CS 5402: HW 7 Report

We analyzed the credit.csv dataset with various supervised and unsupervised methods. However, each method that was used overall was ineffective at reliably classifying instances from the dataset.

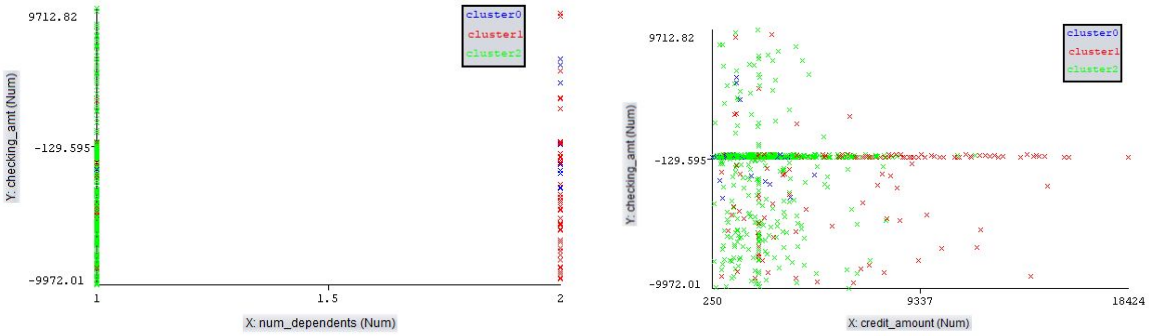
We believe the ineffectiveness of the models was primarily due to the size of the dataset being limited. Comparatively, there were many relevant features in the dataset with respect to the amount of instances. Due to this there was not enough data to make meaningful predictions using certain attributes. This problem was further compounded as cross-fold validation requires that we train on subsets of the dataset, further restricting the amount of data that models could learn from.

That being said, some models still outperformed others by considerable margins. In particular, the model that we determined to classify instances the best was a Gradient Boosting Machine (GBM) with the following hyper-parameters: `n_estimators=100, max_features=10, max_depth=10, subsample=0.5`

This model was selected because of a combination of different metrics used to compare classifiers. We found that the GBM model performed at the same level or slightly higher than the other models in terms of raw accuracy. However, the model's kappa statistic is significantly higher than that of the other models considered in this analysis with a value of 0.0981. Admittedly this is far from ideal, however, the other classifiers had near-zero values, meaning they were comparable to random classifiers. Additionally, looking at the confusion matrix of other classifiers, many unproportionally classified instances as GOOD. While the GBM still suffers from this problem, it is in a less extreme sense. Finally, the ROC AUC value was higher than the other models, which further solidified our decision in this model.

Unsupervised methods were also run on the dataset to help uncover meaningful relations. The Apriori algorithm was run on a subset of the dataset which resulted in a few interesting relationships. We noticed that existing_credits = 1 and housing=OWN were good predictors for credit_history = GOOD (a modification to the dataset which indicates that the credit has been paid) which intuitively makes sense, one would expect that someone owning a house with existing credit (but not too much!) to have a pretty good credit history. We also noticed that GOOD credit_history and no other_parties would imply no other_payment_plans which also seems like a sensical relationship, a lack of ongoing payments can indicate a good credit history and vice versa.

Another unsupervised method that we used was EM clustering which provided another interesting relation in the data. On the left-most figure, we see that instances were clustered on the number of dependents that they had. Then, in the right-most figure, we see the red cluster a higher credit amount. Therefore, from these figures, we can conclude that individuals that had 2 dependents (red) often have significantly more credit available to them than those who have 1 dependent (green).



Procedure:

Data Preprocessing/Cleaning:

To clean the data we first started by addressing the missing values. We noted certain rows which contained mostly missing values (more than 50%) and deleted those. We also ran through and assigned missing values in nominal attributes a new category. For numeric attributes we replaced most of the missing values with the median value so as to minimize skewing the existing data.

To trim down the number of attributes we ran spearman and chi squared tests to determine any dependent attributes we could trim. We also removed a couple of attributes that added far too much noise to the data and prevented little reason to the data.

Another pair of attributes that we dropped was state and location, as there were many different values that these nominal attributes could take, but not enough instances to fill each combination. The result was most classifiers were overfitting by using one or more of these attributes.

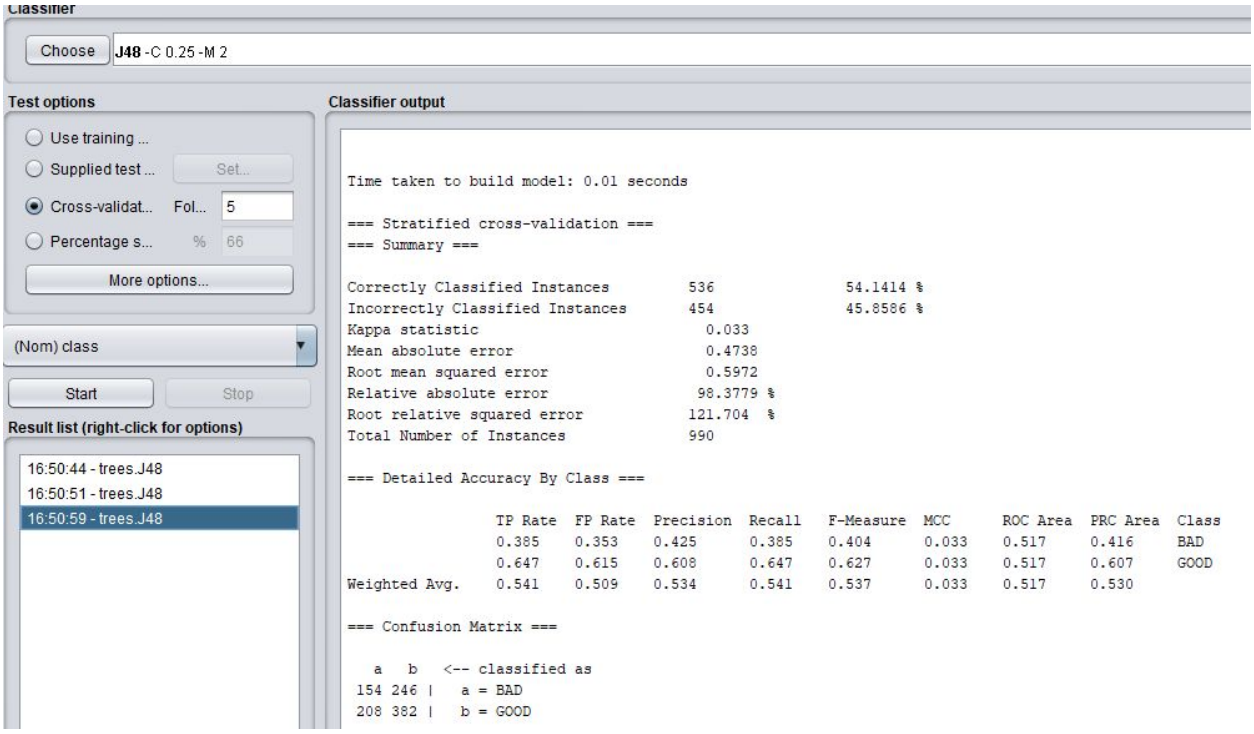
Finally we added a couple of new features to the dataset based on aggregates of existing attributes to help preprocess the data and give the classifier a better contextual understanding of the data.

We also added an optional flag to bin data. This resulted in a binned version and an unbinned version of our cleaned dataset. If possible, we tried each dataset on each method to determine which one gave better results.

An in depth description of our cleaning is documented in our script here. The rest of the source code/images and models described in the following pages is all documented and stored in our **github repository** available here.

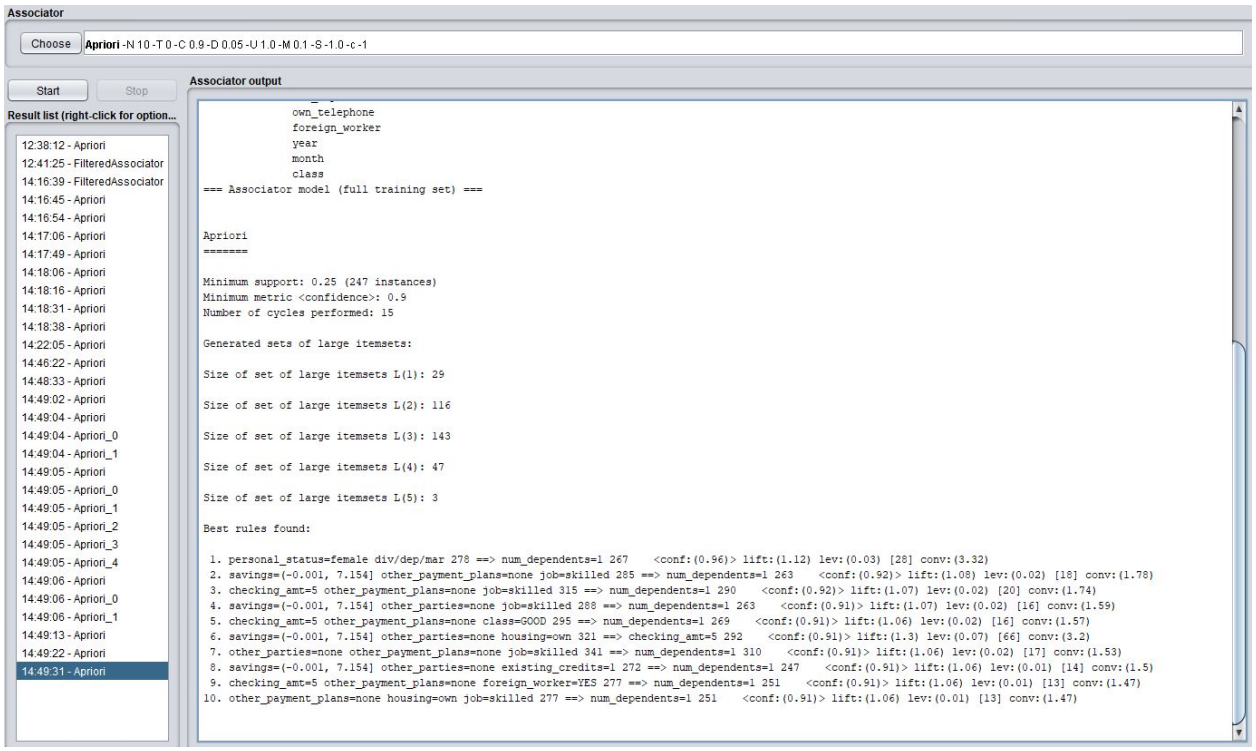
Decision Tree:

It was decided that we would use the J48 decision tree method built into Weka. We used 5 fold cross validation as that was deemed to have the best accuracy (54.14%) as well as better True Positive (0.647 for GOOD, 0.385 for BAD) and False Positive (0.615 for GOOD, 0.353 for BAD) rates, which is rather detrimental to serving as a usable classifier. We did notice that the “state” attribute was causing lots of error in the classifier and had to drop it. We ran the Weka decision tree through the scripts from HW#5 to determine whether any conditions could be dropped, but it seems like Weka did a good enough job of dropping all unnecessary leaves, and we could find no improvements. The source code listing for that python program is provided in the scripts folder of the provided github repo and is labeled decision_tree_pruner.py. The decision tree generated is listed in Appendix A.



Association Rules:

The chosen association rule algorithm was Apriori in Weka due to the Python implementation's lack of configuration options. The algorithm was run on the binned dataset. We then took the rules and ran them through our pruner in python (association_rule_pruner.py in the repo) but found no rules to prune. At any decent confidence level (0.75 or above) most of the rules seemed to focus on repeating attribute values and it was hard to derive meaningful relationships.

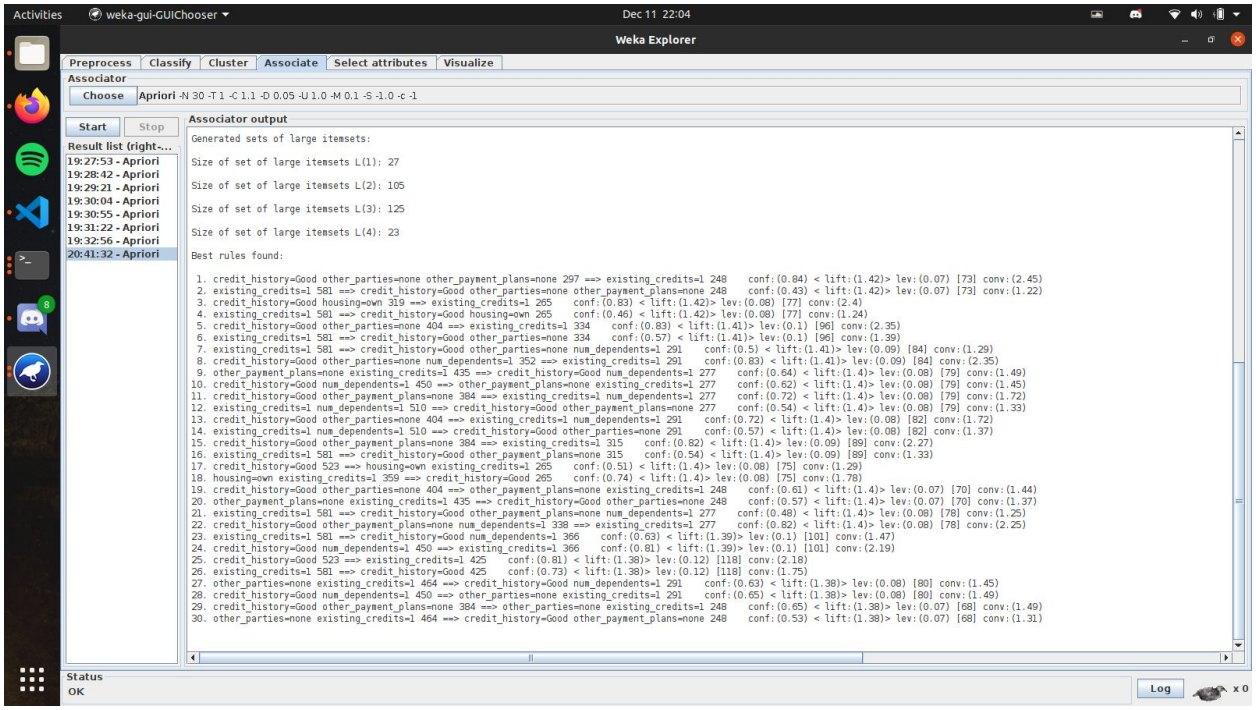



```
 1. other_payment_plans=none 736 ==> savings=0 673    <conf:(0.91)> lift:(1.01) lev:(0) [3]
conv:(1.05)
 2. num_dependents=1 848 ==> savings=0 770    <conf:(0.91)> lift:(1) lev:(-0) [0] conv:(0.98)
 3. other_parties=none 792 ==> savings=0 711    <conf:(0.9)> lift:(0.99) lev:(-0.01) [-9]
conv:(0.88)
 4. other_parties=none num_dependents=1 680 ==> savings=0 609    <conf:(0.9)> lift:(0.99)
lev:(-0.01) [-9] conv:(0.86)
 5. other_payment_plans=none 736 ==> num_dependents=1 642    <conf:(0.87)> lift:(1.02)
lev:(0.01) [11] conv:(1.11)
 6. other_parties=none 792 ==> num_dependents=1 680    <conf:(0.86)> lift:(1) lev:(0) [1]
conv:(1.01)
 7. savings=0 other_parties=none 711 ==> num_dependents=1 609    <conf:(0.86)> lift:(1)
lev:(-0) [0] conv:(0.99)
 8. savings=0 900 ==> num_dependents=1 770    <conf:(0.86)> lift:(1) lev:(-0) [0] conv:(0.99)
 9. num_dependents=1 848 ==> other_parties=none 680    <conf:(0.8)> lift:(1) lev:(0) [1]
conv:(1)
10. savings=0 num_dependents=1 770 ==> other_parties=none 609    <conf:(0.79)> lift:(0.99)
lev:(-0.01) [-7] conv:(0.95)
```

We then changed the algorithm to sort by lift and then found rules that seemed more interesting and were stronger indicators of possible relationships. Those results are listed below.

```
Best rules found:

 1. credit_history=Good other_parties=none other_payment_plans=none 297 ==> existing_credits=1
248    conf:(0.84) < lift:(1.42)> lev:(0.07) [73] conv:(2.45)
 2. existing_credits=1 581 ==> credit_history=Good other_parties=none other_payment_plans=none
248    conf:(0.43) < lift:(1.42)> lev:(0.07) [73] conv:(1.22)
 3. credit_history=Good housing=own 319 ==> existing_credits=1 265       conf:(0.83) <
lift:(1.42)> lev:(0.08) [77] conv:(2.4)
 4. existing_credits=1 581 ==> credit_history=Good housing=own 265       conf:(0.46) <
lift:(1.42)> lev:(0.08) [77] conv:(1.24)
 5. credit_history=Good other_parties=none 404 ==> existing_credits=1 334       conf:(0.83) <
lift:(1.41)> lev:(0.1) [96] conv:(2.35)
 6. existing_credits=1 581 ==> credit_history=Good other_parties=none 334       conf:(0.57) <
lift:(1.41)> lev:(0.1) [96] conv:(1.39)
 7. existing_credits=1 581 ==> credit_history=Good other_parties=none num_dependents=1 291
conf:(0.5) < lift:(1.41)> lev:(0.09) [84] conv:(1.29)
 8. credit_history=Good other_parties=none num_dependents=1 352 ==> existing_credits=1 291
conf:(0.83) < lift:(1.41)> lev:(0.09) [84] conv:(2.35)
 9. other_payment_plans=none existing_credits=1 435 ==> credit_history=Good num_dependents=1
277    conf:(0.64) < lift:(1.4)> lev:(0.08) [79] conv:(1.49)
10. credit_history=Good num_dependents=1 450 ==> other_payment_plans=none existing_credits=1
277    conf:(0.62) < lift:(1.4)> lev:(0.08) [79] conv:(1.45)
11. credit_history=Good other_payment_plans=none 384 ==> existing_credits=1 num_dependents=1
277    conf:(0.72) < lift:(1.4)> lev:(0.08) [79] conv:(1.72)
12. existing_credits=1 num_dependents=1 510 ==> credit_history=Good other_payment_plans=none
277    conf:(0.54) < lift:(1.4)> lev:(0.08) [79] conv:(1.33)
13. credit_history=Good other_parties=none 404 ==> existing_credits=1 num_dependents=1 291
conf:(0.72) < lift:(1.4)> lev:(0.08) [82] conv:(1.72)
14. existing_credits=1 num_dependents=1 510 ==> credit_history=Good other_parties=none 291
conf:(0.57) < lift:(1.4)> lev:(0.08) [82] conv:(1.37)
15. credit_history=Good other_payment_plans=none 384 ==> existing_credits=1 315 conf:(0.82) <
lift:(1.4)> lev:(0.09) [89] conv:(2.27)
16. existing_credits=1 581 ==> credit_history=Good other_payment_plans=none 315 conf:(0.54) <
lift:(1.4)> lev:(0.09) [89] conv:(1.33)
17. credit_history=Good 523 ==> housing=own existing_credits=1 265       conf:(0.51) <
lift:(1.4)> lev:(0.08) [75] conv:(1.29)
18. housing=own existing_credits=1 359 ==> credit_history=Good 265       conf:(0.74) <
lift:(1.4)> lev:(0.08) [75] conv:(1.78)
19. credit_history=Good other_parties=none 404 ==> other_payment_plans=none existing_credits=1
248    conf:(0.61) < lift:(1.4)> lev:(0.07) [70] conv:(1.44)
20. other_payment_plans=none existing_credits=1 435 ==> credit_history=Good other_parties=none
248    conf:(0.57) < lift:(1.4)> lev:(0.07) [70] conv:(1.37)
21. existing_credits=1 581 ==> credit_history=Good other_payment_plans=none num_dependents=1
277    conf:(0.48) < lift:(1.4)> lev:(0.08) [78] conv:(1.25)
22. credit_history=Good other_payment_plans=none num_dependents=1 338 ==> existing_credits=1
277    conf:(0.82) < lift:(1.4)> lev:(0.08) [78] conv:(2.25)
23. existing_credits=1 581 ==> credit_history=Good num_dependents=1 366  conf:(0.63) <
lift:(1.39)> lev:(0.1) [101] conv:(1.47)
24. credit_history=Good num_dependents=1 450 ==> existing_credits=1 366  conf:(0.81) <
lift:(1.39)> lev:(0.1) [101] conv:(2.19)
25. credit_history=Good 523 ==> existing_credits=1 425    conf:(0.81) < lift:(1.38)>
lev:(0.12) [118] conv:(2.18)
26. existing_credits=1 581 ==> credit_history=Good 425    conf:(0.73) < lift:(1.38)>
lev:(0.12) [118] conv:(1.75)
27. other_parties=none existing_credits=1 464 ==> credit_history=Good num_dependents=1 291
conf:(0.63) < lift:(1.38)> lev:(0.08) [80] conv:(1.45)
28. credit_history=Good num_dependents=1 450 ==> other_parties=none existing_credits=1 291
conf:(0.65) < lift:(1.38)> lev:(0.08) [80] conv:(1.49)
```

```
29. credit_history=Good other_payment_plans=none 384 ==> other_parties=none existing_credits=1
248    conf:(0.65) < lift:(1.38)> lev:(0.07) [68] conv:(1.49)
30. other_parties=none existing_credits=1 464 ==> credit_history=Good other_payment_plans=none
248    conf:(0.53) < lift:(1.38)> lev:(0.07) [68] conv:(1.31)
```



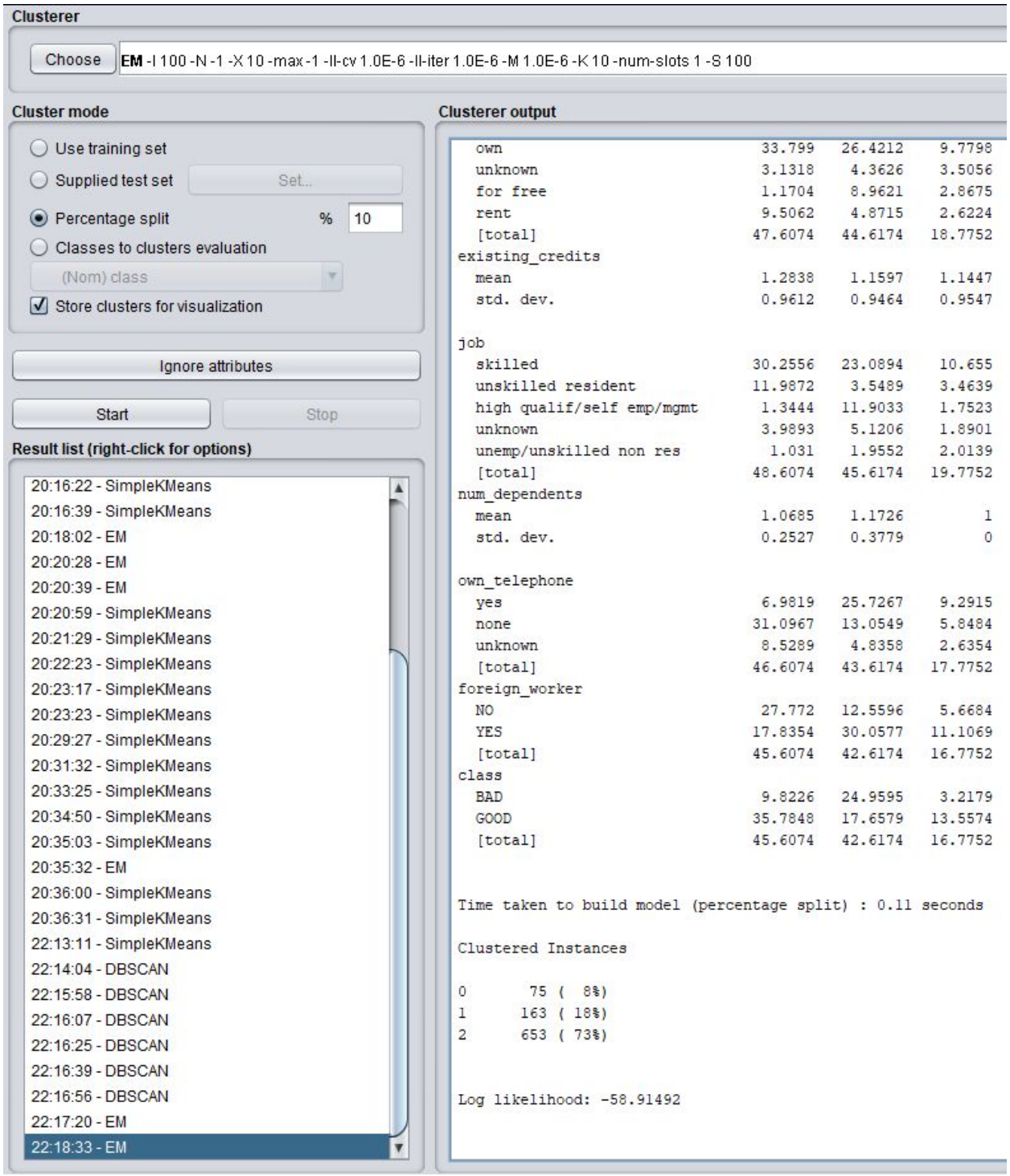
For these rules, we removed some attributes, as they were not producing interesting associations and were filling the top results provided by Weka. The attributes that the apriori algorithm ran on for this set of associations were:

```
checking_amt,
duration,
credit_history,
purpose,
credit_amount,
employment,
installment_commitment,
other_parties,
residence_since,
property_magnitude,
age,
other_payment_plans,
housing,
existing_credits,
job,
num_dependents,
own_telephone,
foreign_worker,
gender,
class,
```

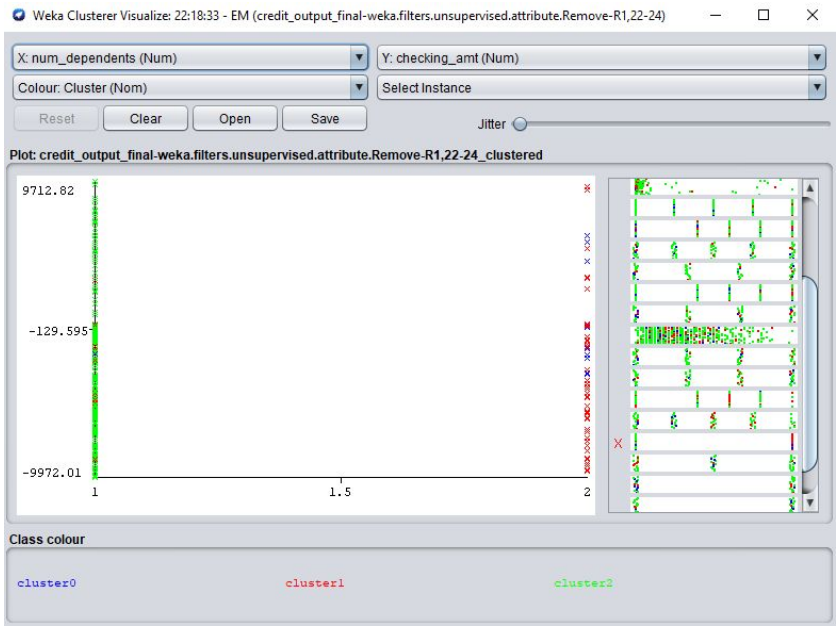
Additionally, these association rules were sorted by lift.

Clustering:

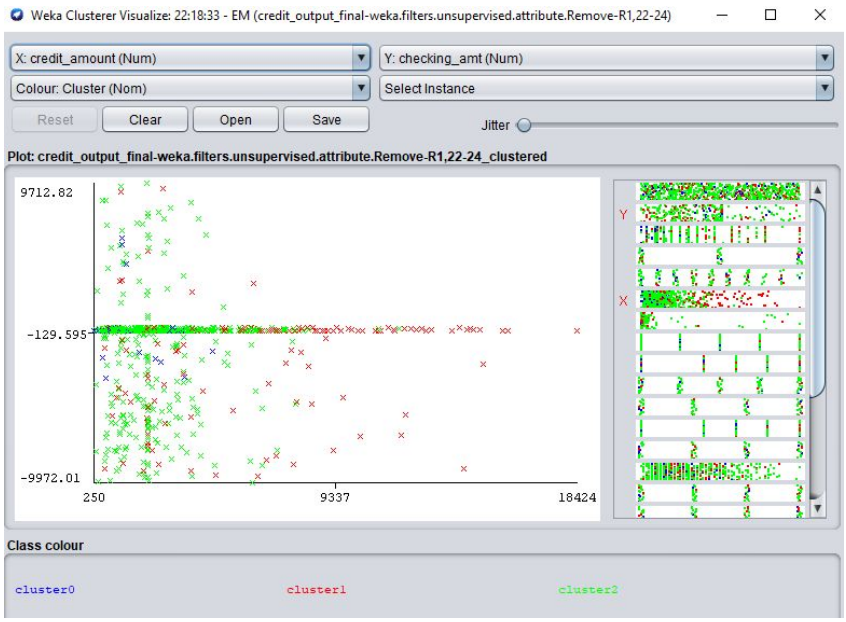
We decided to run the built-in EM classifier in Weka. This allowed the algorithm to pick the ideal number of clusters. The clustering was performed on the unbinned dataset. We settled with a percentage split of 10% of the data for training.



There were a couple of interesting observations in the dataset in regards to certain clusters.

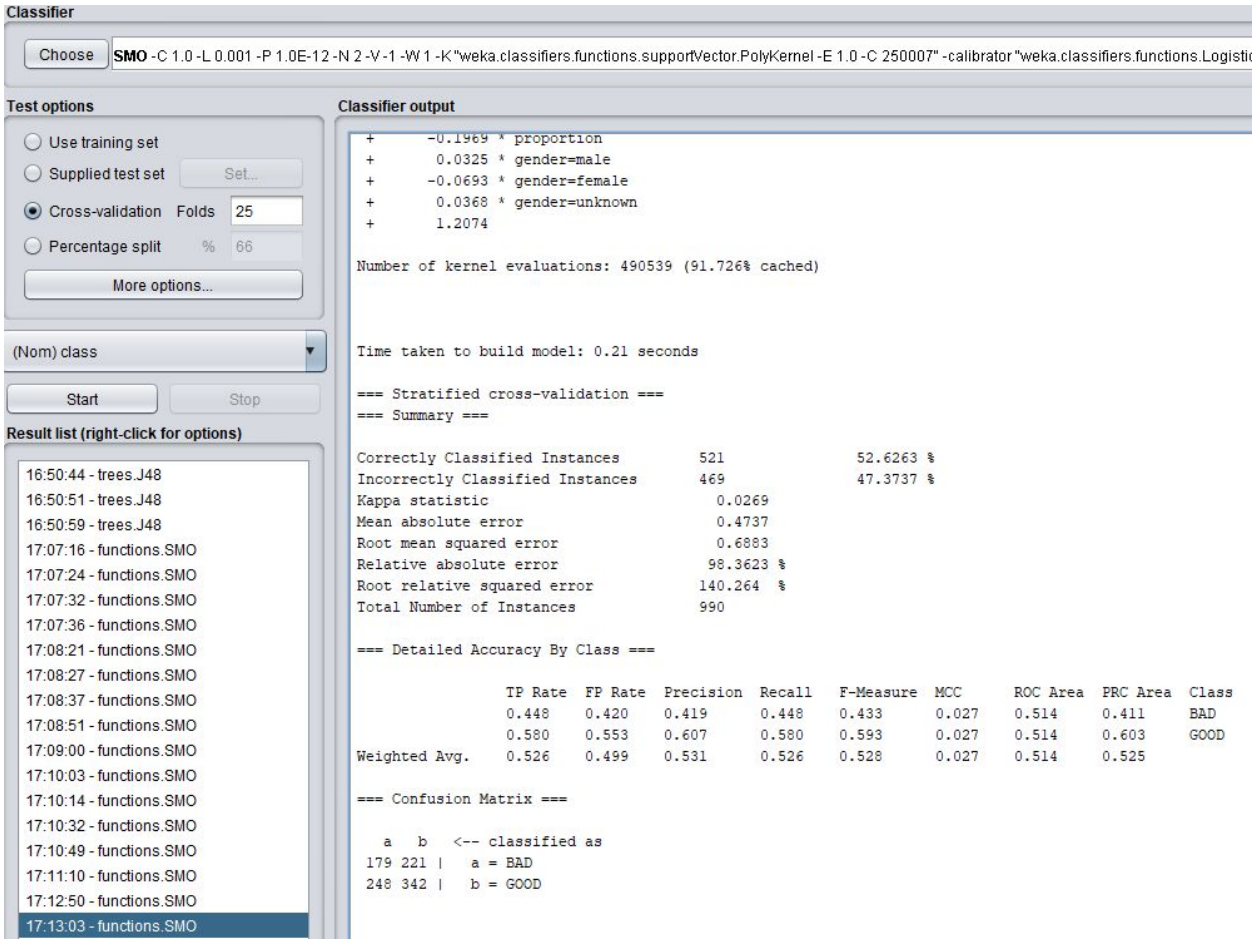


In this graph we noticed a clustering of data on num_dependents. It seems like Cluster2 heavily favored num_dependents = 1 and Cluster1 heavily favored num_dependents = 2. If we move over to another graph we notice that Cluster1 indicates a higher credit_amount and Cluster2 a lower one. This allows us to infer that num_dependents is directly related to the credit amount.



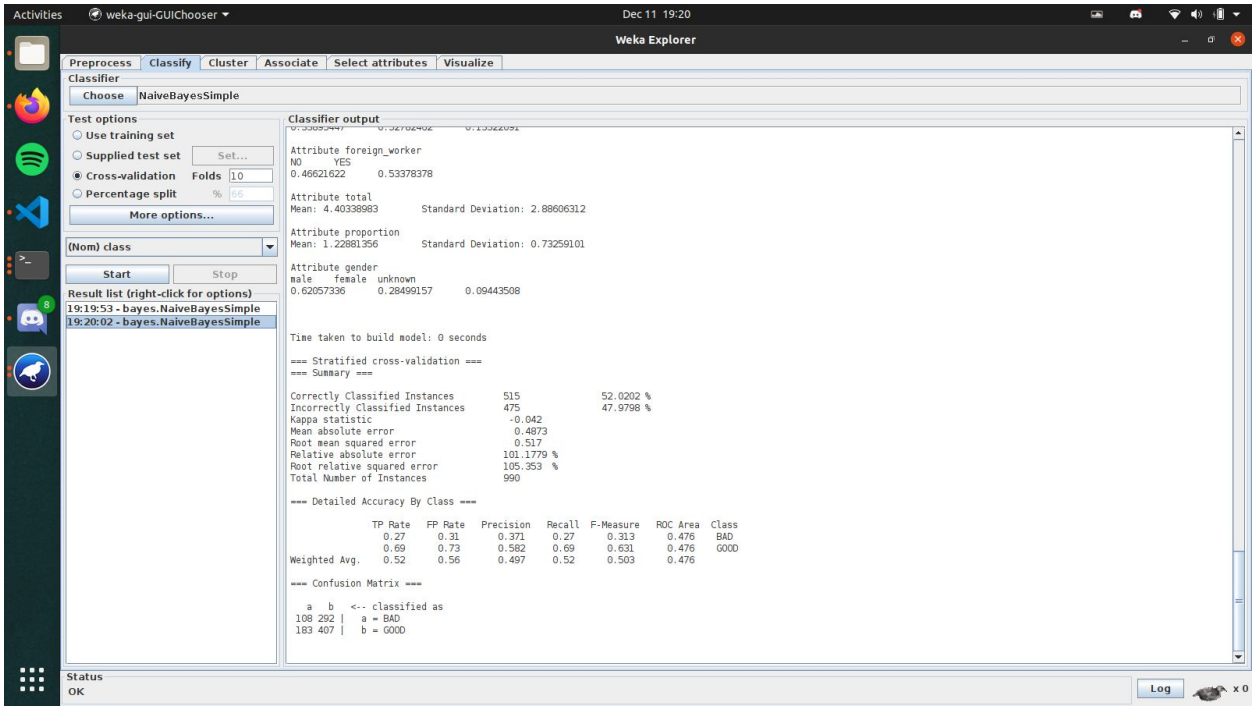
Support Vector Machine

We choose to run the Weka SVM model with 25-fold cross validation, making sure to turn off the settings to normalize data. This model was run on the binned data set. This configuration provided the best results with an accuracy of 52.6263% and a kappa statistic of 0.0269 and reasonable TP and FP rates. While this is quite inaccurate this is the best configuration of the SVM method given the cleaning we did of the dataset.



Simple Bayesian Network:

A simple Bayesian network was constructed using the binned version of our dataset using 10-times cross validation. The resulting network gave an accuracy of 52.0%, A TP rate of .69 for GOOD classifications and a TP of 0.27 for BAD classifications. Additional statistics can be found in the screenshots below. Under no circumstances is this model effective at being a classifier for this data.



Stacking Classifier:

A stacking classifier was created in python on the binned dataset using 5-fold cross validation. The following classifiers were used in the stack: KNN, RandomForest, Naive Bayes, SVM (w/ rbf kernel), GBM classifier (same model used in the GBM section of this project). The following was the output of the script that was run:

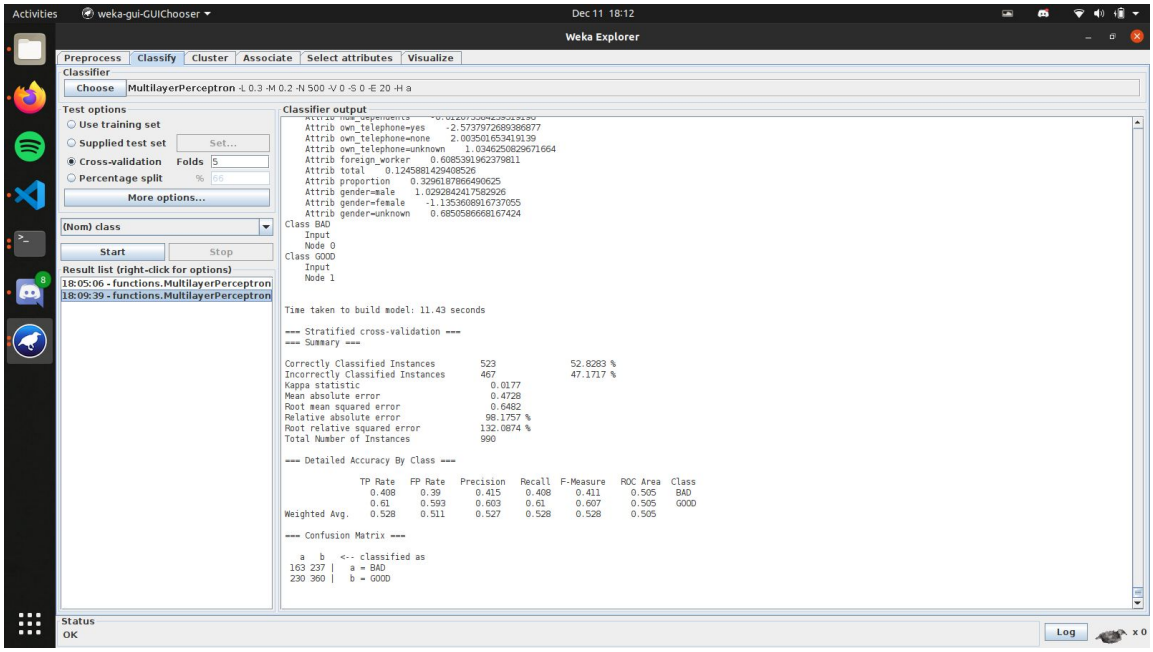
```
Summary of individual classifier

Accuracy: 0.47 (+/- 0.03) [KNN]
Accuracy: 0.55 (+/- 0.02) [Random Forest]
Accuracy: 0.53 (+/- 0.02) [Naive Bayes]
Accuracy: 0.60 (+/- 0.00) [SVM]
Accuracy: 0.53 (+/- 0.02) [GBM]
Accuracy: 0.53 (+/- 0.05) [Stacking]

Summary of stacking classifier

Accuracy: 0.5505
Precision: 0.6438
Recall: 0.7176
ROC AUC: 0.4707
F-measure: 0.6787
Kappa: -0.062
Confusion matrix
[[15 52]
 [37 94]]
0.5505050505050505
```

This model has an accuracy of 55.1%, but has a large amount of false positives for GOOD classifications. As a result, precision is low. Additionally, the Kappa score is negative, so this classifier is certainly not effective.

(Note that the accuracy during cross validation was 0.531, however due to how `sklearn.model_selection.cross_val_score` works, that exact model is not reachable. Data was split 90%/10% to retrain, similar to how the cross-validation would train the model. The statistics above are values given from the 10% test data)

Artificial Neural Network:

Weka's MultilayerPerceptron was used to create a classifier. The model was trained using the binned dataset using 5-fold cross validation, yielding 52.8% accuracy and a 0.61 TP rate for GOOD classifications and 0.408 TP rate for BAD classifications. The model leaves much to be desired with unimpressive precision (0.527) and recall (0.528) and Kappa statistic of 0.0177.



Gradient Boosting Machine:

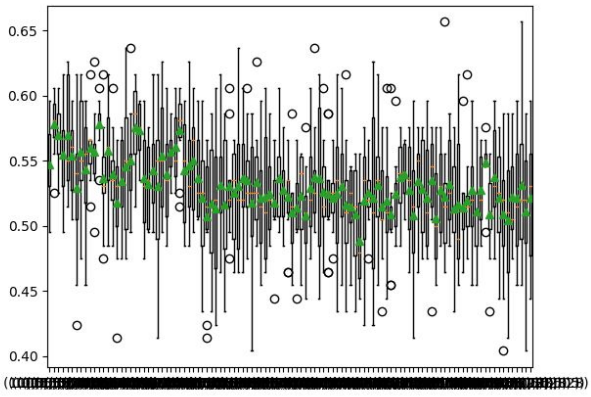
We decided to create the gradient boosting machine using a modified version of the GBM Python script that was used in class. The binned version of our dataset was used as it provided the best results. The following hyper parameters were tested using 10-fold cross validation: n_estimators=[10,50,100], max_features=[5,7,10,20], max_depth=[5,7,10], subsample=[0.2,0.5,0.8].

We found that the GBM using n_estimators=10 almost exclusively classified instances as 'GOOD' and ultimately was not behaving as a good classifier. The hyper-parameters that we found worked best were (100, 10, 10, 0.5), where values were ordered according to the hyper-parameters listed above.

A summary of the model is given below

```
Accuracy: 0.6061
Precision: 0.6329
Recall: 0.8333
ROC AUC: 0.5449
F-measure: 0.7194
Kappa: 0.0981
Confusion matrix
[[10 29]
 [10 50]]
```

(Note that the accuracy during cross validation was 0.557, however due to how `sklearn.model_selection.cross_val_score` works, that exact model is not reachable. Data was split 90%/10% to retrain, similar to how the cross-validation would train the model. The statistics above are values given from the 10% test data)



Appendix A:

```
savings <= 0
|   proportion <= 0: GOOD (108.0/34.0)
|   proportion > 0
|   |   other_parties = none
|   |   |   own_telephone = yes
|   |   |   |   residence_since <= -1
|   |   |   |   |   checking_amt <= 4: GOOD (22.0/2.0)
|   |   |   |   |   checking_amt > 4: BAD (4.0/1.0)
|   |   |   |   residence_since > -1
|   |   |   |   |   num_dependents <= 1
|   |   |   |   |   |   gender = male
|   |   |   |   |   |   |   purpose = radio/tv
|   |   |   |   |   |   |   |   total <= 5
|   |   |   |   |   |   |   |   |   residence_since <= 3: GOOD (11.0/2.0)
|   |   |   |   |   |   |   |   |   residence_since > 3: BAD (4.0/1.0)
|   |   |   |   |   |   |   |   total > 5: BAD (9.0)
|   |   |   |   |   |   |   purpose = education
|   |   |   |   |   |   |   |   duration <= 4: GOOD (2.0)
|   |   |   |   |   |   |   |   duration > 4: BAD (3.0/1.0)
|   |   |   |   |   |   |   purpose = furniture/equipment: BAD (14.0/3.0)
|   |   |   |   |   |   |   purpose = other: GOOD (13.0/3.0)
|   |   |   |   |   |   |   purpose = used car
|   |   |   |   |   |   |   |   personal_status = male single
|   |   |   |   |   |   |   |   |   proportion <= 1
|   |   |   |   |   |   |   |   |   |   total <= 5: BAD (2.0)
|   |   |   |   |   |   |   |   |   |   total > 5: GOOD (11.0/2.0)
|   |   |   |   |   |   |   |   |   proportion > 1: BAD (4.0)
|   |   |   |   |   |   |   |   personal_status = female div/dep/mar: GOOD (0.0)
|   |   |   |   |   |   |   |   personal_status = male div/sep: GOOD (2.0)
|   |   |   |   |   |   |   |   personal_status = male mar/wid: GOOD (2.0)
|   |   |   |   |   |   |   |   personal_status = unknown: GOOD (0.0)
|   |   |   |   |   |   |   purpose = new car
|   |   |   |   |   |   |   |   installment_commitment <= 1: BAD (3.0)
|   |   |   |   |   |   |   |   installment_commitment > 1
|   |   |   |   |   |   |   |   |   employment <= 1: BAD (4.0/1.0)
|   |   |   |   |   |   |   |   |   employment > 1: GOOD (10.0)
|   |   |   |   |   |   |   purpose = business: GOOD (17.0/6.0)
|   |   |   |   |   |   |   purpose = domestic appliance: GOOD (1.0)
|   |   |   |   |   |   |   purpose = repairs: GOOD (0.0)
|   |   |   |   |   |   |   purpose = retraining: GOOD (1.0)
|   |   |   |   |   |   gender = female
|   |   |   |   |   |   |   housing = own
```

```
|  |  |  |  |  |  |  |  foreign_worker = NO: GOOD (11.0/3.0)
|  |  |  |  |  |  |  |  foreign_worker = YES
|  |  |  |  |  |  |  |  |  existing_credits <= -1: GOOD (2.0)
|  |  |  |  |  |  |  |  |  existing_credits > -1: BAD (19.0/5.0)
|  |  |  |  |  |  |  housing = unknown
|  |  |  |  |  |  |  |  age <= 1: BAD (2.0)
|  |  |  |  |  |  |  |  age > 1
|  |  |  |  |  |  |  |  |  installment_commitment <= 2: BAD (2.0)
|  |  |  |  |  |  |  |  |  installment_commitment > 2: GOOD (4.0/1.0)
|  |  |  |  |  |  |  housing = for free: BAD (4.0)
|  |  |  |  |  |  |  housing = rent
|  |  |  |  |  |  |  |  checking_amt <= 1: GOOD (3.0/1.0)
|  |  |  |  |  |  |  |  checking_amt > 1: BAD (5.0)
|  |  |  |  |  |  gender = unknown
|  |  |  |  |  |  |  proportion <= 1
|  |  |  |  |  |  |  |  purpose = radio/tv: BAD (2.0)
|  |  |  |  |  |  |  |  purpose = education: BAD (0.0)
|  |  |  |  |  |  |  |  purpose = furniture/equipment: GOOD (4.0)
|  |  |  |  |  |  |  |  purpose = other: GOOD (3.0/1.0)
|  |  |  |  |  |  |  |  purpose = used car: BAD (3.0)
|  |  |  |  |  |  |  |  purpose = new car: GOOD (1.0)
|  |  |  |  |  |  |  |  purpose = business: BAD (1.0)
|  |  |  |  |  |  |  |  purpose = domestic appliance: BAD (0.0)
|  |  |  |  |  |  |  |  purpose = repairs: BAD (0.0)
|  |  |  |  |  |  |  |  purpose = retraining: BAD (0.0)
|  |  |  |  |  |  |  proportion > 1: GOOD (6.0)
|  |  |  |  |  num_dependents > 1: BAD (30.0/9.0)
|  |  |  own_telephone = none
|  |  |  |  credit_history = Bad
|  |  |  |  |  other_payment_plans = none
|  |  |  |  |  |  num_dependents <= 1: GOOD (58.0/12.0)
|  |  |  |  |  |  num_dependents > 1
|  |  |  |  |  |  |  employment <= 3: GOOD (8.0/2.0)
|  |  |  |  |  |  |  employment > 3: BAD (4.0)
|  |  |  |  |  other_payment_plans = unknown
|  |  |  |  |  |  existing_credits <= 1: BAD (2.0)
|  |  |  |  |  |  existing_credits > 1: GOOD (8.0/1.0)
|  |  |  |  |  other_payment_plans = bank
|  |  |  |  |  |  job = skilled: BAD (3.0/1.0)
|  |  |  |  |  |  job = unskilled resident: GOOD (6.0)
|  |  |  |  |  |  job = high qualif/self emp/mgmt: BAD (1.0)
|  |  |  |  |  |  job = unknown: BAD (2.0)
|  |  |  |  |  |  job = unemp/unskilled non res: GOOD (0.0)
|  |  |  |  |  other_payment_plans = stores: BAD (3.0)
```

```
|  |  |  |  credit_history = Good
|  |  |  |  |  residence_since <= -1
|  |  |  |  |  |  other_payment_plans = none: BAD (14.0/2.0)
|  |  |  |  |  |  other_payment_plans = unknown: GOOD (3.0/1.0)
|  |  |  |  |  |  other_payment_plans = bank: BAD (3.0)
|  |  |  |  |  |  other_payment_plans = stores: GOOD (2.0)
|  |  |  |  |  residence_since > -1
|  |  |  |  |  |  existing_credits <= -1: GOOD (14.0/2.0)
|  |  |  |  |  |  existing_credits > -1
|  |  |  |  |  |  |  employment <= 0
|  |  |  |  |  |  |  |  age <= 1
|  |  |  |  |  |  |  |  |  duration <= 1: BAD (2.0)
|  |  |  |  |  |  |  |  |  duration > 1: GOOD (3.0)
|  |  |  |  |  |  |  |  age > 1: GOOD (9.0)
|  |  |  |  |  |  |  employment > 0
|  |  |  |  |  |  |  |  purpose = radio/tv
|  |  |  |  |  |  |  |  |  job = skilled
|  |  |  |  |  |  |  |  |  |  housing = own
|  |  |  |  |  |  |  |  |  |  |  total <= 2: GOOD (3.0)
|  |  |  |  |  |  |  |  |  |  |  total > 2: BAD (18.0/5.0)
|  |  |  |  |  |  |  |  |  |  housing = unknown: BAD (3.0/1.0)
|  |  |  |  |  |  |  |  |  |  housing = for free: GOOD (1.0)
|  |  |  |  |  |  |  |  |  |  housing = rent: GOOD (3.0/1.0)
|  |  |  |  |  |  |  |  |  job = unskilled resident: GOOD (9.0/1.0)
|  |  |  |  |  |  |  |  |  job = high qualif/self emp/mgmt: GOOD (0.0)
|  |  |  |  |  |  |  |  |  job = unknown: BAD (2.0)
|  |  |  |  |  |  |  |  |  job = unemp/unskilled non res: GOOD (0.0)
|  |  |  |  |  |  |  |  purpose = education
|  |  |  |  |  |  |  |  |  employment <= 3: GOOD (4.0)
|  |  |  |  |  |  |  |  |  employment > 3: BAD (2.0)
|  |  |  |  |  |  |  |  purpose = furniture/equipment
|  |  |  |  |  |  |  |  |  property_magnitude = real estate: GOOD (4.0/1.0)
|  |  |  |  |  |  |  |  |  property_magnitude = no known property: BAD (4.0)
|  |  |  |  |  |  |  |  |  property_magnitude = life insurance
|  |  |  |  |  |  |  |  |  |  total <= 4: BAD (2.0)
|  |  |  |  |  |  |  |  |  |  total > 4: GOOD (5.0/1.0)
|  |  |  |  |  |  |  |  |  property_magnitude = car: GOOD (4.0)
|  |  |  |  |  |  |  |  purpose = other
|  |  |  |  |  |  |  |  |  proportion <= 1: BAD (7.0/1.0)
|  |  |  |  |  |  |  |  |  proportion > 1: GOOD (5.0/1.0)
|  |  |  |  |  |  |  |  purpose = used car: GOOD (8.0)
|  |  |  |  |  |  |  |  purpose = new car
|  |  |  |  |  |  |  |  |  housing = own
|  |  |  |  |  |  |  |  |  |  other_payment_plans = none
```

```
|  |  |  |  |  |  |  |  |  |  |  employment <= 2: GOOD (8.0/2.0)
|  |  |  |  |  |  |  |  |  |  |  employment > 2: BAD (3.0)
|  |  |  |  |  |  |  |  |  |  other_payment_plans = unknown: BAD (1.0)
|  |  |  |  |  |  |  |  |  |  other_payment_plans = bank: GOOD (2.0)
|  |  |  |  |  |  |  |  |  |  other_payment_plans = stores: GOOD (1.0)
|  |  |  |  |  |  |  |  |  housing = unknown: BAD (2.0)
|  |  |  |  |  |  |  |  |  housing = for free: GOOD (1.0)
|  |  |  |  |  |  |  |  |  housing = rent: BAD (2.0)
|  |  |  |  |  |  |  |  purpose = business
|  |  |  |  |  |  |  |  |  employment <= 2: BAD (3.0/1.0)
|  |  |  |  |  |  |  |  |  employment > 2: GOOD (2.0)
|  |  |  |  |  |  |  |  purpose = domestic appliance: BAD (2.0/1.0)
|  |  |  |  |  |  |  |  purpose = repairs: BAD (3.0/1.0)
|  |  |  |  |  |  |  |  purpose = retraining: BAD (2.0/1.0)
|  |  |  |  credit_history = Moderate
|  |  |  |  |  gender = male
|  |  |  |  |  |  age <= 1: BAD (3.0)
|  |  |  |  |  |  age > 1: GOOD (41.0/13.0)
|  |  |  |  |  gender = female
|  |  |  |  |  |  employment <= 1: GOOD (7.0/1.0)
|  |  |  |  |  |  employment > 1
|  |  |  |  |  |  |  duration <= 0: GOOD (3.0/1.0)
|  |  |  |  |  |  |  duration > 0: BAD (9.0)
|  |  |  |  |  gender = unknown: BAD (5.0)
|  |  |  own_telephone = unknown
|  |  |  |  num_dependents <= 1
|  |  |  |  |  job = skilled
|  |  |  |  |  |  credit_amount <= 2: GOOD (8.0/1.0)
|  |  |  |  |  |  credit_amount > 2
|  |  |  |  |  |  |  foreign_worker = NO: BAD (9.0/2.0)
|  |  |  |  |  |  |  foreign_worker = YES
|  |  |  |  |  |  |  |  duration <= 3
|  |  |  |  |  |  |  |  |  employment <= 2: GOOD (6.0/2.0)
|  |  |  |  |  |  |  |  |  employment > 2: BAD (2.0)
|  |  |  |  |  |  |  |  duration > 3: GOOD (6.0)
|  |  |  |  |  job = unskilled resident
|  |  |  |  |  |  gender = male
|  |  |  |  |  |  |  age <= 2
|  |  |  |  |  |  |  |  installment_commitment <= 3: BAD (2.0)
|  |  |  |  |  |  |  |  installment_commitment > 3: GOOD (2.0)
|  |  |  |  |  |  |  age > 2: BAD (5.0)
|  |  |  |  |  |  gender = female: GOOD (2.0)
|  |  |  |  |  |  gender = unknown: BAD (0.0)
|  |  |  |  |  job = high qualif/self emp/mgmt
```

```
|  |  |  |  |  |  installment_commitment <= 1: BAD (4.0/1.0)
|  |  |  |  |  |  installment_commitment > 1: GOOD (6.0)
|  |  |  |  |  job = unknown: BAD (3.0)
|  |  |  |  |  job = unemp/unskilled non res: GOOD (3.0/1.0)
|  |  |  |  num_dependents > 1: GOOD (6.0)
|  |  other_parties = unknown
|  |  |  total <= 8
|  |  |  |  property_magnitude = real estate
|  |  |  |  |  gender = male
|  |  |  |  |  |  residence_since <= 1: BAD (4.0/1.0)
|  |  |  |  |  |  residence_since > 1: GOOD (14.0/2.0)
|  |  |  |  |  gender = female: BAD (4.0/1.0)
|  |  |  |  |  gender = unknown: GOOD (3.0)
|  |  |  |  property_magnitude = no known property
|  |  |  |  |  credit_amount <= 7: GOOD (18.0/5.0)
|  |  |  |  |  credit_amount > 7: BAD (4.0/1.0)
|  |  |  |  property_magnitude = life insurance: GOOD (11.0/1.0)
|  |  |  |  property_magnitude = car
|  |  |  |  |  other_payment_plans = none
|  |  |  |  |  |  housing = own
|  |  |  |  |  |  |  credit_history = Bad: BAD (3.0)
|  |  |  |  |  |  |  credit_history = Good
|  |  |  |  |  |  |  |  installment_commitment <= -1: BAD (2.0)
|  |  |  |  |  |  |  |  installment_commitment > -1: GOOD (8.0/1.0)
|  |  |  |  |  |  |  credit_history = Moderate: BAD (5.0/1.0)
|  |  |  |  |  |  housing = unknown: GOOD (4.0)
|  |  |  |  |  |  housing = for free: BAD (1.0)
|  |  |  |  |  |  housing = rent: BAD (2.0)
|  |  |  |  |  other_payment_plans = unknown: GOOD (2.0)
|  |  |  |  |  other_payment_plans = bank: BAD (4.0)
|  |  |  |  |  other_payment_plans = stores: BAD (0.0)
|  |  |  total > 8: BAD (5.0)
|  |  other_parties = guarantor
|  |  |  duration <= 6
|  |  |  |  credit_history = Bad: GOOD (7.0/1.0)
|  |  |  |  credit_history = Good
|  |  |  |  |  personal_status = male single: GOOD (11.0/4.0)
|  |  |  |  |  personal_status = female div/dep/mar
|  |  |  |  |  |  age <= 2
|  |  |  |  |  |  |  residence_since <= 3: BAD (2.0)
|  |  |  |  |  |  |  residence_since > 3: GOOD (2.0)
|  |  |  |  |  |  age > 2: GOOD (3.0)
|  |  |  |  |  personal_status = male div/sep: GOOD (0.0)
|  |  |  |  |  personal_status = male mar/wid: BAD (3.0)
```

```
|  |  |  |  |  personal_status = unknown: BAD (2.0)
|  |  |  |  credit_history = Moderate: GOOD (2.0)
|  |  |  duration > 6: BAD (4.0)
|  |  other_parties = co applicant
|  |  |  other_payment_plans = none: GOOD (21.0/4.0)
|  |  |  other_payment_plans = unknown: BAD (1.0)
|  |  |  other_payment_plans = bank
|  |  |  |  installment_commitment <= 2: GOOD (3.0)
|  |  |  |  installment_commitment > 2: BAD (2.0)
|  |  |  other_payment_plans = stores: BAD (1.0)
savings > 0
|  other_parties = none
|  |  gender = male
|  |  |  existing_credits <= 1
|  |  |  |  residence_since <= 3
|  |  |  |  |  installment_commitment <= 1
|  |  |  |  |  |  checking_amt <= 3: GOOD (2.0)
|  |  |  |  |  |  checking_amt > 3: BAD (3.0)
|  |  |  |  |  installment_commitment > 1: GOOD (19.0/1.0)
|  |  |  |  residence_since > 3: BAD (10.0/4.0)
|  |  |  existing_credits > 1
|  |  |  |  personal_status = male single
|  |  |  |  |  age <= 2
|  |  |  |  |  |  employment <= 2: BAD (4.0/1.0)
|  |  |  |  |  |  employment > 2: GOOD (3.0)
|  |  |  |  |  age > 2: BAD (3.0)
|  |  |  |  personal_status = female div/dep/mar: BAD (0.0)
|  |  |  |  personal_status = male div/sep: GOOD (2.0)
|  |  |  |  personal_status = male mar/wid: BAD (3.0)
|  |  |  |  personal_status = unknown: BAD (0.0)
|  |  gender = female: GOOD (26.0/4.0)
|  |  gender = unknown: GOOD (6.0/1.0)
|  other_parties = unknown
|  |  checking_amt <= 4: GOOD (4.0)
|  |  checking_amt > 4: BAD (2.0)
|  other_parties = guarantor: BAD (1.0)
|  other_parties = co applicant: BAD (2.0)
```