REVVING UP THE COMPETITION: SPEED PREDICTION IN A CAR-FOLLOWING MODEL

DAB422 – Capstone Project

Github: Car-Following-Model-Speed-Prediction

Sumit Verma

Department of Data Analytics for Business

St Clair College, Windsor, Ontario, Canada. N9A 5K3

W0796783@myscc.ca

Olabisi Fadahunsi

Department of Data Analytics for Business

St Clair College, Windsor, Ontario, Canada. N9A 5K3

W0801587@myscc.ca

Prabhjot Kaur

Department of Data Analytics for Business

St Clair College, Windsor, Ontario, Canada. N9A 5K3

w0793969@myscc.ca

Mahesh Reddy Gaddam

Department of Data Analytics for Business

St Clair College, Windsor, Ontario, Canada. N9A 5K3

w0775855@myscc.ca

Abstract

The escalating number of automobiles has led to a growing concern: traffic congestion, affecting major cities, smaller urban centers, and rural areas. This predicament results in significant time delays during journeys, increased fuel consumption, and environmental pollution. To address these challenges, car-following models and traffic flow simulations play a crucial role in managing traffic flow and improving transportation systems. This research project focused on predicting acceleration in a car-following scenario using three machine-learning models: Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN), predict the speed and intervehicle distance for the subsequent time frame and perform a comparison between the three machine learning models. The models were trained and analyzed using the Next Generation Simulation (NGSIM) dataset. The results indicated that RF outperformed KNN and SVM, achieving the best predictive accuracy, particularly when predicting at short reaction times (0.5 seconds).

The findings underscore the potential of RF for time-sensitive scenarios, with implications for improving traffic management and decision-making systems. Future research should explore the integration of KNN and SVM with other models through ensemble techniques to further enhance prediction results and address real-world challenges.

Table of Content

List of Figures

List of Tables

List of Appendix

1. Introduction

Presently, the escalation in the number of automobiles has resulted in a mounting concern: traffic congestion. This predicament is not limited to major cities but is also becoming increasingly prevalent in smaller urban centers and rural regions. The repercussions are far-reaching, encompassing significant time delays during journeys, excessive fuel consumption, and environmental pollution (Lantian et al 2017). Statistical records indicate that in the United States alone, traffic congestion accounted for approximately 5.5 billion hours of travel time delay and 2.9 billion gallons of additional fuel consumption in the year 2011, imposing a staggering cost of 121 billion dollars. Moreover, global CO_2 emissions from transportation observed a significant growth of 45% between 1990 and 2007, with projections foreseeing a further 40% increase until the year 2030. This trend closely correlates with the concerning global greenhouse effect, underscoring the urgent need for sustainable solutions.

Efficient transportation systems play a vital role in the functioning and prosperity of modern, industrialized societies. Nevertheless, the restricted road capacity and resulting traffic congestion have emerged as significant challenges in numerous countries. Car following models and traffic flow simulations play a crucial role in addressing the problems of traffic congestion, excessive fuel consumption, and environmental pollution caused by the increasing number of automobiles. Car-following model is the study of the interaction between front and rear vehicles in a single lane. It is continuous in time, and the state of the next moment is related to the state of the historical moment which makes it possible for machine learning models to find relevance.

Applying predictive analytics in machine learning models to car-following models enables enhanced capabilities in understanding and predicting driving behavior. By leveraging machine learning techniques, car-following models can analyze historical data, adapt to changing traffic conditions, and anticipate driver reactions more accurately, leading to improved traffic flow management and safer transportation systems.

In this project, our approach involved the utilization of three Machine learning models: Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) to construct and analyze a car-following model based on the NGSIM dataset. The primary objective of this undertaking is to leverage the principles of motion to forecast the speed and intervehicle distance in the subsequent time frame within a car-following scenario. A key aspect of this project entails conducting a comparative analysis of the predictions generated by the three machine learning models.

2. Literature Review

Many Car-following models have been modeled using the NGSIM dataset. Vineet et al, 2022 performed significant modeling and research comparing machine learning models and deep learning models. In their work, they trained RF, KNN, and Convoluted Neural Networks (CNN) to predict acceleration in a single lane. In their results, CNN predicted the best results for the acceleration, the R2_scores, and Root Mean Square Error.

Diverging from previous research endeavors, this project primarily centered on machine learning models possessing dual functionality, capable of both regression and classification tasks. Additionally, these models were specifically chosen for their proficiency in handling sizable datasets, exemplified by their successful application to the NGSIM dataset.

2.1. Assumptions

Certain assumptions were made to simplify the modeling process and capture the underlying dynamics of vehicle movement.

1. **No External Influences**: This model assumes there is no external factors, such as weather conditions, road surface conditions, or complex driver decision-making processes influenced by factors beyond the preceding vehicle.
2. **Neglecting Human Errors**: This model assumes the drivers consistently respond to the behavior of the preceding vehicle. Whereas in real-world situations, human errors, distractions, or abrupt decision-making may influence driving behavior.
3. **No Overtaking**: this work assumes that overtaking or lane-changing is not allowed and these data were dropped in the model, leading to a simplified single-lane car-following scenarios.
4. **No Traffic Control Systems**: This model does not consider the presence of traffic control systems, such as traffic signals or adaptive cruise control, which can influence driver behavior.
5. **Time-Invariant Parameters**: In this model, reaction time will be used, thereby disregarding potential variations.
6. **Short Time Intervals**: Since the dataset contains short time intervals between data points, assuming that driving behavior changes slowly over these intervals

2.2. Equations of motion

In a car-following model, the equation of motion is used to describe the dynamics of individual vehicles as they follow each other along a roadway. The equation of motion in figure 1. governs how a vehicle's position and speed change over time in response to the behavior of the leading vehicle and other relevant factors. The fundamental principle underlying the equation of motion is that the acceleration of a vehicle is determined by the forces acting upon it.

$$v = u + at$$

$$s = ut + \frac{1}{2}at^2$$

$$v^2 = u^2 + 2as$$

$$s = \frac{1}{2}(u + v)t$$

Figure 1. Equation of motion

Where v = final speed

u = initial speed

a= acceleration

t = time at any moment during motion

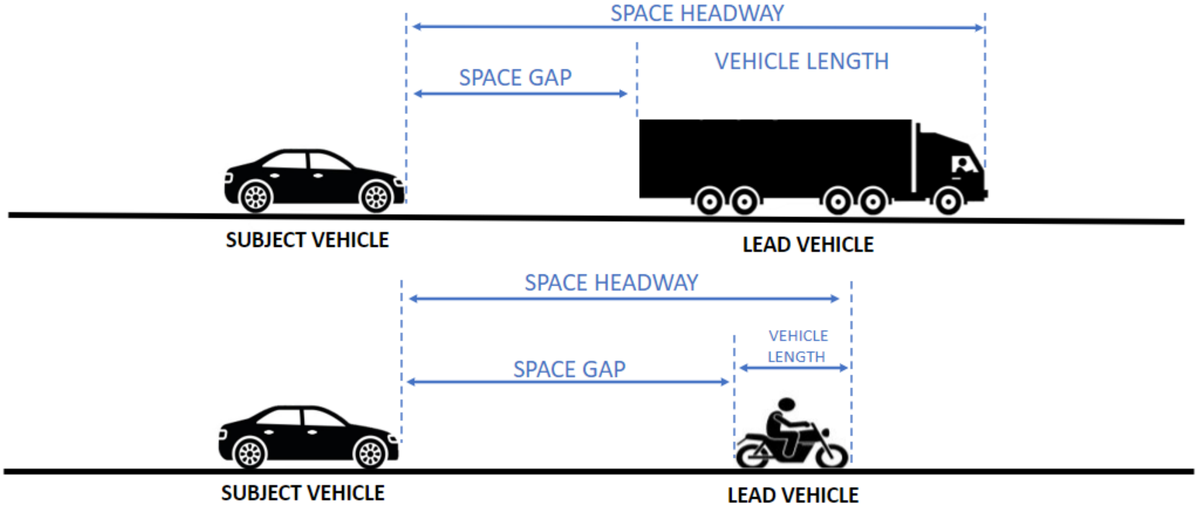

Figure 2. Common Terminologies in the model creation and prediction

Figure 2. above illustrates the standard terminologies utilized in model creation and prediction. These features, along with additional ones, are employed to train the model. In a car-following scenario, the driving behavior of the subject vehicle is influenced by the lead vehicle. For instance, the spacing between the two vehicles varies with the speed and type of the lead vehicle (greater spacing for Heavy vehicles as lead). Moreover, the reaction time of the subject vehicle is also impacted by the speed and type of the lead vehicle.

3. Data

The dataset used in this project is called the Next Generation Simulation (NGSIM) and is publicly available on the US Department of Transportation website. The utilization of this dataset remains prevalent in transportation research, particularly for examining and modeling traffic flow, estimating and predicting traffic-related factors, and studying vehicular ad hoc networks.

The NGSIM dataset consists of 25 columns and 11.8 million rows of vehicle trajectory data which was captured using a network of synchronized digital video cameras in 4 different locations (US 101, I-80). The data contains the location of a vehicle at every one-tenth of a second, which gives the exact position of each vehicle relative to other vehicles.

Dataset Link: https://datahub.transportation.gov/stories/s/Next-Generation-Simulation-NGSIM-Open-Data/i5zb-xe34/

Given the dataset's collection over a decade ago, concerns regarding the accuracy of the NGSIM dataset have emerged in recent years. The investigation revealed notable measurement errors within the NGSIM dataset, attributable in part to the utilization of low-resolution cameras and the mistracking of vehicles based on video images. Montanino et al. undertook the task of removing outliers and noise, subsequently reconstructing Dataset 1 of the I-80 dataset (captured between 4:00 p.m. and 4:15 p.m.). Their efforts yielded substantial enhancements when compared to the original NGSIM dataset.

Reconstructed dataset link: https://github.com/Shuoxuan/NGSIM_Cleaned_Dataset

3.1. Data Preparation

To enhance the predictive capabilities, the data underwent thorough cleaning and transformation processes. The accuracy of the resulting models is directly influenced by the quality of the utilized data.

As previously mentioned, our dataset comprised data from four different locations, but we specifically focused on extracting data from I-80 locations. This decision was made since the timing in the reconstructed dataset for I-80 and I-101 do not match. The model can then be generalized for I-101 later. To ensure the model's generality, redundant fields within the dataset were identified and subsequently removed. Following the procedures outlined in Figure 3, we obtained a clean dataset.

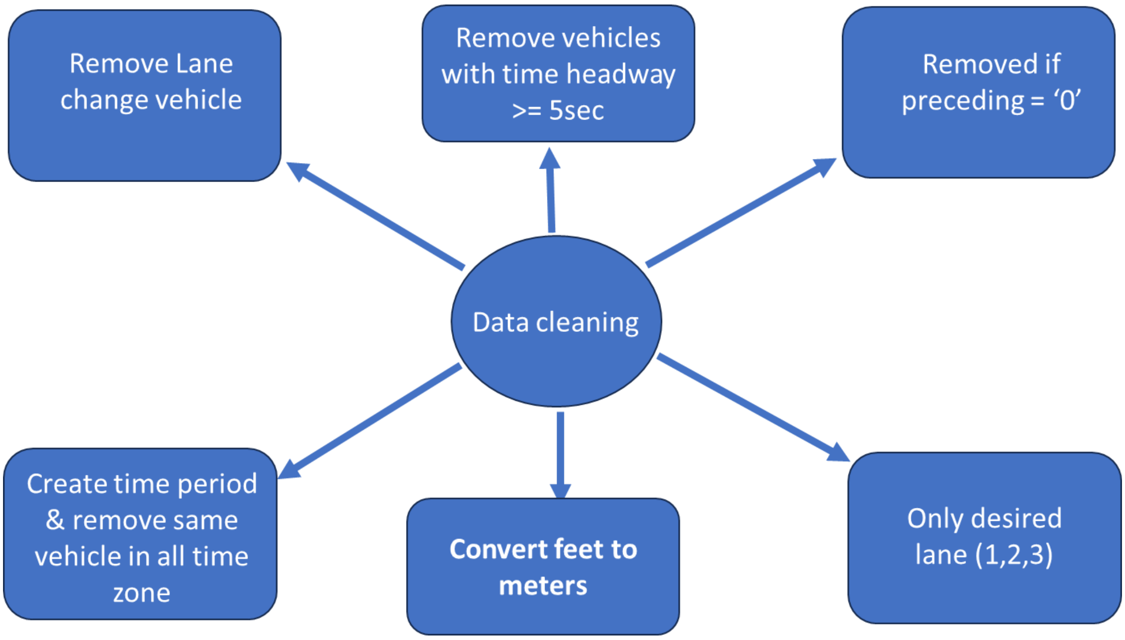

Figure 3. Data Cleaning operation

Figure 4. shows the transformation done on the dataset before creating the models, the following was done on the dataset:

a. Remove non-zero values, remove any entries related to lane changes, and then retain only the rows that meet the desired criteria for lanes in the reconstructed data for each time interval.
b. Combined the 3 cleaned datasets into one
c. all the fields' units were converted to the International System of Units (SI). This entailed converting vehicle length, local_X, local_Y, and space headway from feet to meters, velocity from miles per hour to meters per second, and acceleration from feet per second squared to meters per second squared.
d. Once the data had been cleaned and the units standardized, we proceeded to generate Lead-Following (L_F) vehicle pairs using the Preceding and Vehicle ID fields.
e. Duplicate vehicle pairs present at both locations were eliminated, and a new field for space headway from the rear of the lead vehicle to the front of the subject vehicle was created, eliminating the dependence on the vehicle type.
f. Merged dataset based on Preceding Vehicle ID and Frame ID
g. mapping is done for the values 1, 2, and 3 to the corresponding classes 'Motorcycle', 'Car', and 'Heavy Vehicle', respectively
h. Calculate the space gap by subtracting the length of the preceding vehicle from the space headway and the corresponding reaction time
i. Remove vehicles with time headway >= 5 seconds

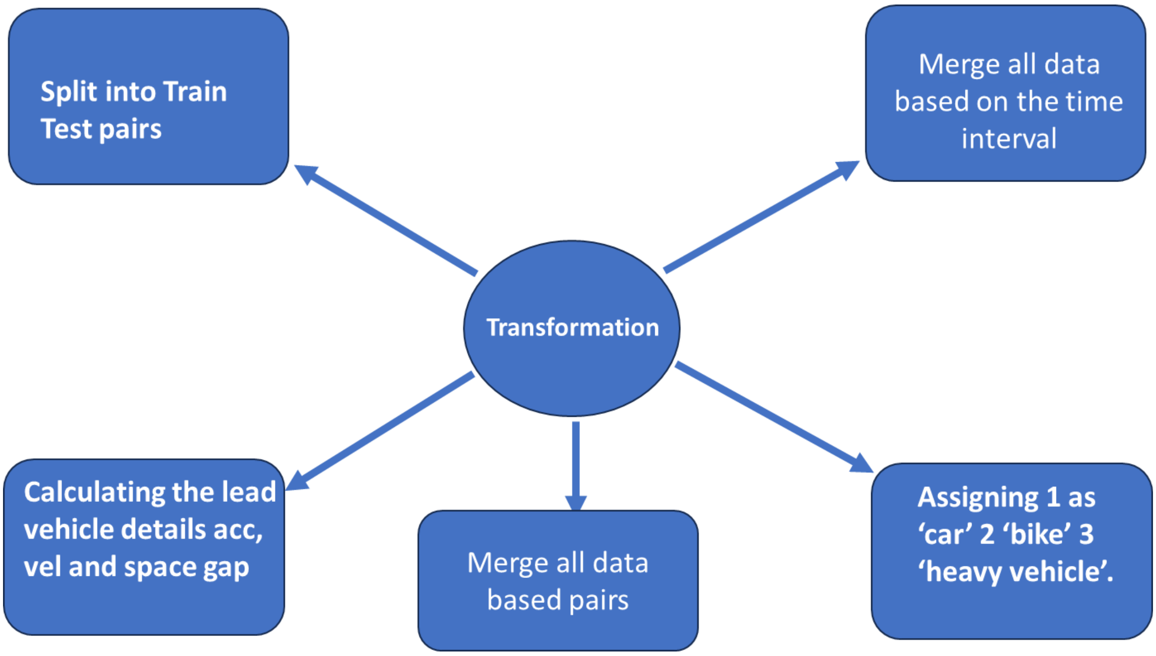

Figure 4.Data Transformation Operation

3.2. Exploratory Data Analysis

Based on the data presented in Figure 5, there is a significant concentration of vehicles traveling at speeds between 20 miles per hour and 60 miles per hour, which establishes a favorable range for the car-following model. The correlation analysis displayed in Figure 6. reveals a strong association between various factors such as vehicle ID, frame ID, following vehicle, and the preceding vehicle. These findings are instrumental in guiding the selection of relevant features for the model.

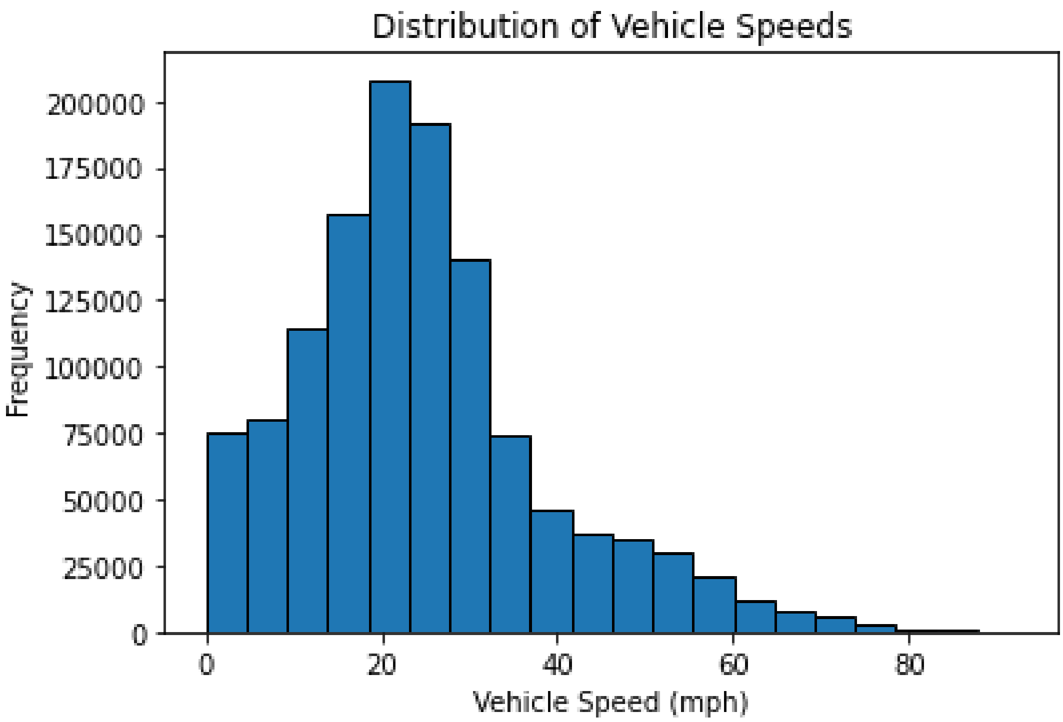

Figure 5. Distribution of vehicle speed

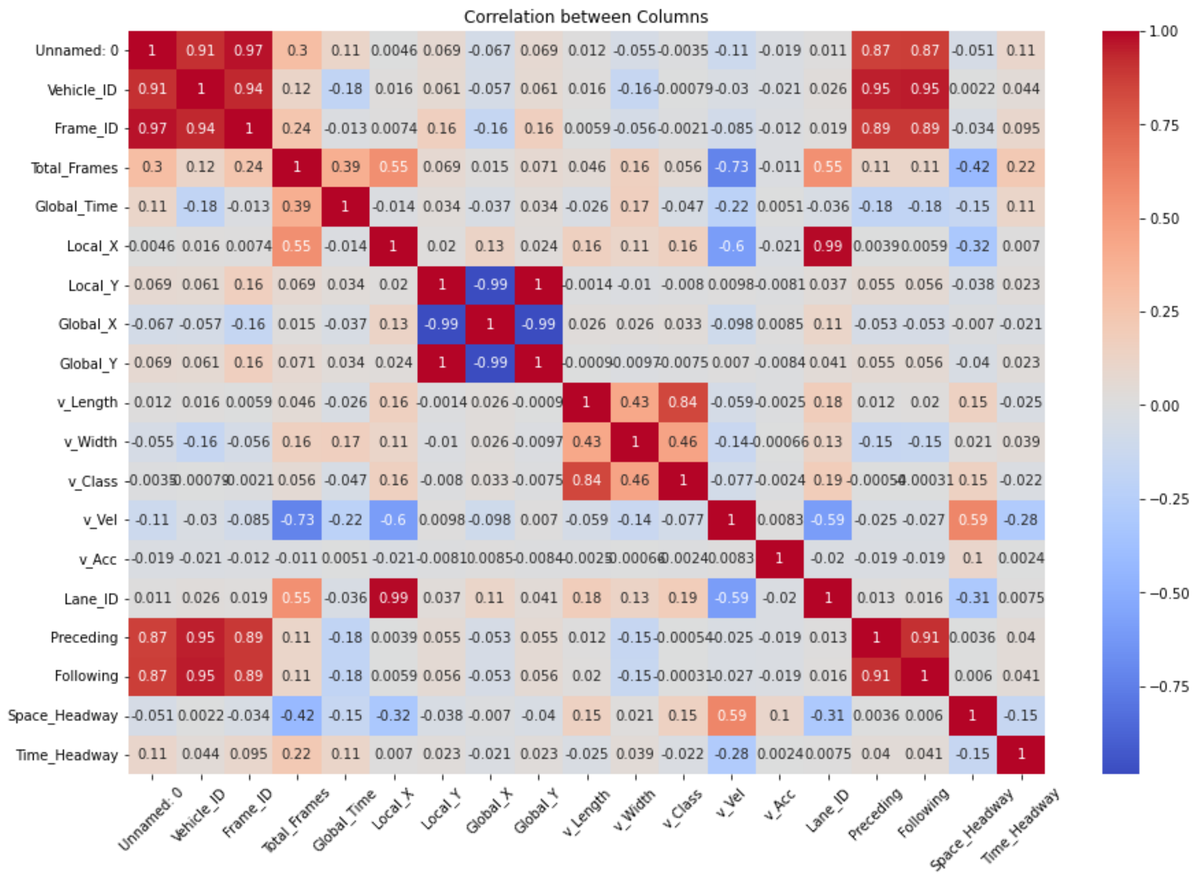

Figure 6. Correlation analysis of the features of the dataset

4. Machine Learning models

Car-following models aim to replicate how drivers adjust their speed and spacing in response to the behavior of the preceding vehicle. By leveraging machine learning techniques, these models can improve their accuracy, robustness, and ability to capture complex patterns in real-world traffic scenarios. Figure 7. below describes more terminologies in a car-following model, where the following vehicle is known as the subject vehicle and the leading vehicle is the preceding vehicle. The space gap is the intervehicle distance in a pair of lead/following vehicles and each vehicle has its velocity and acceleration.

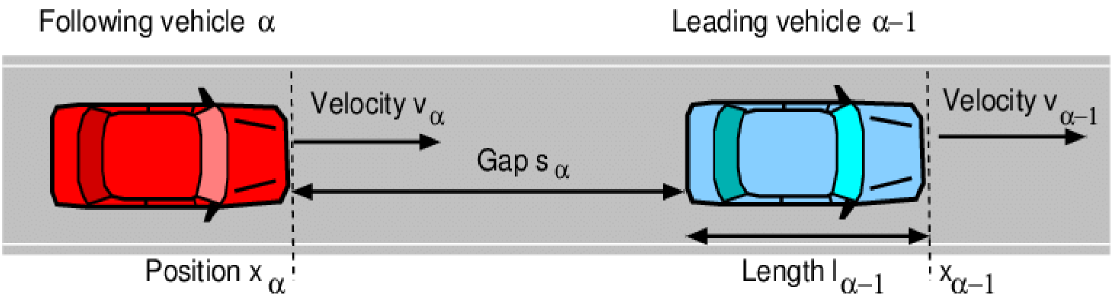

Figure 7. Car-following velocity and space gap

This project focused on predicting the acceleration of a subject vehicle in a single lane using the NGSIM dataset. Three machine learning models, namely Random Forest (RF), Support Vector Machine (SVM), and K-Nearest Neighbors (KNN) were utilized for this purpose. After obtaining acceleration predictions from each model, a loop was implemented, employing equations of motion to calculate the subject vehicle's speed and intervehicle distance for the subsequent time frames. This comprehensive approach allowed for a deeper understanding of the subject vehicle's dynamics and its interactions with other vehicles in the single-lane scenario.

4.1. Random Forest (RF)

Random Forest could handle complex relationships, diverse data types, missing data, and large datasets, along with its ensemble approach and feature importance analysis, making it a powerful and suitable choice for predicting the acceleration of subject vehicles in a single lane using the NGSIM dataset (IBM, 2023a).

The choice for Random Forest was due to its robustness and ability to handle both classification and regression tasks effectively on a large dataset like NGSIM. This is very essential in this project as it considered the different classes of vehicles and then performed a regression to predict the target variable.

- The lead/following pairs (L-F_pair) were first passed through a label encoder to transform the categorical values into numerical labels.
- The dataset was split into train/test pairs (80/20) and the model was fitted using the rf.fit function.
- The RandomForestRegressor function was utilized to predict the acceleration in the next time frame.
- The equation of motion was passed into a loop which was then used to calculate the corresponding speed and intervehicle spacing for different reaction times (0.5, 1, 1.5, 2 seconds).
- The model was then used to determine the r2_score, and root mean square error for each reaction time (0.5, 1, 1.5, 2 seconds).

4.2. Support Vector Machine (SVM)

SVM has the inherent ability to handle high-dimensional data and its resistance to overfitting makes it a reliable choice for accurately predicting the subject vehicle's acceleration in real-world scenarios based on the NGSIM dataset. Support Vector Machine (SVM) was selected for this project because of its unique strengths in handling both classification and regression tasks effectively (geeksforgeeks.com, 2022).

- Feature scaling was done using the StandardScaler to standardize the features in both the training and test datasets.
- Perform a fitting operation using svr.fit so the model will learn the relationship in the input/target variables and for onward generalization on new and unseen data.
- The svr.predict function was to predict the acceleration in the next time frame
- The equation of motion was then passed into a loop which was then used to calculate the corresponding speed and intervehicle spacing for different reaction times (0.5, 1, 1.5, 2 seconds).
- The model was then used to determine the r2_score, and root mean square error for each reaction time (0.5, 1, 1.5, 2 seconds).

4.3. K-Nearest neighbors (KNN)

K-Nearest Neighbors (KNN) is a simple and effective supervised learning algorithm used for both classification and regression tasks. It operates based on the assumption that similar data points tend to have similar labels or target values (IBM, 2023b). In the context of this project, KNN was used for regression to predict the acceleration of the subject vehicle in a single lane.

By considering the k-nearest neighboring data points from the training set, KNN can estimate the acceleration value for the subject vehicle at a particular time frame based on historical data.

- The function knn_regressor created an instance of the KNeighborsRegressor class with a specific hyperparameter while n_neighbors was set at 15.
- Knn_regressor.fit function was used to fit the KNeighborsRegressor model to the training data
- knn_regressor.predict function was utilized to predict the acceleration in the next time frame
- The equation of motion was then passed into a loop which was then used to calculate the corresponding speed and intervehicle spacing for different reaction times (0.5, 1, 1.5, 2 seconds).
- The model was then used to determine the r2_score, and root mean square error for each reaction time (0.5, 1, 1.5, 2 seconds).

5. Results

The study successfully developed predictive models for acceleration using Random Forest (RF), K-Nearest Neighbors (KNN), and Support Vector Machine (SVM) algorithms. The models utilized the equation of motion to calculate speed and intervehicle distance. By analyzing the data, the models predicted the acceleration for the next time frame. The study found that RF outperformed both KNN and SVM in terms of predictive accuracy for velocity and spacing. Figure 8 and Figure 9 clearly demonstrate RF's superiority, consistently achieving the lowest error rates and displaying better generalization capabilities. Specifically, all three models showed improved predictive accuracy for velocity and inter-vehicle distance when predicting reaction times at both 0.5 and 1-second intervals. However, RF showed the most significant improvements, especially at the 0.5-second reaction time mark. This indicates RF's proficiency in capturing underlying patterns related to short reaction times. Additionally, the analysis revealed that as the reaction time increased, the R2_score reduced for all the models. Figures 10 to 18 provide a visual representation of the actual versus predicted acceleration, velocity, and speed using RF, KNN, and SVM at a 0.5-second reaction time. These figures further support RF's superior performance in predicting acceleration based on the given data. Table 1. shows the comparison between the three models with respect to the efficiency in use.

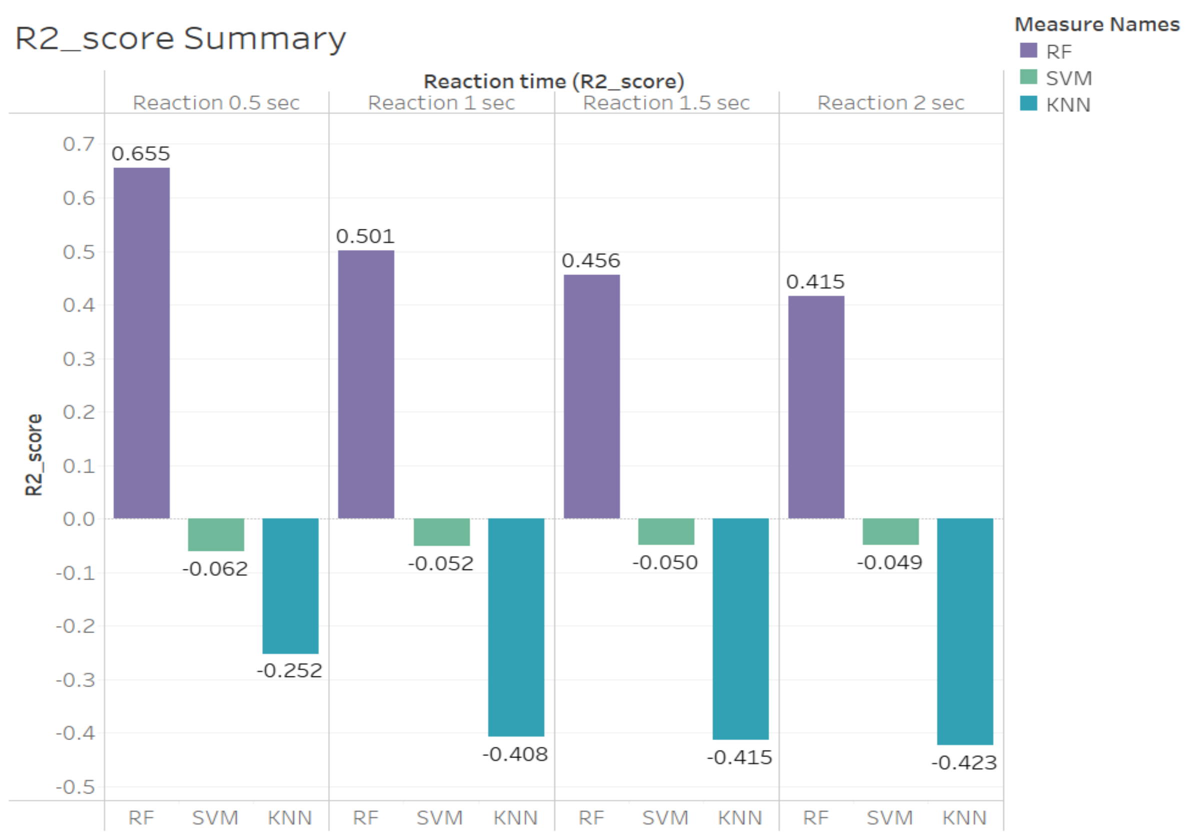

Figure 8. Comparison of models using R2_score based on reaction time

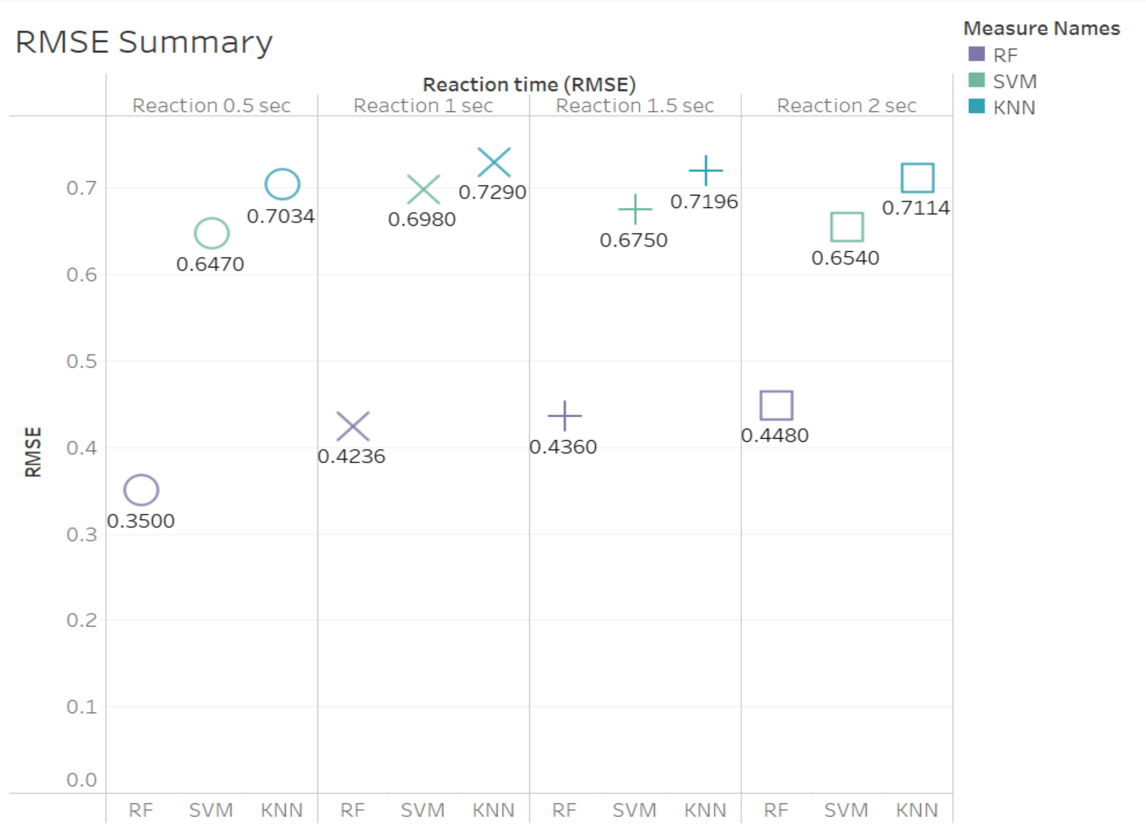

Figure 9. Comparison of RMSE based on reaction time

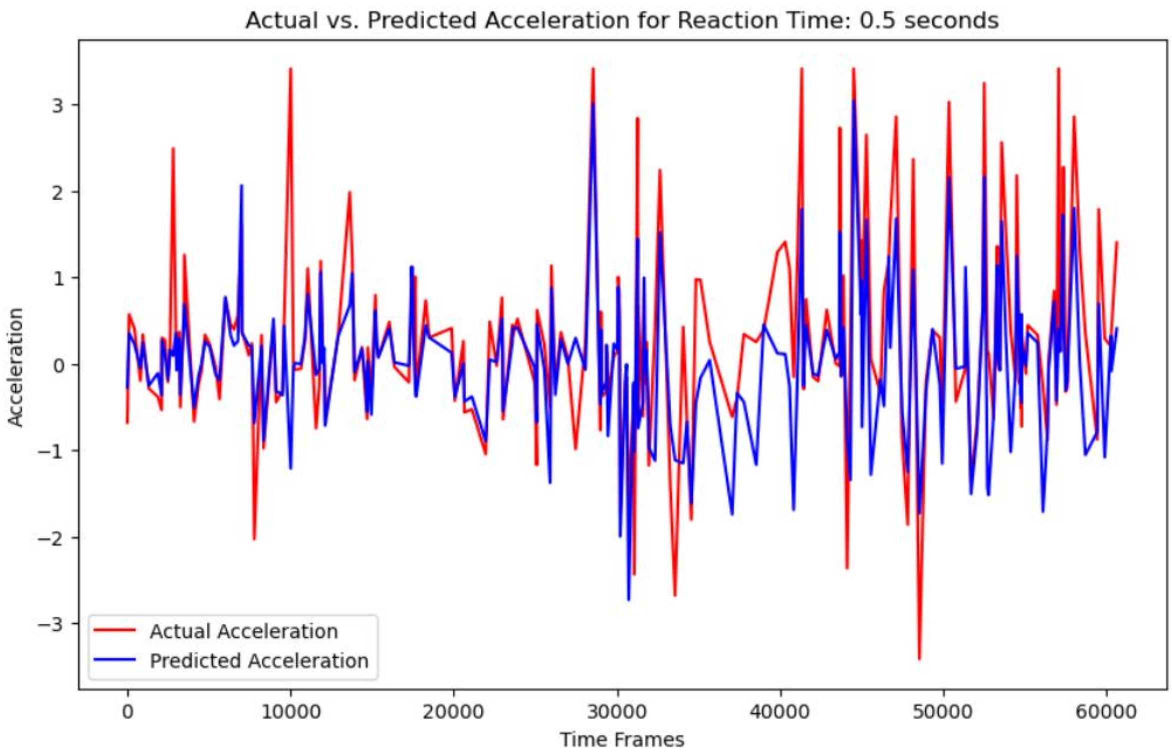

Figure 10. RF Actual vs Predicted Acceleration for Reaction time - 0.5second

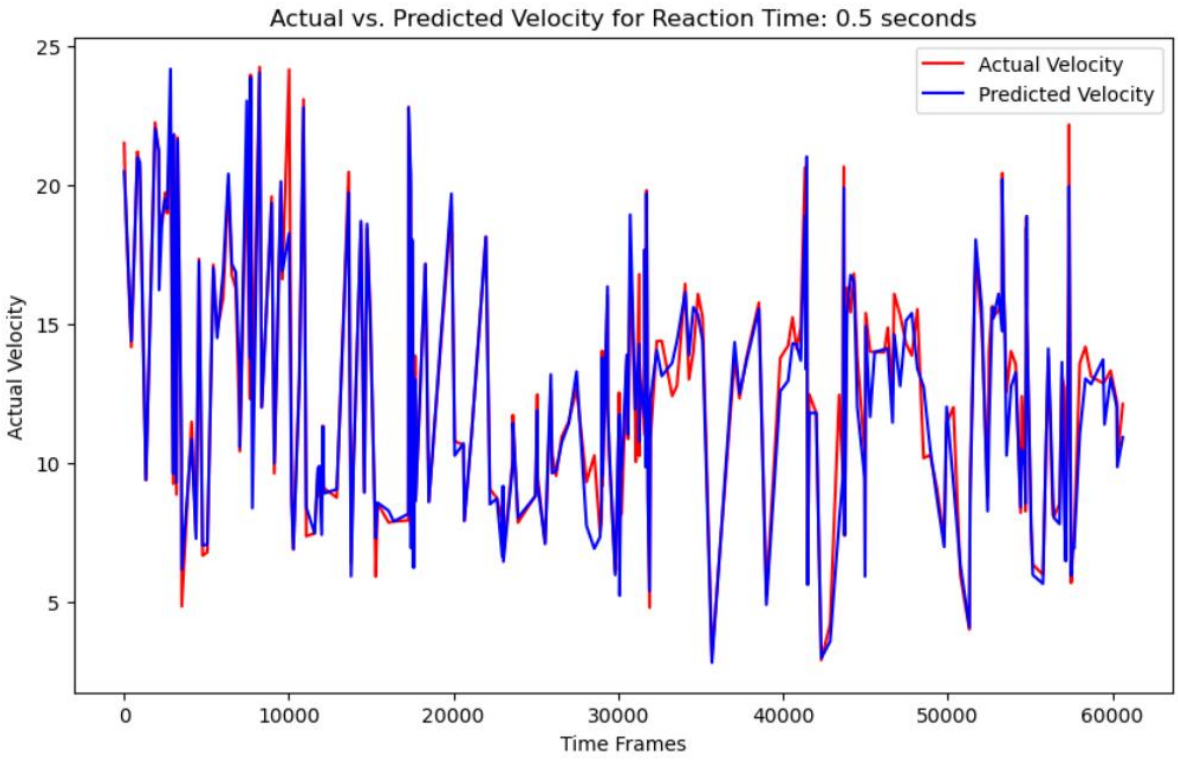

Figure 11. RF Actual vs Predicted Velocity for Reaction time - 0.5second

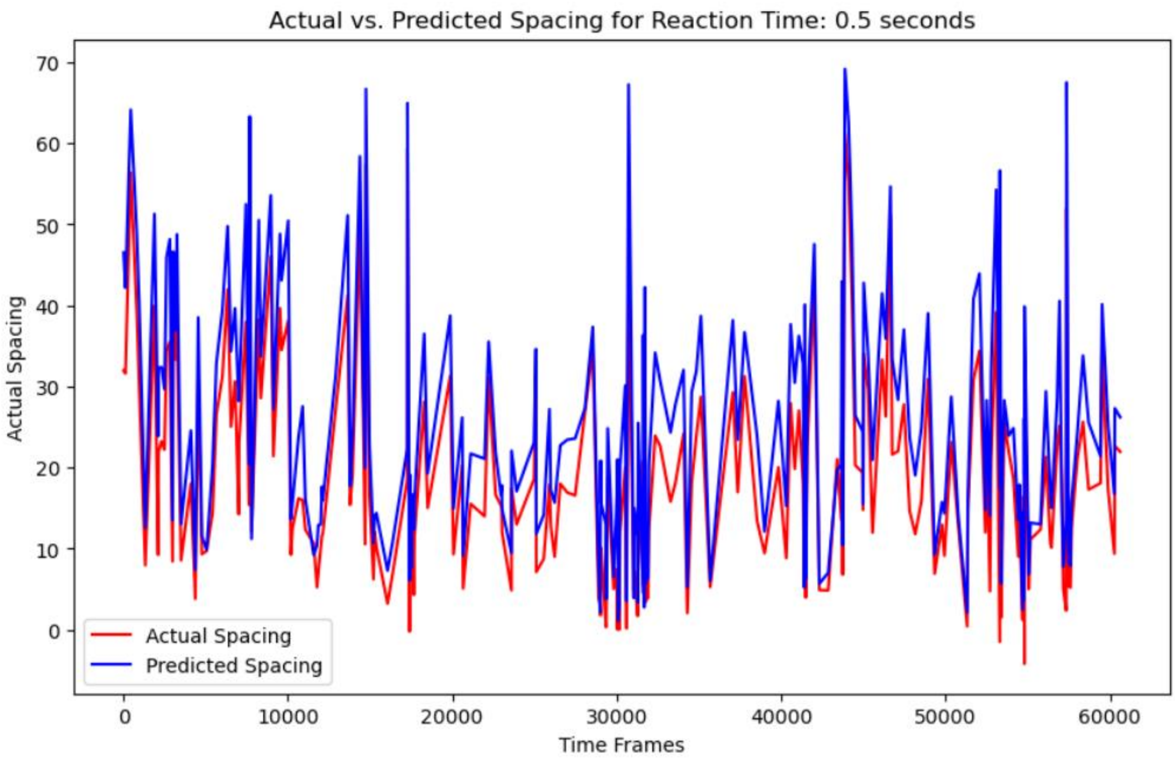

Figure 12. RF Actual vs Predicted inter-vehicle distance for Reaction time - 0.5second

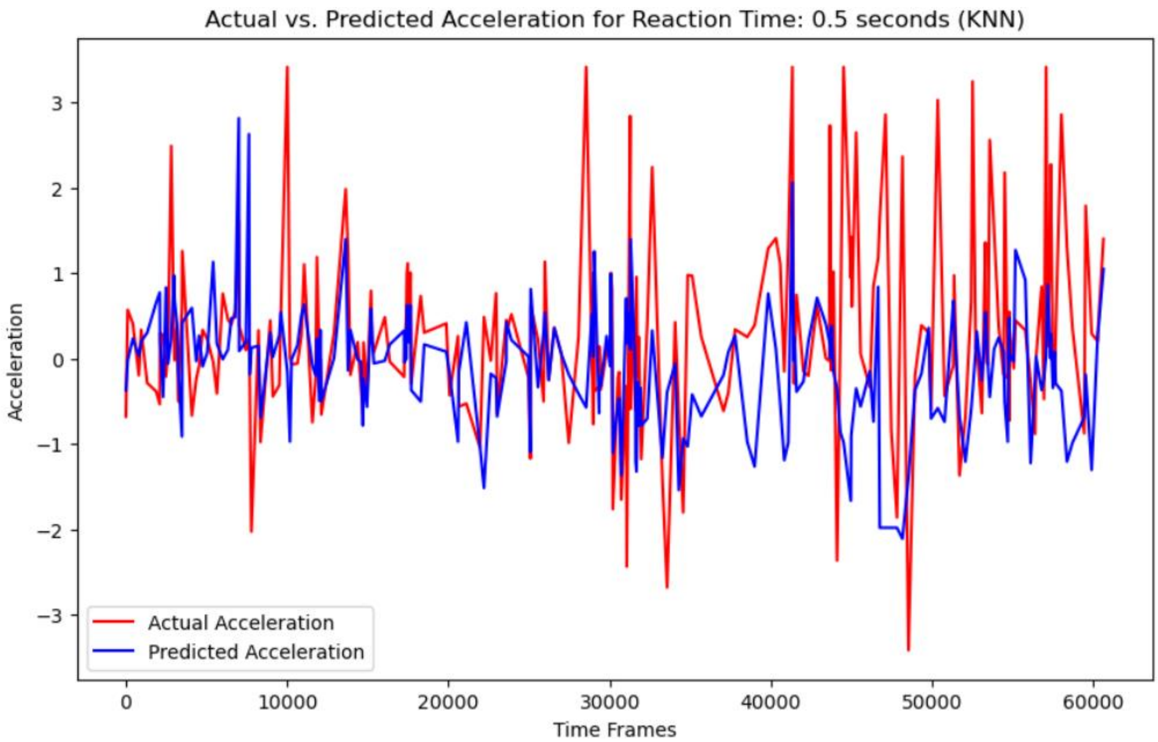

Figure 13. KNN Actual vs Predicted Acceleration for Reaction time - 0.5second

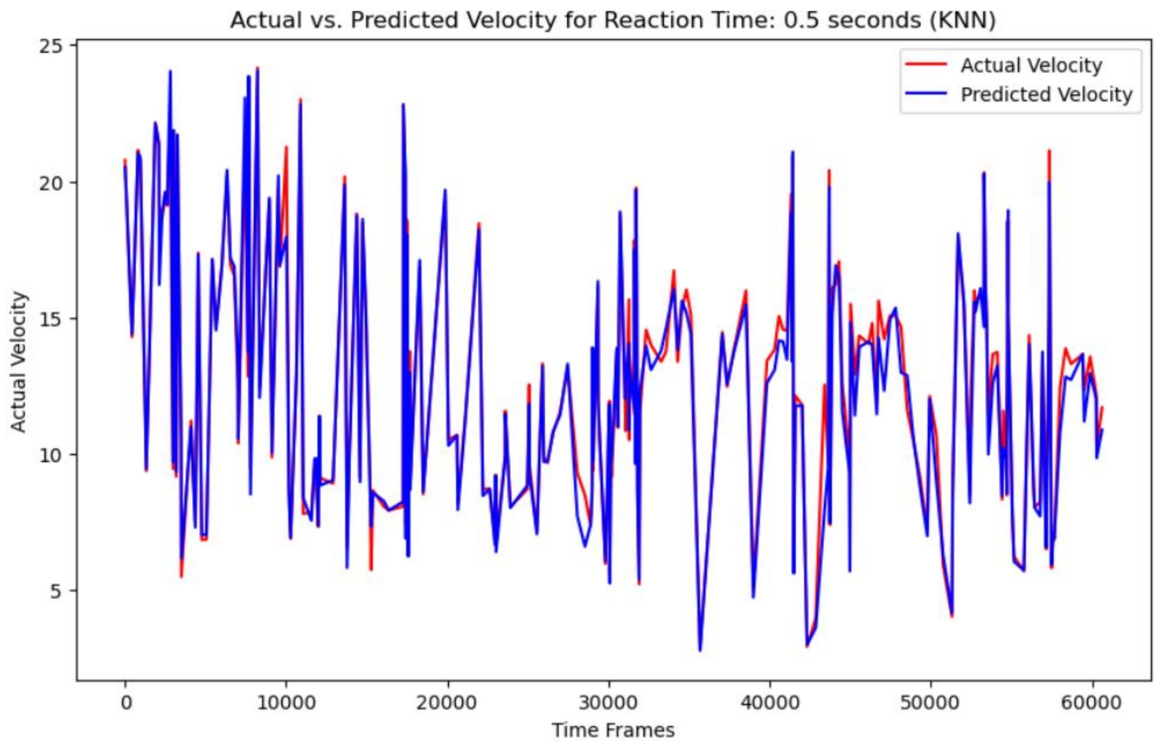

Figure 14. KNN Actual vs Predicted Velocity for Reaction time - 0.5second

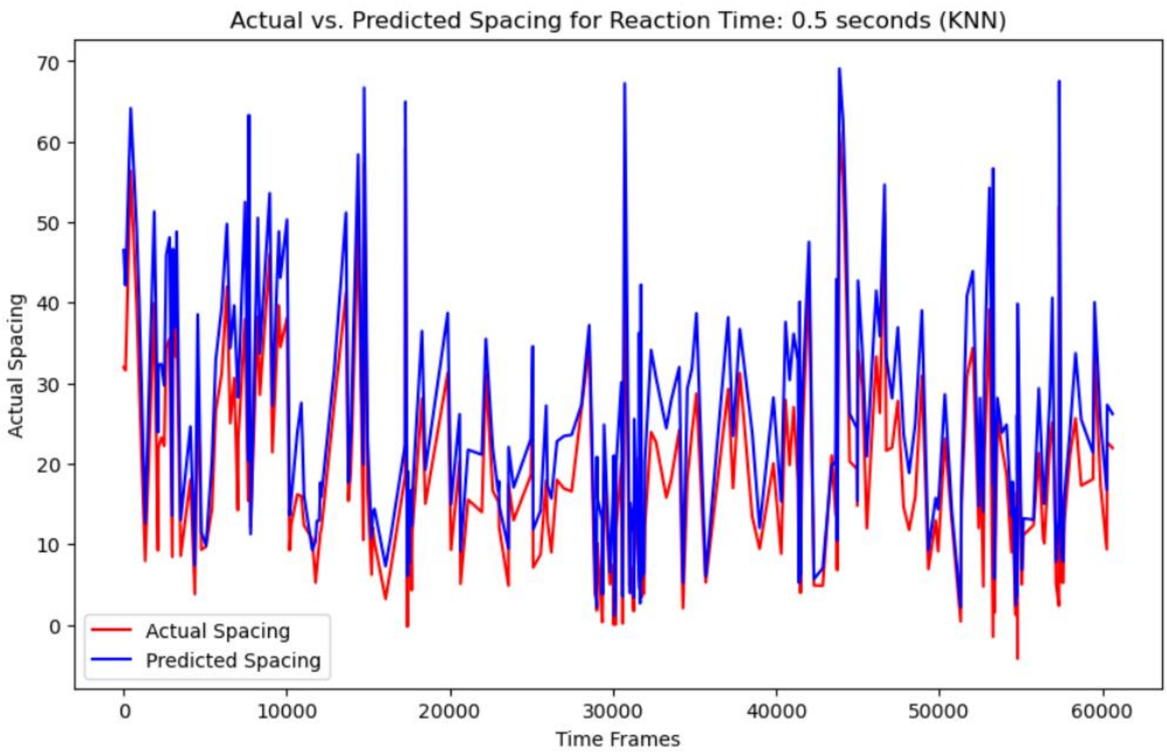

Figure 15. KNN Actual vs Predicted inter-vehicle distance for Reaction time - 0.5second

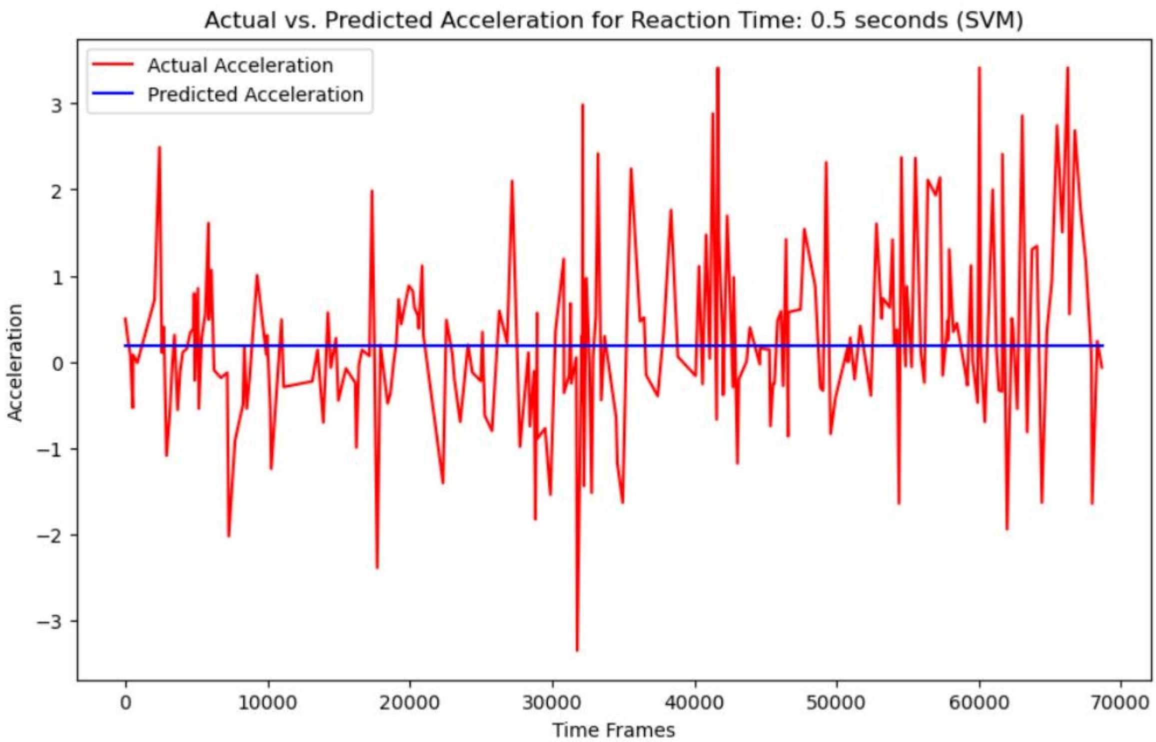

Figure 16. SVM Actual vs Predicted Acceleration for Reaction time - 0.5second

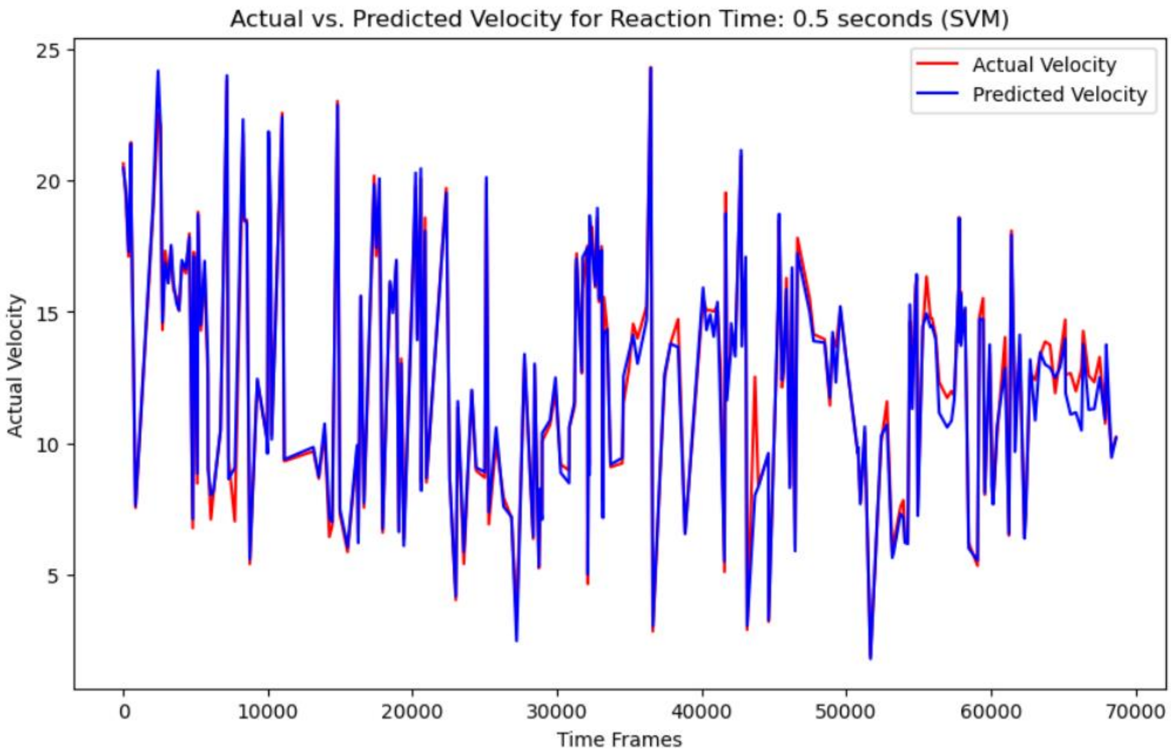

Figure 17. SVM Actual vs Predicted Velocity for Reaction time - 0.5second

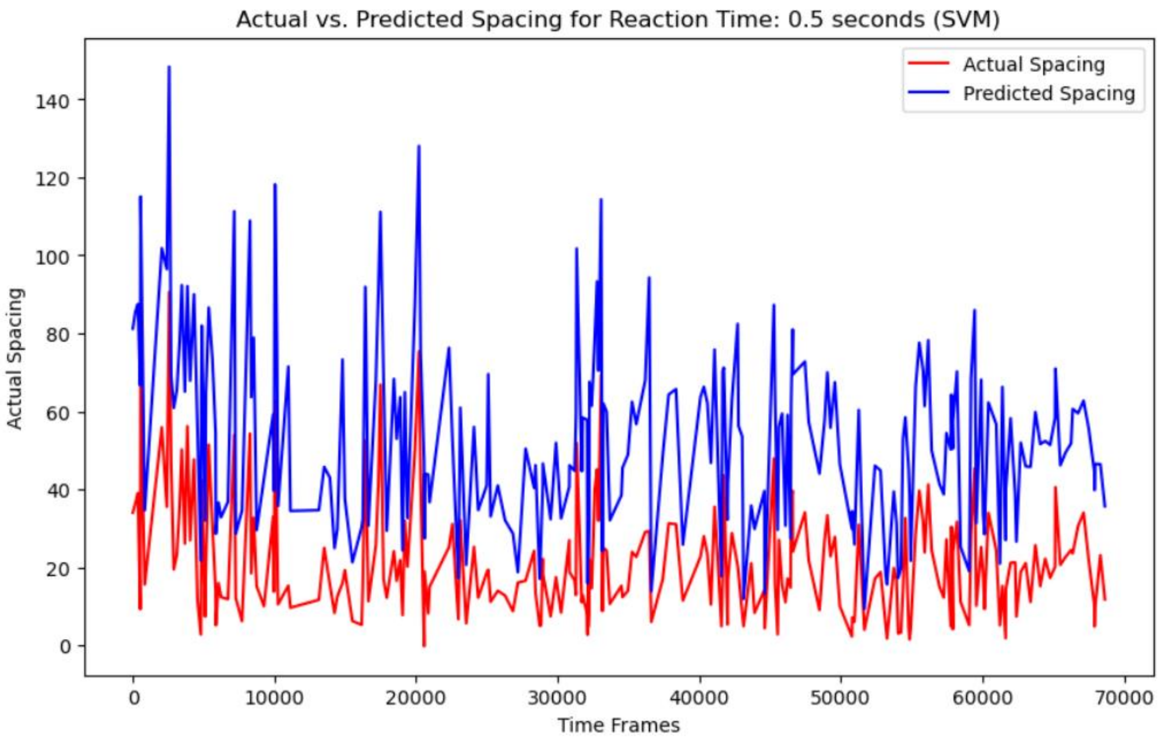

Figure 18. SVM Actual vs Predicted inter-vehicle distance for Reaction time - 0.5second

Table 1. Efficiency comparison between models

MODEL NAME	KNN	RF	SVM
Time to fit	35 s	180s	2hr
Split of data(train/test)	80/20	80/20	80/20
Loading the model	0.2 s	5 s	15s
Train/Test pairs	1024/257	1024/257	1024/257
Time taken(pairs reaction)	9 min	25 min	65 min
Test set completion	58 min	225 min	900 min

6. Conclusion

In conclusion, the research findings highlight the efficacy of Random Forest as the best model for predicting reaction time in the velocity and spacing task. The improved prediction accuracy, especially at 0.5-second intervals, showcases the potential of RF in time-sensitive scenarios. These results have implications for applications in fields such as traffic management, human-computer interaction, driver assistance systems, and other time-critical decision-making processes. Researchers and practitioners should consider utilizing Random Forest for similar prediction tasks to achieve more accurate and reliable results. Further studies could explore additional features and more complex datasets to validate these findings and potentially enhance predictive performance in real-world scenarios.

7. Future Work

Future research endeavors should carefully consider the selection of ensemble techniques and the integration of KNN and SVM models with other classifiers, such as Random Forest (RF), Gradient Boosting Machines (GBM), or Neural Networks. The research should investigate the impact of different combinations and weighting schemes on prediction accuracy, computational efficiency, and model interpretability.

Moreover, exploring ensemble strategies for handling imbalanced datasets or noisy data can provide valuable insights into improving the robustness of predictive models. Techniques like bagging, boosting, or stacking could be adapted to take advantage of the unique characteristics of KNN and SVM in addressing such challenges.

8. Acknowledgement

We express our heartfelt gratitude to the U.S. Department of Transportation for their dedication to collecting and sharing the NGSIM data with the public. This dataset has been instrumental in enriching our project and contributing to the advancement of research in microscopic data-driven car-following models.

Furthermore, we extend our sincere appreciation to Prof. Umair Durrani at St. Clair College for his invaluable guidance and unwavering support. His mentorship has been pivotal in transforming our project from a vision to a successful reality.

9. GitHub link

GitHub link: Car-Following-Model-Speed-Prediction

References

Geeksforgeeks.org, 2023. Introduction to Support Vector Machines (SVM). Available at: Introduction to Support Vector Machines (SVM) - GeeksforGeeks . Accessed 22nd July 2023.

IBMa, 2023. What is random forest? Available at: https://www.ibm.com/topics/random-forest

Accessed 22nd July 2023.

IBMb, 2023. What is the k-nearest neighbors' algorithm? Available at: What is the k-nearest neighbors algorithm? | IBM Accessed 22nd July 2023.

Li et al. 2022. A Decision-Making Strategy for Car Following Based on Naturalist Driving Data via Deep Reinforcement Learning**. Available at:** A Decision-Making Strategy for Car Following Based on Naturalist Driving Data via Deep Reinforcement Learning - ProQuest Accessed 12th April, 2023.

Lantian et al 2017. An improved car-following model with multiple preceding cars' velocity fluctuation feedback. Available at: An improved car-following model with multiple preceding cars' velocity fluctuation feedback - ScienceDirect (oclc.org) Accessed 12th July, 2023.

M. Montanino, V. Punzo, "Trajectory data reconstruction and simulation-based validation against macroscopic traffic patterns," Transportation Research Part B: Methodological, vol. 80, pp. 82-106, https://www.sciencedirect.com/science/article/abs/pii/S0191261515001393. Accessed 12th July, 2023.

Next Generation Simulation (NGSIM) Open Data https://datahub.transportation.gov/stories/s/Next-Generation-Simulation-NGSIM-Open-Data/i5zb-xe34/

Updated NGSIM dataset https://github.com/Shuoxuan/NGSIM_Cleaned_Dataset

Vineet D., Neel C., Rakesh S., 2022. Data-Driven Car-Following model. Available at; VineetDhamija/DataDrivenCarFollowing: 2022 Data Analytics Capstone (github.com)

Appendices

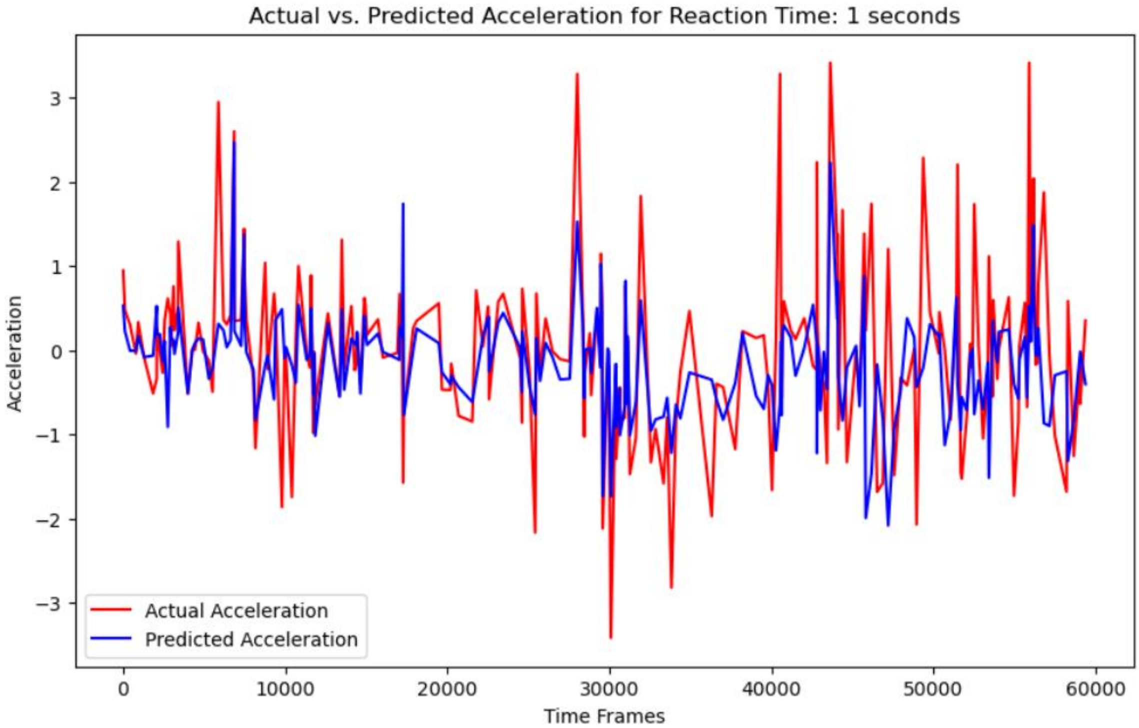

Figure 19. RF Actual vs Predicted Acceleration for Reaction time - 1second

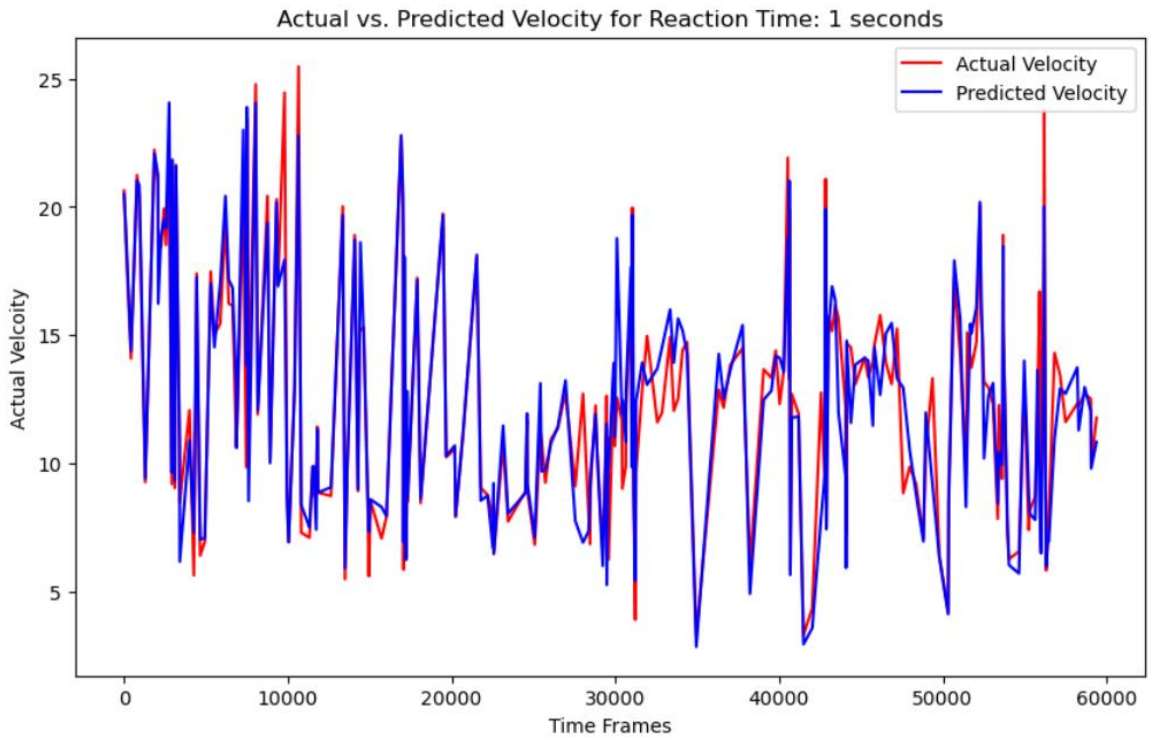

Figure 20. RF Actual vs Predicted Velocity for Reaction time - 1second

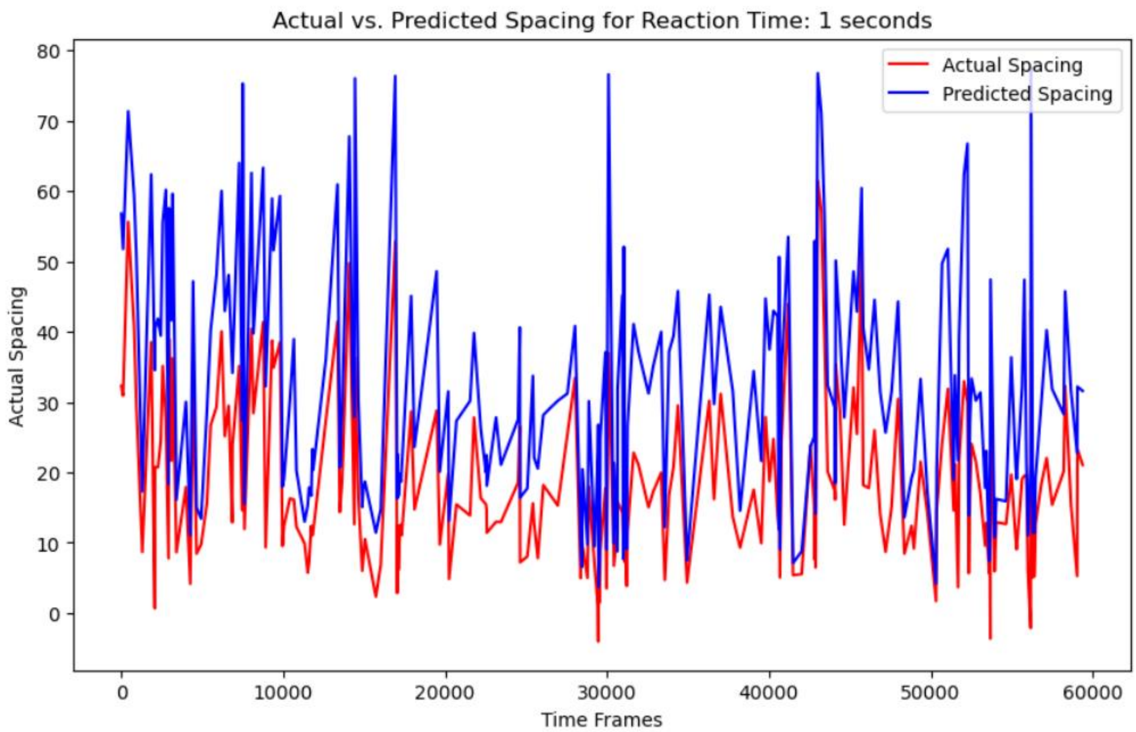

Figure 21. RF Actual vs Predicted Inter-Vehicle Distance for Reaction time - 1second

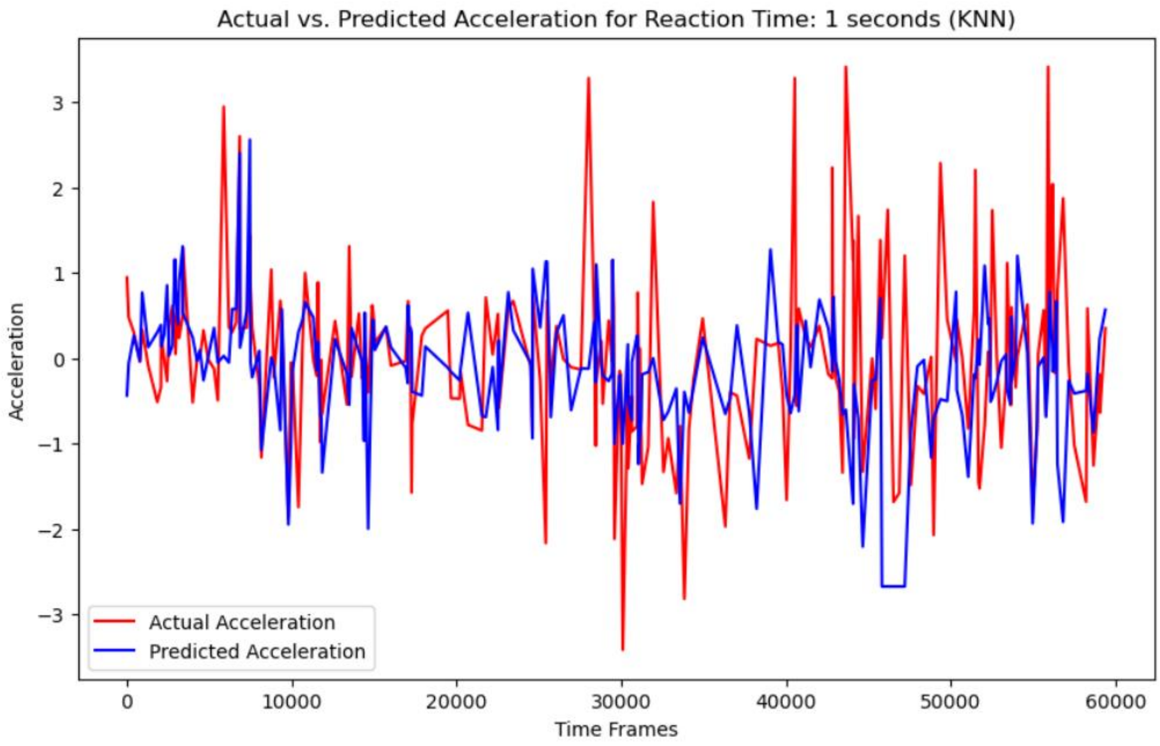

Figure 22. KNN Actual vs Predicted Acceleration for Reaction time - 1second

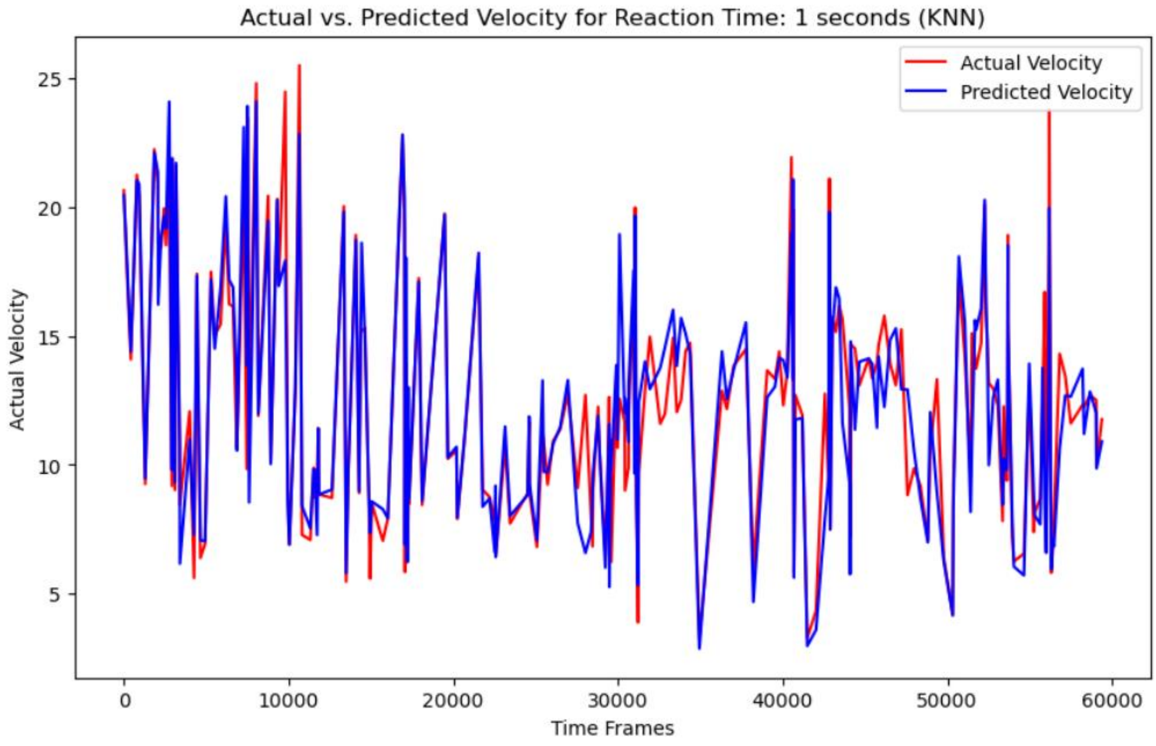

Figure 23. KNN Actual vs Predicted Velocity for Reaction time - 1second

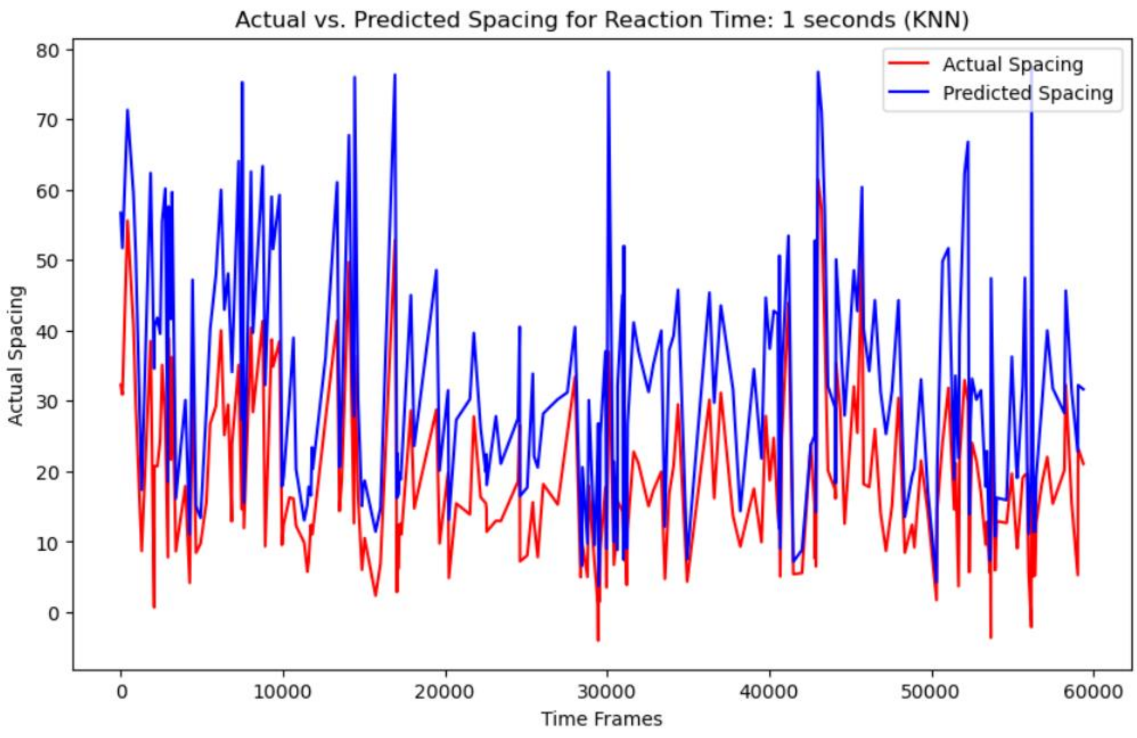

Figure 24. KNN Actual vs Predicted Inter-Vehicle Distance for Reaction time - 1second

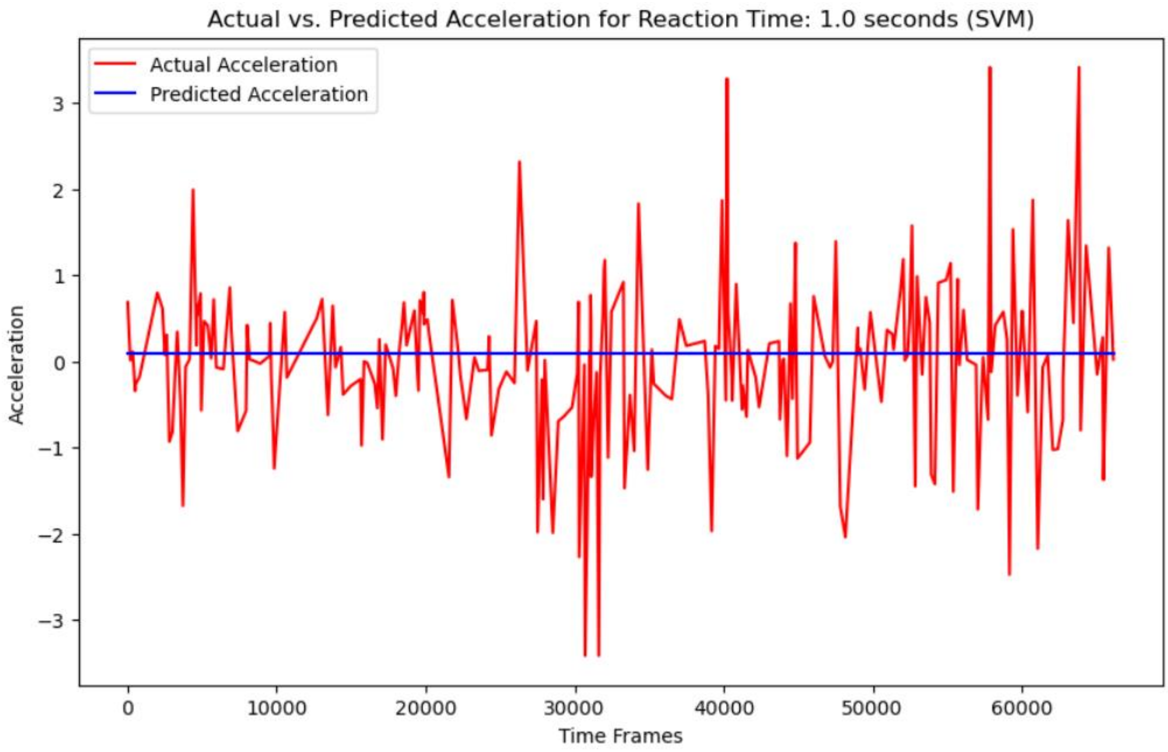

Figure 25.SVM Actual vs Predicted Acceleration for Reaction time - 1second

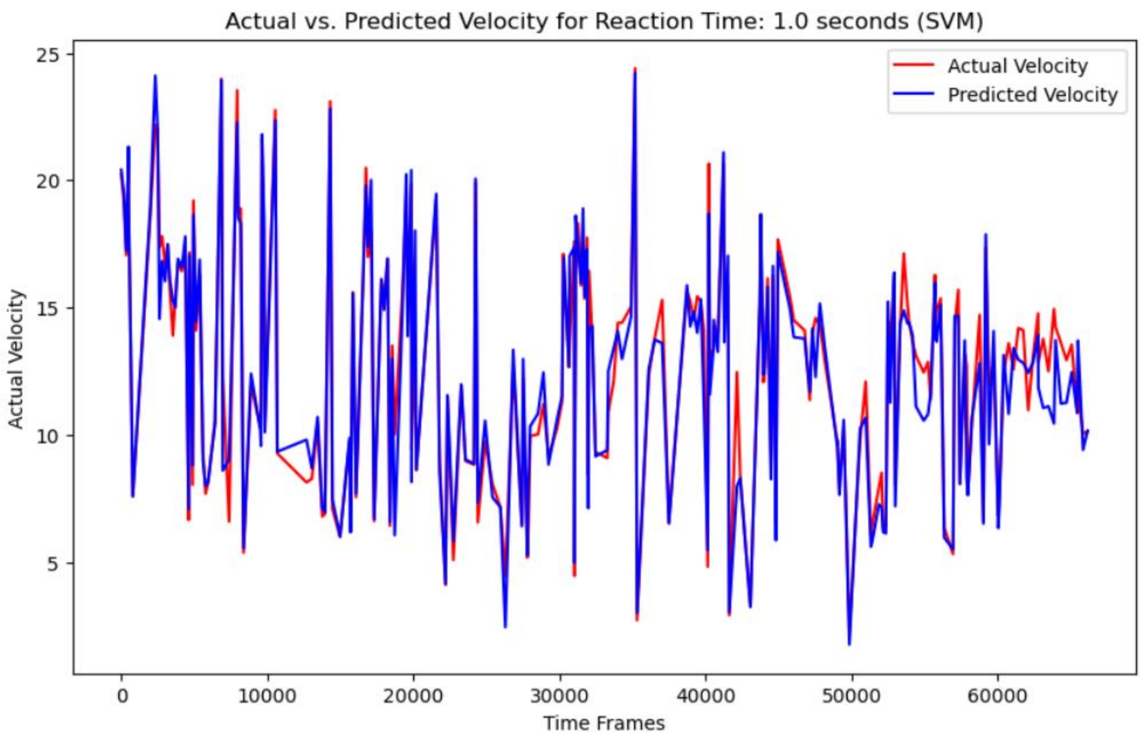

Figure 26. SVM Actual vs Predicted Velocity for Reaction time - 1second

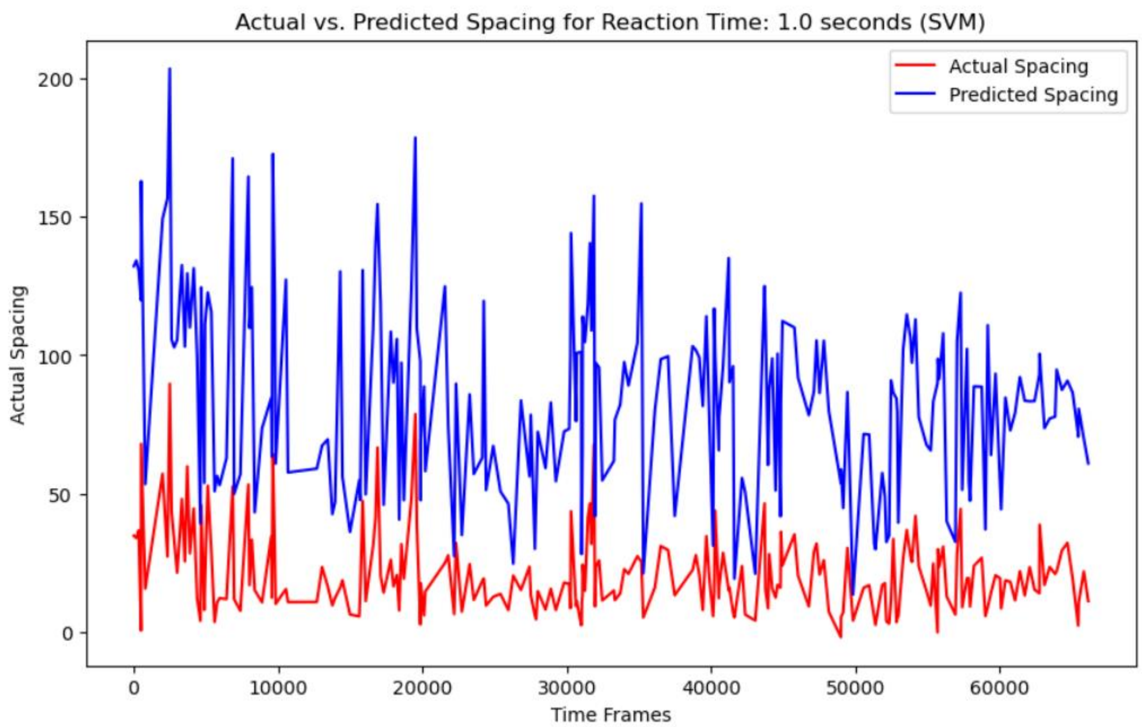

Figure 27. Actual vs Predicted Inter-Vehicle Distance for Reaction time - 1second