School of Computing, Maths and Digital Technologies

Mark Bellingham

14032098

Data Mining Project

2017

DATA ENGINEERING

Contents

1 Introduction

1.1 Aims and Objectives

This report aims to analyse the data of 2000 loan customers and decide how this data can be used to make predictions about the reliability of future customers.

1.2 The Task

The task is to consider the benefits and disadvantages of different data mining algorithms and choose the most appropriate one for this task. The next task is to prepare the data for the modelling process, by choosing relevant attributes, and removing and/or modifying noise and outliers as necessary according to strict criteria. A series of experiments will be performed on this cleaned dataset to determine which methods provide the best accuracy in determining whether a loan outcome should be successful.

1.3 The Data

The data for this report is a set of 2000 records of people who have previously taken out a loan. It contains personal information such as name, address, age and gender; and financial information such as loan amount, income, CCJs (County Court Judgements) and whether the loan was repaid or not.

1.4 Terminology

Weka – Software providing a GUI interface that is used to perform analysis on datasets.

Dataset – A collection of data records that are related to each other in some way.

Attribute – Name given to a collection of data values that belong in the same group, such as gender or loan amount.

Nominal – Data whose purpose is a name or label. It can be in number form but the value has no numeric meaning.

Numeric – Data that can be identified as belonging to a number scale where a higher value means that the object has more of that item.

Continuous – Can have any numerical value or one that is within a set range.

Discrete – Can only have particular values, as defined by the dataset and there must be clear confines to those values. Nominal, binary and rank-ordered data are always discrete. (Boslaugh, 2013)

Noise – Where data has a value that is acceptable for the attribute but has been recorded incorrectly. (Bramer, 2016)

Outlier – Where data has the correct label but its value is outside the accepted range for that attribute. (Bramer, 2016)

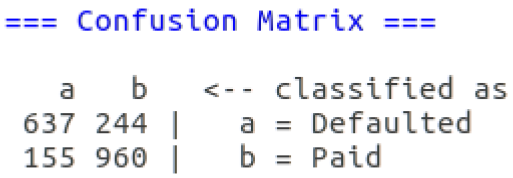
Algorithm – A set of rules or instructions that are performed in a given order to achieve a recognisable goal.

Standard Deviation – The difference between the minimum and maximum value in an attribute.

Classification – Determining if an outcome should be given one label or another.

Clustering – Grouping data values together according to various criteria, such as how close different postcodes are in location.

Confusion Matrix – A matrix showing how many of each class correctly or incorrectly identified.



```
=== Confusion Matrix ===

   a   b   <-- classified as
 637 244 |   a = Defaulted
 155 960 |   b = Paid
```

2 Data Summary

2.1 Table of the data in the dataset

Attribute Name	Data Type	Discrete / Continuous	Min	Max	Mean	Standard Deviation	Values
Customer ID	Numeric	Continuous	555574	1110985	837077.517	160763.319	
Forename	Nominal	Discrete					
Surname	Nominal	Discrete					
Age	Numeric	Continuous	17	89	52.912	20.991	
Gender	Nominal	Discrete					F, M, Male, Female, H, D, N, 1, 0
Years at Address	Numeric	Continuous	1	580	18.526	23.202	
Employment Status	Nominal	Discrete					Unemployed Self- Employed Employed Retired
Country	Nominal	Discrete					
Current Debt	Numeric	Continuous	0	9980	3309.325	2980.629	
Postcode	Nominal	Discrete					
Income	Numeric	Continuous	3000	220000	38319	12786.506	
Own Home	Nominal	Discrete					Rent Own Mortgage
CCJs	Numeric	Continuous	0	100	1.052	2.469	
Loan Amount	Numeric	Continuous	13	54455	18929.628	12853.189	
Outcome	Nominal	Discrete					Paid Defaulted

2.2 Individual attribute analysis

None of the attributes have missing data.

Customer ID - Not useful

The ID number is an arbitrary value and means nothing other than to identify a single record.

Forename - Not useful

A person's name has no bearing on whether they are likely to pay back a loan. Since nearly everyone has a unique name, it would be impossible to provide any kind of statistical analysis using this data.

Surname - Not useful

A person's name has no bearing on whether they are likely to pay back a loan. Since nearly everyone has a unique name, it would be impossible to provide any kind of statistical analysis using this data.

Age - Somewhat useful

A person nearing or past retirement age might be more likely to have income or health issues in the future. UK Law states that a person between the ages of 14 and 18 needs a guarantor to take out a loan (Citizens Advice, n.d.) In practise a bank is likely to refuse a loan to a person under the age of 18 (HSBC, n.d.)

Gender – Somewhat useful

The data for this attribute has some noise issues. At first glance Male seems similar to M and Female to F. Genders listed as H, N, D, 1 and 0 are more difficult, there are 8 records listed in this way. There are also some other anomalies where a person who has a name for one gender has been classified as the other.

Years at Address - Somewhat useful

Someone who has spent a long time at one address is likely to have a stable and reliable lifestyle. Should only be used in conjunction with other factors. There are outliers with this attribute. It is impossible for someone to have lived at one address for 580 years. The data should be investigated to see if there is an obvious mistake, or it may be safer to discard these entries. This problem affects 4 records.

Employment Status – Useful

An employed person will have a regular income, which they can use to pay the loan instalments. Someone who is unemployed or retired is more likely to face income pressures in the future, unless they have an income stream from elsewhere.

Country – Useful

A person who is resident in another country is unlikely to be approved for a loan in the UK unless they can also show strong ties to this country because of the difficulty in chasing up defaulters. There are 4 entries where a customer is shown to be living in a country other than the UK, but each of these examples also has a UK postcode. These will need to be examined in more detail.

Current Debt – Useful

A person's current debt levels need to be checked against the total amount that they are judged to be able to borrow to ensure that they are not borrowing beyond their means.

Postcode – Somewhat useful

A postcode is useful if the client wishes to perform statistical analysis. It is possible that people living in low-income areas are more likely to suffer income pressures in the future. It would be possible to do clustering on the first part of the postcode.

Income - Useful

A person's income is used to predict they amount that they would reasonably be able to pay back. This should be considered together with the person's current debt when deciding how much they can borrow. There are a couple of very high outliers, the highest of which is very likely to be a mistake because it contains many inconsistencies.

Own Home – Somewhat useful

Someone who owns their own home is more likely to have a stable lifestyle and is more likely to be able to keep up with payments. Being accepted for a mortgage also shows that the person has been judged previously to be responsible enough to take on a large responsibility. Someone who does not own their own home is not necessarily a high risk and other factors such as employment status, years at address and income will play a part. Someone who owns their own home is likely to have lower outgoings and therefore lower income pressures.

CCJs - Useful

Someone who has recently shown unreliability in keeping up with payments to the extent that the creditor had to take them to court is likely to be of a high risk in the future. There are a couple of outliers here, entries that are significantly higher than the rest. One record is listed with 10 CCJs and another with 100. (OnlineMortgageAdvisor, 2016) states that some specialist lenders will lend to people with 2 CCJs or even those with 3 or 4 in certain circumstances, so a person having 10 CCJs seems possible though high, but someone having 100 CCJs is inconceivable.

Loan Amount – Useful

A person should only be allowed to borrow an amount up to that which they are judged to be able to pay back. Someone with a large loan but only a small income would be deemed higher risk.

Outcome – Useful

Someone who has previously been judged to be reliable enough to pay back a loan should also be considered in the future. If a person has previously defaulted on a loan, they would be considered higher risk. This attribute is the one that the classifiers will be trying to predict. In the dataset there are 1118 records classed as Paid and 882 classed as Defaulted.

3 Data Mining Algorithm Selection

3.1 Algorithm Selection

When choosing which algorithms to use to analyse the data, the first thing to do is to decide whether this is a classification problem or a clustering problem. It can be determined that this is a classification problem because the analysis is trying to identify whether a person is likely to pay back their loan or not. While it is possible to perform clustering analysis on some aspects of the data, such as the postcode information, this report will focus on the classification problem.

Classifiers use a training set which is a sample of the data that is being analysed. The objective is to use this sample to create a set of rules that can then be used to make predictions about new data.

There are many different classification algorithms such as Naïve Bayes and C4.5 (also known as J48). This report will analyse the data using two of them. To choose which two to use, some basic tests were performed using the unmodified data with 7 algorithms and two methods, 50 : 50 Training Strategy and 10 Fold Cross Validation (which are explained in a later chapter). The results of these tests can be seen in Appendix 8.1. In both tests J48 and Bayes Net performed the best and will be used in the rest of the report.

The Zero R classifier is included simply as a gauge. This classifier works out which is the most common outcome and defines all outcomes as this. It is useful because if a different classifier scores less overall accuracy than Zero R, you can see quite quickly that it is not particularly good for this study.

3.2 C4.5 Algorithm

The C4.5 algorithm was published by Ross Quinlan in 1993 and is implemented in Weka as J48. C4.5 is itself an improvement on an earlier algorithm called ID3, also published by Quinlan in 1979. The program creates a decision tree by generating classifiers that are either:

- *a leaf,* indicating a class, or

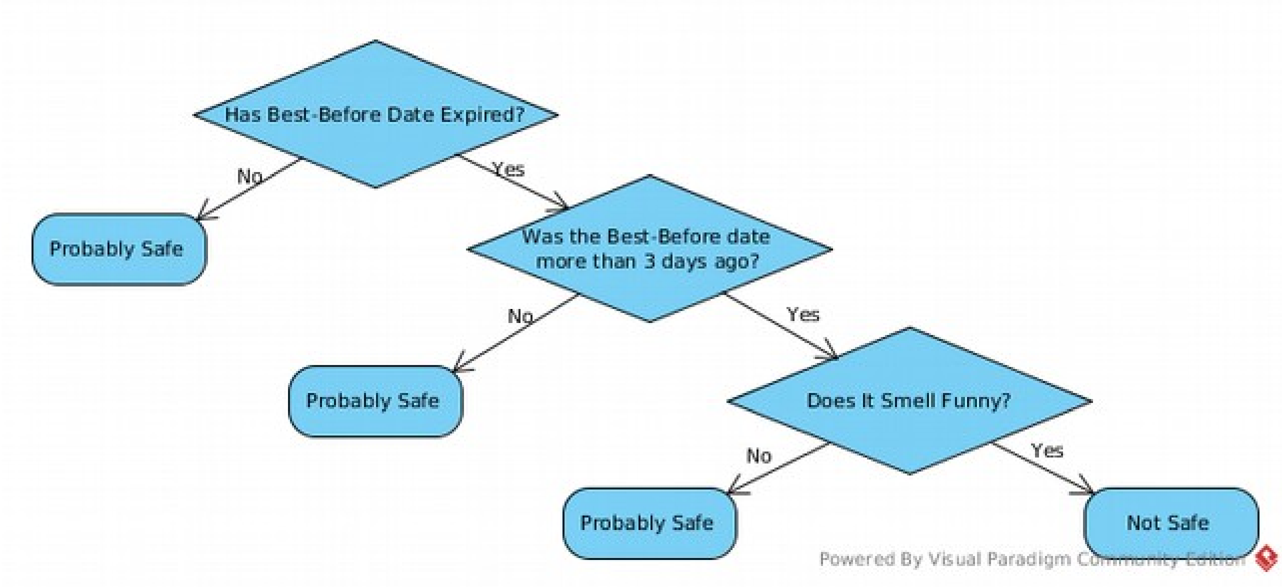

Has Best-Before Date Expired?
No
Yes
Probably Safe
Was the Best-Before date more than 3 days ago?
No
Yes
Probably Safe
Does It Smell Funny?
No
Yes
Probably Safe
Not Safe
Powered By Visual Paradigm Community Edition

Illustration 3.1: Example Decision Tree

- *a decision node,* that specifies some test to be carried out on a single attribute value, with one branch and subtree for each possible outcome of the test. (Quinlan, 1993, p. 5)

The decision tree moves though the nodes, splitting attribute values according to set criteria. If all outcomes on a particular branch match the test, then this is determined to be a leaf and all outcomes that match the chain of rules leading there are forecast to be the one at that leaf. Illustration 1 gives an example of a simple decision tree. The way in which J48 decides whether values belong to a class or whether they need more analysis is through entropy. Lower entropy values signify belonging to a single class whereas higher ones mean that the values belong to more than one class. (Witten et al., 2017)

Pruning is the act of removing unnecessary structure from the tree in order to simplify it without having a significant effect on the outcome. There are two strategies, pre-pruning and post-pruning, which as their names imply happen before or after classification. Pre-pruning works during the tree-building process and is advantageous in that the program does not need to do work that will later be discarded. Post-pruning can make decisions based on the full picture so for example, a node may seem insignificant by itself but when combined with another node can create a scenario where the whole is greater than the sum of its parts. (Witten et al., 2017)

J48 as implemented in Weka, has a number of configurable options, which alter the outcome. This paper will review some of those options, to give an outline of what is available. One option is Binary Splits, which can be set to True or False. The effect of this is that when True, a node can only have 2 options leading from it but when it is set to False a node can have more than two. When this option is set to True, the resulting tree will have many more leaves as the algorithm tries to account for all possible nodes. Another option is pruning. Using this option the program can reduce how deep the number of nodes will extend to without affecting the error rate too much. Pruning is useful because it helps to obtain smaller trees and to avoid over-fitting, which is where the model becomes too specific in trying to classify the data.

Advantages:

- Produces a model that is easily interpreted
- Works well with noise

Disadvantages:

- Small variations in data can produce a very different tree
- Needs a relatively large training set to perform well

3.3 Bayes Net Algorithm

The Bayes Net (Bayesian Network) is an algorithm based on probabilities. The basic formula for Naïve Bayes is: P(a/b) = (P(b/a) * P(b)) / P(a) where a is the outcome you are trying to predict and b is the data you already have. It works on the principle that the more information you have about something, the more you can be sure about its true value, as uncertainty decreases the probability distribution narrows. (Gelman et al., 2004) Naïve Bayes assumes that “every attribute (every leaf in the network) is independent from the rest of the attributes, given the state of the class variable (the root in the network)”

(Friedman et al., 1997). Bayesian Network does not make such assumptions about the data, all dependence needs to be modelled.

It is a probabilistic model that uses a direct cyclic graph with nodes representing attributes and these nodes connected by edges. (Ali et al., 2012). Illustration 2 gives an example of a cyclic graph. Here there are two scenarios that could cause a warm house, either the weather is warm or the heating has come on. The weather has a direct effect on whether the heating comes on. All three attributes can be either true or false.

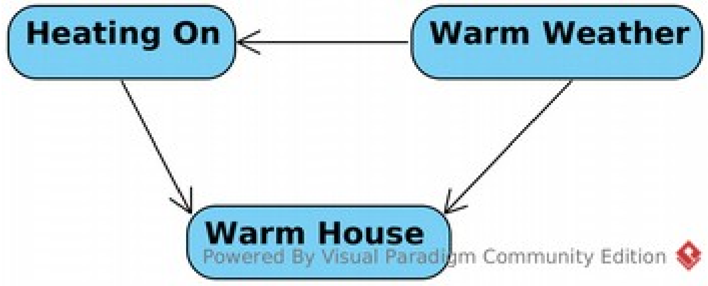


Illustration 3.2: Cyclic Graph

One type of classification (Naïve Bayes or Bayes Net) is not necessarily better than the other, they are just different ways of interpreting the data, though there is some research to suggest that Bayesian Networks may be worse when using data sets with more than 15 attributes (Friedman et al., 1997). The data set for this report has 15 attributes but there will be fewer once any redundant ones have been excluded, therefore this weakness should not be a problem.

Bayes Net has fewer options to change in the main dialogue compared to J48, a particular one of interest being the Search Algorithm. This allows you to select a number of different algorithms such as Genetic Search and Hill Climber. Once selected, these algorithms have options of their own to allow for further tweaking.

Advantages:

- Works well with datasets that are small and incomplete
- Gives fast responses (once the model is compiled)
- Combines knowledge from different sources
- The possibility of using Structural Learning

Disadvantages:

- Computationally intensive, although with modern computer systems including ones that utilise multiple GPUs this is less of a problem.
- Can make continuous variables slightly more discrete, which could be a problem if the relationship is linear (this can also be an advantage by improving efficiency)
- Information needs to be structured and organised
- Does not support feedback loops

  Uusitalo, 2007)

3.4 Algorithm Comparison

The two chosen classifiers have very different ways of predicting the outcome. J48 is based on decision trees whereas BayesNet is based on the probability of an outcome. J48 has many more options that can be changed compared to BayesNet. These options are explained in Chapter 5.1 – Modelling Results and Discussion. Which classifier can be determined to be the best depends on many factors, not least the dataset, its attribute types and integrity. Bayesian Networks work better with some datasets and J48 works better with others.

4 Data Exploration and Preparation

Customer ID

This attribute will be removed for reasons set out in chapter 2.2

Forename

This attribute will be removed for reasons set out in chapter 2.2

Surname

This attribute will be removed for reasons set out in chapter 2.2

Age

There is no noisy data for this attribute. The age range in this dataset is 17 to 89. Since it is possible to get a loan at the age of 17 with a guarantor (Citizens Advice, n.d.) and the National Counties building society and its Family building society offshoot will lend to people aged 89 (Collinson, 2015), no data will be removed or modified from this attribute.

Gender

The data for this attribute has more serious problems than appears at first. A few records have noise errors where they are recorded using different identifiers. These records could be amended according to the person’s name or removed. However, on closer inspection of the other records, it appears that many of them (perhaps half?) have the wrong gender for their name. Should these be amended or should the attribute be removed altogether? Is the gender relevant when deciding whether or not someone will pay back their loan? Two surveys performed by credit institutions suggest that it is (Herron, 2014), (CreditInfoCenter, 2016). While it seems far more likely that the gender attribute is wrong than the name attribute, through a typographical or selection error, there does not seem to be any criterion that explains how it has happened on this scale. For example, if all of the records had the wrong gender then a simple transcription error could be deduced. Because of the difficulties in accurately assessing what has happened to this attribute, only a small number of changes will be made. Records listed as Male will be converted to M, Female to F, those marked D, H, N, 1 and 0 will be given either M or F depending on the commonly associated gender for their name. All other records will be left as they are.

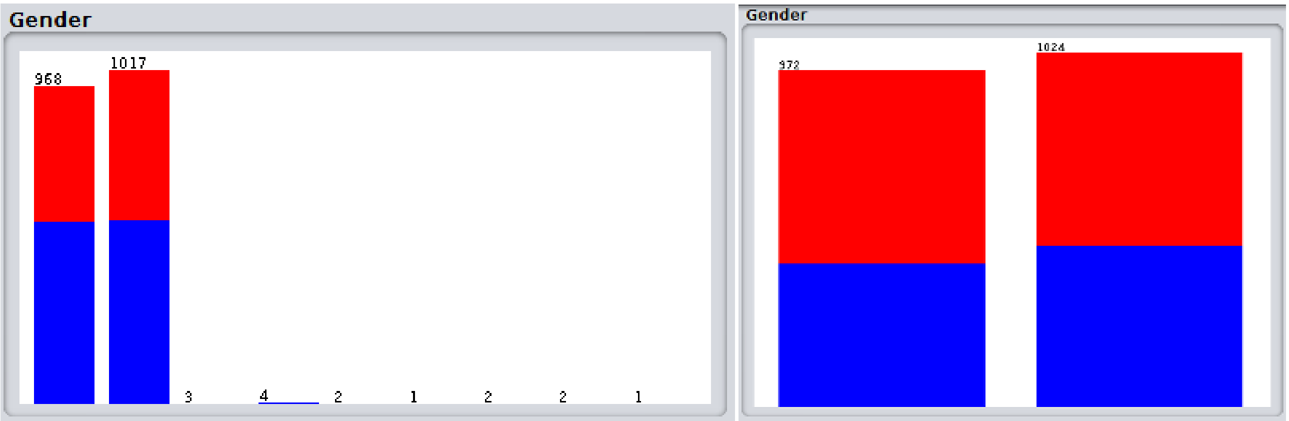


Illustration 4.1: Gender attribute before and after cleaning

Attribute value	Original total	Modified	Deleted	Final total
Male	3	3	0	0
Female	**4**	**4**	**0**	**0**
M	1017	0	0	1027
F	**968**	**0**	**0**	**973**
0	1	1	0	0
1	**2**	**2**	**0**	**0**
D	1	1	0	0
H	**2**	**2**	**0**	**0**
N	2	2	0	0
Total records	**2000**	**15**	**0**	**2000**

Years at Address

There are 4 records that show someone living at an address for longer than is physically possible. Each of these 4 entries has a 0 at the end of the value and when this 0 is removed, the value is less than the person's age. For this reason it seems safe to determine that these entries are a simple typing error and can be corrected. These 4 records will be corrected.

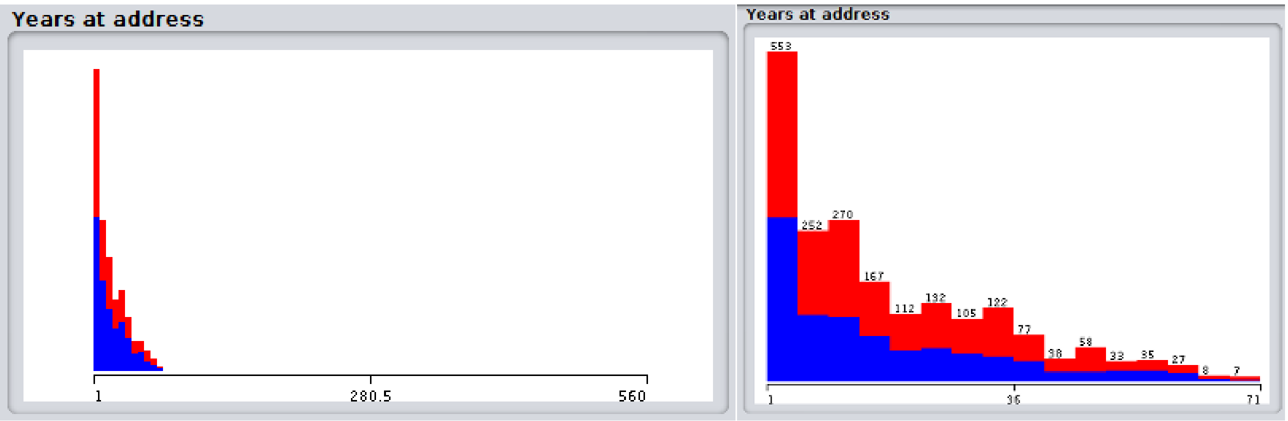


Illustration 4.2: Years at address attribute before and after cleaning

Attribute	Original total	Modified	Deleted	Final total
Years at address	2000	4	0	2000
Total records	**2000**	**4**	**0**	**2000**

Employment Status

There are no noise anomalies with this attribute.

Country

There are 6 records that have a country listed as one other than the UK. Most UK banks will not offer a standard loan to people who do not live in this country (Lloyds Bank, n.d.) (HSBC, n.d.) However it is not impossible to get something tailor-made for you (Frimpong, 2016) and all of the records that have a country which is not the UK also have a UK postcode, indicating some tie to this country. Therefore these entries will not be edited.

Current Debt

There are no outliers or noise for this attribute.

Postcode

This attribute will be removed because clustering will not be performed on this dataset.

Income

Two records for this attribute could be seen as outliers. They have values of 220,000 and 180,000; the next highest is 54,500 and the data is evenly spread from this point down. It is possible that both of these outliers are simple errors where an extra zero was added when entering the data but without further evidence to support this hypothesis, it cannot be concluded. The person who is listed with a salary of £220,000 is also listed with the wrong gender for their name and owning their home at the age of 17. While all of this is not impossible, it is highly unlikely. If this record is removed, that leaves one solitary record with a value that is far beyond the rest. Therefore it would be best to remove both of these records as not falling within normal bounds. Although they could be altered, they represent just 0.1% of the dataset.

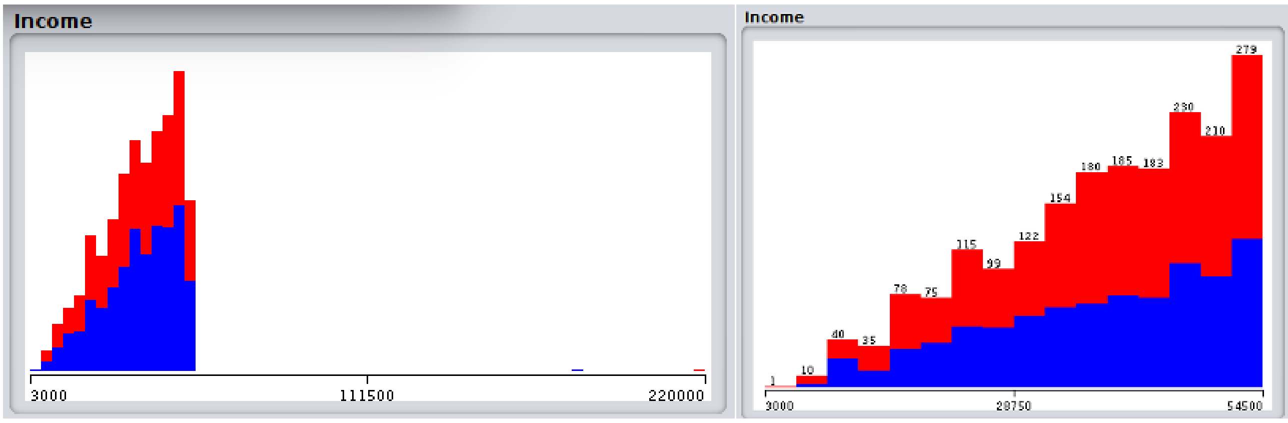


Illustration 4.3: Income attribute before and after cleaning

Attribute	Original total	Modified	Deleted	Final total
Income	2000	0	2	1998
Total records	**2000**	**0**	**2**	**1998**

Own Home

There are no data anomalies for this attribute.

CCJs

There are two records that could be viewed as outliers. One has 100 CCJs and the other has 10. While as noted earlier it could be possible to have 10, it is unlikely and this value is still more than 3 times that of the next highest. It is possible that these entries are typographical errors but there is no evidence to support this. Both of these records should be removed because they are a barrier to creating a good clean set of rules. Although they could be altered, they represent just 0.1% of the dataset.

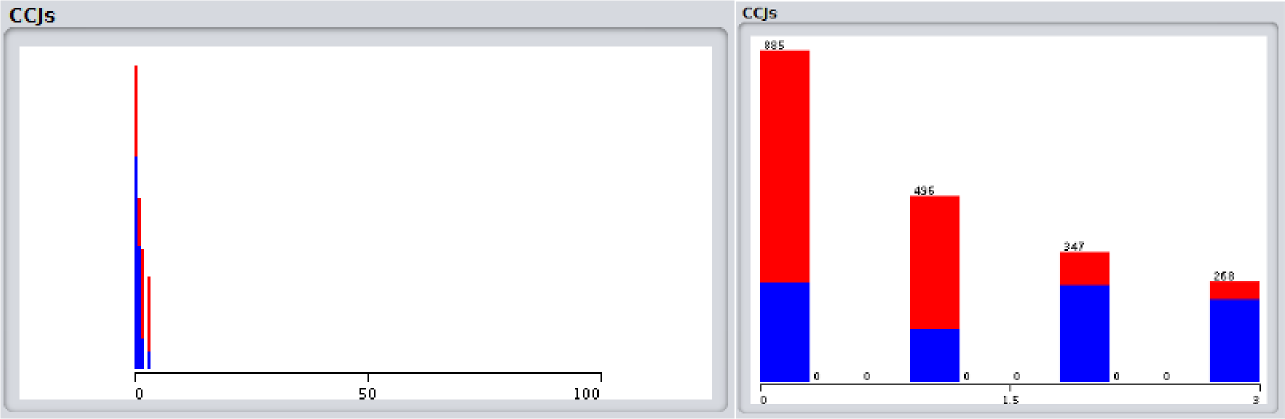


Illustration 4.4: CCJs attribute before and after cleaning

Attribute	Original total	Modified	Deleted	Final total
CCJs	1998	0	2	1996
Total records	**1998**	**0**	**2**	**1996**

Loan Amount

There does not appear to be any noise or outliers with this data. Although some of the entries are quite low (less than £100) they could refer to an overdraft or other means of credit.

Outcome

This is the class attribute. There is no noisy data for this attribute.

5 Modelling Results and Discussion

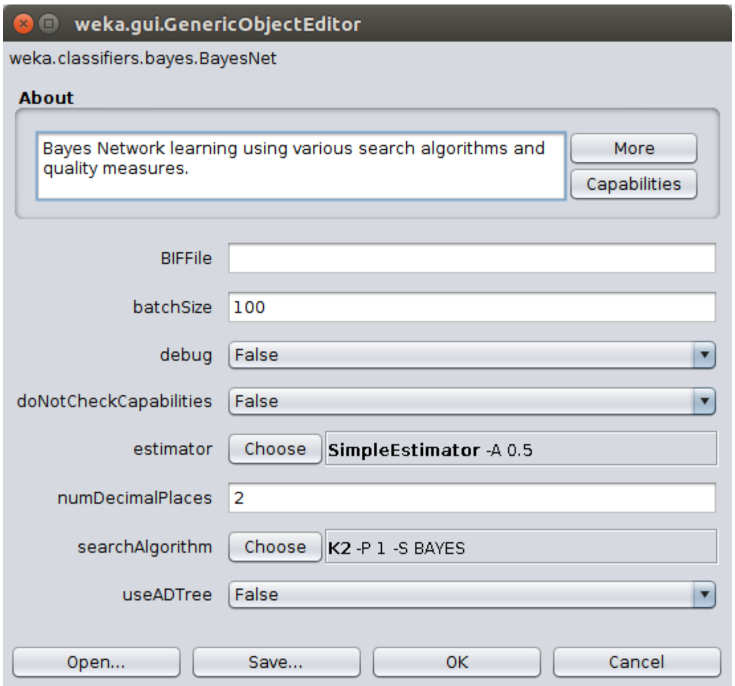
5.1 Experiment 1: Investigate Training and Testing Strategies

Aim:

To investigate whether 50:50 percentage split (50:50) or 10 Fold Cross Validation (10 Fold CV) gives the best classification accuracy when using BayesNet and J48.

Methodology:

The modified dataset will be classified using the selected algorithms, BayesNet and J48. Each algorithm will use it's default settings, as defined in Weka.



weka.classifiers.bayes.BayesNet

About

Bayes Network learning using various search algorithms and quality measures.

BIFFile

batchSize 100

debug False

doNotCheckCapabilities False

estimator **SimpleEstimator** -A 0.5

numDecimalPlaces 2

searchAlgorithm K2 -P 1 -S BAYES

useADTree False

Illustration 5.1: BayesNet settings dialogue

batchSize: preferred number of instances to process if batch prediction is being performed

estimator: Select estimator algorithm for finding the conditional probability tables of the Bayes Network

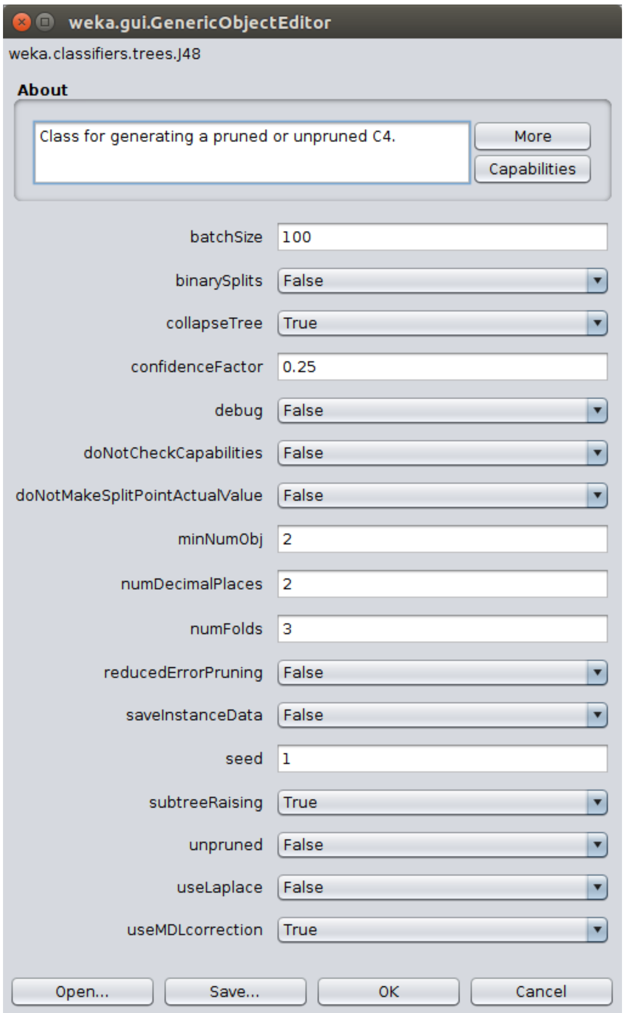
searchAlgorithm: Select method used for searching network structures

useADTree: The data structure for increasing speed on counts. Is memory intensive.

50:50 Training Strategy: The data is split into two parts, a training set and a test set. These can be of any proportions but usually not more unequal than 70:30. First a classifier is constructed from the training set and it is then used to try and predict the outcomes in the test set. The predictive accuracy of this test is $p=C/N$ where C is the number of correctly classified records and N is the total number of records tested.

10 Fold Cross Validation: The data is split into 10 equal, but random parts. 9 of them are used to train the algorithm and the tenth for testing. This is repeated 10 times so that all of the parts are tested on and the average taken of all results. The standard error for this test is stated as $\sqrt{p(1-p)/N}$ (Bramer, 2016).

Weka provides a number of ways to interpret the results but the ones most interesting to this experiment are the overall classification accuracy and the confusion matrix. The confusion matrix shows how many records were correctly classified and how many were misclassified for each outcome. From this, the percentage accuracy can be worked out by dividing the correct outcomes by the total number and multiplying by 100. An example of the Confusion Matrix can be seen at the end of Appendix 8.3 – Detailed Output.



weka.classifiers.trees.J48

About

Class for generating a pruned or unpruned C4.

batchSize	100
binarySplits	False
collapseTree	True
confidenceFactor	0.25
debug	False
doNotCheckCapabilities	False
doNotMakeSplitPointActualValue	False
minNumObj	2
numDecimalPlaces	2
numFolds	3
reducedErrorPruning	False
saveInstanceData	False
seed	1
subtreeRaising	True
unpruned	False
useLaplace	False
useMDLcorrection	True

Illustration 5.2: J48 settings dialogue

batchSize: Same as for BayesNet
binarySplits: Determines whether nodes should have two or more than two outcomes
collapseTree: Whether parts are removed that do not reduce training error
confidenceFactor: Used for pruning – smaller values incur more pruning
minNumObj: Minimum number of instances per leaf
numFolds: Determines the amount of data used for reduced-error pruning. One fold is used for pruning, the rest for growing the tree
reducedErrorPruning: Either J48 or C4.5 pruning
unpruned: Whether pruning is performed
useLaplace: Whether counts at leaves are pruned based on Laplace
useMDLCorrection: Whether MDL correction is used when finding splits on numeric attributes

Results:

Strategy	Technique	% Paid	% Default	Classification Accuracy %
50:50	BayesNet	85.04	66.90	77.36
10 Fold CV	BayesNet	85.02	67.20	77.15
50:50	J48	81.74	70.92	77.15
10 Fold CV	J48	83.50	69.24	77.20

Discussion:

One of the most interesting things about the results is that J48 manages to achieve almost the same overall result as with the unmodified dataset (Appendix 8.1), yet it does so in a very different way. For the unmodified dataset the algorithm takes into account only CCJs and Current debt for both 50:50 and 10 Fold Cross Validation, with a tree size of 5 and only 3 leaves. Yet after the dataset has been cleaned the tree size in both strategies has grown massively to 133 with 72 leaves. The accuracy for Paid has gone down slightly and Default has increased sightly, although Paid still has a better accuracy overall.

BayesNet has seen some improvement with 50:50 seeing almost 10 percentage points increase and 10 Fold Cross Validation increasing by more than 7 percentage points. BayesNet has also seen improvement in the accuracy of classifying each of Paid and Default, with Paid having the greater improvement in both strategies.

Examining the tree created by J48 shows that it makes exactly the same decisions for each strategy, so the only reason why 10 Fold Cross Validation scores higher is because it runs the test on all of the data whereas 50:50 Training Strategy only tests half of it.

After this experiment there is very little to choose between the two strategies. However, 10 Fold Cross Validation is slightly ahead when using J48 and 50:50 Training Strategy is slightly ahead when using BayesNet so these are the strategies that will be used in future experiments.

5.2 Experiment 2: Binary Splits vs non Binary Splits with J48

Aim:

To investigate whether Binary Splits will have a greater accuracy on predicting the class than non Binary Split.

Methodology:

The dataset will be analysed using the J48 algorithm with 10 Fold Cross Validation. There will be two experiments, with Binary Splits set to True and Binary Splits set to False. Only J48 will be used for this experiment because BayesNet is not a tree-based algorithm so does not have the Binary Splits option.

Binary Spits: Determines whether decisions at each node can have a maximum of two outcomes or if there can be more than two. With it set to True each node is restricted to 2 options only.

Results:

J48 – 10 Fold Cross Validation

Strategy	% Paid	% Default	Classification Accuracy %
Binary Splits True	85.11	70.83	78.81
Binary Splits False	83.50	69.24	77.20

Discussion:

Setting Binary Splits to True brings an improvement to all three areas of analysis, with each one gaining approximately 1.6 percentage points increase. The tree is also significantly different, having 44 leaves, down from 72 and an overall size of 87, down from 133. This makes the model not only more accurate but also simpler and easier to understand. CCJs are chosen as the root node, which means that Weka considers this attribute to be the most important.

The results of this experiment clearly show that going forward J48 should have the Binary Splits option set to True.

5.3 Experiment 3: Confidence Factor using J48

Aim:

To find the optimal value at which to set the Confidence Factor.

Methodology:

This experiment is investigating pruning. The Confidence Factor describes how aggressive the pruning should be, with smaller values meaning that the tree will incur more pruning.

This experiment will run a series of tests with the Confidence Factor set at different values at regular intervals. The default value, which was used for the previous tests is 0.25. This test will use values of 0.005, 0.05, 0.1, 0.25 and 0.5.

Results:

J48 – 10 Fold Cross Validation – Binary Splits True

Confidence Factor Value	% Paid	% Default	Tree Size	No of Leaves	Classification Accuracy %
0.005	86.46	66.63	11	6	77.71
0.05	85.83	72.53	35	18	79.96
0.1	86.10	72.30	41	21	80.01
0.25	85.11	70.83	44	87	78.81
0.5	81.70	68.45	243	122	75.85

Discussion:

Setting the Confidence Factor (CF) to 0.005 achieves the best %Paid result so far, it also achieves the worst %Default result so far leaving the overall accuracy no better than in Experiment 1. A value of 0.5, which means less pruning than the default value used in previous experiments, gives lower accuracies in every class thereby proving that pruning is beneficial. Using a CF of 0.1 achieved the best overall result in this experiment but to be certain that no further tweaking could be made, some more tests were run using values of 0.15 and 0.075.

J48 – 10 Fold Cross Validation – Binary Splits True

Confidence Factor Value	% Paid	% Default	Tree Size	No of Leaves	Classification Accuracy %
0.075	85.83	72.53	41	21	79.96
0.1	86.10	72.30	41	21	80.01
0.15	85.92	71.85	41	21	79.71

Small changes to the Confidence Factor did not have any effect on the size of the tree or the number of leaves but did have a small effect on the accuracy. The best result is still the one that uses a CF of 0.1, which produces a tree size of 41 with 21 leaves. This tree can be seen in Appendix 8.2: J48 Final Tree. The detailed output from Weka can be seen in Appendix 8.3.

One factor that is consistent throughout all of the experiments is that CCJs is always chosen as the root node. This means that Weka considers the CCJ attribute to be the most important when determining whether someone is likely to pay back their loan.

5.4 Experiment 4: Search Algorithms with Bayesian Networks

Aim:

BayesNet comes with 8 different algorithms for searching network structures. The aim is to try all of these at their default values to see which gives the best result.

Methodology:

Each algorithm will be used at their default values. The default algorithm for BayesNet as used in Weka is K2, which is the one used in the first experiment. As 50:50 Training Strategy performed best for BayesNet in the first experiment, this is the strategy that will be used for all subsequent experiments. A brief synopsis of some of the algorithms is provided below

Hill Climber: incrementally changes one of the elements for as long as it continues to improve the result. It is not certain to find the best possible outcome.

Tabu Search: Decides on a potential solution and then examines its immediate neighbours to see if a better solution can be found. It can sometimes opt for a sightly worse solution if there is no better one. If it has made a number of decisions to get to a point, it might get stuck when there could be a better solution by a different route. (Bouckaert et al., 2016)

Results:

BayesNet – 50:50 Training

Search Algorithm	% Paid	% Default	Classification Accuracy %
Genetic Search			Did not complete
Hill Climber	85.39	66.90	77.56
K2	85.04	66.90	77.36
LAGD Hill Climber	85.39	66.90	77.56
Repeated Hill Climber	85.39	66.90	77.56
Simulated Annealing	85.39	66.90	77.56
Tabu Search	85.39	66.90	77.56
TAN	85.04	66.90	77.36

Discussion:

As can be seen in the table above, changing the search algorithm made very little difference to the overall classification or to the classification of Paid and Default. This result was repeated when modifying the parameters. One reason for this could be because the search algorithms employed have weaknesses in that they can get stuck travelling along a certain path, where had they made a different decision earlier on that was not necessarily the best decision at the time, could lead to a more successful outcome later on.

6 Conclusion

The conclusion from these experiments is that for this dataset, J48 should be used with 10 Fold Cross Validation, Binary Splits set to True and a Confidence Factor of 0.1. This is the model that produces the most accurate results, both overall and for each outcome.

A tree based model is easier to understand and explain than one based on probabilities, which makes it easier to justify and apply the rules in a real working scenario.

7 References

Ali, A., Elfaki, M., Norhayati, D., 2012. Using Naïve Bayes and Bayesian Networkfor Prediction of Potential Problematic Cases in Tuberculosis. Int. J. Inform. Commun. Technol. IJ-ICT 1. doi:10.11591/ij-ict.v1i2.1424

Boslaugh, S., 2013. Statistics in a nutshell: [a deskop quick reference], 2. Aufl. ed. O'Reilly, Beijing.

Bouckaert, R.R., Frank, E., Hall, M., Kirkby, R., Reutemann, P., Seewald, A., Scuse, D., 2016. WEKA manual for version 3-8-1. Univ. WAIKATO.

Bramer, M.A., 2016. Principles of data mining, Third edition. ed, Undergraduate topics in computer science. Springer, London.

Citizens Advice, n.d. Young people – money and consumer rights - Citizens Advice [WWW Document]. Citiz. Advice. URL https://www.citizensadvice.org.uk/debt-and-money/young-people-and-money-advice/young-people-money-and-consumer-rights/ (accessed 2.21.17).

Collinson, P., 2015. Are you a mortgage misfit? The Guardian.

CreditInfoCenter, 2016. Do Women Have More Debt Than Men? [WWW Document]. CreditInfoCenter. URL http://www.creditinfocenter.com/debt/women-have-more-debt.shtml (accessed 2.22.17).

Friedman, N., Geiger, D., Goldszmidt, M., 1997. Bayesian network classifiers. Mach. Learn. 29, 131–163.

Frimpong, M., 2016. Can a non UK resident get a mortgage? - Enness Private [WWW Document]. Enness Priv. Clients. URL https://www.ennessprivate.co.uk/ask-expert/non-uk-resident-get-a-mortgage/ (accessed 2.22.17).

Gelman, A., Carlin, J.B., Stern, H.S., Rubin, D.B. (Eds.), 2004. Bayesian data analysis, 2. ed. ed, Texts in statistical science. Chapman & Hall, Boca Raton, Fla.

Herron, J., 2014. Men, Women And Debt: Does Gender Matter? [WWW Document]. Bankrate. URL http://www.bankrate.com/finance/debt/men-women-and-debt-does-gender-matter.aspx (accessed 2.22.17).

HSBC, n.d. Loans | Apply For a Loan | HSBC UK [WWW Document]. HSBC. URL https://www.hsbc.co.uk/1/2/loans (accessed 2.21.17).

Lloyds Bank, n.d. Lloyds Bank - UK Loans - How to Get a Personal Loan [WWW Document]. Lloyds Bank. URL https://www.lloydsbank.com/loans/help-and-guidance/how-to-get-a-loan.asp (accessed 2.22.17).

OnlineMortgageAdvisor, 2016. Mortgage with CCJs [WWW Document]. OnlineMortgageAdvisor.co.uk. URL https://www.onlinemortgageadvisor.co.uk/bad-credit-mortgages/mortgages-and-ccjs/ (accessed 2.21.17).

Quinlan, J.R., 1993. C4.5: Programs for Machine Learning. Morgan Kaufmann.

Uusitalo, L., 2007. Advantages and challenges of Bayesian networks in environmental modelling. Ecol. Model. 203, 312–318. doi:10.1016/j.ecolmodel.2006.11.033

Witten, I.H., Frank, E., Hall, M.A., Pal, C.J., 2017. Data mining: practical machine learning tools and techniques, Fourth edition. ed. Morgan

Kaufmann/Elsevier, Amsterdam Boston Heidelberg London New York Oxford Paris San Diego San Francisco Singapore Sydney Tokyo.

8 Appendix

8.1 Tables analysing the unmodified dataset

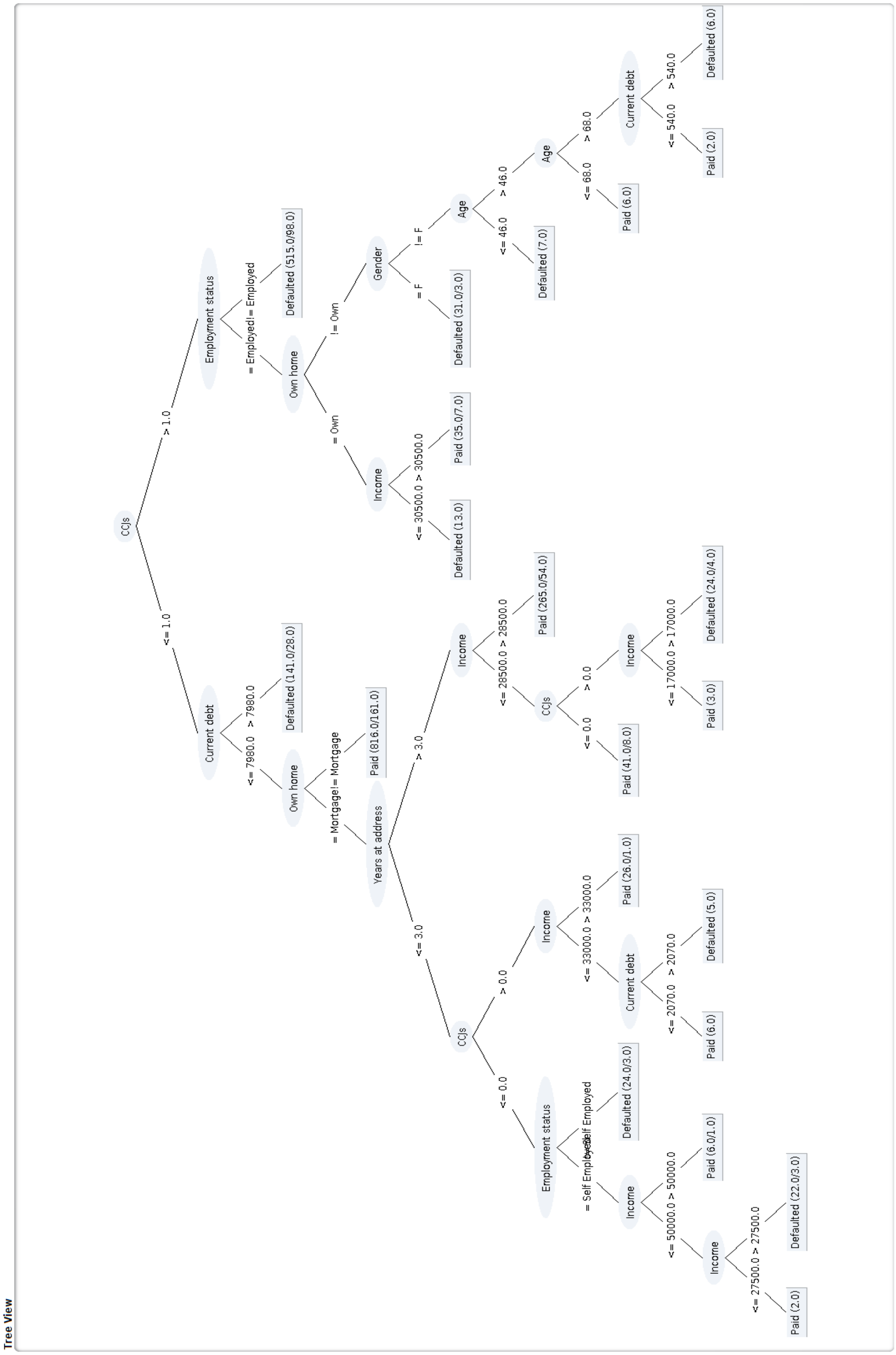
Table using 50 : 50 Training Strategy

Method	% Paid correct	% Defaulted correct	% Classification Accuracy
J48	86.35	65.60	77.30
Random Tree	83.69	15.83	54.10
Random Forest	83.51	41.51	65.20
Naïve Bayes	72.16	34.86	55.90
Bayes Net	72.52	61.93	67.90
Naïve Bayes Updatable	72.16	34.86	55.90
Zero R	100.00	0.00	56.40

Table using 10 Fold Cross Validation

Method	% Paid correct	% Defaulted correct	% Classification Accuracy
J48	84.97	67.12	77.10
Random Tree	87.48	18.71	57.15
Random Forest	85.60	30.73	61.40
Naïve Bayes	67.35	46.94	58.35
Bayes Net	73.44	65.20	69.80
Naïve Bayes Updatable	67.35	46.94	58.35
Zero R	100.00	0.00	55.90

8.2 J48 Final Tree using 10 Fold Cross Validation, Binary Splits – True and a Confidence Factor of 0.1



8.3 Detailed output of J48 Final Tree using 10 Fold Cross Validation, Binary Splits – True and a Confidence Factor of 0.1

```
=== Run information ===

Scheme:      weka.classifiers.trees.J48 -C 0.1 -B -M 2
Relation:     loandataset-final
Instances:    1996
Attributes:   11
              Age
              Gender
              Years at address
              Employment status
              Country
              Current debt
              Income
              Own home
              CCJs
              Loan amount
              Outcome
Test mode:    10-fold cross-validation

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

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

CCJs <= 1.0
|   Current debt <= 7980.0
|   |   Own home = Mortgage
|   |   |   Years at address <= 3.0
|   |   |   |   CCJs <= 0.0
|   |   |   |   |   Employment status = Self Employed
|   |   |   |   |   |   Income <= 50000.0
|   |   |   |   |   |   |   Income <= 27500.0: Paid (2.0)
|   |   |   |   |   |   |   Income > 27500.0: Defaulted (22.0/3.0)
|   |   |   |   |   |   Income > 50000.0: Paid (6.0/1.0)
|   |   |   |   |   Employment status != Self Employed: Defaulted (24.0/3.0)
|   |   |   |   CCJs > 0.0
|   |   |   |   |   Income <= 33000.0
|   |   |   |   |   |   Current debt <= 2070.0: Paid (6.0)
|   |   |   |   |   |   Current debt > 2070.0: Defaulted (5.0)
|   |   |   |   |   Income > 33000.0: Paid (26.0/1.0)
|   |   |   Years at address > 3.0
|   |   |   |   Income <= 28500.0
|   |   |   |   |   CCJs <= 0.0: Paid (41.0/8.0)
|   |   |   |   |   CCJs > 0.0
|   |   |   |   |   |   Income <= 17000.0: Paid (3.0)
|   |   |   |   |   |   Income > 17000.0: Defaulted (24.0/4.0)
|   |   |   |   Income > 28500.0: Paid (265.0/54.0)
|   |   Own home != Mortgage: Paid (816.0/161.0)
|   Current debt > 7980.0: Defaulted (141.0/28.0)
CCJs > 1.0
|   Employment status = Employed
|   |   Own home = Own
|   |   |   Income <= 30500.0: Defaulted (13.0)
|   |   |   Income > 30500.0: Paid (35.0/7.0)
|   |   Own home != Own
|   |   |   Gender = F: Defaulted (31.0/3.0)
|   |   |   Gender != F
|   |   |   |   Age <= 46.0: Defaulted (7.0)
|   |   |   |   Age > 46.0
|   |   |   |   |   Age <= 68.0: Paid (6.0)
|   |   |   |   |   Age > 68.0
|   |   |   |   |   |   Current debt <= 540.0: Paid (2.0)
|   |   |   |   |   |   Current debt > 540.0: Defaulted (6.0)
|   Employment status != Employed: Defaulted (515.0/98.0)

Number of Leaves  :       21

Size of the tree :    41


Time taken to build model: 0.02 seconds

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

Correctly Classified Instances        1597               80.01   %
```

```
Incorrectly Classified Instances       399               19.99   %
Kappa statistic                          0.5903
Mean absolute error                      0.3142
Root mean squared error                  0.4026
Relative absolute error                 63.7081 %
Root relative squared error             81.0745 %
Total Number of Instances             1996

=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC
Area  Class
                 0.723    0.139    0.804      0.723    0.762      0.593    0.771     0.710
Defaulted
                 0.861    0.277    0.797      0.861    0.828      0.593    0.771     0.755
Paid
Weighted Avg.    0.800    0.216    0.800      0.800    0.799      0.593    0.771     0.735

=== Confusion Matrix ===

   a   b   <-- classified as
 637 244 |   a = Defaulted
 155 960 |   b = Paid
```