A New Approach to Estimating Destinations in Open Automated Fare Collection Systems based on errors-against-errors strategy

Mostafa Shafaati[1], Mahmoud Saffarzadeh*[2]

[1] PHD candidate, Department of Civil and Environment Engineering, Tarbiat Modares University, Tehran, Iran, email: (mostafa.shafaati@modares.ac.ir), Phone: +989127326455

[2] Professor, Faculty of Civil and environment Engineering, Tarbiat Modares University, Tehran, Iran, email: (saffar_m@modares.ac.ir), Phone: +989121125096

Abstract

In transit systems, automatic fare collection systems (AFCs) are widely used. Passengers are often required to use their smart cards only when entering stops, so their destination is unknown. Methods have been proposed for addressing the problem, but most of those require network-level AFC data. The problem remains unresolved when only one line's AFC data is available. This paper tries to solve this issue for specific applications, like crowding-related problems such as calculating perceived travel times. In our method, rather than minimizing errors, the model is constructed so that desirable errors are produced to counter undesirable errors. The task is accomplished by employing an imbalanced binary class classification based on thresholding for each stop. A classification indicates whether a passenger is alighting or has already alighted at the study or previous stops. Although the model may produce incorrect predictions for a particular stop, it will be adjusted to make a deliberate error: for every incorrect prediction of alighting, there will be a few incorrect predictions of not alighting. Using this technique, we estimate how many passengers are on board the bus. Our model has the functionality of an Automatic Passenger Counting (APC) system when the line does not have one.

Keywords: Public Transportation, Estimating Destinations, Automatic Fare Collection, Binary Imbalanced Classification, Thresholding Method

*Corresponding Author: Mahmoud Saffarzadeh

1. Introduction

There has been an increase in the use of Automated Fare Collection (AFC) systems in public transportation in recent years [1, 2]. Generally, smartcard data extracted from the AFC in public transportation can be used for a number of purposes, including recognizing human mobility patterns [3] and etc. These systems can be either open or closed. Open systems only record boarding stops, while closed systems record both boarding and alighting [4]. Open systems are the most widely-used AFC system around the world [5] which comes with a disadvantage — information on destination stops is non-existent and thus presents a challenge for fare collection. Therefore, many researchers attempt to calculate OD matrices in such systems [6-9].

The solution is often based on trip-chaining theory [10], and then the transactions for which trip-chaining models cannot be applied (often due to only one transaction on the day of study) are obtained using machine learning models or other methods. It cannot be denied that these methods (which will be reviewed in section 2) are helpful. However, most of these approaches have only been proposed at the network level. A problem will arise when only one line's AFC data is accessible and the Automated Passenger Counting (APC) data is unavailable. It is the case in Tehran's Bus Rapid Transit (BRT), where stops and buses do not have APC systems, and the AFC systems are open. It is difficult to find studies that have offered a solution to such a problem. Therefore, in this study, the main contribution is to propose a method to estimate the destination stop in open AFC systems when only one line's AFC is available. We can accomplish this task using simple binary logistic models. The binary logistic models are used in conjunction with thresholding technique in this paper. The proposed method is more beneficial for certain purposes such as when one needs crowding levels and the number of passengers in each vehicle. Our proposed solution may also be considered a heuristic. Heuristic solutions have been found in various transportation studies [11, 12], and etc. Continuing with the paper, we will review the most important papers concerning the research issue in the second section, explain the methodology in the third section, discuss the model's results in the fourth section, and conclude the study in the last section.

2. Literature Review

We emphasize that finding studies that attempt to estimate boarding stops is difficult since boarding stops are recorded in many transit systems [13-20].

To estimate alighting locations, trip-chaining is the most commonly used method in the literature [21]. Barry et al. [22] were the first to propose this method. By considering a one-month period rather than a single day, Trepanier et al. [23] contributed to the development of the method. By assuming 3 a.m. as the time when the virtual day begins, Barry et al. [24]

developed the concept of a virtual day to solve the problem. The concept was also addressed by [13, 25], which started the virtual day at 4 and 5 a.m., respectively. It was possible to estimate more destinations and minimize the number of trips involving only one transaction with the help of this new concept.

Others have also used a method that considers the day after the trip day [26]. Using data related to 5 working days, [16] developed a more sophisticated model that identifies the source of errors. In the literature, there is a great deal of documentation on network and route level analysis with APC, AVL (Automated Vehicle Location), and AFC. Regarding route level analysis, Iterative Proportional Fitting (IPF) is one example that was used by [27, 28]. The method, however, requires too much information to operate correctly, whereas there are times when we only have a single piece of information, such as the AFC data. A method based on information theory has been proposed by [29]. They solve an optimization problem in which entropy is maximized. A disadvantage of this model is that it assumes all stops will have the same probability of passengers alighting, while in reality some stops may have a higher probability of passengers alighting. Moreover, the AFC data may have to be accompanied by other data such as total person-kilometer, Stated Preference (SP), or Revealed Preference (RP) surveys. Using maximum likelihood estimation, Lu [30] solved the problem using APC data, rather than AFC. Wang [31] utilized the trip-chaining method to determine the alighting stops at the network level. Surveys were also used to validate the results. Gordon [32] improved the method so that it could be applied to passengers who do not use smart cards. In accordance with the findings of [33], the final alighting stop can be considered to be the first boarding stop of the day. Furthermore, some parts of the algorithm were modified to reduce the average distance between actual and estimated alighting stops. As a result of their proposed improvement, they claimed that this average distance would be reduced from 806 meters to 530 meters. Alsger et al. [33] share the same assumption with [17, 18, 24, 27, 34].

Machine learning and deep learning models have been used in some studies. As an example, Cheng et al. [35] have improved their accuracy by 2% by using Latent Dirichlet Allocation. The two-hidden-layer rectified linear unit was used by [36] based on supervised machine learning. In addition, there are those who have combined the trip-chaining method with machine learning models. Yan et al. [37] based their machine learning model on the trip-chaining results of their first step. Based on neural networks, Assemi et al. [38] accomplished the same task. We note that similar to the last two studies, the present study will use trip-chaining as the initial step, and then a machine-learning model will be developed based on the results of trip-chaining. The accuracy of most alighting-related studies ranges from 75 to 96 percent [21].

By using the daily boarding data, Ozgun et al. [39], calculated alighting counts for round trip lines that are balanced in daily passenger counts on both forward and backward routes. The boarding patterns of each line are used to determine vehicle occupancy levels on a trip basis. By utilizing the smartcard data of the Melbourne transit network, Hamedmoghadam et al. [40] proposed a method based on statistical pattern recognition. Their model can provide acceptable accuracy at the network level. Jin et al. [41] employed a spatio-temporal, distance decay, and built environment constraint approach to infer alighting stops. Despite the fact that bus stop and zone level data are used for validation purposes in this paper, the authors emphasize that their methods are based upon a network approach. Cerqueira et al. [42] provide a comparison between the principles, based on which different models have been developed to estimate alighting stops in public transportation. A multimodal transport network was used as the source of data for this study. Based on the estimated number of alighted stops in the paper, a confidence measure is proposed. Studies have been conducted using simpler methods. According to [43], in their study of crowding on public transportation users, they first estimated the destinations by using trip-chaining method. Based on the percentage of each origin-destination out of all ODs, the one-transactions were distributed randomly among the different origin-destinations (ODs) obtained by trip-chaining. The same method was used to estimate the destinations in another study in which the Experienced Service Reliability Gap (ESRG) was the main focus [44]. According to another network-level study [45], passengers can be grouped into seven different types to estimate destinations more effectively. Using Mobility Knowledge Graphs for obtaining destinations in open smart card systems is suggested by [46].

Based on a review of the literature, we address a critical gap in this study: estimating the destinations for one-transaction trips for one line with only AFC data available. A new method will be presented in this study in order to achieve this objective. The task will be accomplished using the following logic: When minimizing errors is not possible, neutralize the errors rather than minimizing them. To achieve our objective, we will employ imbalanced binary class classification. Using the thresholding method, classification problems will be solved. Based on our proposed logic, we will calculate the best threshold.

3. Methodology

We will describe the proposed method in detail in this section using the dataset that is available. The framework is also illustrated in figure 1.

3.1. Preparing The Raw Dataset

An example of raw AFC data is shown in Table 1. Serial is the serial number for the card, Code indicates the type of card (student cards, journalist cards, ordinary cards with a code of 132, etc.), Line indicates the code of the studied line, date indicates the date of the card's use, time indicates the time of day when the card is used, and reader indicates the reader number. The reader's code ranges from 1001 to 1099 in this line. Specifically, the data pertains to the period from November 23rd to 27th, 2019 (equal to five working days in Iran, with each working week beginning on Saturday and ending on Wednesday) and to the first BRT line in Tehran, the capital of Iran. Line 1 is the most populated line in Tehran, serving more than 90,000 transactions per day on an ordinary day (i.e. when there are no disruptions, such as during Covid19). Aside from identifying the origin of the line, using this database also other characteristics, including whether there is a subway station nearby (with a maximum distance of 200 meters) and whether another BRT line intersects the origin can be obtained. For the westbound direction, the proposed methodology will be applied. The transactions were recorded from 5 a.m. to 10 p.m.

Data cleaning and preprocessing are the first steps, followed by transactions related to IDs that have more than one transaction per day. This method requires that the trip-chaining theory for one lane be applied, which means the first stop of the day at which the smart card is used will serve as the origin, and the second stop will serve as the destination. Except for stops 3 or 4, where eastbound and westbound readers were separated, most readers in this line were used by passengers in both directions. For simplicity's sake, those stops have been ignored. The direction of trips cannot, therefore, be determined. The assumption leads to more trips being estimated using a trip-chaining-based approach similar to that used by [47]. As a way of clarifying the concept, one can imagine that an ID is recorded at stops 5, 10, and 20 in the morning, afternoon, and evening, respectively. According to this example, three trips have been made if the direction is unknown: starting at stop 5 and ending at stop 10, traveling from 10 to 20, and traveling from 20 to 5. As explained by [33], another important assumption is that each passenger returns to their first origin stop on the day of study. Trip-chaining was used to estimate all these observations, which means that the origins and destinations were known in all cases. Data have now been prepared for the development of our proposed model.

Also, we present the data used in this paper in table 2. All variables are categorical. “Dstop” is the destination stop, “Ostop” is the origin, “Time” indicates the time interval during which the transaction was conducted, “metroOrigin” indicates whether a subway station is located near the stop, and “SecOrigin” indicates whether the stop is located at the intersection of two BRT lines.

Our proposed method focuses on predicting the number of passengers alighting the vehicle at each stop rather than the passengers themselves, since the most critical factor in studies related to crowding levels is the number of passengers in each vehicle. In the case of boarding at stop 1 and alighting at stop 5, and boarding at stop 1 and alighting at stop 10, the prediction might be that passenger 1 alighted at stop 10 and passenger 2 alighted at stop 5. This error will be explained in more detail later in the article.

3.2. Theoretical Bases of the Method

In our method, we break the problem down into several binary imbalanced classifications rather than solving a multiclass classification problem. When the number of stops is 27, there are 26 imbalanced binary classifications in which the following question must be answered for each passenger:

- Does this particular passenger alight in this or previous stops?

In light of the answer to that question being either Yes or No, a binary class classification model must be developed for each destination stop. If the answer is Yes, then the observation belongs to the Positive class, otherwise it belongs to the Negative class. Equation 1 is the confusion matrix for a binary-class classification model:

$$CM = \begin{pmatrix} TN & FN \\ FP & TP \end{pmatrix} \qquad [1]$$

Table 3 shows the elements of equation 1 and their definitions in this study.

FN and FP together are incorrect predictions by the model. The model's errors are caused by these factors. However, these two sources of error have one important distinction: FP indicates that the passenger has not alighted at that particular stop or the previous stops, whereas the model indicates that alighting has taken place. In the event that the prediction is a subset of the FN set, then the model indicates that alighting has not occurred while in actuality it has occurred. For our proposed method, we aim to develop models such that these sources of error are neutralized. As a matter of fact, the method is designed to minimize equation 2:

$$A = | FN - FP | \qquad [2]$$

Where: FN = False Negative, FP = False Positive

We may be inclined to be conservative when solving crowding-related problems, therefore, FN may be chosen to be slightly greater than FP. In other words, it may be desirable to make the model yield a higher result than the actual number of passengers. The threshold-moving or thresholding approach should be utilized for all manipulations concerning equation 2.

3.3. The Threshold-Moving Approach

In this section, threshold-moving is explained briefly. Those interested in the topic can see [6, 48-50] for further information.

First, some metrics need to be introduced:

$$\mathrm{Re}call = \frac{TP}{TP + FN} \qquad [3]$$

$$Specifity = \frac{TN}{FP + TN} \qquad [4]$$

$$G\_mean = \sqrt{\mathrm{Re}call \times Specifity} \qquad [5]$$

$$\mathrm{Pr}ecision = \frac{TP}{TP + FP} \qquad [6]$$

$$TruePositiveRate(TPR) = \mathrm{Re}call \qquad [7]$$

$$FalsePositiveRate(FPR) = 1 - \frac{1}{\mathrm{Re}call} \qquad [8]$$

$$F_{1score} = 2\frac{precision \times recall}{precision + recall} \qquad [9]$$

We have already defined the variables in equations 3 to 9.

A threshold-moving approach is used for dealing with classification problems involving imbalanced classes. This approach changes the decision threshold. The most challenging aspect may be the choice of a new threshold. ROC curves and ROC scores can be used to compare different models and determine the best one. Alternatively, precision-recall may be calculated and precision-recall curves may be plotted. The latter approach aims to achieve a balance between precision and recall. As a result of each method, the decision threshold is shifted to solve the classification problem. "Threshold-moving," "threshold-tuning," or "thresholding" are all terms that describe this process. It should be mentioned that one way to find the best threshold is to maximize G-mean (equation 5) too.

The first method would be to use a ROC curve, and the second method would be to use a precision-recall curve. ROC curves are diagnostic plots that evaluate the probability predictions made by a model on a test dataset [51]. In order to interpret the TPR and FPR of predictions on the positive (minority) class, a series of increasing threshold values is plotted. FPR is plotted on the x-axis and TPR is plotted on the y-axis in this graph. This plot is referred to as a Receiver Operating Characteristic (ROC) curve. In the diagram, a diagonal line indicates a model with no skill (predicts the majority class in every instance), while a point at the top left indicates a model with perfect skills. To assess the trade-off between different thresholds, the ROC curve can be used as a diagnostic tool. In accordance with the

ROC curve, 0.207 is the optimal threshold. In Figure 2, we can see the ROC curve for the binary model of stop 13 and where the best point in our data can be found.

On the other hand, precision-recall curves focus on the performance of a classifier only with respect to the positive class [50]. Precision-recall curves are calculated for probability predictions by creating crisp class labels and calculating precision and recall for each threshold. The thresholds are arranged ascendingly based on a line plot with recall on the x-axis and precision on the y-axis. As shown in figure 3, there is a horizontal line representing a no-skill model, whose precision is determined by the ratio of positive examples in the dataset (for example, TP / (TP + TN)). A dot appears in the upper right corner of the perfect skill classifier, which indicates its full precision and recall. The precision-recall curve for our data is shown in Figure 3. Based on this curve, the best threshold was determined to be 0.271. In this method, in fact F1_score is maximized.

In imbalanced classification, Geometric Mean, or G-Mean, is a metric designed to achieve a balance between specificity and recall. The threshold with the highest G-Mean would be selected by testing each threshold returned by ROC curves on the model. As we already calculated the recall (TPR) and complement to the specificity when calculating the ROC curve, we can calculate the G-Mean for each threshold directly. It was necessary to test all well-known methods to begin our first attempt at minimization of "A" (equation 2). Since none of these methods were invented to minimize "A" (equation 2), we were obliged to test all well-known methods first. When different thresholds were tested without the use of these methods, and the best threshold was compared based on the logic presented in this paper, it was found that the threshold that has the highest value is closer to the optimal threshold to achieve the objective of this paper among the thresholds derived from maximizing G-mean, maximizing F-score, and ROC curve. Typically, the probability threshold for binary classification problems is 0.5. However, the threshold can be adjusted either upwards or downwards in the thresholding approach. As an example, if the threshold is set at 0.3, all observations with predicted probabilities greater than 0.3 are considered positive, and the others are considered negative. Figures F1 to F3 (in the supplementary data file) illustrate all the plots that are necessary for estimating the best thresholds. The threshold provided by these plots may be adequate for the first attempt, but none of them would provide the optimal threshold for this study.

3.4. Preprocessing

Preprocessing was required for this dataset as well, due to the fact that many duplicate transactions were recorded at the same point in time and stop. It was assumed that these were caused by equipment failures or by passengers misusing their cards more than once when entering the stop. Preprocessing includes the detection and deletion of anomalies in this study. A total of 166,866 observations were considered for the building of the models after preprocessing.

3.5. Partitioning Data to Train, Validation, and Test Sets

To validate the proposed method, 20000 observations were randomly selected before training began. Subsequently, those observations were removed, and the models were developed using 60% and 40% of the new dataset as train and test subsets, respectively. On the basis of our logic, the best thresholds were selected. The validation set was used to validate the entire model. Another validation set consisting of 20000 observations was randomly selected too. With the same threshold obtained with the first training and testing set, the second sets of models were developed and validated again using the new 20000 observations. As a matter of fact, our model was validated twice. It began with stop 7 and ended with stop 20 in the learning process. The learning process refers to the process of training the models. Additionally, the purpose of starting from stop 7 is to summarize the paper. The training could have been started at stop 2, but it would have made the paper longer. Further, the calculations and methods are the same for every single stop.

4. Results, Validation, and Discussion

In this section, the results and three methods of validation are presented and explained in detail.

4.1. Calculation Results

Detailed results of the study are presented in table 4. This table summarizes the results of calculating the best threshold. As an example, we will only discuss the results of one stop (stop No. 13) in table 5. For estimating the first attempt to obtain the best threshold, we used the F-score, recall (ROC curve), and G-score. Liblinear solver was used to solve the logit models. To clarify the solution, Tables 5 and 6 present the results of the calculations for stop 13. The greatest value found in the ROC curves, precision-recall plots, and maximizing G_mean can be considered a good starting point for our first attempt. To obtain the best threshold in table 5, we tested the different thresholds using an attempt-and-fail approach. Starting with the greatest threshold value in Table 4, which is 0.271, the attempt-and-fail process begins. Using the previous threshold plus 0.01 in each step, the new threshold is calculated until "A" approaches 0 and then becomes negative. Therefore, the best threshold for stop 13 is between 0.371 and 0.381. For improving the reliability of our model, we should adjust the threshold so that FN exceeds FP by a small margin, so 0.371 would be the appropriate threshold to use.

We emphasize that the paper's strategy accounts for the close proximity of A to 0. Based on the logic of this paper, “A” is the absolute value of the model's errors for each model, and, since the objective of this paper is to make the errors fight against each other, “A” must be close to zero. Additionally, this paper may be criticized for using only one model. The purpose of this work is not to compare the performance of different machine learning models, but to propose a strategy and utilize threshold-moving to complete the task. We are not concerned with the model itself. Further, as long as threshold-moving is an option, it appears that using more advanced models than logit is not necessary. Additionally, advanced models may be computationally intensive. As a result, logit seems to be an appropriate means of achieving our objectives. It may be necessary to provide the following explanations regarding thresholding and its application in this paper. If the typical threshold is equal to 0.5, and the minority class is the positive one and the majority is the negative one, then decreasing classification threshold would lead to the goal of this paper. Reducing the threshold from 0.5 will increase both TPs (in which the predicted and actual destinations are the same) and FNs (increasing the chance of not alighting at the studied stop in order to improve the reliability of crowding-level analysis). Furthermore, such a reduction results in lower TNs (decreasing the number of persons not alighting at the studied stop for enhancing reliability in crowding level analysis problems) and FPs (decreasing the number of persons alighting at the studied stop for enhancing reliability in crowding level analysis problems). In the case where majority class is positive and minority class is negative, increasing threshold from 0.5 results in increasing TNs, FNs, and decreasing FPs, and TPs. Again, this is exactly what this paper is intended to accomplish. Generally, when the minority is the positive class, and the typical threshold is 0.5, reducing the threshold will favor us, and when the minority is the negative class, increasing the threshold will be helpful. This is what exactly occurs in this research. As shown in table 4, the best threshold for models in which the negative class is the majority is less than 0.5, and for models in which the positive class is the majority, the best threshold takes a value greater than 0.5.

4.2. Validating The Method

Three validation approach were used. According to our knowledge, the first is commonly used, while the second and third are new.

4.2.1. The First Validation Approach: Prediction with Accepted Error

The two validation sets observations were used for validation in this subsection. In the first approach, it is assumed that the model is correct if the predicted stop matches the actual stop. Figures 4 and 5 illustrate the results of this approach. As a reminder, the method described in this paper was designed for problems in which estimating the number of passengers in public transportation vehicles is of vital importance. In the case of analyzing crowding-related problems on a particular line of transit, this would be the case. The purpose of the research should not be to predict the exact location at which passengers will alight. In this subsection, however, that the accuracy is examined. A noteworthy aspect of this study is the existence of 14 different classification models, all of which attempt to predict whether a particular passenger will alight at a particular stop and the previous stops or not. In this regard, from one point of view, it is essential that each prediction be validated multiple times since, for example, if passenger P alights at stop 12, it will need to pass the validation test for six models (the models for stops 7, 8, 9, 10, 11, 12) in order for the prediction to be accurate. As a result of this logic, Table 7 is produced. Before discussing table 7, the measure “Diff” is introduced in equation 10:

$$\text{Diff} = \text{the estimated stop number} - \text{the actual stop number} \quad [10]$$

Table 7 shows the number of members for each "Diff" for both validation sets. As the models have been developed for stops 7 to 20, and the number of stops is 27, then the minimum value of "Diff" will be -20 (7 - 27), and the maximum value will be +18 (20 - 2). The developed model allows for the use of 13102 observations of the first validation set and 13181 observations of the second validation set (those observations with predicted destinations ranging from 7 to 20 because if the predicted stop is greater than 20, then we don’t know exactly what number it would have taken if the model was developed to cover all stops. It may have taken 21, 22, 23, 24, 25, or 26, thus calculating “Diff” wouldn’t have been possible). Using the second validation set, the number of observations in which the estimated stop is the seventh and the actual stop is the 20th is 64. In both validation samples, Diff equal to 0 has the largest number of members (1321 in the first validation set and 1377 in the second validation set). Accordingly, the greatest number of N is associated with the best performance of each model, where the model has correctly predicted the estimated stops without committing any errors. Also, figures 4, and 5 illustrate table 7 visually.

We introduce three parameters for each validation set before discussing Table 8. First, there is the Accepted Classification Error Categories (ACEC), which displays the category of the selected values of Diff as the accepted error. In the event that Diff is considered to be zero, then the ACEC will be "a". In the event that Diff is accepted as -1, and +1 as well, then ACEC will be "b", etc. Number of Members in the Category (NMC) represents the number of observations in the validation sets that fall into a specified category, while Percentage of Members in the Category (PMC) represents the percentage of observations in each category. It is important to emphasize that PMC can be viewed as the measurement accuracy in the first validation approach. In the event that the selected ACEC is "a", this means that the accepted error is zero stops. As a result, the validation observations in which the predicted and actual stops are the same are taken into account in this case. Based on Table 8, NMCs for ACEC "a" are 1321 and 1371 observations, which means in each validation set, these observations have been correctly estimated, which corresponds to PMC being 10% approximately. If ACEC is “b” or the accepted error is 1 instead of 0, the accuracy of the measurement or PMC would be approximately 25%. It should be mentioned that the purpose of Table 8 is to present a validation solution. The 11th category or ACEC of “l” means that the difference between the predicted destination stop and the real stop is equal to or less than 11, it does not mean that the model predicts whether the passenger will alight at stop 11 or the previous stops. Therefore, there is no relationship between the number of classification models and the categories in table 8. Furthermore, technically there should be 26 models for a 27-stop line, but in order to summarize the already lengthy paper, we did not build and present all models. As the logic is exactly the same in all models, it seems unnecessary to present them all. Table 4 and Figure F1 (in the supplementary data file) illustrate the size of each class. In stop 11 model, the number of members of the positive class (those who alight at stop 11 or the stops preceding this one) is 11590, while the number of members of the negative class is 76529, which account for 16.407 and 83.593 percent, respectively. The number of positive and negative class members at stop 18 is 45808 and 42311, representing 51.984 and 48.016 percent, respectively. In relation to table 8, and the ACEC, we note that those categories reflect the accepted error for the model in the first validation approach. The model's accuracy for both validation samples is almost 25 percent if we accept that the difference between predicted and actual stop numbers is 1 (category "b"). In the case of 2 stops of accepted error (category "c"), the accuracy is nearly 41%, while for 11 stops (category "l"), it is approximately 94%. Therefore, these categories do not have anything to do with the number of members of each class as well. In other words, ACEC does not have any relationships with each class’s size in the classification models for different stops. Table 8 is illustrated visually in Figures 6 and 7. We can see that plots in Figures 6 and 7 are almost similar, but there are some minor differences. As a matter of fact, it appears that the model has almost similar performance in both validation sets. However, due to the fact that our proposed method does not minimize the errors, expecting this method to perform well is not reasonable. As a result, the validation problem should be approached differently.

4.2.2. The Second Validation Approach: Comparing the Number of False Predictions

In the second approach, instead of comparing the predicted to the actual class number, false predicted classifications are compared. Table 7 is considered again, and then two quantities can be calculated using Equations 11 and 12:

$$I_{1i} = \left\{ \begin{array}{l} N_i \text{ if i = 0} \\ N_i - N_{-i} \text{ if i} \neq 0 \end{array} \right\} \quad [11]$$

$$I_{2i} = \frac{N_{-i}}{N_i} \quad [12]$$

Where:

i = |Diff|= (see Table 7),

These two equations will result in Table 9. Summing all I_{1i}s and I_{2i}s, the result for the first validation set would be:

$\sum_{i=0}^{20} I_{1i} = 1276$,$\sum_{i=0}^{18} I_{2i} = 14.82$

And for the second validation set, it would be:

$\sum_{i=1}^{19} I_{1i} = 1289$,$\sum_{i=0}^{18} I_{2i} = 15.19$

I_{1i} average and I_{2i} average for the first validation set would equal 60.76 and 0.78, respectively, when divided by 21 (the number of "i"s). In the second validation set, these quantities are 61.38 and 0.79, respectively. As an important point to mention, I_{2i} may work properly if all stops are included in the model. Based on the logic of this paper, I_{2i} should be equal or greater than 1 but in this paper's examples, I_{2i} values are less than 1 for both validation sets because the models have been developed for stops 7 to 20, which was done for the purpose of summarizing the already lengthy paper. It is possible for this index to be more informative if the models were developed for every destination stop starting at stop 2 and ending at stop 27. Therefore, in this particular example, I_{1i} will be more focused than I_{2i}. According to these numbers, there are more than one prediction in which the actual alighting stop is smaller than the predicted one for every false prediction where the actual alighting stop is greater than predicted. Thus, based on the purpose of this paper, the model appears to be working as intended. It is explained earlier that the purpose of this model is to calculate in-vehicle crowding levels where the number of passengers per bus in each block is crucial. I_{1i} is equal to N_1 minus N_{-1} when i equals 1, where N_1 refers to the number of predicted cases in which the predicted stop number exceeds the actual stop number by 1. It is considered a member of N_1 passengers if, for example, a passenger's actual alighting stop is 15, but the predicted alighting stop is 16. Similarly, if the predicted number for another passenger is 12, but the actual number is 13, then this individual is considered a member of N_{-1}. In the case where N_1 has 100 members and N_{-1} has 95 members, then $I_{1(i=1)}$ will equal five (100 - 95). Thus, I_{1i} works in this manner. It should be remembered that the very basic concept of our proposed model is to produce desirable errors against the undesired errors created by the model. Since it is impossible to prevent the model from making undesirable errors, we adjust the threshold so that the model can make desirable errors. The estimation of in-vehicle crowding levels, especially when calculating perceived travel time in the vehicle based on crowding levels, has a great deal of importance, as shown in [43, 52]. In such an instance, it may be preferable to have an error in the direction of overestimating the passengers in the vehicle as opposed to an error in the direction of underestimating them. Because, if necessary, we can develop strict countermeasures that will prevent crowding if implemented. We therefore choose to have a higher average I_{1i} based on this logic. In light of these explanations, we have acceptable I_{1i} values for both validation samples.

4.2.3. The Third Validation Approach: The Aggregated Approach

Another proposed approach examines the performance of the whole model in relation to the entire set of stops. To accomplish this, two large confusion matrices have been added to the supplementary data file (tables T2 and T3). However, as in the proposed method of this study, increasing or decreasing precision, recall, or any of the other convenient measures are not the objective of our method, it may not be necessary to calculate and report these measures for these two big matrices. It may even be misleading to report those measures for tables T2, and T3 (in the supplementary data file). We can, however, define another validation method using these matrices based on the core idea of our model. If real stop is j, predicted stop should be larger or equal to j. As a compromise, if j-1, and j-2 are also considered acceptable, then the number of desirable predictions should exceed the number of undesirable predictions in each row. It is exactly as expected from our model that when it produces incorrect predictions, the number of predictions in which the predicted stop number exceeds the actual stop number is greater than the number of predictions in which the actual stop number exceeds the predicted stop number. In fact, the model must have a gravitational pull to predict the passenger alighting stop after the real alighting stop, in order to ensure that our analysis will be more reliable in the future. Specifically, if the model predicts a worse situation than the actual situation, then we can ensure that the countermeasures taken to reduce the problem of in-vehicle crowding will have a greater impact from a satisfaction perspective. Nevertheless, it is possible that this strategy will not be efficient from an economic perspective, so when employing the proposed method presented in this paper, good engineering judgement, along with trade-off skills, are essential for each project in which estimating the destination stop in one line based solely on AFC data is required. Tables T2, and T3 (in the supplementary data file) illustrate the desirable predictions in green and red font colors, respectively, while the undesirable predictions are illustrated in black font. If the model is to be validated using the third approach, the inequality 13 which is called "The Last Relation (LR)" in this paper, must be satisfied:

$$\sum_{j=7}^{20}\sum_{k=2}^{j-3} O_{jk} \leq \sum_{j=7}^{20}\sum_{k=j-2}^{27} O_{jk} \qquad [13]$$

Both Tables T2 and T3 (in the supplementary data file) satisfy the LR. Nevertheless, determining the compromised predictions can be a challenging task, which is not the purpose of this study. Future research should address this issue.

4.3. The Model's Feasibility

To the best of our knowledge, there is no study that estimates the destinations using only AFC data of a single line, so comparing this method with other methods proposed in other studies is irrelevant. However, we note that although the proposed solution requires training N-1 models where there are N stops, all trained models are simple binary logistic models. For each stop, the model would run approximately 45.8 seconds if, on average, ten attempts and failures are required to find the best threshold. In the case of 27-stop lines, a total of 26 models need to be developed, which results in a total of 1190.8 seconds or 19.84 minutes using Python. In addition, our method has the advantage of requiring the minimum amount of input information. We observe in table 2 that the only features used in this study are time, origin, destination, date, and whether the stop intersects with another BRT or subway line. Due to the small number of features in the model, it is not computationally intensive. Therefore, it appears that our model has the quality of feasibility. We note that the time taken to import the data was excluded when reporting the running time. The supplementary data file contains all the plots and tables related to the models used in this study as well as other results and almost all of our calculations. We attempted to be as concise as possible in the main body of the paper.

5. Conclusion

This paper focuses on estimating destinations for one transaction observation in an open AFC system when only one line of AFC data is available. To the best of our knowledge, no similar study has been conducted under the same conditions. Logic dictated that if it is impossible to prevent or minimize errors, we should focus on minimizing their damages instead. In order to understand the logic behind such an approach, we need to examine the situation that warrants it. The estimation of crowding levels used to determine the in-vehicle time coefficient and calculating perceived travel time reliability introduced by (Jenelius 2018) have been applied to some research projects and research problems, for example. As well as examining the existence of the "Dynamic Effect of Crowding" introduced by [43] calculating perceived travel time based on in-vehicle crowding levels and determining the alighting stop for a studied Bus Rapid Transit (BRT) line or a bus line is a significant factor to be considered. Solutions can be found in the literature. The complexity still exists, especially when the information required is restricted to only AFC data. Additionally, goal-oriented solutions appear to be a good strategy or at least worthy of discussion. Often, in problems relating to estimating destinations in open AFC systems, the ultimate goal may not be estimating the destination, but rather, it is one step in the solution of a larger issue. The results of this step will be used in subsequent steps or other aspects of a particular problem. It can be argued that the problem can be viewed differently depending on what the purpose of estimation is, that is instead of trying to solve unsolvable problems using complicated methods, it may be possible to find the solution using simpler, more commonly used models. As far as this study is concerned, this approach appears to be effective. Several simple ideas are included in the proposed method, which also prove useful to our problem. In summary, the proposed method consists of the following steps:

Step 1. Before beginning any calculation, it is necessary to perform preprocessing.

Step 2. Using trip-chaining to determine the origins and destinations of passengers who have used their smart cards more than once.

Step 3. The development of n-1 machine learning models or logit for n stops. It is a simple binary imbalanced classification problem for each model.

Step 4. Establishing the best thresholds for each stop’s model. The ideal threshold is one that minimizes "A", which is the difference between FP and FN.

Step 5. Validating the model using the methods described in subsection 4.2. I_{1i} and I_{2i} should have positive averages and greater than 1. Further, in the model's validation step, the LR (Last Relation) must be satisfied. Depending on how conservative the researcher is, the indices and numbers mentioned may differ.

Step 6. Predicting one-transaction trips using the validated model.

We mention that all above steps have been explained, and discussed thoroughly in section 3, and 4. There might be criticism of the proposed method in this paper due to its lack of acceptable precision. Although this criticism may be valid, some points should be taken into account:

1. It would still be impossible to claim that trip-chaining alone could determine all alighting stops with 100% accuracy. Since all methods, including trip-chaining, are based on certain assumptions, approximations are inevitable wherever assumptions exist.

2. Generally, trip-chaining cannot be applied in its entirety since one-transaction cases are common, which necessitates complementary analysis. There is no doubt that the model will perform better if we have enough information, but there are times when our input is limited. In the case where only one line of AFC data is available, for example.

It is true that in our model, everything is somehow approximate, but that is the nature of the problem. It seems that what matters most is to base the approximation on logical assumptions. Although the solution may be less approximate if AVL data is available, a survey is conducted, or the whole network's AFC is available, this study focuses on the issue of only having access to the AFC data for one line. The paper further assumes that the ultimate goal is to analyze crowding in public transportation vehicles, or simply, the number of passengers in the vehicles. It seems that problems like this should be approached with caution so that the outcome will have considered the worst case scenario, thereby increasing the probability of receiving results that are representative of the worst scenario. The question is how much higher the probability of receiving the worst results should be. In order to answer this question, a great deal of trade-off must be made between reducing costs and providing comfort. An overestimation of the number of passengers in public transportation vehicles may lead decision-makers to increase comfort, which in turn may increase costs. A significant underestimation, on the other hand, may reduce both comfort levels and costs. For this reason, finding a specific point and recommending it for all lines with different characteristics is not reasonable when validating the method and proposed measures. Thresholds are also subject to this principle. Every single problem should have its own threshold calculated. With regard to the proposed measures of our method, it is important to standardize I_{1i}, and I_{2i}, which can be an important consideration in future studies. The same applies to all other measures proposed in this study. To determine different levels of these measures and categorizing them into different levels such as "Excellently acceptable", "acceptable", "poorly acceptable", "acceptable under special circumstances", "not acceptable", more studies with smartcard data as well as surveys in different lines, networks, and cities are necessary. It is imperative to consider the economic implications of various countermeasures to alleviate or eliminate crowding when conducting such a study. We emphasize that although, the problem of this paper requires that every single problem and line be considered as a separate issue, it is possible to conduct a large scale study in order to generalize the validation measures that are proposed in this article. In such a study, a variety of countries should be considered. Also, factors such as demand, land use, population, etc., are important too.

Finally, this paper is mainly novel in its proposal of a framework to deal with the issue of estimating destinations in open AFC systems when only a single line's AFC data is available. Model development is based on a strategy in which, rather than minimizing errors, the model produces desirable errors against undesirable errors to the extent that these two types of errors cancel each other out. By using threshold-moving or thresholding, this goal is achieved. We recommend our method to be used when analyzing in-vehicle crowding is of concern which means that, our model is able to fulfill the function of Automatic Passenger Counting (APC) systems where APC is not available. The model has been validated through three approaches, but two of these three approaches appear to be more appropriate in light of the logic behind the model.

The supplementary data is available at:

References

[1] Huang, D., Yu, J., Shen, S., and et al. "A method for bus OD matrix estimation using multisource data", Journal of Advanced Transportation, 2020 (2020) 1-13. https://doi.org/10.1155/2020/5740521

[2] Welch. T.F., Widita. A. "Big data in public transportation: a review of sources and methods", Transport reviews, 39(6) (2019) 795-818. https://doi.org/10.1080/01441647.2019.1616849

[3] Cats, O. "Identifying human mobility patterns using smart card data", Transport Reviews, 44(1) (2024) 213-243. https://doi.org/10.1080/01441647.2023.2251688

[4] Mohammed, M., Oke, J. "Origin-destination inference in public transportation systems: A comprehensive review", International Journal of Transportation Science and Technology, 12(1) (2023) 315-328. https://doi.org/10.1016/j.ijtst.2022.03.002

[5] Li, T., Sun. D, Jing, P., and et al. "Smart card data mining of public transport destination: A literature review", Information, 9(1) (2018) 18. https://doi.org/10.3390/info9010018

[6] Ma, X., Wu, J., Wang, Y., and et al. "Mining smart card data for transit riders' travel patterns", Transportation Research Part C: Emerging Technologies, 36 (2013) 1-12. https://doi.org/10.1016/j.trc.2013.07.010

[7] Kusakabe, T., Asakura, Y. "Behavioural data mining of transit smart card data: A data fusion approach", Transportation Research Part C: Emerging Technologies, 46 (2014) 179-191. https://doi.org/10.1016/j.trc.2014.05.012

[8] A. Bhaskar, Chung, E. "Passenger segmentation using smart card data, IEEE Transactions on intelligent transportation systems", 16(3) (2014) 1537-1548. DOI: 10.1109/TITS.2014.2368998

[9] Ma, X., Liu, C., Wen, H., and et al. "Understanding commuting patterns using transit smart card data", Journal of Transport Geography, 58 (2017) 135-145. https://doi.org/10.1016/j.jtrangeo.2016.12.001

[10] Lei, D., Chen, X., Cheng, L., and et al. "Minimum entropy rate-improved trip-chain method for origin–destination estimation using smart card data", Transportation Research Part C: Emerging Technologies, 130 (2021) 103307. https://doi.org/10.1016/j.trc.2021.103307

[11] Mahpour, A., Amiri, A.M., M. Deldar, M., and et al."A heuristic method to determine traffic bottlenecks based on ant colony: a case study of Iran", Case Studies on Transport Policy, 6(4) (2018) 716-721. https://doi.org/10.1016/j.cstp.2018.09.001

[12] Bonassa, A.C., da Cunha, C.B., Isler, C.A."A multi-start local search heuristic for the multi-period auto-carrier loading and transportation problem in Brazil, European Journal of Operational Research", 307(1) (2023) 193-211. https://doi.org/10.1016/j.ejor.2022.08.036

[13] Nunes, A.A., Dias, T.G., e Cunha, J.F. "Passenger journey destination estimation from automated fare collection system data using spatial validation", IEEE transactions on intelligent transportation systems, 17(1) (2015) 133-142. DOI: 10.1109/TITS.2015.2464335

[14] Alsger, A.A., Mesbah, M., Ferreira, L., and et al. "Use of smart card fare data to estimate public transport origin–destination matrix", Transportation Research Record, 2535(1) (2015) 88-96. https://doi.org/10.3141/2535-10

[15] Nassir, N., Hickman, M., Ma, Z. “Activity detection and transfer identification for public transit fare card data”, Transportation, 42 (2015) 683-705. https://doi.org/10.1007/s11116-015-9601-6
[16] Jun, C., Dongyuan, Y. “Estimating smart card commuters origin-destination distribution based on APTS data”, Journal of Transportation Systems Engineering and Information Technology, 13(4) (2013) 47-53. https://doi.org/10.1016/S1570-6672(13)60116-6
[17] Li, D., Lin, Y., Zhao, X., and et al. “Estimating a transit passenger trip origin-destination matrix using automatic fare collection system, ” in: Database Systems for Adanced Applications: 16th International Conference, DASFAA 2011, International Workshops: GDB, SIM3, FlashDB, SNSMW, DaMEN, DQIS, Hong Kong, China, April 22-25, 2011. Proceedings 16, Springer, 2011, pp. 502-513. https://doi.org/10.1007/978-3-642-20244-5_48
[18] Nassir, N., Khani, A., Lee, S.G., and et al.“Transit stop-level origin–destination estimation through use of transit schedule and automated data collection system”, Transportation research record, 2263(1) (2011) 140-150. https://doi.org/10.3141/2263-16
[19] Chu, K.K.A., Chapleau, R. “Enriching archived smart card transaction data for transit demand modeling”, Transportation research record, 2063(1) (2008) 63-72. https://doi.org/10.3141/2063-08
[20] Hofmann, M., O'Mahony, M. “Transfer journey identification and analyses from electronic fare collection data”, in: Proceedings. 2005 IEEE Intelligent Transportation Systems, 2005., IEEE, 2005, pp. 34-39. DOI: 10.1109/ITSC.2005.1520156
[21] Hussain, E., Bhaskar, A., Chung, E. “Transit OD matrix estimation using smartcard data: Recent developments and future research challenges”, Transportation Research Part C: Emerging Technologies, 125 (2021) 103044. https://doi.org/10.1016/j.trc.2021.103044
[22] Barry, J.J., Newhouser, R., Rahbee, A., and et al.“Origin and destination estimation in New York City with automated fare system data”, Transportation Research Record, 1817(1) (2002) 183-187. https://doi.org/10.3141/1817-24
[23] Trépanier, M., Tranchant, N., Chapleau, R. “Individual trip destination estimation in a transit smart card automated fare collection system”, Journal of Intelligent Transportation Systems, 11(1) (2007) 1-14. https://doi.org/10.1080/15472450601122256
[24] Barry, J.J., Freimer, R., Slavin, H., “Use of entry-only automatic fare collection data to estimate linked transit trips in New York City”, Transportation research record, 2112(1) (2009) 53-61. https://doi.org/10.3141/2112-07
[25] Munizaga, M., Devillaine, F., Navarrete, C., and et al.“Validating travel behavior estimated from smartcard data”, Transportation Research Part C: Emerging Technologies, 44 (2014) 70-79. https://doi.org/10.1016/j.trc.2014.03.008
[26] Kumar, P., Khani, A., He, Q. “A robust method for estimating transit passenger trajectories using automated data”, Transportation Research Part C: Emerging Technologies, 95 (2018) 731-747. https://doi.org/10.1016/j.trc.2018.08.006
[27] Cui, A. “Bus passenger origin-destination matrix estimation using automated data collection systems”, Massachusetts Institute of Technology, 2006. http://hdl.handle.net/1721.1/37970
[28] Ji, Y., Mishalani, R.G., McCord, M.R. “Estimating transit route OD flow matrices from APC data on multiple bus trips using the IPF method with an iteratively improved base: Method and

empirical evaluation", Journal of Transportation Engineering, 140(5) (2014) 04014008. https://doi.org/10.1061/(ASCE)TE.1943-5436.0000647

[29] Ait-Ali, A., Eliasson, J. "Dynamic Origin-Destination Estimation Using Smart Card Data: An Entropy Maximisation Approach", arXiv preprint arXiv:1909.02826, (2019). https://doi.org/10.48550/arXiv.1909.02826

[30] Lu, D. "Route level bus transit passenger origin-destination flow estimation using APC data: Numerical and empirical investigations", The Ohio State University, 2008. http://rave.ohiolink.edu/etdc/view?acc_num=osu1228268640

[31] Wang, W. "Bus passenger origin-destination estimation and travel behavior using automated data collection systems in London, UK", Massachusetts Institute of Technology, 2010. http://hdl.handle.net/1721.1/60814

[32] Gordon, J.B. "Intermodal passenger flows on London's public transport network: automated inference of full passenger journeys using fare-transaction and vehicle-location data", Massachusetts Institute of Technology, 2012. http://hdl.handle.net/1721.1/78242

[33] Alsger, A., Assemi, B., Mesbah, M., et al. "Validating and improving public transport origin–destination estimation algorithm using smart card fare data", Transportation Research Part C: Emerging Technologies, 68 (2016) 490-506. https://doi.org/10.1016/j.trc.2016.05.004

[34] Farzin, J.M. "Constructing an automated bus origin–destination matrix using farecard and global positioning system data in Sao Paulo, Brazil", Transportation research record, 2072(1) (2008) 30-37. https://doi.org/10.3141/2072-04

[35] Cheng, Z., Trépanier, M., Sun, L. "Probabilistic model for destination inference and travel pattern mining from smart card data", Transportation, 48(4) (2021) 2035-2053. https://doi.org/10.1007/s11116-020-10120-0

[36] Jung, J., Sohn, K. "Deep-learning architecture to forecast destinations of bus passengers from entry-only smart-card data", IET Intelligent Transport Systems, 11(6) (2017) 334-339. https://doi.org/10.1049/iet-its.2016.0276

[37] Yan, F., Yang, C., Ukkusuri, S.V. "Alighting stop determination using two-step algorithms in bus transit systems", Transportmetrica A: Transport Science, 15(2) (2019) 1522-1542. https://doi.org/10.1080/23249935.2019.1615578

[38] Assemi, B., Alsger, A., Moghaddam, M., and et al. "Improving alighting stop inference accuracy in the trip chaining method using neural networks", Public Transport, 12(1) (2020) 89-121. https://link.springer.com/article/10.1007/s12469-019-00218-9

[39] Özgün, K., Günay, M., Başaran, D., and et al. "Estimation of alighting counts for public transportation vehicle occupancy levels using reverse direction boarding", Journal of Public Transportation, 25 (2023) 100070. https://doi.org/10.1016/j.jpubtr.2023.100070

[40] Hamedmoghadam, H., Vu, H.L., Jalili, M., and et al. "Automated extraction of origin-destination demand for public transportation from smartcard data with pattern recognition", Transportation Research Part C: Emerging Technologies, 129 (2021) 103210. https://doi.org/10.1016/j.trc.2021.103210

[41] Jin, M., Wang, M., Gong, Y., and et al.“Spatio-temporally constrained origin–destination inferring using public transit fare card data”, Physica A: Statistical Mechanics and its Applications, 603 (2022) 127642. https://doi.org/10.1016/j.physa.2022.127642
[42] Cerqueira, S., Arsenio, E., Henriques, R. “Is there any best practice principles to estimate bus alighting passengers from incomplete smart card transactions? ”, Transportation Research Procedia, 72 (2023) 3395-3402. https://doi.org/10.1016/j.trpro.2023.11.780
[43] Shafaati, M., Saffarzadeh, M. “Does Crowding Have a More Complicated Effect on Public Transport Users with Respect to Perceived Travel Time? ”, Transportation Research Record, (2024) 03611981241230297. https://doi.org/10.1177/03611981241230297
[44] Shafaati, M., Saffarzadeh, M. “Evaluating Perceived travel time and travel time reliability in the transit system of Tehran (Case study: the first line of the BRT system of Tehran) ”, Amirkabir Journal of Civil Engineering, 55(12) (2024) 4-4. 10.22060/CEEJ.2023.21968.7864
[45] Gao, W., Lu, S., Liu, K., and et al. “A Two-Stage Algorithm Based on Variable Distance Threshold for Estimating Alighting Stops Using Smart Card Data”, Journal of Transportation Engineering, Part A: Systems, 150(1) (2024) 04023123. https://doi.org/10.1061/JTEPBS.TEENG-7744
[46] Zhang, Q., Ma, Z., Zhang, P., and et al. “Mobility knowledge graph: review and its application in public transport”, Transportation, (2023) 1-27. https://link.springer.com/article/10.1007/s11116-023-10451-8
[47] Jafari Kang, M., Ataeian, S., Amiripour, S.M. “A procedure for public transit OD matrix generation using smart card transaction data”, Public Transport, 13 (2021) 81-100. https://doi.org/10.1007/s12469-020-00257-7
[48] Provost, F. “Machine learning from imbalanced data sets 101, ” in: Proceedings of the AAAI’2000 workshop on imbalanced data sets, AAAI Press, 2000, pp. 1-3. http://hdl.handle.net/2451/27763
[49] Zhou, Z., Liu, X. “Training cost-sensitive neural networks with methods addressing the class imbalance problem”, IEEE Transactions on knowledge and data engineering, 18(1) (2005) 63-77. DOI: 10.1109/TKDE.2006.17
[50] Fernández, A., García, S., Galar, M., and et al. “Learning from imbalanced data sets”, Springer, 2018. https://doi.org/10.1007/978-3-319-98074-4
[51] Fawcett, T. “ROC graphs: Notes and practical considerations for researchers, ” Machine learning, 31(1) (2004) 1-38. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=117478d45bb201d95725a60715736c726b9a6241
[52] Jenelius, E. “Public transport experienced service reliability: Integrating travel time and travel conditions”, Transportation Research Part A: Policy and Practice, 117 (2018) 275-291. https://doi.org/10.1016/j.tra.2018.08.026

Biographies

Mostafa Shafaati was born in 1992 in the small town of Garmsar in Semnan Province, Iran. It was in 2014 that he received his Bachelor's degree in Civil Engineering from Isfahan University of Technology. He earned his MSc in Highway and Transportation Engineering from Tarbiat Modares University of Tehran in 2017. He is currently a PhD candidate in Transportation Planning at the same university under the supervision of Professor Saffarzadeh. His research interests include public transportation, traffic safety, and machine learning.

Mahmoud Saffarzadeh is a well-known professor of transportation in Iran. He received his Bachelor of Science in Civil Engineering from Shahid Bahonar University of Kerman in 1987. Later, he attended Carlton University in Canada, where he earned his MSc and PhD degrees in 1991 and 1995, respectively. At present, he is a professor at Tarbiat Modares University (TMU) of Tehran, as well as the head of the Faculty of Civil and Environmental Engineering at TMU. In addition to being the founding father of the Transportation Planning group at the TMU, he is also the founder and CEO of the Tarahan Parseh Transportation Research Institute, the editor-in-chief of the International Journal of Transport Engineering, and many others. With numerous publications in different areas of transportation science, he is a prominent figure in Iranian transportation science.

List of Figure Captions

List of Table Captions

Table 9. The result of the second validation approach

Figures

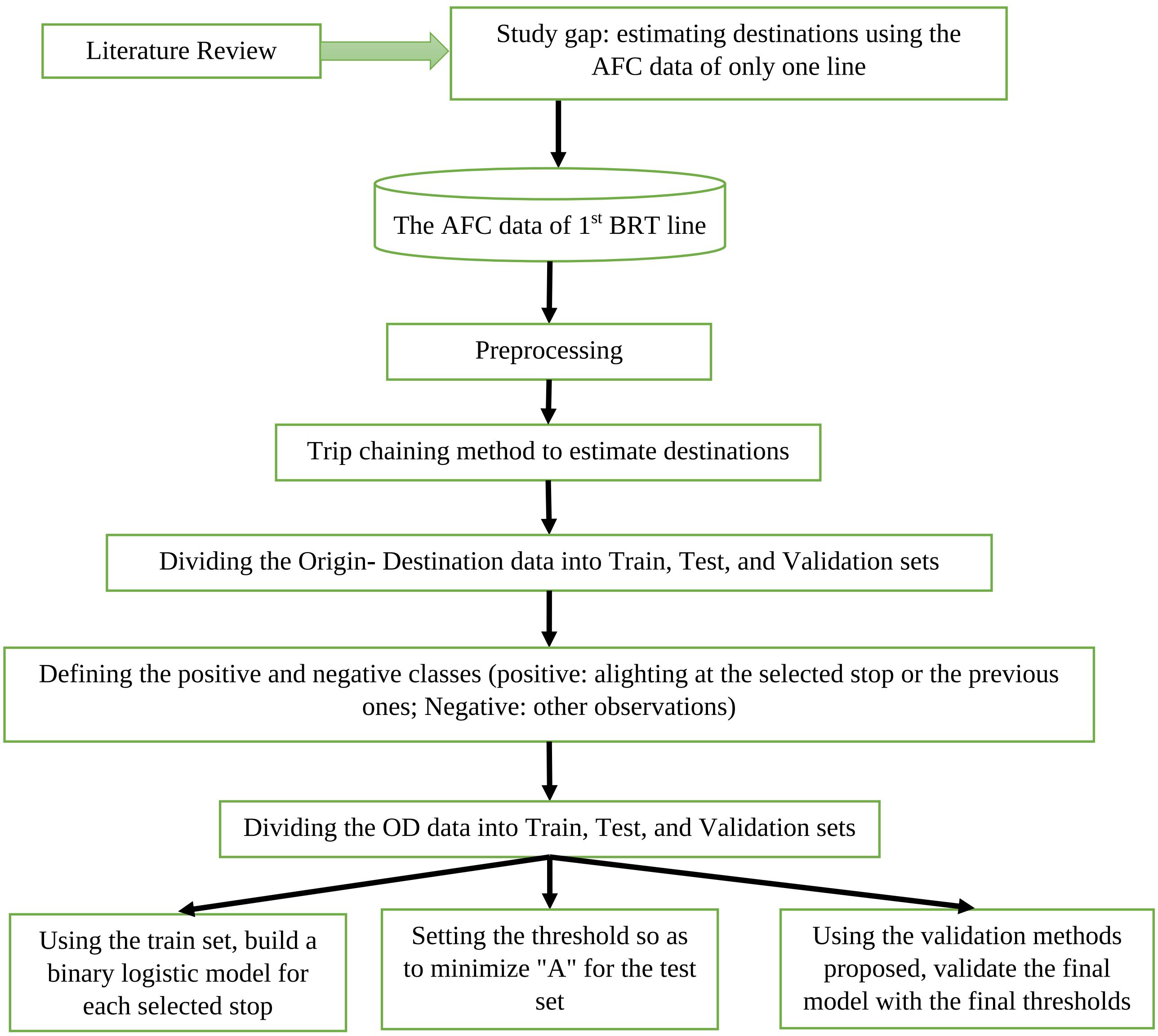

Figure 1. The framework of the proposed method illustrated

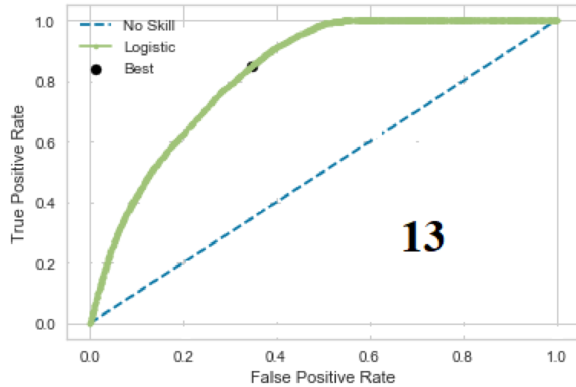

Figure 2. Calculation of stop 13's primary threshold using ROC curves

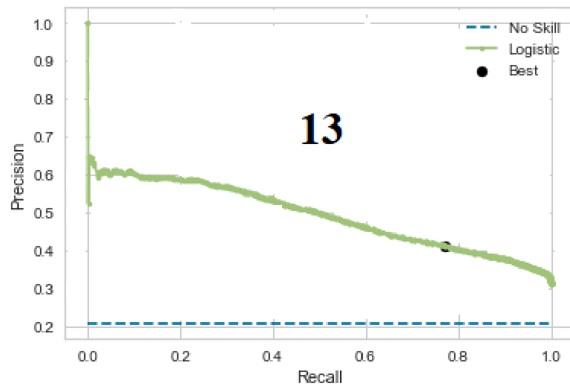

Figure 3. Calculating the primary threshold for stop 13 using the Precision-Recall curve

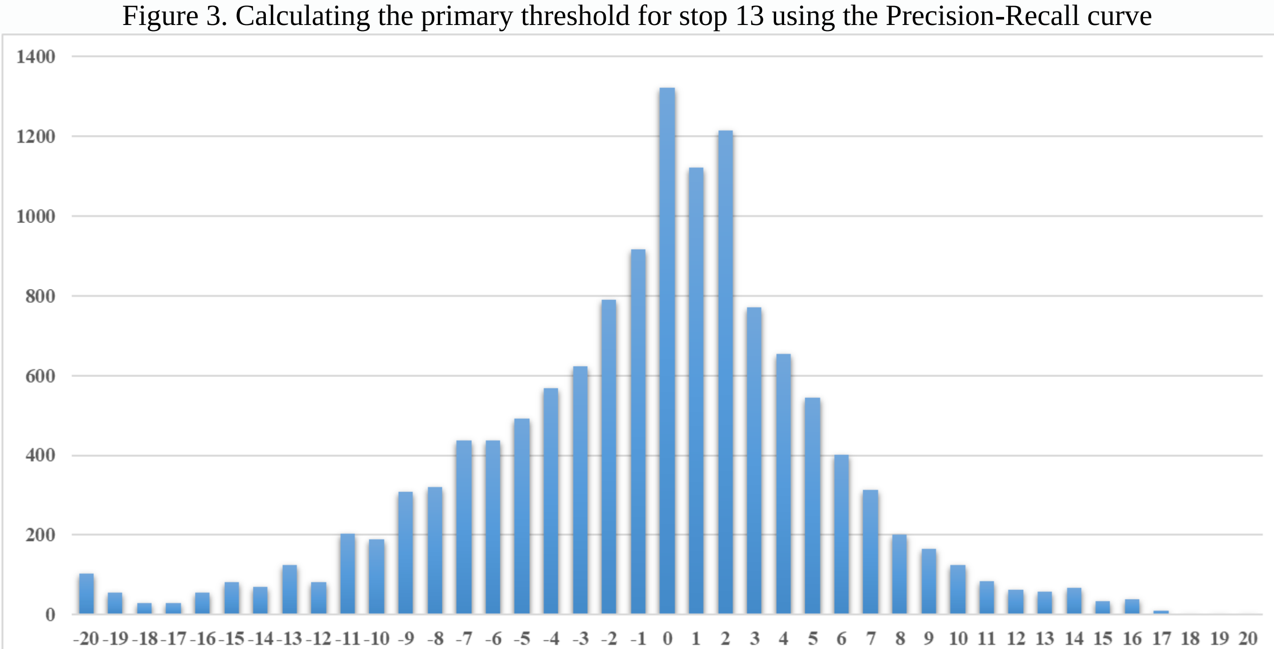

Figure 4. Bar plot of N and diff (y-axis is N and x-axis is diff) (first validation sample)

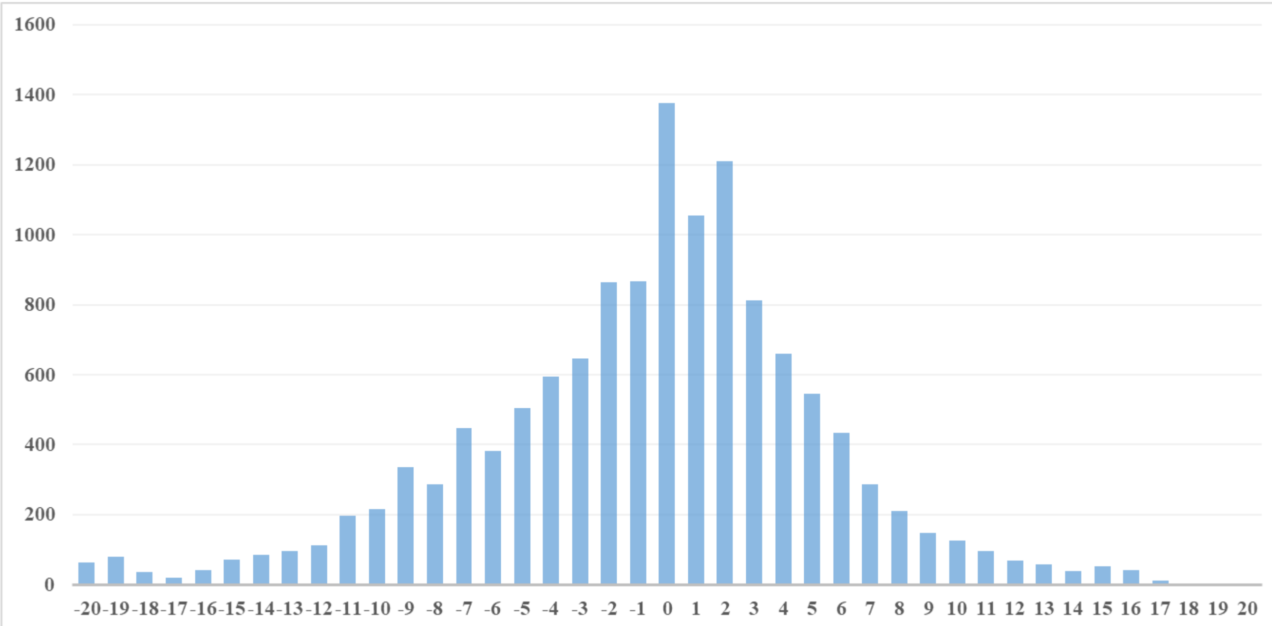

Figure 5. Bar plot of N and diff (y-axis is N and x-axis is diff) (second validation sample)

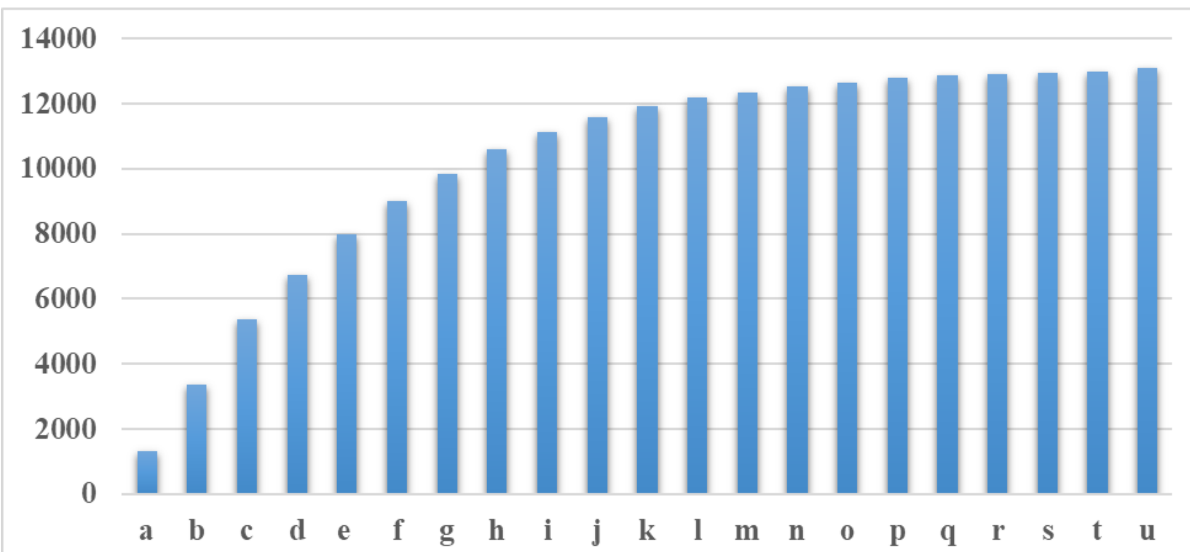

Figure 6. The number of validated observations based on accepted errors (the first validation sample)

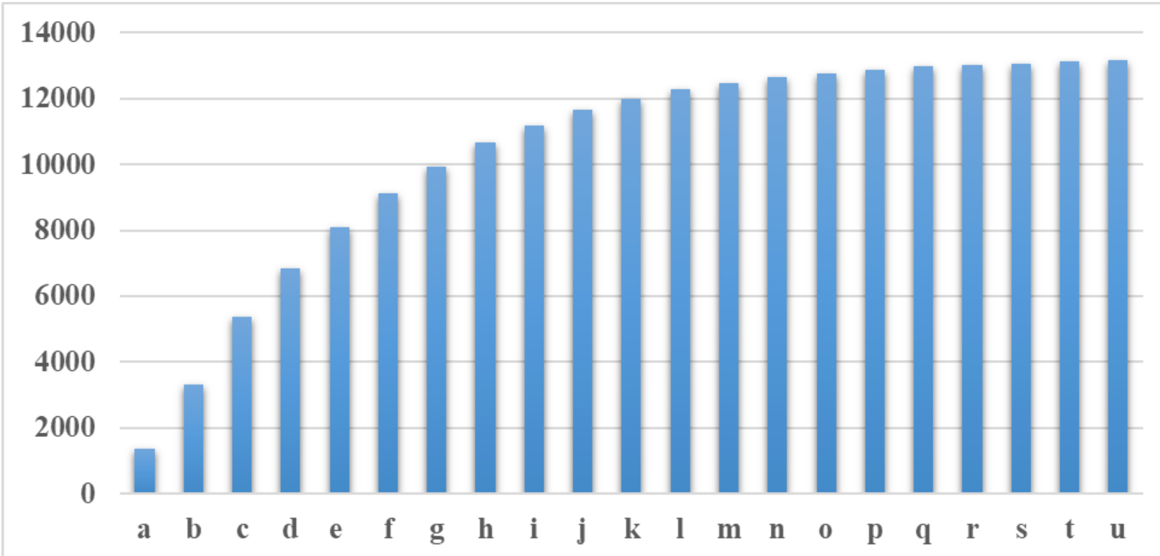

Figure 7. The Number of validated observations based on accepted errors (the second validation sample)

Tables

Table 1. Example of AFC data

Serial	Code	Line	Date	Time	reader
65013	132	2501	2019-11-26	09:11	1015
65013	132	2501	2019-11-26	12:11	1058
197333	132	2501	2019-11-28	12:11	1091
197333	132	2501	2019-11-28	12:11	1091
204101	132	2501	2019-11-24	18:11	1047
209237	132	2501	2019-11-27	06:11	1094

Table 2. Describing the data used in the study

Variable	**Category**	**Frequency**
Dstop	2	198
	3	1112
	4	540
	5	3672
	6	1251
	7	2059
	8	2983
	9	1927
	10	1805
	11	3652
	12	4789
	13	6185
	14	3922
	15	4152

Variable	Category	Frequency
	16	9121
	17	10361
	18	18402
	19	6777
	20	13180
	21	2324
	22	7115
	23	6895
	24	3813
	25	5192
	26	9682
	27	15756
Ostop	1	21287
	2	6022
	3	10517
	4	4254
	5	16985
	6	7211
	7	5213
	8	6634
	9	5395
	10	4014
	11	4336
	12	5551
	13	5025
	14	3484
	15	2906
	16	2946
	17	4438
	18	8709
	19	4548
	20	6261
	21	1739
	22	4635
	23	2678
	24	1077
	25	620
	26	380
TIME	5	2564
	6	12482
	7	23075
	8	16575
	9	12386
	10	9533
	11	7840
	12	7779
	13	6839

Variable	Category	Frequency
	14	7403
	15	6561
	16	7799
	17	8781
	18	6962
	19	4940
	20	3351
	21	1995
Day	1	26903
	2	28195
	3	39798
	4	28753
	5	23216
metroOrigin	0	110476
	1	36389
secOrigin	0	116536
	1	30329

Table 3. The elements of the confusion matrix for each model at each stop

Element	**Definition**
P	Positive class (Those who alight at the current or previous stops)
N	Negative class (Those who don't alight at the current or previous stops)
TN	True Negative (The actual and predicted classes are both negative)
FP	False Positive (The actual class is negative but the predicted is positive)
FN	False Negative (The actual class is positive but the predicted is negative)
TP	True Positive (The actual and predicted classes are both positive)

Table 4* The results of calculating the best threshold

stop	threshold	N*	P**	PercentN	PercentP	G_Mean	GThreshold	ThresholdROC	F_Score	F-Threshold
7	0.277	82772	5347	93.932	6.068	0.826	0.077	0.057	0.357	0.182
8	0.290	81002	7117	91.923	8.077	0.804	0.092	0.088	0.390	0.187
9	0.300	79826	8293	90.589	9.411	0.791	0.120	0.095	0.403	0.207
10	0.319	78733	9386	89.348	10.652	0.784	0.135	0.109	0.426	0.257
11	0.340	76529	11590	86.847	13.153	0.781	0.164	0.126	0.459	0.198
12	0.341	73661	14458	83.593	16.407	0.767	0.207	0.159	0.499	0.232
13	0.371	69934	18185	79.363	20.637	0.744	0.235	0.204	0.533	0.273
14	0.393	67598	20521	76.712	23.288	0.737	0.283	0.251	0.559	0.283
15	0.420	65077	23042	73.851	26.149	0.729	0.307	0.267	0.584	0.294
16	0.480	59618	28501	67.656	32.344	0.726	0.383	0.341	0.648	0.341
17	0.571	53378	34741	60.575	39.425	0.732	0.454	0.399	0.715	0.391
18	0.675	42311	45808	48.016	51.984	0.755	0.613	0.472	0.814	0.433
19	0.702	38261	49858	43.42	56.58	0.764	0.614	0.549	0.841	0.489
20	0.733	38261	49858	43.42	56.58	0.782	0.712	0.685	0.880	0.472

*stop: stop number, threshold: the best threshold based on this paper's method, P: Number of Positive Class members, N: Number of Negative Class Members, GThreshold: The best threshold calculated based on maximizing G-mean, ThresholdROC: the best threshold based on ROC curves, F_threshold: the best threshold based on maximizing F-Score.

Table 5. Trying different thresholds to obtain the best confusion matrix

T1* / CM**, threshold =0.271		T2 / CM, threshold =0.281	
33243	13379	34337	12285
2864	9260	3299	8825
T3 / CM, threshold =0.291		T4 / CM, threshold =0.301	
35085	11537	36202	10420
3642	8482	4179	7945
T5 / CM, threshold =0.311		T6 / CM, threshold =0.321	
37331	9291	37742	8880
4654	7470	4817	7307
T7 / CM, threshold =0.331		T8 / CM, threshold =0.341	
38226	8396	38773	7849
5062	7062	5337	6787
T9 / CM, threshold =0.351		T10 / CM, threshold =0.361	
39734	6888	40044	6578
5812	6312	5993	6131
T11 / CM, threshold =0.371		T12 / CM, threshold =0.381	
40359	6263	40578	6044
6174	5950	6281	5843

*Attempt No. 1
**Confusion matrix

Table 6. Calculating A for each attempt

threshold	**FP**	**FN.**	**A= FP-FN**
0.271	13379	2864	10515
0.281	12285	3299	8986
0.291	11537	3642	7895
0.301	10420	4179	6241
0.311	9291	4654	4637
0.321	8880	4817	4063
0.331	8396	5062	3334
0.341	7849	6337	1512
0.351	6888	5812	1076
0.361	6578	5993	585

0.371	6263	6174	89
0.381	6044	6281	-237

Table 7. Validating the results with two validation sets

First validation sample		**Second validation sample**	
Diff	**N**	**Diff**	**N**
-20	104	-20	64
-19	55	-19	80
-18	30	-18	35
-17	30	-17	20
-16	55	-16	43
-15	81	-15	72
-14	69	-14	84
-13	125	-13	95
-12	82	-12	112
-11	203	-11	196
-10	188	-10	216
-9	309	-9	335
-8	320	-8	288
-7	436	-7	448
-6	438	-6	381
-5	492	-5	504
-4	569	-4	596
-3	622	-3	646
-2	789	-2	864
-1	916	-1	867
0	1321	0	1377
1	1121	1	1056
2	1214	2	1209
3	771	3	812
4	653	4	659
5	545	5	545
6	402	6	433
7	313	7	287
8	201	8	210
9	165	9	147
10	126	10	127
11	84	11	95
12	64	12	70
13	59	13	57
14	67	14	40
15	34	15	54
16	39	16	43
17	10	17	12
18	0	18	2

* Diff shows the difference between the predicted and actual stop number.

**N is the number of observations that match each different outcome.

Table 8. The results of the first validation approach

First validation sample			Second validation sample		
ACEC*	NMC**	PMC***	ACEC	NMC	PMC
a (|Diff| = 0)	1321	10.08	a (|Diff| = 0)	1377	10.44
b (|Diff| ≤ 1)	3358	25.62	b (|Diff| ≤ 1)	3300	25.03
c (|Diff| ≤ 2)	5361	40.91	c (|Diff| ≤ 2)	5373	40.76
d (|Diff| ≤ 3)	6754	51.54	d (|Diff| ≤ 3)	6831	51.82
e (|Diff| ≤ 4)	7976	60.87	e (|Diff| ≤ 4)	8086	61.34
f (|Diff| ≤ 5)	9013	68.79	f (|Diff| ≤ 5)	9135	69.30
g (|Diff| ≤ 6)	9853	75.20	g (|Diff| ≤ 6)	9949	75.47
h (|Diff| ≤ 7)	10602	80.91	h (|Diff| ≤ 7)	10684	81.05
i (|Diff| ≤ 8)	11123	84.89	i (|Diff| ≤ 8)	11182	84.83
j (|Diff| ≤ 9)	11597	88.51	j (|Diff| ≤ 9)	11664	88.49
k (|Diff| ≤ 10)	11911	90.90	k (|Diff| ≤ 10)	12007	91.09
l (|Diff| ≤ 11)	12198	93.10	l (|Diff| ≤ 11)	12298	93.30
m (|Diff| ≤ 12)	12344	94.21	m (|Diff| ≤ 12)	12480	94.68
n (|Diff| ≤ 13)	12528	95.61	n (|Diff| ≤ 13)	12632	95.83
o (|Diff| ≤ 14)	12664	96.65	o (|Diff| ≤ 14)	12756	96.77
p (|Diff| ≤ 15)	12779	97.53	p (|Diff| ≤ 15)	12882	97.73
q (|Diff| ≤ 16)	12873	98.25	q (|Diff| ≤ 16)	12968	98.38
r (|Diff| ≤ 17)	12913	98.55	r (|Diff| ≤ 17)	13000	98.62
s (|Diff| ≤ 18)	12943	98.78	s (|Diff| ≤ 18)	13037	98.90
t (|Diff| ≤ 19)	12998	99.20	t (|Diff| ≤ 19)	13117	99.51
u (|Diff| ≤ 20)	13102	100.00	u (|Diff| ≤ 20)	13181	100.00

*ACEC (Accepted Classification Error Categories): a = 0 / b = 1 (-1,0,1) / c = 2 (-2, -1, 0, 1, 2) / d = 3 / e = 4 / f = 5 / g =6 / h = 7 / i = 8 / j = 9 / k =10 / l = 11 / m = 12 / n = 13 / o = 14 / p = 15 / q = 16 / r = 17 / s =18/t = 19/ u = 20

**NMC: The Number of Members in the Category; ***PMC: Percentage of Members in the Category

Table 9. The result of the second validation approach

1st validation sample			2nd validation sample		
i	**I_{1i}**	**I_{2i}**	**i**	**I_{1i}**	**I_{2i}**
0	1321	1	0	1377	1
1	205	1.22	1	189	1.21
2	425	1.53	2	345	1.39
3	149	1.23	3	166	1.25
4	84	1.14	4	63	1.10
5	53	1.10	5	41	1.08
6	-36	0.91	6	52	1.13
7	-123	0.71	7	-161	0.64

1st validation sample			2nd validation sample		
i	**I_{1i}**	**I_{2i}**	**i**	**I_{1i}**	**I_{2i}**
8	-119	0.62	8	-78	0.72
9	-144	0.53	9	-188	0.43
10	-62	0.67	10	-89	0.58
11	-119	0.41	11	-101	0.48
12	-18	0.78	12	-42	0.62
13	-66	0.47	13	-38	0.60
14	-2	0.97	14	-44	0.47
15	-47	0.41	15	-18	0.75
16	-16	0.70	16	0	1.00
17	-20	0.33	17	-8	0.60
18	-30	0.00	18	-33	0.05
19	-55	0.00	19	-80	0.00
20	-104	0.00	20	-64	0.00