IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received February 1, 2021, accepted March 4, 2021, date of publication March 9, 2021, date of current version March 16, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3064929

Cancel-for-Any-Reason Insurance Recommendation Using Customer Transaction-Based Clustering

ZHALEH SADREDDINI[1], (Member, IEEE), ILKNUR DONMEZ[2], AND HALIM YANIKOMEROGLU[1], (Fellow, IEEE)

[1]Department of Systems and Computer Engineering, Carleton University, Ottawa, ON K1S 5B6, Canada
[2]Department of Computer Engineering, Istanbul Arel University, 34537 Istanbul, Turkey

Corresponding author: Zhaleh Sadreddini (zhaleh.sadreddini@ieee.org)

ABSTRACT In the travel insurance industry, *cancel-for-any-reason* insurance, also known as a cancellation protection service (CPS), is a recent attempt to strike a balance between customer satisfaction and service provider (SP) profits. However, some exceptional circumstances, particularly the COVID-19 pandemic, have led to a dramatic decrease in SP revenues, especially for non-refundable tickets purchased early with CPS. This paper begins by presenting a risk group segmentation of customers in an online ticket reservation system. Then, a CPS fee is recommended depending on the different customer risk groups provided by the cluster segmentation via different clustering algorithms such as centroid-based K-means, hierarchical agglomerative, DBSCAN, and artificial neural network-based SOM algorithms. According to the implemented cluster metrics, which include the Silhouette index, Davies-Bouldin index, Entropy index, and DBCV index, the SOM algorithm presents the most appropriate result. After predicting the new customer cluster, a CPS fee will be calculated with the proposed adaptive CPS method based on the cluster segmentation weights. Determining the weight of each cluster is related to the total CPS revenue threshold for all clusters defined by the SP. Therefore, to avoid a loss for SPs, the total CPS revenue will be kept constant with the threshold that the SP has been adjusted. The experimental results based on real-world data show that the risk group segmentation of customers helps to maintain a balance between CPS fees and SP profits. Finally, according to the calculated weights, the proposed model pegs the SP gain/loss variation with a 0.00012 exchange ratio.

INDEX TERMS Clustering algorithms, cancellation protection service, risk group segmentation, user satisfaction, service provider revenue.

I. INTRODUCTION

In the airline industry, as the price of flight tickets has increased over time, customers have often planned their trips as early as possible to take advantage of affordable booking opportunities. However, changing one's flight plans can result in high penalties for customers, which may turn their profits into a loss. Motivated by this, some airline companies and agencies have proposed *cancel-for-any-reason* insurance, also known as a cancellation protection service (CPS) [1], [2]. As the name implies, *cancel-for-any-reason* insurance allows customers to cancel their trip for any reason and receive a partial refund of their prepaid and non-refundable expenses, such as airline tickets and hotel accommodations. Although this insurance has some benefits from the customer's point of view, especially for early birds purchasing non-refundable tickets, some exceptional cases, such as the COVID-19 pandemic, led to a dramatic decrease of service provider (SP) revenues. To help SPs overcome this loss, an optimal CPS fee calculation is required to balance SPs revenues alongside customer expectations concerning the quality of experience (QoE) and CPS fees.

In this paper, an adaptive customer transaction-based CPS calculation method is proposed to capture the trade-off between customer satisfaction and SP profits. The goal is to minimize the CPS fee for customers with low or

The associate editor coordinating the review of this manuscript and approving it for publication was Massimo Cafaro.

no-cancellation ratios, and to maximize the CPS fee for customers with high ticket cancellation ratios, which negatively impact SP profits.

The CPS fee calculation method proposed in this study uses real-world customer transaction-based behavior data [2], and presents CPS fee recommendations on the basis of different customer risk groups. These customer risk groups are provided by the cluster segmentation via different clustering algorithms, such as centroid-based K-means, hierarchical agglomerative, density-based spatial clustering of applications with noise (DBSCAN), and artificial neural network based self-organizing map (SOM) algorithms. In the cluster segmentation, several customer-based ticket profiles are considered, including ticket status, ticket type, ticket price, CPS status, CPS fee, and refund fee. In addition, the cluster segmentation considers customer transactions based on the statistical analysis methods to create new features for customer segmentation, such as refundable ticket cancellation ratio with CPS (RTCR), non-refundable ticket cancellation ratio (NTCR), no cancellation ratio of refundable tickets (NCRR), and no cancellation ratio of non-refundable tickets (NCRR). When a customer wants to book a ticket, an adaptive CPS fee will be proposed on the basis of the customer cluster. Note that clustering metrics are used to validate the clustering methods and each customer belongs to the cluster with the nearest euclidean and cosine distance between the customer and each cluster centroid. For the clusters which contain the most and least profitable customers, new weights are proposed to calculate the new CPS fee. The SP gain/loss (SPGL) prediction will also be evaluated for suggested weights and the adaptive CPS fee. The extensive simulations driven by real-world data involving different use cases are carried out to show the profit/loss variance related with different cluster weights that are used for recommending new CPS values.

Last but not least, the performance of the previously proposed CPS fee functions without machine learning (ML)-based customer classification [1] is compared with the currently proposed adaptive CPS fee function on the basis of customer transaction-based clustering. The experimental results show that the proposed risk group segmentation of customers helps to maintain a balance between the calculation of the CPS fee and SPGL. Finally, according to the calculated weights, the proposed model pegs the SPGL variation in terms of the total CPS fee of the higher and lower risk groups with a 0.00012 exchange ratio.

A. RELATED WORKS

There are several works that shed light on optimal purchase times for airline tickets and help to predict ticket prices [3]–[9]. However, predicting actual ticket prices is a more difficult task than predicting optimal purchase times for various reasons: a lack of sufficient datasets, external factors influencing ticket prices, the dynamic behavior of ticket pricing, competition among airlines and their proprietary nature, ticket pricing policies, etc. Moreover, early purchasing presents a risk due to the commitment to a specific flight date that may need to be changed, usually for a fee.

In order to determine an optimal *cancel-for-any-reason* travel insurance fee, the multiple criteria decision-making (MCDM) method introduced in our previous work [1] which designs a decision-based management framework of ticket reservation systems from the perspective of covering a pre-booked ticket in case of any cancellations by the customers. In addition to the MCDM methods, different clustering algorithms can be used to calculate an optimal CPS fee based on the customer risk groups.

Regarding the clustering algorithms, the clustering of customer transaction data is one of the most critical tasks in successful marketing and customer relationship management [10]–[12]. Clustering is used to categorize customers into different groups on the basis of their purchasing behaviors. With the rapid increase in the availability of customer behavior data, several studies have used product-specific variables. Furthermore, a selection of early segmentation methods used general variables, such as customer demographics, lifestyle, attitude, and psychology, because such variables are intuitive and easy to operate [13]. In [14], the authors propose a segmentation methodology to identify similarities between customers. The authors in [15] provided a novel MST-based clustering algorithm called LDP-MST to propose a minimum spanning tree-based clustering with local density peaks. Regarding this approach, the authors in [16] proposed the algorithm with sensitivity of local density and density-adaptive metric. In [17], the authors presented the robust density peaks clustering algorithm using fuzzy neighborhood. The authors in [18] introduced this kind of clustering using geodesic distances (DPC-GD). Taking account the kind of approach that these studies are using, some of them are incapable of handling large-scale transaction data due to their high computational complexity. Thus, a PurTreeClust clustering algorithm was proposed for large-scale transaction data [19].

SOM is a well-known unsupervised learning strategy that can be applied to a wide range of data visualizations, dimensionality reduction or clustering problems [20], [21]. A new hierarchical agglomerative clustering algorithm for SOMs is introduced in [22] based on neighborhood relations of the SOM prototypes in the data space and detailed local density distribution in their receptive fields. In [23], the authors proposed an algorithm for high-speed learning in hardware SOM while analyzing the drawbacks of the SOM algorithm for FPGA implementations and proposing a new learning algorithm to solve them. In addition, the authors in [24] present information-theoretic-cluster visualizations (IT-vis) for SOMs.

The authors in [25] segmented customer experiences using a K-means method. In [26], the authors proposed two new centroid-based *k*-NN classification algorithms to select optimal k-values for each test sample for efficient and effective *k*-NN classification. In [27], a segmentation of cashback website customers was presented. The segmentation was based on

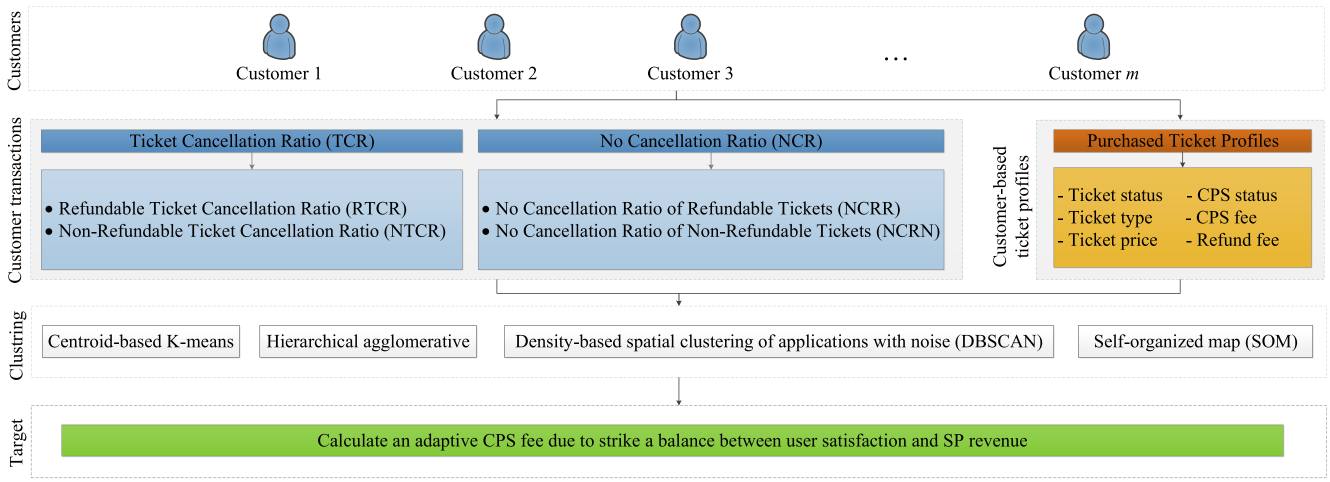


FIGURE 1. Schematic of the research methodology.

customers' commercial activity and the role within the site's social network. In [28], an apriori algorithm was applied for segmenting customers by mining for association rules in the database of a Turkish supermarket chain.

II. MOTIVATION AND CONTRIBUTIONS

The contributions are summarized as follows:

- A new CPS fee function is proposed on the basis of the customer transactions and their ticket-based profiles to minimize the CPS fee for the customers with less ticket cancellation ratio and no-cancellation ratio, and maximize the CPS fee for the customers with high ticket cancellation ratio which causes to decrease the profit of the SP.
- K-means clustering, hierarchical agglomerative clustering, DBSCAN, and artificial neural network-based SOM algorithms are used to classify customers' risk groups.
- The performance of the previously proposed CPS fee functions [1] without machine learning (ML)-based customer classification is compared with the currently proposed adaptive CPS fee function.
- The extensive simulations driven by real-world data with considering different use cases are carried out to show the enhanced performance of the proposed adaptive CPS fee calculation on the basis of the customer classification of the SOM algorithm.

The rest of the paper is organized as follows. Section III presents the research methodology, which consists of [A] data collection, [B] cluster analysis, [C] the proposed adaptive CPS method. Section IV evaluates the performance of the proposed model with simulations. A conclusion and a discussion of future work is given in Section V.

III. RESEARCH METHODOLOGY

This section is divided into three sub-sections: [A] data collection, [B] cluster analysis, and [C] adaptive CPS method. A schematic of our methodology is presented in Fig. 1, where the aim of the proposed model is to calculate an adaptive CPS fee.

A. DATA COLLECTION

In this sub-section, driven by real-world data, there are two data collection steps, as shown in Fig. 1: the customer transactions and the customer-based ticket profiles.

1) CUSTOMER TRANSACTIONS

Each customer has a transaction history in the proposed data center of the SP. This could be a Ticket Cancellation Ratio (TCR) or a No Cancellation Ratio (NCR), as described below.

- **Refundable Ticket Cancellation Ratio with CPS (RTCR).** This refers to the percentage of ticket cancellations among the refundable tickets bought with a CPS payment.
  In this case, the airline refunds the ticket price due to the refund policy of the refundable ticket type. The gain/loss of the SP in the case of such a ticket cancellation is given as follows:

$$SP_F = \begin{cases} \Upsilon + \vartheta + \Gamma - \phi - \mu, & \tau \geqslant 2, \\ \Upsilon + \vartheta + \Gamma - \phi, & \tau < 2, \end{cases} \tag{1}$$

  where Υ is the ticket sale price for each ticket purchased, ϑ shows the CPS fee that was paid by the customer while booking the flight ticket, Γ is the airline refund fee, ϕ is the fee which the SP should pay to the airline based on the contract for each ticket sold, and μ shows the maximum assurance value paid by the SP for each ticket cancelled with CPS. τ shows the difference between the cancellation time and the flight time. Typically, when $\tau > 2$, the airline refunds the ticket price to the SP. When this threshold is violated, no refund will be paid to the SP.

Note that the customer gain/loss can be calculated with the following formula:

$$0 \leq \frac{\Upsilon - \Gamma}{\Upsilon + \vartheta - \mu} < 1,$$

where 0 indicates a 100% gain and 1 indicates a 100% loss for the customer.

- **Non-Refundable Ticket Cancellation Ratio with CPS (NTCR).** This shows the total number of cancelled non-refundable tickets for which the CPS fee has been paid. Refunds are rarely supported by airlines in such cases. Therefore, the SP must pay μ by itself. The SPGL for the NTCR, SP_P, can be calculated as a cost function for non-refundable tickets similar to (1) where the only difference is eliminating Γ since $\Gamma \cong 0$ for the NTCR. It should be noted that, in the case of a high NTCR, the $SPGL \leqslant 0$ which shows the SP's loss where the refund value must be paid by the SP instead of the airline. This loss is calculated as follows:

$$SP_P = \begin{cases} \Upsilon + \vartheta - \phi - \mu, & \tau \geqslant 2, \\ \Upsilon + \vartheta - \phi, & \tau < 2. \end{cases} \quad (2)$$

- **No Cancellation Ratio of Refundable Tickets with CPS (NCRR).** This refers to the ratio of refundable tickets bought without any cancellation where all CPS payments are reserved for the SP.
- **No Cancellation Ratio of Non-Refundable Tickets with CPS (NCRN).** This refers to the ratio of non-refundable tickets bought without any cancellation.

For both NCRR and NCRN, the SP gain is calculated as follows:

$$SP_N = \Upsilon - \phi + \vartheta. \quad (3)$$

Finally, the total gain/loss of the SP based on the customer transactions can be calculated as follows:

$$SPGL = SP_F + SP_P + SP_N. \quad (4)$$

2) CUSTOMER-BASED TICKET PROFILES

The proposed real-world data center [2] collects the following ticket attributes:

- **Ticket status.** There are two different status: booking or refund.
- **Ticket type.** This indicates a ticket type as refundable or non-refundable.
- **Ticket price.** This refers to the total ticket price, which the customer is supposed to pay.
- **CPS status.** This indicates that a customer has purchased a ticket with or without a CPS.
- **CPS fee.** This refers to the CPS fee calculated on the basis of the proposed method.
- **Refund fee.** In case of a ticket cancellation, a refund fee will be calculated.

B. CLUSTER ANALYSIS

For the risk group segmentation, four different clustering algorithms are used. These methods are K-means as a centroid based clustering method, agglomerative clustering as a hierarchical based clustering method, DBSCAN as a density-based spatial clustering algorithm, and SOM as a neural network based clustering method. The proposed clustering methods are described as follows:

1) CLUSTERING METHODS

- **K-means Clustering.** K-means clustering is a common algorithm that aims to partition data into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centroid) [29], [30]. To process the training data, the K-means algorithm starts with a first group of randomly selected cluster centroids, and then performs iterative calculations to optimize the positions of the centroids. After all of the data has been addressed, the centroids of all clusters are recalculated and the data is again addressed to the cluster at the nearest updated centroid. The algorithm will continue until there are no differences between clusters or until a threshold value (centroid position change amount) is used to stop the algorithm.
- **Agglomerative Clustering.** Clustering data hierarchically is the concept followed by agglomerative hierarchical clustering. This involves forming a tree structure with clusters and sub-clusters. The clustering method starts at the bottom. The algorithm proceeds iteratively by consecutively merging pairs which are nearest to each other [31], [32].
- **DBSCAN and ordering points to identify the clustering structure (OPTICS) Clustering.** DBSCAN is a density-based spatial clustering algorithm [33]. For a set of points in space, it groups points with many nearby neighbors and marks the outlier points that lie alone in low-density regions. Max radius of the neighborhood on which the proximity search will be performed by epsilon and the minimum number of data points within the max radius to be considered a cluster as minpts are the parameters for this algorithm [34]. The advantage of this clustering algorithm is to find the different shaped clusters. In addition, OPTICS is an extended version of DBSCAN which will try to find a cluster for different max radius values [35], [36].
- **SOM Clustering.** SOM, or Kohonen Systems, is an unsupervised learning strategy that can be applied to a wide range of data visualizations, dimensionality reduction or clustering problems. SOM produces a low-dimensional, discretized representation of the input space of the training samples, called a map. After training a SOM on the input data, it can be used to visualize the high-dimensional input data in a (typically) two-/three-dimensional view, preserving its topological properties [20], [21].

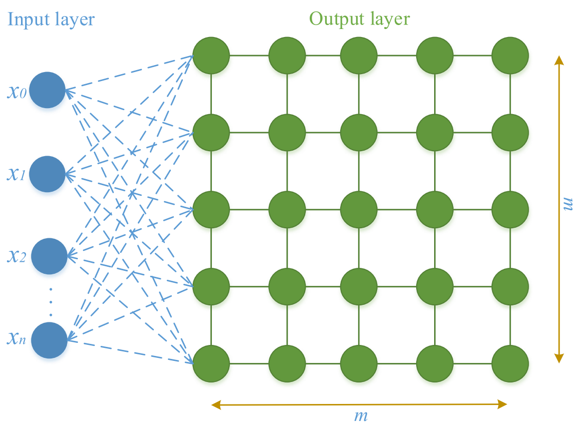


FIGURE 2. General diagram for the SOM.

As we can see in Fig. 2, SOM architecture involves input and output neurons, and weights from each input neuron are connected to each output neuron. The weight connections for only the first five output neurons are shown in the figure.

In an SOM algorithm, an input neuron is selected and the distances between it and each output neuron are calculated. The closest neuron is called the best matching unit (BMU) or winner neuron. Then, the weight between the selected input neuron and the winner neuron are updated. The weights of the BMU and neurons close to it in the SOM grid are updated towards the input vector. The magnitude of the change decreases with time and with the grid-distance from the BMU. This process is repeated for all inputs for a predetermined number of cycles. As a result, the network associates the output nodes with the input dataset. The SOM algorithm can be summarized with the following steps:

- Pick a random input sample,
- Find best matching BMU,
- Update weight vectors of the nodes in the neighborhood of the BMU, and
- Repeat until iteration limit is reached.

2) CLUSTERING METRICS

The quality of clustering is evaluated using clustering metrics. Silhouette Index, Davies-Bouldin Index and Entropy are the most widely used clustering metrics. They are described below.

- **Silhouette Index (SI).** This is a measure of how similar an object is to its own cluster compared to other clusters. If two different SI values have been compared after the clustering process, the total distance between the nearest neighbor should be a large value whereas the total cluster distances inside a cluster should be small [37], [38].
- **Davies-Bouldin Index (DBI).** This is the ratio of the within-cluster mean distribution distance to the inter-cluster distance [39].

  For optimal clustering, the within cluster scatter for cluster i should be small and the mean distance between the ith cluster centroid to the jth cluster centroid should be larger.
- **Entropy.** This is used to measure the success of clustering in clustering algorithms. Entropy refers to the disorder or irregularity of the system. In an optimal clustering process, the irregularity of objects within clusters should be low whereas the irregularity between clusters should be high.
- **Density-Based Clustering Validation (DBCV) Index.** Density-based clustering algorithms look for high-density areas separated by low-density areas containing noise objects. These algorithms are relatively good at finding non-spherical clusters. If the cluster is not spherical, the indexes recommended for globular cluster may not be validated. The density-based relative confirmation index for clusters of different shapes in [40] evaluates the clustering quality based on the relative density between object pairs. In this study, the DBCV index is used for evaluating DBSCAN algorithm.

After the cluster analysis, customer cluster profiles will be defined. Thereafter, an adaptive CPS fee will be calculated depending on the customer cluster as the process is detailed in Fig. 3. By the time of a ticket booking, depending on

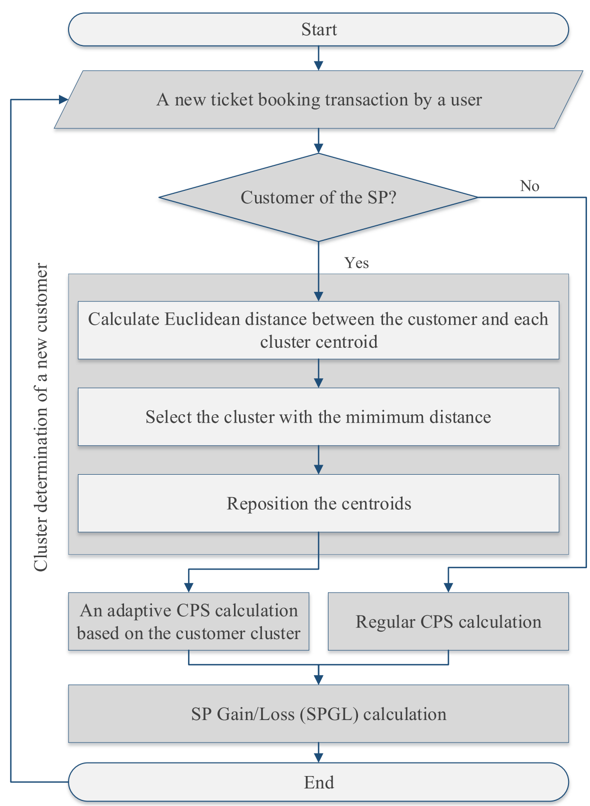


FIGURE 3. The flowchart of the proposed model.

the previous customer transactions given in Fig. 1, cluster prediction for a new customer will be calculated as follows:

- Calculate euclidean similarity between the customer and each cluster centroid.
- Select the cluster with the minimum euclidean distance.
- Re-position the centroids.

C. ADAPTIVE CPS METHOD

The proposed adaptive CPS method provides a CPS discount for customers in the minimum risk group whereas the CPS fee will be increased for the customers in the highest risk group. Therefore, not only will the profit of the SP be fixed, but the proposed method will also strike a balance between customer clusters.

The proposed adaptive CPS fee is denoted as $\Theta_{k,i}$ where k is the index of the cluster, $1 \leq k \leq K$, and i is the index of transactions in each cluster C_k where kth cluster has N_k transactions in total, $0 \leq i \leq N_k$. In the proposed method, $K = 4$ and C_1, C_2, C_3 and C_4 are the clusters where union of each cluster are denoted by C contains all clusters. The $\Theta_{k,i}$ according to the customer clusters can be calculated as follows:

$$\Theta_{k,i} = \begin{cases} \xi_{k,i}, & \text{if } k = 1 \text{ or } k = 2, \text{ and } \exists i \in N_k \\ \xi_{k,i} \times w_3, & \text{else if } k = 3 \text{ and } \exists i \in N_k, \\ \xi_{k,i} \times w_4, & \text{else if } k = 4 \text{ and } \exists i \in N_k, \end{cases} \quad (5)$$

where w_3 and w_4 shows the weights of the C_3 and C_4, respectively. The $\xi_{k,i}$ shows the proposed CPS fee of the ith transaction of the customer belonging to the kth cluster. It should be noted that, the fixed CPS fee defined in our previous work [1] without considering k index was calculated as follows:

$$\xi_i = \Upsilon_i \cdot \alpha + \beta,\ 0.08 < \alpha \leqslant 0.2,\ 4.8 < \beta \leqslant 9.8, \quad (6)$$

where Υ_i is the ticket sale price for the ith transaction of the customer. A selected value for α and β are provided in relation to a non-refundable, refundable, domestic or international ticket. Thus, there are 2^2 different states. These values are determined according to the airline refund instructions and SP's decision.

After the calculation of $\Theta_{k,i}$, defining an optimal weight for each cluster, C_k, $1 \leq k \leq K$, is the next step to strike a balance among proposed CPS fees for each customer and the total CPS revenue of the SP, $\sum CPS_F$. The SP will be able to propose a new CPS fee, where the total CPS revenue will be kept constant (e.g., the total CPS income in our previous study [1] is taken into account as the total CPS income threshold). Thus, the SP will be able to adjust a threshold for the minimum CPS income to prevent loss even in the high NTCR. The total CPS fee of the proposed method is calculated as follows:

$$\psi_C = \sum_{k=1}^{K} \sum_{i=1}^{N_k} \Theta_{k,i}, \quad (7a)$$

$$\text{s.t. } \psi_C == \sum CPS_F, \quad (7b)$$

$$w_4 == \frac{\lambda - (\gamma \times w_3)}{\zeta}, \quad (7c)$$

$$1 < w_3 < 2,\ 0 < w_4 < 1, \text{ and } \zeta > 0, \quad (7d)$$

where

$$\gamma = \psi_{C_k} \mid \min(SPGL), \quad (7e)$$

$$\zeta = \psi_{C_k} \mid \max(SPGL), \quad (7f)$$

$$\lambda = \gamma + \zeta. \quad (7g)$$

The total CPS fee of the highest and lowest risk group clusters is presented by γ and ζ, respectively. The sum of these two CPS fee values are λ. It should be noted that C_3 is the highest risk group and C_4 is the lowest risk group which is obtained on the basis of the comprehensive analysis of real-world data in the section III.C. Total CPS fee of each C_3 and C_4 will be updated when a new transaction is recorded. On the other hand, the $1 < w_3 < 2$ means $[0 - 100]\%$ increase in the proposed CPS fee in comparison with the fixed CPS method, whereas the $0 < w_4 < 1$ means $[0 - 100]\%$ discount in the proposed CPS. The reason for selecting $K = 4$ is described in the following sub-section.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

A. PRE-PROCESSING OF DATA

In this stage, the unnecessary features in real-world data are removed, and the normalization is applied. The new features are generated by joining information gathered from different columns. For instance, the "Ticket Status" column is separated from the "Booking" and the "Refund" column. The transactions are grouped according to the customers, and after grouping, some aggregation functions are performed.

B. NUMBER OF CLUSTERS

Determining the optimal number of clusters in a data set is one of the fundamental issues in clustering. The Elbow method is used to decide the number of clusters.

The steps in the Elbow method are explained as follows [41], [42].

- Executing clustering algorithm for different values of K (e.g., by varying K from 1 to 10 clusters).
- For each K, calculating the within-cluster sum of squares (WCSS).
- Plotting the curve of WCSS according to the number of clusters, K.
- The location of a bend in the plot is generally considered an indicator of the appropriate number of clusters.

In this study, the optimal number of clusters is determined as 4, $K = 4$.

C. CLUSTERING ALGORITHMS AND QUALITY METRICS

Four different clustering algorithms are applied to the real-world data to find the best customer segmentation.

1) K-MEANS CLUSTERING

After executing the K-means algorithm for different cluster numbers, the best score is allocated to $K = 4$. The input data is six-dimensional, and since it is not possible to display data in six dimensions, the cluster results are shown in Fig. 4 for two features in two dimensions.

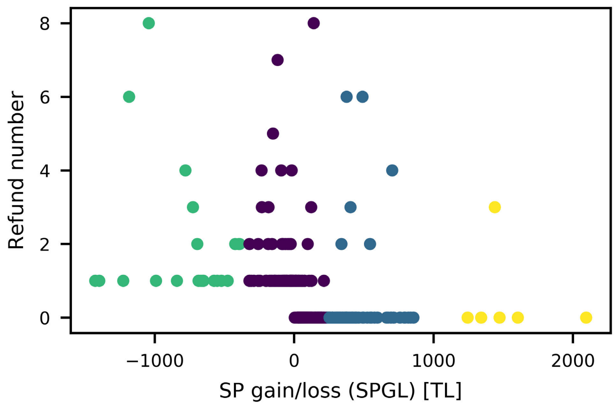


FIGURE 4. K-means result visualisation.

In Table 1, the cluster quality is demonstrated using some cluster metrics. For a comparatively better cluster, the SI should be larger, the DBI should be smaller, and the Entropy value should also be smaller. As we can see in Table 1, the clustering success criterias reached optimal values by using four clusters.

TABLE 1. K-means quality metrics.

Cluster ID	SI	DBI	Entropy
1	0.68	2.03	-
2	0.63	0.58	4.46
3	0.80	0.44	3.05
4	0.84	0.22	1.80

2) AGGLOMERATIVE CLUSTERING

The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. The algorithm starts by treating each object as a singleton cluster. Then, pairs of clusters are successively merged on the basis of their proximity until all clusters have been merged into the required number of clusters, which is four in the proposed algorithm.

In Fig. 5, the clusters that are generated using agglomerative clustering can be seen for only two inputs which are refund number and the SPGL.

As we can see in Table 2, agglomerative clustering using four clusters yielded better results.

3) DBSCAN CLUSTERING

The real-world data is used in clustering step is six-dimensional. Therefore, it is recommended to apply the t-distributed stochastic neighbor embedding (t-SNE) [43] dimension reduction technique before DBSCAN clustering with high dimensional data [44], [45].

To evaluate the success of the DBSCAN clustering method, DBCV index value, SI, DBI, and Entropy values are applied,

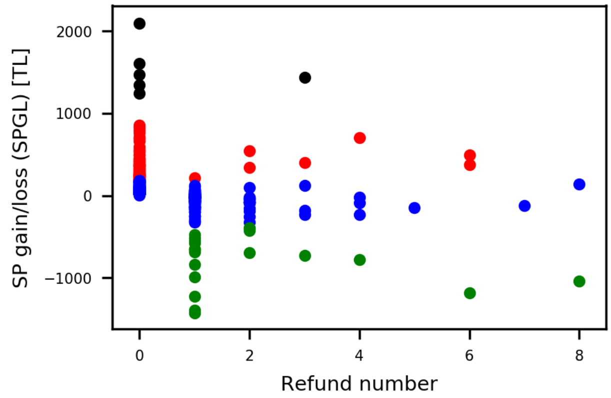


FIGURE 5. Agglomerative clustering result visualisation.

TABLE 2. Agglemerative clustering quality metrics.

Cluster ID	SI	DBI	Entropy
1	0.69	2.04	-
2	0.71	4.97	4.45
3	0.66	1.6	3.05
4	0.71	4.95	3.27

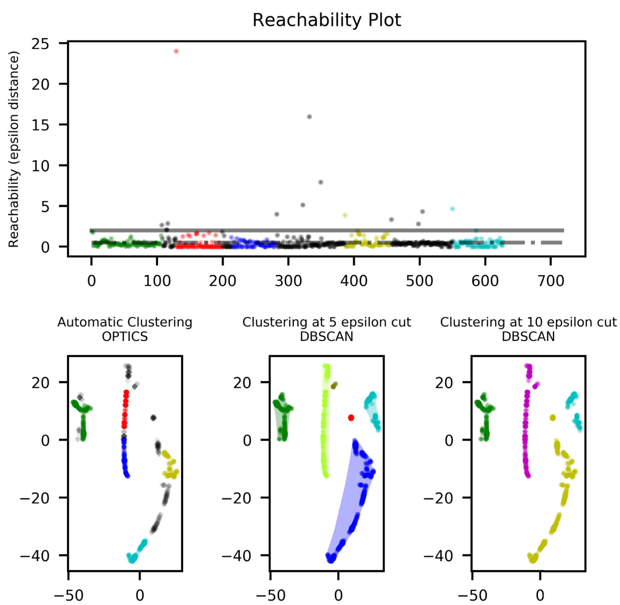


FIGURE 6. DBSCAN and OPTICS result visualisation.

TABLE 3. DBSCAN quality metrics.

DBCV	SI	DBI	Entropy
0.533	0.520	0.666	6.306

as shown in Table 3. Although it may not be appropriate to compare the results with the previous algorithms because of the dimension difference between the reduced dimension and the real-world data original dimension, all the classical cluster evaluation metrics gave worse results in DBSCAN. The reason may be the data converges while reducing the size.

When DBSCAN and OPTICS are applied to the two-dimensional version of real-world data, the clusters can be seen in Fig. 6. It is shown that DBSCAN with epsilon 10 gave a better result for our two-dimensional data compared with OPTICS algorithm. All the data points are clustered in DBSCAN with epsilon 10.

4) SOM CLUSTERING

The Kohonen SOM is an unsupervised neural network commonly used for high-dimensional data clustering. In this study, it has six inputs and 20×20 Kohonen layers.

After applying SOM to the customer data, similar customer behaviors are located to the closer place and the different customer behaviors are located far away from each other on the SOM topological map. As seen in Fig. 7 the topology of the data points are changing during the iterations.

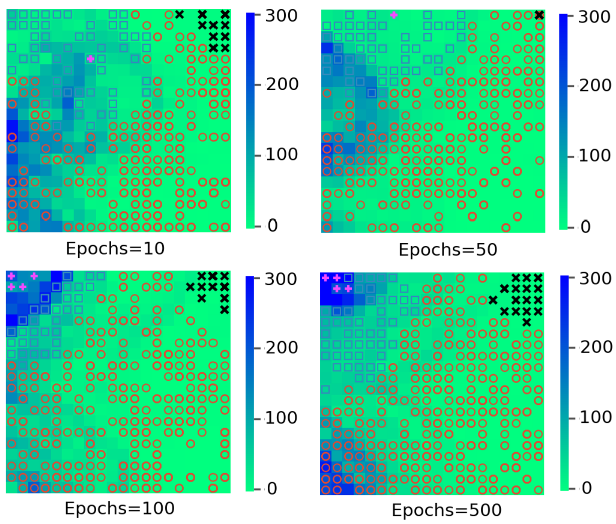


FIGURE 7. Embedded six-dimensional dataset using SOM.

A Kohonen layer is a computational layer that consists of processing units organized in a 2D lattice-like structure. A distinct property of SOMs is that they can map high-dimensional input vectors onto two dimensional space and preserve original topology of a dataset, as we can see in Fig. 7. In our study, training the neural network takes 500 epochs. In terms of artificial neural networks, an epoch refers to one cycle through the full training dataset.

TABLE 4. SOM quality metrics.

Cluster ID	SI	DBI	Entropy
1	0.69	2.02	-
2	0.64	0.57	4.46
3	0.81	0.42	3.04
4	0.86	0.21	1.79

Table 4 demonstrates the SI, DBI, and Entropy metrics of SOM clustering results according to the different cluster numbers. It shows that the quality indexes give the best results for cluster number four. Regarding the comparison of clustering algorithms, SOM has extremely good results according to the agglomerative clustering algorithm, and it gives a slightly better result than the K-means algorithm. Therefore, in customer clustering, and recommending new CPS fee values for the customer transaction, the SOM algorithm with four clusters is used.

Fig. 8 and Fig. 9 present the distances between each cluster according to different distance metrics like Cos-similarity and Euclidean.

While Cosine looks at the angle between vectors (not taking into account their weight or magnitude), Euclidean distance is similar to using a ruler to actually measure the distance. It should be noted that, Cosine distance is equal to 1 − Cos-similarity.

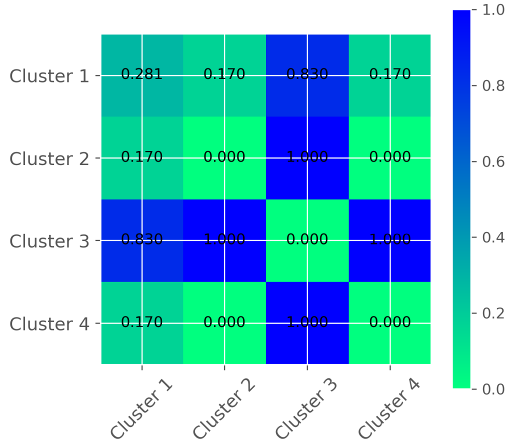


FIGURE 8. Inter-cluster Cosine distances.

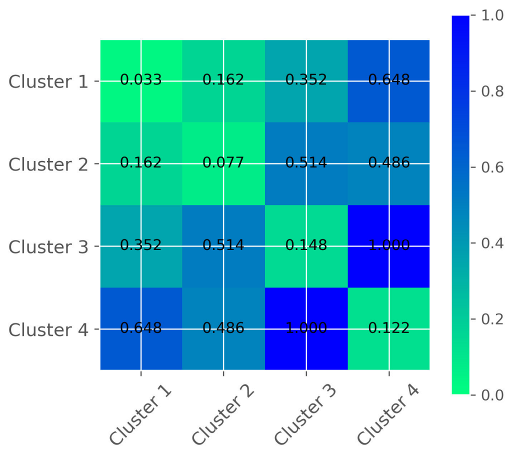


FIGURE 9. Inter-cluster Euclidean distances.

In inter cluster relation Fig. 8, C_3 is totally separated from C_2 and C_4 and also separated from C_1. Both C_2 and C_4 are lapped.

As we can see in Fig. 9, the Euclidean distance gives better results. All the clusters are separated from each other. The closest clusters to each other are C_1 and C_2. One of them

TABLE 5. Feature values of each cluster.

Cluster ID	Booking #	Refund #	Non-refundable ticket #	Refundable ticket #	Total customer #	Total transaction #
1	742	185	407	520	605	927
2	191	23	156	58	86	214
3	8	42	38	12	21	50
4	22	3	20	5	6	25

TABLE 6. SPGL and CPS fee for each cluster.

Cluster ID	SPGL status	SPGL mean value	Total CPS fee	Mean CPS fee per transaction
1	22973	37	32840	35.42
2	35465	412	31179	145.69
3	-16288	-775	6808	135.98
4	9201	1533.5	7933	317.2

TABLE 7. Assigning new customer to a cluster C.

User ID	Booking status	Refund status	Non-refundable ticket status	Refundable ticket status	SPGL status	SPGL mean value	Cluster ID
126	1	0	1	0	29	33	1
56	1	0	1	0	29	29	1
350	1	0	1	0	143	143	2
626	1	0	0	1	51	151	2
513	1	0	1	0	22	-953	3
654	1	0	1	0	48	-360	3
356	1	0	0	1	285	623	4
67	1	0	0	1	31	1185	4

is a neutral cluster that contains transactions that bring no benefit or no loss, $SPGL \cong 0$. The second cluster contains profitable transactions for the SP, $SPGL \geqslant 0$. Although the distance between these two clusters is low, they have different effects on the SPGL. In the proposed adaptive CPS method, both C_3 and C_4 which are absolutely different from each other are considered.

5) CLUSTER PROPERTIES

When we apply four different clustering algorithms, SOM algorithm presented the best results according to the clustering metrics. Therefore, SOM is used to assign the new customer to the most appropriate cluster, and predict the CPS fee. According to the SOM results, the features of each cluster is shown in Table 5 where $K = 4$, and $|\text{Transaction}| == |\text{Booking}| + |\text{Refund}| == |\text{Non-refundable ticket}| + |\text{Refundable ticket}|$. In this table, C_1 represents neutral customers, which cause $SPGL \cong 0$. The transactions of the customers in C_2 represent profits for the SP. C_1 and C_2 cover the largest part of all transactions. However, the customers that have most profitable transactions are not in these clusters. In fact, C_4 is related to the most profitable customers and C_3 is the customer group that always brings loss for the SP. Table 6 presents the SPGL, the mean value of the SPGL, total CPS of the customers at each cluster, and mean CPS value per transaction. It shows that C_3 has a negative effect whereas C_4 has the maximum mean value for the SPGL.

6) CLUSTER PREDICTION

A user who makes a new booking transaction for the first time will be assigned to a neutral cluster by the proposed algorithm. If the user has a transaction history, however, this will be taken into account to decide the cluster of the customer.

In this study, for assigning a new customer to a cluster, all current and previous customer transactions are merged. To represent the customer behavior, the mean and sum values of features are taken into account e.g., refund number, booking number, refundable ticket number and non-refundable transaction number. After finding the customer behavior summary for each customer, the closest cluster centroids to the new customer behavior summary are calculated. These centroids are generated using SOM algorithm. The Euclidean distance which gave the better results than cosine distance is used to measure the closeness.

In Table 7, the current transactions for eight customers are shown. While determining the clusters of new customers, the previous transactions of these customers are taken into account. It is clear that the previous transactions of customers are effective in defining their customer group (Table 7).

D. PERFORMANCE EVALUATION OF THE PROPOSED ADAPTIVE CPS METHOD

1) PROPOSED CPS FEE FOR A NEW CUSTOMER

If no previous transactions of the customer have been recorded, the insurance amount to be paid is calculated with the standard CPS formula [1]. In this study, the new CPS fee will be proposed, especially for the customer who made high gain/loss for the SP. Regarding the SPGL, the calculated CPS fee discount for lower risky group customers will be balanced by the higher CPS fee for customers who bring a higher amount of loss, so that the total profit of the insurance company will not change.

After determining the cluster of new customers using their previous transactions, if the customer is in C_3 or C_4, the recommended CPS fee is calculated. The CPS calculation is presented in (5). To provide the compensation and make a balance between discount for the profitable customer and higher CPS fee for non-profitable customers, the w_3 and w_4 values were selected on the basis of the conditions in (5). As we can see in the formula, there is an infinite number of options for w_3 and w_4. For compensation, the more an SP makes a discount in the CPS fee for one cluster, the more the SP has to increase the CPS fee for the other cluster. In this paper, to compensate the total CPS fee and total SP profit w_3 and w_4 are considered as 1.15 and 0.87, respectively. In Table 8, there are current CPS fees calculated by (6), and the proposed CPS fees calculated by (5) for two randomly selected customers belonging to different clusters C_1 to C_4. Note that both w_3 and w_4 values can be dynamically set in the range of $1 < w_3 < 2$ and $0 < w_4 < 1$, as shown in (7).

TABLE 8. The proposed CPS fee for each customer.

User ID	Customer cluster	Current CPS fee	Proposed CPS fee
5520	1	16.5	16.5
5576	1	22.53	22.53
6126	2	184.22	184.22
8245	2	117.32	117.32
5135	3	16.5	18.48
16531	3	18.51	20.7312
22829	4	28.21	25.9532
13054	4	240	220.8

As we can see in Table 8, there is no difference in the CPS values of the first and second clusters. In third cluster, which contains risky customers, suggested CPS fees are higher than the current CPS fees and for the forth cluster, which contains the most profitable customers, proposed CPS fees are lower than the current CPS values.

2) PROFIT/LOSS VARIATION ACCORDING TO WEIGHTS

The aim of this study is to propose a more appropriate insurance fee for good customers and higher insurance fee for risky customers using cluster coefficients that will not change the total SPGL.

The proposed weight, w_3, is used to calculate the new CPS value of C_3 and w_4 is used to calculate the new CPS value of C_4. It should be noted that, there are infinite opportunities to select w_3 and w_4 that can balance the SPGL on the basis of the SP's decision. As we can see in Fig. 10, using different w_3 and w_4 values, the SP can change the SPGL ratios. In the same figure, the loss and profit amounts are indicated for different w_3 and w_4 values. Decreasing the w_4 values means making a discount in the CPS fee for profitable customers, whereas increasing the w_3 values means raising the CPS fee for customers that cause losses. The w values can be determined by the SP according to their strategic plan.

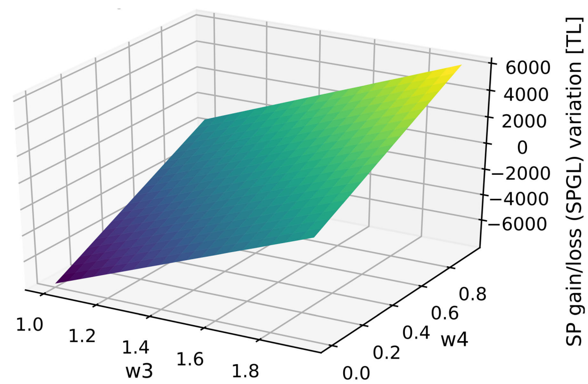


FIGURE 10. Gain/loss variance according to w_3 and w_4 values.

Total current CPS fee, total proposed CPS fee, SPGL value, and estimated SPGL value related to recommended CPS fees for each cluster are shown in Fig. 11. In this figure, the transactions of 500 customers are used. As we can see in this figure, there are no differences between current/proposed CPS fees and current/estimated SP gain/loss values in C_1 and C_2. In C_3, while the total CPS fee is increasing, the total SPGL value is decreasing. By contrast, in C_4 while total CPS Fee is decreasing, the total SPGL value is increasing.

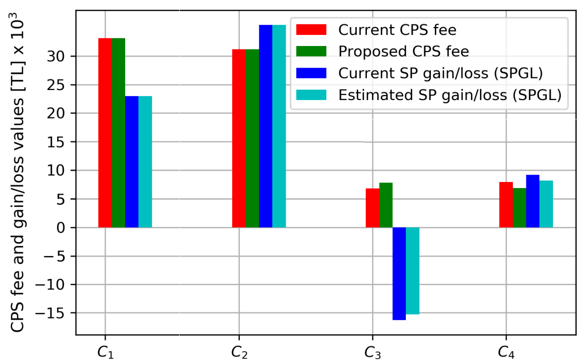


FIGURE 11. Comparison of the CPS fee and SP gain/loss values based on different clusters.

Fig. 12, Fig. 13, and Fig. 14 show the comparison of the current and the proposed CPS fees for 1200 customer transactions in both clusters C_3 and C_4. The proposed CPS fee for each customer transactions in Fig. 12 is higher than the current CPS fee. It means that the higher risky group customers have to pay higher CPS fees. In contrast. Fig. 13 illustrates that the proposed adaptive CPS fee function calculates the lower value for the less risky group of customers. Finally, Fig. 14 proves that the SPGL in terms of the total CPS income does not change in the proposed method in comparison to the previous study [1], where the exchange ratio is almost fixed as 0.00012. Therefore, both current and proposed total CPS fees for C_3 and C_4 are overlapped. The reason is that the proposed adaptive CPS values are calculated on the basis of making a balance between the CPS fee and total SPGL per transaction.

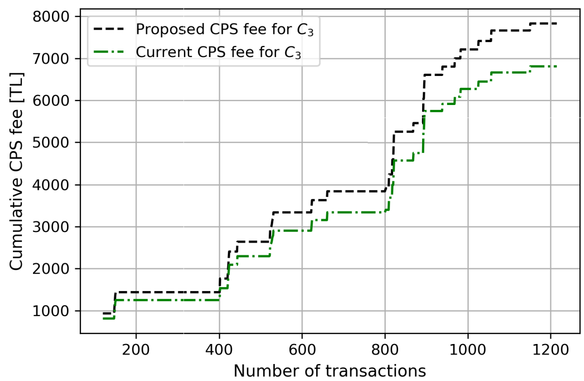


FIGURE 12. Comparison of the current CPS fee calculated by (6) with the recommended CPS fee calculated by (7) for 1200 different customer transactions in cluster C_3.

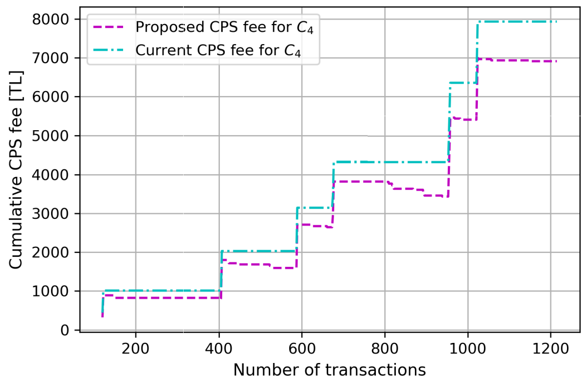


FIGURE 13. Comparison of the current CPS fee calculated by (6) with the recommended CPS fee calculated by (7) for 1200 different customer transactions in cluster C_4.

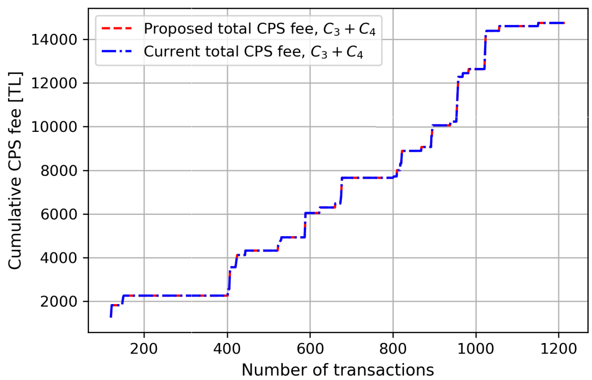


FIGURE 14. Comparison of the total CPS fees provided by Fig. 12 and Fig. 13 in 1200 different customer transactions for both clusters C_3 and C_4.

V. CONCLUSION AND FUTURE WORK

This paper provided a risk group segmentation of customers in an online ticket reservation system. The customer segmentation was developed on the basis of four types of clustering algorithms: K-means clustering, hierarchical agglomerative clustering, density-based spatial clustering of applications with noise (DBSCAN), and artificial neural network based self-organizing map (SOM) clustering. These clustering algorithms were linked with different customer-based ticket profiles and customer transaction based data analysis. After the segmentation, we implemented a number of cluster metrics, including the Silhouette index, Davies-Bouldin index, DBCV index, and Entropy index, which demonstrated that the SOM clustering algorithm was the most appropriate one for our dataset. In our proposed model, therefore, a customer who books a ticket will be assigned to an appropriate cluster, and a CPS fee will be calculated using the proposed adaptive CPS method according to the customer segmentation weights. The experimental results show that the proposed risk group segmentation of customers helps to maintain a balance between the calculation of the CPS fee and the SP gain/loss (SPGL).

This study can be extended by considering new pricing models on the basis of applying reinforcement learning (RL) algorithms to user behavior anomaly analysis.

ACKNOWLEDGMENT

The authors would like to extend a special thanks to the Turna.com data-center team for the dataset used in this study. Dr. Zhaleh Sadreddini was a Visiting Professor at Carleton University when this work started.

REFERENCES

[1] Z. Sadreddini, "A novel cancellation protection service in online reservation system," *IEEE Access*, vol. 8, pp. 129094–129107, 2020.

[2] *Turna Website*. Accessed: Jan. 2020. [Online]. Available: https://www.turna.com/en

[3] A. Lantseva, K. Mukhina, A. Nikishova, S. Ivanov, and K. Knyazkov, "Data-driven modeling of airlines pricing," *Procedia Comput. Sci.*, vol. 66, pp. 267–276, Jan. 2015.

[4] C. Bhavuk and M. C. Kaur, "Airfare analysis and prediction using data mining and machine learning," *Int. J. Eng. Sci. Invention*, vol. 6, no. 11, pp. 10–17, 2017.

[5] J. S. Domínguez-Menchero, J. Rivera, and E. Torres-Manzanera, "Optimal purchase timing in the airline market," *J. Air Transp. Manage.*, vol. 40, pp. 137–143, Aug. 2014.

[6] V. H. Vu, Q. T. Minh, and P. H. Phung, "An airfare prediction model for developing markets," in *Proc. Int. Conf. Inf. Netw. (ICOIN)*, Jan. 2018, pp. 765–770.

[7] Y. Xu and J. Cao, "OTPS: A decision support service for optimal airfare ticket purchase," in *Proc. IEEE Int. Conf. Big Data*, Dec. 2017, pp. 1363–1368.

[8] E. Santana, S. Mastelini, and S. Jr, "Deep regressor stacking for air ticket prices prediction," in *Proc. Anais do Simpósio Brasileiro de Sistemas de Informação (SBSI)*, May 2017, pp. 25–31.

[9] J. A. Abdella, N. Zaki, K. Shuaib, and F. Khan, "Airline ticket price and demand prediction: A survey," *J. King Saud Univ.-Comput. Inf. Sci.*, to be published. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S131915781830884X?via%3Dihub

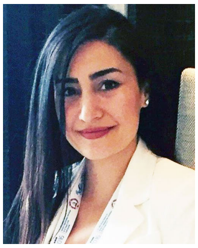
[10] X. Chen, S. Peng, J. Z. Huang, F. Nie, and Y. Ming, "Local PurTree spectral clustering for massive customer transaction data," *IEEE Intell. Syst.*, vol. 32, no. 2, pp. 37–44, Mar. 2017.

[11] S. Maldonado, C. Bravo, J. López, and J. Pérez, "Integrated framework for profit-based feature selection and SVM classification in credit scoring," *Decis. Support Syst.*, vol. 104, pp. 113–121, Dec. 2017.

[12] S. Ozan and L. O. Iheme, "Artificial neural networks in customer segmentation," in *Proc. 27th Signal Process. Commun. Appl. Conf. (SIU)*, Apr. 2019, pp. 1–4.

[13] R. J. Kuo, L. M. Ho, and C. M. Hu, "Integration of self-organizing feature map and K-means algorithm for market segmentation," *Comput. Oper. Res.*, vol. 29, no. 11, pp. 1475–1493, Sep. 2002.

[14] F.-M. Hsu, L.-P. Lu, and C.-M. Lin, "Segmenting customers by transaction data with concept hierarchy," *Expert Syst. Appl.*, vol. 39, no. 6, pp. 6221–6228, May 2012.
[15] D. Cheng, Q. Zhu, J. Huang, Q. Wu, and L. Yang, "Clustering with local density peaks-based minimum spanning tree," *IEEE Trans. Knowl. Data Eng.*, vol. 33, no. 2, pp. 374–387, Feb. 2021.
[16] M. Du, S. Ding, Y. Xue, and Z. Shi, "A novel density peaks clustering with sensitivity of local density and density-adaptive metric," *Knowl. Inf. Syst.*, vol. 59, no. 2, pp. 285–309, May 2019.
[17] M. Du, S. Ding, and Y. Xue, "A robust density peaks clustering algorithm using fuzzy neighborhood," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 7, pp. 1131–1140, Jul. 2018.
[18] M. Du, S. Ding, X. Xu, and Y. Xue, "Density peaks clustering using geodesic distances," *Int. J. Mach. Learn. Cybern.*, vol. 9, no. 8, pp. 1335–1349, Aug. 2018.
[19] X. Chen, J. Z. Huang, and J. Luo, "PurTreeClust: A purchase tree clustering algorithm for large-scale customer transaction data," in *Proc. IEEE 32nd Int. Conf. Data Eng. (ICDE)*, May 2016, pp. 661–672.
[20] A. A. Akinduko, E. M. Mirkes, and A. N. Gorban, "SOM: Stochastic initialization versus principal components," *Inf. Sci.*, vols. 364–365, pp. 213–221, Oct. 2016.
[21] W. L. Chang, L. M. Pang, and K. M. Tay, "Application of self-organizing map to failure modes and effects analysis methodology," *Neurocomputing*, vol. 249, pp. 314–320, Aug. 2017.
[22] K. Tasdemir, P. Milenov, and B. Tapsall, "Topology-based hierarchical clustering of self-organizing maps," *IEEE Trans. Neural Netw.*, vol. 22, no. 3, pp. 474–485, Mar. 2011.
[23] G. C. Cardarilli, L. Di Nunzio, R. Fazzolari, M. Re, and S. Spanò, "AW-SOM, an algorithm for high-speed learning in hardware self-organizing maps," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 2, pp. 380–384, Feb. 2020.
[24] L. E. Brito da Silva and D. C. Wunsch, "An information-theoretic-cluster visualization for self-organizing maps," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2595–2613, Jun. 2018.
[25] M. Z. Hanafi, "Segmentation of customers' experiences of YouTube streaming application users in South Jakarta using K-means method," in *Proc. Int. Conf. Smart Technol. Appl. (ICoSTA)*, Feb. 2020, pp. 1–5.
[26] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, "Efficient kNN classification with different numbers of nearest neighbors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1774–1785, May 2018.
[27] M. T. Ballestar, P. Grau-Carles, and J. Sainz, "Customer segmentation in e-commerce: Applications to the cashback business model," *J. Bus. Res.*, vol. 88, pp. 407–414, Jul. 2018.
[28] S. S. Güllüoğlu, "Segmenting customers with data mining techniques," in *Proc. 3rd Int. Conf. Digit. Inf., Netw., Wireless Commun. (DINWC)*, Feb. 2015, pp. 154–159.
[29] H.-P. Kriegel, E. Schubert, and A. Zimek, "The (black) art of runtime evaluation: Are we comparing algorithms or implementations?" *Knowl. Inf. Syst.*, vol. 52, no. 2, pp. 341–378, Aug. 2017.
[30] A. Coates and A. Y. Ng, "Learning feature representations with k-means," in *Neural Networks: Tricks of the Trade*. Berlin, Germany: Springer, 2012, pp. 561–580.
[31] F. Nielsen, "Hierarchical clustering," in *Introduction to HPC With MPI for Data Science*. Cham, Switzerland: Springer, 2016, pp. 195–211.
[32] O. Maimon and L. Rokach, *Data Mining and Knowledge Discovery Handbook*, 2nd ed. Boston, MA, USA: Springer, 2005.
[33] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD*, 1996, vol. 96, no. 34, pp. 226–231.
[34] E. Schubert, J. Sander, M. Ester, H. P. Kriegel, and X. Xu, "DBSCAN revisited, revisited: Why and how you should (Still) use DBSCAN," *ACM Trans. Database Syst.*, vol. 42, no. 3, pp. 1–21, Aug. 2017.
[35] M. Ankerst, M. M. Breunig, H.-P. Kriegel, and J. Sander, "OPTICS: Ordering points to identify the clustering structure," *ACM SIGMOD. Rec.*, vol. 28, no. 2, pp. 49–60, Jun. 1999.
[36] H. K. Kanagala and V. V. Jaya Rama Krishnaiah, "A comparative study of K-means, DBSCAN and OPTICS," in *Proc. Int. Conf. Comput. Commun. Informat. (ICCCI)*, Jan. 2016, pp. 1–6.
[37] P. J. Rousseeuw, "Silhouettes: A graphical aid to the interpretation and validation of cluster analysis," *J. Comput. Appl. Math.*, vol. 20, pp. 53–65, Nov. 1987.
[38] R. C. de Amorim and C. Hennig, "Recovering the number of clusters in data sets with noise features using feature rescaling factors," *Inf. Sci.*, vol. 324, pp. 126–145, Dec. 2015.
[39] D. L. Davies and D. W. Bouldin, "A cluster separation measure," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. PAMI-1, no. 2, pp. 224–227, Apr. 1979.
[40] D. Moulavi, P. A. Jaskowiak, R. J. Campello, A. Zimek, and J. Sander, "Density-based clustering validation," in *Proc. SIAM Int. Conf. Data Mining*. Philadelphia, PA, USA: SIAM, 2014, pp. 839–847.
[41] T. M. Kodinariya and P. R. Makwana, "Review on determining number of cluster in K-means clustering," *Int. J.*, vol. 1, no. 6, pp. 90–95, 2013.
[42] M. Syakur, B. Khotimah, E. Rochman, and B. Satoto, "Integration k-means clustering method and elbow method for identification of the best customer profile cluster," in *Proc. IOP Conf. Ser., Mater. Sci. Eng.*, vol. 336, no. 1. Bristol, U.K.: IOP Publishing, 2018, Art. no. 012017.
[43] G. E. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 15, 2002, pp. 857–864.
[44] T. DeLise, "Data segmentation via t-SNE, DBSCAN, and random forest," 2020, *arXiv:2010.13682*. [Online]. Available: http://arxiv.org/abs/2010.13682
[45] M. C. Cieslak, A. M. Castelfranco, V. Roncalli, P. H. Lenz, and D. K. Hartline, "T-distributed stochastic neighbor embedding (t-SNE): A tool for eco-physiological transcriptomic analysis," *Mar. Genomics*, vol. 51, Jun. 2020, Art. no. 100723.



ZHALEH SADREDDINI (Member, IEEE) received the Ph.D. degree in computer engineering from Karadeniz Technical University, Turkey, in 2018. From 2018 to 2020, she had been an Assistant Professor with the Department of Computer Engineering, Istanbul Arel University. She had been a Visiting Professor with Carleton University, Canada, in 2019, collaborating with Prof. Halim Yanikomeroglu on resource allocation in 5G. She is currently an Experienced Researcher - Networks with Ericsson Research. Her research interests include resource management, overbooking strategies, ML, and MCDM.

ILKNUR DONMEZ received the B.S. degree from the Electronic and Communication Department, Istanbul Technical University, Turkey, the M.S. degree in mathematics from Ege University, Turkey, and the Ph.D. degree from the Computer Engineering Department, Istanbul Technical University. She is currently an Assistant Professor with the Department of Computer Engineering, Istanbul Arel University, Turkey. Her research interests include data science, machine learning, and natural language processing.

HALIM YANIKOMEROGLU (Fellow, IEEE) is currently a Full Professor with the Department of Systems and Computer Engineering, Carleton University, Canada. His group has made substantial contributions to 4G and 5G wireless research. His collaborative research with industry has resulted in 37 granted patents. He is a Distinguished Lecturer of the IEEE Communications Society and the IEEE Vehicular Technology Society. He is the Chair of the Steering Committee of the IEEE Wireless Communications and Networking Conference (WCNC).