Coursework Assessment 2

SET 09120 Data analytics 2019/20

Murphy, Brogan 40340890

2019

# Introduction

The aim of this coursework is to first clean the dataset provided and transform it where necessary using openrefine. Secondly, we must use Weka, a data mining software to perform such algorithms on our dataset such as classification, regression, association and clustering to explore our data and document the findings within the report.

# Data Preparation

Data preparation is where we take data and clean out any errors, restructure and transform data in order to provide a dataset that upon analysis will provide meaningful, accurate results. The tool I used to clean the data set provided was openrefine.

## Data Cleaning

The dataset provided was littered with many errors that would’ve compromised analysis of this dataset, so I cleaned it with openrefine using tools such text and numeric facets which allowed me to clean data with ease. I firstly gave each column its relevant header. Below is the changes I made that became my first version of the dataset

**Changes Made to CleanCourseworkV1**

|  |  |
| --- | --- |
| **Row** | **Changes Made** |
| Case\_No | N/A |
| Checking\_Status | Removal of ‘’ |
| Credit\_History | Removal of ‘’ |
| Purpose | Removal of ‘’, fixing of spelling errors, acknowledging absence of vacation value |
| Credit\_Amount | Recognising outliers and changing them:  Case No 432: 111328000 to 11328: extra 1 at start and zero’s at end  Case No 660: 63610000 to 6361: too many zero’s at end  Case No 560: 19280000 to 1928: too many zero’s at end  Case No 595: 13580000 to 1358: too many zero’s at end  Case No 648: 13860000 to 1386: too many zero’s at end  Case No 444: 7190000 to 7190: too many zero’s at end  Case No 514: 5850000 to 585: too many zero’s at end, also only wanted finance for radio/tv so 585 seems the likely amount  Case No 452: 5180000 to 518: too many zero’s at end, also only wanted finance for radio/tv so 518 seems the likely amount |
| Saving\_Status | Removal of ‘’ |
| Employment | Removal of ‘’ |
| Personal\_Status | Removal of ‘’ |
| Age | Identifying outliers and changing them:  Case No 68: 222 to 22 – extra 2  Case No 80: 222 to 22 – extra 2  Case No 174: 333 to 33 – extra 3  Case No 192: -34 to 34 – minus by accident  Case no 233: -35 to 35 – minus by accident  Case No 280: -29 to 29 – minus by accident  Case No 26: 6 to 26 – 26 - possibly missed ‘2’ by accident as it could be a loan for a new house/flat, hence needing furniture, plus employment for between 1 and 4 years  Case No 54: 1 – 18 – possibly missed ‘8’ by accident, could be a new driver looking to buy car  Case No 305: 0.44 – 44 – accidental ‘0.’  Case No 333: 0.24 – 24 - accidental ‘0.’  Case No 448: 0.35 – 35 - accidental ‘0.’ |
| Job | Removal of ‘’, changed 2 rows of ‘yes’ to ‘unskilled resident’ |
| Class | No changes |

## Data Conversion

To create a second version of the dataset, I used the already cleaned dataset to produce an all-nominal dataset. Changes made are: All numerical variables changed to nominal. For credit amount, anything that was under 4750 was returned as “Lowest credit amount”, anything between 4750 and 9500 is returned “Lower credit value”, anything between 9500 and 14250 is returned as “Medium credit amount” and anything over 14250 is returned as “High credit amount”.

For Age, anything equal to 18, between 18 and 39 and equal to 39 was returned as “Young adult”. Anything equal to 40, between 40 and 65 and equal to 65 was returned as “Middle aged adult”. Anything over 65 was returned as “Older adult”.

# Data Analytics

Data mining can be used to analyse our new clean dataset in order to draw conclusions, create rules and discover interesting patterns. This can be achieved through the use of 4 algorithms; Classification, Regression, Association and Clustering.

## Classification

Classification is used for accurately predicting a target class for each case of data. In our case, it will identify which people are likely to be granted the loan. I used the J48 algorithm that is part of the weka software which can produce decision trees to predict the class depending based on the use of other variables. I also used pruning as it downsizes the tree to make it easier to analyse and therefore produce more accurate information. Within weka, when I was using classification I changed the object editor’ confidenceFactor setting to 0.5 and minNumObj to 15, which produced a tree with 76.9% correctly classified instances:

Correctly Classified Instances 769 76.9 %

Incorrectly Classified Instances 231 23.1 %

From this, I was able to produce 6 rules about the data. These are:

**Rule 1 –** IF Checking\_Status = no checking THEN good (394/46)

This shows that there are 394 instances of people who have no current account with the bank which are granted the loan, however 46 of these instances are incorrectly classified under this.

**Rule 2 –** IF Checking\_Status = 0<=x<200 AND Credit\_Amount = Lowest credit value AND Saving\_Status = No known savings THEN good (31/4)

This shows that if the people’s current account who have a balance between 0 and 200, they’ve chosen 4750 or below as their amount and they have no savings, they will get the loan. We can see there is only 4 incorrect instances which shows accuracy. This can also mean the bank is possibly lenient when it comes to granting a credit amount classed as ‘lowest credit amount’.

**Rule 3 –** IF Checking\_Status = 0<=x<200 AND Credit\_Amount = High credit amount THEN bad (7)

We can see that of the people’s current account with the bank has a balance between 0 and 200 and they choose the highest credit amount which is 14250 or over, they will not be granted the loan. There are no incorrect instances with this rule which shows that the bank must be strict on who they grant the high credit amount to.

**Rule 4 –** IF Checking\_Status = <0 AND Credit\_History = delayed previously THEN bad (12/3)

This shows that if the person has less than 0 in their current account and have been delayed in the past for applying for loans, they will not be granted the loan on this occasion. There are 3 out of 12 incorrect classifications, meaning that the machine has a 25% inaccuracy on this rule. This does show that overall the bank do not give out loans to those who have bad credit history and a poor bank balance.

**Rule 5 –** IF Checking\_Status = 0<=X=<200 AND Credit\_Amount = Lowest credit value AND Saving\_Status = <100 AND Purpose = radio/tv THEN good (43/9)

We can see that people who have a balance of between 0 and 200, they’ve chosen Lowest credit amount, they have less than 100 saved and the purpose of the loan is for radio/tv, the loan is granted. There is 9 incorrect instances out of 43 which is fairly accurate and shows the bank has drawn the conclusion that the lowest credit value is reasonable for a new tv or radio and will grant the loan.

**Rule 6 –** IF Checking\_Status = <0 AND Credit\_History = existing paid AND Saving\_Status = <100 AND Purpose = new car THEN bad (31/9)

This shows that those who have a balance of less than 0 in their bank account, a credit history of all previous loans paid off, savings of less than 100 and the purpose of the loan is for a new car, the loan will not be granted. Out of 31 instances, 9 have been incorrectly classified which could mean the machine learning is slightly inaccurate, but still accurate enough to show that the bank won’t grant loans for those who want to buy new cars if they have little savings and no money in their current accounts.

## Regression

Not actually used as other three algorithms seemed more accurate and efficient.

## Association

For association, I used the apriori algorithm. Apriori is used for mining frequent and produce meaningful rules based on the dataset, e.g. people who request a loan in the lowest credit value are very likely to be granted the loan. I kept the metric for confidence as a minimum of 0.9 as I wanted the rules produced to be accurate

**Rule 1** - **Checking\_Status**=no checking **Credit\_History**=critical/other existing credit **Credit\_Amount**=Lowest credit value 121 ==> **Class**=good 116 <conf:(0.96)> lift:(1.37) lev:(0.03) [31] conv:(6.05)

This was my most ‘confident’ rule which means it is the most accurate association rule, with an accuracy of 96% meaning that when these attributes occur the rule is true 96% of the time. It shows that people with no current account with the bank, a credit history of existing unpaid debts and a choice of an amount less than or equal to 4750, they will be granted the loan.

**Rule 2** - **Checking\_Status**=no checking **Purpose**=radio/tv **Credit\_Amount**=Lowest credit value 114 ==> **Class**=good 108 <conf:(0.95)> lift:(1.35) lev:(0.03) [28] conv:(4.89)

This rule has a confidence of 95%, meaning it is an accurate association rule. We can see that people with no current account with the bank, their reason for applying for the loan being for a radio/tv, asking for a loan less than or equal to 4750 being granted the loan.

**Rule 3 - Checking\_Status**=no checking **Credit\_Amount**=Lowest credit value **Personal\_Status**=male single **Job**=skilled 120 ==> **Class**=good 113 <conf:(0.94)> lift:(1.35) lev:(0.03)

Rule has a confidence of 94% meaning it is an accurate association rule. It shows that people with no current account with the bank, choosing a credit amount of 4750 or less, being a single male who has a skilled job were very likely to be granted the loan.

**Rule 4 - Checking\_Status**=no checking **Employment**=>=7 115 ==> **Class**=good 107 <conf:(0.93)> lift:(1.33) lev:(0.03) [26] conv:(3.83)

This rule has a confidence of 93% making it an accurate association rule. It shows that those with no current account with the bank and who have been employed for equal to or more than 7 years are very likely to be granted the loan.

**Rule 5 - Checking\_Status**=no checking **Personal\_Status**=male single **Job**=skilled 150 ==> Class=good 139 <conf:(0.93)> lift:(1.32) lev:(0.03) [34] conv:(3.75)

This rule has a confidence of 93% making it an accurate association rule. We can see that single men who don’t have a checking account with the bank who work in a skilled job had a great chance of being granted the loan.

**Rule 6** - **Checking\_Status**=no checking **Credit\_Amount**=Lowest credit value **Job**=skilled 219 ==> **Class**=good 202 <conf:(0.92)> lift:(1.32) lev:(0.05) [48] conv:(3.65)

This rule has a confidence of 93% making it an accurate association rule. In our last rule we can see how yet again the people who don’t have a checking account with the bank, who ask for an amount less than or equal to 4750 and work a skilled job have a high chance of being accepted for the loan.

## Clustering

Clustering is where attributes of the dataset are grouped together in clusters of similar objects. In weka I used the simplekmeans algorithm and set the amount of clusters to be equal to 6 as I must produce 6 rules.

Clustered Instances

0 165 ( 17%)

1 83 ( 8%)

2 135 ( 14%)

3 184 ( 18%)

4 187 ( 19%)

5 246 ( 25%)

Clustered instances shows us how many times each cluster occurs within the thousand entries, we can see that cluster 5 occurs the most times.

These clusters are as follows:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Cluster 0 | Cluster 1 | Cluster 2 | Cluster 3 | Cluster 4 | Cluster 5 |
| Checking\_Status | no checking | <0 | no checking | No checking | <0 | 0<=X<200 |
| Credit\_History | critical/other existing credit | critical/other existing credit | existing paid | existing paid | existing paid | existing paid |
| Purpose | new car | used car | radio/tv | Radio/tv | new car | radio/tv |
| Credit\_Amount | 2882 | 4947 | 2311 | 3323 | 4502 | 2547 |
| Saving\_Status | <100 | <100 | <100 | No known savings | <100 | <100 |
| Employment | 1<=X<4 | unemployed | >=7 | >=7 | >=7 | 1<=X<4 |
| Personal\_Status | female div/dep/mar | male single | male single | male single | male single | female div/dep/mar |
| Age | 34 | 44 | 37 | 41 | 36 | 28 |
| Job | skilled | high qualif/self emp/mgmt | Unskilled resident | skilled | skilled | skilled |
| Class | good | good | good | good | bad | good |

We can see that out of 6 clusters, cluster 4 is the only one where the loan won’t be granted. We can see that if they had less than 0 in their checking account, they’d paid off all existing loans, they wanted a loan for a new car, the mean credit amount was 4502, they have less than 100 saved up, they’d been employed for seven or more years, they were a single male with a mean age of 36 and worked a skilled job they were not likely to be granted the loan.

We can see that job type didn’t play a part in the bank granting loans as skilled, high qualif/self emp/mgmt and unskilled all had instances of class being good, meaning they got the loan despite the job type.

We can also see that a persons finances (saving status and checking status) do not possibly have an effect on whether or not the loan is granted as we can see that the highest amount in the checking account in the cluster is between 0 and 200 and savings were either not known or less than 100, these are both low amounts; showing that the bank probably does not consider finances in making a decision to grant a loan.

# Conclusion

To conclude, the 3 algorithms I used managed to produce accurate and meaningful rules to a certain extent. Classification and the use of the J48 algorithm managed to produce just over three quarters of correctly classified instances, meaning we could create rules that had high coverage and accuracy if we choose rules with a good amount of correctly classified instances e.g. (394/46) meant that only 11% were incorrectly classified by the machine learning.

Association and the use the apriori algorithm were efficient as they produced ready-made rules that were based on confidence – the amount of times that particular rule occurred. I would say this is the most efficient algorithm at producing accurate rules as the higher the confidence is, the more accurate that rule is which is a good indicator for a good rule.

Clustering produced rules that somewhat did not make sense and could therefore be said that it is a bad type of analysis, however the dataset wasn’t the most rational which could contribute to strange rules being produced.