PATIPHAN KAEWWICHIAN, Ph.D.
E-mail: patiphan.ka@rmuti.ac.th
Faculty of Engineering
Rajamangala University of Technology Isan
Khon Kaen Campus, Khon Kaen, 40000, Thailand

Intelligent Transport Systems (ITS)
Original Scientific Paper
Submitted: 6 Oct. 2020
Accepted: 11 Jan. 2021

MULTICLASS CLASSIFICATION WITH IMBALANCED DATASETS FOR CAR OWNERSHIP DEMAND MODEL – COST-SENSITIVE LEARNING

ABSTRACT

In terms of the travel demand prediction from the household car ownership model, if the imbalanced data were used to support the transportation policy via a machine learning model, it would negatively affect the algorithm training process. The data on household car ownership obtained from the study project for the expressway preparation in the Khon Kaen Province (2015) was an unbalanced dataset. In other words, the number of members of the minority class is lower than the rest of the answer classes. The result is a bias in data classification. Consequently, this research suggested balancing the datasets with cost-sensitive learning methods, including decision trees, k-nearest neighbors (kNN), and naive Bayes algorithms. Before creating the 3-class model, a k-folds cross-validation method was applied to classify the datasets to define true positive rate (TPR) for the model's performance validation. The outcome indicated that the kNN algorithm demonstrated the best performance for the minority class data prediction compared to other algorithms. It provides TPR for rural and suburban area types, which are region types with very different imbalance ratios, before balancing the data of 46.9% and 46.4%. After balancing the data (MCN1), TPR values were 84.4% and 81.4%, respectively.

KEYWORDS

cost matrix; decision trees; k-nearest neighbors (kNN); cross-validation; tour-based model.

1. INTRODUCTION

Data classification is an analysis method used to define data patterns, classification models, and classification rules. This method predicts different data types, either present or future, such as travel demand predictions. Several minor models were used, including the household car ownership models, trip generation models, tour generation models, trip distribution models, travel time choice models, and travel route choice models [1], with either trip or tour used as the unit of analysis [2]. There are several techniques for data classification [3], e.g. the decision tree (DT) presenting different logical conditions; k-nearest neighbors (kNN) used for the mathematic calculation to find distance or weight; and naive Bayes used to find the probability in the training data. The selection for a high performing technique should rely on the parameters indicating the data classification performance, e.g. accuracy, precision, recall, F1-score. Still, these techniques do not work well on every dataset. For example, some work more effectively on the balanced data than on the imbalanced one; the flat data contains the classes with a similar number of datasets [4]. The imbalanced data has courses with a different number of datasets. At this point, the imbalanced data classification becomes a thought-provoking issue because some of the minority classes include either significant or outstanding data. Consequently, for more effective data analysis, the model's performance to classify the minority class needs to be improved before algorithm training with suitable parameters for the imbalanced data [5, 6].

In the imbalanced data, the numbers of each class would be completely different. This imbalanced class is a critical issue often found in the research fields of medical science [7], marketing, banking, and production industry [8, 9]. However, it is still rare in transportation planning, especially in using the data with the machine learning model, which are popular and state-of-the-art approaches [10], to predict the household car ownership.

Due to the problem, several methods have been purposively invented to fix these imbalanced data at a data level and an algorithm level to improve

the minority class [11]. Precisely at a data level, the imbalanced data could be solved via sampling techniques. Meanwhile, at an algorithm level, the algorithm's performance would be improved with any helpful technique during the data training process to effectively predict the unseen data while testing the model, such as cost-sensitive learning methods (CSL) [12]. The classification performance at both levels was similar [13]. In increasing data, CSL methods performed better than the sampling methods [14]. Consequently, this research aimed to improve the minority class with a cost matrix table with two categories.

This research proposed a useful technique to improve the algorithm's performance to classify the household car ownership demand model with the 3-class problem. The study used CSL methods to solve the imbalanced data with its negative effect on the classification performance on the minority class, and the feature section, a feature-level data management technique, to find the first ten parameters with the optimal weight. Finally, the data classification performance would be affirmed by the true positive rate (TPR), F1-score, accuracy, false negative rate (FNR), and false positive rate (FPR).

The paper is organized as follows: after the introduction, section 2 will focus on class distribution balancing, performance indicators, and solutions to the class imbalance problem at an algorithm level. Section 3 presents the algorithms selected for the study. The description of the experiment in our research can be found in Section 4, while the obtained results and discussion are presented in Section 5. The concluding remarks and future work are outlined in Section 6.

2. CLASS DISTRIBUTION BALANCING

This section will explain the problem that might exist due to the imbalanced data distribution in each target class and the classification performance indicators for the imbalanced data. The final part is a review of the CSL methods.

2.1 The class imbalance problem

The imbalanced data can be practically seen as unequal numbers of samples in each target class, with most classification problems in research with two categories, as seen in *Figure 1*. Specifically, this research is mainly focused on the imbalanced datasets with a 3-class problem found in transportation engineering studies, e.g. travel mode choice [15]. In other words, there is one class indicating a lower data number than the other courses in the same dataset. Similarly, the literature review on the minority class was the one that most catches our attention [16, 17]. In case that the transportation problem data is imbalanced, most standard algorithms cannot classify the information correctly because they were designed as an accuracy-oriented model. The results can be biased by the majority of classes, which are easier for algorithm training.

The majority class classification or negative instances affect the accuracy metrics more than the correct prediction on the minority class or positive

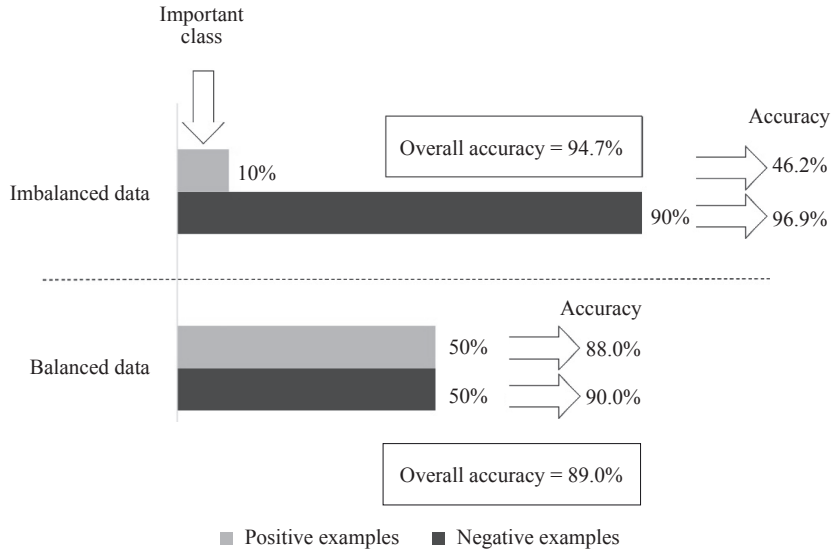


Figure 1 – The illustration of balanced and imbalanced data sets

class example. Therefore, this positive class example might be ignored (or treated as noise) since the standard prediction rule states that the negative class example notably provides a higher accuracy rate.

From those mentioned above, this article aimed to improve classification performance on the minority class, i.e., household without a car (Class 0), whereas families with one car and 2+ cars (Class 1 and 2) were the majority class. At this point, the researcher also chose to use the imbalance ratio (IR) defined by the negative class example or the majority class divided by the number of positive class examples or the minority class as the consideration values of each area type. To be exact, if the IR were higher than 9 [18], the dataset would be highly imbalanced. On the contrary, if the IR were lower than 9, the dataset imbalance would be either moderate or low.

2.2 Performance evaluation

A validation technique was necessary to affirm the algorithm's classification performance appropriately. It could guide the model creation; therefore, this research intended to suggest a useful method to validate the algorithm's classification performance on the 3-class imbalanced data for each target class. On this matter, both accurate and inaccurate results would be directly recorded in a confusion matrix table, as presented in *Table 1* adapted from [19].

As presented in the confusion matrix table, several values were regularly used to validate the model's classification performance, including accuracy, precision, recall, sensitivity, or TPR, FPR, FNR, and F1-score. Explicitly, accuracy represented the model's correctness by considering all classes. Precision was an indicator of the model's accuracy by separately considering each category one by one. TPR represented the model's correctness by separately considering each class one by one. FPR represented a ratio of the minority class misclassified as the majority. FNR represented a majority class ratio misclassified as the minority, and F1-score was the result of the precision-recall evaluation by considering each category one at a time.

However, if the data were imbalanced, the model's classification performance on the minority class would be typically evaluated with a TPR since TPR could define the actual travel distribution data [20]. The table presented TP, FP, TN, and FN, where TP correctly predicted data from the target class. FP was the dataset classified as Class 0, but in other classes, TN was the correctly predicted data in any class besides Class 0. FN were the datasets classified to be in other classes but was in Class 0. It was noted that TN was the opposite of TP, while FN was the opposite of FP. All these values could be used to find accuracy=TP+TN/(TP+FN+FP+TN)·100; precision=TP/(TP+FP)·100; TPR=TP/(TP+FN)·100; FPR=FP/(FP+TN)·100; FNR=FN/(FN+TP)·100; and F1-score=(2·precision·TPR)/(Precision + TPR). If these values were high, both precision and TPR would be high too.

2.3 Cost-sensitive learning, CSL

The cost-sensitive approach assigns unequal weights to each class so that the minority would have more weight, whereas the majority class had lower weight. In effect, the CSL method gives weights to all values using the cost matrix containing. Similar to the confusion matrix, where numbers of rows and columns were equal to the class number, the incorrect prediction was assigned with more weights than the correct prediction, and the accurate prediction values were 0. The model would consider the importance within the cost matrix and minimize the total weight.

To balance the data with the CSL method, the researcher invented the cost matrix table by randomly adjusting false negative (FN) and false positive (FP) values. The minimum was 5.0, and the maximum was 25. This data adjustment was run until the cost matrix parameter was lower or stable [21].

Table 1 – Confusion matrix for classification (class 0)Table 1 – Confusion matrix for classification (class 0)

	True positive class (0)	True negative class (1)	True negative class (2)
Positive prediction (0)	True positive (TP) (P0,T0)	False positive (FP) (P0,T1)	False positive (FP) (P0,T2)
Negative prediction (1)	False negative (FN) (P1,T0)	True negative (TN) (P1,T1)	True negative (TN) (P1,T2)
Negative prediction (2)	False negative (FN) (P2,T0)	True negative (TN) (P2,T1)	True negative (TN) (P2,T2)

3. ALGORITHMS SELECTED

In this section, the researcher suggested the study algorithm and the critical problem that the standard machine learning algorithms could not work effectively with the imbalanced data.

Many machine learning algorithms utilize the class distribution in the training datasets to find the likelihood of each class examples for the model to predict the data. Accordingly, several machine learning algorithms, e.g. decision trees (DT), k-nearest neighbors (kNN), and naive Bayes, will recognize that the minority class is as important as the majority class in this research.

However, there are also machine learning algorithms used to classify information in addition to the above algorithms, for instance, the artificial neural network (ANN), a technique based on the use of computer simulations of human brain activity [22]. This neural network is a processing unit that produces either linear or nonlinear transmission between input and output variables; support vector machines (SVM) are one of the most popular and discussed machine learning algorithms. The learning strategy is finding the optimal split hyperplane to maximize margins and reduce training errors based only on margin data points [23, 24].

3.1 Decision trees

DT is an explanation technique by summarizing truths or the related data to construct the rule of the DT. This technique has often been implemented the most since it helps the model to interpret and make the data more understandable. In this regard, the model was created using the repeated attribute partitioning.

At each level of the tree (from the root node), the algorithm would find the information gain ratio (IG) of each attribute or feature and compare them with the class to find the attribute with the highest IG. Assign it as the root of the decision tree (the selected attribute could classify the data examples for model creation and assign them with the same class if possible (maximizing the class-homogeneity)). The ultimate goal of the decision trees algorithm is to separate all data into subgroups with the same answers or classes, i.e., the sequence of slicing data to generate appropriate if-then rules. From root to leaf, the resulting rules can illustrate an example. All information available is complete. In other words, this process is repeated until the last node (leaf node) and every node can classify the sample into individual subgroups with a homogeneous class. After that, this process stops, and finally, a decision tree model is created.

To avoid overfitting, trees are generally pruned to improve the predictability of decision structures (see [25, 26] for more details).

3.2 k-Nearest neighbors (kNN)

The kNN algorithm compares the unknown sample with the k training sample, the closest neighbor of the new sample. The preliminary theoretical results can be found in [27], and a comprehensive overview can be found in [28]. The first step of applying the kNN algorithm on a new example is to find the k proximity training examples. “Proximity” is determined from a distance in the n-dimensional space depending on the number of attributes in the training example.

Different metrics, such as the Euclidean distance, can calculate the distance between the new example and the training examples. Because the length is often based on absolute value, it is necessary to normalize data before training and use the kNN algorithm.

In the next step, the kNN algorithm classifies the unknown sample by voting on the majority of the neighbors it finds. In the case of a regression, the predicted value is the average of the found values of the neighbor.

In an imbalanced training dataset, an example of a small class occurs sparingly in the data space. Given the testing dataset, the calculated closest neighbor k has a high probability of finding a sample from a prevalent type. Test cases from small class sizes were likely to be classified incorrectly. Research in [29] and [30] report this notice.

3.3 Naive Bayes

Naive Bayes is a technique for constructing classifiers, high-bias, low-variance classifiers, and building a good model even with a small dataset. It is based on the Bayes' theorem and it is a probabilistic classifier. Naive Bayes classifiers assume that a specific component's estimation is independent of estimating other elements for a given class variable. Bayes’ theorem: $P(C|A) = P(A|C) * P(C)/P(A$, where $P(C|A)$ is the probability value that data with attribute A will have class C, $P(A|C)$ is the

Table 2 – A cost matrix available for FN and FP

	C (P0,T0)	C (P0,T1)	C (P0,T2)	C (P1,T0)	C (P1,T1)	C (P1,T2)	C (P2,T0)	C (P2,T1)	C (P2,T2)
NMC	0	0	0	0	0	0	0	0	0
MCN1	0	1	1	5	0	1	5	1	0
MCN2	0	1	1	10	0	1	10	1	0
MCN3	0	1	1	15	0	1	15	1	0
MCN4	0	1	1	20	0	1	20	1	0
MCN5	0	1	1	25	0	1	25	1	0
MCNP1	0	2	2	5	0	1	5	1	0

Note: NMC is an unadjusted unbalanced dataset; the case of MCN1-5 is to reduce the error of FN by setting the penalty higher than the others, with fines starting from 5, 10, 15, 20, and 25, respectively; The MCNP1 case attempts to reduce an FN and FP error, the where FN-error should be lower than the FP-error. In this regard, both MCN1-5 and MCNP1 considered the model's performance to predict Class 0.

likelihood of attribute A having class C in training data [31], P(A) is the probability of attribute A, and P(C) is the class C probability.

Although it expects an impossible condition that attribute values are restrictively free, it performs shockingly well on substantial datasets where this condition is assumed and holds good [32].

3.4 Parameters

This section presents the default parameters derived from the RapidMiner Studio Educational 9.7 Software Tool for each algorithm, including the decision tree default parameters (criterion, gain ratio, 20 maximal depth of a tree, 0.25 the confidence level, 0.1 the minimal gain of a node, and 2.0 minimal leaf size). kNN default parameters are used to measure the distance between the predicted data with the k number of neighboring data [33] (k=5, measure types: Euclidean Distance) and Naive Bayes default parameters (Laplace correction). The researcher also defined the weights in the cost matric table to see any consequential effects, as presented in *Table 2*.

Before and after data balancing, the data were mainly applied to create and test the performance of the household car ownership demand model via each machine learning algorithm. All results were later compared to the statistical significance tests (T-Test) (alpha=0.05).

4. DATASETS

In this research, the travel data from the Engineering, Economical, Financial, and Environmental Feasibility Study for the Khon Kaen Expressway Master Plan 2015 (Thailand) was implemented. Because the study area population has similar characteristics, systematic random sampling was used to achieve the number of households equal to 2,015 families (2% of the total households in the target area and 4,757 people provided with travel information, 616 without travel information). The data collection was conducted through a face-to-face interview. The participants were chosen from 73 zones, and the GIS database was used to categorize the area of this study; 10 more zones from the suburban and urban areas were added, so the total was 83 zones. The residential density classified these zones into 4 area types, including central business district (CBD), urban area, suburban area, and rural areas, as shown in *Figure 2*. These area types indicate travel characteristics of each household car and are one of the variables that indicate the source region type, as well as the primary destination location of each

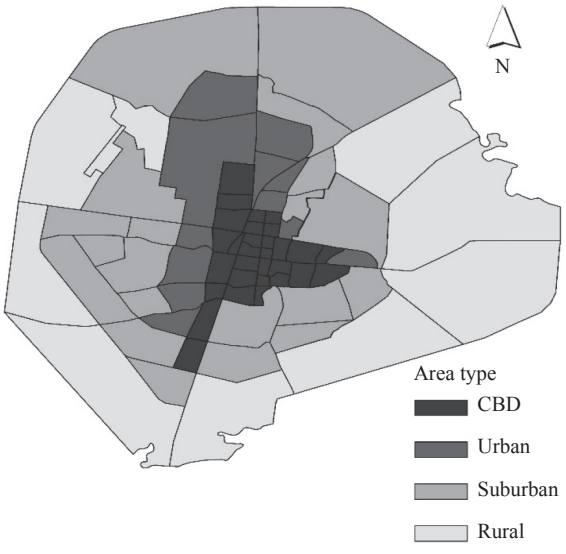


Figure 2 – Area types in the Khon Kaen municipality (Thailand)

Table 3 – Data set summary.

Dataset (Area types)	Total instance	% Minority class (Class 0)	Imbalance ratio, IR
Rural	809	26.1	2.83
Total	4852	18.0	4.55
Urban	1054	16.8	4.95
CBD.	1358	16.4	5.09
Suburban	1631	16.1	5.20

trip under one tour. *Table 3* presents the household car ownership dataset summary and the imbalanced ratio [34] around the study area. Class 0 was the minority class, and the rest was the majority class.

5. RESULTS

In term of the performance test of the DT, kNN, and naive Bayes algorithms on the imbalanced data, the researcher compared the predicted results from each algorithm on each of the area types to one another before using the best performing algorithm to create the model along with balancing the data with the cost-sensitive learning methods (CSL). In the meantime, k-fold cross-validation was used to develop and validate the model's performance before and after data balancing. The default parameters of each algorithm and another ten parameters selected by weight optimization were implemented for model creation and validation.

5.1 An experimental comparison

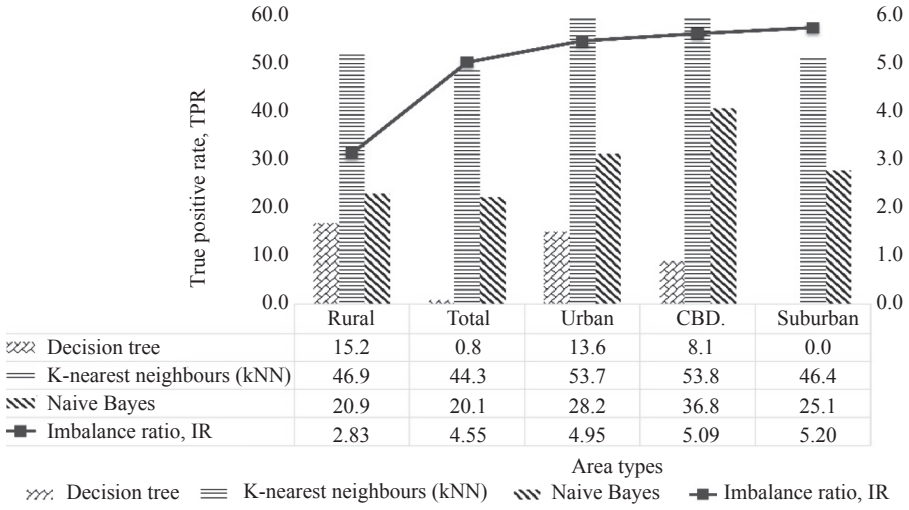
The findings indicated that the kNN algorithm provided a high TPR with a higher accuracy rate in classifying the dataset in the minority class (Class 0) in every imbalanced ratio (*Figure 3a*). Also, it provided a low error rate in organizing the datasets in courses other than Class 0, compared to different algorithms (FNR) (*Figure 3b*). Apparently, if the imbalanced data was classified by standard classification algorithms, the results would be completely biased by the majority class, Class 1 and 2. Hence, FNR was close to 100%; for instance, the DT model in the suburban area showed IR = 5.20, whereas the kNN algorithm gave the lowest FNR in every IR depending on each area type. To highlight the data classification algorithm's performance, the researcher decided to use the kNN algorithm to create and validate the model's performance with every single imbalanced dataset in each area type. CSL methods were also implemented to balance the dataset before training it with the kNN algorithm.

5.2 Data balancing (algorithm-level)

In this section, the researcher attempted to solve the imbalanced data at an algorithm level based on the imbalance in each area type using cost-sensitive learning methods (CSL) and the (kNN algorithm), high performing algorithm for this study. This study was strictly conducted to improve the model's performance in classifying the minority class or "positive" class with a higher TPR. Additionally, the researcher defined the cost matrix table (*Table 2*).

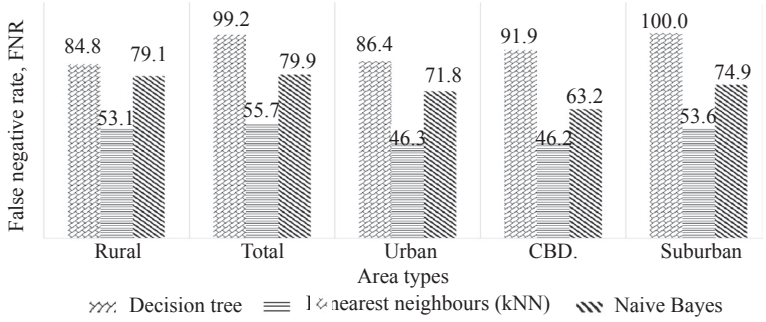
The fact is that this research began with a non-cost classification or NMC that classifies the imbalanced data that was not adjusted with CSL, which means that the cost of every data prediction was equal. In fact, the MCN1-5 case was an attempt to reduce error from the FN by defining a higher fine than for other mistakes in which the penalty could be 5, 10, 15, 20, and 25; meanwhile, the MCNP1 case was an attempt to reduce error from both FN and FP where the FN-error should be lower than the FP-error. In this regard, both MCN1-5 and MCNP1 considered the model's performance to predict Class 0.

Figures 4 and 5 illustrated the impact of the TPR and FPR on Class 0 only in the case of FN-error reduction, respectively. *Figure 4* shows that the TPR in all area types was higher in every dataset when defined with different costs. Still, the FPR in *Figure 5* is also increased. Significantly, when the cost adjustment reached a certain level, the TPR seemed to stabilize, indicating that when the data with the IR from 2.83–5.20 was already balanced with the CSL, the kNN algorithm was assigned to create the model, and later the classification ratio of minority class (Class 0) became more accurate. Consequently, an appropriate cost selection helped maximize the model's prediction performance on the minority class (Class 0), but an increase of the FPR was still unavoidable.



	Rural	Total	Urban	CBD.	Suburban
Decision tree	15.2	0.8	13.6	8.1	0.0
K-nearest neighbours (kNN)	46.9	44.3	53.7	53.8	46.4
Naive Bayes	20.9	20.1	28.2	36.8	25.1
Imbalance ratio, IR	2.83	4.55	4.95	5.09	5.20

a) True positive rate value of class 0 for each algorithm in all area types



b) False negative rate value of class 0 for each algorithm in all area types

Figure 3 - Performance comparison of classifiers on TPR and FNR for the minority class (class 0)

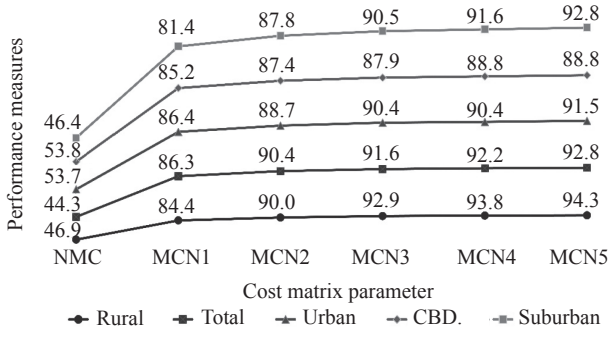


Figure 4 – Effects of the TPR on a minority class (class 0), case minimizing mistakes FN

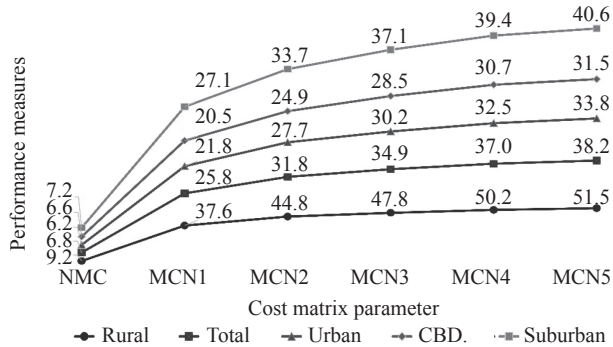


Figure 5 – Effects of the FPR on a minority class (class 0), case minimizing mistakes FN

The reduction of both FN-error and FP-error (MCNP1) for better prediction performance on the minority class (Class 0) was given in *Figures 6 and 7*. The figures showed that the defined cost matrix table affected the decrease of the TPR and the FPR when compared to the FN-error reduction (MCN1) that resulted when the dataset used to create and validate the model's performance had more of the majority class (Class 1 and 2) than the minority class (Class 0). Despite this, the k-NN algorithm classification provided more of the majority class than the minority class. Thus, the FN-error reduction added more chance or higher possibility for the majority class to be chosen for data prediction. In contrast, it was less possible for the minority class to be the choice despite the higher cost.

After considering the F1-score from every area type (IR = 2.83 – 5.20) in *Figure 8*, the FN-error reduction (defining a higher fee in the FN cost matrix table than that of the other errors) provided a higher F1-score of the minority class (Class 0) at the MCN1 compared to the case of the non-balancing data (at the NMC). However, at the MCN2-5, F1-score seemed to decrease. It usually happened that

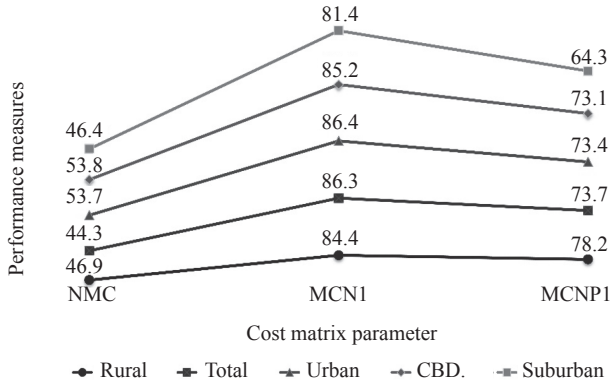


Figure 6 – Effects of the TPR on a minority class (class 0), case minimizing mistakes FN and FP

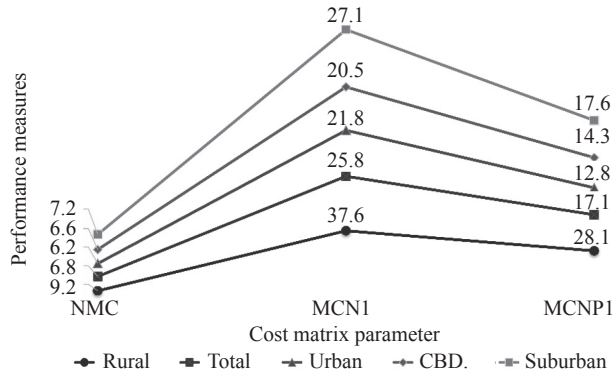


Figure 7 – Effects of the FPR on a minority class (class 0), case minimizing mistakes FN and FP

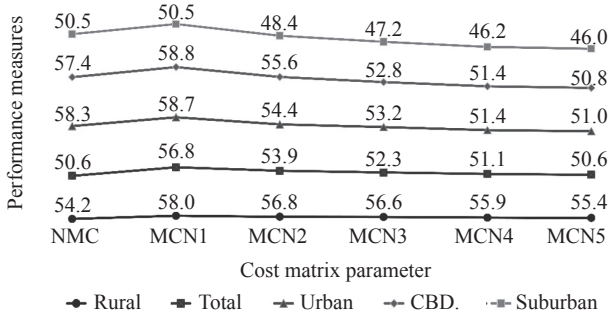


Figure 8 – Effects of the F1-score on a minority class (class 0), case minimizing mistakes FN

when the prediction performance of Class 0 was continually increasing until it was finally stable, precision would be continuously reducing as well. Accordingly, the F1-score of Class 0, an overall performance indicator, seemed to decline gradually.

When comparing the FN-error and FP-error reduction cases in *Figure 9* to the FN-error reduction alone, the imbalanced ratio of the dataset classified by the area types (IR = 2.83 – 5.09) was found to be low and moderate. At the same time, the F1-score seemed to be higher. Despite the fact that the IR of each area-type was increasing (IR = 5.20, suburban area type), the F1-score of the minority class (Class 0) was likely decreasing because the reduction of both the FN-error and the FP-error made the prediction biased by the majority class (Class 1, 2).

The research solved the imbalance problem at the algorithm level using CSL methods according to the imbalance of each area type: rural, total, urban, CBD, and suburban (IR 2.83–5.20). It shows that when using the default parameter of the kNN algorithm, the MCN1 provides class 0 predictive performance (the minority class) after balancing the data with the best cost matrix (higher than NMC in all area types). The TPR values for each area type were 84.4%, 86.3%, 86.4%, 85.2%, and 81.4%, respectively. The results have shown that balancing datasets before processing is beneficial. As a result, the model had a higher TPR (lower learning error rate); in other words, choosing the appropriate cost table would improve the predictive performance of the higher value of the minority class (class 0).

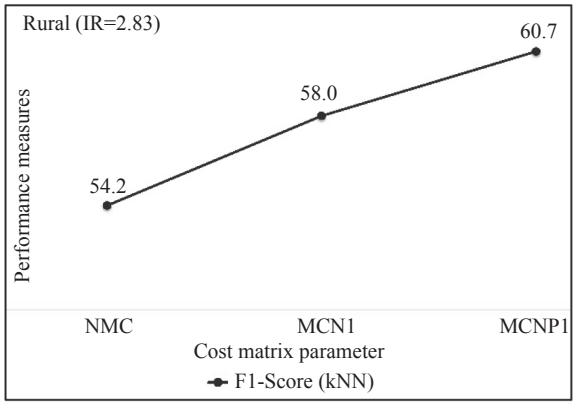
For kNN, the next key point is to find the best k parameter in the case of MCN1, in which case the highest F1-Score and the lowest FPR start by using the default k = 5, and then add different k values up to 100, while k values lower than 5, such as 1 or 3, are not taken into account because they may not be well distinguished [35].

For all datasets within the study area, the results confirm the suitability of using k equal to 5 as shown in *Figure 10*, showing different validity and k-values, where k = 5 gives the best classification accuracy.

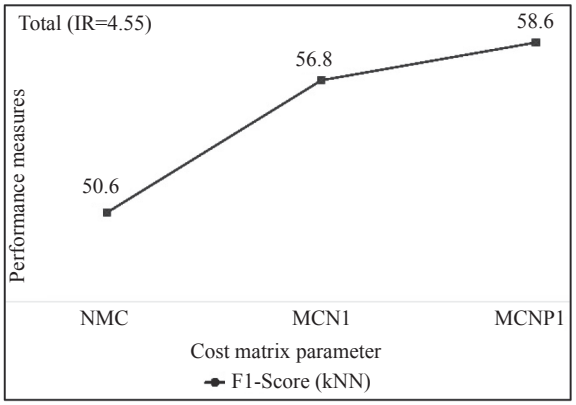
6. CONCLUSION AND FUTURE WORK

Explicitly, this research aimed to invent a useful model for the household car ownership demand prediction in five target area types. The imbalanced ratio ranged from 2.83–5.20, which negatively affected the model's performance to predict the minority class (Class 0). This research also highlighted the significance of data preparation. The parameters were selected from trip-based and tour-based models via weight optimization to find the first ten parameters with optimal model creation weights. Later, a cross-validation method was used to create and test if the model was high performing when classifying the data with standard algorithms.

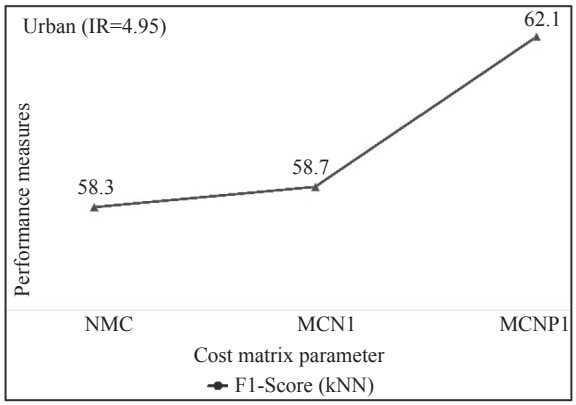
Conclusively, the research outcome revealed that when using the best-performing kNN algorithm to create the household car ownership demand model with a 3-class problem and data balancing by a cost-sensitive learning method at an



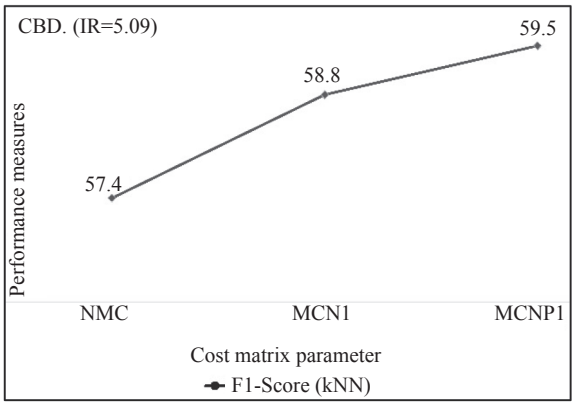
a) Rural area type



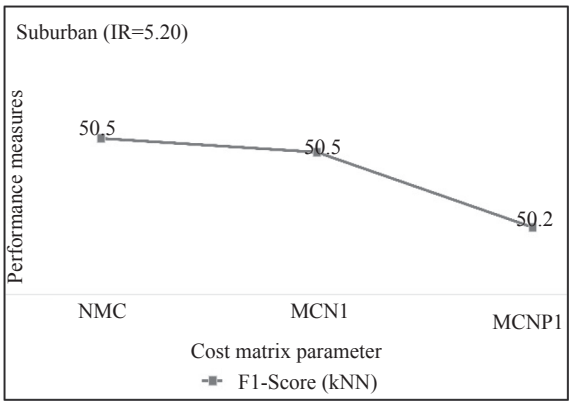
b) Total area type



c) Urban area type



d) CBD area type



e) Suburban area type

Figure 9 – Effects of the F1-score on a minority class (class 0), case minimizing mistakes FN and FP

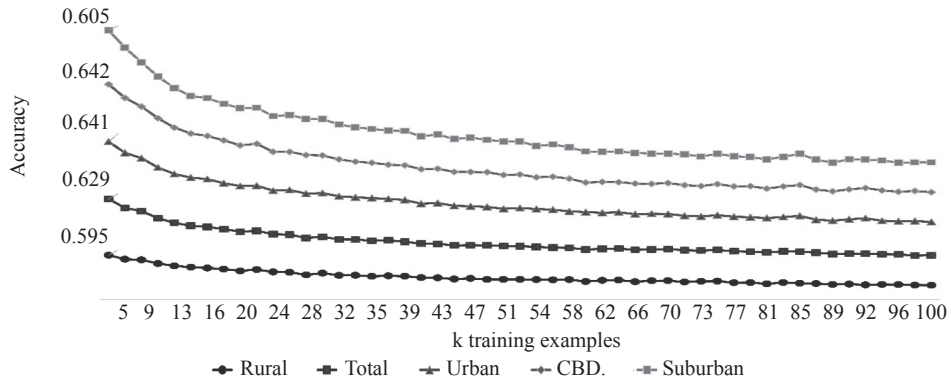


Figure 10 – Finding k training examples appropriate in each area type

algorithm level, the mode's classification performance on the minority class (Class 0) with only the reduced FN-error was significantly improved. However, both the FN-error and the FP-error were reduced based on the cost matrix table; the TPR of the minority classes also decreased significantly in each of the imbalanced ratios.

When the TPR of the minority class was increased, the TPR from other parameters could be either increased or decreased depending on each specific case. For example, the F1-score with the reduced FN would be reduced, but when both the FN-error and the FP-error were reduced, this score would be increased in every imbalanced ratio; IR = 5.20 was an exception. As a result, it was necessary to define the cost matrix table carefully. It impacted the model's classification performance on the household car ownership demand in the study; it could be either high or low. Hence, future work should focus on developing the model with better performance to solve class imbalance with sampling techniques at a data level, under-sampling, over-sampling, and combinations of methods; with ensemble classifier, and semi-supervised classifier at an algorithm level. It will be tested to increase prediction and policy formulation opportunities for better urban transportation planning through a machine learning model with appropriate household characteristics.

ACKNOWLEDGMENTS

I would like to thank Mr. Sorasak Seawsirikul, a lecturer at the Department of Civil Engineering, Faculty of Engineering at the Rajamangala University of Technology Isan, Khon Khen Campus, for his assistance with the coding process. I would also like to express gratitude to the Faculty of Engineering, Khon Kaen University, for supporting this work by providing the Khon Kaen Expressway Master Plan (Thailand) for 2015, which contributed significantly to this article.

ปฏิภาณ แก้ววิเชียร Ph.D.
E-mail: patiphan.ka@rmuti.ac.th
คณะวิศวกรรมศาสตร์ มหาวิทยาลัยเทคโนโลยีราชมงคลอีสาน วิทยาเขตขอนแก่น 40000

การจำแนกประเภทข้อมูลหลายคำตอบบนชุดข้อมูลที่ไม่สมดุล สำหรับแบบจำลองความต้องการเป็นเจ้าของรถยนต์-วิธีการเรียนรู้แบบมีค่าใช้จ่าย

บทคัดย่อ

เพื่อทีจะทำนายความต้องการเดินทาง ข้อมูลการครอบครองรถยนต์ส่วนบุคคลของแต่ละครัวเรือนหากเป็นข้อมูลที่ไม่สมดุล เมื่อนำไปใช้เพื่อส่งเสริมนโยบายด้านการขนส่งผ่านแบบจำลองการเรียนรู้ของเครื่องจักร จะส่งผลกระทบต่อประสิทธิภาพการเรียนรู้ของ algorithms ข้อมูลที่ได้จากโครงการศึกษาทางพิเศษในจังหวัดขอนแก่น (ประเทศไทย) ปี 2558 จำนวน 2,015 ครัวเรือน เป็นชุดข้อมูลที่ไม่สมดุล กล่าวคือข้อมูลมีจำนวนสมาชิกของ class คำตอบส่วนน้อย น้อยกว่า class คำตอบทีเหลือ เป็นผลให้เกิดความเอนเอียงในการจำแนกประเภทข้อมูล งานวิจัยนีจึงนำเสนอการปรับข้อมูลให้สมดุลด้วยวิธีการเรียนรู้แบบมีค่าใช้จ่ายก่อนนำไปสร้างแบบจำลอง ทีมี 3 Classes คำตอบ ด้วย decision trees K-Nearest neighbors (kNN) และ Naïve bayes algorithm วิธี k-folds cross-validation ถูกใช้ในการแบ่งข้อมูลเพื่อสร้างและทดสอบประสิทธิภาพแบบจำลอง ค่าความสามารถในการส่งคำตอบกลับ ถูกใช้วัดประสิทธิภาพแบบจำลอง ผลลัพธ์ทีได้ แสดงให้เห็นว่า kNN ให้ประสิทธิภาพการทำนาย class คำตอบส่วนน้อยสูงกว่า algorithm อืน โดยให้ค่าความสามารถในการส่งคำตอบกลับ สำหรับชนิดของพืนทีRural และ Suburban ซึงเป็นชนิดของพืนทีทีมีอัตราส่วนความไม่สมดุลทีแตกต่างกันมากก่อนปรับสมดุลของข้อมูล เท่ากับ 46.9 % และ 46.4 % ในขณะทีภายหลังปรับสมดุลของข้อมูลด้วย Cost matrix (MCN1) ค่า ความสามารถในการส่งคำตอบกลับ เท่ากับ 84.4 % และ 81.4 % ตามลำดับ

คำสำคัญ

ตารางค่าใช้จ่าย; ต้นไม้ตัดสินใจ; K-Nearest neighbors; Naïve bayes; การตรวจสอบไขว้; แบบจำลอง Tour-Based

REFERENCES

[1] Karlaftis MG, Vlahogianni EI. Statistical Methods Versus Neural Networks in Transportation Research: Differences, Similarities and Some Insights. *Transportation Research Part C: Emerging Technologies*. 2011;19(3): 387-399.

[2] Kaewwichian P, Tanwanichkul L, Pitaksringkarn J. Car Ownership Demand Modeling Using Machine Learning: Decision Trees and Neural Networks. *International Journal of GEOMATE*. 2019;17(62): 219-230.

[3] Flach P. *Machine Learning: The Art and Science of Algorithms that Make Sense of Data.* Cambridge University Press; 2012.

[4] Chawla NV. Data Mining for Imbalanced Datasets: An Overview. In*: Data Mining and Knowledge Discovery Handbook*. Springer; 2009. p. 875-886.

[5] Longadge R, Dongre S. Class Imbalance Problem. In: *Data Mining Review*. arXiv Preprint; 2013.

[6] Branco P, Torgo L, Ribeiro RP. A Survey of Predictive Modeling on Imbalanced Domains. *ACM Computing Surveys (CSUR)*. 2016;49(2): 1-50.

[7] Mazurowski MA, Habas PA, Zurada JM, Lo JY, Baker JA, Tourassi GD. Training Neural Network Classifiers for Medical Decision Making: The Effects of Imbalanced Datasets on Classification Performance. *Neural*

Networks. 2008;21(2-3): 427-436.

[8] Gu Q, Cai Z, Zhu L, Huang B. *Data Mining on Imbalanced Data Sets*. Paper presented at the 2008 International Conference on Advanced Computer Theory and Engineering; 2008.

[9] López V, et al. Analysis of Preprocessing vs. Cost-Sensitive Learning for Imbalanced Classification: Open Problems on Intrinsic Data Characteristics. *Expert Systems with Applications*. 2012;39(7): 6585-6608.

[10] Pamuła T. Neural Networks in Transportation Research–Recent Applications. *Transport Problems*. 2016;11.

[11] Sun Y, Wong AK, Kamel MS. Classification of Imbalanced Data: A Review. *International Journal of Pattern Recognition and Artificial Intelligence.* 2009;23(4): 687-719.

[12] Ling CX, Sheng VS. Cost-Sensitive Learning and the Class Imbalance Problem. *Citeseer.* 2008: 231-235.

[13] Maloof MA. Learning When Data Sets Are Imbalanced and When Costs Are Unequal and Unknown. In: *ICML-2003 Workshop on Learning from Imbalanced Data Sets II*; 2003.

[14] Weiss GM, McCarthy K, Zabar B. Cost-Sensitive Learning vs. Sampling: Which is Best for Handling Unbalanced Classes with Unequal Error Costs? *DMIN*. 2007;7(35-41): 24.

[15] Xie C, Lu J, Parkany E. Work Travel Mode Choice Modeling with Data Mining: Decision Trees and Neural Networks. *Transportation Research Record*. 2003;1854(1): 50-61.

[16] He H, Garcia EA. Learning from Imbalanced Data. *IEEE Transactions on Knowledge and Data Engineering*. 2009;21(9): 1263-1284.

[17] Galar M, et al. A Review on Ensembles for the Class Imbalance Problem: Bagging, Boosting, and Hybrid-Based Approaches. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*. 2011;42(4): 463-484.

[18] García S, Herrera F. Evolutionary Undersampling for Classification with Imbalanced Datasets: Proposals and Taxonomy. *Evolutionary Computation*. 2009;17(3): 275-306.

[19] Rivas-Perea P, et al. Lp-SVR Model Selection Using an Inexact Globalized Quasi-Newton Strategy. *Journal of Intelligent Learning Systems and Applications*. 2013;5(1): 19-28.

[20] Biagioni JP, et al. Tour-Based Mode Choice Modeling: Using an Ensemble of (Un-) Conditional Data-Mining Classifiers. In: *88th Annual Meeting of the Transportation Research Board.* Washington, DC; 2008.

[21] Domingos P. Metacost: A General Method for Making Classifiers Cost-Sensitive. *Proceedings of the fifth ACM SIGKDD international conference on Knowledge discovery and data mining;* 1999. p. 155-164.

[22] Liu Z, Guo J, Cao J, Wei Y, Huang W. A Hybrid Short-term Traffic Flow Forecasting Method Based on Neural Networks Combined with K-Nearest Neighbor. *Promet – Traffic&Transportation.* 2018;30(4): 445-456.

[23] Zhao H, Sun D, Zhao M, Cheng S. A Multi-Classification Method of Improved SVM-based Information Fusion for Traffic Parameters Forecasting. *Promet – Traffic&Transportation.* 2016;28(2): 117-124.

[24] Zhang Y, Xie Y. Travel Mode Choice Modeling with Support Vector Machines. *Transportation Research Record*. 2008;2076(1): 141-150.

[25] Wu J, Yang M, Rasouli S, Cheng L. Investigating Commuting Time Patterns of Residents Living in Affordable Housing: A Case Study in Nanjing, China. *Promet – Traffic&Transportation.* 2019;31(4): 423-433.

[26] Wets G, Vanhoof K, Arentze T, Timmermans H. Identifying Decision Structures Underlying Activity Patterns: An Exploration of Data Mining Algorithms. *Transportation Research Record*. 2000;1718(1): 1-9.

[27] Cover T, Hart P. Nearest Neighbor Pattern Classification. *IEEE Transactions on Information Theory.* 2006;13(1): 21-27.

[28] Agarwal Y, Poornalatha G. *Analysis of the Nearest Neighbor Classifiers: A Review.* Paper presented at the Advances in Artificial Intelligence and Data Engineering, Singapore; 2021.

[29] Mani I, Zhang I. kNN Approach to Unbalanced Data Distributions: A Case Study Involving Information Extraction. *Proceedings of Workshop on Learning from Imbalanced Datasets*; 2003.

[30] Batista GE, Prati RC, Monard MC. A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. *ACM Sigkdd Explorations Newsletter*. 2004;6(1): 20-29.

[31] Vit C. Comparative Study on Classification Algorithms. *International Journal of Pure and Applied Mathematics.* 2018;118(24).

[32] Agrawal R. Predictive Analysis of Breast Cancer Using Machine Learning Techniques. *Ingeniería Solidaria.* 2019;15(3): 1-23.

[33] Zhang S, Li X, Zong M, Zhu X, Wang R. Efficient kNN Classification with Different Numbers of Nearest Neighbors. *IEEE Transactions on Neural Networks and Learning Systems.* 2018;29(5): 1774-1785.

[34] Buda M, Maki A, Mazurowski MA. A Systematic Study of the Class Imbalance Problem in Convolutional Neural Networks. *Neural Networks.* 2018;106: 249-259.

[35] Napierala K, Stefanowski J. Types of Minority Class Examples and Their Influence on Learning Classifiers from Imbalanced Data. *Journal of Intelligent Information Systems*. 2016;46(3): 563-597.