Proceedings of the Seventh International Conference on Research in Intelligent and Computing in Engineering pp. 295–298 DOI: 10.15439/2022R43 ISSN 2300-5963 ACSIS, Vol. 33

Bank Loan Analysis using Data Mining Techniques

Thi-Nhi Trinh
Faculty of Information Technology
Hung Yen University of Technology and Education
Hung Yen, Vietnam
nhittcntt@gmail.com

Hoang-Diep Nguyen
Faculty of Information Technology
Hung Yen University of Technology and Education
Hung Yen, Vietnam
diep82003@gmail.com

***Abstract*—Nowadays, a bank loan can provide people with cash to fund home improvements or start a business. However, some customers who are accepted with a loan cannot repay or someone usually repays in a delayed time. Therefore, to minimize losses, examining loan applications is particularly evident for the bank. This paper study on bank loan analysis using data mining techniques. We use association rules mining, clustering, and classification techniques on the applicant's profile to help the bank quickly decide for a loan applicant.**

***Index Terms*—bank loan analysis, association rules mining, clustering, classification.**

I. Introduction

The bank has to decide whether to go ahead with the loan approval or not, this was based on the applicants' profiles. There are two kinds of situation risks for banks that are associated with the bank's decision: (1) if the applicants can repay the loan - good credit risk, the bank approves their application, (2) if the applicants are unable to repay the loan - bad credit risk, the bank does not approve this loan. Analyzing patterns from customer profiles in the dataset and making the right decisions based on patterns plays an extremely important role in the Bank's business. In this paper, we present the approaches for analyzing the credit data that contains the loan applicants. First, we use the Apriori algorithm to find associations given the rules to predict whether an applicant is 'good' or 'bad'. Second, we use K-Means clustering to group applicants with similar attributes into different groups which helps a bank manager to understand easily the characteristics of the group of applicants. Finally, we use classification algorithms including Naïve Bayes and K Nearest Neighbor to classify an applicant into 'good' or 'bad', which is an important task to help a bank manager quickly make a decision for a loan applicant.

II. Background

A. Mining Association Rules

We summarize the association rules issue as follows:

Let I = {$i_1, i_2, \ldots, i_n$}, set I includes the elements.

D: a set of all transactions where each transaction T is a set of elements such that T $\subseteq$ I.

X, Y be a set of elements such that X, Y $\subseteq$ I. An association rule implies the form X $\Longrightarrow$ Y, where X $\subset$ I, Y $\subset$ I, X $\cap$ Y = $\varnothing$.

Support

The rule X$\Longrightarrow$ Y holds with support ***s*** if ***s***% of transactions in D contains X $\cup$ Y. Rules that have a ***s*** greater than user-specified support is said to have minimum support.

Confidence

The rule X$\Longrightarrow$ Y holds with confidence ***c*** if ***c***% of transactions in D contains X $\cup$ Y. Rules that have a ***c*** greater than user-specified confidences are said to have minimum confidence.

B. K-Means Clustering

K-means algorithm performs the following steps:

1. Select *k* applicants from *S* to be used as cluster centroids (random)
2. Assign applicants to clusters according to their similarity to the cluster centroids.
3. For each cluster, recalculate the cluster centroid using the newly calculated cluster members.
4. Go to step 2 until the process converges.

C. Classification

Classification is a data mining technique used to predict class levels for data instances. In this paper, we are using two different classification techniques to predict the class level including Naïve Bays and K Nearest Neighbors.

Naive Bayes Algorithm

The Bayesian approach determines the class of document *x* as the one that maximizes the conditional probability P(C | *x*).

$$P(C \mid x) = \frac{P(x \mid C)\, P(C)}{P(x)} \tag{1}$$

To compute P(x |C) we use equation as follows:

$$P(x \mid C) = P(x_1, x_2, \ldots, x_n \mid C) = \prod_{i=1}^{n} P(x_i \mid C) \tag{2}$$

Where $P(x_i|C)$ is calculated as the proportion of items from class C that include attribute value x_i; P(C) is the probability of sampling for class C.

K Nearest Neighbor

K Nearest Neighbor is to predict the class of a new sample using the class label of the closest sample. We can summarize K-Nearest Neighbor algorithm as the following steps:

(1) Determine parameter K = number of nearest neighbors.
(2) Calculate the distance between the query instance and all the training samples.
(3) Sort the distance and determine the nearest neighbors based on the K-th minimum distance.
(4) Gather the category *Y* of the nearest neighbors.
(5) Use the simple majority of the category of nearest neighbors as the prediction value of the query instance.

III. EXPERIMENTS

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	checking_sta	duration	credit_history	purpose	amount	savings_statu	employment	personal_stat	other_parties	property_mag	other_paymei	housing	existing_cred	job	own_telephor	foreign_worker	class
2	<0	lo_1_year	critical/other	radio/tv	1000_2000	no known sav	>=7	male single	none	real estate	none	own	two	skilled	yes	yes	good
3	0<=X<200	up_2_years	existing paid	radio/tv	up_2000	<100	1<=X<4	female div/de	none	real estate	none	own	one	skilled	none	yes	bad
4	no checking	lo_1_year	critical/other	education	up_2000	<100	4<=X<7	male single	none	real estate	none	own	one	unskilled resi	none	yes	good
5	<0	up_2_years	existing paid	furniture/equi	up_2000	<100	4<=X<7	male single	guarantor	life insurance	none	for free	one	skilled	none	yes	good
6	<0	1_2_years	delayed previ	new car	up_2000	<100	1<=X<4	male single	none	no known pro	none	for free	two	skilled	none	yes	bad
7	no checking	up_2_years	existing paid	education	up_2000	no known sav	1<=X<4	male single	none	no known pro	none	for free	one	unskilled resi	yes	yes	good
8	no checking	1_2_years	existing paid	furniture/equi	up_2000	500<=X<100(	>=7	male single	none	life insurance	none	own	one	skilled	none	yes	good
9	0<=X<200	up_2_years	existing paid	used car	up_2000	<100	1<=X<4	male single	none	car	none	rent	one	high qualif/se	yes	yes	good
10	no checking	lo_1_year	existing paid	radio/tv	up_2000	>=1000	4<=X<7	male div/sep	none	real estate	none	own	one	unskilled resi	none	yes	good
11	0<=X<200	up_2_years	critical/other	new car	up_2000	<100	unemployed	male mar/wid	none	car	none	own	two	high qualif/se	none	yes	bad
12	0<=X<200	lo_1_year	existing paid	new car	1000_2000	<100	<1	female div/de	none	car	none	rent	one	skilled	none	yes	bad
13	<0	up_2_years	existing paid	business	up_2000	<100	<1	female div/de	none	life insurance	none	rent	one	skilled	none	yes	bad
14	0<=X<200	lo_1_year	existing paid	radio/tv	1000_2000	<100	1<=X<4	female div/de	none	car	none	own	one	skilled	yes	yes	good
15	<0	1_2_years	critical/other	new car	1000_2000	<100	>=7	male single	none	car	none	own	two	unskilled resi	none	yes	bad
16	<0	1_2_years	existing paid	new car	1000_2000	<100	1<=X<4	female div/de	none	car	none	rent	one	skilled	none	yes	good
17	<0	1_2_years	existing paid	radio/tv	1000_2000	100<=X<500	1<=X<4	female div/de	none	car	none	own	one	unskilled resi	none	yes	bad
18	no checking	1_2_years	critical/other	radio/tv	up_2000	no known sav	>=7	male single	none	life insurance	none	own	two	skilled	none	yes	good
19	<0	up_2_years	no credits/all	business	up_2000	no known sav	<1	male single	none	car	bank	own	three	skilled	none	yes	good
20	0<=X<200	1_2_years	existing paid	used car	up_2000	<100	>=7	female div/de	none	no known pro	none	for free	one	high qualif/se	yes	yes	bad
21	no checking	1_2_years	existing paid	radio/tv	up_2000	500<=X<100(	>=7	male single	none	car	none	own	one	skilled	yes	yes	good
22	no checking	lo_1_year	critical/other	new car	up_2000	<100	1<=X<4	male single	none	car	none	own	three	skilled	yes	yes	good
23	<0	lo_1_year	existing paid	radio/tv	up_2000	500<=X<100(	1<=X<4	male single	none	real estate	none	rent	one	skilled	none	yes	good

Figure 1: The German Credit data set

A. *Dataset*

In this paper, we use the German Credit Data that contains data on 17 variables of 1000 past applicants for credit. Each applicant was rated as "good credit" (700 cases) or "bad credit" (300 cases). The attributes of each credit applicant are included as follows:

1. Status of account
2. Time in a month
3. History of credit
4. Purpose
5. Credit money
6. Savings bonds/account
7. Employment since
8. Personal status and sex
9. Other debtors / guarantors
10. Property
11. Payment plans
12. Housing
13. Number of existing credits
14. Job
15. Telephone
16. foreign worker
17. Class

B. *Experimental Results*

Weka is a software that helps people analyze data and build predictive models quickly and accurately. In this paper, we use Weka to run the Apriori algorithm, K-means clustering, Naïve Bays, and K Nearest Neighbor. We dynamically change the parameter to obtain a better result.

In this section, we present the evaluation of three data mining techniques for analyzing credit data. We run four algorithms in Weka including Apriori, K-means, Naïve Bays, and K-Nearest Neighbor, and analyze their results.

(1) Mining Association Rule with Apriori Algorithm

▪ **Run Apriori with default value of parameters by Weka**

Scheme: weka.associations.Apriori -N 10 -T 0 -C 0.9 -D 0.05 -U 1.0 -M 0.1 -S -1.0 -c -1

Set Minimum support is 0.6 (600 instances)

Set confidence is 0.9

Set number of executed is 8

Return best results

1. other_parties=none 907 ==> foreign_worker=yes 880 conf:(0.97)
2. job=skilled 630 ==> foreign_worker=yes 611 conf:(0.97)
3. other_parties=none other_payment_plans=none 742 ==> foreign_worker=yes 718 conf:(0.97)
4. other_parties=none housing=own 647 ==> foreign_worker=yes 625 conf:(0.97)
5. other_parties=none class=good 635 ==> foreign_worker=yes 611 conf:(0.96)
6. existing_credits=one 633 ==> foreign_worker=yes 609 conf:(0.96)
7. housing=own 713 ==> foreign_worker=yes 685 conf:(0.96)
8. other_payment_plans=none 814 ==> foreign_worker=yes 782 conf:(0.96)
9. class=good 700 ==> foreign_worker=yes 667 conf:(0.95)
10. other_payment_plans=none foreign_worker=yes 782 ==> other_parties=none 718 conf:(0.92)

▪ **Run Apriori with customization value of parameters**

In this case, we change parameter *car = 'true'* to enable class association rules to be mined instead of (general) association rules. Then we use *classIndex = -1* to make the last attribute taken as a class attribute. Here, the 'class' attribute in our dataset is the last attribute that has a 'good' or 'bad' applicant. Then, we run Apriori again, and the results are as follows:

Set minimum support is 0.2 (200 instances)

Set confidence is 0.9

Set number of executed is 16

Return best results

1. checking_status=no checking other_parties=none other_payment_plans=none housing=own 244 ==> class=good 228 conf:(0.93)
2. checking_status=no checking other_parties=none other_payment_plans=none housing=own foreign_worker=yes 236 ==> class=good 220 conf:(0.93)

3. checking_status=no checking other_parties=none other_payment_plans=none job=skilled 217 ==> class=good 202 conf:(0.93)

4. checking_status=no checking other_payment_plans=none housing=own 256 ==> class=good 238 conf:(0.93)

5. checking_status=no checking other_payment_plans=none housing=own foreign_worker=yes 247 ==> class=good 229 conf:(0.93)

6. checking_status=no checking other_parties=none other_payment_plans=none 313 ==> class=good 290 conf:(0.93)

7. checking_status=no checking other_payment_plans=none job=skilled 230 ==> class=good 213 conf:(0.93)

8. checking_status=no checking other_parties=none other_payment_plans=none foreign_worker=yes 303 ==> class=good 280 conf:(0.92)

9. checking_status=no checking other_payment_plans=none job=skilled foreign_worker=yes 223 ==> class=good 206 conf:(0.92)

10. checking_status=no checking other_payment_plans=none 330 ==> class=good 303 conf:(0.92)

(2) K-Mean Clustering

We run K-means algorithm with various the number of clusters from 2 to 6. And we found the averaged Sum of Squared Error as shown in Figure 2.

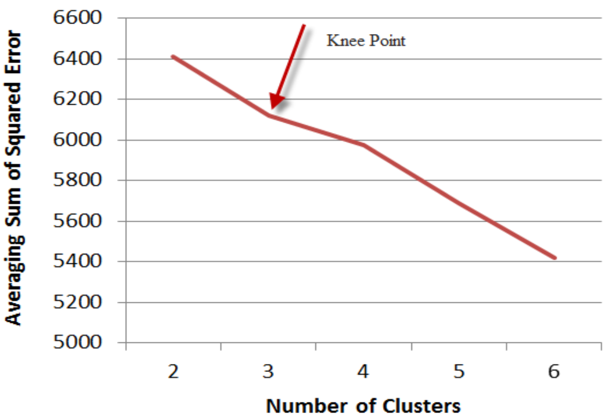

Figure 2: The averaged Sum of Squared Error

The knee point with k=3 shows that this data set should be grouped into 3 clusters. The results for running K-means with 3 clusters are illustrated in Figure 3 and Figure 4.

```
Number of iterations: 3
Within cluster sum of squared errors: 6123.0
Missing values globally replaced with mean/mode

Cluster centroids:
                                     Cluster#
Attribute              Full Data            0                1               2
                          (1000)        (518)            (209)           (273)
================================================================================
checking_status      no checking  no checking               <0       0<=X<200
duration               1_2_years    1_2_years        1_2_years       lo_1_year
credit_history     existing paid existing paid   existing paid  existing paid
purpose                 radio/tv      new car         used car        radio/tv
amount                   up_2000      up_2000          up_2000       1000_2000
savings_status              <100         <100             <100            <100
employment                1<=X<4       1<=X<4              >=7          1<=X<4
personal_status      male single  male single      male single     male single
other_parties               none         none             none            none
property_magnitude           car          car no known property    real estate
other_payment_plans         none         none             none            none
housing                      own          own              own             own
existing_credits             one          one              one             one
job                      skilled      skilled          skilled         skilled
own_telephone               none         none              yes            none
foreign_worker               yes          yes              yes             yes
class                       good         good             good            good



Time taken to build model (full training data) : 0.04 seconds

=== Model and evaluation on training set ===

Clustered Instances

0       518 ( 52%)
1       209 ( 21%)
2       273 ( 27%)
```

Figure 3: The results run K-means with 3 clusters

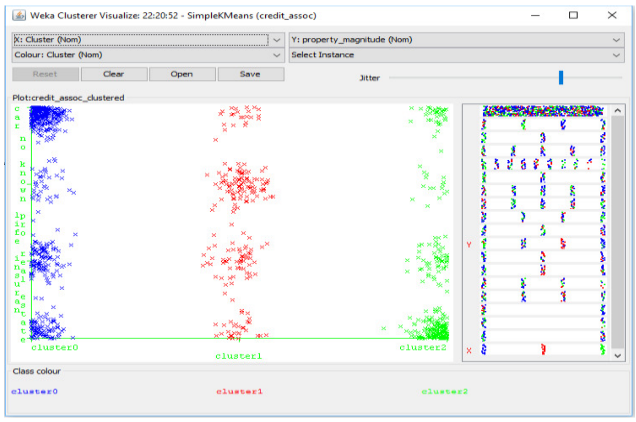

Figure 4: The results run K-means with 3 clusters and plot by Property Magnitude attribute

(3) Classification.

a) Data preparation

The data is separated into two *.csv files in which 70% data for training is stored in credit-train.csv and 30% data for testing is stored in credit -test.csv.

TABLE 1: THE DATASET SETTING FOR CLASSIFICATION

Class	**Training data (number of items)**	**Testing data (number of items)**
Good	480	220
Bad	220	80
Total	**700**	**300**

b) Classification with Weka

Run Naive Bayes Classify

- The results from training dataset

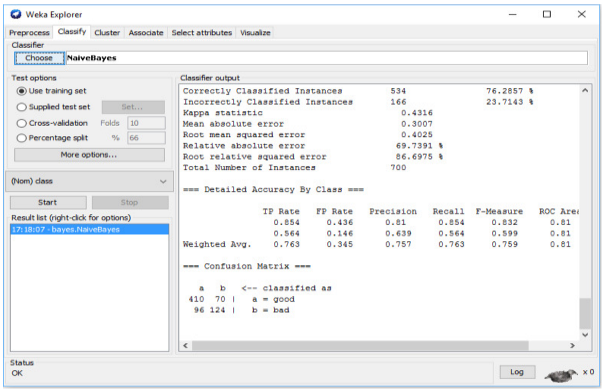

Figure 5: Illustrates the results for training with Naive Bayes

- The results for the testing dataset

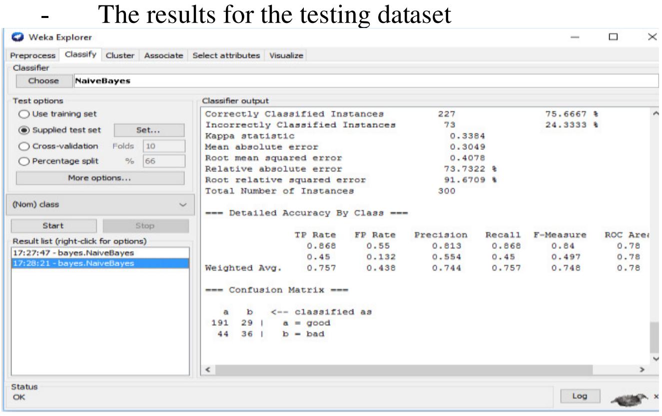

Figure 6: Illustates the results for testing with Naive Bayes

Run K-Nearest Neighbor Classify

- The results for training dataset

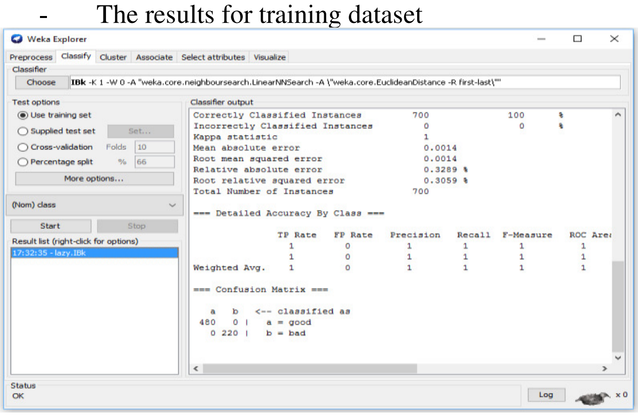

Figure 7: Illustates the results for training with KNN

- The results for testing dataset

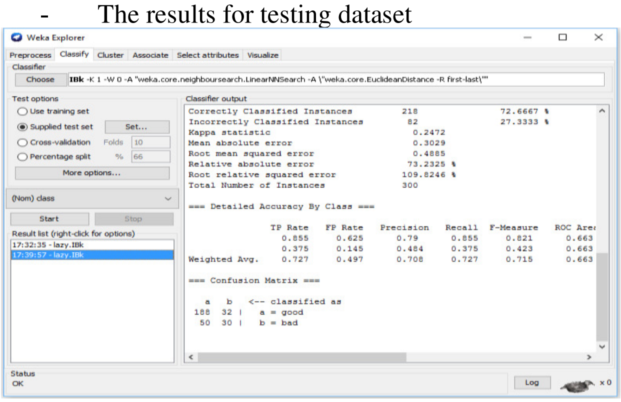

Figure 8: Illustates the results for testing with KNN

Analysis. We can summarize the results as the tables below.

TABLE 2: SUMMARY OF THE RESULTS

Methods	Traning (Accuracy)	Testing (Accuracy)
Naïve Bayes	534/480	227/300
KNN	700/700	218/300

TABLE 3: CONFUSION MATRIX FOR RUNNING NAÏVE BASE

True \ Assigned	C_1	C_2	Total
C1	191	29	220
C2	44	36	80

TABLE 4: CONFUSION MATRIX FOR RUNNING KNN

True \ Assigned	C_1	C_2	Total
C1	188	32	220
C2	50	30	80

As the results, we can see that the trained KNN algorithm better than the Naïve Bayes algorithm for the German Credit dataset that is given above. However, for the results of the classification on the testing dataset, Naïve Bayes outperforms the KNN algorithm.

V. CONCLUSIONS

In this paper, we present approaches for analyzing the credit data and provide the information for the bank manager to make a decision regarding loan approval. Besides, we can conclude as follows. (1) For mining association rules, (a) some association rules have very high confidence, but it is not important in particular; (b) the number of rules depends on the confidence and minimum support; (c) the confidence of a rule does not depending on the minimum support. (2) It is better to classify the German Credit dataset into 3 clusters. (3) Even though the KNN algorithm trained better than the Naïve Bayes algorithm for the German Credit dataset, Naïve Bayes outperforms the KNN algorithm for classifying new applicants for loan service.

REFERENCES

[1] G. Eason, B. Noble, and I. N. Sneddon, "On certain integrals of Lipschitz-Hankel type involving products of Bessel functions," Phil. Trans. Roy. Soc. London, vol. A247, pp. 529–551, April 1955. *(references)*

[2] J. Clerk Maxwell, A Treatise on Electricity and Magnetism, 3rd ed., vol. 2. Oxford: Clarendon, 1892, pp.68–73.

[3] I. S. Jacobs and C. P. Bean, "Fine particles, thin films and exchange anisotropy," in Magnetism, vol. III, G. T. Rado and H. Suhl, Eds. New York: Academic, 1963, pp. 271–350.

[4] K. Elissa, "Title of paper if known," unpublished.

[5] R. Nicole, "Title of paper with only first word capitalized," J. Name Stand. Abbrev., in press.

[6] Y. Yorozu, M. Hirano, K. Oka, and Y. Tagawa, "Electron spectroscopy studies on magneto-optical media and plastic substrate interface," IEEE Transl. J. Magn. Japan, vol. 2, pp. 740–741, August 1987 [Digests 9th Annual Conf. Magnetics Japan, p. 301, 1982].

[7] M. Young, The Technical Writer's Handbook. Mill Valley, CA: University Science, 1989.