a head-up to the daily commuters who use TTC as public transportation. This can be executed by observing the different features in the dataset. There are a few machine learning models we can use to solve the issue of acknowledging and forecasting delays in TTC public transportation. To reach the optimized and accurate predicting model, the initial stage is to pre-process the data acquired. After careful analysis, we can have different models to justify the delays with a solution.

1) ARIMA Time Series Analysis: Analyzing historical data on transit delays within the Toronto Transit Commission (TTC) public transportation system is the time series analysis for TTC delays. Each data point will represent a delay incident on a specific transport route as the data is gathered over time. To comprehend the variables causing delays and forecast upcoming delays, the analysis seeks to find trends, seasonality, and patterns in the occurrence of delays. Time series models are very good at dealing with data which are accumulated over a long time. The TTC may manage and understand transit delays, increase service reliability, and improve passenger experience by using a variety of time series approaches, including decomposition, autocorrelation, and ARIMA models.

2) Support Vector Regression: Support Vector Regression (SVR) can be utilized to forecast TTC (Toronto Transit Commission) delays. To build a regression model that can predict future delays, SVR analyses previous data to find patterns and linkages. It effectively handles complicated, high-dimensional data by optimizing a hyperplane to reduce prediction errors. By minimizing delays and enhancing the dependability of Toronto's public transit system, SVR enables transportation authorities to implement focused policies, make educated decisions, and improve overall commuter experiences.

3) Random Forest Regression: Random Forest is a technique that can forecast TTC delays by using a variety of decision trees. It is accurate for delay analysis since it can handle non-linear correlations, interactions, and outliers in the data. By building an ensemble of decision trees and then averaging their forecasts, Random Forest Regression can be used to anticipate the length of a bus or train delay.

4) Regression with Gradient Boosting: Gradient Boosting is a potent ensemble technique that combines weak learners (often decision trees) to create a robust predictor of TTC delays. Compared to individual decision trees or random forests, it is usually more accurate.

5) Decision Trees: This algorithm can be used for Toronto Transit Commission (TTC) delay prediction. Decision trees produce a tree-like classification model by recursively segmenting the data according to attributes like the time of day, the weather, and previous delays. The paths from the root to the leaf nodes depict the decision-making process, and each leaf node indicates a delay forecast. Transportation authorities can use Decision Tree Classification to pinpoint the main causes of delays and take preventative action to lessen interruptions, boost service dependability, and improve the overall commuter experience in Toronto's public transportation system.

6) Support Vector Machine: It is a classification technique that identifies the ideal hyperplane to divide various reasons for delay. The hyperplane classifies classes in general delay, collision, mechanical etc. High-dimensional feature spaces and non-linear connections are also supported. For the classification of data, an algorithm is applied, which divides the data into different classes. For improving classification accuracy, that hyperplane is adjusted.

7) Gradient Boosting Classifier: This technique combines weak learners (usually decision trees) to create a robust predictive model for categorizing delays. It is renowned for being extremely accurate and durable. In our TTC dataset, we have different columns like date, route, day etc. According to these features, we can train the model to classify the incidents.

IV. Result & Discussion

A. *ARIMA Time Series Analysis*

For ARIMA Time Series Analysis for the value of p,q,and d, we used auto_arima library and we could find that the forecasted time series predicting the delays. We could observe that, the forecasted values didn't align with the actual delays.

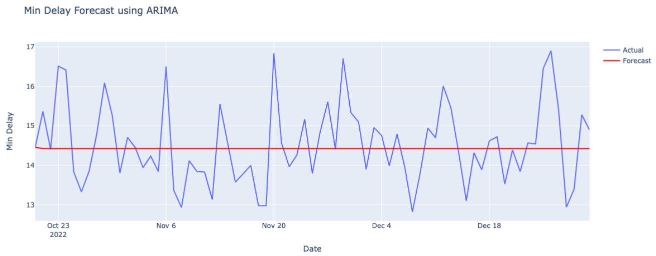


Fig. 13: ARIMA Time Series Analysis

Since we modeled our dataset with three regression model and classification models, and later optimized by hyperparameter tuning using RandomizedSearchCV.

B. *Regression*

1) Before Tuning: The Random Forest model has the best performance, with the lowest MSE and MAE, and the highest R-squared value. The Gradient Boosting model has the second-best performance, while the SVR model has the worst performance.

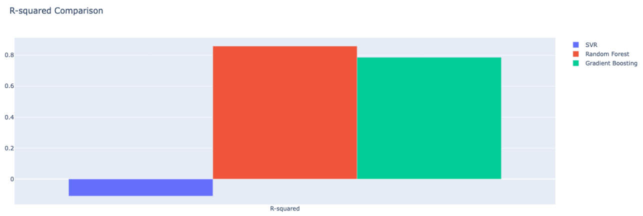
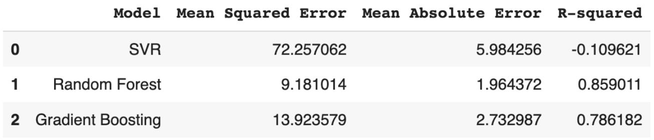


Fig. 14: Regression R-squared(before tuning)



	Model	Mean Squared Error	Mean Absolute Error	R-squared
0	SVR	72.257062	5.984256	-0.109621
1	Random Forest	9.181014	1.964372	0.859011
2	Gradient Boosting	13.923579	2.732987	0.786182

Fig. 15: Regression Results(before tuning)

2) After Tuning: From the regression after hyperparameter tuning, it is observed that,

- The Random Forest model has a lower Mean Squared Error and a lower Mean Absolute Error after hyper parameter is tuned compared to the model before tuning. The R-squared value is also higher after tuning.
- The Gradient Boosting model has a slightly lower Mean Squared Error and a lower Mean Absolute Error after tuning. The R-squared value is also slightly higher after tuning.

After tuning, The Random Forest model has the best performance, as it has the lowest Mean Squared Error and the highest R-squared value among all three models.

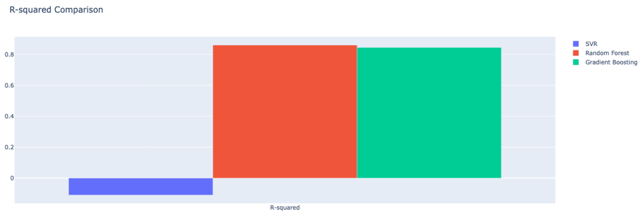
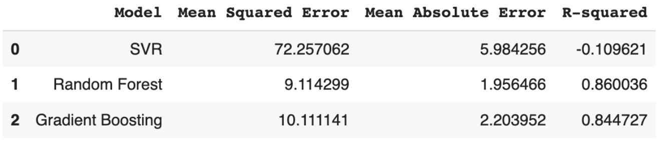


Fig. 16: Regression R-squared(after tuning)



	Model	Mean Squared Error	Mean Absolute Error	R-squared
0	SVR	72.257062	5.984256	-0.109621
1	Random Forest	9.114299	1.956466	0.860036
2	Gradient Boosting	10.111141	2.203952	0.844727

Fig. 17: Regression Results(after tuning)

C. Classification

1) Before Tuning: The following observation can be made with the metrics of different classifier model used,

- The decision tree model has an accuracy of 0.73, which means that it correctly classified 73% of the instances in the test set. The precision and recall are also both 0.73, indicating that the model is equally good at identifying positive and negative instances.
- The support vector machine model has an accuracy of 0.53, which means that it correctly classified 53% of the instances in the test set. The precision is 0.51, indicating that the model is slightly better at identifying negative instances than positive ones. The recall is 0.53, indicating that the model is equally good at identifying positive and negative instances.
- The gradient boosting model has an accuracy of 0.75, which means that it correctly classified 75% of the instances in the test set. The precision and recall are also both 0.75, indicating that the model is equally good at identifying positive and negative instances

The gradient boosting model performed the best, followed by the decision tree model, and then the support vector machine model.

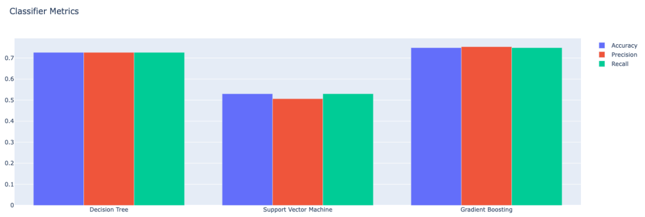
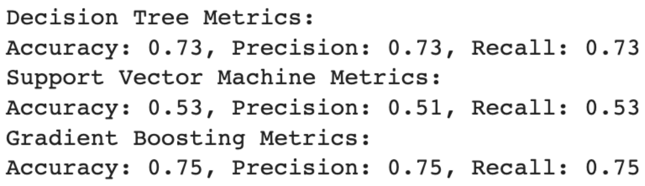


Fig. 18: Classification Results (before tuning)



```
Decision Tree Metrics:
Accuracy: 0.73, Precision: 0.73, Recall: 0.73
Support Vector Machine Metrics:
Accuracy: 0.53, Precision: 0.51, Recall: 0.53
Gradient Boosting Metrics:
Accuracy: 0.75, Precision: 0.75, Recall: 0.75
```

Fig. 19: Classification Results (before tuning)

2) After Tuning: From the regression after hyperparameter tuning, it is observed that,

The gradient boosting model performed the best, followed by the decision tree model, and then the support vector machine model.

- The decision tree model has an accuracy of 0.78, which means that it correctly classified 78% of the instances in the test set. The precision and recall are also both 0.78, indicating that the model is equally good at identifying positive and negative instances.
- The support vector machine model has an accuracy of 0.53, which means that it correctly classified 53% of the instances in the test set. The precision is 0.51, indicating that the model is slightly better at identifying negative instances than positive ones. The recall is 0.53, indicating

that the model is equally good at identifying positive and negative instances.

- The gradient boosting model has an accuracy of 0.79, which means that it correctly classified 79% of the instances in the test set. The precision and recall are also both 0.79, indicating that the model is equally good at identifying positive and negative instances.

The gradient boosting model performed the best in both models before and after the tuning, followed by the decision tree model, and then the support vector machine model. The decision tree and gradient boosting models both showed slight improvements in performance, while the support vector machine model's performance remained unchanged.

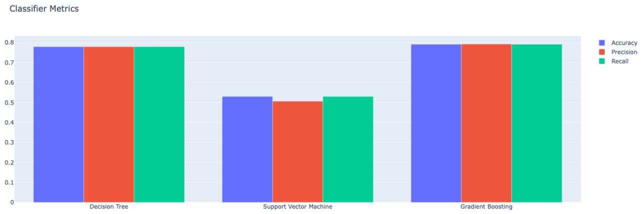
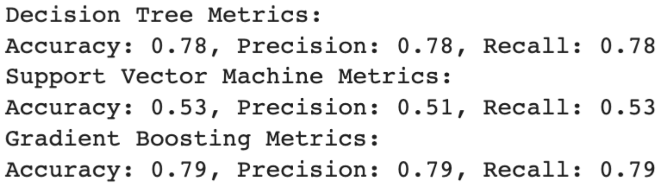


Fig. 20: Classification Results (after tuning)



```
Decision Tree Metrics:
Accuracy: 0.78, Precision: 0.78, Recall: 0.78
Support Vector Machine Metrics:
Accuracy: 0.53, Precision: 0.51, Recall: 0.53
Gradient Boosting Metrics:
Accuracy: 0.79, Precision: 0.79, Recall: 0.79
```

Fig. 21: Classification Results (after tuning)

As we explored various machine learning models, including Time Series Analysis, Regression, and Classification, to predict delays with the provided routes, vehicles, and incidents, we can accurately predict delays for any month and time. Our experiments with different hyperparameters and model training for forecasting showed that the Time Series analysis or ARIMA model was not suitable for the dataset. On the other hand, the Random Forest algorithm performed best for regression and Gradient Boosting for classification. We hypertuned the models using algorithms such as Grid Search CV and Randomized SearchCV to achieve optimized parameters. After applying these hyperparameters to our models, we observed progressive results. The impact was significant for Regression models, but it worked even better with classification.

V. Conclusion

In summary, our project aims to revolutionize the Toronto Transit Commission (TTC) experience by predicting delays in services, a crucial factor affecting daily commuters. With the goal of enhancing TTC's reliability and user satisfaction, we've employed an array of advanced algorithms, including Support Vector Regression, RandomForestRegressor, and GradientBoostingRegressor, to forecast delays accurately. Through meticulous data preprocessing and visualization, we've uncovered valuable insights, such as the dominance of ”Medium” delays and the correlation between incident types and delay durations. By providing real-time delay information to commuters, we empower them to make informed travel decisions and consequently reduce frustration and enhance urban mobility. This project not only signifies a step forward in data-driven public transportation optimization but also presents an opportunity for collaboration with transit authorities to implement tangible improvements. In conclusion, our project serves as a bridge between data science and efficient urban transportation. By analyzing the complex interplay of variables contributing to delays, we've gained a comprehensive understanding of the underlying patterns. Through predictive algorithms and real-time information dissemination, we aspire to alleviate the burden of delays on commuters and inspire a shift towards sustainable and reliable public transportation. In doing so, we believe our work has the potential to reshape the city's transport landscape, fostering a smoother, more efficient, and environmentally conscious commuting experience.

VI. Experience Reflection

Engaging in this group project to predict Toronto Transit Commission (TTC) delays has been a collaborative journey that offered a rich learning experience. As a team, we united our diverse skills and perspectives to navigate the intricacies of data analysis. The project not only enhanced our technical proficiency but also underscored the significance of teamwork in tackling complex real-world challenges. Throughout the project, we delved into data preprocessing, model selection, and interpretation of results, collectively gaining a holistic understanding of the data science process. Collaborating with fellow team members provided insights into different problem-solving strategies and encouraged open discussions about the best approaches. Each member brought their unique strengths to the table, contributing to a well-rounded analysis and fostering a cooperative environment that fostered creativity and innovation. In the end, this project was a testament to the power of collaboration and the value of combining our individual strengths to achieve a common goal. As a team, we not only advanced our analytical skills but also honed our ability to communicate complex findings effectively, a crucial skill in data science and beyond. This experience has solidified the notion that successful outcomes emerge when diverse minds come together to navigate intricate challenges, setting the stage for further collaborative endeavors in the field of data science and beyond.

References

[1] Tang, Y. (2021). Airline flight delay prediction using machine learning models. 2021 5th International Conference on E-Business and Internet. https://doi.org/10.1145/3497701.3497725.

[2] Obulesu, Dr. O. (2023). Train delay prediction using machine learning and Deep Learning Techniques. International Journal of Computational Sciences and Engineering, 13(1), 35–44. https://doi.org/10.37622/ijcse/13.1.2023.35-44

[3] Al-Saghir, R. (2022). Predicting Delays in the Supply Chain with the Use of Machine Learning.

[4] Mohamed, hanaa, Altabbakh, shahinaz, & Elzahed, H. (2018). Machine learning techniques for analysis of Egyptian flight delay. Journal of Scientific Research in Science, 35(part 1), 1–17. https://doi.org/10.21608/jsrs.2018.11706

[5] Pierre, S. (2022 November 04). A Guide to Time Series Forecasting in Python. https://builtin.com/data-science/time-series-forecasting-python

[6] Shukla, S. (n.d). Regression and Classification — Supervised Machine Learning. https://www.geeksforgeeks.org/regression-classification-supervised-machine-learning/