Journal of Sustainable Development of Transport and Logistics
journal home page: https://jsdtl.sciview.net

Ampadu, V.-M. K., Haq, M. T., & Ksaibati, K. (2022). An assessment of machine learning and data balancing techniques for evaluating downgrade truck crash severity prediction in Wyoming. *Journal of Sustainable Development of Transport and Logistics, 7*(2), 6-24. doi:10.14254/jsdtl.2022.7-2.1.

ISSN 2520-2979

An assessment of machine learning and data balancing techniques for evaluating downgrade truck crash severity prediction in Wyoming

Vincent-Michael Kwesi Ampadu *, Muhammad Tahmidul Haq **, Khaled Ksaibati ***

* *Department of Civil & Architectural Engineering, University of Wyoming,*
1000 E University Avenue, Laramie, WY 82071, USA
Graduate Research Assistant
Tel: 307-761-1910
vampadu@uwyo.edu
** *Wyoming Technology Transfer Center, University of Wyoming,*
1000 E. University Ave., Rm 3029, Laramie, WY 82071, USA
Ph.D., P.E., Postdoctoral Research Associate
Tel: 307-761-8078
mhaq@uwyo.edu
*** *Wyoming Technology Transfer Center, Department of Civil & Architectural Engineering, University of Wyoming,*
1000 E University Avenue, Laramie, WY 82071, USA
Ph.D., P.E., Director
Tel: 307-766-6230; Fax: 307-766-6784
khaled@uwyo.edu

Article history:

Received: August 31, 2022
1st Revision: October 29, 2022
Accepted: November 12, 2022

DOI:
10.14254/jsdtl.2022.7-2.1

Abstract: This study involved the investigation of various machine learning methods, including four classification tree-based ML models, namely the Adaptive Boosting tree, Random Forest, Gradient Boost Decision Tree, Extreme Gradient Boosting tree, and three non-tree-based ML models, namely Support Vector Machines, Multi-layer Perceptron and k-Nearest Neighbors for predicting the level of severity of large truck crashes on Wyoming road networks. The accuracy of these seven methods was then compared. The Final ROC AUC score for the optimized random forest model is 95.296 %. The next highest performing model was the k-NN with 92.780 %, M.L.P. with 87.817 %, XGBoost with 86.542 %, Gradboost with 74.824 %, SVM with 72.648 % and AdaBoost with 67.232 %. Based on the analysis, the top 10 predictors of severity were obtained from the feature importance plot. These may be classified into whether safety equipment was used, whether airbags were deployed, the gender of the driver and whether alcohol was involved.

Corresponding author: *Vincent-Michael Kwesi Ampadu*
E-mail: vampadu@uwyo.edu

Keywords: crash severity, performance, extreme gradient boosting tree, adaptive boosting tree, random forest, gradient boost decision tree, adaptive synthetic algorithm

1. Introduction

Road crashes often result in injuries, fatalities, and property damage, which imposes a huge burden on the economy of most nations. The incidence of crashes is exceedingly high, especially within mountainous regions. This is primarily due to the difficult road geometry and steep downgrade lengths. Both crash frequency and severity remain very high when crashes do occur. For instance, in the U.S., there were 499,000 police-reported crashes involving large trucks in 2018. Out of this total number, 107,000 (21%) were injury crashes, and 4,415 (1%) were fatal crashes (FMCSA, n.a.). Moreover, yearly, vehicle crashes sum up to an estimated $230 billion with respect to injury, property damage and mortality, which represents 2% of the gross domestic product, G.D.P. (Wilson, 2004).

The challenging mountainous downgrades of Wyoming causes it to rank very highly as one of the locales within the U.S. with the most significant risk of truck crashes. From a national perspective, Wyoming has significantly higher fatality rates for crashes taken altogether (24.7 deaths per 100,000 people) in addition to truck-related crash rates (1.82 %) per annum (Weber & Murray, 2014). These high rates are primarily caused by heavy truck traffic on the interstate and mountainous highways of Wyoming. These are, in turn, caused by the oil drilling and coal mining activities in the state (CDC, n.a.). Combined downgrades consisting of steep downgrades and curves are a close second. Two examples of such extremes of grade steepness in Wyoming are Teton Pass, west of Jackson, and Highway 14A, between Lovell and Burgess Junction, which are both a solid, consistent 10% downgrade.

Thirdly, such high elevations characterizing the mountainous regions of Wyoming are commonly associated with unfavorable weather conditions. About 43% of the crashes recorded were a product of inexperienced drivers and unfamiliarity with the downgrades. Moreover, failing to downshift before the grade and excessive speeding accounted for 82% of identified crashes (Lill, 1977).

Methodologically, a wide range of modelling techniques has been utilized in crash severity prediction. These are typically traditional regression models. The problem with regression models, however, is that most are based on specific assumptions that pose difficulties in detecting and interpreting interactions among independent variables (Su et al., 2008). One such example of these studies involves investigating the contributing geometric factors of truck crashes on downgrades after estimating three crash prediction negative binomial models by Milhan et al., 2020. For this reason, several researchers have adopted Machine Learning (ML methods) for crash prediction; specifically, classification tree-based Machine Learning (ML), Support Vector Machine (SVM), k-Nearest Neighbour (k-NN) and the Artificial Neural Network (ANN) methods. However, many studies do not exist to compare the performance of different ML-based models.

This study was therefore performed to fill this research gap by analyzing the performance of seven machine learning models to evaluate downgrade truck crash severity prediction in Wyoming.

2. Literature review

Machine learning and data-driven techniques are emerging as significant in several areas, including smart spam classifiers for the protection of email via learning from large quantities of spam data and user responses, advertising systems learning to match suitable ads with the proper context, fraud detection systems and anomaly event detection systems in experimental physics. These successful applications are driven by two essential factors: the usage of effective statistical models capable of capturing complex data dependencies as well as scalable learning systems capable of learning the model of interest from large datasets (Li, Abdel-Aty & Yuan, 2020).

2.1. Crash severity prediction models

Generally speaking, two methods exist for crash risk prediction: statistical and machine learning. Statistical methods include conditional logit models, log-linear models and logistic regressions. These

models, however, are generally built on matched-case control data and founded on strong assumptions (Abadi, 2016; Shi & Abdel-Aty, 2015; Xu, Wang, Chen & Li, 2015. Due to these limitations, machine learning models such as Support Vector Machine (SVM) (Yu & Abdel-Aty, 2013), Random Forest (Guo et al., 2015) etc., have become more popular. Recent advances in deep learning have also been aimed at solving transportation problems. For instance, Ma et al. (2017) developed a convolutional neural network (CNN) to predict traffic speed in Beijing. This process consists of the conversion of spatial and temporal traffic dynamics into images describing the time and space relations of traffic flow. Results indicate that CNN exhibited higher performance as compared to Random Forest and k nearest neighbor.

A wide range of ML modelling techniques has been utilized for crash severity analysis. These typically fall into the broad categories of classification tree-based models, support vector machine models and neural networks.

In recent times, the classification tree-based ML methods have principally been used to identify contributing factors and predict crash risk. Chang and Chien (2013) used the classification and regression tree (CART) method to evaluate the impacts of drivers and vehicle-related factors on the injury severity of large truck crashes (Chang & Chien, 2013). In 2014, Yu et al. developed crash severity analysis models by selecting the most significant variables associated with severe crash occurrence by using the random forest (R.F.) method. Following from this, a regression model was used to perform a crash severity analysis.

In making predictions with regard to traffic crash severity, Iranitalab et al. (2017) compared the performance of four statistical and machine learning methods, including Nearest Neighbor Classification (N.N.C.), Multinomial Logit (M.N.L.), Random Forests (R.F.) and Support Vector Machines (SVM). The study also investigated the effects of data clustering methods, including Latent Class Clustering (L.C.C.) and K-means clustering (K.C.). The output from this study suggested that clustering methods may improve prediction performance under certain conditions. Tang et al. (2019) suggested a two-layered stacking framework to predict crash injury severity. The first layer integrated the merits of the three base classification methods; Adaboost, G.B.D.T. and R.F. and the second layer completed the classification of crash injury severity based on a logistic regression model (Tang, Liang, Han, Li & Huang, 2019). Schlögl et al. (2019) compared various statistical learning tools to identify traffic accident contributing factors. A set of statistical learning techniques, including all four types of logistic regression, tree-based ensemble methods, the B.R.N.N. and the Pegasos SVM were compared based on their predictive performance. Results indicated the satisfactory performance of the tree-based methods since they displayed accuracies from 75% to 90%, whereas exhibiting sensitivities from 30% to 50%.

Jinhong Li et al. (2020) employed the tree-based machine learning (ML) technique to identify and investigate crash severity factors to develop the appropriate countermeasures to such crashes. Techniques such as adaptive boosting (AdaBoost), gradient boost decision tree (G.B.D.T.) and random forest (R.F.), were used to analyze these factors. Following this, a baseline model in the form of a mixed logit model was developed to compare with the factors identified by the ML model. The analysis was conducted on crash data obtained from the Texas Crash Records Information Systems (C.R.I.S.) from 2011 to 2015. This study concluded that the G.B.D.T. model outperformed other ML techniques by way of prediction accuracy and its capacity to identify more significant contributing factors identified by the mixed logit model simultaneously. Besides, the G.B.D.T. method competently identifies both categorical and numerical factors in addition to the directions and magnitudes of the impacts of these identified factors. Of all the identified factors, driving under the influence of alcohol, drugs, and fatigue are the most essential contributing factors to large truck crash severity. In addition, the existence of medians and curbs, as well as lanes and shoulders with adequate width, can prevent severe large truck crashes.

Of the numerous machine learning techniques used in practice, gradient tree boosting is the one technique that dominates in several applications. Tree boosting has been identified as a technique capable of generating state-of-the-art results on several standard classification benchmarks. A variant of tree boosting for ranking, LambdaMART achieves state-of-the-art output for ranking problems. Besides being a stand-alone predictor, it is also integrated into real-world production pipelines for ad click-through rate prediction (Li et al., 2022). Finally, XGBoost is the go-to choice of ensemble method widely used in many machine learning and data mining challenges, inclusive of which is the Netflix prize and the numerous challenges hosted by Kaggle - the machine learning competition site. For instance, of the 29 challenge-winning solutions Kaggle published on their blog in 2015, 17 used XGBoost to train the model, whereas the other majority combined XGBoost with neural network ensembles. Of these

solutions, eight uniquely used XGBoost to train the model, while most others combined XGBoost with neural networks in ensembles. The second most popular method, deep neural networks, was used in 11 solutions as a means of comparison. The success of this system was also witnessed in the KDDCup 2015 in which every winning team used XGBoost in the top 10. Furthermore, the winning teams reported that ensemble methods outperformed a well-configured XGBoost by only a marginal amount. The most significant factor behind the success of XGBoost is its scalability under all scenarios. This system runs ten times faster than existing popular solutions on a single machine and scales to billions of examples in memory-limited or distributed settings (Burez & Van den Poel, 2008).

As a segue back to truck crashes, according to Jing Li et al. (2020), secondary incidents are more prone to severe injuries and fatalities vis-a-vis normal incidents. Secondary incidents occur within the "influence area" of a primary incident. Limited efforts have been made to unveil the factors affecting the severity of secondary incidents. Data related to incidents occurring on Interstate 5 in California in the course of five years was collected to fill this gap. Detailed dynamic traffic flow conditions, geometric characteristics and weather conditions were obtained. To begin with, a random forest (R.F.) feature selection approach was adopted. Following this, Support Vector Machine (SVM) models were developed to investigate the effects of contributing factors. The determination was made that the SVM model has a high capacity for solving classification problems with limited datasets and was, therefore, highly suited for this study. The SVM model is limited significantly by the fact that it cannot identify the impacts of explanatory variables on the dependent variable (Li et al., 2022).

2.2. Data balancing techniques in crash severity prediction modeling

Researchers across a variety of disciplines since the 1980s have attempted severally to answer the question; "How does class imbalance affect the predictive capability of asymptotic classification algorithms such as Maximum Likelihood logistic regression (M.L.L.R.)?" (Burez & Van den Poel, 2008; Cosslett, 1981; García, Mollineda & Sánchez, 2008; Gu, Cai, Zhu & Huang, 2008; Liu, Wu & Zhou, 2008; Seiffert, Khoshgoftaar & Van Hulse, 2009; Sun, Wong & Kamel, 2009; Williams, Myers & Silvious, 2009). Studies such as these have successfully minimized or removed class imbalances using basic sampling methods. Under and over-sampling are two common methods required to minimize class imbalance. Over-sampling duplicates minority-class events, whereas under-sampling eliminates majority-class events.

A relatively recently famous oversampling technique is the Synthetic Minority Over Sampling Technique (SMOTE) - defined in the study by Chawla et al. (2002). Taking each minority class sample, SMOTE creates new instances of the same class using k-nearest neighbors within a bootstrapping procedure. SMOTE is also potentially useful for handling both continuous and categorical features. The required technique is referred to as SMOTE-NC when using categorical input factors. Conversely, undersampling techniques have to do with techniques/strategies required to balance datasets through a reduction in the majority class number of samples. This so-called R.U.M.C. technique is detailed in the study initiated by Japkowicz (2020) and Batista et al. (2004). This technique also consists of a random undersampling of the majority class until the dataset is balanced. The obvious advantage of the capability to effectively handle imbalanced datasets via balancing them is correlated with resampling techniques. The disadvantages of undersampling (such as R.U.M.C.) include the fact that it eliminates potentially valuable data leading to information loss. The main disadvantage of oversampling (including SMOTE) is that by creating very similar observations of existing samples, overfitting is likely to occur. However, another disadvantage of oversampling is that it increases the number of training observations, thus increasing the learning time (Cover & Hart, 1967). Gilberto Rivera et al. (2020) studied the application of machine-learning algorithms and natural-language processing to the news provided by the RSS service. Their goal was to classify them based on whether they were about a traffic incident or otherwise in order to notify citizens where such accidents had specifically occurred. This classification process investigated the "bag-of-words" technique with five learners: Classification and Regression Trees (CART), Support Vector Machine (SVM), Random Forest, kNN and Naïve Bayes on a class imbalanced benchmark. This class imbalance is dealt with via five sampling algorithms; adaptive synthetic sampling (A.D.A.S.Y.N.), synthetic minority oversampling technique (SMOTE), random undersampling, random oversampling and borderline SMOTE. Consequently, the final classifier reached

a sensitivity of 0.86 and an area under the precision-recall curve of 0.86, which is generally acceptable considering the complexity level of assessing unstructured texts in Spanish (Rivera et al., 2020).

A significant aspect of a reliable prediction model is the selection of appropriate sample datasets for training or fitting models. Since high imbalance datasets are known to occur frequently in real-world applications, strategies need to be developed to deal with them. What typically happens, though, is that under such conditions, standard machine learning classifiers overlook the minority ones and get overwhelmed by the majority classes (Kotsiantis, Zaharakis & Pintelas, 2006). In recent times, the effects of class imbalance have drawn more and more attention. Several different solutions to the class imbalance problem were previously proposed at both the algorithmic and data levels. Solutions at the data level include several different forms of resampling to pre-process the data to obtain balanced training datasets. Solutions proposed at the algorithmic level typically include the creation of new algorithms or the modification of existing ones. Compared to the algorithmic level approach, the data level approach appears simple, offering greater promise at overcoming the class-imbalance problem (Mduma, Kalegele & Machuve, 2019). This study therefore focuses on the data level approach. Generally speaking, three resampling techniques exist to deal with imbalanced datasets; undersampling techniques, oversampling techniques, and mixed techniques. Oversampling techniques consist those that balance the number of instances between classes by increasing the number of minority classes until the dataset is balanced. On the other hand, undersampling techniques include those necessary to balance classes by reducing the number of instances from the majority class. Finally, mixed techniques include those one's that are a synthesis of oversampling and undersampling techniques. In recent years, several studies have explored crash severity prediction with data balancing techniques. For instance, Mujalli et al. (2016) employed three different data balancing techniques; oversampling, undersampling and a mixed technique to balance some traffic accident data. Following from this, different Bayes Classifier models were developed based on the imbalanced and balanced datasets. These results suggested that by making use of the balanced training datasets; particularly those created by making use of oversampling techniques, the performance of the Bayesian network classifier in classifying a traffic accident according to its severity level was enhanced. It was determined that using the balanced training datasets reduced the misclassification of A.K. level crashes (Mujalli, López & Garach, 2016).

Schlogl et al. (2019) deployed a combination of synthetic minority oversampling and maximum dissimilarity undersampling for the purposes of balancing the training dataset. Conclusions from this study supported the notion that a trade-off between sensitivity and accuracy was inherent to imbalanced classification problems. Rivera et al. (2020) evaluated five classification algorithms: Classification and Regression Tree (CART), Support Vector Machine (SVM), kNN, Random Forest and Naïve Bayes on an original class-imbalanced dataset. For this particular study, five resampling algorithms were used; random undersampling, adaptive synthetic sampling, borderline SMOTE, synthetic minority oversampling technique (SMOTE) and random oversampling. Results suggested that the imbalance between both classes after the classes were put into binary format as "traffic accident" and "not traffic accident" negatively affected the performance of both classifiers (Rivera et al., 2020). Furthermore, amongst the sampling algorithms tested, random oversampling obtained the most encouraging results. Abou Elassad et al. (2020) created a proactive decision support system to predict traffic crash events. Support Vector Machine, Random Forest, and Multilayer Perceptron Machine Learning (M.L.P.) techniques were applied for the development of crash prediction models. The study in addition compared different data balancing techniques to improve the predictive performance through three balancing techniques; undersampling, oversampling and synthetic minority over-sampling (SMOTE). The highest performances were obtained using the SMOTE technique with M.L.P. (Abou Elassad, Mousannif & Al Moatassime, 2020).

Based on the above-described literature, it is obvious that various ML-based modeling approaches have been used for the purposes of crash severity prediction. Among these models, classification tree-based ML models (Adaptive Boosting tree (AdaBoost), Extreme Gradient Boosting tree (XGBoost), Multi-layer Perceptron (M.L.P.), Support Vector Machines (SVM), Random Forest (R.F.), Gradient Boost Decision Tree (G.B.D.T.) and the k- Nearest Neighbors (kNN)) are the most popular techniques which have been used for crash severity prediction. There are however few studies that have considered the tree-based ML models as a group and compared them with other types of ML methods. This study is therefore aimed at filling the research gap and making the comparison between the predictive

performances for downgrade truck crash injury severity analysis amongst seven machine learning models.

2.3. Modeling techniques

As is described in the literature review above, this study explores seven representative machine learning methods- inclusive of which are four representative classification tree-based ML models (XGBoost, AdaBoost, RF, G.B.D.T.) and three non-tree-based ML models (SVM, kNN and M.L.P.) to develop crash severity prediction models.

2.3.1. Extreme gradient boosting (XGBoost)

The Extreme Gradient Boosting (XGBoost) is a form of the gradient boosted regression tree (Zou et al., 2018). Chen et al. 2016 discovered that a set of optimizations making use of reductions in the objective function whereas maintaining the optimal computational speed, XGBoost is a rapid and efficient tree boosting algorithm (Chen & Guestrin, 2016). The XGBoost method is an additive learning algorithm in which the first learner is fitted based on the input data, then according to the residuals of the first learner, a second learner is subsequently fitted to reduce the residual of the first weak learner. Ultimately, the prediction of the model is generated by summing the prediction of each learner. The trees are built using binary splits. The splitting point is determined by the following process: XGBoost assesses the particular feature and split-point that maximizes the gain. The maximum gain is attained where the sum of the loss from the child nodes ought to reduce the loss in the parent node. In mathematical terms, this is expressed as

$$Gain = \frac{1}{2}\left((reg_\alpha(\frac{{G_L}^2}{H_L+\lambda})\right) + (reg_\alpha\left(\frac{{G_R}^2}{H_R+\lambda}\right) - (reg_\alpha\left(\frac{G_L+G_R)^2}{H_L+H_R+\lambda}\right)),$$
$$reg_\alpha(x) = sign(x) \times max(0, |x| - \alpha) \quad (1)$$

The G terms provide the sum of the gradient of the loss function and the H terms provide the sum of the Hessian (XGBoost simply makes use of the second partial derivative) of the loss function. $G_{.L.}$ represents the sum of the gradient over the data going into the left child node, and the node $G_{.R.}$ is the sum of the gradient over the data going into the right child node. Similar occurs for $H_{.L.}$ and $H_{.R.}$ Alpha and lambda are the L_1 and L_2 regularization terms, respectively. The gain is a bit different however for each loss function.

2.3.2. Adaptive boosting (AdaBoost)

The AdaBoost algorithm is based on the fundamental idea of combining a succession of weak learners by way of a weighted majority vote for the purposes of making classifications. This data is updated repeatedly by taking the previous weak learners' mistakes into account.

The AdaBoost algorithm is given below.

Given: $(x_1, y_1) \dots \dots \dots (x_m, y_m)$ $where\ x_i \in X, y_i \in \{-1, +1\}$

Initialize: $D_1(i) = \frac{1}{m} for\ 1, \dots \dots \dots, m$

For $t = 1, \dots \dots, T$:

Train weak learner using distribution D_t

Obtain weak hypothesis $h_t: X -> \{-1, +1\}$

Aim: select h_t with low weighted error.

$h_t \varepsilon_i = Pr_i \sim D_i[h_t(x_i) \neq y_i]$

Choose $\alpha_t = \frac{1}{2}\ln(\frac{1-\varepsilon_t}{\varepsilon_t})$

Update for $I = 1, \dots \dots, m$

$D_{t+1}(i) = D_t(i)\exp(-\alpha_t y_t h_t(x_i)/Z_t$

Where Z_t is a normalization factor (chosen such that D_{t+1} will be a distribution)

Output the final hypothesis:

$$H(x) = \sum^{t} sign(\alpha_t h_t(x)) \quad (2)$$

2.3.3. Random forest (R.F.)

For the Random Forest- (R.F.) method, each tree in the ensemble is created from a sample drawn with replacement from the training set. This method combines Breiman's bagging concept and Ho's "random subspace method" for the direct purpose of building an assembly of decision trees comprising several sub-samples of the dataset (Breiman, 2001). A preplanned group of classification trees is created from the bootstrap sample and synthesized to generate a final prediction. The model combines classifiers by averaging out their probabilistic predictions in place of allowing each classifier to vote for a single class.

The input samples for RF are represented as

$$x = \{[x_{i1}, x_{i2}, x_{i3}, \dots \dots, x_{in}], y_i\}$$

where $i = 1,2,3, \dots, m$;
m represents the number of crash samples;
n on the other hand, represents the number of independent variables.

The values of the dependent variable y where y = 0, 1, or 2 correlate with different levels of crash severity. The output is the probability of a single sample belonging to different severity levels. The R.F. algorithm makes use of three fundamental calculation operations; sample set selection (bootstrap samples), decision tree generation and combination. Formally, a random forest is a predictor consisting of a collection of randomized base regression trees *{rn(x, Θm, Dn), m ≥ 1}*, where Θ1, Θ2, are i.i.d. outputs of a randomizing variable Θ. These random trees are combined to form the aggregated regression estimate $\bar{r}n(X, Dn) = E\Theta\,[rn(X, \Theta, Dn)]$, where EΘ denotes expectation with respect to the random parameter, conditionally on X and the data set Dn.

2.3.4. Gradient boosting decision tree (G.B.D.T.)

The G.B.D.T. algorithm generalizes boosting to arbitrary differentiable loss functions. The core concept is to assemble several weak models to produce a powerful ensemble. Assuming F(x) is an approximation function of the dependent variable y based on a set of independent variables x. F(x) can be expressed as $F(x) = \sum M_{m=1}(\gamma_m h_m(x))$, where γ_m and $h_m(x)$ are the basic functions, usually referred to as weak learners in the boosting context. The loss function may be defined as $L(y, F(x)) = log(1 + e^{-yF(x)})$. Due to its similarity with other boosting algorithms, G.B.D.T. builds the additive model in a greedy fashion.

2.3.5. Support vector machine (SVM)

Vapnik et al. (2013) determined that the support vector machine (SVM) is a supervised linear classifier utilized in solving classification problems by building hyperplanes.

Considering a training set represented by $\{(xi,)\}_{i=1}^{N}$, where x_i is the n-dimensional dependent variable and y_i represents the independent variable, assume $y_i = 1$ represents the independent variable and positive group whereas $y_i = -1$ represents the negative group. SVM correlates each point x_i from the input space n to the feature space H by way of the mapping function $\Phi(xi)$ and identifies a linear decision surface which separates the negative data points from the positive ones in the feature space. The hypothesis function for binary classification is

$$h(xi) = \{+1\ if\ w.x + b \geq 0, -1\ if\ w.x + b < 0\} \quad (3)$$

The datapoints above or on the hyperplane are classified as +1 and the points below are classified as class -1. In other words, the equation *(w.x + b) = 0* yields a value > =1 for positive class and < = -1 for negative class. Merging into a single equation;

y(w.x + b)(-1) = 0; y being +1 for positive class and -1 for negative class. The goal of SVM is to maximize the margin between distinct classes. And the distance between two hyperplanes *w.x + b =1* and *w.x - + b = -1 is 2/ ||w||*. So, to increase the margin, we need to minimize ||w||.

$$Max\frac{2}{||w||} \rightarrow Max\frac{1}{||w||} \rightarrow \min||w|| \rightarrow \frac{min1}{2||w||^2} \quad (4)$$

Maximizing the margin is equivalent to minimizing the expression; $J(w) = ½||w||^2 + C[1/N\Sigma^n{}_i max (0, 1- yi * (w.xi) + b))]$. A larger C gives a narrow margin and a smaller C yields a wider margin. Larger λ gives a wider margin and smaller λ yields a narrow margin. Infinitely larger *C* or smaller λ yields a hard margin SVM classifier. We can use any of the above equations for minimizing the cost function using the gradient descent approach.

Gradient descent;

$\Delta W = if\ y_i w.x_i < 1, \lambda w - y_i x_i\ else\ \lambda w + 0.$

Weights are updated via the following equation.

$$W = W - learning\ rate * (\Delta W) \quad (5)$$

2.3.6. k-Nearest neighbor (k-NN)

Cover and Hart figured out that the k-Nearest neighbor (k-NN) classifier is a standard non-parametric classifier (Cover & Hart, 1967). Instances are typically represented by some feature vectors as a point in the feature space. To classify an instance, the k-NN classifier computes the distances between the point and points in the training data set. Several different distance measures are obtained; the Euclidean distance, Manhattan distance, Jaccard Distance, Tanimoto Distance and Kullback-Leibler. Following from that, it makes assignments of the point to the class among its k nearest neighbors (where k is an integer). The nearest neighbor decision rule assigns to an unclassified sample point, the classification of the nearest of a set of previously classified points. This rule is independent of the underlying joint distribution on the sample points and their classifications and consequently, the probability of error R*- The minimum probability of error over all decision rules taking underlying probability structure into account. Therefore, for any number of categories, the probability of error of the nearest neighbor rule is bounded above by twice the Bayes probability of error. It may therefore be stated that half the classification information in an infinite sample set is contained in the nearest neighbor.

As mentioned, several distance measures are used in k-NN. The relevant equation related to the Euclidean distance is as follows;

$$d(p,q) = ((q_1 - p_1)^2 + \cdots + (q_n - p_n)^2)^{0.5} = (\sum_{i=1}^{n} (q_i - p_i)^2)^{0.5} \quad (6)$$

2.3.7. Multilayer perceptron (M.L.P.)

A Multilayer Perceptron is a type of neural network in which the mapping between inputs and outputs is non-linear.

The Multilayer Perceptron possesses both input and output layers in addition to single or multiple hidden layers with several neurons stacked together. Even though the neuron by necessity needs to have an activation function in order to impose a threshold such as ReLu or sigmoid, neurons in a Multilayer Perceptron can use any arbitrary activation function.

Multilayer Perceptrons might also be classified as feedforward algorithms because such inputs are typically combined with the initial weights in a weighted sum and subject to the activation function just as occurs in the Perceptron. The only significant difference is that each linear combination is propagated to the next layer.

Each layer then feeds the next with the results of their computation. This therefore goes the entire way through the hidden layers to the output layer. Backpropagation is therefore a learning mechanism which enables the Multilayer Perceptron to iteratively adjust weights in the network in order to minimize the cost function.

One hard requirement however exists for backpropagation to work properly- Functions combining inputs and weights in a neuron- the weighted sum and threshold functions need to be

differentiable. Such functions must be differentiable with a bounded derivative since Gradient Descent is typically the optimization function used in Multilayer Perceptron Learning.

In the course of each iteration, following the weighted sums being forwarded through all layers, the gradient of the Mean Squared Error is computed across all input and output pairs. Propagating it backwards requires the weights of the first hidden layer to be updated with the value of the gradient. In this manner, weights are propagated backwards to the originating point of the neural network.

$$\Delta_w(t) = -\frac{\varepsilon dE}{dw(t)} + \alpha\Delta_w(t-1) \quad (7)$$

where $\Delta_w(t)$ is the gradient current iteration, ε is the bias, dE is the error, $dw(t)$ is the weight vector, α is the learning rate and $\Delta_w(t-1)$ is the Gradient Previous iteration.

This process repeats itself until the gradient for each input-output pair converges, thus implying that the newly computed gradient for each input-output pair has converged. This new gradient has therefore not been modified more than a previously specified convergence threshold as compared to the previous iteration.

2.4. Data collection and processing

2.4.1. Data description

Generally, law enforcement typically follow the K.A.B.C.O. scale for the classification of severity levels of a crash: K- Fatal Injury, A- Incapacitating injury, B- Non incapacitating injury,

C- Possible injury and O- No Injury. In crash severity prediction research, the response classes are often categorized into varying levels. These include the A.K. level (A representing the incapacitating crash and K representing the fatal crash) and non- A.K. level crashes (Fiorentini & Losa, 2020). Furthermore, for the purposes of this research, the response class (severity levels of crashes) are categorized into five levels; Property Damage Only (P.D.O.), Suspected Minor Injury, Possible Injury, Suspected Serious Injury and Fatal Injury. Detailed information related to the crash dataset will be presented in the data collection and description section.

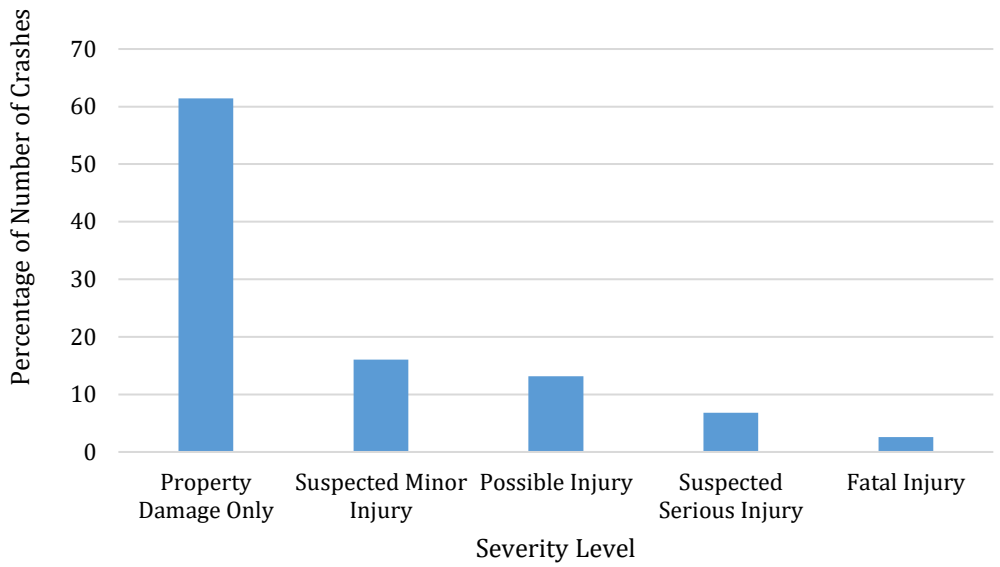
A large and comprehensive crash dataset used in this research was developed based on the crash records collated from the Wyoming Department of Transportation, (W.Y.D.O.T.). It contained crash records for the entire state of Wyoming from 2010 to 2019. Originally, each record in the database contained 120 attributes inclusive of which was information about drivers, vehicles, crash characteristics, roadway and environmental conditions. These attributes formed the candidate set of predictors. Selection of the predictors which avoid the problem of multi collinearity from the candidate set was another challenge. For each of the crash responses, the predictors were selected using a backwards stepwise selection procedure. In this approach, hypothesis tests are conducted on the full candidate set, and predictors are removed sequentially that do not have p-values below the 0.05 significance level.

2.4.2. Dependent and independent variables

Severity level of the crash was the dependent variable in this study. This variable was categorized into five levels: accidents with Property Damage Only (P.D.O.) (y = 0), Suspected Minor Injury (y=1), Possible Injury (y=2), Suspected Serious Injury (y=3) and Fatal Injury (y=4).

In the dataset as shown in Figure 1, there were 61.43 % of P.D.O. level crashes, 16.03 % of Possible Injuries, 13.14 % of Possible Injuries, 6.79 % of Suspected Serious Injuries and 2.60 % of Fatal Injuries.

Figure 1: Training dataset

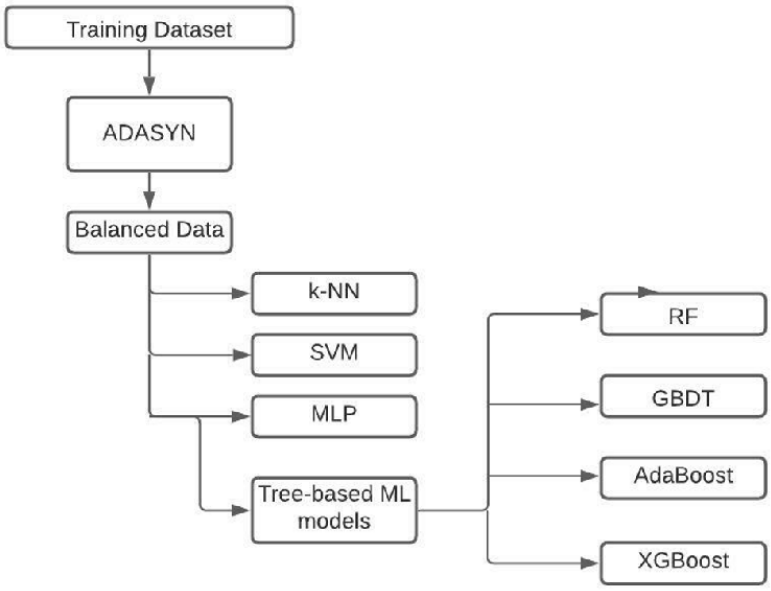


As can be observed, the data is highly imbalanced because a class distribution with an imbalance ratio exceeding 1.5 can be considered imbalanced (Fernandez et al., 2008). Out of the entire dataset comprising roughly 120 major predictors, only 15 were selected as the independent variables for this research. Each of them were associated with 51,976 crashes on downgrades from the W.Y.D.O.T. database. Potential issues with the data such as multicollinearity between variables and endogeneity were determined to not be an issue with these predictors. For each predictor or feature, there are different categories. For instance, under the Gender column there is Male, Female and "Other" and under alcohol, there is a "Yes" and a "No". Machine learning algorithms do not comprehend string data but understand numbers. A means was sought to convert the data into numerical format in order to feed it to the Machine Learning algorithm and the method used was the one-hot encoding in which every feature is deconstructed into their respective classes. For example, a row corresponding to a Male, would have genderMale represented by a 1, genderFemale by a 0 and genderOther by a 0 as well. In the same way, a row corresponding to Female, would have genderMale represented by a 0, genderFemale by a 1 and genderOther by a 0. The data is now in a form that the ML algorithm can understand (Categorical numerical form)

2.5. Design of study

This research aims to predict the severity level of crashes on Wyoming downgrades based on a comparison of different classification models. The final cleaned dataset was randomly split up into 75% and 25% with 75% representing the dedicated training dataset and the 25% representing the dedicated testing dataset. As discussed, the A.D.A.S.Y.N. library is used to balance the dataset. Figure 2 illustrates the modeling scenarios.

Figure 2: Study scenarios



2.6. Methodology

The machine learning models called to build the various algorithms were built using the scikit learn API. For the AdaBoost model, the underlisted methodology was followed;

Taking the Adaboost Classifier as an example, the AdaBoostClassifier() class was instantiated using an alias, after which the ".fit" method was called on the independent training set (X_train) and the dependent training set (y_train) and used to train the model. The ".predict" method is then called and used to make predictions on both the independent testing set (X_test) and the independent training set (X_train). Following from this, the ".predict_proba" method is called to make probabilistic predictions on the independent testing set.

Area under the curve; A.U.C. is typically used for binary classification, but since this is a multi-class classification, in order to evaluate the model on the score, the target was binarized into 5 classes. The model was then evaluated on the f1 score, A.U.C. score and the Accuracy score, using the methods from the scikitlearn metrics API for both training and testing sets. Results were presented in percentage format.

This exact same methodology was followed for Random Forest, Support vector machine, XGBoost, K nearest neighbours, Gradient boosting tree and Multi-layer perceptron. For the above-mentioned models, the following classes were instantiated respectively for each of them; RandomForestClassifier(), LinearSVC(), XGBClassifier(), KNeighborsClassifier(), GradientBoostingClassifier() and the MLPClassifier(). The only difference is the ".predict_proba_lr" method used to make probabilistic predictions on the independent testing set in the Support Vector Machine model.

After going through this process for all the models, the performance of the models for all 3 metrics were compared and ranked. The random forest came out on top across all 3 metrics. After this, the best performing model (Random Forest) is optimized by tuning hyperparameters. The 3 hyperparameters (criterion, max depth, and number of estimators) were tuned. The values of the criterion were "gini" and "entropy"; for the max depth- "None", "2" and "3", and for the number of estimators; "50", "70", "100", "120" and "150". The best performing set of hyperparameters was criterion being "gini", number of estimators being "200" and max depth being "None". A comparison of the optimized model and the baseline model was made and a 0.12 % increase in performance was determined.

A confusion matrix was then built using the sklearn metrics API. Finally, a plot of A.U.C. scores via the optimized random forest against all the other models was made.

The feature importance plot is accessed from the Random Forest Importance's API. The method is called from the random forest ".method" and used to build the feature importance plot and as a result, the longer the bar, the more important the feature is to predicting severity of the crashes whereas the

shorter the bar, the smaller its importance is to predicting severity. The various feature importance plots of all the predictors should add up to 1.

2.7. Prediction evaluation measures

Two main types of evaluation measures were used to measure the prediction performance of the models. These are the threshold-based measures and the non-threshold-based measures.

Threshold-based measures such as specificity, precision, sensitivity, and the F-measure rely on one specific threshold. Since each of these measures are decided based on one specific threshold, they do not provide a comprehensive evaluation of the model performance. This problem may be eliminated by making use of non-threshold-based measures like ROC-AUC (Fernandez et al., 2018).

The Receiver Operating Characteristic (R.O.C.) is the curve generated when the longitudinal axis represents the true positive rate, and the transversal axis represents the false positive rate for different thresholds (Rivera, Florencia, García, Ruiz & Sánchez-Solís, 2020). For an R.O.C., the area under the curve (A.U.C.) represents the degree of separability between classes. A maximum ROC-AUC value close to 1 describes a classifier with excellent performance in separating classes whereas a value close to 0.5 describes a valueless test. Since ROC-AUC is threshold-independent, it can be used to measure the overall performance of a prediction model. Furthermore, the ROC-AUC doesn't emphasize one class over the other, so it remains unbiased against the minority class (Kotsiantis, Zaharakis & Pintelas, 2006). In this study therefore, R.O.C.- A.U.C. is selected as the evaluation measure.

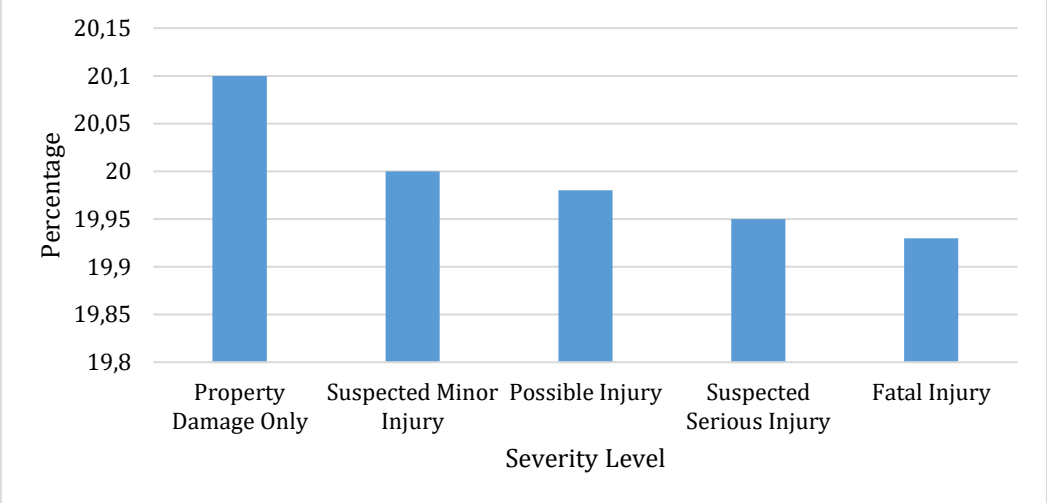
3. Results and discussion

There are three components of the data analysis. The first component requires balancing of the various classes in the target variable whereas the second evaluates the performance of the different ML models for both the training and testing datasets to determine which one ranks highest. Finally, the relative importance of the various features based on the best performing ML model is determined. All models are programmed in Python version 3.9, using scikit learn, imbalanced-learn 0.8.0, XGBoost 1.4.2, pandas 1.3.3, matplotlib 3.4.3, numpy 1.21.2.

3.1. Balancing dataset

Based on the plot indicated in Figure 1, the majority class, P.D.O. (Property Damage) has over 60 % of the total data in the severity column, with the least class (Fatal Injury) having just around 2.6%. This is a very big difference which needs to be taken care of to prevent bias when the machine learning models are trained on them. To balance the data, the A.D.A.S.Y.N. (Adaptive Synthetic) algorithm is used. Figure 3 shows the new distribution of the severity classes after balancing.

Figure 3: Training dataset after balancing using A.D.A.S.Y.N.



The classes in the target variable are now balanced with the majority class having 20.1% of the data and lowest class having 19.93%.

3.2. Evaluating performance of different ML models

The first step at this stage is to split the dataset into a 75% versus 25% split where 75% represents the size of the training set and 25% represents the size of the testing set. The various performance metrics for XGBoost, AdaBoost, Random Forest, G.B.D.T. as well as SVM, k-NN and M.L.P. are displayed in Table 1. They include Accuracy, F1 and A.U.C.

Table 1: Performance metrics for the various ML techniques

	Accuracy (%)	F1	A.U.C.
Random Forest	92.28	92.24	95.18
K Nearest Neighbours	88.46	88.41	92.78
Multilayer Perceptron	80.51	80.17	87.82
XGBoost	78.47	77.97	86.54
Gradient Boost	59.73	58.65	74.82
Support Vector Machines	56.21	53.02	72.65
Adaboost	47.57	47.44	67.23

As can be seen, for all 3 metrics, the random forest model achieved the highest score, but our main metric here is the A.U.C. score. From here on, the random forest model would be optimized to try to improve its results by tuning the hyperparameters. The three hyperparameters tuned are the criterion, number of estimators and the maximum depth. Using a maximum depth significantly reduces the performance of the Random Forest, so it will be exempted whereas focusing only on the criterion and number of estimators. The most optimized model is the one in which the criterion is specified as "gini" and number of estimators is "200". The output is illustrated in Table 2.

Table 2: Optimized performance for random forest model

Model Performance for Training Set	**%**
Accuracy:	99.28
F1 score:	99.28
AUC score:	99.55
Model Performance for Test Set	%
Accuracy:	92.47
F1 score:	92.44
AUC score:	95.29

The confusion matrix shown in Figure 4 below shows what the model predicted vrs the actual label. The diagonal represents the correct predictions. So, for example, 11 data points were predicted to be of severity type 1, which should have been 5 instead.

Figure 4: Confusion matrix illustrating true versus predicted label

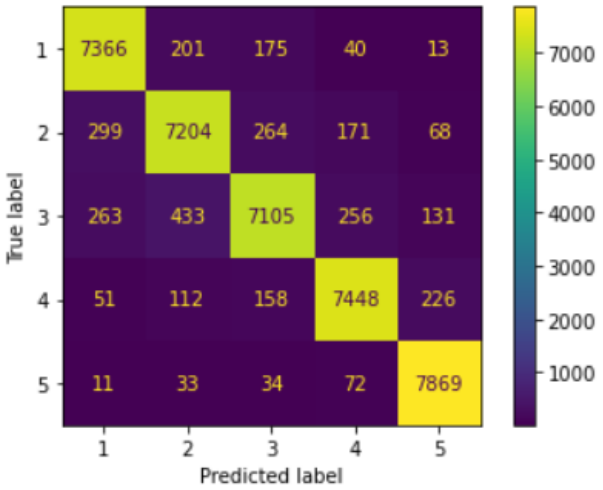


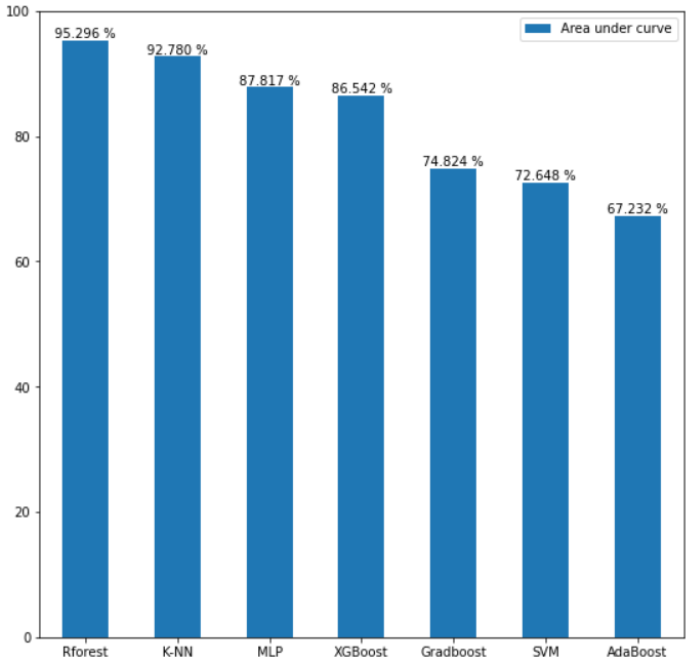
Table 3 compares the optimized random forest model to the baseline random forest model.

Table 3: Comparison of optimized and baseline random forest model

	Area Under Curve (%)
Optimized Random Forest	95.29
Baseline Random Forest	95.17

The optimized random forest model performs slightly better than the baseline, with a 0.12% increase, which is still significant as it can actually improve the prediction of the severity of a crash per certain features. Figure 5 shows a plot of the ROC AUC scores including the optimized random forest model.

Figure 5: R.O.C.- A.U.C. scores (Optimized Rforest)



Finally, to determine the relative importance of the various predictors, the feature importance plot is made. The top 10 predictors are indicated in Figure 6 below. The longer the bar, the more important that feature is to predicting the severity of vehicle crashes. The top 10 predictors of severity

based on this analysis can be summarized into four categories; whether safety equipment was used, whether air bags were deployed, the gender of the driver and whether alcohol was involved.

More specifically, all three categories of safety equipment use- Usage (Booster Seat, Child Restraint, Helmet Used, Lap Belt Only, Passive Restraint Only, Forward facing Child restraint, Rear Facing Child Restraint, Shoulder and Lap Belt, Shoulder Belt Only), Non-usage and unavailability of the data, all three categories of Airbag deployment- Air bag deployed (Deployed Combination, Deployed Front, Deployed Side and Other types of deployment), Air bag not deployed and unavailability of the data, all three categories of gender- male, female and unavailability of the data and finally, alcohol usage were determined to be essential to the prediction of the severity of vehicle crashes. The listed predictors of severity impact crashes in the following ways;

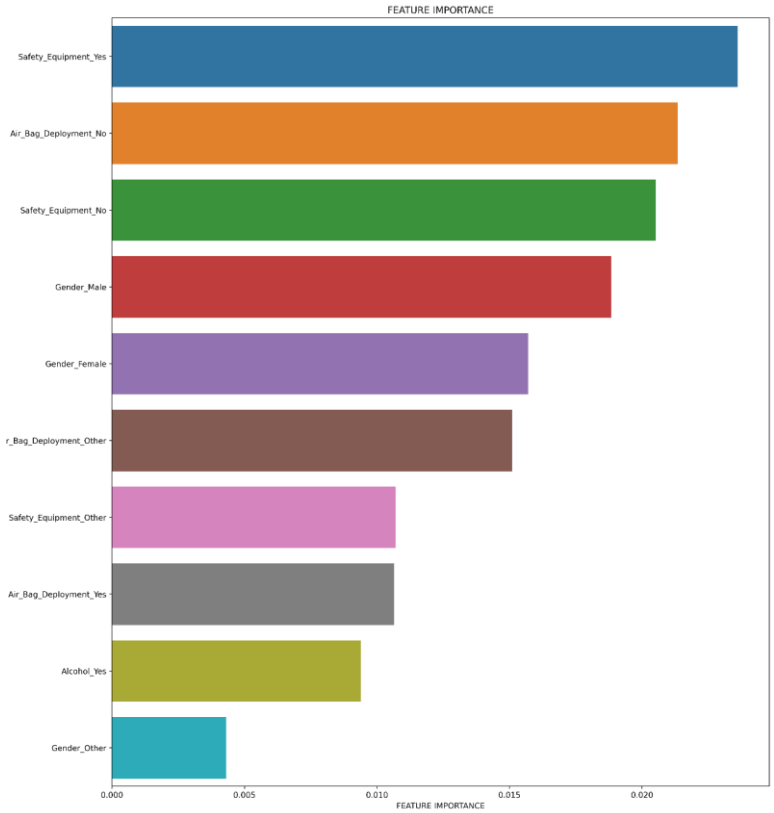
Safety Equipment: The most popular type of safety equipment is the seat belt. Seat belts have averted more vehicle fatalities due to accidents as compared to any other safety feature of cars. Prior to the invention and adoption of seat belts, the fatality rate in car accidents was concerningly high. However, in the 1960's, the United States federal government created the National Highway Traffic Safety Administration (NHTSA). This organization mandated seatbelts to be used in automobiles. In recent times, the vast majority of front and rear seat belts have both a lap belt and a shoulder harness.

Airbags: Airbags inflate when an electronic sensor signals that a collision has occurred. They also have the capacity to cushion the impact and prevent serious injury; even stopping the driver or passenger from contacting the windshield or steering wheel during the impact.

Gender: Research indicates that the impact of crashes on men and women tend to differ. Several reasons are hypothesized to account for this difference. The key reason has to do with the fact that men tend to drive at higher speeds (often close to the speed limit) on interstates and highways whereas women tend to be more cautious when driving and as such drive more closely to the speed limit. This results in more men being susceptible to accidents and accompanying injuries, property damage and fatalities. This is easily observed by comparing the feature importance plot of males to females.

Alcohol usage: Alcohol and drugs impact driving behavior by affecting the central nervous system, awareness, vision, and perception/reaction times. The net effect of such impairment on driver injuries in car crashes is a subject currently being rigorously studied. Drinking alcohol can lead to accidents (minor and serious) such as road traffic accidents, drowning, poisoning, and falls.

Figure 6: Feature importance of top 10 predictors



4. Conclusions and recommendations

The purpose of this research was to compare the performance of different classification models (XGBoost, AdaBoost, Random Forest, G.B.D.T., SVM, k- N.N. and M.L.P.) as applied to large truck crash severity prediction on Wyoming roads. The raw data is originally largely unbalanced with the majority class, P.D.O. having over 60% of the total data in the severity column, with the least class; Fatal Injury having just around 2.6%. In order to balance the data, the A.D.A.S.Y.N. (Adaptive Synthetic) algorithm is used. After balancing, the classes in the target variable are now balanced with the majority class having 20.10 % of the data and lowest class having 19.93 %.

After splitting the dataset into a 75% - 25% split between the training dataset and testing dataset, and evaluating the performance of all seven models tested, for all 3 metrics (Accuracy, F1 and A.U.C.), the random forest model achieved the higher score. The random forest model was then optimized to try to improve its results by tuning the hyperparameters. The most optimized model specified "gini" as its criterion and "200" as number of estimators. The optimized random forest model performs slightly better than the baseline, with a 0.12 % increase, which is still significant as it has the potential to improve the prediction of crash severity based on certain features. The Final ROC AUC score for this optimized random forest model is 95.296 %. The next highest performing model was the k-NN with 92.780 %, M.L.P. with 87.817 %, XGBoost with 86.542 %, Gradboost with 74.824 %, SVM with 72.648 % and AdaBoost with 67.232 %.

Finally, the top 10 predictors of severity based on this analysis were obtained from the feature importance plot and were categorized into 4 distinct groups; whether safety equipment was used, whether air bags were deployed, the gender of the driver and whether alcohol was involved. The listed predictors of severity impact crashes in the following ways:

Safety Equipment: The most popular type of safety equipment is the seat belt. Before the invention and adoption of seat belts, the fatality rate was alarmingly high in car accidents. The NHTSA, in the 1960's mandated seatbelts to be used in automobiles. Currently, most front, and rear seat belts are equipped with both a lap belt and a shoulder harness. Since then, seat belts are likely to have averted more car accident fatalities than any other safety feature of cars.

Airbags: Airbags inflate when an electronic sensor signals that a collision has occurred. They can also reduce impact and avoid serious injury; even keeping the driver or passenger from making contact with the windshield or steering wheel during the impact.

Gender: Research indicates that the impact of crashes on men and women tend to differ. Several reasons are hypothesized to account for this difference. The key reason has to do with the fact that men tend to drive at higher speeds (often close to the speed limit) on interstates and highways whereas women tend to be more cautious when driving and as such drive more closely to the speed limit. This results in more men being susceptible to accidents and accompanying injuries, property damage and fatalities. This is easily observed by comparing the feature importance plot of males to females.

Alcohol usage: Alcohol and drugs exert an influence on driving behavior by affecting the central nervous system, vision, awareness, and perception/reaction times. Alcohol consumption can therefore lead to minor and serious accidents including road traffic accidents, drowning, poisoning and falls.

Acknowledgments

The authors would like to acknowledge that this work is part of a funded project by the Wyoming Department of Transportation (W.Y.D.O.T., contract no. RS09126) and Mountain-Plains Consortium (MPC-540). All Figures, tables, and equations listed in this paper will be included in the final report at the conclusion of this study.

Author contributions

The authors confirm contribution to the paper as follows; Study conception and design: K. Ksaibati, Vincent Ampadu; Analysis and interpretation of results: Vincent Ampadu, Muhammad Haq, K. Ksaibati; Draft manuscript preparation: Vincent Ampadu, K. Ksaibati. All authors reviewed the results and approved the final version of the manuscript.

Declaration statements

No potential conflict of interest is reported by the authors.

Citation information

Ampadu, V.-M. K., Haq, M. T., & Ksaibati, K. (2021). An assessment of machine learning and data balancing techniques for evaluating downgrade truck crash severity prediction in Wyoming. *Journal of Sustainable Development of Transport and Logistics, 7*(2), 6-24. doi:10.14254/jsdtl.2022.7-2.1

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... & Zheng, X. (2016). {TensorFlow}: a system for {Large-Scale} machine learning. In *12th USENIX symposium on operating systems design and implementation (OSDI 16)* (pp. 265-283).

Abou Elassad, Z. E., Mousannif, H., & Al Moatassime, H. (2020). A proactive decision support system for predicting traffic crash events: A critical analysis of imbalanced class distribution. *Knowledge-Based Systems*, *205*, 106314.

Batista, G. E., Prati, R. C., & Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data. *ACM SIGKDD explorations newsletter*, *6*(1), 20-29.

Breiman, L. (2001). Random forests. *Machine Learning*, *45*(1), 5-32.

Burez, J., & Van den Poel, D. (2008). Separating financial from commercial customer churn: A modeling step towards resolving the conflict between the sales and credit department. *Expert Systems with Applications*, *35*(1-2), 497-514.

Chang, L. Y., & Chien, J. T. (2013). Analysis of driver injury severity in truck-involved accidents using a non-parametric classification tree model. *Safety Science*, *51*(1), 17-22.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, *16*, 321-357.

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Cosslett, S. R. (1981). Maximum likelihood estimator for choice-based samples. *Econometrica: Journal of the Econometric Society*, 1289-1316.

Cover, T., & Hart, P. (1967). Nearest neighbor pattern classification. *IEEE transactions on information theory*, *13*(1), 21-27. https://doi.org/10.1109/TIT.1967.1053964.

Fernández, A., García, S., del Jesus, M. J., & Herrera, F. (2008). A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets. *Fuzzy Sets and Systems*, *159*(18), 2378-2398.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., & Herrera, F. (2018). *Learning from imbalanced data sets* (Vol. 10, pp. 978-3). Berlin: Springer.

Fiorentini, N., & Losa, M. (2020). Handling imbalanced data in road crash severity prediction by machine learning algorithms. *Infrastructures*, *5*(7), 61.

FMCSA (Federal Motor Carrier Safety Administration). Federal Regulatory Guide. 917–920.

García, V., Mollineda, R. A., & Sánchez, J. S. (2008). On the k-NN performance in a challenging scenario of imbalance and overlapping. *Pattern Analysis and Applications*, *11*(3), 269-280.

Gu, Q., Cai, Z., Zhu, L., & Huang, B. (2008, December). Data mining on imbalanced data sets. In *2008 International Conference on advanced computer theory and engineering* (pp. 1020-1024). IEEE.

Guo, P. T., Li, M. F., Luo, W., Tang, Q. F., Liu, Z. W., & Lin, Z. M. (2015). Digital mapping of soil organic matter for rubber plantation at regional scale: An application of random forest plus residuals kriging approach. *Geoderma*, *237*, 49-59.

Iranitalab, A., & Khattak, A. (2017). Comparison of four statistical and machine learning methods for crash severity prediction. *Accident Analysis & Prevention*, *108*, 27-36.

Izmailov, R., Vapnik, V., & Vashist, A. (2013, August). Multidimensional splines with infinite number of knots as SVM kernels. In *The 2013 International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7). IEEE.

Japkowicz, N. (2000, June). The class imbalance problem: Significance and strategies. In *Proc. of the Int'l Conf. on Artificial Intelligence* (Vol. 56, pp. 111-117).

Kotsiantis, S. B., Zaharakis, I. D., & Pintelas, P. E. (2006). Machine learning: a review of classification and combining techniques. *Artificial Intelligence Review*, *26*(3), 159-190.

Li, J., Guo, J., Wijnands, J. S., Yu, R., Xu, C., & Stevenson, M. (2022). Assessing injury severity of secondary incidents using support vector machines. *Journal of Transportation Safety & Security*, *14*(2), 197-216. https://doi.org/10.1080/19439962.2020.1754983.

Li, J., Liu, J., Liu, P., & Qi, Y. (2020). Analysis of factors contributing to the severity of large truck crashes. *Entropy*, *22*(11), 1191.

Li, P., Abdel-Aty, M., & Yuan, J. (2020). Real-time crash risk prediction on arterials based on LSTM-CNN. *Accident Analysis & Prevention*, *135*, 105371. https://doi.org/10.1016/j.aap.2019.105371.

Lill, R. A. (1977). A Review of BMCS Analysis and Summary of Accident Investigations, 1973-1976 With Respect to Downgrade Runaway Type Accidents. *American Truckers Association*..

Liu, X. Y., Wu, J., & Zhou, Z. H. (2008). Exploratory undersampling for class-imbalance learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, *39*(2), 539-550.

Ma, X., Dai, Z., He, Z., Ma, J., Wang, Y., & Wang, Y. (2017). Learning traffic as images: a deep convolutional neural network for large-scale transportation network speed prediction. *Sensors*, *17*(4), 818.

Mduma, N., Kalegele, K., & Machuve, D. (2019). A survey of machine learning approaches and techniques for student dropout prediction. *Data Science Journal*, *18*(1).

Moomen, M., Rezapour, M., Raja, M. N., & Ksaibati, K. (2020). Predicting injury severity and crash frequency: Insights into the impacts of geometric variables on downgrade crashes in Wyoming. *Journal of Traffic and Transportation Engineering (English edition)*, *7*(3), 375-383.

Mujalli, R. O., López, G., & Garach, L. (2016). Bayes classifiers for imbalanced traffic accidents datasets. *Accident Analysis & Prevention*, *88*, 37-51.

Rivera, G., Florencia, R., García, V., Ruiz, A., & Sánchez-Solís, J. P. (2020). News classification for identifying traffic incident points in a Spanish-speaking country: A real-world case study of class imbalance learning. *Applied Sciences*, *10*(18), 6253.

Schlögl, M., Stütz, R., Laaha, G., & Melcher, M. (2019). A comparison of statistical learning methods for deriving determining factors of accident occurrence from an imbalanced high resolution dataset. *Accident Analysis & Prevention*, *127*, 134-149.

Seiffert, C., Khoshgoftaar, T. M., & Van Hulse, J. (2009). Hybrid sampling for imbalanced data. *Integrated Computer-Aided Engineering*, *16*(3), 193-210.

Shi, Q., & Abdel-Aty, M. (2015). Big data applications in real-time traffic operation and safety monitoring and improvement on urban expressways. *Transportation Research Part C: Emerging Technologies*, *58*, 380-394. https://doi.org/10.1016/j.trc.2015.02.022.

Su, X., Zhou, T., Yan, X., Fan, J., & Yang, S. (2008). Interaction trees with censored survival data. *The International Journal of Biostatistics*, *4*(1). https://doi.org/10.2202/1557-4679.1071

Sun, Y., Wong, A. K., & Kamel, M. S. (2009). Classification of imbalanced data: A review. *International Journal of Pattern Recognition and Artificial Intelligence*, *23*(04), 687-719.

Tang, J., Liang, J., Han, C., Li, Z., & Huang, H. (2019). Crash injury severity analysis using a two-layer Stacking framework. *Accident Analysis & Prevention*, *122*, 226-238.

The Centers for Disease Control and Prevention. Retrieved from https://www.cdc.gov/

Weber, A., & Murray, D. C. (2014). *Evaluating the impact of commercial motor vehicle enforcement disparities on carrier safety performance*. American Transportation Research Institute.

Williams, D. P., Myers, V., & Silvious, M. S. (2009). Mine classification with imbalanced data. *IEEE Geoscience and Remote Sensing Letters, 6*(3), 528-532.

Wilson, J. (2004). Measuring personal travel and goods movement. *Tr News*, *234*, 28.

Xu, B., Wang, N., Chen, T., & Li, M. (2015). Empirical evaluation of rectified activations in convolutional network. *arXiv preprint arXiv:1505.00853*.

Yu, R., & Abdel-Aty, M. (2013). Utilizing support vector machine in real-time crash risk evaluation. *Accident Analysis & Prevention*, *51*, 252-259.

Yu, R., & Abdel-Aty, M. (2014). Using hierarchical Bayesian binary probit models to analyze crash injury severity on high speed facilities with real-time traffic data. *Accident Analysis & Prevention*, *62*, 161-167.

Zhou, F., Yin, H., Zhan, L., Li, H., Fan, Y., & Jiang, L. (2018, June). A Novel Ensemble Strategy Combining Gradient Boosted Decision Trees and Factorization Machine Based Neural Network for Clicks Prediction. In *2018 International Conference on Big Data and Artificial Intelligence (BDAI)* (pp. 29-33). IEEE.

Journal of Sustainable Development of Transport and Logistics (ISSN: 2520-2979) is published by **Scientific Publishing House "CSR", Poland, EU** and **Scientific Publishing House "SciView", Poland, EU**
Publishing with JSDTL ensures:
• Immediate, universal access to your article on publication
• High visibility and discoverability via the JSDTL website
• Rapid publication
• Guaranteed legacy preservation of your article
• Discounts and waivers for authors in developing regions
Submit your manuscript to a JSDTL at https://jsdtl.sciview.net/ or submit.jsdtl@sciview.net