Proceedings of the 2023 International Conference on Machine Learning and Automation
DOI: 10.54254/2755-2721/31/20230132

Prediction on traffic accidents' severity levels leveraging machine learning-based methods on imbalanced data

Jialiang Chen

School of Statistics and Mathematics, Central University of Finance and Economics, Beijing, 102206, China

2018310889@email.cufe.edu.cn

Abstract. Traffic accidents are a significant problem in many countries, resulting in thousands of injuries and deaths every year. By estimating the severity of traffic accidents, traffic safety together with the crash survival rates could be improved, by taking effective prevention measures at the location where accidents are plentiful and severe. This paper studies the prediction by different classification methods on traffic accident severity levels. The data set used includes 1.6 million traffic accidents recorded in the United Kingdom, ranging from 2000 to 2016. It is a difficult task, since the levels are imbalanced distributed, making it difficult to classify the records accordingly. To tackle this problem, this work compared several classification methods on the task and evaluates their performances from the aspects of time, accuracy, and adaptability on imbalanced data sets. Experiments suggest that among the methods, the decision tree is the most recommended. This paper also provides suggestions for improvements on similar tasks.

Keywords: traffic accident, machine learning, severity level prediction.

1. Introduction

This paper studies the rating of traffic accidents by severity from a statistical perspective. The rating is a practice in which people sort things into different classes or levels by to what extent they are. In rating traffic accident by severity, it is usual that a big portion of accident records will be considered as unimportant, thus rated in lower classes or levels. This is often seen in governmental or business statistics as the rating results will become references of how they will react to the accidents, or which pre-arranged plan will they take to a certain accident [1, 2].

Classification models are designed and used to solve rating problems. These models are supervised machine learning models that are used in various applications [3]. The input of these models is a matrix X, where each column represents a different aspect or feature, either numerical or categorized, of objects the model is going to classify [4]. According to this matrix, classification models will work out a column vector y as the output, where each line represents the predicted class or category of the object. This paper studies performances of different classification models on a same data set with an imbalanced distribution of the classes, which often occurs in traffic accident severity problems.

This paper first applies different classification models to the accident data set and compare their performance by both running time and the accuracy of the predicted classifications. Then an additional grading method will be applied to further investigate these models and compensate for some of the weaknesses in the traditional accuracy-based evaluation methods.

2. Method

2.1. Data set

The data set used in this paper is 1.6 million UK traffic accidents, which is the amassed traffic data from 2000 to 2016 by the UK government with over 1.6 million accidents documented in the process [5]. This is a typical case of an imbalanced-distributed data set. The original source of these accident data is the police reports, which directly overlook minor incidents and roughly classifies the accidents included into 3 levels of severity. The distribution of this severity levels is rather imbalanced, with about 85 per cent of the accidents included in the data set classified as the third or the least severe level.

2.2. Models

2.2.1. Logistic regression. It is one of the most commonly used linear models. Despite the name, this method is often used as a classifier. It is therefore mentioned to also as the log-linear classifier. In this method, a logistic function is applied to model the probabilities of the possible outcomes of single trials.

2.2.2. Perceptron. The perceptron is a linear algorithm for classification. It is mostly applied on large scale learning. Neither does this algorithm require a learning rate, nor is it regularized. Additionally, the perceptron's model is updated only when mistake occurs, so it is expected to train slightly faster than Stochastic Gradient Descent, at the expense of sparser models [6].

2.2.3. Passive-aggressive algorithms (PAA). They are a set of large-scale machine learning algorithms. While they do not need a learning rate, these algorithms actually include a regularization parameter [7].

2.2.4. K nearest neighbors (KNN). It is a neighbor-based classification method. These classification methods store instances of the training data without constructing a general internal model, so they are sorted as non-generalizing or instance-based models. For each point in the data set, these classifiers decide their predictions by considering its nearest neighbors, collect their votes and take the simple majority. In the particular KNN method, a data class will be assigned to a query point, if and only if it has more representatives among the k nearest neighbors of the point than any other classes, where hyper-parameter k is given by the user.

2.2.5. Support vector machines (SVMs). They are a family of supervised methods. In a high or infinite dimensional space, a SVM constructs at least one hyper-plane to perform classification. This is known as support vector classification (SVC). Different sub-methods of SVC use slightly different mathematical formulations to create hyper-planes, by which they divide the whole space, and accordingly sort points into different classes [8].

2.2.6. Stochastic gradient descent (SGD). In classification tasks with convex loss functions, Stochastic Gradient Descent (SGD), serves as a very simple yet useful optimization technique, especially when the number of samples is very large. These tasks include SVMs and Logistic Regression.

2.2.7. Decision tree. It is a non-parametric supervised learning method. Its application includes a wide range of both classification and regression tasks. The method first learns some basic decision rules from the data features, and then builds a tree-shaped model to predict the value of the target variable. A decision tree classifier is a classification model using decision tree methods that can be applied on both binary and multi-class tasks [9].

Among its parameters, max depth is an important one in the decision tree method. As mentioned above, decision tree method builds a tree-shaped model to predict the value. The maximum depth of this tree represents the maximum number of choices or decision this method will make in predicting the value of a certain target variable.

2.2.8. Random forest. As a classifier, Random Forest is an augmented algorithm which takes based on decision trees with some improvements involving randomness. Similar to Extra-Trees, this algorithm includes a perturb-and-combine technique, which means a diverse set of classifiers will be created or employed in its construction, and the averaged prediction of those individual classifiers will be considered as the final prediction [10].

In a Random Forest, each tree will be built from a sample drawn with replacement from the training set. During the construction, when nodes are split, the best split will be decided from either all or a random subset of input. This randomness is aimed at making forest estimator more stable, reducing its variance.

3. Result and discussion

3.1. Performance of various models

Table 1 shows the result produced by different classification methods. Each method has been called for at least 10 times to record the average run-time and accuracy.

Table 1. Performance and time consumption of various models.

Record	Method	Time (s)	Accuracy (%)
1	Logistic Regression	1.30	84.72
2	Perceptron	1.06	67.57
3	PAA	0.89	81.58
4	SVC	554.63	84.74
5	SGD	0.99	84.73
6	KNN	49.06	84.49
7	Decision Tree	0.68	84.79
8	Random Forest	2.90	84.75
9	MLP	8.18	84.74

The table above shows the result produced by different classification methods. Each method has been called for at least 10 times to record the average run-time and accuracy.

In all experiences, the data set are randomly split into training and testing sets. The training set makes up 80 per cent of the original data, while the remaining 20 per cent forms the testing set.

The first 3 experiments are on the performances of linear models. In experience 3, the regularization parameter is set as 1.0. The accuracy produced by perceptron and passive-aggressive classifiers are relatively lower, but they also consume less time than the logistic regression.

The 4th experiment calls support vector machine in this classification task. While producing a decent accuracy, the obviously large time consumed makes this method less favored. In the following record, a stochastic gradient descent technique is applied with the frame of SVM algorithm. The resulting accuracy was slightly lower, but the performance of this trick on saving run-time is outstanding.

The 6th experiment involves the k-th nearest neighbors. This model takes less time than support vector machine did to work out a decent result but is still much more time-consuming in comparison with other methods.

Experiment 7 and 8 features tree structures. In experiments, the maximum depth of decision trees are unlimited. Although applying more classifiers in the construction of model, the improvement of the random forest on accuracy, compared with a single decision tree, is minor. While run-time is acceptable for both, it takes the random forest approximately 3 times longer to make the prediction.

Out last recorded experiment uses multi-layer perceptron as its method. As a representative of supervised neural network models, the accuracy of this method turns out to be decent. However, the time it takes to predict is longer than less complicated models, such as linear ones.

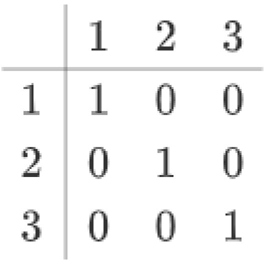
To sum up, different classification methods are used in this paper to predict the severity of accidents. Most methods present approximately the same accuracy of 85 per cent and more significant differences are found in the running time. Methods like decision tree and passive aggressive algorithm produce the result in less than one second, while more complex methods, like random forest and MLP, consume more than ten seconds. It can be implied from this that the classification task here is rather a simple one, and, by some means, it indeed is, as millions of those accident records are divided into only 3 levels by their severity. It could be first ruled out the less practical methods such as the original support vector machine, which takes as long as minutes to finish the prediction, and then focus on the result of those quicker methods.

Further investigation involves the distribution of those 3 levels. As was previously pointed out, among those accident records, about 85 per cent are rated in the third level, which means the 85 per cent accuracy can be reached by simply regarding all records as level 3 accidents, instead of employing any of the classification methods. While this seems intolerable, it is by some means logical. This data set was made by the British government from police reports, which means most accidents would be regarded as minor incidents, and would be answered only by limited actions, or the cost would be dramatically raised.

However, the method of simply putting everything in level 3 will still be problematic. Even a rather small number of severe accidents may cause relatively big effects if they are underestimated, or vice versa. For example, the government’s ability to deal with these kinds of accidents will be doubted by the public if severe accidents are ignored or insufficiently dealt with, or the cost to the traffic-related departments may go over the budget if too many less- severe accidents are to be overreacted.

3.2. Performance measured by imbalanced-aware index

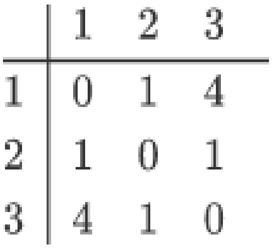
To have one single value that reflects the performance of these algorithms on the task, the class labels could be converted to numbers. To be more precise, the accuracy in the task is conventionally calculated in this task as shown in Figure 1.



	1	2	3
1	1	0	0
2	0	1	0
3	0	0	1

Figure 1. Example of result distribution (Picture credit: Original).

The first row represents the actual class of an accident record, and the first column represents the prediction a classification method makes. The summed-up score from all records, which equals to the number of precise predictions, are then divided by the most score possible, which equals to the number of records, and the result is known as the accuracy. Due to the fact that most records are in level 3, it could be assured that 85% from the heavy weight of the bottom right cell. However, if the way to calculate is modified, for example, as demonstrated in Figure 2.



	1	2	3
1	0	1	4
2	1	0	1
3	4	1	0

Figure 2. Example of modified calculation (Picture credit: Original).

The scores are compared, and the smallest value is calculated as the best, the all-level-3 method will somehow be punished because of the fact that every mistaken prediction that regards a level 3 accident as level 1 or vice versa will now cost 4 times the loss than they did in the calculation of accuracy. Obviously, the values of loss in the table above are determined as the squared difference between their class labels. This leaves plenty of space of operation. For example, if considering level 2 and 3 to be approximately the same, while level 1 is so different that any mistaken prediction involved is intolerable, than changing the number representing it from 1 to 100, which can severely punish some of the mistakes, such as what it is made in the all-level-3 method. Moreover, this matrix does not even have to be symmetric. For example, if the government has plenty of money planned on dealing accidents so that they can allow most overestimation, then the matrix of loss can be designed as lower-triangular.

Table 2 demonstrates some of the loss-matrix results. The matrix here uses the squared differences of class labels as loss values.

Table 2. Performance of various models using imbalanced-aware matrix.

Record	Method	Loss
1	All-level-3	27317
2	Logistic Regression	27317
3	Perceptron	50124
4	PAA	27317
5	SGD	27317
6	Decision Tree	27267
7	Random Forest	27267
8	MLP	27308

4. Conclusion

In this paper, several classification methods are used on the 1.6 million UK traffic accident data set to predict the severity of recorded accidents. Of all methods featured, the decision tree is the fastest to work out the result. While most algorithms are able to return the value within acceptable time, kth nearest neighbors and support vector machines consumes more than 10 seconds, which displays an obvious gap. To evaluate the algorithms left for discussion, this paper first considers the traditional standard of accuracy. However, there is only minor differences between algorithms in this aspect: most of them get an accuracy of about 85 per cent. Given that the original data set has a single class that makes up about 85 per cent, this could not be considered as a coincidence. Other standards such as F1 scores are available in this particular task, but not every imbalanced tasks. If a single number is needed to reflect the performance of an algorithm in an imbalanced classification task as such, one may need to re-design the evaluation so that class labels are replaced by numerical representatives. The matrices of loss that comes into discussion here are very maneuverable tools to evaluate the prediction, because they can be adjusted in different tasks accordingly. In this paper's task, a simple matrix in which loss are weighted by the squared difference of class labels, can further reflect what the algorithms actually did. Although more methods still have similar results in this loss value, decision trees and its derivative, random forest, gets less loss records than other algorithms. It can be predicted that they are at least not just putting all records into the 3rd level. It is thus recommended to consider a same or similar algorithm in such tasks that involves classifying records in imbalanced data sets.

In fact, what this loss matrix do in this task is converting a categorized data and a classification task to a numerical data and a regression task. The key point in making this loss matrix successful is to design its values properly. In regression tasks people have originally values from experiments, statistics, or surveys, while in classification tasks, they either fail to get them, or have sorted them into different categories. An imbalanced data set is the result of a less imaginative situation, because it covers up or overlooked important factors that makes classification methods work. As a result, it is necessary to restore them in the evaluation. This provides a hint that it may be useful to consider some regression methods or techniques in such classification models, despite the toughness and potential problematic in the process of conversion.

References

[1] Rastogi, A., & Sangal, A. L. (2021). Accident Risk Rating of Streets Using Ensemble Techniques of Machine Learning. In Innovations in Computer Science and Engineering: Proceedings of 8th ICICSE, 623-631.

[2] Gutierrez-Osorio, C., & Pedraza, C. (2020). Modern data sources and techniques for analysis and forecast of road accidents: A review. Journal of traffic and transportation engineering, 7(4), 432-446.

[3] Jordan, M. I., & Mitchell, T. M. (2015). Machine learning: Trends, perspectives, and prospects. Science, 349(6245), 255-260.

[4] Lever, J. (2016). Classification evaluation: It is important to understand both what a classification metric expresses and what it hides. Nature methods, 13(8), 603-605.

[5] Haynes, S., Estin, P. C., Lazarevski, S., Soosay, M., & Kor, A. L. (2019). Data analytics: Factors of traffic accidents in the uk. In 2019 10th International Conference on Dependable Systems, Services and Technologies (DESSERT), 120-126.

[6] Gardner, M. W., & Dorling, S. R. (1998). Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. Atmospheric environment, 32(14-15), 2627-2636.

[7] Shalev-Shwartz, S., Crammer, K., Dekel, O., & Singer, Y. (2003). Online passive-aggressive algorithms. Advances in neural information processing systems, 16.

[8] Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. (2020). A comprehensive survey on support vector machine classification: Applications, challenges and trends. Neurocomputing, 408, 189-215.

[9] Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. Journal of Applied Science and Technology Trends, 2(01), 20-28.

[10] Speiser, J. L., Miller, M. E., Tooze, J., & Ip, E. (2019). A comparison of random forest variable selection methods for classification prediction modeling. Expert systems with applications, 134, 93-101.