Car classification report

**Validation and training set**: At the very beginning, the cars.csv data is divided into two parts: validation\_data (20%) and training\_data (80%). While dividing the data, class imbalance is also kept in mind such that both datasets maintain the same percentage of all classes. The validation\_data is kept aside for validating the trained ML model’s efficiency.

**Data Cleaning**: Firstly, the column names were cleaned by replacing the spaces with an underscore (“\_”). After that all the string columns were mapped to a numeric data set. Since all of the data was ordinal hot encoding was not required. A simple numeric mapping on the number line with respect to the values was done.

**Visualization of Features**: Various plotting techniques were used to understand the data and their inter-relationship. Histogram plots were used to understand the nature of their distribution. Correlation matrix was used to see how related the features were to the “default” as well as amongst each other. It was observed that all features had very little correlation amongst each other. Hence it was concluded that PCA was not required for dimension reduction. After checking the variance explained by all the Eigen vectors individual, it was concluded that by removing one feature we would lose on 12% variation and hence PCA was not used.

**Feature importance and selection**: Statistical scoring techniques which only consider the relationship of the feature with the target feature were used. Chi\_squared score and Mutual Information score were used to rank the features. Deliberately, 5 new random features which were synthesised randomly (having a Gaussian distribution) were introduced. It was observed that all the features were scored more than random features, and hence all were selected for training the ML model. A wrapper based feature selection algo RBFS was used to rank the features. All the techniques were consistent in their ranking : Safety being ranked the highest and no of doors being ranked the lowest.

**Comparison and selection of Machine Learning model**: Since the data was unbalanced, precision was used as a scoring metric for the ML models. The data was divided into k-folds (stratified to take care of unbalanced dataset) and various models were tested on the same. A box plot of all the model metric scores was used to compare which model to choose. It was observed that most of the models performed very poorly. Later Random Over-Sampling was used to balance the dataset. Oversampling was used instead of under-sampling because the dataset was very small. The model performances were drastically improved with Random Forest and RBF SVM having the best performance. Since Random Forest avoids the problem of over-fitting as it builds multiple decision trees over different samples of data (Bagging), it was chosen as the final model. After finalizing the model, it’s hyper parameters were tuned using grid search. The tuned model was then trained and saved for later use on validation data.

**Machine Learning model validation**: The validation data was pre-processed in the same way as the training data. Random Forest classifier had an in-sample precision score of 0.971 and the precision score on validation data was 0.969. The confusion matrix clearly showed that Random Forest was able to predict the minority classes very well. Since our out sample validation score is equally good, we can conclude that the ML model is good at predicting in production.