**Group Project**

Wong, Yu Kwan

Ngan, Tsang Kwong

Kirstin Megga Ramos

Pillai, Rani Krishna

Ailin Zhang

24W --Supervised Learning (SEC. 001)

Merlin James

April 08, 2024

**Analysis Report**

With the aim of enhancing road safety and reducing fatalities, both the police department and the individuals can benefit from a predictive AI solution by assessing the likelihood of fatal collisions. For police departments, such a tool aids in implementing effective security measures and planning road conditions in high-risk areas. Similarly, for individuals, a predictive tool allows them to evaluate the need for additional precautions based on specific factors like time, weather conditions, and neighborhood.

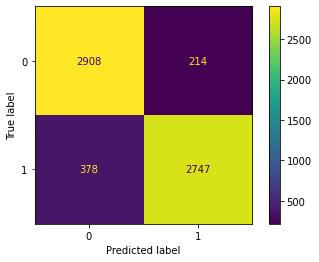
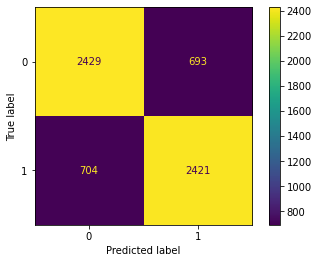
This project developed a predictive software by using machine learning and API technology. The dataset utilized in the project is from the Toronto police department over a six-year period. Section 1 in the project report involves data exploration, including data loading, statistical analysis, assessment of missing data, and graph visualization. In Section 2, the focus shifts to data modeling, transformations, feature selection, data splitting and preprocessing using pipeline class. Section 3 delves into the application of various algorithms and classifiers, such as logistic regression, SVM, Naive Bayes, voting classifier, and random forest, to build predictive models. Furthermore, grid search and randomized grid search techniques are employed for model fine-tuning. Following a comparative analysis of model scores and accuracies, the report recommends the most effective model in Section 4. Subsequently, the selected machine learning model is transformed into an analytics API using the Flask framework and the Pickle module for deployment on a local host in Section 5. The API's functionality is validated by developing a client using test data. Section 6 the conclusion was be drawn.

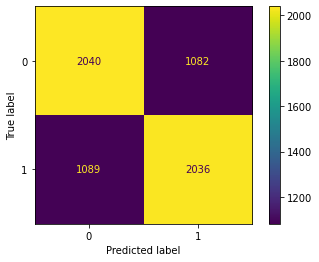
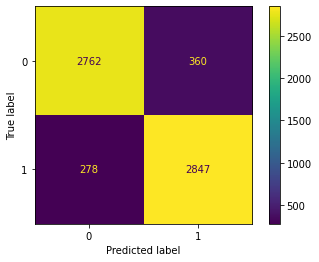
1. **Data exploration**
2. **Data modelling**
3. **Predictive model building**
4. Naive Bayes
5. logistic regression
6. SVM

In this section, firstly we employed an SVM classifier using four distinct kernels: **linear, rbf, poly, and sigmoid**. The parameter C was specifically set to 0.1 for the linear kernel, while default settings were applied to the other three kernels. Accuracy scores for both the training and test sets are summarized in Table 1 below. Additionally, Figure 1 illustrates the confusion matrix corresponding to each kernel. Notably, the rbf kernel achieved accuracy scores of 0.917 and 0.905 on the training and test sets, respectively. Similarly, the poly kernel exhibited accuracies of 0.911 and 0.898 on the training and test sets. Comparatively, the linear kernel yielded accuracies of 0.787 and 0.776, while the sigmoid kernel attained scores of 0.662 and 0.652 on the training and test sets, respectively. Based on these findings, it is evident that the **rbf and poly** kernels demonstrated superior performance in terms of accuracy.

Table 1 Comparison of accuracy on training set and test set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Kernel | Linear | rbf | poly | sigmoid |
| Accuracy on training set | 0.787 | 0.917 | 0.911 | 0.662 |
| Accuracy on testing set | 0.776 | 0.905 | 0.898 | 0.652 |

 (a) (b)



(c) (d)

Figure 1: confusion matrix: (a) linear, (b) rbf, (c) poly, (d) sigmoid

After fine-tuning the SVM models using GridSearchCV to determine the optimal parameters and best estimator, the process involved a trade-off between computational resources and model performance. The parameter grid was defined as follows: 'classifier\_\_kernel': ['poly', 'rbf'], 'classifier\_\_C': [0.1, 1, 10, 100], 'classifier\_\_gamma': [0.01, 0.1, 1.0].

A total of 120 fits were conducted, with 5-fold cross-validation for each of the 24 candidate parameter combinations. The best parameters identified were **{'classifier\_\_C': 10, 'classifier\_\_gamma': 1.0, 'classifier\_\_kernel': 'rbf'}**, and the corresponding best estimator was a Pipeline containing an SVM classifier with these parameters.

The accuracy scores obtained on both the training and testing data were noteworthy, with an accuracy score of **0.981** on the training data and **0.963** on the testing data. This indicates that the model generalizes well to unseen data, demonstrating strong predictive performance.

1. voting classifier
2. random forest

**4. Model scoring and evaluation**

**5. Deploying the model**

**6. Conclusion**