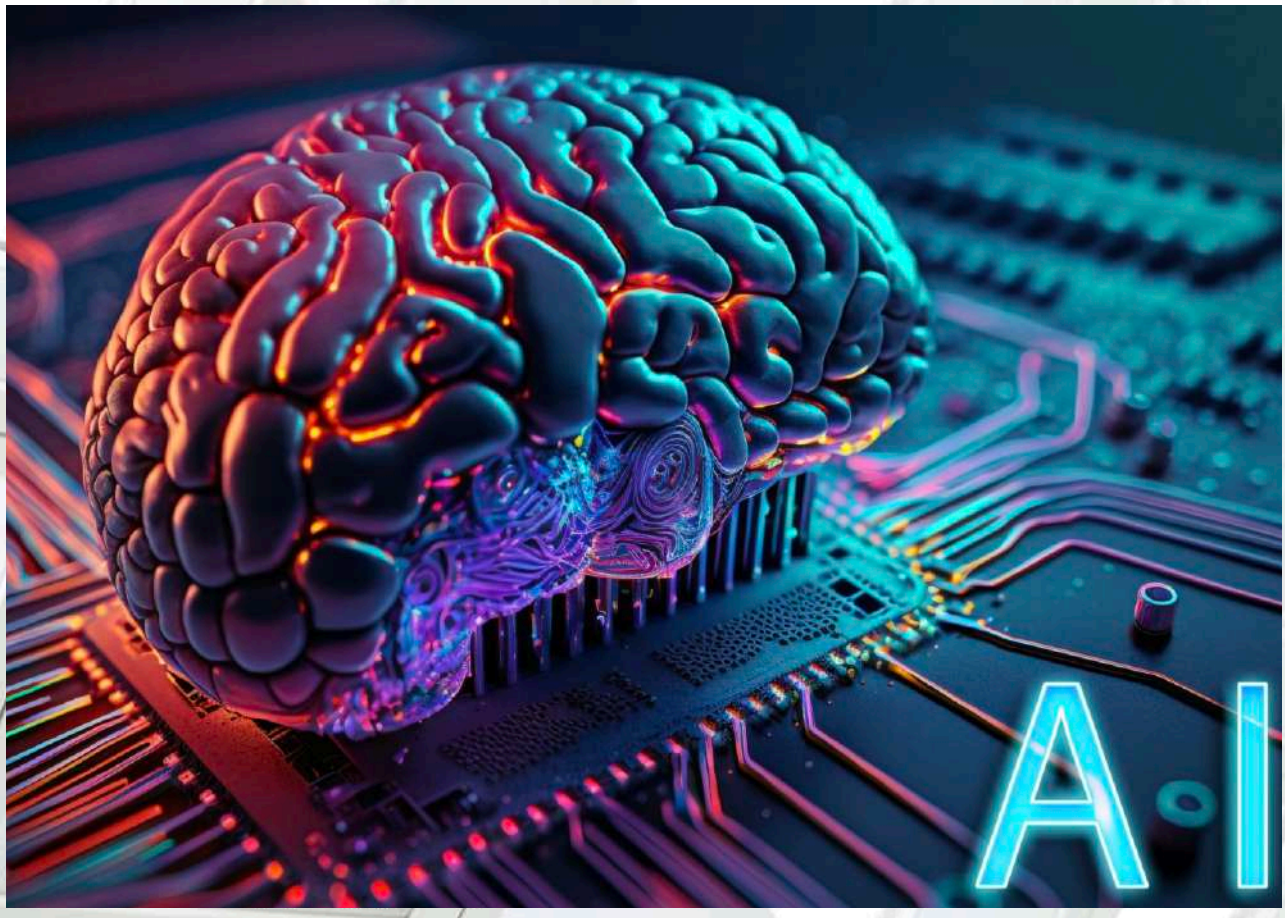
16 FEBRUARY 2024

ARTIFICIAL INTELLIGENCE FOR CREDIT SCORING



Caption

By Rathin Sinha

TABLE OF CONTENTS

ABSTRACT

The paper presents an analysis of Artificial Intelligence in credit scoring. Financial firms base their credit choices on credit scores. As a result of scientific and technological advancements, AI technology has joined the financial sector and ushered in a new era of personal credit inquiry.

WEKA software implements machine learning algorithms and presents the output accordingly for this project.

Weka, short for Waikato Environment for Knowledge Analysis provides machine learning, preprocessing, visualisation, and data mining techniques.

The German Credit Dataset, obtainable from the UCI machine learning repository, served as the dataset for this investigation. This dataset was used for classification purposes and a decision tree was built using it. J48 algorithm was applied to design the decision tree.

The report presents the features of the dataset, the J48 algorithm that is used for the classification of the dataset and the interpretation of the output.

Keywords: credit risk, finance, decision tree, weka

INTRODUCTION

The idea of credit scoring is crucial in the intricate world of finance since it helps establish a person's or a company's creditworthiness. Financial firms utilise credit scoring, a statistical process, to evaluate the creditworthiness of people or companies applying for loans or credit cards.

When credit reporting first started, Individuals received financial services from CSPs based on credit reports, small and medium-sized enterprises, and businesses.

Due to the abundance of Big data and the rise in computational power, credit scoring methods have evolved from traditional techniques to computational methods involving AI and ML.

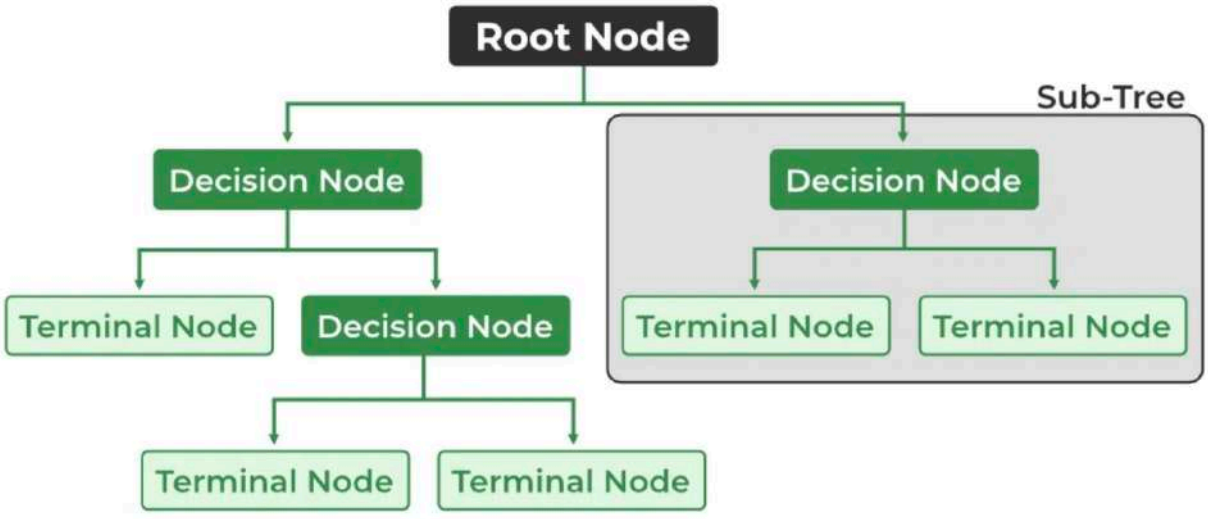
The paper presents an approach based on computational intelligence and data mining to find unnoticed tendencies and base forecasts on accuracy.

We employ the German credit data set from the UCI Machine Learning Repository, which includes a sample of 1000 debtors categorised as either "good" or "bad," to achieve the goal (UCI Machine Learning Repository, German data set). [1]

METHODOLOGY

This section of the report focuses on the classification-based supervised learning technique used, along with the rationale behind the German Credit dataset selection.

DECISION TREE



DECISION TREE

Decision trees, a supervised learning method, are mostly employed in regression and classification applications. They are like flowcharts in that they have leaf nodes that indicate algorithmic results, branches that reflect rules, and core nodes that describe features. When one of the stopping criteria is met, the training data is recursively separated into subgroups depending on attribute values [3].

- ***Root Node:*** *It represents the entire dataset and is the top node in the tree. It expedites the decision-making process.*

- ***Decision Node:*** *A node that represents an input feature selection. Internal nodes can be connected to leaf nodes or other internal nodes by branching off of them.*

- ***Terminal Node:*** *A node representing a number without children or a class name.*

- ***Splitting:*** *The division of a node using a split criterion and a chosen feature into two or more sub-nodes.*

- ***Sub-Tree:*** *A subsection of the decision tree begins at an internal node and ends at a leaf node.*

- ***Parent Node:*** *The node that divides into one or more offspring nodes.*

- ***Child Node:*** *The nodes that break away from their parent node [3].*

Benefits of using Decision Tree:

1. Minimal data preparation is needed.

2. Easy to comprehend and interpret. One can visualise trees.

3. It is helpful to think through any situation that can lead to an issue.

4. Able to manage issues with several outputs.

5. Performs well even when the true model used to create the data slightly deviates from its assumptions.

However, large variance in decision tree analysis is one potential barrier.

DATA

This dataset was chosen for this project for various reasons.

- It is available in the arff format. This makes it easy for preprocessing in WEKA
- It has no missing values. This makes the prediction more accurate.
- The dataset is multivariate, which assists in identifying trends that might be insignificant when looking at each variable separately.
- The data has 1000 instances and 20 variables.

While the dataset has 20 attributes, for this report, the focus was only on a select few.

LIST OF VARIABLES USED

1. *Status of existing checking account (*Type - Qualitative)
2. *Credit History (*Type - Qualitative)
3. *Savings Account/Bonds* (Type - Qualitative)
4. *Number of existing credits at this bank* (Type - Numerical)
5. *Job* (Type - Qualitative)
6. *Class* (Type - Qualitative)

J48 ALGORITHM

To understand the J48 algorithm, it is imperative to know about ID3 and C4.5 algorithms. The ID3 algorithm is an algorithm for creating decision trees. By evaluating the values of their properties, it classifies objects or records. Starting with a collection of information and attributes, it constructs a decision tree using an order from top to bottom for the provided data. The results are

utilised to divide the records. One characteristic is tested at each tree node on the basis of optimising the information gain measurement and minimising the entropy assessment. Until all of the entries in a subtree are homogenous—that is, all of the records belong to the same class—this process is repeated recursively. These uniform records end up as a decision tree leaf node. C4.5 is the evolution of ID3. It splits the records recursively to create a decision tree for a specific set of data. Nominal and numerical attributes are taken into account by the C4.5 algorithm. [4]

The version of C4.5 that is available on *WEKA* is the J48 algorithm. It has many supplementary capabilities, for example, the capacity to derive rules, account for missing data, prune decision trees, and continuous attribute value ranges. [4]

RESULTS

To demonstrate the results of the analysis, let's focus on three attributes, credit history, savings status, and job.

```
Attribute 3:  (qualitative)
        Credit history
        A30 : no credits taken/
          all credits paid back duly
              A31 : all credits at this bank paid back duly
        A32 : existing credits paid back duly till now
              A33 : delay in paying off in the past
        A34 : critical account/
          other credits existing (not at this bank)
```

Figure 1: Attribute 'Credit History'

Figure 1 shows one of the six attributes used for credit scoring analysis. Credit history is categorised into five sets ranging from A30 to A34

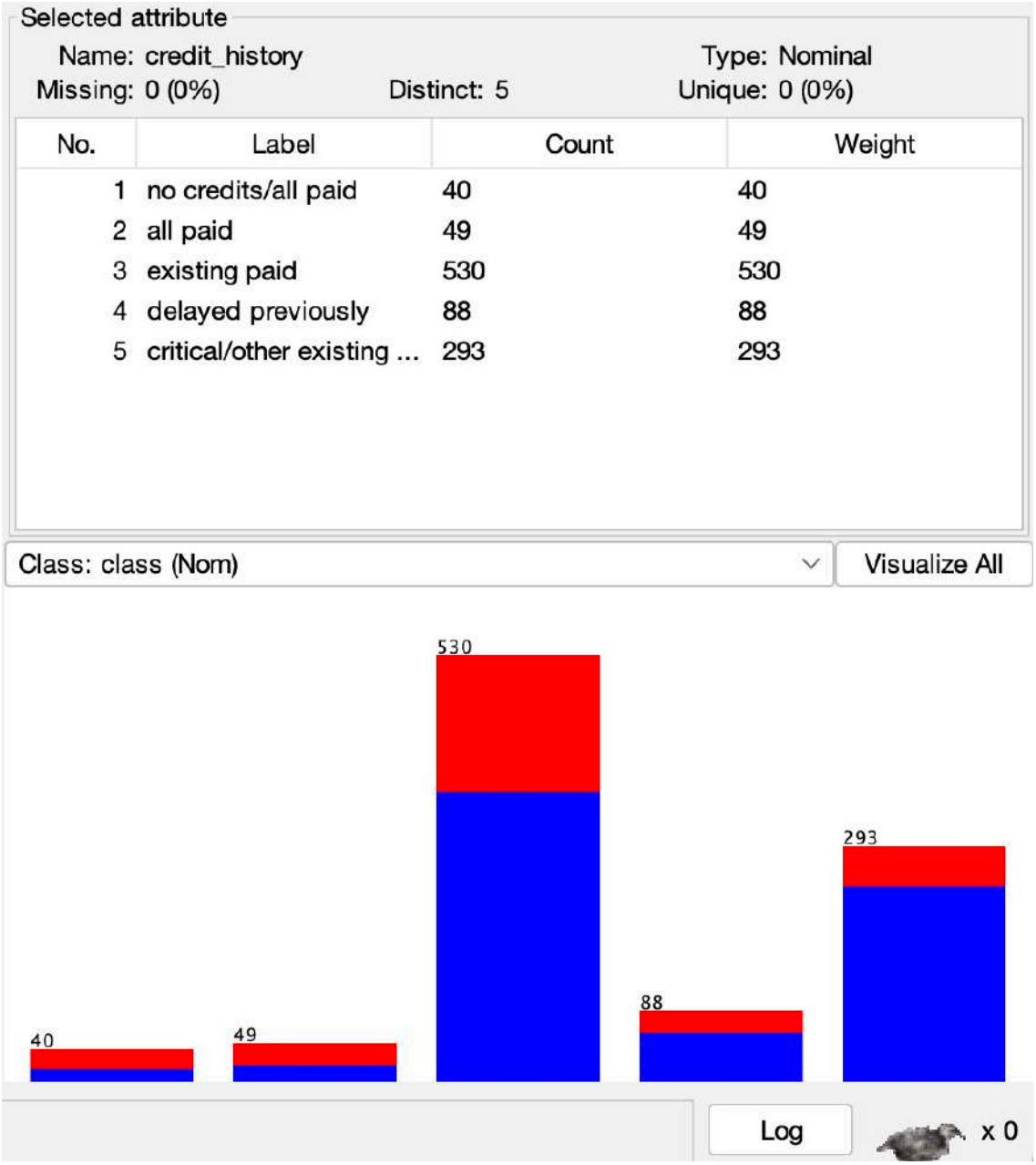


Figure 2: Evaluating credits based on credit history

Figure 2 presents clients' credit history for credit evaluation. Blue color represents "good" debtors and red color represents "bad" debtors. Clients who took no credits and those who paid all credits had the lowest counts, at 40 and 49, respectively. The majority of the clients under each of these groups were labelled as "bad" debtors. Clients with critical accounts had the largest proportion of "good" creditors. The number of clients with existing credits was

the highest among the five categories, with 60% labelled as "good" borrowers and only 10% classified as "bad" debtors.

```
Attibute 6:  (qualitative)
        Savings account/bonds
        A61 :          ... <  100 DM
        A62 :   100 <= ... <  500 DM
        A63 :   500 <= ... < 1000 DM
        A64 :          .. >= 1000 DM
              A65 :   unknown/ no savings account
```

Figure 3: Attribute “Savings account”

As shown in Figure 3, the savings account is divided into five categories depending on the balance present. The categories range from A61 to A65.

From Figure 4, savings status has five possibilities.

- Savings up to 100 DM
- Savings up to 500 DM and more than 100 DM
- Savings up to 1000 DM and more than 500 DM
- Savings of more than 1000 DM
- No savings account

Clients with up to 100 DM savings, had the highest count at 603. Of them, more than 60% were classified as “good” debtors. For clients with savings up to 1000 and above, the counts were 63 and 48 respectively, and almost 90% were categorised as “good” debtors in each of the two categories. Clients with unknown savings had the second-highest count at 183 and more than 80% of them were predicted as “good” debtors.

Selected attribute

Name: savings_status Type: Nominal
Missing: 0 (0%) Distinct: 5 Unique: 0 (0%)

No.	Label	Count	Weight
1	<100	603	603
2	100<=X<500	103	103
3	500<=X<1000	63	63
4	>=1000	48	48
5	no known savings	183	183

Class: class (Nom) Visualize All

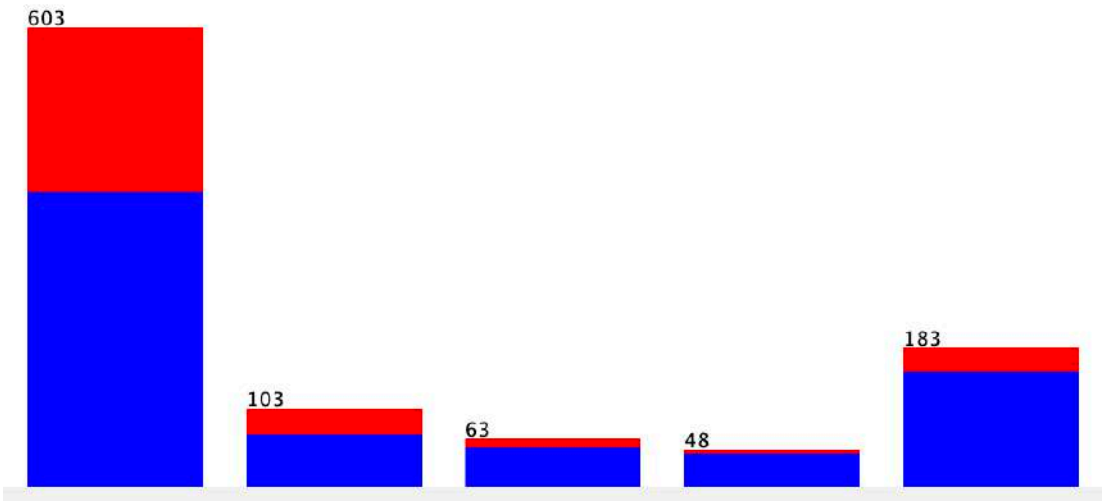


Figure 4: Credit evaluation based on savings status

Based on profession, the clients were segregated into four categories. Figure 5 describes the four job categories ranging from A171 to A174.

```
Attribute 17: (qualitative)
        Job
        A171 : unemployed/ unskilled  - non-resident
        A172 : unskilled - resident
        A173 : skilled employee / official
        A174 : management/ self-employed/
           highly qualified employee/ officer
```

Figure 5: Attribute “Job”

In Figure 6, the number of clients that were either unemployed or unskilled and were non-residents was 22. Of them, the majority were described as “good” debtors. 200 unskilled clients were also residents. 30% of them were “bad” debtors.

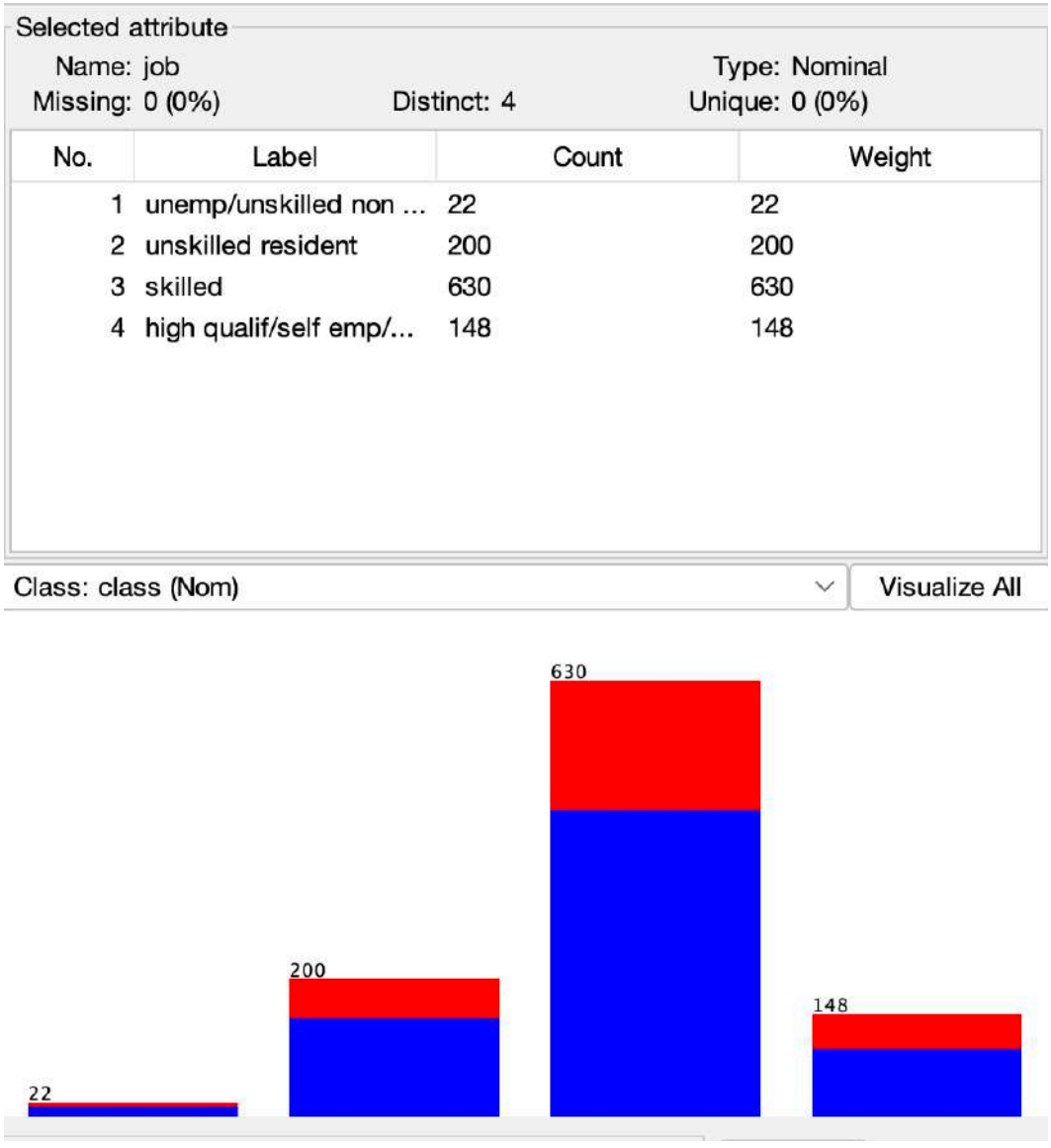


Figure 6: Credit evaluation based on job type

Skilled officials had the highest count which was 630, with 60% of them being labelled as “good” debtors. 148 people were either highly qualified, self-employed, were in management or were officers. And 30-35% of them were “bad” debtors.

Figure 7 displays the output information of the classifier. The classification was carried out on 1000 instances for 6 attributes using J48 -C 0.25 -M 2 classifier. The chosen test mode is K-fold cross-validation (where *k = 10*). K-fold cross-validationis a technique for determining model performance.

```
=== Run information ===

Scheme:       weka.classifiers.trees.J48 -C 0.25 -M 2
Relation:     german_credit-weka.filters.unsupervised.attribute.Remove-R2,4-5,7-15,18-20
Instances:    1000
Attributes:   6
              checking_status
              credit_history
              savings_status
              existing_credits
              job
              class
Test mode:    10-fold cross-validation

=== Classifier model (full training set) ===
```

Figure 7: Output Information

In figure 8, the size of the tree is 43, which means that the total number of nodes is 43. Number of leaves is 33. Leaves refer to the nodes that do not split further. Therefore, 33 nodes in the tree did not split.

For clients having an account balance of less than zero and no credits, 13 were properly predicted as 'bad' debtors, while 3 were incorrectly labelled as 'bad' debtors. This is shown as 'bad (13.0/3/0)'.

In the same category of account balance(<0), clients that had paid all the existing credits were split into different categories based on savings account. Those who had balance up to 100 DM in savings account were further divided into four groups depending on the profession. Of the unskilled non-residential clients, 2 were properly predicted as “bad” debtors and none were misclassified. 4 of the clients with savings of more than 1000 DM were correctly labelled as “good” debtors.

```
J48 pruned tree
------------------

checking_status = <0
|   credit_history = no credits/all paid: bad (13.0/3.0)
|   credit_history = all paid: bad (22.0/6.0)
|   credit_history = existing paid
|   |   savings_status = <100
|   |   |   job = unemp/unskilled non res: bad (2.0)
|   |   |   job = unskilled resident: good (30.0/13.0)
|   |   |   job = skilled: bad (74.0/31.0)
|   |   |   job = high qualif/self emp/mgmt: good (18.0/7.0)
|   |   savings_status = 100<=X<500: bad (8.0/3.0)
|   |   savings_status = 500<=X<1000: good (3.0)
|   |   savings_status = >=1000: good (4.0)
|   |   savings_status = no known savings
|   |   |   job = unemp/unskilled non res: bad (0.0)
|   |   |   job = unskilled resident: good (3.0)
|   |   |   job = skilled: bad (15.0/5.0)
|   |   |   job = high qualif/self emp/mgmt: bad (3.0/1.0)
|   credit_history = delayed previously
|   |   job = unemp/unskilled non res: bad (0.0)
|   |   job = unskilled resident: good (3.0/1.0)
|   |   job = skilled: bad (9.0/1.0)
|   |   job = high qualif/self emp/mgmt: bad (0.0)
|   credit_history = critical/other existing credit: good (67.0/18.0)
checking_status = 0<=X<200
|   savings_status = <100
|   |   job = unemp/unskilled non res: good (3.0/1.0)
|   |   job = unskilled resident: good (38.0/12.0)
|   |   job = skilled: good (79.0/34.0)
|   |   job = high qualif/self emp/mgmt: bad (32.0/12.0)
|   savings_status = 100<=X<500
|   |   credit_history = no credits/all paid: bad (4.0/1.0)
|   |   credit_history = all paid: bad (6.0/1.0)
|   |   credit_history = existing paid: bad (22.0/9.0)
|   |   credit_history = delayed previously: good (10.0/2.0)
|   |   credit_history = critical/other existing credit: good (5.0/1.0)
|   savings_status = 500<=X<1000: good (11.0/3.0)
|   savings_status = >=1000
|   |   existing_credits <= 1: good (9.0/1.0)
|   |   existing_credits > 1: bad (5.0/2.0)
|   savings_status = no known savings: good (45.0/7.0)
checking_status = >=200: good (63.0/14.0)
checking_status = no checking: good (394.0/46.0)

Number of Leaves  :     33

Size of the tree :      43


Time taken to build model: 0.01 seconds
```

Figure 8: Pruned Tree

Out of the customers who had up to 200 DM in checking account and more than 500 DM in savings account, 11 were correctly labelled as “good” debtors and 3 were inaccurately labelled as “bad” debtors as represented by “good (11.0/3.0)”.

Figure 9 shows the decision tree that has been generated for this analysis. The tree has 33 terminal nodes and 43 nodes in total.

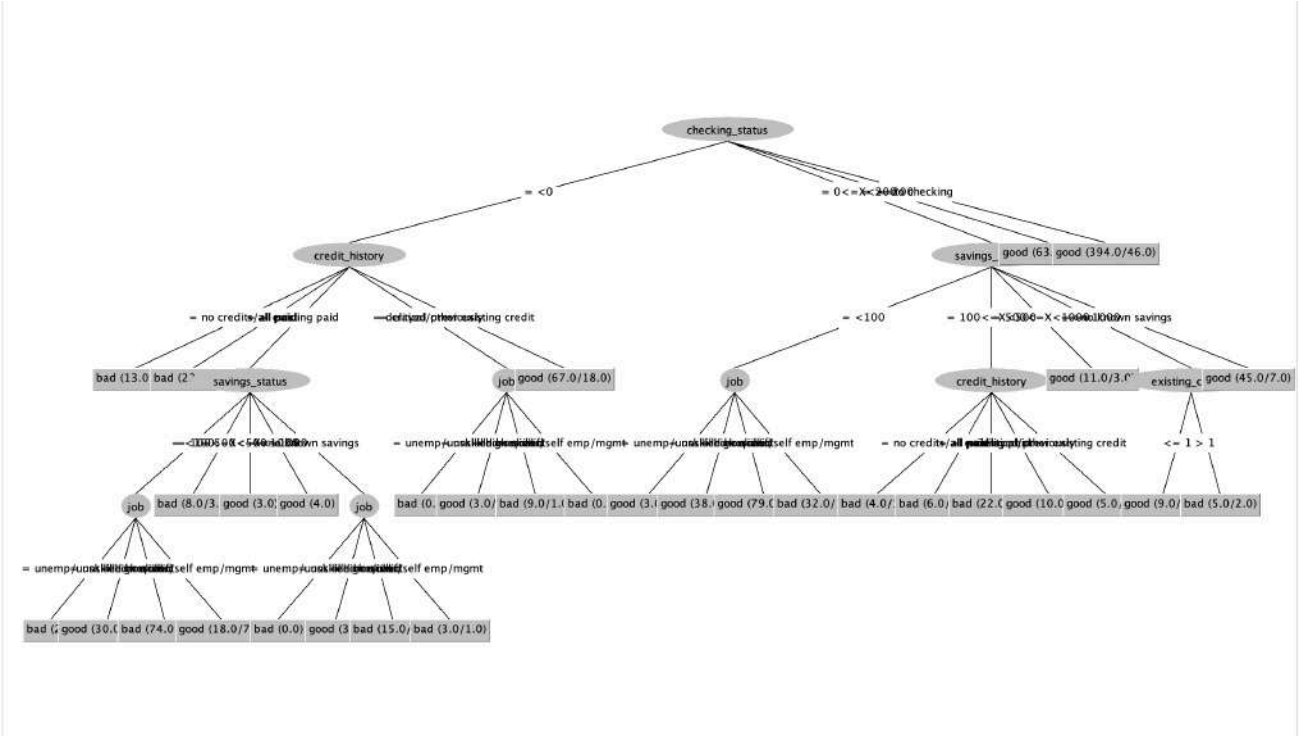


Figure 9: Decision tree visualization

EFFICIENCY MEASURE

The total number of instances is 1000, out of which 717 have been correctly classified. Additionally, a confusion matrix is shown. The confusion matrix can be divided into four categories: True Positives (TP), False Positives (FP), True Negatives (TN), and False Negatives (FN).

```
=== Stratified cross-validation ===
=== Summary ===

Correctly Classified Instances         717               71.7    %
Incorrectly Classified Instances       283               28.3    %
Kappa statistic                          0.2623
Mean absolute error                      0.3593
Root mean squared error                  0.4396
Relative absolute error                 85.5056 %
Root relative squared error             95.9249 %
Total Number of Instances             1000

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 0.863    0.623    0.764      0.863    0.810      0.270    0.702     0.813     good
                 0.377    0.137    0.541      0.377    0.444      0.270    0.702     0.464     bad
Weighted Avg.    0.717    0.477    0.697      0.717    0.700      0.270    0.702     0.709

=== Confusion Matrix ===

   a   b   <-- classified as
 604  96 |   a = good
 187 113 |   b = bad
```

Figure 10: Summary of the analysis

Out of 1000 instances, 604 out of 1000 examples are True Positives (TP), implying that these 604 instances were appropriately classified as 'good' debtors. 113 customers were accurately labelled as True negative.

The total number of misclassifications is 283 of which 187 were False Positves(misclassified as good) and 96 were False Negatives(misclassified as bad).

To compute True Positive rate, TP/(TP+FN) is applied, and the numerical result for class 'good' is 0.863. The True Negative Rate or True Positive Rate for class 'bad' is 0.377, determined using the formula TN/(TN+FP). The False Positive rate is determined with the formula FP/(FP+TN). The False Negative rate is estimated by using FN/(FN+TP).

The proportion of accurately classified positive instances to all positively classified instances is known as precision.

$$Precision = TP/(TP+FP)$$

Recall, is the proportion of correctly identified positive cases to all positive instances in the dataset

$$Recall = TP/(TP+TN)$$

CONCLUSION

The J48 decision tree algorithm achieved a 71.7% efficiency rate. The algorithm's output identified more customers as "good" debtors than not.

REFERENCES

1. UCI Machine Learning Repository, German data set, available from: https://archive.ics.uci.edu/dataset/144/statlog+german+credit+data

2. Škegro, Frane; Zoroja, Jovana; Šimičević, Vanja (2017): Credit Scoring Analysis: Case Study of Using Weka, In: Proceedings of the ENTRENOVA - ENTerprise REsearch InNOVAtion Conference, Dubrovnik, Croatia, 7-9 September 2017, IRENET - Society for Advancing Innovation and Research in Economy, Zagreb, pp. 88-93

3. Decision Tree. Available from: https://www.geeksforgeeks.org/decision-tree/

4. Y.Fakir , M. Azalmad, R. Elaychi (2020) Study of The ID3 and C4.5 Learning Algorithms . *Journal of Medical Informatics and Decision Making* - 1(2):29-43. https://doi.org/10.14302/issn.2641-5526.jmid-20-3302