Continuous Sigmoid Transform for Rule Threshold Optimization in Anti-Money Laundering

資訊管理與財務金融學系 財務金融碩士班
學生：吳茂嘉
指導教授：戴天時

Contents

- Introduction
- Related Work
- Data Description & Preprocessing
- Model
- Result
- Conclusion

Introduction

- Anti-Money Laundering
- Suspicious Activity Investigation
- Rule Based System

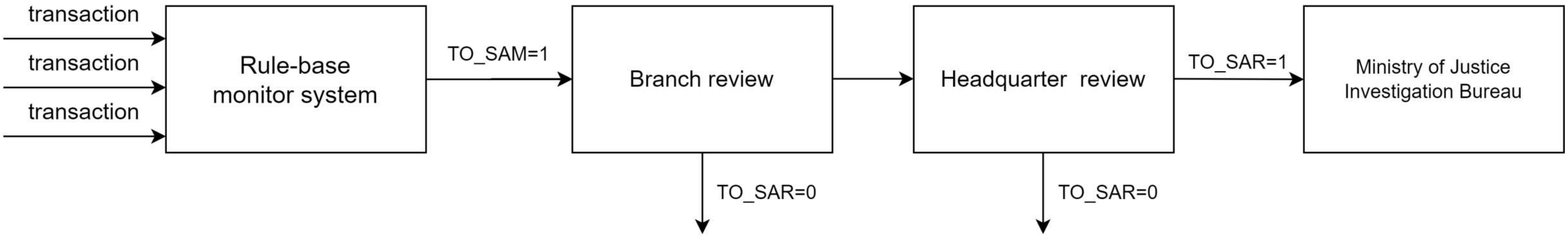
Anti-Money Laundering

- Money laundering is the illegal process of making large amounts of money generated by a criminal activity, such as drug trafficking or terrorist financing, appear to have come from a legitimate source
- The American Bankers Association report indicates that approximately $300 billion was laundered through the US in 2022
- In my thesis, we focus on transaction monitoring, which is a processes to monitor customer or account transactions in order to identify suspicious activity that may be tied to money laundering activities or other financial crimes
- To combat the money laundering, many regulations have been established by the Financial Supervisory Commission(FSC) and Ministry of Justice investigation Bureau in Taiwan

Suspicious Activity Investigation

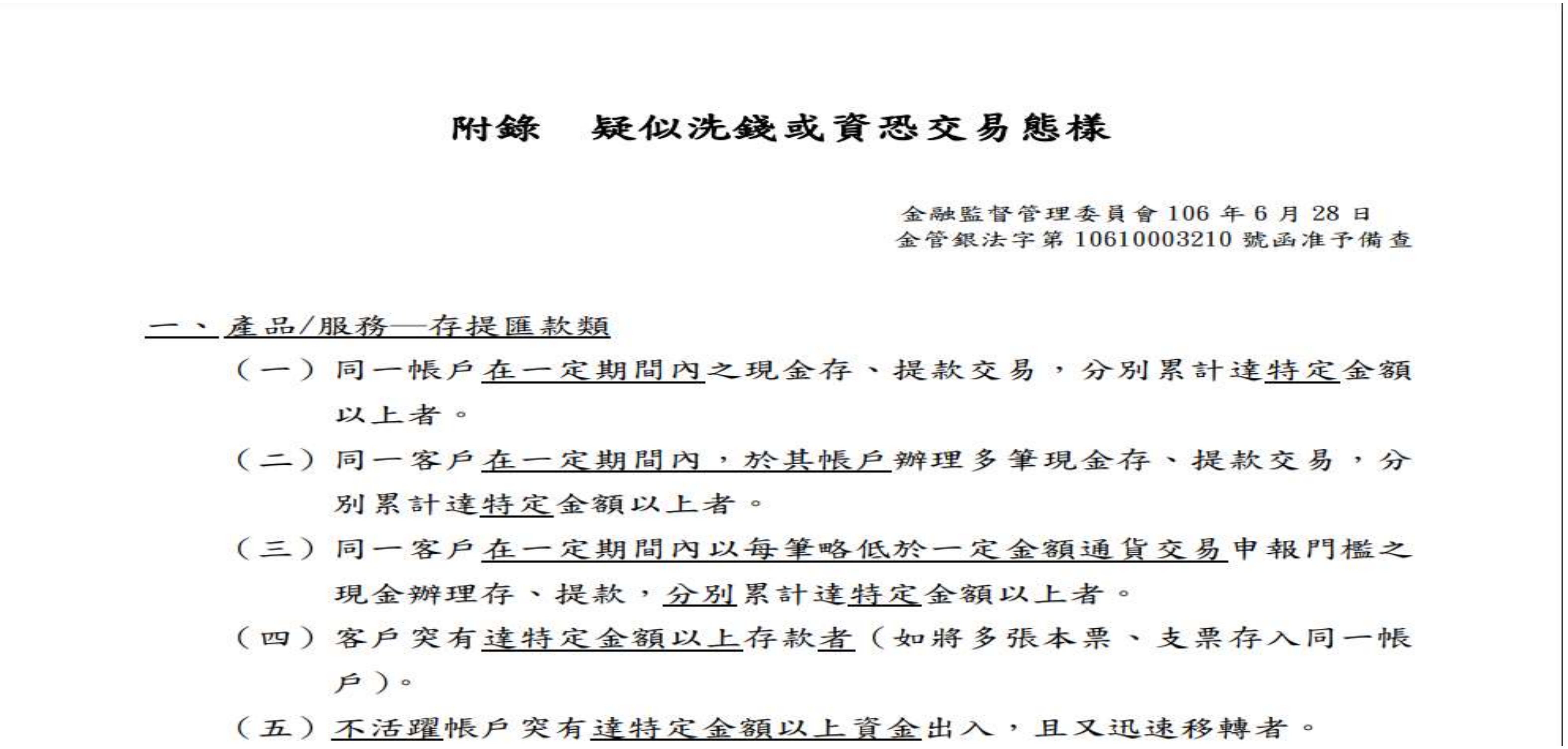
- Any transaction deemed suspicious by the rule-based system will be flagged as SAM=1. This transaction will then require at most two manual reviews by the clerk and compliance officer before being sent to the MJIB and classified as a Suspicious activity report (SAR)

The following picture show the procedure of suspicious activity investigation



Rule Based System

- Financial institutions are required to follow the abstract guidelines to design its own scenario rule
- We take TWN-A11-01 in Bank T to introduce how the rule work



附錄　疑似洗錢或資恐交易態樣

金融監督管理委員會 106 年 6 月 28 日
金管銀法字第 10610003210 號函准予備查

一、產品/服務─存提匯款類

（一）同一帳戶在一定期間內之現金存、提款交易，分別累計達特定金額以上者。

（二）同一客戶在一定期間內，於其帳戶辦理多筆現金存、提款交易，分別累計達特定金額以上者。

（三）同一客戶在一定期間內以每筆略低於一定金額通貨交易申報門檻之現金辦理存、提款，分別累計達特定金額以上者。

（四）客戶突有達特定金額以上存款者（如將多張本票、支票存入同一帳戶）。

（五）不活躍帳戶突有達特定金額以上資金出入，且又迅速移轉者。

Rule Based System

- Financial institutions are required to follow the abstract guidelines to design its own scenario rule
- We take TWN-A11-01 in Bank T to introduce how the rule work

TWN-A11-01:

樣態說明:

同一帳戶在一定期間內之現金存、提款交易，分別累計達到特定金額以上者

樣態規則:

同一帳戶於過去 15 日之

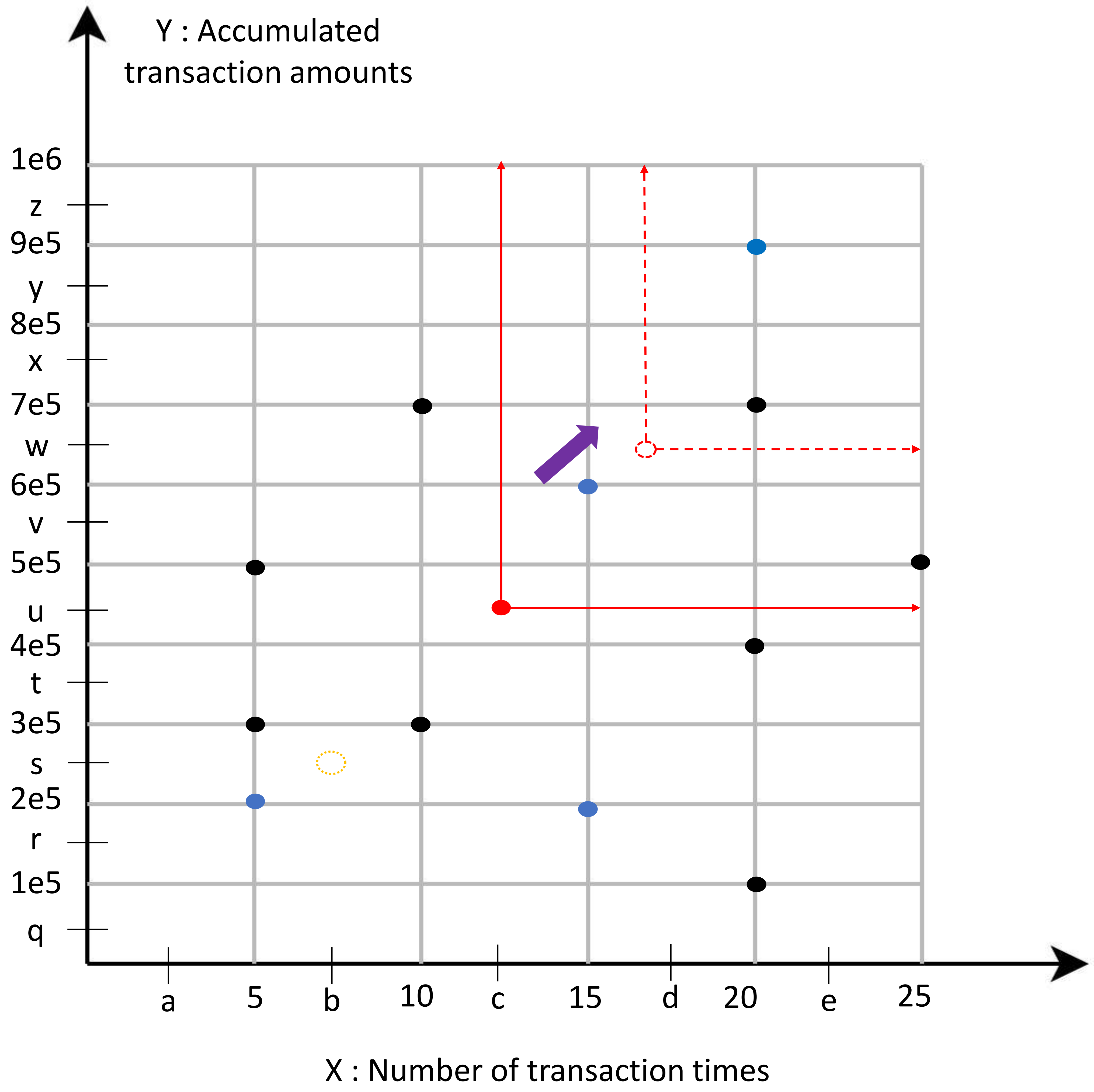
現金存款累計≥ A11_Credit_amount 元**且**次數≥ A11_Credit_count 次

或

現金提款累計≥ A11_Debit_amount 元**且**次數≥ A11_Debit_count 次

Rule Based System

- Consider a scenario that include only two features: Credit_amount and Credit_count
- The blue point denoted SAR=1 and the black point denoted SAR=0
- Red solid and dashed line are two different possible thresholds



Related Work

- Machine Learning with scenario rule
- Brute force based method

Machine Learning with scenario rule

- Numerous studies employ machine learning and deep learning models to identify unique patterns of fraud by analyzing customer information, transaction data and so on
  - Tang et al 2017. used the support vector machine with Radial basis function kernel to replace traditional predefined rules
  - Vassallo et al 2021. applied eXtreme Gradient Boosting with NCL-SMOTE to improve the recall in the transaction level
  - Alarab et al 2020. introduce the graph neural network to analyze the interconnections between transaction and illicit behavior
- In fact, many countries, like Taiwan, China, India, Hungary, Pakistan, etc., still require the implementation of the rule-based system due to model interpretability
- Only a few paper focus on combining the scenario rule and deep learning model together
  - Khan et al 2015. combined the scenario rule with Bayesian network together to detect suspicious transaction patterns. Every account are assigned their own Bayesian scores corresponding to the suspicious levels of their transactions

Brute force based method

- Both above paper applying brute force method in the final step to find the optimal thresholds. Badics et al. implemented the SIMD parallelization algorithm to accelerate the search of threshold among nearly 240,000 combinations by using GPU
  - Gupta et al 2022. focused on minimizing duplicate reports across scenario rules. They developed a two-stage model aimed at simultaneously optimizing the thresholds of various scenarios to reduce the frequency of behaviors repeatedly identified by different scenarios
  - Badics et al 2023. reformulated the threshold optimization problem to the microeconomics problem by considering the cost of missing fraudulent cases and manual reviews

Data Description & Preprocessing

- Data Description
- Data Preprocessing

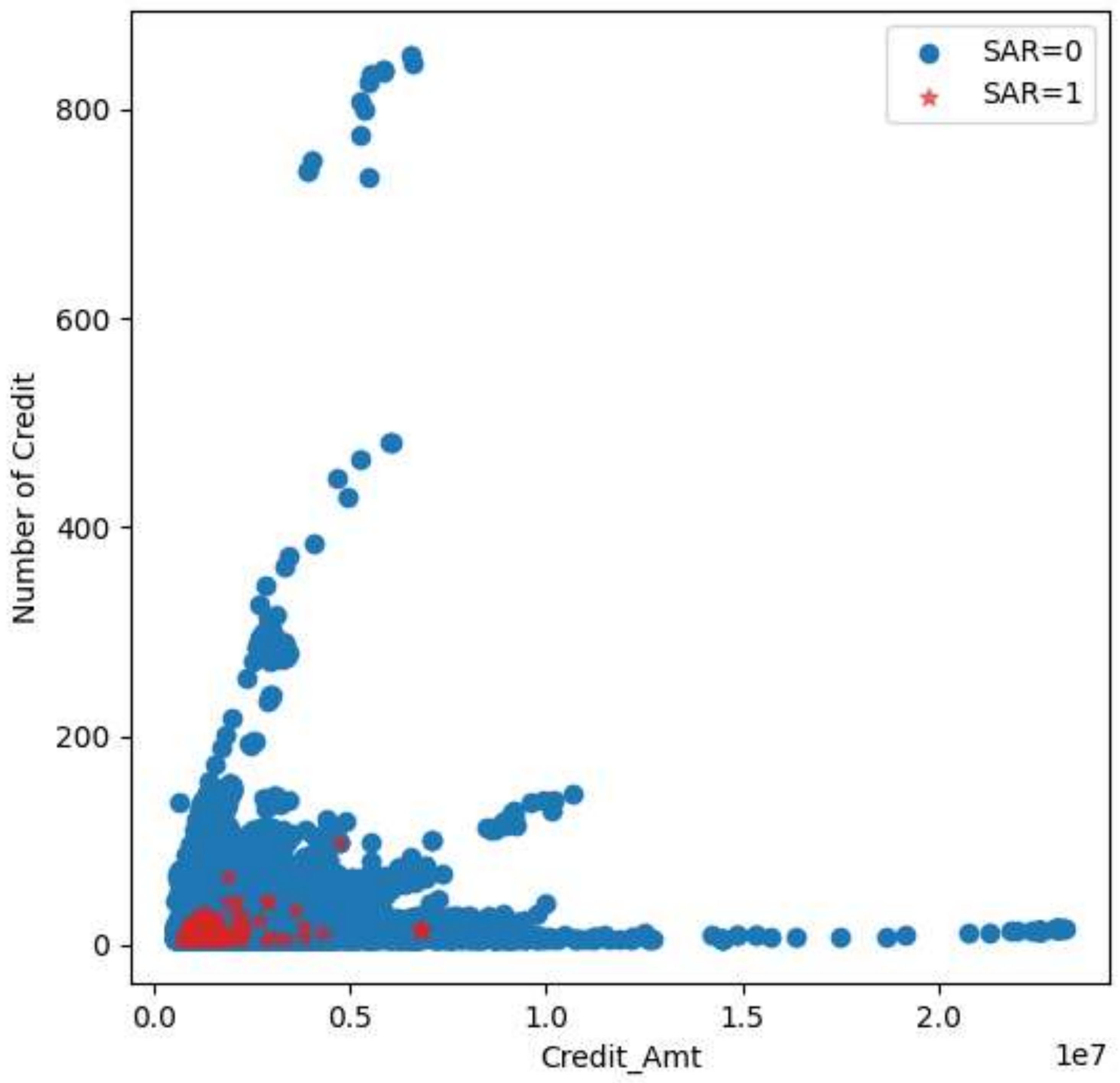
Data Description

- Our experiment use the real-world data from Bank T which is a commercial bank in Taiwan
- The raw transaction data start from 2019/03/04 to 2020/12/31 containing 179,426,210 transactions
- By the original thresholds settings, 446 SAR=1 cases in total 86770 cases
- We splits the dataset into three parts

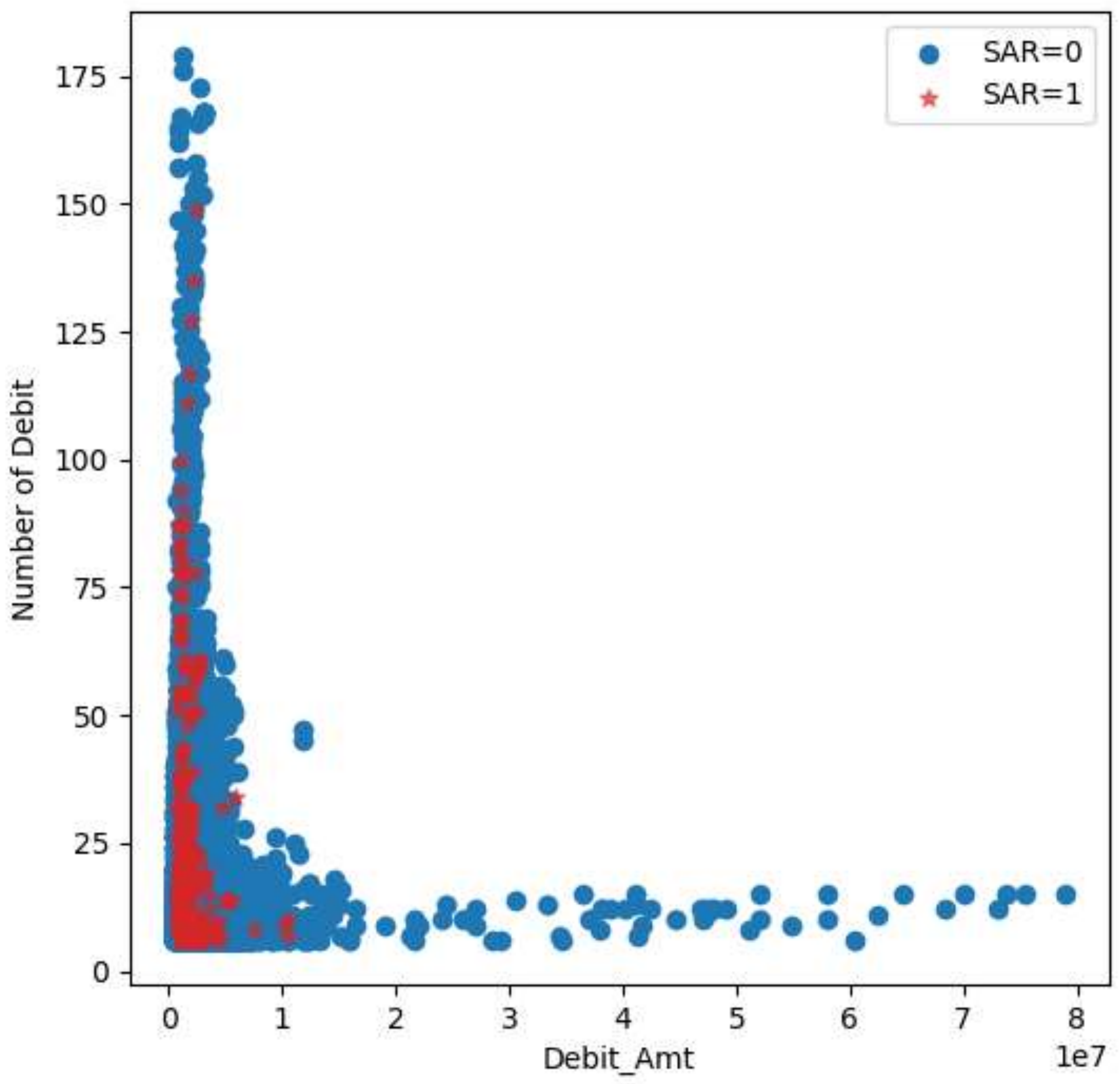
	SAR=1	SAR=0	Total
Train set 2019/03~2020/02	248	52449	52697
Valid set 2020/04~2020/07	99	18846	18945
Test set 2020/08~2020/12	99	15028	15127

Data Preprocessing

- The subsequent figures are the scatter plot of SAR=1 and SAR=0, which is categorized base on the credit and debit. Most of data are squeezed into the left bottom corner. SAR=1 and SAR=0 are mixed together



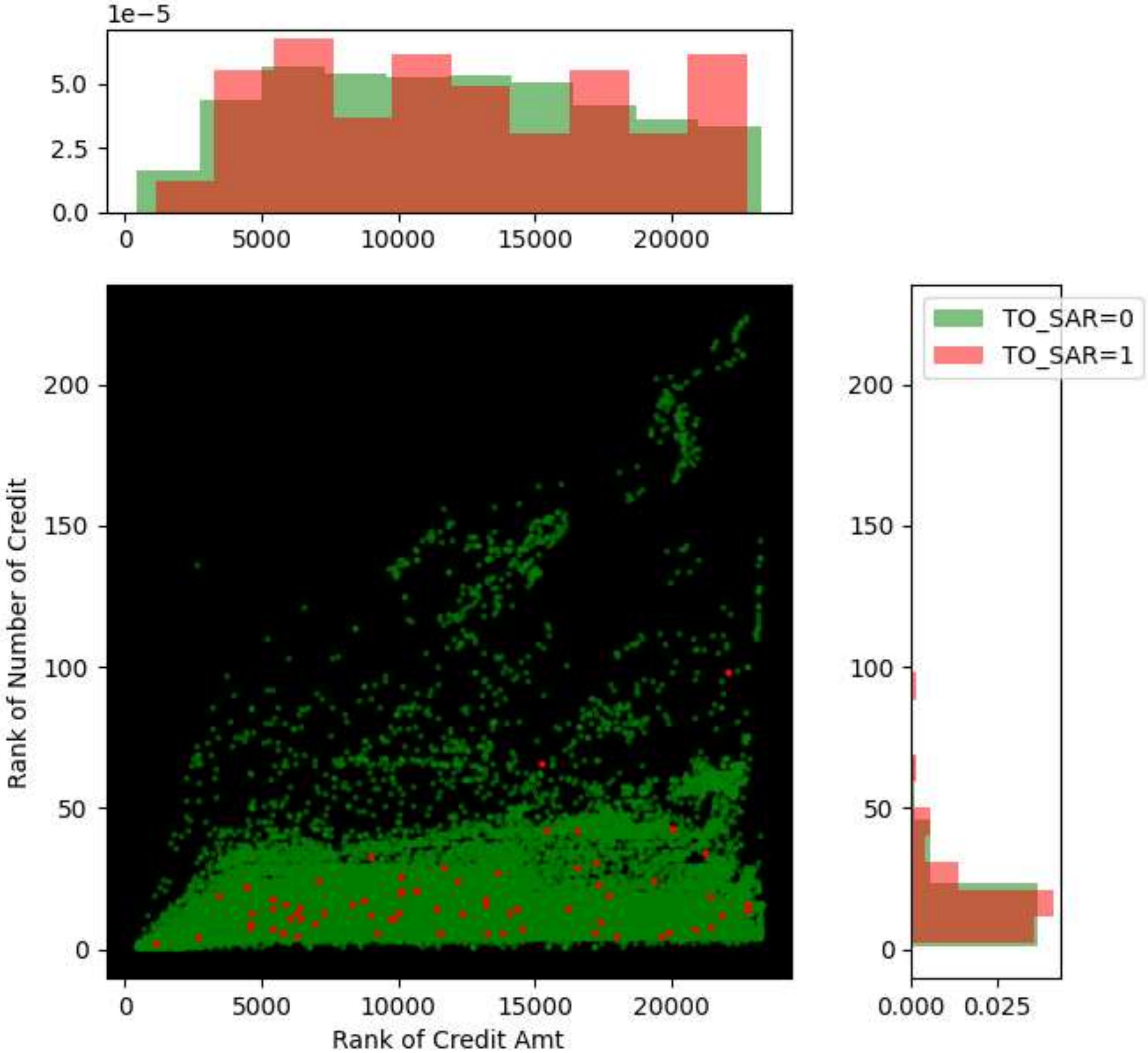
The distribution of SAR=1 and SAR=0 in Credit

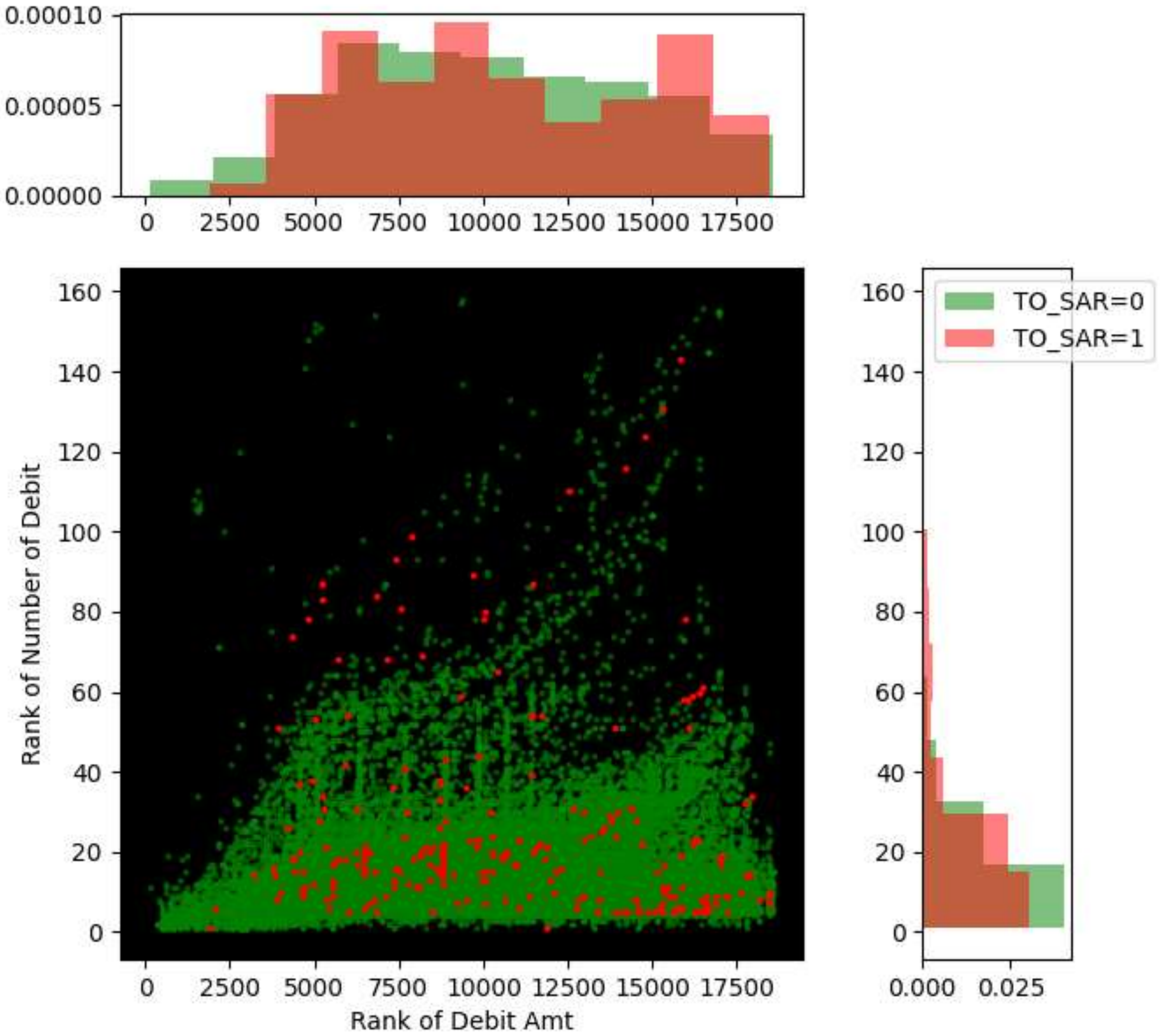


The distribution of SAR=1 and SAR=0 in Debit

Data Preprocessing

- Sorting the data for each features will ensure that SAR=1 is no longer squeezed together
- The histograms represent the distributions of X and Y axis
- In further experiment, we will use the sorted data with Min Max normalization to train the model





Model

- Metrics
- Loss Function
- Discrete Gradient Descent
- Continuous Sigmoid Transform

Metrics

- Our goal is to minimize labor costs while still maintaining compliance with regulatory requirements
- Since the FSC requires capturing adequate number of SAR=1 cases, we use Recall to evaluate this capability
- Human Resource Saving (HRS) evaluates the number of tellers required for manual review after the rule-based system
- Here, we define SAR=1 as the positive samples and SAR=0 as the negative samples
- We want to find the threshold that maximizes while ensuring the recall is greater than 0.8 simultaneously

$$Recall = \frac{TP}{TP + FN}$$

$$HRS = \frac{TN + FN}{TP + TN + FP + FN}$$

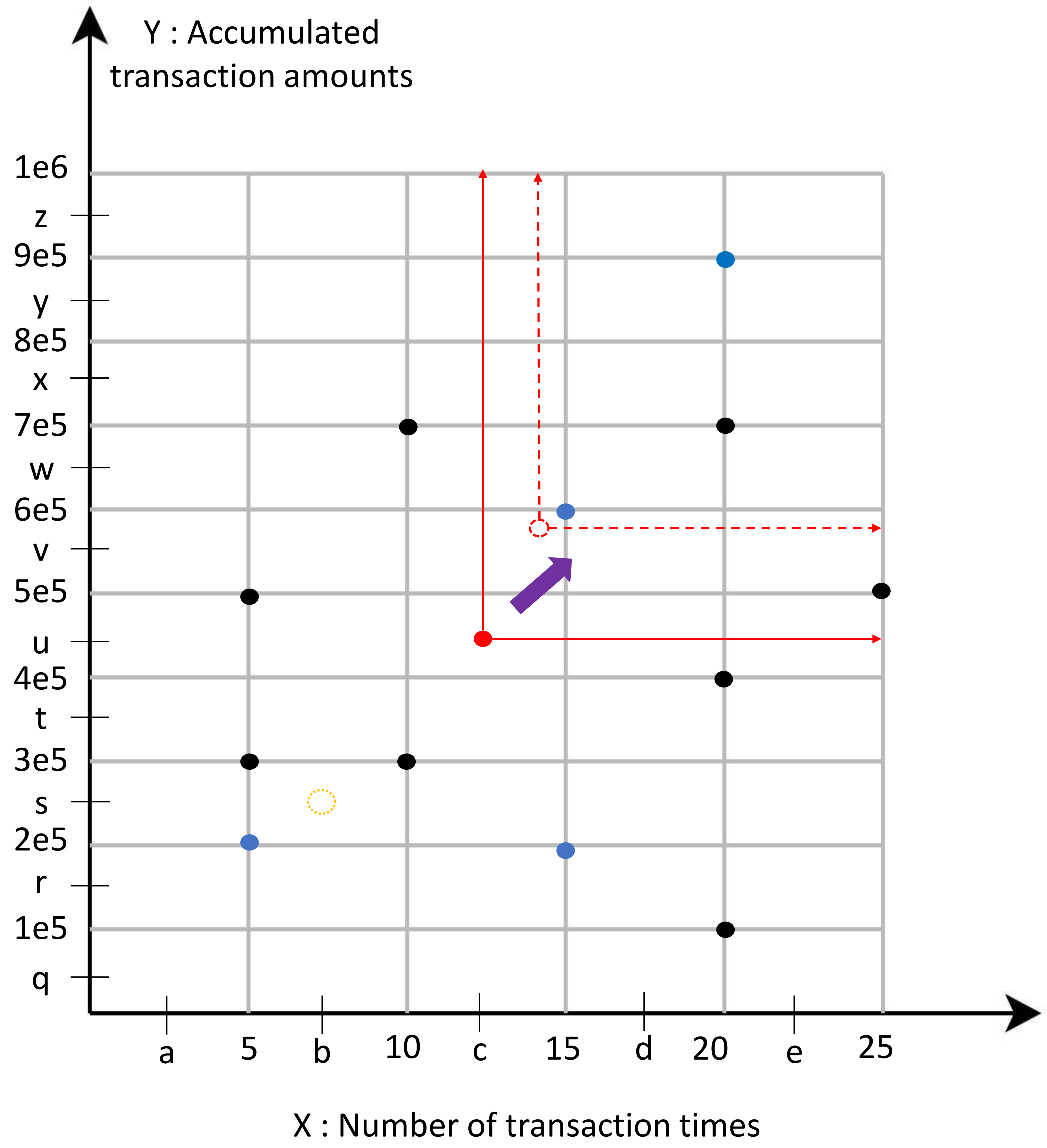
Metrics

- If the solid red line is a set of threshold

$$Recall = \frac{2}{4} = \frac{1}{2}$$

$$HRS = \frac{8}{12} = \frac{2}{3}$$

- Obviously, this set of threshold fails to meet the FSC requirement

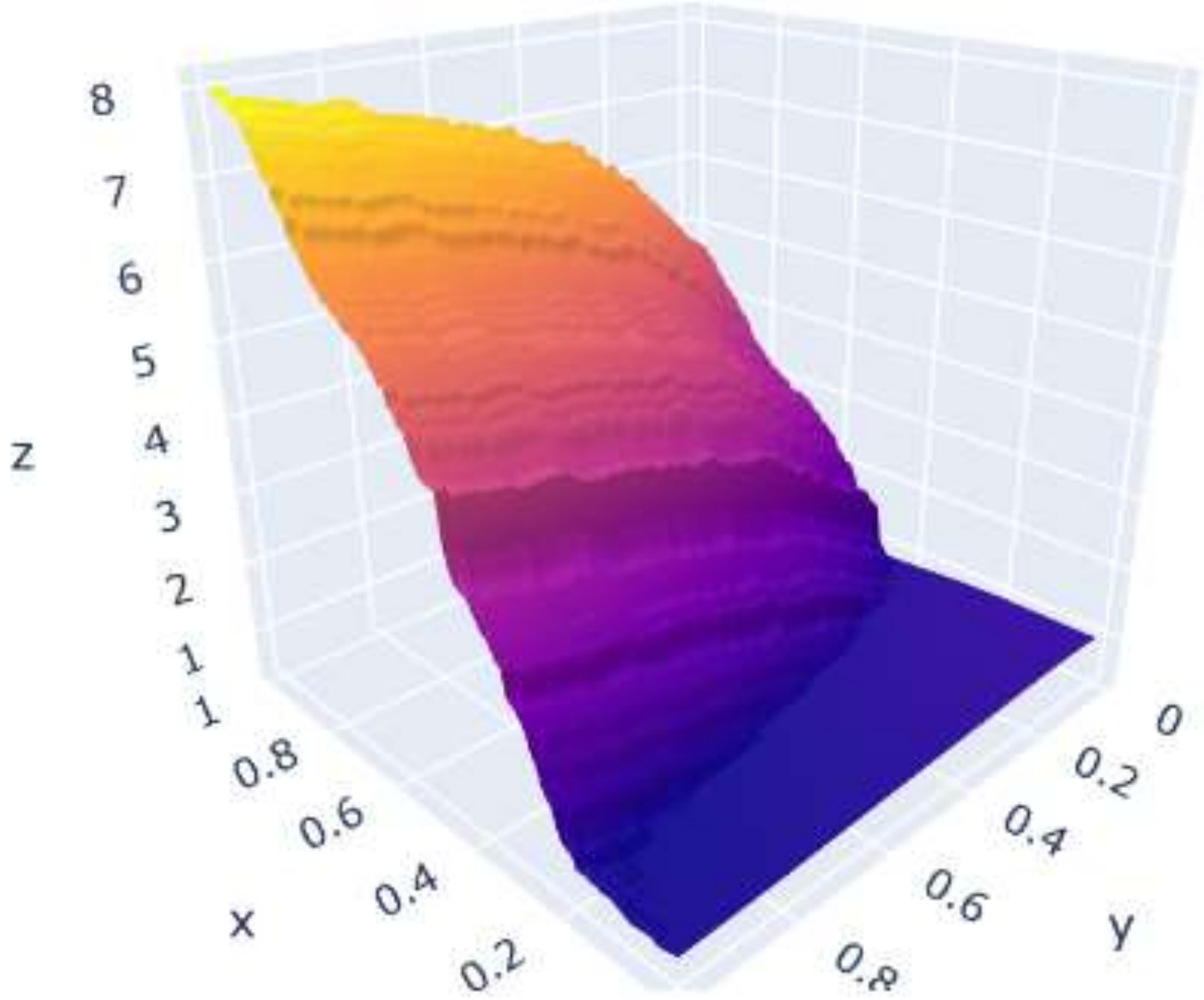


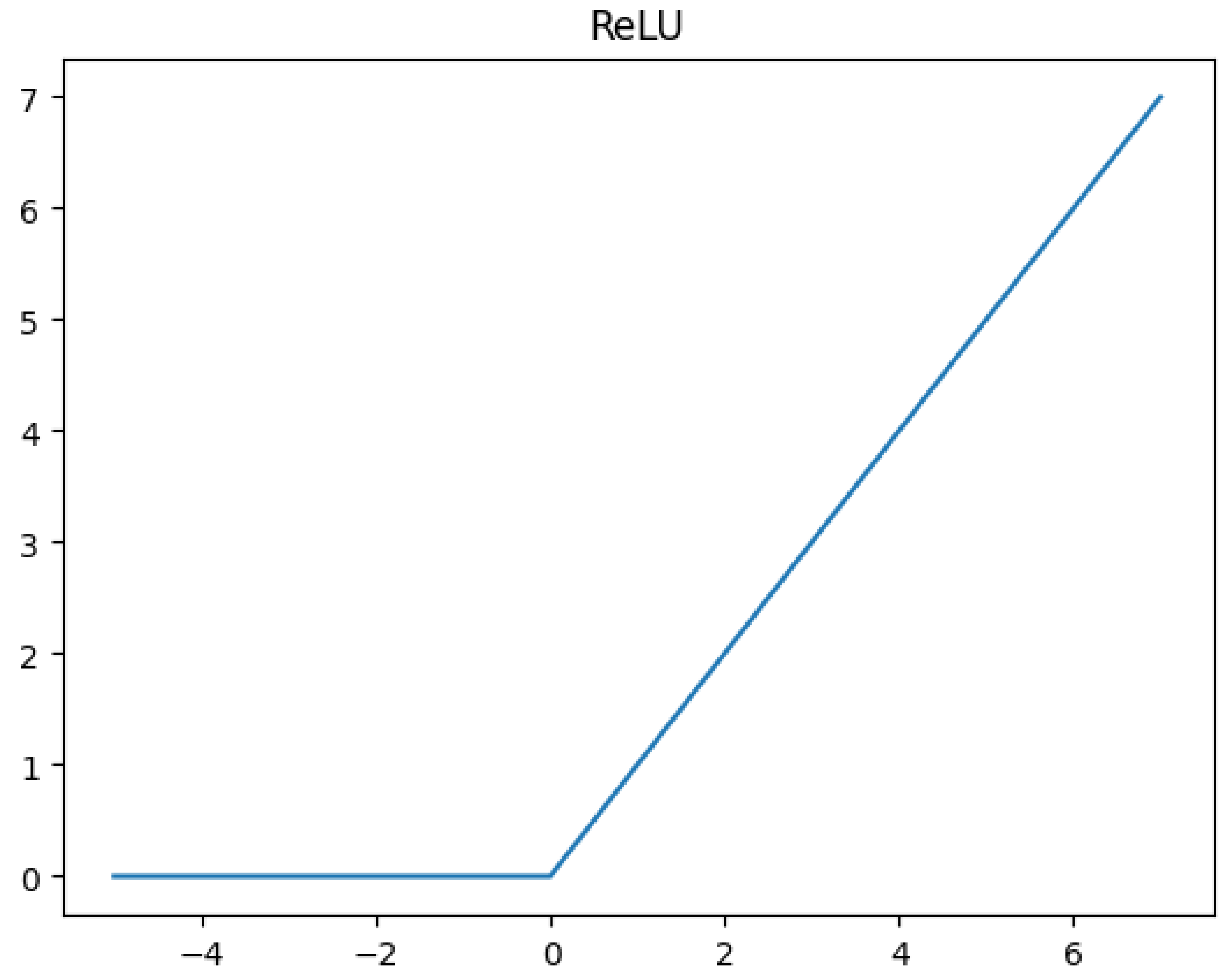
Loss Function

- The traditional loss function does not take the FSC requirement into account. To address this, we have designed a new loss function to tackle this task

$$L(HRS, recall) = (1 - HRS) + W \times ReLU(minrecall\ - recall)$$

- The parameter W represents a weight that determines the importance of recall, while $minrecall$ is an exogenous variable denoted by the FSC requirement, which is set to 0.8 in this article
- The ReLU function is a non-linear function that outputs the input directly if it is positive, otherwise, it will output zero. We use it to assess whether the recall exceeds $minrecall$. If not, an additional penalty is applied





Discrete Gradient Descent

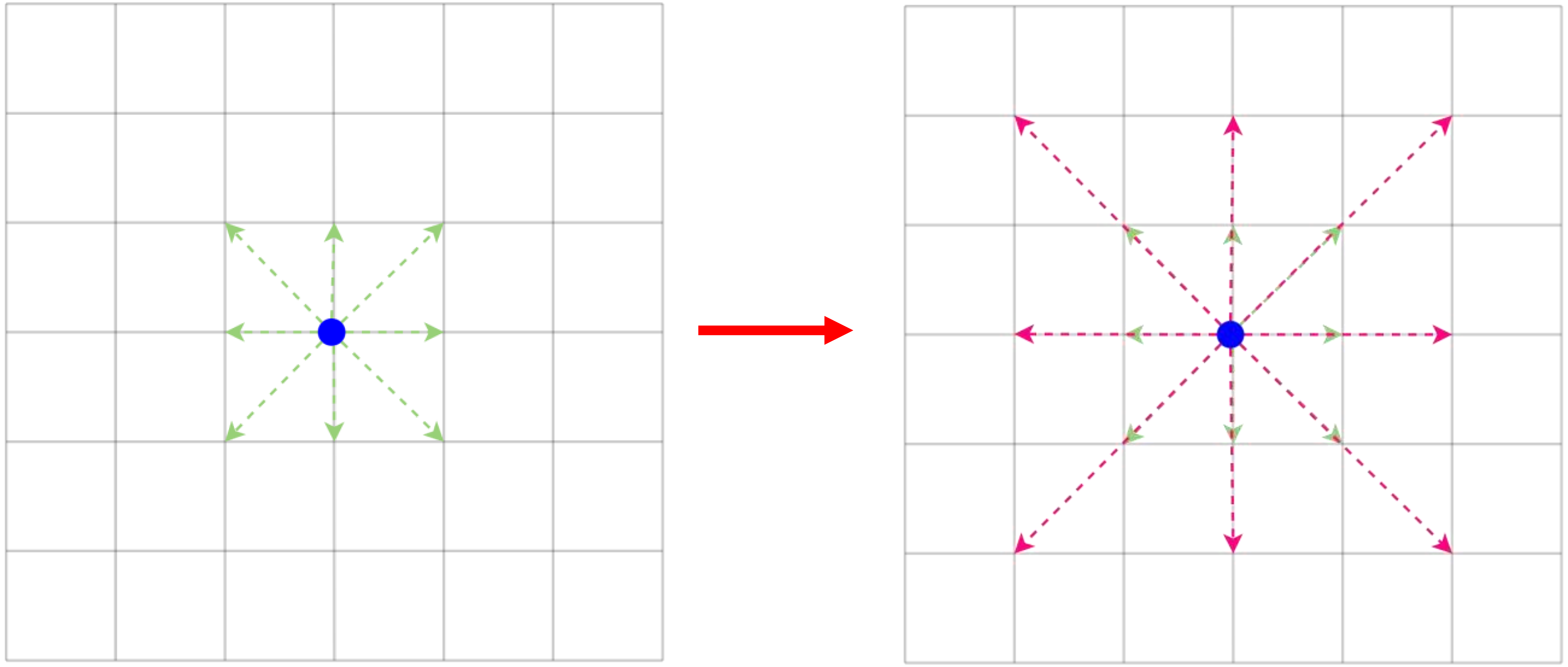
- Gradient descent is a popular strategy to find the optimal minimum in recent.
- Creating a multidimensional grid, where each dimension represents a feature, we can identify the current set of thresholds
- Both HRS and recall are not continuous functions, so we can not obtain the gradient by differentiation.
- We use the finite difference method to replace it

Discrete Gradient Descent

- The main concept of gradient descent is to find the largest gradient for updating. The finite difference method requires an auxiliary point to approximate derivatives. Considering our data distribution is uneven, determining the size of the area to search becomes a significant issue
- Our solution is to dynamically expand the search area by selecting three possible threshold values for each feature during each update, so that there are a total $3^K - 1$ possible threshold values in each update
- We expand the search area when all gradient are zero

Blue point is a current threshold

[-1, 0, +1]



[-2, 0, +2]

Continuous Sigmoid Transform

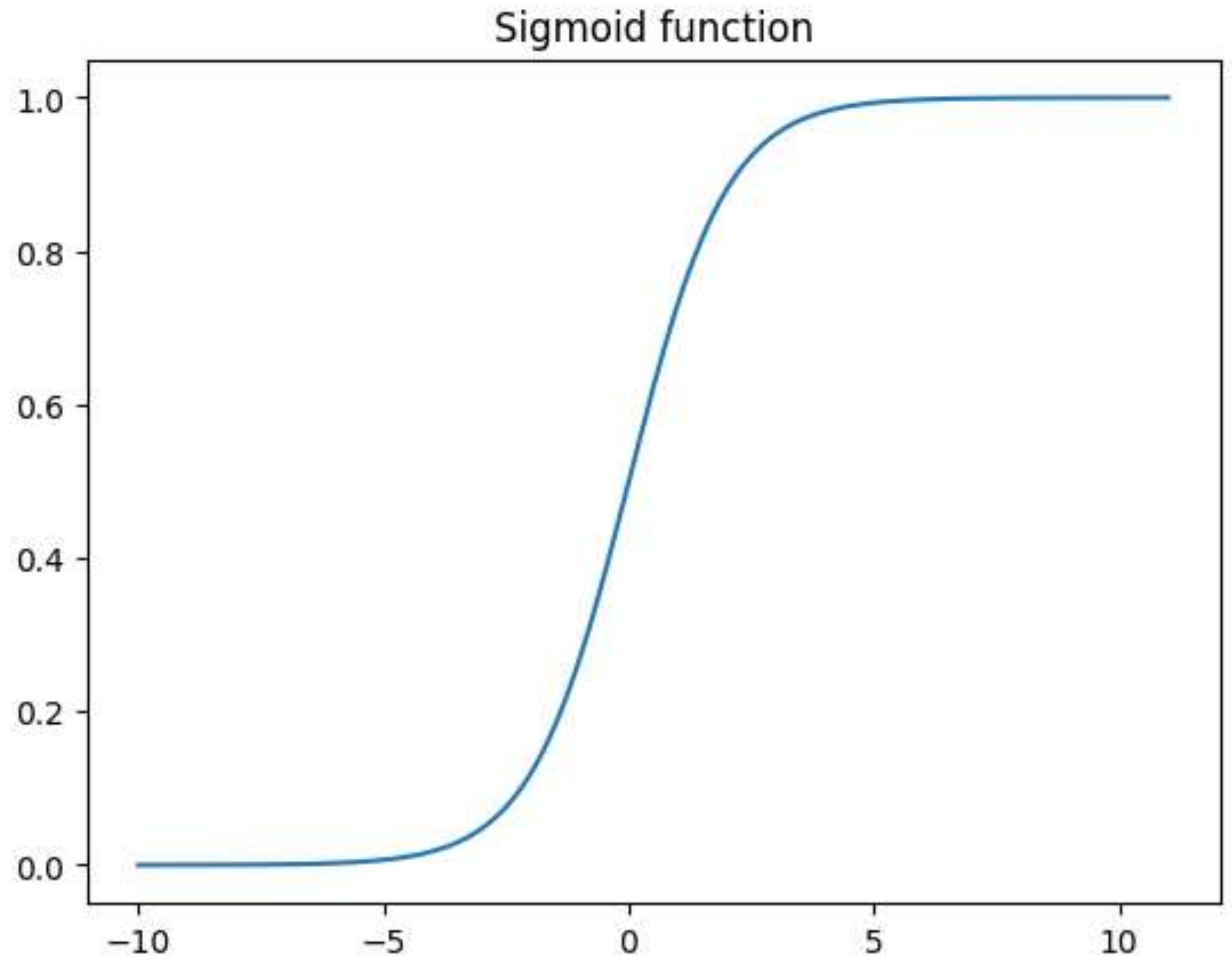
- To deal with the discrete problem, we provide a novel method that we call the Continuous Sigmoid Transform (CST)
- The sigmoid function, which is widely used for an binary classification problems, has a characteristic “S-shaped”. It maps any input value to an output value between 0 and 1
- For each feature, we denote the input value as the difference between the feature value and the corresponding threshold, so that when the feature value exceeds the threshold, the output of the sigmoid function will approximate 1

$$f(x) = \frac{1}{1 + e^{-(x-t)}}$$

現金存款累計≥ A11_Credit_amount 元 **且** 次數≥ A11_Credit_count 次

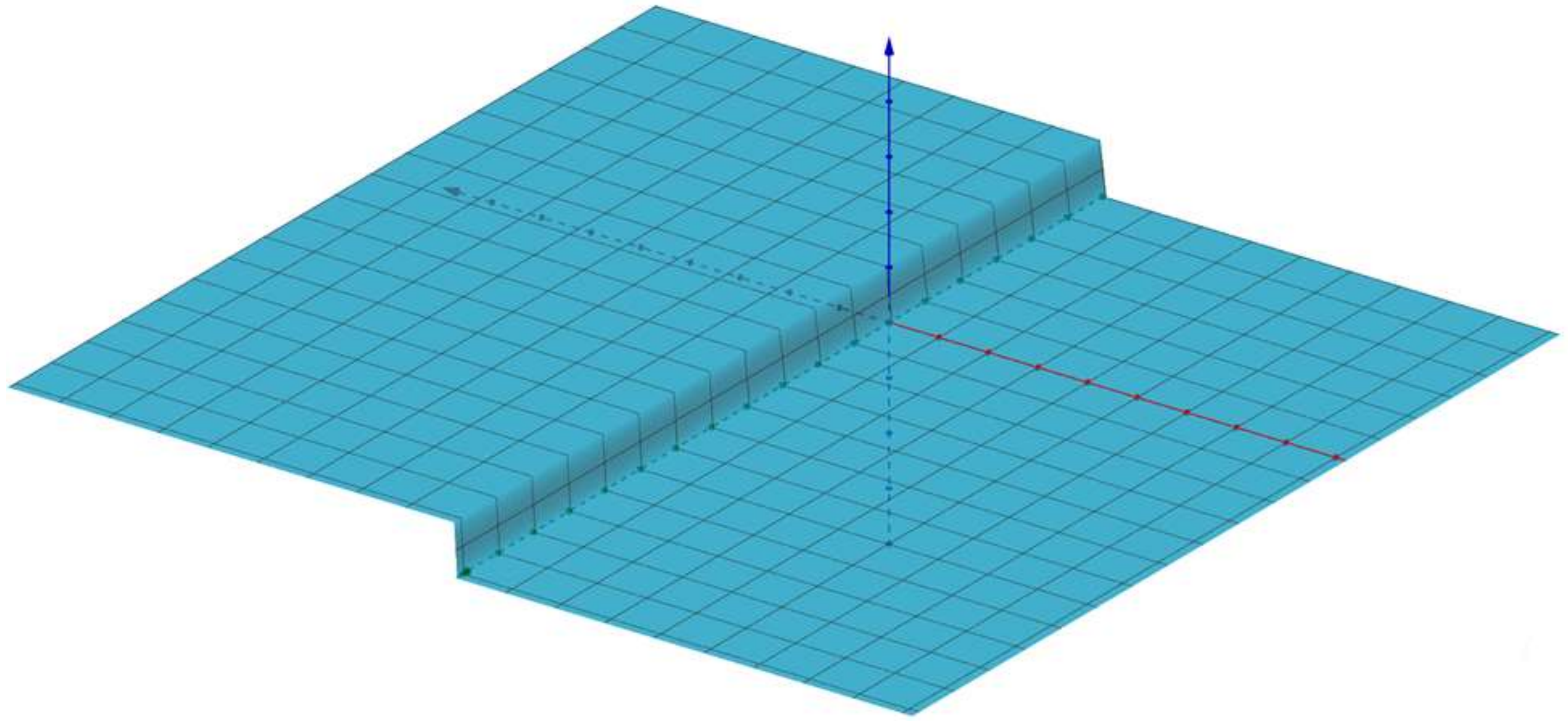
或

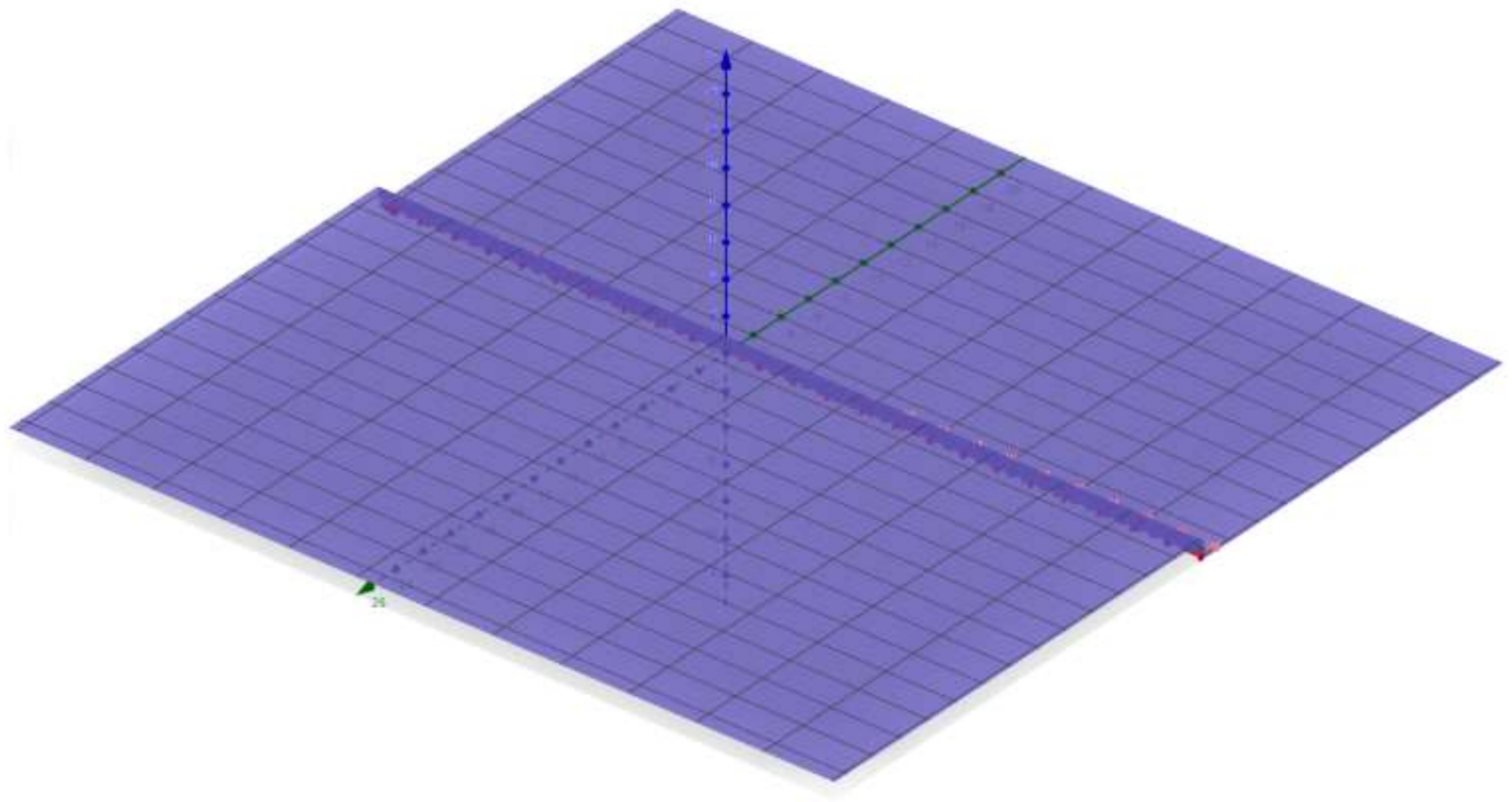
現金提款累計≥ A11_Debit_amount 元 **且** 次數≥ A11_Debit_count 次



Continuous Sigmoid Transform

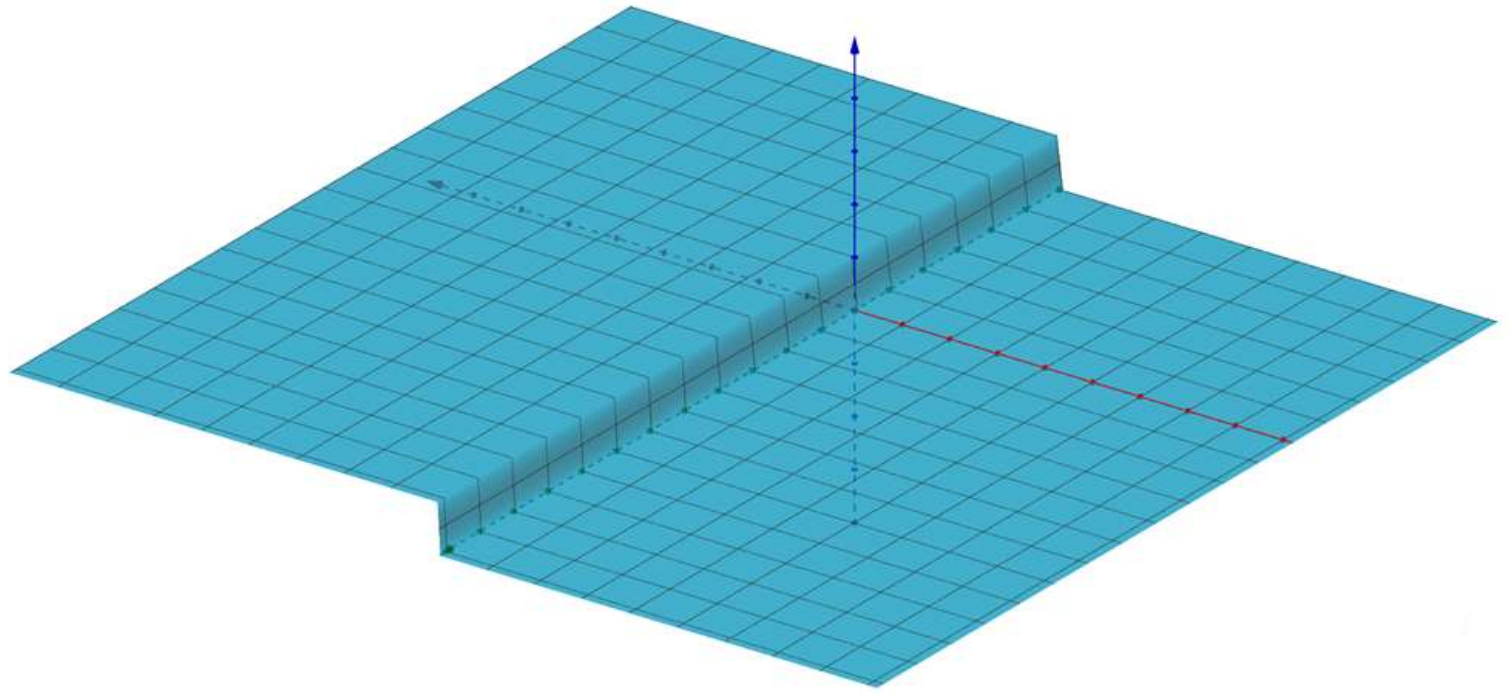
- To deal with the discrete problem, we provide a novel method that we call the Continuous Sigmoid Transform (CST)
- The sigmoid function, which is widely used for an binary classification problems, has a characteristic “S-shaped”. It maps any input value to an output value between 0 and 1
- For each feature, we denote the input value as the difference between the feature value and the corresponding threshold, so that when the feature value exceeds the threshold, the output of the sigmoid function will approximate 1
- Therefore, we use a continuous function to determine whether the feature value exceeds the threshold or not



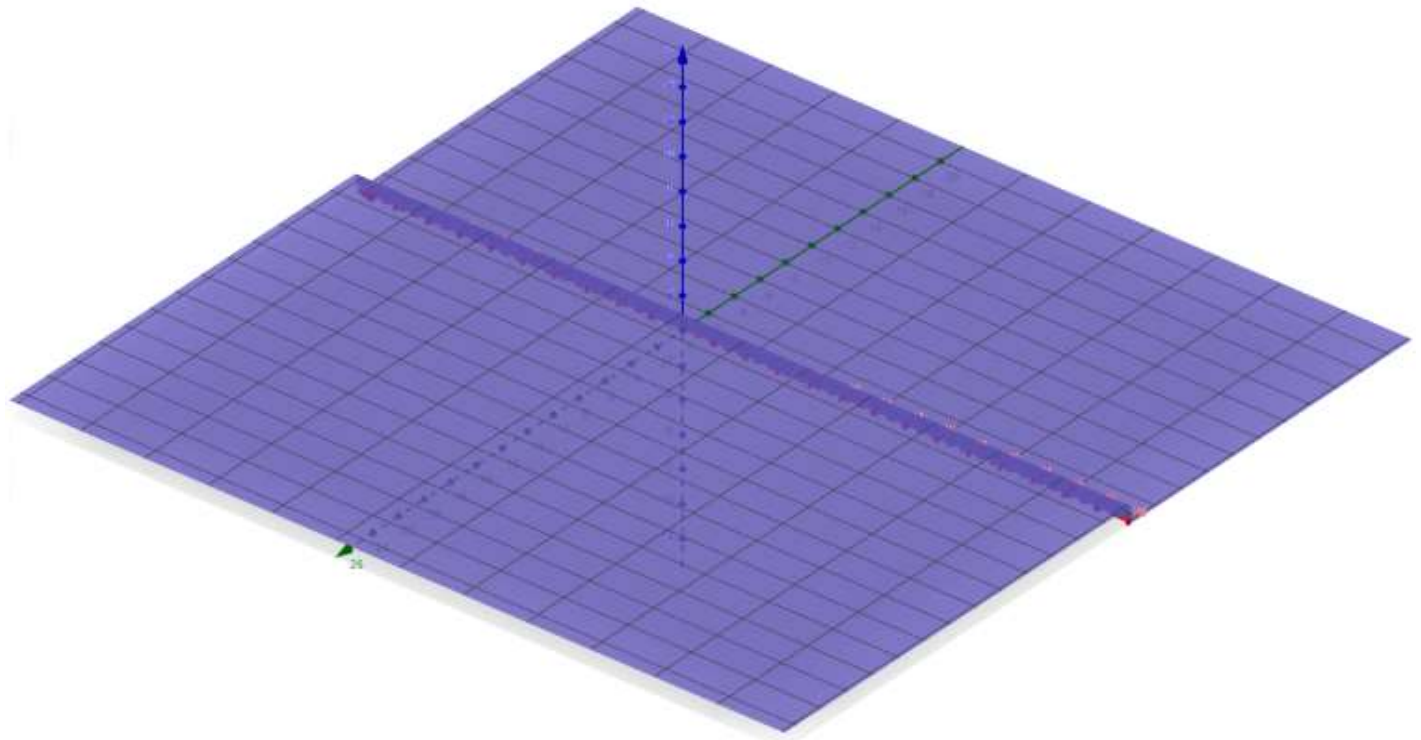


Continuous Sigmoid Transform

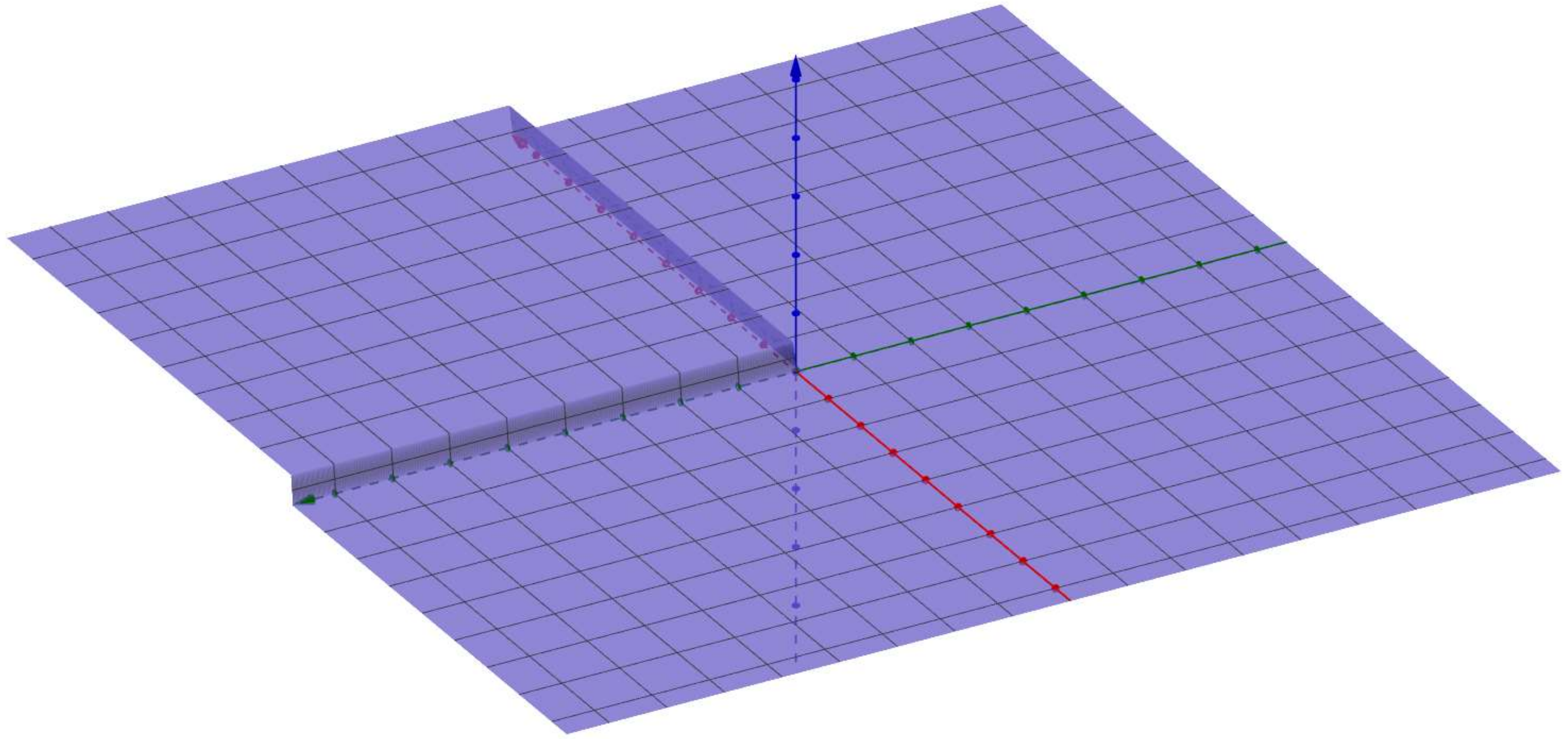
- The logical relationship between each feature can be described by the mathematical operation
- The logical operator “AND” can be denoted as multiplication



×



=

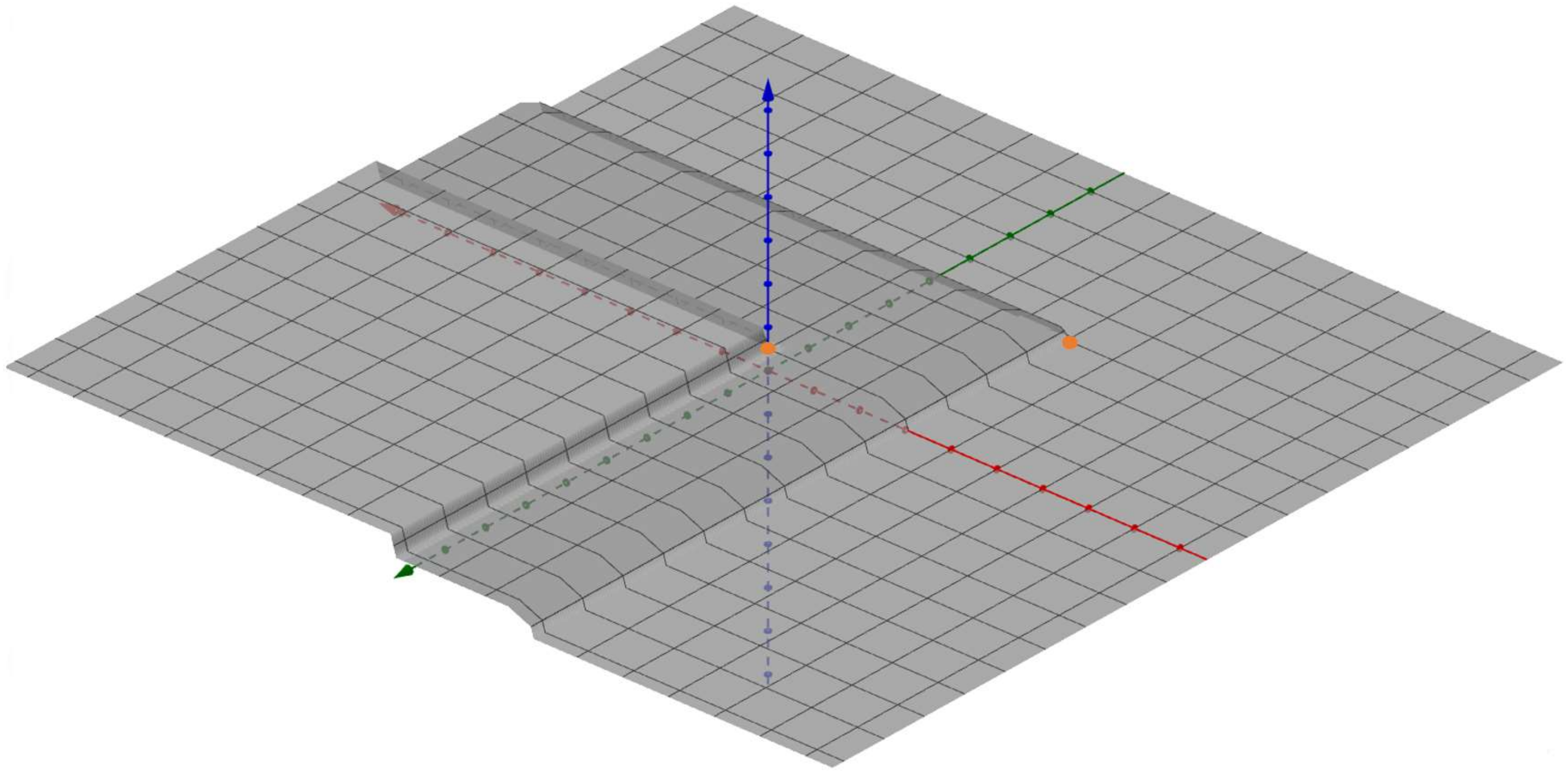


Continuous Sigmoid Transform

- The logical operator “OR” can be denoted by multiplication and addition

Continuous Sigmoid Transform

- Finally, we use multiple sigmoid function to represent the rule, and sum all outputs to obtain the number of SAR=1 predictions. Therefore allows us to calculate the HRS and recall
- Here we demonstrate two data points in the same set of thresholds
- Finally, we convert the threshold optimization problem into a usual continuous optimization problem, allowing us to implement the gradient descent method



Result

- Estimated Time for Brute Force Method
- Model Result
- Compare Different Preprocessing Method
- Implement in Kaggle

Estimated Time for Brute Force Method

- Selecting q-quantile values to calculate the required time
- For given K feature, the time complexity is $O(n^K)$
- A Complete combination requires almost 2 million years to finish

q-quantile		10	20	30	40	10^12(est.)
Times (sec.)		66.14100	1088.44561	5516.77918	17361.85421	5.63*10^14
Recall		0.8000195	0.8008570	0.8002568	0.8001591	
HRS		0.1999806	0.1991430	0.1997433	0.199841	
Threshold	Credit amount	1320920	881735.4509	1084438	1124048	
	Credit count	0.5	0.5	0.5	11.43609	
	Debit amount	290191.3	467395.3536	860889.1	50	
	Debit count	0.5	8.264854	5.426185	0.5	

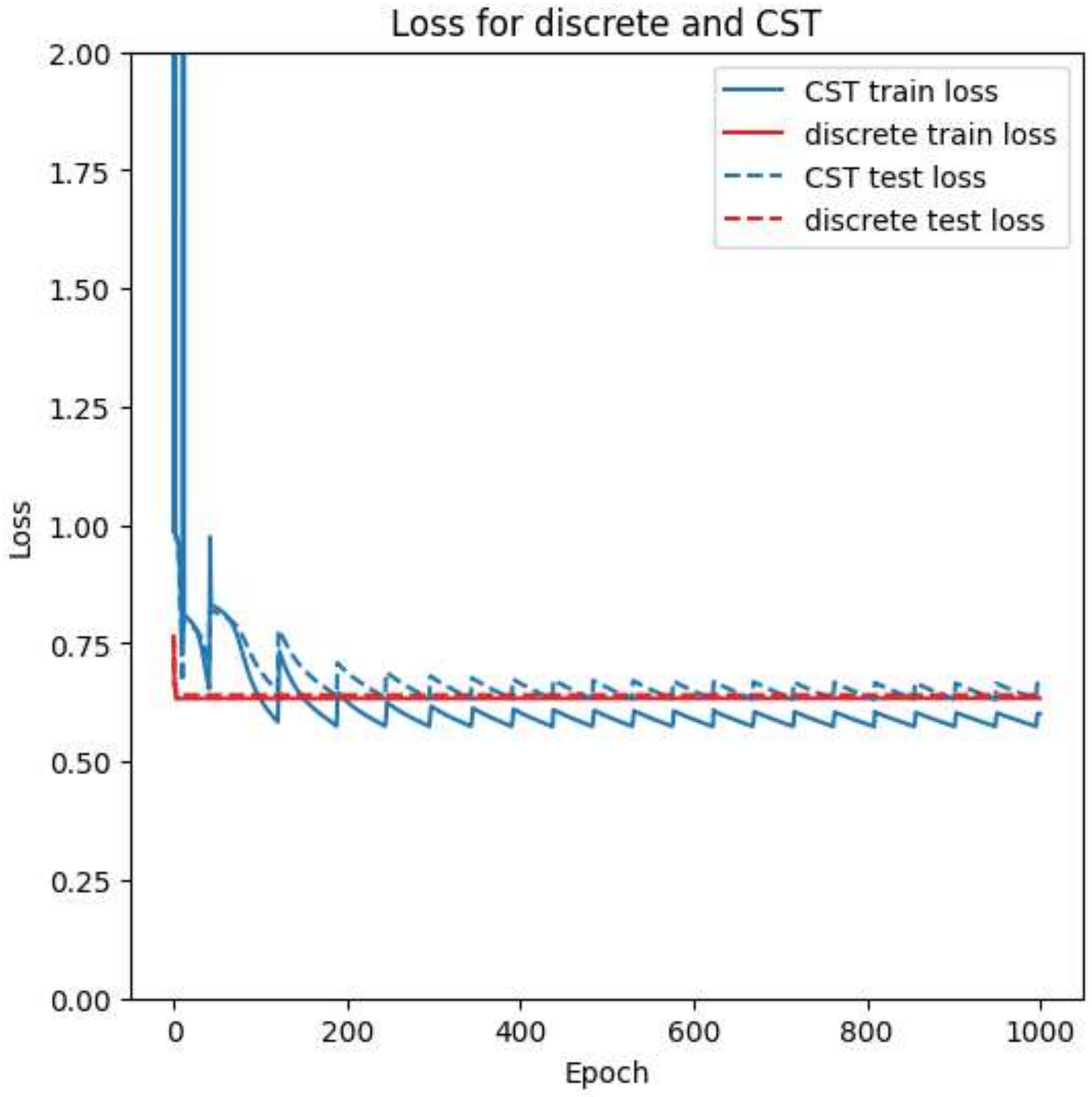
Model Result

- We compare four different method to find the optimal threshold
- Four methods meet the FSC requirements in both the train set and test set
- CST has highest HRS in both train set and test set

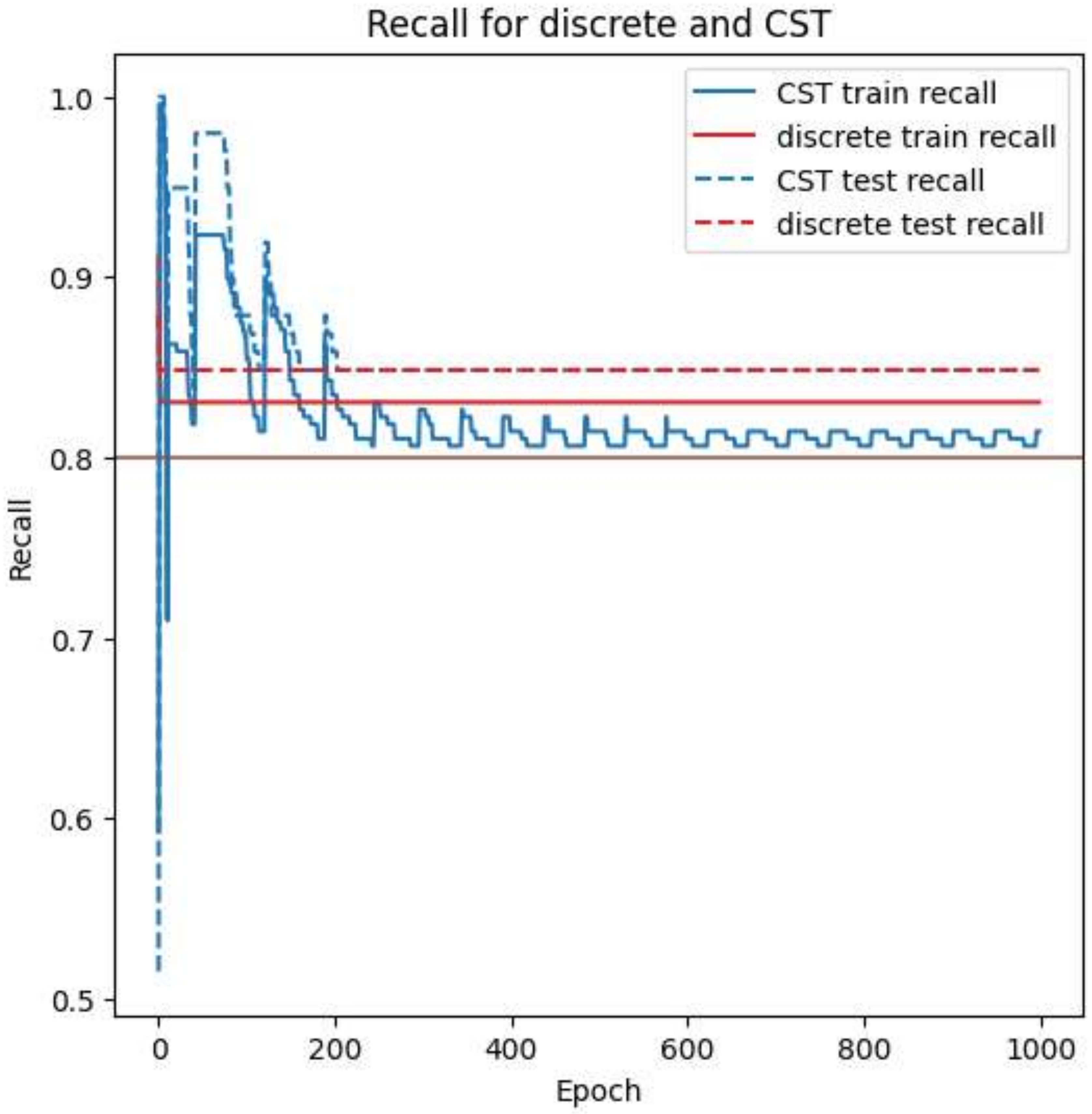
	Train Loss	Train Recall	Train HRS	Test Loss	Test Recall	Test HRS	Training time (sec.)
Brute force	0.800058	0.842697	0.199942	0.813314	0.919192	0.186686	17361.854
Discrete gradient	0.633945	0.830645	0.366055	0.640775	0.848485	0.359225	269.123165
CST	0.609807	0.806452	0.390193	0.607284	0.808081	0.392716	201.572707
Simulated Annealing (100 averages)	0.765700	0.898674	0.234300	1.184471	0.221589	0.920101	2.692259

Model Result

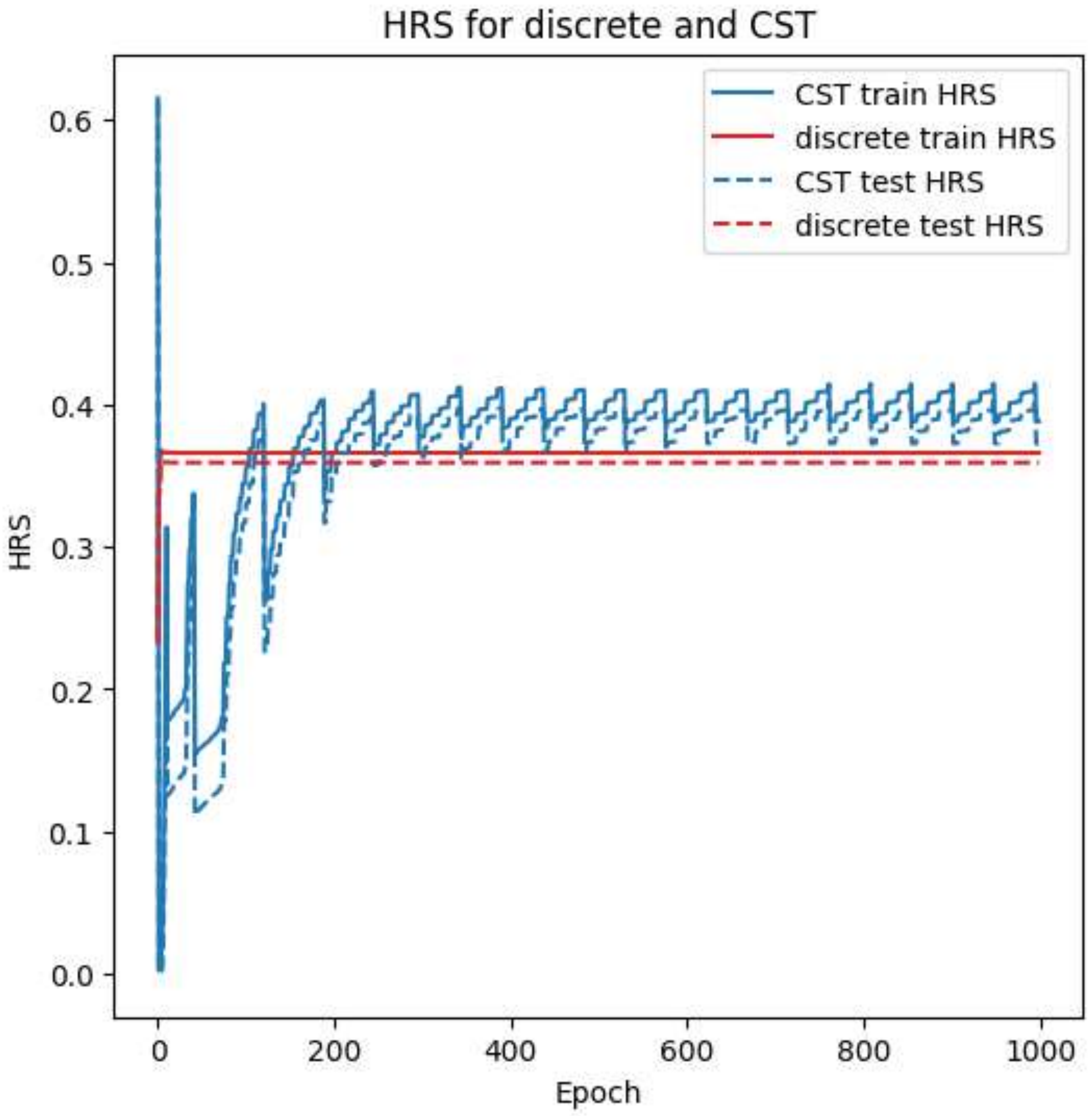
Loss



Recall



HRS



Compare Different Preprocessing Method

- Compare to the four normalize method in CST model
- CST has highest HRS in both train set and test set

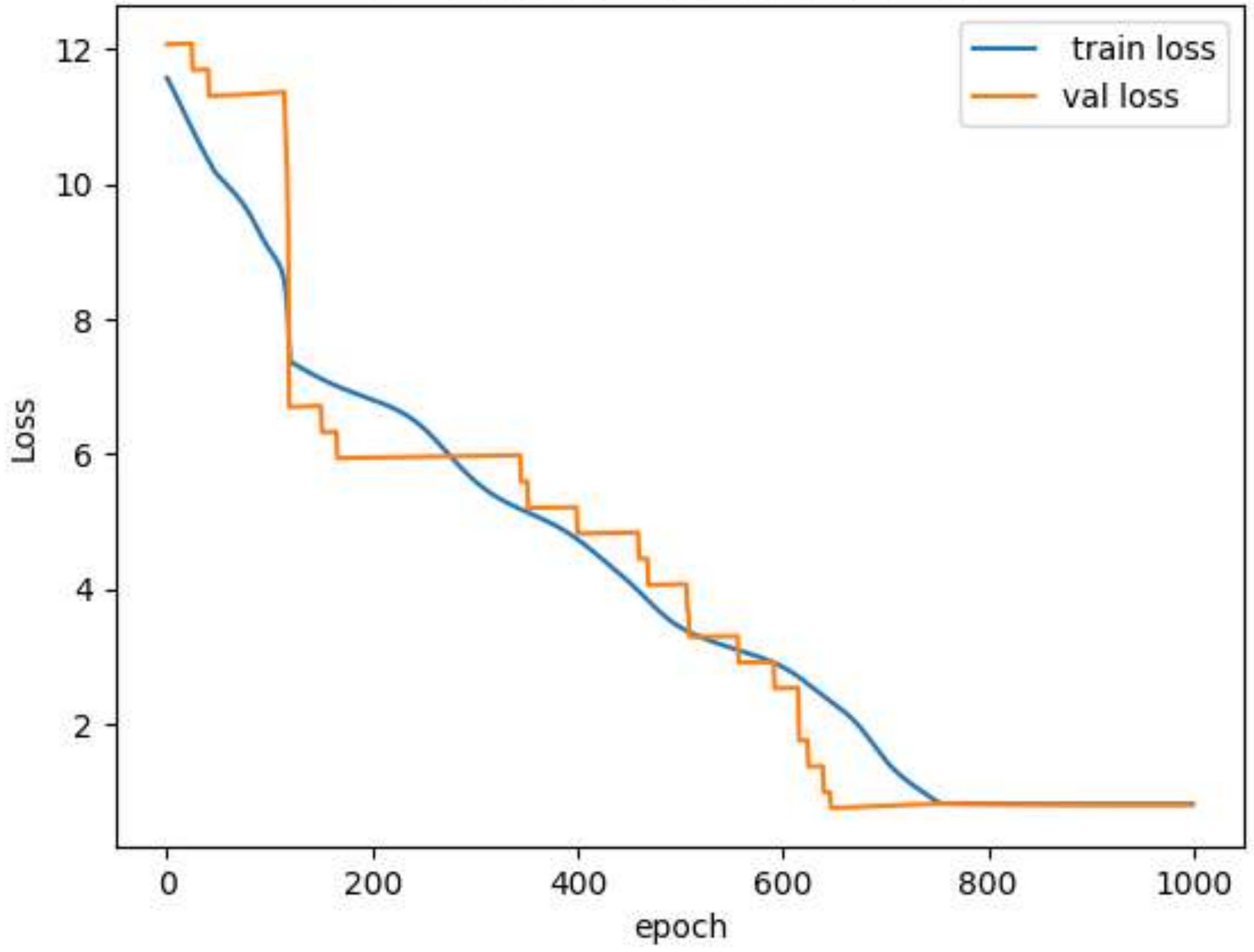
	Train Loss	Train Recall	Train HRS	Test Loss	Test Recall	Test HRS
Rank Min Max	0.609807	0.806452	0.390193	0.607284	0.808081	0.392716
Real	0.977162	0.920968	0.056709	0.762405	0.858586	0.237595
Min Max	0.765784	0.933871	0.234216	0.759541	0.858586	0.240459
Rank	0.733852	0.853226	0.266148	0.735013	0.858586	0.264987

Implement in Kaggle

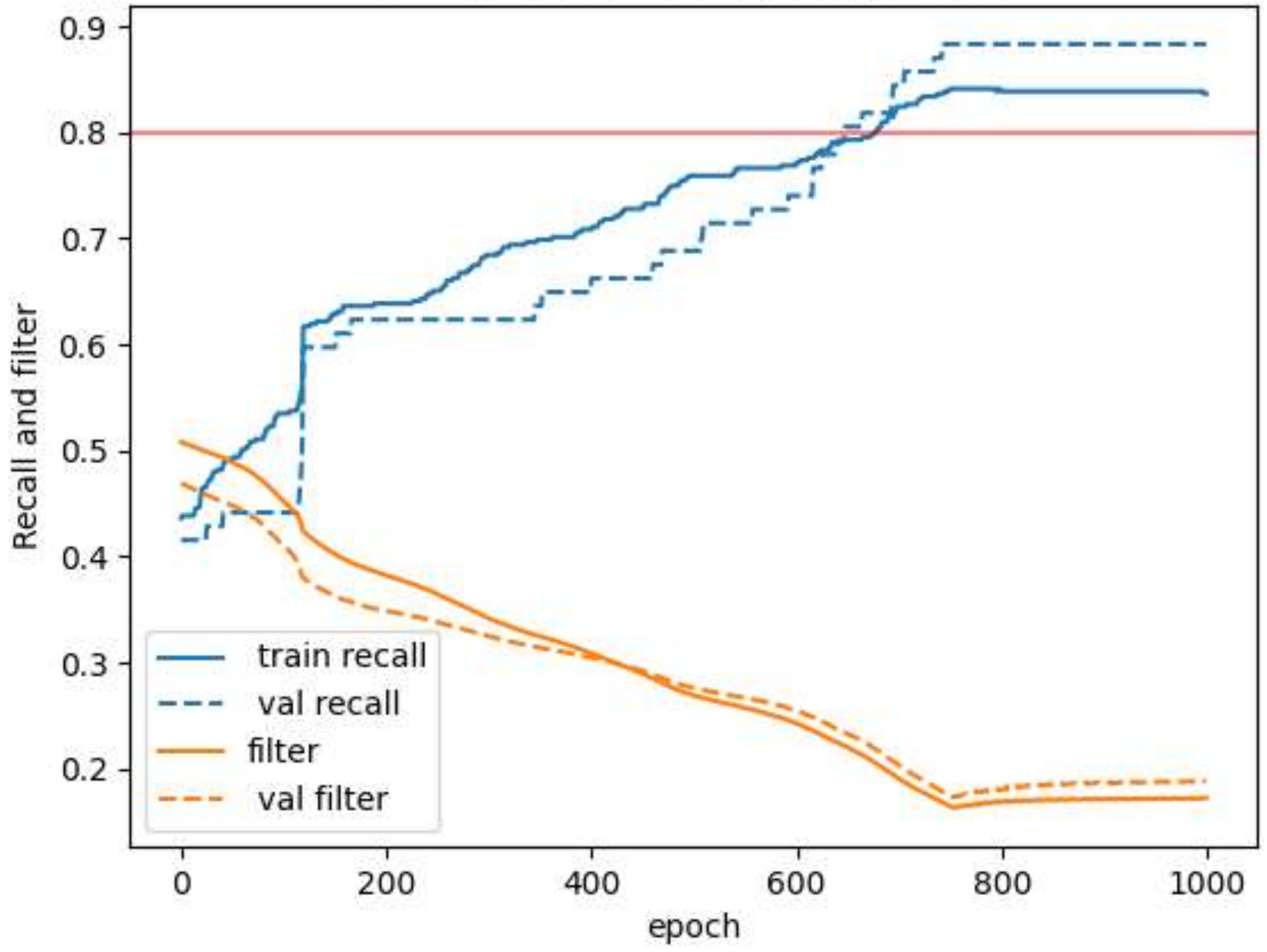
- A Kaggle dataset for credit card fraud detection comprises 29 features and 284,807 data
- Select four features and use the same logic as TWN-A11-01

Loss: 0.8273674 Recall: 0.836144 HRS: 0.1726325 Time: 221.50234

Loss



Recall and HRS



Conclusion

- Conclusion

Conclusion

- Rank Min Max with CST is a great method to optimize the thresholds in rule based system
- Test on different datasets to demonstrate that our model is robust
- This model has been adopted by a bank in Taiwan.