Why Do Different Problems Need Different Classifiers?

03/15/2020

STAT 615-Group 8

Yinqiu Xu, Haoyue Shi,

Peibin Rui, Hao Jiang, Zihang Wang

Background

When facing a classification problem in real life, there are so many statistical algorithms for us to utilize. You may think about whether there is a *best* classifier that can work well for all kinds of classification problems. Unfortunately, such a classifier does not exist. Let's think about a situation that you are going to clean a room. You may use a broom, a mop and a vacuum cleaner, but you definitely won't take a shovel and begin to dig the floor. Different tools are used in different situations. It is also true in the field of classification. Just as a saying goes - "there is no free lunch". This is to say that there is no classifier that works best for every problem. We will explain this in the following paragraphs.

Assume that we are working for a national bank. It is going to develop a credit card system to promote its recently issued credit cards. The customer data of different branch banks in different cities may vary greatly. And different banks may also have different specific goals to achieve. How to classify people to different classes? How to decide what kinds of people are qualified to issue a credit card? We would like to apply classification algorithms.

The regression-based classifiers, similarity classifiers, and Empirical Risk Minimization-based classifiers are three common kinds of classification algorithms. We choose one specific classifier from each of the three general kinds of classifiers to illustrate the idea that using different classifiers for different problems is necessary.

Logistic regression is a typical regression-based classifier. Its output is a number between 0 and 1, which stands for the probability to issue a credit card to the applicant. Obviously, the larger the output is, the more likely to issue a credit card to the applicant. And we need a threshold to decide whether to issue a credit card or not.

The k-nearest neighbors (KNN) classifier is based on similarity. Given a new input object, KNN will find its k nearest neighbors in the data set and assign the class most common among the k nearest neighbors as the class of the new data. For example, 7 out of 10 David's nearest neighbors (more than half members) were tested positive to COVID-19.

Then without medical testing, he will be classified as a virus carrier based on KNN classifier. We should warn him to take this virus seriously and take care of himself. When applying KNN classifier on computers, it is easy to perform but has a little bit longer runtime because unlike David, the computer does not know who his neighbors are directly. Therefore, it has to compute the distance between David's house and everyone else's house to find out the k nearest neighbors.

Support Vector Machine (SVM) is an ERM-based classifier. Despite the obstacle understanding name, SVM is actually based on a very simple idea. Just consider making classification on a plane. There may be many straight lines that can divide two classes. SVM wants to find the "best" line, which means this line can maximum the distance between the nearest elements of each class. Think about that there are two kinds of balls that differ in weights scattered on the table. Obviously, we can't find a straight line to divide them. We can use a trick by slapping the table heavily so that all the balls would bounce into the air. Heavy balls would bounce lower than light balls. Then we can separate two kinds of balls easily. This is exactly how the SVM works when we can't find a linear boundary to separate.

Now we can delve into several scenarios and discuss why we must consider different classifiers for different problems.

Scenario 1: Logistic Regression

Suppose the bank now has a huge amount of data and they try to build a flexible model that can be easily adjusted according to the financial situation and development strategy of our bank. For simplicity, we only consider data with two independent variables which are yearly income and yearly expenditure and one dependent variable which is whether This person has been issued a credit card or not.

The classifier we would consider here is logistic regression. Since it has two advantages which are flexibility and interpretability.

For logistic regression, if the bank wants to conduct an aggressive strategy, they will lower the threshold and issue credit cards to people with higher risk of not paying back. If the bank wants to conduct a conservative strategy, it will increase the threshold and issue credit cards to people with lower risk of not paying back. While the other two classifiers are not quite flexible. To show the comparison between these three classifiers, we show the partial outputs and corresponding data in the following **Table 1**.

Table 1 Partial outputs of 3 classifiers

Classifier	*1*	*2*	*3*	*4*	*5*	*6*	*7*
Income(10,000yuan)	0.364	1.843	3.407	3.392	1.275	0.566	0.732
Expenditure(10,000yuan)	1.507	1.684	3.646	0.025	0.014	0.380	0.471
Logistic regression	0.2144	0.6834	0.346	0.916	0.7165	0.709	0.563
KNN	0	1	0	1	1	0	1
SVM	0	0	0	1	1	1	1
True label	0	1	0	1	1	1	1

We can see that for different kinds of people by adjusting the threshold the logistic regression can give different decisions. Because the output of logistic regression is in a probability sense which is a continuous variable. For example, the sixth and seventh samples have high incomes and low expenditure. A conservative strategy with a higher threshold will result in the rejection of the 7th sample. An aggressive strategy with a lower threshold will result in the acceptance of the 7th sample. The same analysis can be used on the pairs of 2nd and 3rd samples, 4th and 5th samples. Contrarily, the output of KNN and SVM is a binary variable either 0 or 1. Then we can see that there is no flexibility to change the decisions. Therefore, logistic regression is flexible enough to adapt to different goals.

Moreover, when we implement a method, we always want to explain it succinctly and make it easy to understand. For the other two classifiers, we don't have an explicit form to show the process. However, for logistic regression, we can show that the model can be written as

$$logit(\pi) = 0.326 + 0.179 \times income - 0.082 \times expenditure$$

Here π stands for the probability to issue a credit card to the applicant and the $logit(\pi)$ in the model is a monotonic increasing function of which means a higher output $logit(\pi)$ corresponding to a higher probability. The model can be interpreted as a higher income has a positive effect on the probability of getting a credit card and higher expenditure has a negative effect on the probability of getting it.

There are also some disadvantages to logistic regression. One that needs to be mentioned is the linear form assumption of those parameters. Once the relationship is non-linear or there are too many features we need to consider. Then it's unreasonable and inconvincible to assume the linear relationship.

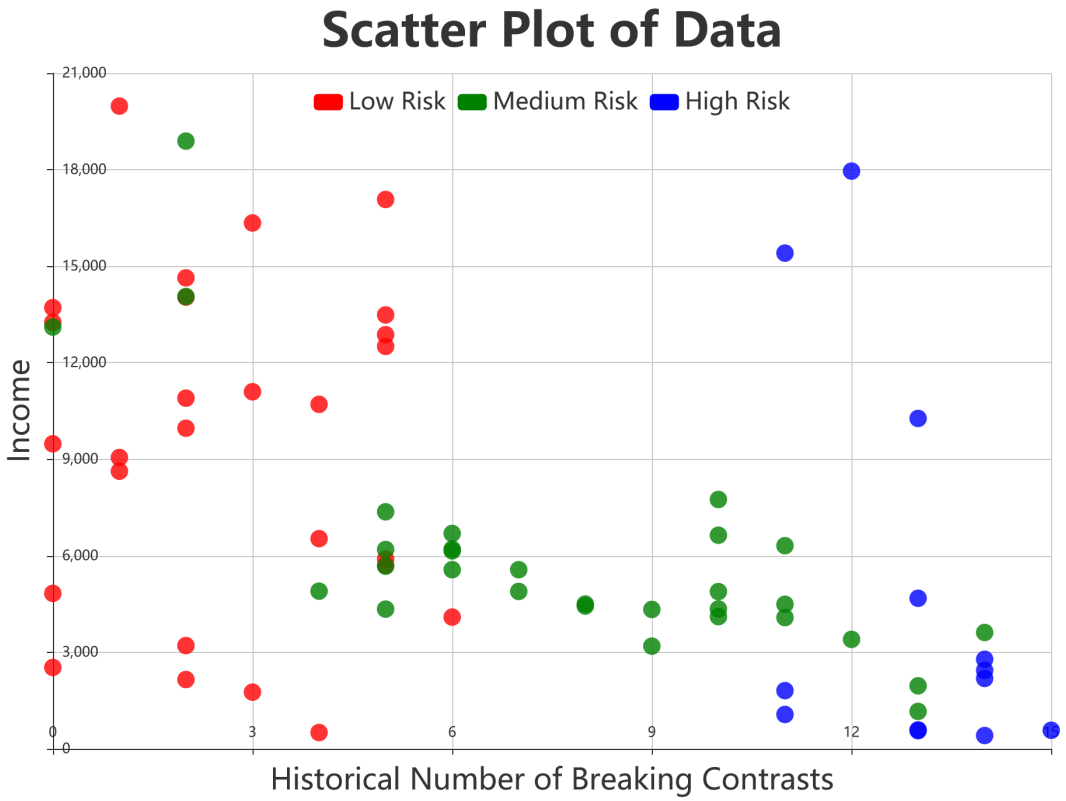
The other shortage is for multi-levels problems, logistic regression will lose its interpretability. For example, we cannot explain 1.45 as one category or any practical meaning. So it's more suitable for binary cases. Outliers which are extreme cases will also affect the estimation of model parameters. We may not be able to find the true influence of dependent variables. Therefore, logistic regression can be influenced by outliers.

Scenario 2: K-Nearest Neighbors (KNN)

Suppose a product manager of the bank wants to know whether a customer has the ability to repay the loan, however, all the information he/she has is a three-class dataset with 'low risk', 'medium risk', 'high risk' as dependent variables and his/her historical numbers of breaking contracts and salary as independent variables. Under this situation, we may find it hard to perform logistic regression. Although we can construct a model to fit the three levels as we treat them as 0, 1 and 2, however, we cannot explain the output clearly. Imagine we get the result 1.45, we can only tell that this is the expectation of the dependent variables but cannot give it a specific meaning like what we do in a binary case, which is the probability of success. When we consider SVM, it is only directly applicable for two-class tasks. Therefore, a method that reduces three classes to two classes is needed and this kind of transformation results in information loss. Actually, any

kind of transformation will cause information loss. An extreme example is that when we use the sample mean to reduce the sample dimensions to one, we lose other information like variance, median and minimum/maximum value, etc. This kind of information loss indicates that SVM may not be a good option in the first place. So, we need another classification method to classify new customers with their characteristics.

Now let us take a quick look at the sample data we get. We can see the trend that people with higher income and fewer breaking contracts are more likely to be treated as low risk and people with lower income and more breaking contracts are more likely to be treated as high risk.



Notice that there are some people with quite high income but still be classified as *high-risk*. One possible reason is his/her expenditure is also very high and does not have enough savings to resist any risk, which is quite common due to the fact people now influenced by many consumption-oriented ideas. Ironically, someone may have a luxury brand handbag but cannot even afford hospital testing fees.

Then we show how to classify a new customer based on this method. There are 9 new customers that already have their labels and we want to test our classifier's accuracy. In this case, the best result is given by choosing k equals 4 and here is the result:

Table 2 Nine Example Results of True and Predicted Labels

True Label	Low	Low	Low	Medium	Medium	Medium	High	High	High
Predicted Label	Low	Low	Low	Medium	Medium	Medium	Low	High	Low

As we can see from the above, we make 7 out of 9. The result is not bad after all considering we cannot get similar results from logistic regression and SVM. However, one thing ignored is that the cost of treating *low-risk* people as *high-risk* is quite different from treating *high-risk* people as *low-risk* people intuitively. If we treat a *low-risk* as *high-risk*, which means we make decisions more cautiously, we may not experience a loss. However, if we treat a *high-risk* as *low-risk*, there is a potential that this person will break the contract again and lead to financial loss. This kind of situation requires us to pay more attention to people with larger numbers of breaking contracts behavior and be more discreet when we make the decision.

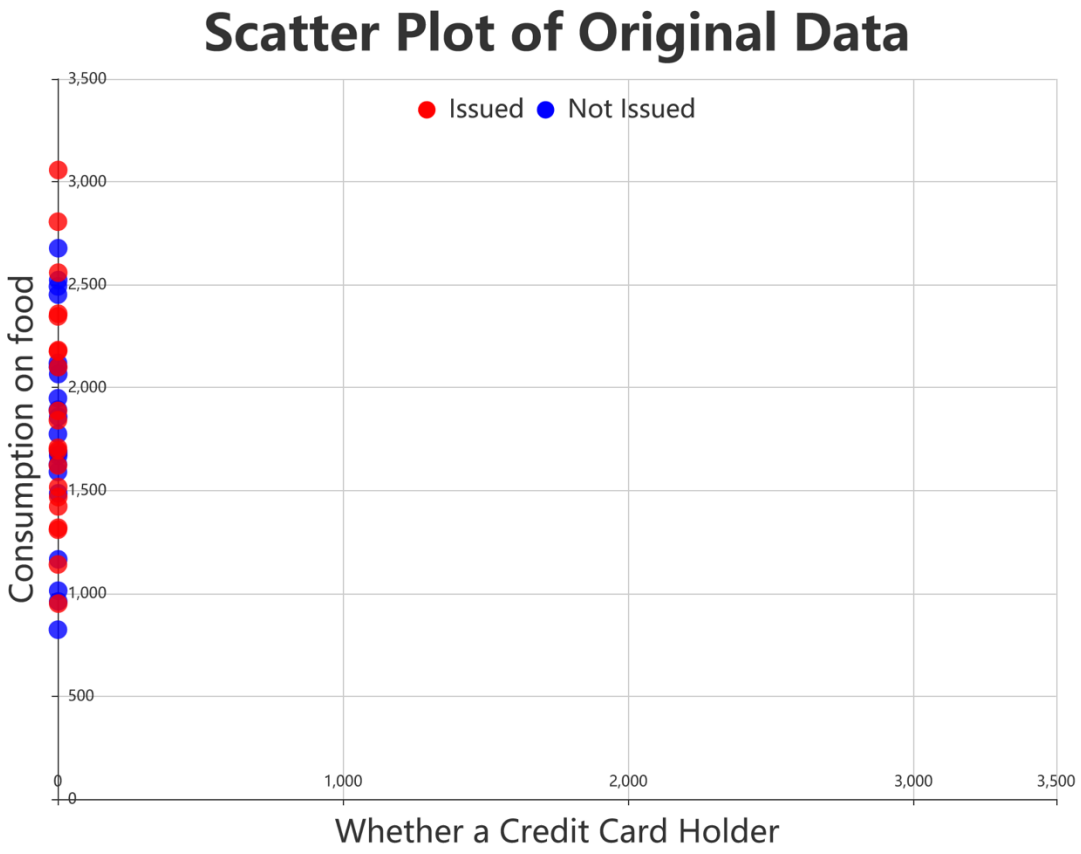
This classification method indeed has its drawback, we must be careful when some classes are extremely unbalanced, for example, if we have 2 *low-risk* people and 1000 *high-risk* people in our dataset, it may not be a good idea to use this method. If lucky David has the antibody to COVID-19 and unluckily his 1000 coworkers get infected and only 2 colleges have the antibody, there will be a really great chance that he is classified as infected.

Another problem of KNN is that it's not suitable for handling categorical data.

Firstly, the core part of KNN is to determine the distance between samples, but for category variables, calculating the distance between them has no practical meaning. For example, 1 represents the occupation of the client as a teacher, 2 represents the occupation of the client as a worker, and 3 represents the occupation of the client as an

accountant. We can't conclude that the difference between accountant and teachers is greater because of "$3-1>3-2$".

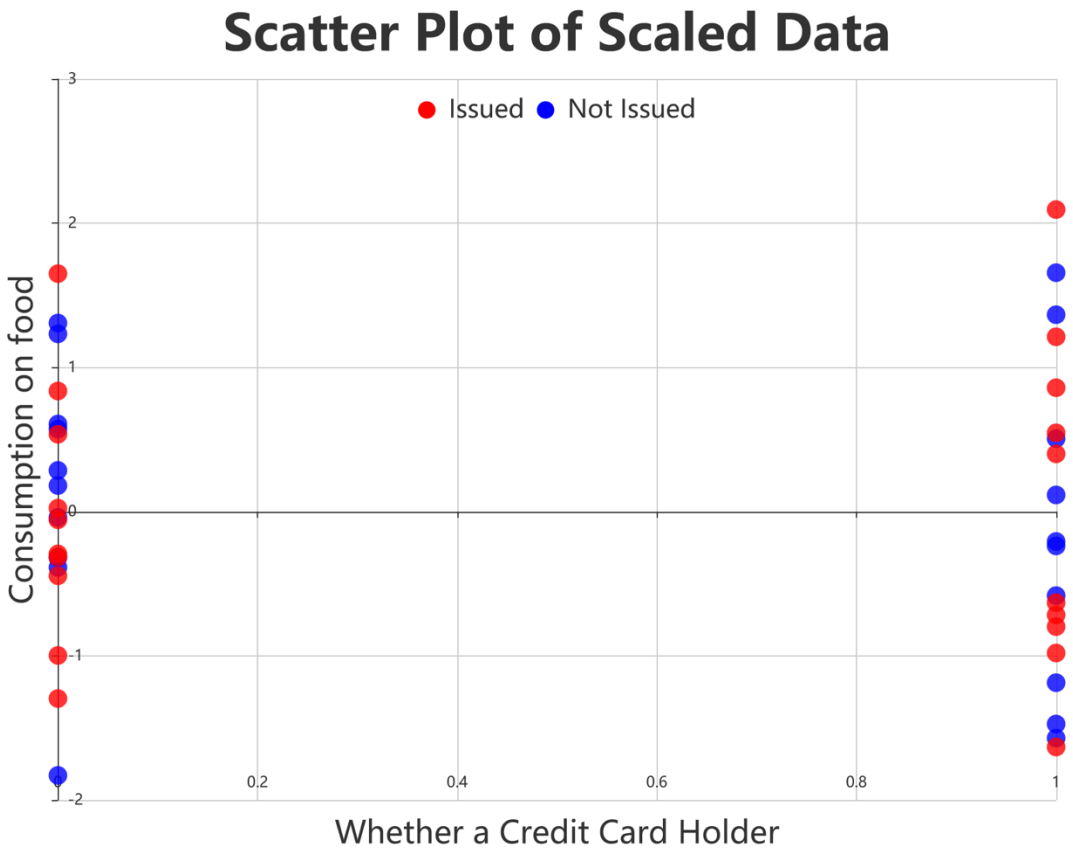
Secondly, if there are both numerical and categorical variables in the sample, the value of the numerical variables may make the category variables have a smaller impact on total distance, or even make the category variables "useless".



The picture above is an example. When y represents the customer's monthly consumption of food, because the y value of all samples is much larger than 1, the distance on the x-axis at this time has almost no effect on the total distance in the KNN. This is equivalent to discarding the information of whether the customer has another bank's credit card. This is obviously not the result we want.

Even if we can eliminate the impact of magnitude through a method called standardization, we still cannot solve the problem. We normalize the previous data to get the following

results. We will find that the distance on the x-axis is too large. To consider both types of customers at the same time requires a very large k, which is obviously unreasonable.



In summary, KNN is not suitable for processing data containing type variables because this algorithm is based on distance. And Logistic regression is suitable because it does not consider distances between samples, which can avoid problems shown above.

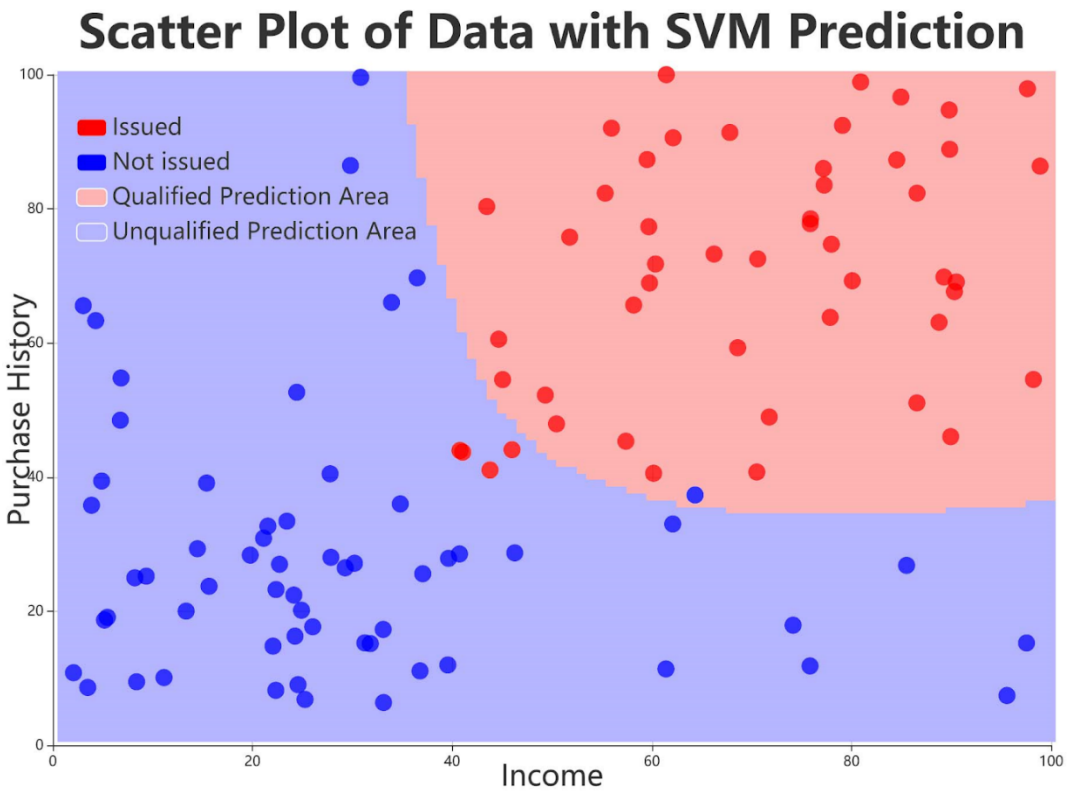
Scenario 3: Support Vector Machine (SVM)

Imagine such a scenario that the bank wants to decide whether users are qualified to be issued a credit card or not. The customers data only has two explanatory variables: a person's income and his purchase history. Also, the data tells us whether this person has ever been issued a credit card or not which is the response variable. We would like to build a classification model based on these data and want to make it as accurate as possible.

When looking at this dataset, we can find that the majority of issued users are located on the top right corner of the graph while the unissued users are mostly located on the other parts. The region where issued users distribute is surrounded by unissued users. If we try to separate them, the boundary definitely will not be a straight line or a line approximately straight. So logistic regression will perform badly on making classifications.

We notice that the pattern of how data is distributed may affect the choice of the best classifier. Different classifiers are suitable for different data distributions.

Since there are few unissued users' data distributed on the top left corner and bottom right corner, KNN will have difficulty distinguishing the data points in these regions.



However, recall the process of how SVM works when the data cannot be classified by a linear boundary on a plane. SVM will perform a trick to transform the data as if each data point is given an altitude on the chart. Then SVM can simply split the two groups by cutting in the middle altitude. In this example, the shape of the boundary given by SVM is like a semicircle. It will have higher accuracy.

We perform logistic regression, KNN with k equals to 4 and SVM on the data of this toy example. The results are as below. We can find that the result of classification accuracy also supports that SVM has the highest accuracy among these three classifiers.

Table 3 Accuracy of Three Classifiers

Classifier	Logistic Regression	KNN	SVM
Accuracy	83.64%	95.45%	98.18%

In summary, in this particular classification problem where the data has a circle-like boundary, SVM can make the best classification among the three classifiers. Therefore, using different classifiers for different problems is necessary.

Summary

In order to illustrate that different problems should use different classifiers, we selected three typical classifiers and created different banking business scenarios in previous sections.

First of all, logistic regression is a very flexible classifier with good interpretability. When a bank would like to change the qualification criteria according to its business strategy, logistic regression is a good choice so that the bank can achieve its purpose by adjusting the threshold and obtain different classification results to make decisions. And we can also take advantage of the interpretability of logistic regression.

Secondly, when the bank wants to divide customers into three or more categories, we should choose the KNN algorithm, because logistic regression will lose its interpretability under multi-levels classification problem and SVM is only applicable for binary cases. Besides, based on the characteristics of the data itself, KNN is no longer applicable when there are category variables in the data.

Last but not least, if the data has a circle-like nonlinear boundary and the bank wants to make classifications most accurately, SVM is best for this scenario since it can produce

a satisfactory nonlinear decision rule which best fits the distribution of the data. Whereas, logistic regression and KNN will have lower accuracy for this kind of data.

To conclude, different problems have different requirements, different data have different characteristics, and different algorithms have different advantages and disadvantages. What we should do is to find the best match that gives the best classification results to meet the needs. Like a butter knife is best for applying butter to breads and a meat cleaver is best for separate ribs. In other words, different problems should use different classifiers.