Integer Programming Approach for Distribution Shift Detection: A case study of Credit scoring

Mai Anh Bui Thi ⋆ Quoc Trung Bui ⋆ Ngoc Dung Nguyen ⋆ Thanh Hung Pham ⋆
Hoang Duong Chu ⋆
⋆ Hanoi University of Science and Technology

Abstract

Data discretization is a crucial technique in data mining and knowledge discovery processes. Its primary objective is to convert continuous attributes into discrete ones by assigning categorical values to intervals, resulting in the transformation of quantitative data into qualitative data. This paper proposes a novel discretization approach using integer programming, which is then used for calculating Population Stability Index (PSI) to monitor the stability of a population and particularly applying in the context of FICO Score. Traditional binning methods such as Equal Width Binning (EWB) and Equal Frequency Binning (EFB) have limitations such as sensitive to outliers, creating redundant bins and hard to detect minor changes in the population. The proposed method overcomes these limitations, offers a more accurate and precise way to monitor population stability. We experiment with pseudo generated data and the results demonstrate the superior performance of our proposed method compared to EWB and EFB, enabling better decision-making and risk management for credit bureaus and financial institutions. The proposed method offers a practical solution for monitoring population stability in other fields as well.

Index terms: *Data discretization, EWB, EFB, integer programming, PSI, FICO Score.*

1 Introduction

Credit scores are a critical factor in determining an individual's creditworthiness and access to financial products. The most widely used credit score in the United States is the FICO Score, which is calculated based on an individual's credit history and other relevant data. As FICO Scores play such a critical role in financial decision-making, it is important to monitor their stability over time, particularly in situations where there are changes to the population being scored.

Population Stability Index (PSI) is a widely used measure for monitoring changes in the distribution of a population over time. PSI is calculated by dividing the population into bins, or groups, based on their FICO Score, and then comparing the distribution of scores in the current period to a reference period. If the distribution of scores in the current period differs significantly from the reference period, it may indicate that there has been a change in the population being scored.

The choice of binning method can affect the result of PSI significantly. However, the traditional binning methods used in PSI calculations, such as Equal Width Binning (EWB) and Equal Frequency Binning (EFB), have some limitations as discussed in [1]. Furthermore, these methods require a predefined number of bins but in [2, 3], the authors showed that PSI with too many bins can detect minor changes in the distribution and often causes false alarm while too few bins can make it miss differences. In this paper, we propose a new approach to binning, which uses integer programming to identify optimal cut points for discretizing the data. This approach aims to address the limitations of EWB and EFB, by providing more precise and robust binning method and we experiment it in the context of FICO Score and PSI.

The remainder of the paper is organized as follows. In Section 2, we review prior research on PSI and binning methods for calculating PSI including EWB and EFB. In Section 3, we describe the proposed method in detail, including the mathematical formulation of the optimization problem and the parameters used in the optimization. In Section 4, we present experimental results comparing the performance of the proposed method to EWB and EFB, and discuss the potential trade-offs of the proposed method. Finally, in Section 5, we conclude with a summary of the contribution of the proposed method and potential avenues for future research.

2 Related Work

Population Stability Index. The Population Stability Index (PSI), which had studied in many researches [4, 5, 6], measures the distributional differences between two populations over time or across different segments. It is commonly used in the context of credit risk and marketing analytics, to track changes in the characteristics of a customer base over time or between different segments. PSI is calculated by comparing the expected and observed distributions of a variable using the formula:

$$PSI = \sum((Expected\ \% - Observed\ \%) \times \ln(\frac{Expected\ \%}{Observed\ \%})) \quad (1)$$

As mentioned in [6], the resulting PSI value indicates the extent to which the distribution has changed, with values below 0.1 considered low, 0.1-0.25 considered moderate, and above 0.25 considered high, suggesting significant differences between the populations.

Data Discretization. Discretization is the process of transforming continuous variables into discrete features by creating a series of consecutive intervals that cover the full range of values for the variable. These discrete values are subsequently treated as categorical data. It is possible to conduct data discretization with either supervised or unsupervised methods, meaning that class labels may or may not be utilized [7, 8]. In the context of unsupervised univariate data discretization, two most commonly used approaches are Equal Width Binning (EWB) and Equal Frequency Binning (EFB). The EWB method involves dividing the value range into equally sized bins, EWB is better for graphical representations (histograms) and is very straighforward, but it have problems if the data is not evenly distributed, it's sparse, or has outliers, as you will have many empty, useless bins [1]. EFB will guarantee that every bin contains roughly the same amount of data, which can be useful for handling skewed distributions or outliers, but it might lose some information about the shape of the distribution or create artificial boundaries between bins [1]. amsmath

3 Methodology

Problem Formulation. In this part, we propose our integer programming model aiming to find the optimal cut points for data discretization in the context of FICO Score and PSI. Given a population P with N observations (for instance, N people with their corresponding FICO Score), we denote $V = [v_0, v_1, v_2, ..., v_{n-2}, v_{n-1}, v_n, v_{n+1}]$ is the array of $n+2$ possible values in P where $v_0 < v_1 < v_2 < ... < v_{n-1} < v_n < v_{n+1}$. We first calculate the histogram array of this population with bin edges V: $h = [h_0, h1, ..., h_{n-1}, h_n]$ where h_0 is the the percentage of observations taking the value of v_0, h_1 is the percentage of observations taking the value of v_1 , h_{n-1} is the percentage of observations taking the value of h_{n-1} and finally, h_n is the percentage of observations taking the value of v_n or v_{n+1}. From this array, we obtain the histogram matrix H of size $(n+2, n+2)$ of this distribution as follow:

$\forall 0 \leq j \leq n, 0 \leq k \leq n+1$:

$$H[j,k] = \begin{cases} \sum h[j:k] & \text{if } j > k \\ 0 & \text{otherwise} \end{cases}$$

Here, $H[j,k]$ is the percentage of the population taking the value in the half-closed interval $[v_j, v_k)$. Note that we take the closed interval for each element in the last column of H, i.e., $H[j, n+1]$ is the proportion of the population taking the value in the closed interval $[v_j, v_{n+1}]$. Clearly, H is a upper triangular matrix with all elements on/below the its diagonal be 0.

We track the population P over M time steps, e.g. M days. The label array y of size $M-1$ is a binary array: If there is no significant change in the distribution between two consecutive time steps (normal event), the label of them is 0. Otherwise, 1 is assigned as their label (abnormal event). The histogram matrices H_0,H_1, ... , H_{M-1} are also calculated, where H_0 is the histogram matrix of the first time step, H_1 is the the histogram matrix of the second time step, etc. After that, a 3-D array PSI of size $(M-1, n+2, n+2)$ is obtained by applying the equation (1) in element-wise manner to the histogram matrices of two consecutive time steps:

$$\forall 0 \leq i \leq M-2 : PSI[i] = H_{i+1} - H_i$$

We denote C_1 and C_0 is the element-wise addition of the PSI matrix of all normal or abnormal event, i.e. two successive time steps which have significant or insignificant shift in their distributions, respectively.

$$C_1 = \sum_{\odot, y[i]=1} PSI[i]; C_0 = \sum_{\odot, y[i]=0} PSI[i]$$

where $\sum_{\odot}$ is the element-wise addition operation. Then, C_0 and C_1 are divided by the number of two consecutive time steps having slight or significant changes so as to overcome the problem of imbalanced distribution between the two kind of events.

Decision Variables. The candidate solution for our problem is a binary matrix X of size $(n+2, n+2)$ where $X[j,k]$ is equal to 1 or 0, indicating that the interval $[v_j, v_k)$ is selected or not. In other words, whether or not both v_j and v_k are chosen as final cut points. It is straight forward that X is also a upper triangular matrix like H since we can not choose a bin $[v_j, v_k)$ where $k \leq j$. We denote the set of all feasible solutions is $|X|$ and the state of X that produce the optimal cut points (optimal solution) is X^*.

Constraints. In this part, we introduce some constraints for X to satisfy the problem. Assume that the smallest and largest number of bins are min_bins and max_bins, respectively, we have the following constraints:

$$\sum_{1 \leq k <= n+1} X_{0,k} = 1 \tag{2}$$

$$\sum_{0 \leq j \leq n} X_{j,n+1} = 1 \tag{3}$$

$$\sum_{0 \leq j \leq k-1} X_{j,k} \leq 1, \forall 1 \leq k \leq n \tag{4}$$

$$\sum_{j+1 \leq k \leq n+1} X_{j,k} \leq 1, \forall 1 \leq j \leq n \tag{5}$$

$$\sum_{0 \le i' \le i-1} X_{i',i} = \sum_{i+1 \le i" \le n+1} X_{i,i"} \quad (6)$$

$$\sum_{0 \le j,k \le n+1} X_{j,k} \ge min\_bins \quad (7)$$

$$\sum_{0 \le j,k \le n+1} X_{j,k} \le max\_bins \quad (8)$$

In this model, the constraint (2) ensures that the minimum value in the distribution (v_0) together with exactly one other value from v_1 to v_{n+2} are selected as final cut points. The similar condition for the maximum value in the distribution is hold by the constraint (3). The two constraints (4) and (5) indicate that there are at most one element having the value of 1 in each row and column of X (except for the first row and the last column) since we can not choose the bin $[v_j, v_{k'})$ when we had chosen a bin $[v_j, v_k)$ before (in case $k' \neq k$). Beside that, the constraint (6) makes sure that if one point had been chosen as the end of a bin, it must be the beginning of the latter bin or in other words, there is no break in our cut points. Finally, the solution will delight our desirable number of bins by performing the two last constraints.

Objectives. In this study, two objective functions were defined for the optimization problem of discretizing data using integer programming. The first objective Obj_1 aims to maximize the Population Stability Index (PSI) when there is a significant shift in the distributions between two consecutive time steps. This objective is important because it allows the detection of significant changes in the population over time.

$$\max_{|X|} Obj_1 = \max_{|X|} \sum_{j,k} C_1 \odot X \quad (9)$$

Where $\sum_{j,k}$ means the sum of all element in the matrix and $\odot$ denotes the element-wise product.

The second objective Obj_0 aims to minimize the PSI when there is no or slight change in distributions between two successive time steps. This objective is important because it ensures that the method does not unnecessarily create new bins when there is no significant change in the population.

$$\min_{|X|} Obj_0 = \min_{|X|} \sum_{j,k} C_0 \odot X \quad (10)$$

To balance these two objectives, a hyper-parameter α was introduced to combine them into a single objective function. The value of α can be tuned to find the optimal compromise between the two objectives.

$$\max_{|X|} Total\_Obj = \alpha \times Obj_1 - (1 - \alpha) \times Obj_0 \quad (11)$$

In summary, our proposed integer programming model with a given α (α in range 0-1 with step 0.05), which can be tuned, aims at finding X^* in the feasible solution space $|X|$ that maximize the $Total\_Obj$ as follow:

$$X^* = \arg\max_{|X|} Total\_Obj$$

$$s.t. (2), (3), (4), (5), (6), (7), (8).$$

4 Empirical Analysis

- The proposed algorithm dominates the traditional algorithms (Equal-width and Equal-Frequency binning) in terms of the objective values and the number of detected events (There are errors in data reflecting by the shift of distribution)

Pseudo Data Generation. In this section, we present the pseudo-data generation method for experimenting and evaluating our discretization algorithm. As credit scoring data usually records on a large number of customers and lasts for months to years, our generation method must ensure scalability, robustness and follow the natural flow of real-world generated data. The idea is as follows, the data will be generated sequentially day by day. Each day will follow a certain distribution with 2 possibilities: shift little (True) or shift much (False) compared to the previous day. To do this, the next day is generated by randomly replacing some current data points with a new sample. Compared to the current day, this new sample has the same distribution but varies significantly on mean and variance that can assure data shifting. The ratio of replacing data points is more or less based on generating a false day or a true day. Beside, we combine 4 data shifting detection tests: Kullback-Leibler divergence [9], Population Stability Index, Wasserstein distance [10], Jensen-Shannon distance [11] for ensuring significant deviation of the false day. Furthermore, we extend data complexity by minimizing the gap between true and false days. Because the data points must be integer and range between 300 and 850, some final post-processing steps are performed like rounding and clamping. As the result, the pseudo data is generated with 4 different types of distribution: Normal, Gamma, Logistic, Uniform as shown in the Figure 1. Each data set (both training and test data set) is generated with the ratio between the class True and False of 0.7 or 0.8 or 0.9; number of days varied from 1 month to 3 years and number of samples from one thousand to one millions. Moreover, we only generated one data set per one combination of (distribution, class ratio, number of days, number of samples) when generating training data

sets while we repeated the generating process 5 times for each combination when creating test data sets, with the aim to reduce the randomness in the test data sets.

Evaluation Metrics. In order to compare the performance of our proposed method to EWB and EFB, we employed seven metrics including Preparing Data Time, Solving Time, Total Objective Value, Accuracy, Precision_0, Recall_1 and Inverse Weighted F2. While the first four metrics give some overview of algorithms's performance, other report a more detail look at the result of classifying normal (class 0 or negative class) and abnormal events (class 1 or positive class).

- *Preparing Data Time:* This metric measures how long (in seconds) to calculate two matrices C_0 and C_1 in our proposed method from the raw distribution. We only reported this metric for our method because EWB and EFB does not require this step.

- *Solving Time:* The solving time (in seconds) measures the computational performance of an algorithm. In our method, this is the time required to find the optimal solution X^* given two matrices C_0 and C_1. On the other hand, in terms of EWB and EFB, this metric refer to the time to calculate the cut points directly from the distribution.

- *Objective Values:* Given the cut points, the Objective_0, Objectiev_1 and Total Objective can be calculated using three equations (9), (10) and (11). It reflect the effectiveness of our proposed method in balancing the two objectives of maximizing the PSI for significant changes and minimizing the PSI for no or slight changes. We compared this metric of our model to EWB and EFB in training data set and all test data sets.

- *Accuracy:* In binary classification, accuracy is the number of correct predictions made as a ratio of all predictions made. In imbalanced classification, where the number of instances in one class is significantly higher than the other class(es), accuracy can be a misleading metric to evaluate the performance of a model. This is because a classifier that always predicts the majority class will have a high accuracy, even though it does not perform well in identifying the minority class.

- *Precision_0:* This is the precision of class 0 (two consecutive time steps with no or slight changes). This metric tell how well the algorithm at distinguishing normal events from abnormal events.

- *Recall_1:* This is the recall of class 1 (i.e. two consecutive time steps with considerable changes). The higher this metric is, the higher the chance that the algorithm can detect abnormal events.

- *Inverse_Weighted_F2:* The F2-measure is an example of the Fbeta-measure, which is the harmonic mean of the precision and recall, with a β value of 2.0. The F2 score is a useful metric when the goal is to achieve high recall while maintaining reasonable precision, which is often the case in imbalanced data sets where the positive class is rare and typically used in cases where False Negatives are considered worse than False Positives. The F2 score of each class is calculated separately and we need to average them to get a single metric. This can be done through several ways [**?**], but none of them consider the problem of imbalanced data. Therefore, we proposed a new way to deal with this issue: Suppose n, n_0 and n_1 are the number of observations of the population, class 0 and class 1 respectively, the *Inverse Weighted F2* is calculated as follow:

$$Inverse\_Weighted\_F2 = \frac{n_1}{n} \times F2_0 + \frac{n_0}{n} \times F2_1$$

where $F2_0$ and $F2_1$ are respectively the F2 score of class 0 and class 1.

Empirical Settings. All the reported results are the average of all training or test data sets. While experimenting with EWB and EFB, the number of bins varied from 5 to 25 and the best testing results were reported. Also, we choose the hyperparameter α in our IP model in the range from 0 to 1 with a step of 0.05. To solve the proposed integer programming model for feature selection, we use scip[1] and OR-Tools [2].

Results. We present our experimental results through the following research questions.

1) *The proposed algorithm is efficient in terms of solving large-size data sets?* In order to analyze this criteria, we compared the *Preparing Data Time* and *Solving Time* of our proposed method with those of EFB and EWB. Regarding the time for preparing data, it is of $O(n)$ w.r.t the number of days in training data set and ranges from ... (s) for 30 days to ... (s) for 1095 days. While required time for solving of EFB and EWB are inconsiderable (smaller than 0.001s), out proposed IP model require around 850s to train and it does not influenced neither by the distribution nor the number of samples / days in the training data set. It shows that it takes reasonable time to get the discretization solution by using our proposed method in large-scale data sets.

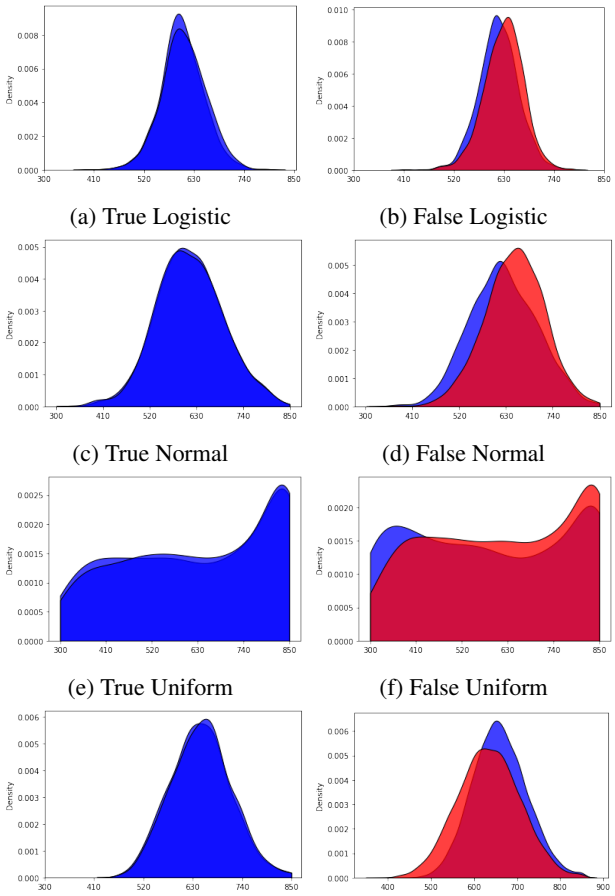
2) *The proposed algorithm dominates the traditional algorithms (EFB and EWB) in terms of the objective values and the classification performance (Are there errors in data reflecting by the shift of distribution or not?).*

[1]https://www.scipopt.org

[2]https://developers.google.com/optimization/

- *Objective Values:* The Figure 2 shows the changes of Obj_0 and Obj_1 while changing α. The two objectives increase when α go up in training phase as well as test phase although the values for both objectives in test data are lower than that in training data. As illustrated in the Figure 3, the $Total\_Obj$ value of all three binning methods follow a similar trend in both training and test data: Starting with a very small negative value, it grows rapidly when alpha increases from 0 to 0.15 after falling off back to a small positive value. Then, a slowly increment in the $Total\_Obj$ followed by a fluctuation when α reaches to 0.5. It is obvious that all methods got the highest $Total_Obj$ value when α is 0.95. Moreover, despite it get a lower value in some test data set, our IP model often achieved better $Total\_Obj$ in all training and test data.
- *Classification Performance:* In order to compare the classification performance when applying three binning methods, we examined three metrics: *Precision_0*, *Recall_1* and *Inverse_Weighted_F2* as shown above. The table 1 gives the overview of the classification when applying binning methods to classify normal/abnormal events. Overall, our IP model had a consistently superior performance in all metrics compared to EWB and EFB. While the three methods are competitive in terms of *Accuracy* and *Precision_0*, the scores of *Recall_1* and *Inverse_Weighted_F2*, which are more important metrics since they put more weight on abnormal events as well as take into account the imbalanced distribution between normal and abnormal events, especially looking at Uniform distribution (as shown in the **Pseudo Data Generation** part, it is similar to the FICO Score distribution in the real world[3]). Beside that, we further examined the classification performance of our IP model when changing α in Figure 4. We noticed that setting α be equal to 0 make poor performance because it make the model pay attention only on the normal events and ignore the abnormal ones, which is contradictory to our purpose of detecting abnormal events. Furthermore, although the consistent between *Accuracy* and *Precision_0* when training and test, there are considerable gaps between the scores of training/test *Recall_1* and *Inverse_Weighted_F2*, which indicate that our model tend to over-fit to training data. Finally, we explored the impact of the size of training data to the classification in the test data. As shown in Figure 5, due to not considering much information about the training data rather than the overall its histogram, the performance of EWB and EFB do not vary much when the size of training is altered. On the other hand, the effectiveness of our IP model is depended on the size of training data. When the size of training is small, the IP model easily fit to the data but not generalize very well on test data sets. When the size of training data increase, the model loose some of effectiveness in the training data but gain more performance when testing but there is still a significant gap between the results of training and testing, which may be a sign indicating that out IP model still suffer from over-fitting.

[3]https://www.fico.com/blogs/us-average-fico-score-hits-700-milestone-consumers



(a) True Logistic (b) False Logistic

(c) True Normal (d) False Normal

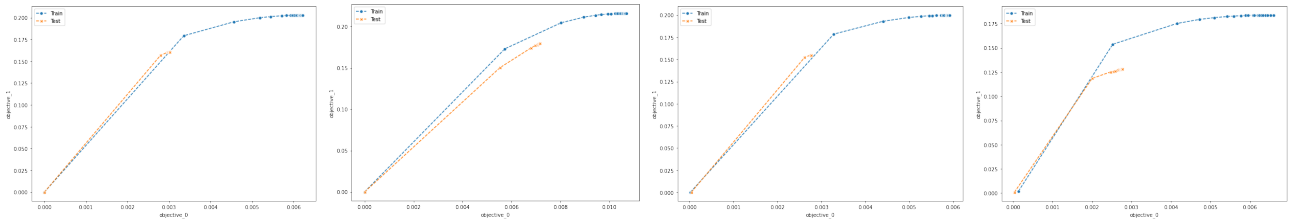
(e) True Uniform (f) False Uniform

(g) True Gamma (h) False Gamma

Figure 1: Pseudo generated data.

5 Conclusions and Future Work

In this paper, we have proposed a binary integer programming model for data discretization and applied it in the context of PSI and FICO Score. We first formulate



(a) Gamma (b) Logistic (c) Normal (d) Uniform

Figure 2: The changes of Objective_0 and Objective_1 when modifying α.

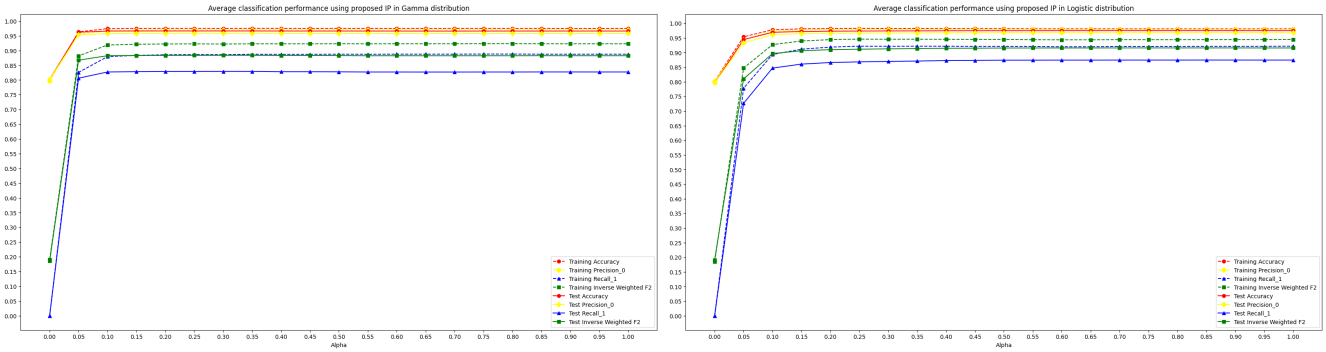
(a) Training Gamma (b) Test Gamma (c) Training Logistic (d) Test Logistic

(e) Training Normal (f) Test Normal (g) Training Uniform (h) Test Uniform

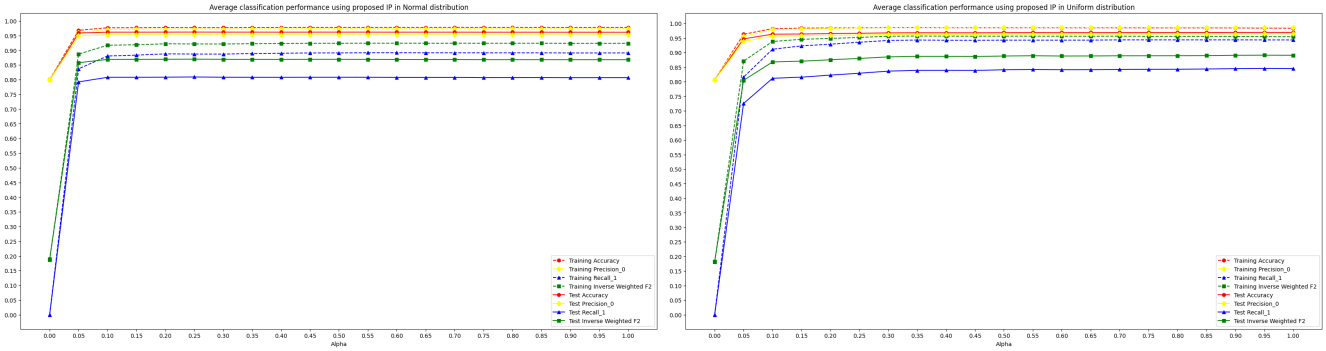
Figure 3: The changes of Objective_0 and Objective_1 when modifying alpha.

Distribution	Method	Accuracy		Recall_1		Precision_0		Inverse_Weighted_F2	
		Train	Test	Train	Test	Train	Test	Train	Test
Gamma	**EFB**	96.07	96.47	81.62	82.07	95.13	95.63	87.60	87.80
	EWB	96.30	96.44	82.77	81.80	95.43	95.59	88.32	87.59
	IP	**97.48**	**96.63**	**88.61**	**82.84**	**97.01**	**95.82**	**92.20**	**88.32**
Normal	**EFB**	96.16	96.05	80.08	80.21	95.23	95.09	86.08	86.50
	EWB	96.40	96.00	81.49	79.94	95.52	95.03	87.09	86.31
	IP	**97.72**	**96.18**	**88.72**	**80.83**	**97.23**	**95.25**	**92.16**	**86.92**
Uniform	**EFB**	92.72	92.49	62.97	60.70	91.72	91.34	73.20	71.70
	EWB	95.22	94.79	75.81	72.60	94.54	93.85	82.66	80.64
	IP	**98.43**	**96.94**	**94.37**	**84.43**	**98.66**	**96.42**	**95.50**	**89.05**
Logistic	**EFB**	97.32	**97.54**	87.14	**87.75**	96.66	**96.88**	91.33	**91.80**
	EWB	96.78	96.71	84.21	83.70	96.01	95.86	89.29	89.01
	IP	**98.06**	97.45	**92.08**	87.36	**97.89**	96.77	**94.39**	91.53

Table 1: Result of each discretization method for every distribution.

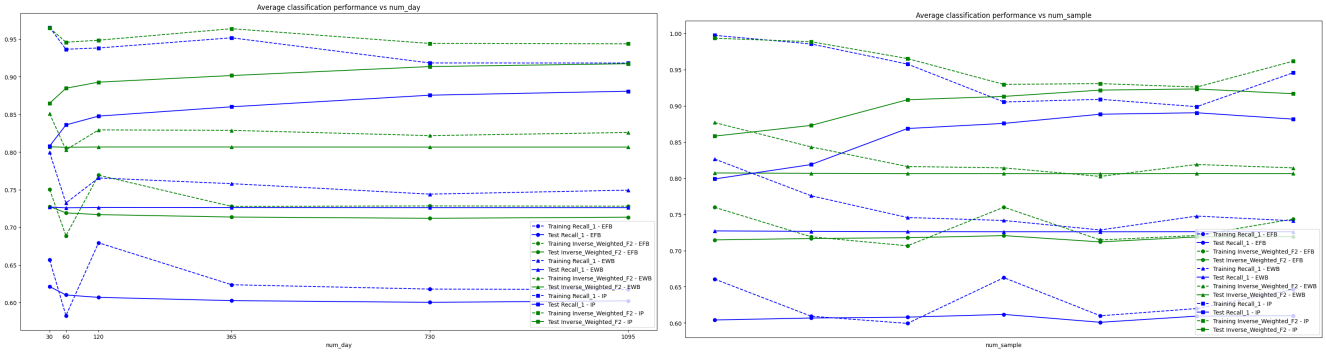


(a) Gamma (b) Logistic



(c) Normal (d) Uniform

Figure 4: The changes of classification performance when modifying α.



(a) Num Day (b) Num Sample

Figure 5: The influences of num_day and num_sample in *Recall_1* and *Inverse_Weighted_F2*.

the problem as a binary integer programming problem, then propose the objective function together with a set of linear constraints. We have empirically proved that our proposed model outperforms EWB and EFB in terms of total objective value and classification performance while keeping reasonable time for data preparation and solving.

References

[1] A. J. Ferreira and M. A. Figueiredo, "An unsupervised approach to feature discretization and selection," *Pattern Recognition*, vol. 45, no. 9, pp. 3048–3060, 2012.

[2] R. Taplin and C. Hunt, "The population accuracy index: A new measure of population stability for model monitoring," *Risks*, vol. 7, no. 2, p. 53, 2019.

[3] J. Du Pisanie and I. Visagie, "On testing the hypothesis of population stability for credit risk scorecards," *ORiON*, vol. 36, no. 1, pp. 19–34, 2020.

[4] B. Yurdakul, *Statistical properties of population stability index*. Western Michigan University, 2018.

[5] A. Becker and J. Becker, "Dataset shift assessment measures in monitoring predictive models," *Procedia Computer Science*, vol. 192, pp. 3391–3402, 2021.

[6] N. Siddiqi, *Credit risk scorecards: developing and implementing intelligent credit scoring*. John Wiley & Sons, 2012, vol. 3.

[7] S. Garcia, J. Luengo, J. A. Sáez, V. Lopez, and F. Herrera, "A survey of discretization techniques: Taxonomy and empirical analysis in supervised learning," *IEEE transactions on Knowledge and Data Engineering*, vol. 25, no. 4, pp. 734–750, 2012.

[8] S. Ramírez-Gallego, S. García, H. Mouriño-Talín, D. Martínez-Rego, V. Bolón-Canedo, A. Alonso-Betanzos, J. M. Benítez, and F. Herrera, "Data discretization: taxonomy and big data challenge," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 6, no. 1, pp. 5–21, 2016.

[9] I. Csiszár, "I-divergence geometry of probability distributions and minimization problems," *The annals of probability*, pp. 146–158, 1975.

[10] L. V. Kantorovich, "Mathematical methods of organizing and planning production," *Management science*, vol. 6, no. 4, pp. 366–422, 1960.

[11] B. Fuglede and F. Topsoe, "Jensen-shannon divergence and hilbert space embedding," in *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, 2004, pp. 31–.