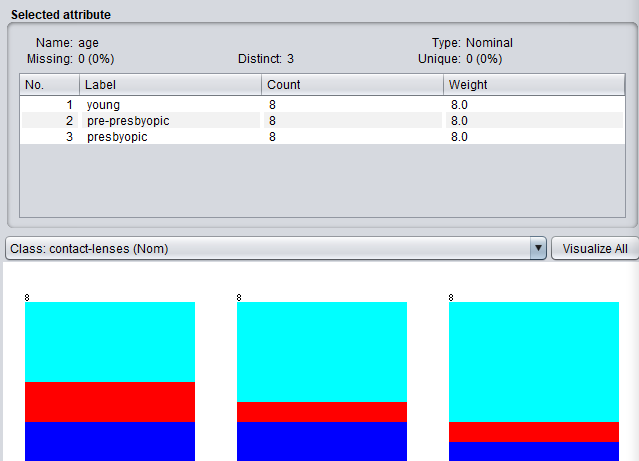
# Homework Writeup

## Using Weka Explorer

### The Data

First things first, we had to go find and load our dataset. Since I have multiple harddrives, I had to move from my C: drive to my G: drive and access the directory pertaining to Weka. Once there, one more move was into Weka-3.8.5/data and there lies all the available resources!

After loading it up for the first time, we can see the data has 5 attributes, 24 instances, and a class/ relation of contact-lenses. When clicking through each attribute, in the selected attribute column, it contains the different values that it contains, the count of each, and weight for each (usually count \* 1). For example, age has only 3 distinct attributes: young, pre-presbyopic, and presbyopic. Underneath the selected attribute container depicts a graphical representation of the selected attribute. For age we get:



One might ask themselves here, what each color represents. These colors have to be associated with our class, contact-lenses. So by clicking on contact lenses we see that the color blue is associated with soft, 5/24, hard is red, 9/24, and none is light blue, 15/24.

ALL CLASSIFIERS BELOW ARE BASED ON THE FREQUENCY TABLE

### How to read “Detailed Accuracy By Class” - [How to interpret weka classification?](https://stackoverflow.com/questions/2903933/how-to-interpret-weka-classification)

* “TP Rate: rate of true positives (instances correctly classified as a given class)
* FP Rate: rate of false positives (instances falsely classified as a given class)
* Precision: proportion of instances that are truly of a class divided by the total instances classified as that class
* Recall: proportion of instances classified as a given class divided by the actual total in that class (equivalent to TP rate)
* F-Measure: A combined measure for precision and recall calculated as 2 \* Precision \* Recall / (Precision + Recall)
* As for the ROC area measurement, it is one of the most important values output by Weka. An "optimal" classifier will have ROC area values approaching 1, with 0.5 being comparable to "random guessing" (similar to a Kappa statistic of 0).
* Kappa is a chance-corrected measure of agreement between the classifications and the true classes. It's calculated by taking the agreement expected by chance away from the observed agreement and dividing by the maximum possible agreement.”

### 

### ZeroR - [The ZeroR Classifier .. What it is and How it Works](https://www.youtube.com/watch?v=kUbYN4AcPmA&t=40s)

This is the simplest classification method, which relies on the target (class) and ignores all other predictors. Since there is no predictability with ZeroR, its main purpose is to create a baseline performance as benchmark for all other classifiers. It works by constructing a frequency table and selecting its most common value. Below is the run:

=== Run information ===

Scheme: weka.classifiers.rules.ZeroR

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

Test mode: evaluate on training data

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

ZeroR predicts class value: none

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances 15 62.5 %

Incorrectly Classified Instances 9 37.5 %

Kappa statistic 0

Mean absolute error 0.3683

Root mean squared error 0.4242

Relative absolute error 100 %

Root relative squared error 100 %

Total Number of Instances 24

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.000 0.000 ? 0.000 ? ? 0.500 0.208 soft

0.000 0.000 ? 0.000 ? ? 0.500 0.167 hard

1.000 1.000 0.625 1.000 0.769 ? 0.500 0.625 none

Weighted Avg. 0.625 0.625 ? 0.625 ? ? 0.500 0.462

=== Confusion Matrix ===

a b c <-- classified as

0 0 5 | a = soft

0 0 4 | b = hard

0 0 15 | c = none

Okay the run above showed that it took almost no time to solve and the most common item found in our target(class) was: none. Which makes sense, cause we have 15 none assignments and like 9 actual contact prescriptions. Since the class value was predicted to be none, there were only 15 correctly classified instances in our data, 62.5%, and the other 9 are incorrect, since their contact lense values were either hard or soft. And we had no false positives or false negatives, which is no surprise given our dataset (this can be seen in the confusion matrix). Since we had a positive prediction value of 0.625 and 0.0 for negative prediction value, our accuracy/baseline moving forward is 0.625. (our ROC says for anything other than guessing none, we might as well have thrown a dart, since 0.2 for soft and 0.16 for hard [random chance is 0.5])

### 

### OneR - [The OneR Classifier .. What it is and How it Works](https://www.youtube.com/watch?v=phnkMGDZUNI)

OneR creates “one rule”, which has to be very general, simple, and accurate (instead of ignoring all predictors, it chooses the best one if you will). What it does is, find one rule for each attribute/predictor and then selects the rule with the smallest fail percentage. To find the rule, we must first create a frequency table for each attribute against the target/class. The total error calculated for each attribute is the measure of each attribute's contribution. Low total error = higher contribution to predictability of model (cause we are choosing the lowest error attribution rule to choose as our “OneR”) OneR handles missing values by treating them as separate attribute value.

Algorithm Breakdown

For each attribute:

For each value of the attribute, make a rule that fits the following:

Count how often each value of class appears

Find the most frequently appearing class

Make a rule that assigns the frequent class to this attribute value

Calculate the % error of the rule for each attribute

Choose the smallest % error rule

=== Run information ===

Scheme: weka.classifiers.rules.OneR -B 6

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

Test mode: evaluate on training data

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

tear-prod-rate:

Reduced -> none

normal -> soft

(17/24 instances correct)

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances 17 70.8333 %

Incorrectly Classified Instances 7 29.1667 %

Kappa statistic 0.5

Mean absolute error 0.1944

Root mean squared error 0.441

Relative absolute error 52.7933 %

Root relative squared error 103.9561 %

Total Number of Instances 24

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 0.368 0.417 1.000 0.588 0.513 0.816 0.417 soft

0.000 0.000 ? 0.000 ? ? 0.500 0.167 hard

0.800 0.000 1.000 0.800 0.889 0.775 0.900 0.925 none

Weighted Avg. 0.708 0.077 ? 0.708 ? ? 0.816 0.693

=== Confusion Matrix ===

a b c <-- classified as

5 0 0 | a = soft

4 0 0 | b = hard

3 0 12 | c = none

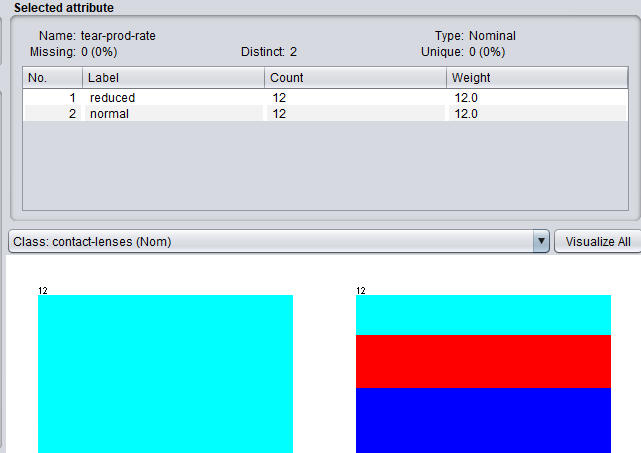
Interesting enough, the classifiers rule was actually a two condition rule. The attribute it choose to work off was tear-production rate, which easily seen for the first statement of the rule:

tear-prod-rate:

Reduced -> none

normal -> soft

(17/24 instances correct)



When the tear-prod-rate was reduced, the dataset shown consistently 100% of the time, no glasses are given. When tear-prod-rate is normal, always recommending soft is not the best statement. By traveling back to the preprocess tab and seeing the breakdown of tear-prod-rate, we can see that normal has almost ⅓ of each attribute labeled as a different class each, with soft being the utmost common. The 7 misclassified instances must have been due to a result of the final rule, since there were 4 hard, normal and 3 none, normal cases. Nonetheless, our OneR rule still did a phenomenal job at 71%, which is a very common thing for the OneR classifier to work on par with extreme ML algorithms (which is kinda cool). Looking at the detailed breakdown by class, its interesting to note that precision is ? for both hard and none for normal tear-prod, since they were never classified as such due to the OneR. Soft has a precision of 0.417 and none had a precision of 1.00, which makes sense due to part one of rule. Also here, our Kappa rule says its at 0.5, which means that our classifier is doing better than chance (50% better?)! <- This statement is also proved by the ROC of each class value

### 

### NaiveBayes - [Naive Bayes, Clearly Explained!!!](https://www.youtube.com/watch?v=O2L2Uv9pdDA)

The Naive Bayes is known as Naive due to the fact that it treats every attribute as equals and all are statistically independent from one another. When an attribute is missing a value, it adds one to everything, so that every attribute has a chance to occur. Surprisingly, this way is fairly accurate nonetheless, with high bias and low variance. Basically we multiply the probability of each element with one another multiplied by the possibility that statement occurs.

 or 

=== Run information ===

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

Test mode: evaluate on training data

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

Naive Bayes Classifier

Class

Attribute soft hard none

(0.22) (0.19) (0.59)

==========================================

age

young 3.0 3.0 5.0

pre-presbyopic 3.0 2.0 6.0

presbyopic 2.0 2.0 7.0

[total] 8.0 7.0 18.0

spectacle-prescrip

myope 3.0 4.0 8.0

hypermetrope 4.0 2.0 9.0

[total] 7.0 6.0 17.0

astigmatism

no 6.0 1.0 8.0

yes 1.0 5.0 9.0

[total] 7.0 6.0 17.0

tear-prod-rate

reduced 1.0 1.0 13.0

normal 6.0 5.0 4.0

[total] 7.0 6.0 17.0

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances 23 95.8333 %

Incorrectly Classified Instances 1 4.1667 %

Kappa statistic 0.925

Mean absolute error 0.1809

Root mean squared error 0.2357

Relative absolute error 49.1098 %

Root relative squared error 55.5663 %

Total Number of Instances 24

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 0.053 0.833 1.000 0.909 0.889 1.000 1.000 soft

1.000 0.000 1.000 1.000 1.000 1.000 1.000 1.000 hard

0.933 0.000 1.000 0.933 0.966 0.917 1.000 1.000 none

Weighted Avg. 0.958 0.011 0.965 0.958 0.960 0.925 1.000 1.000

=== Confusion Matrix ===

a b c <-- classified as

5 0 0 | a = soft

0 4 0 | b = hard

1 0 14 | c = none

If we wanted to find the example of something, such as the probability of no contacts, given that the user has young, myope, no, normal = (15/24)(5/11)(8/15)(8/15)(4/15) = 0.0215. Then we can take the opposite, such as the probability of soft glasses, given that the user has young, myope, no, normal: (5/24)(3/11)(3/15)(6/15)(6/15) = 0.00181.

Given that the user has young, myope, no, normal:

Probability of no contacts = 0.0215 / (0.0215 + 0.00181) = 92.23%

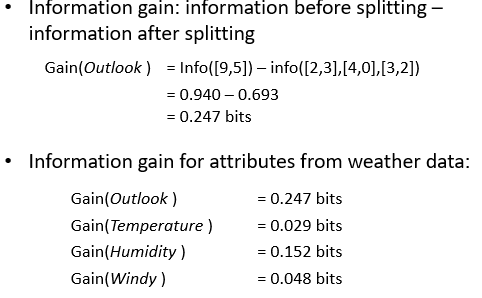
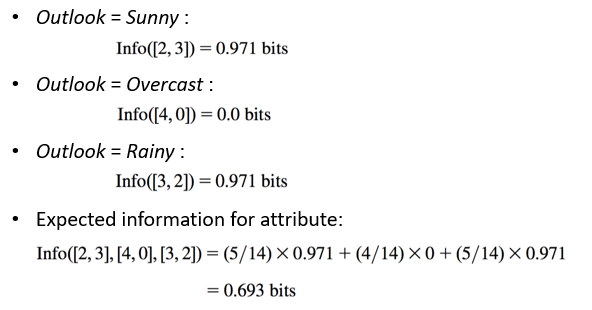
Probability of soft contacts = 0.0215 / (0.0215 + 0.00181) = 7.76%

The Naive Bayes has given us so far the best Kappa and ROC area so far seen. There were 23 correctly classified, with the only off being another no contact instance. Not really much else to mention other than that Laplace rule can be seen here, adding an additional one count to everything to make sure everyone has as probability.

### 

### J48

J48 creates a decision tree by top down learning, using recursive divide and conquer strategy. First one attribute is chosen to the root node, then create branches for all values of each attribute. Once thats done, we split the instances into subgroups, creating one for every single branch that runs off the root node. Then we continue this, making branches and breaking the instances, until all instances have the same class. The root node is often choosen based on the purest node, since it is also the most informational. The required information needed to guess the class is called distribution entropy, which is given to us in *bits*. Entropy is greatest when the likelihood of all classes are equal and lowest when one class has probability 1.





=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: contact-lenses

Instances: 24

Attributes: 5

age

spectacle-prescrip

astigmatism

tear-prod-rate

contact-lenses

Test mode: evaluate on training data

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

J48 pruned tree

------------------

tear-prod-rate = reduced: none (12.0)

tear-prod-rate = normal

| astigmatism = no: soft (6.0/1.0)

| astigmatism = yes

| | spectacle-prescrip = myope: hard (3.0)

| | spectacle-prescrip = hypermetrope: none (3.0/1.0)

Number of Leaves : 4

Size of the tree : 7

Time taken to build model: 0 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0 seconds

=== Summary ===

Correctly Classified Instances 22 91.6667 %

Incorrectly Classified Instances 2 8.3333 %

Kappa statistic 0.8447

Mean absolute error 0.0833

Root mean squared error 0.2041

Relative absolute error 22.6257 %

Root relative squared error 48.1223 %

Total Number of Instances 24

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

1.000 0.053 0.833 1.000 0.909 0.889 0.974 0.833 soft

0.750 0.000 1.000 0.750 0.857 0.845 0.988 0.917 hard

0.933 0.111 0.933 0.933 0.933 0.822 0.967 0.972 none

Weighted Avg. 0.917 0.080 0.924 0.917 0.916 0.840 0.972 0.934

=== Confusion Matrix ===

a b c <-- classified as

5 0 0 | a = soft

0 3 1 | b = hard

1 0 14 | c = none

Okay this tree here it produced ended up becoming an extremely accurate one, with our Kappa stat being 0.85, with 22 correct identifications and 2 off. The two mistakes were when hard was classified as none (young,hypermetrope,yes,normal,hard since hypermetrop -> none), and none classified as soft (presbyopic,myope,no,normal,none since no astigmatism -> soft). After realizing these two leaves (astigmatism = no and spectacle-prescrip = hypermetrope) were the only with false calls, they are the only with a division sign in their call (spectacle-prescrip = hypermetrope: none (3.0/1.0) with the 3.0 representing how many was correctly classified when reaching this leaf and one being incorrect.)

## 

## Block\_1.csv

### ZeroR

=== Run information ===

Scheme: weka.classifiers.rules.ZeroR

Relation: block\_1

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

ZeroR predicts class value: FALSE

Time taken to build model: 0.09 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.7 seconds

=== Summary ===

Correctly Classified Instances 572820 99.6359 %

Incorrectly Classified Instances 2093 0.3641 %

Kappa statistic 0

Mean absolute error 0.0073

Root mean squared error 0.0602

Relative absolute error 100 %

Root relative squared error 100 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.000 0.000 ? 0.000 ? ? 0.500 0.004 TRUE

1.000 1.000 0.996 1.000 0.998 ? 0.500 0.996 FALSE

Weighted Avg. 0.996 0.996 ? 0.996 ? ? 0.500 0.993

=== Confusion Matrix ===

a b <-- classified as

0 2093 | a = TRUE

0 572820 | b = FALSE

The data set here is pretty self explanatory, with over 5 million false matches, the zero rule was False, which gave us about a 99.6359% correctly classified instances. The ROC area here also shows that the zeror given was insanely accurate, but our kappa value says 0, meaning this rule basically is guessing randomly. Baseline for future testing seems to be 99.6%

#### 

### OneR

=== Run information ===

Scheme: weka.classifiers.rules.OneR -B 6

Relation: block\_1

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

cmp\_plz:

< 0.5 -> FALSE

>= 0.5 -> TRUE

? -> FALSE

(573656/574913 instances correct)

Time taken to build model: 1.69 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.62 seconds

=== Summary ===

Correctly Classified Instances 573656 99.7814 %

Incorrectly Classified Instances 1257 0.2186 %

Kappa statistic 0.7593

Mean absolute error 0.0022

Root mean squared error 0.0468

Relative absolute error 30.1313 %

Root relative squared error 77.6381 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.953 0.002 0.633 0.953 0.760 0.775 0.975 0.603 TRUE

0.998 0.047 1.000 0.998 0.999 0.775 0.975 1.000 FALSE

Weighted Avg. 0.998 0.047 0.998 0.998 0.998 0.775 0.975 0.998

=== Confusion Matrix ===

a b <-- classified as

1994 99 | a = TRUE

1158 571662 | b = FALSE

The rule made here cmp\_plz: {(< 0.5 -> FALSE), (>= 0.5-> TRUE), (?-> FALSE)} does better than zeror’s baseline, which makes total sense here. Here we see there are actually quite a lot of false positives. However, the OneR we created from the training set actually did well enough for classifying most matched cases. (This is not surprising to me since, I used cmp\_plz as my extension last week and analyzed multiple trends that occured in the aforementioned attribute)

#### 

### NaiveBayes

=== Run information ===

Scheme: weka.classifiers.bayes.NaiveBayes

Relation: block\_1

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

Naive Bayes Classifier

Class

Attribute TRUE FALSE

(0) (1)

=====================================

id\_1

mean 34440.8686 33267.6928

std. dev. 21738.7124 23629.1518

weight sum 2093 572820

precision 1.2298 1.2298

id\_2

mean 50889.3272 66621.9382

std. dev. 24376.4778 23620.1892

weight sum 2093 572820

precision 1.2267 1.2267

cmp\_fname\_c1

mean 0.997 0.7115

std. dev. 0.0397 0.3896

weight sum 2091 572720

precision 0.0233 0.0233

cmp\_fname\_c2

mean 0.9954 0.8966

std. dev. 0.0514 0.2757

weight sum 128 10197

precision 0.0417 0.0417

cmp\_lname\_c1

mean 0.9976 0.3138

std. dev. 0.0392 0.3316

weight sum 2093 572820

precision 0.0114 0.0114

cmp\_lname\_c2

mean 0.9599 0.1454

std. dev. 0.1766 0.1655

weight sum 53 186

precision 0.0417 0.0417

cmp\_sex

mean 0.9895 0.955

std. dev. 0.1667 0.2074

weight sum 2093 572820

precision 1 1

cmp\_bd

mean 0.9952 0.2219

std. dev. 0.1667 0.4156

weight sum 2093 572758

precision 1 1

cmp\_bm

mean 0.9976 0.4868

std. dev. 0.1667 0.4998

weight sum 2093 572758

precision 1 1

cmp\_by

mean 0.9962 0.2198

std. dev. 0.1667 0.4141

weight sum 2093 572758

precision 1 1

cmp\_plz

mean 0.9545 0.002

std. dev. 0.2083 0.1667

weight sum 2089 571529

precision 1 1

Time taken to build model: 1.26 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 2.52 seconds

=== Summary ===

Correctly Classified Instances 574877 99.9937 %

Incorrectly Classified Instances 36 0.0063 %

Kappa statistic 0.9913

Mean absolute error 0.0001

Root mean squared error 0.0082

Relative absolute error 1.4263 %

Root relative squared error 13.5578 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.983 0.000 1.000 0.983 0.991 0.991 0.999 0.997 TRUE

1.000 0.017 1.000 1.000 1.000 0.991 0.998 1.000 FALSE

Weighted Avg. 1.000 0.017 1.000 1.000 1.000 0.991 0.998 1.000

=== Confusion Matrix ===

a b <-- classified as

2057 36 | a = TRUE

0 572820 | b = FALSE

Wow, above 99.99%> for correctly classified items. There were only 36 incorrectly classified instances (false negatives). Kappa stat is also above 0.99 (BASICALLY IDENTIFOR LEVEL!). Extremely good even though still using a naive set (where some attributes should definitely be weighed significantly more)

### J48

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_1

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.875: FALSE (499198.52/1.0)

| cmp\_lname\_c1 > 0.875

| | cmp\_fname\_c1 <= 0.777778: FALSE (71294.94/1.97)

| | cmp\_fname\_c1 > 0.777778

| | | cmp\_bd <= 0: FALSE (1125.41)

| | | cmp\_bd > 0

| | | | cmp\_by <= 0: FALSE (38.02/0.02)

| | | | cmp\_by > 0: TRUE (97.0/1.0)

cmp\_plz > 0

| cmp\_bd <= 0

| | cmp\_by <= 0: FALSE (755.99/1.0)

| | cmp\_by > 0

| | | cmp\_lname\_c1 <= 0.909091: FALSE (187.2)

| | | cmp\_lname\_c1 > 0.909091

| | | | cmp\_fname\_c1 <= 0.625: FALSE (3.01)

| | | | cmp\_fname\_c1 > 0.625: TRUE (10.0/1.0)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.727273: FALSE (212.86/4.0)

| | cmp\_lname\_c1 > 0.727273

| | | cmp\_fname\_c1 <= 0.909091

| | | | cmp\_by <= 0: FALSE (10.03)

| | | | cmp\_by > 0: TRUE (18.01)

| | | cmp\_fname\_c1 > 0.909091: TRUE (1962.02)

Number of Leaves : 13

Size of the tree : 25

Time taken to build model: 9.78 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.43 seconds

=== Summary ===

Correctly Classified Instances 574903 99.9983 %

Incorrectly Classified Instances 10 0.0017 %

Kappa statistic 0.9976

Mean absolute error 0

Root mean squared error 0.0041

Relative absolute error 0.4696 %

Root relative squared error 6.8476 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.000 0.999 0.996 0.998 0.998 1.000 0.999 TRUE

1.000 0.004 1.000 1.000 1.000 0.998 1.000 1.000 FALSE

Weighted Avg. 1.000 0.004 1.000 1.000 1.000 0.998 1.000 1.000

=== Confusion Matrix ===

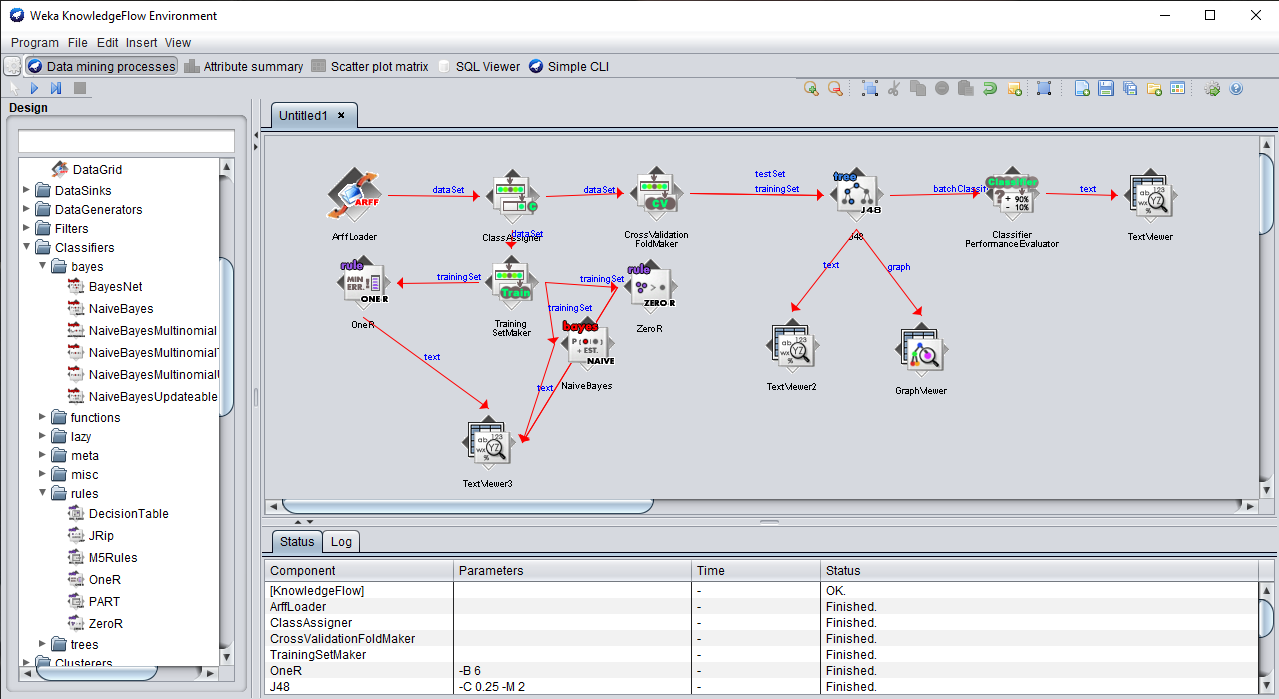
a b <-- classified as

2085 8 | a = TRUE

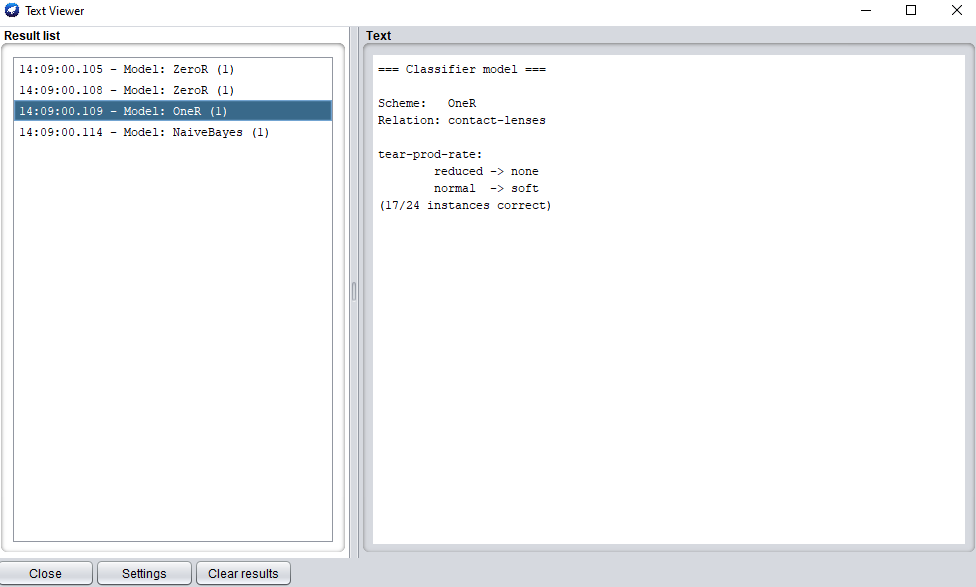
2 572818 | b = FALSE

Decision tree made here was best! There were only 10 items misclassified!!! (Kappa = 0.9976)

## Using KnowledgeFlow



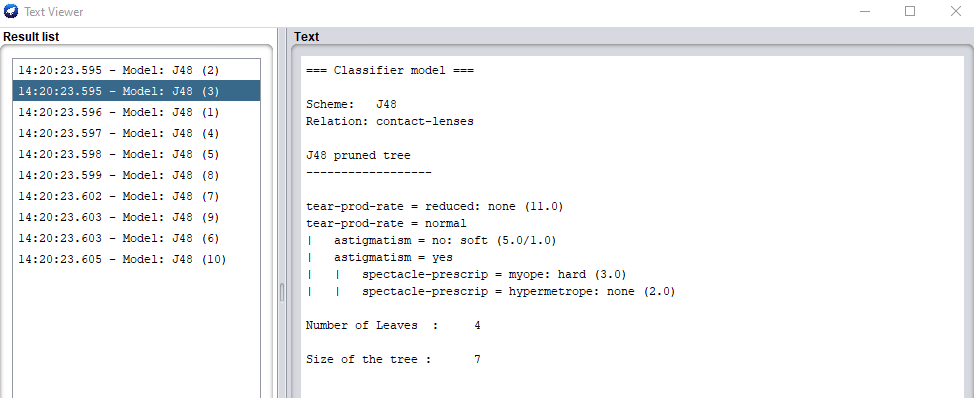
Okay this program is actually seriously cool. This way I don’t have to modify the test and can just do multiple at once. For example, I still do the KnowledgeFlow assignment here, but I added my training set maker, since I used the training set option on every classifier run (instead of the cross-validation)... With the training setmaker, I added OneR, ZeroR, and NaiveBayes and having them print it out in the text viewer:

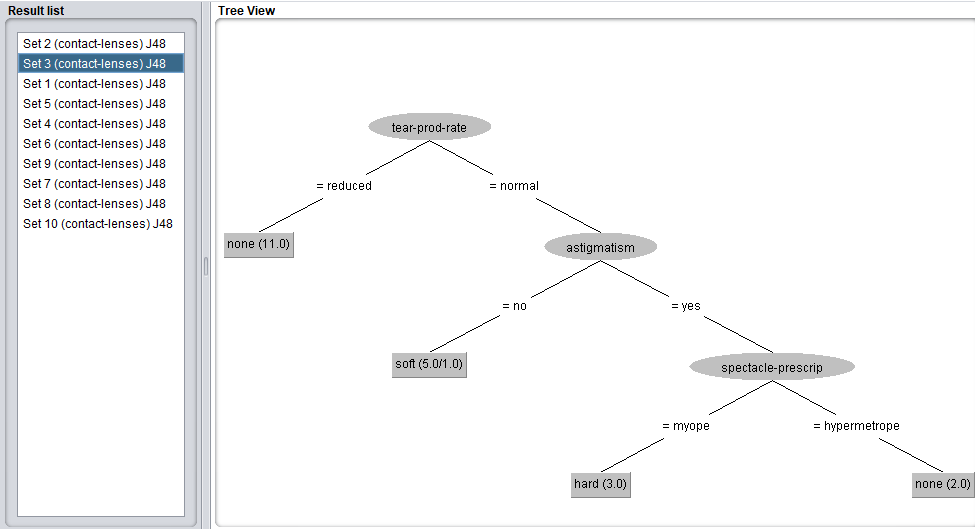


### Contact-lenses

Looking at the Cross validation effect on J48 is super interesting, with the data always breaking down into the same two trees:

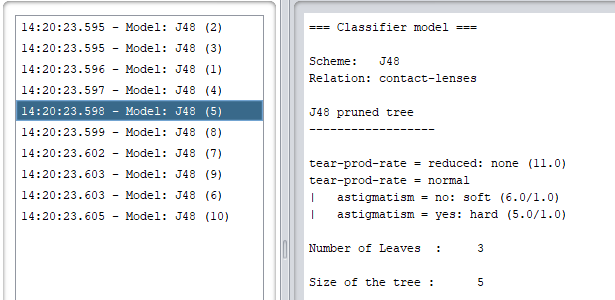
#### Tree v1:

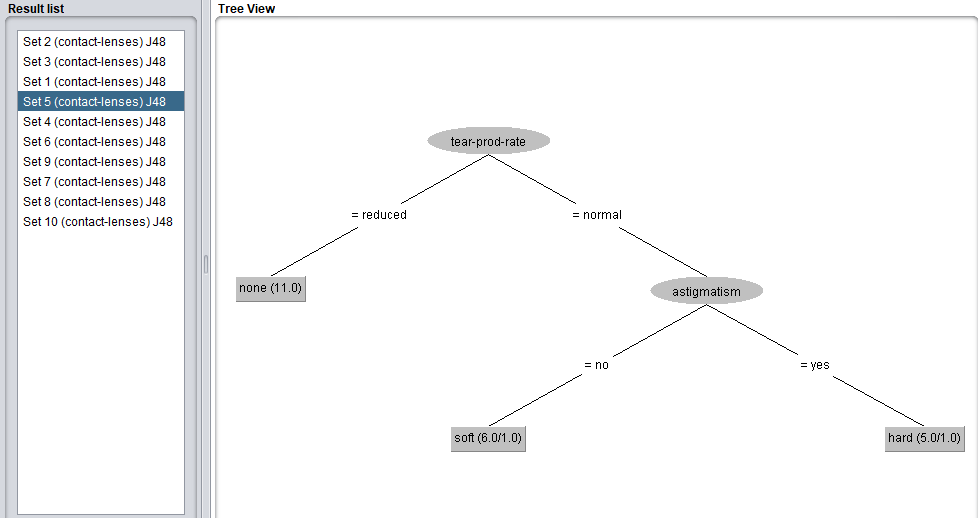




This top one works better due to the fact, it has more additional breakdown (less incorrect classification)

#### Tree v2:





This may have been a result of the sets being broken down differently each run, causing the training set to be different each time, which ends up revealing these two consistent decision trees.

#### 

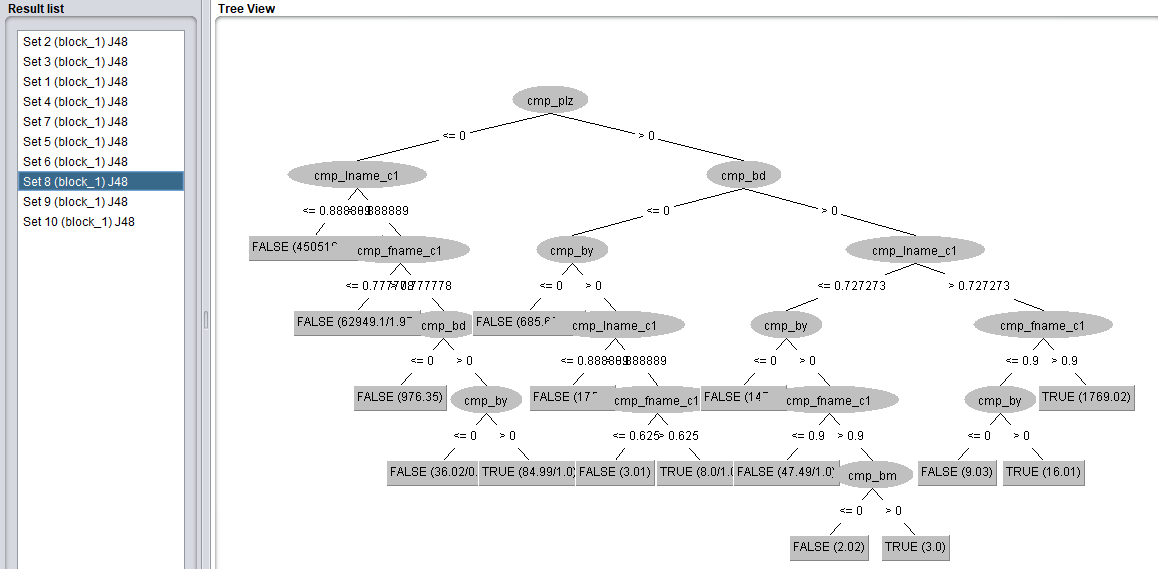
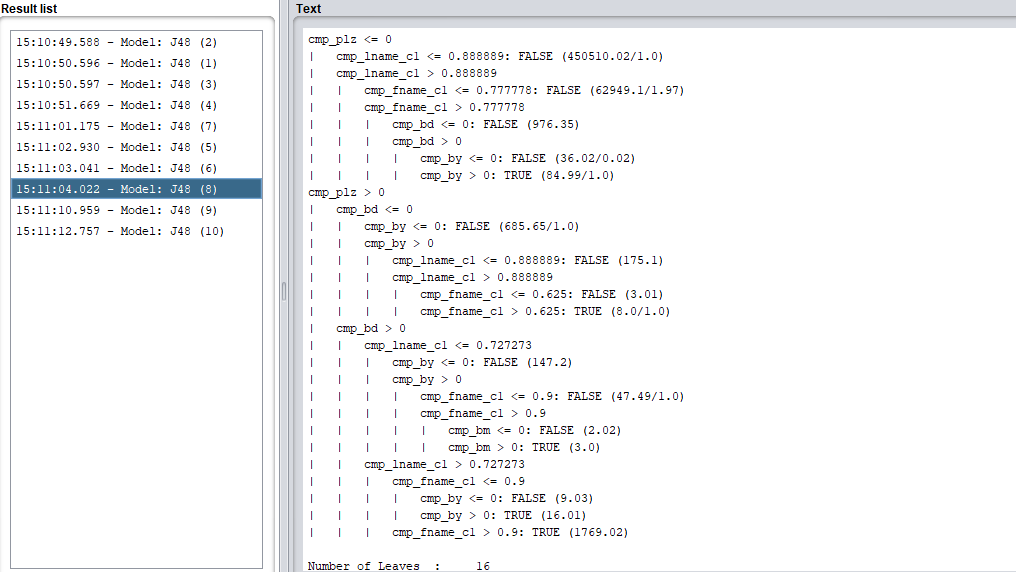
#### Performance Evaluation:

The resulting evaluation show that there was 4 incorrect instances, with 83.3% correct classification. Both ROC Area and Kappa says, this is better than guessing, but not a perfect identifor yet. Mostly mislabeled none instances with hard or soft. This must be due to decision tree v2: which stops at astigmatism, and doesn’t care about the spectacle-prescp attribute.

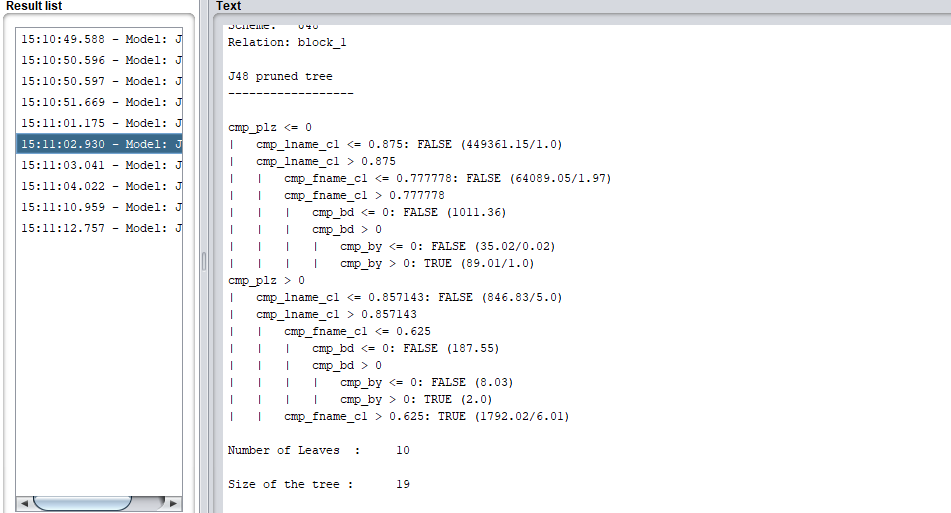
### Block\_1.csv

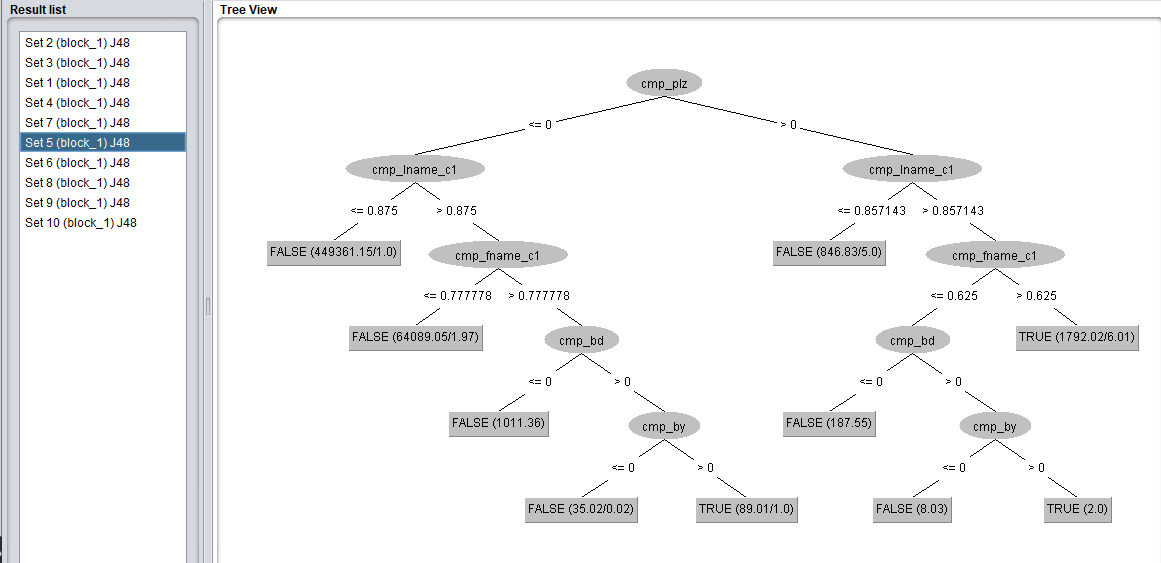
Okay, since there are a lot more data points, each tree it created was unique, the one I decided to showcase was the one with the largest tree (most unique training set) and smallest (most similar training set)

#### Largest Tree:



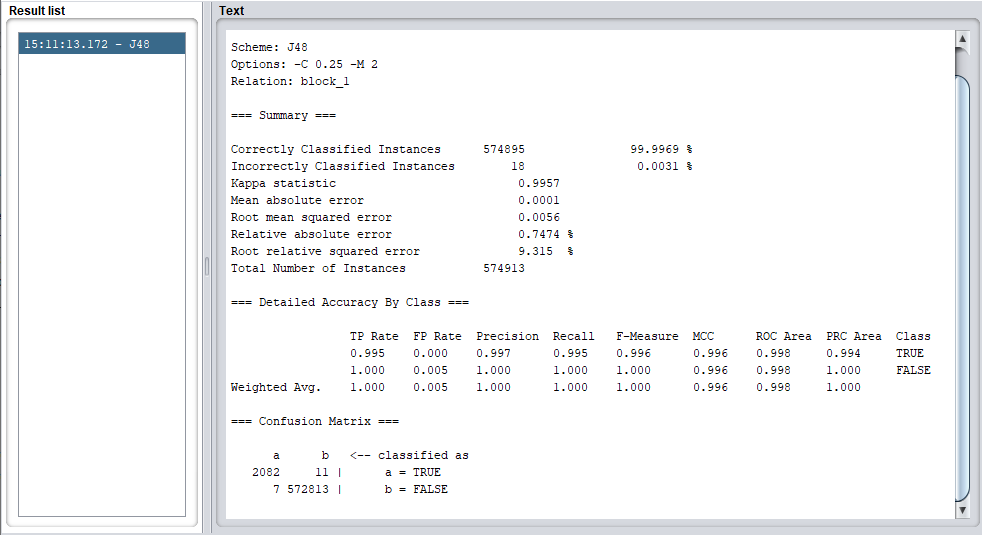
#### Smallest Tree:





The reason each comes out so unique is because the training set it chooses is different for each set. In this case, the training data it must have used was more kneighbors, if you will (more like one another). This will end up giving more errors when actually run.

#### Performance Evaluation:



Since this is the resulting overview of each iteration of the decision tree, we see that it actually has a higher incorrectly classified instance than when using weka explorer on block\_1. This increase may have been due to decision trees like the smallest, which would incorrectly label more than others. Nonetheless, we still have over 99.99% success on correctly classifying (kappa = 0.995)

# 

# Extension Writeup

Since we only tested our decision tree making algorithm on just the first block, I decided to run the same decision tree test (using training set as test) and the following was the results:

## Decision tree results

### Block 2

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_2

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.888889: FALSE (500387.63)

| cmp\_lname\_c1 > 0.888889

| | cmp\_fname\_c1 <= 0.909091: FALSE (70233.79)

| | cmp\_fname\_c1 > 0.909091

| | | cmp\_by <= 0: FALSE (1004.46/1.0)

| | | cmp\_by > 0

| | | | cmp\_bd <= 0: FALSE (27.07)

| | | | cmp\_bd > 0: TRUE (75.06/1.07)

cmp\_plz > 0

| cmp\_bd <= 0

| | cmp\_by <= 0: FALSE (761.18/1.24)

| | cmp\_by > 0

| | | cmp\_lname\_c1 <= 0.833333: FALSE (174.29)

| | | cmp\_lname\_c1 > 0.833333

| | | | cmp\_fname\_c1 <= 0.75: FALSE (5.01)

| | | | cmp\_fname\_c1 > 0.75: TRUE (5.06)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.636364: FALSE (222.71/3.0)

| | cmp\_lname\_c1 > 0.636364

| | | cmp\_fname\_c1 <= 0.909091

| | | | cmp\_by <= 0: FALSE (7.03)

| | | | cmp\_by > 0: TRUE (18.01/0.01)

| | | cmp\_fname\_c1 > 0.909091: TRUE (1991.71/1.0)

Number of Leaves : 13

Size of the tree : 25

Time taken to build model: 6.97 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.27 seconds

=== Summary ===

Correctly Classified Instances 574906 99.9988 %

Incorrectly Classified Instances 7 0.0012 %

Kappa statistic 0.9983

Mean absolute error 0

Root mean squared error 0.0035

Relative absolute error 0.3477 %

Root relative squared error 5.7958 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.998 0.000 0.999 0.998 0.998 0.998 1.000 0.999 TRUE

1.000 0.002 1.000 1.000 1.000 0.998 1.000 1.000 FALSE

Weighted Avg. 1.000 0.002 1.000 1.000 1.000 0.998 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2088 5 | a = TRUE

2 572818 | b = FALSE

### Block 3

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_3

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.875: FALSE (498664.73/1.0)

| cmp\_lname\_c1 > 0.875

| | cmp\_fname\_c1 <= 0.8: FALSE (71882.89)

| | cmp\_fname\_c1 > 0.8

| | | cmp\_bd <= 0: FALSE (1038.43/1.0)

| | | cmp\_bd > 0

| | | | cmp\_by <= 0: FALSE (23.02)

| | | | cmp\_by > 0: TRUE (82.04/0.07)

cmp\_plz > 0

| cmp\_bd <= 0

| | id\_2 <= 9168

| | | cmp\_lname\_c1 <= 0.818182: FALSE (17.0)

| | | cmp\_lname\_c1 > 0.818182

| | | | cmp\_bm <= 0: FALSE (4.0)

| | | | cmp\_bm > 0: TRUE (4.0)

| | id\_2 > 9168: FALSE (957.3/2.0)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.769231: FALSE (230.54/4.0)

| | cmp\_lname\_c1 > 0.769231: TRUE (2009.04/10.02)

Number of Leaves : 11

Size of the tree : 21

Time taken to build model: 9.27 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.91 seconds

=== Summary ===

Correctly Classified Instances 574895 99.9969 %

Incorrectly Classified Instances 18 0.0031 %

Kappa statistic 0.9957

Mean absolute error 0.0001

Root mean squared error 0.0056

Relative absolute error 0.8612 %

Root relative squared error 9.2597 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.000 0.995 0.996 0.996 0.996 1.000 0.994 TRUE

1.000 0.004 1.000 1.000 1.000 0.996 1.000 1.000 FALSE

Weighted Avg. 1.000 0.004 1.000 1.000 1.000 0.996 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2085 8 | a = TRUE

10 572810 | b = FALSE

### Block 4

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_4

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.888889: FALSE (500331.53/1.0)

| cmp\_lname\_c1 > 0.888889

| | cmp\_fname\_c1 <= 0.833333: FALSE (70291.85/0.98)

| | cmp\_fname\_c1 > 0.833333

| | | cmp\_bd <= 0: FALSE (1028.37)

| | | cmp\_bd > 0

| | | | cmp\_bm <= 0: FALSE (29.03)

| | | | cmp\_bm > 0

| | | | | cmp\_by <= 0: FALSE (6.01/1.0)

| | | | | cmp\_by > 0: TRUE (80.08/0.07)

cmp\_plz > 0

| cmp\_bd <= 0: FALSE (924.8/4.29)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.727273

| | | cmp\_by <= 0: FALSE (150.0)

| | | cmp\_by > 0

| | | | cmp\_fname\_c1 <= 0.8: FALSE (48.43/1.0)

| | | | cmp\_fname\_c1 > 0.8

| | | | | cmp\_bm <= 0: FALSE (4.02)

| | | | | cmp\_bm > 0: TRUE (4.14/0.14)

| | cmp\_lname\_c1 > 0.727273

| | | cmp\_fname\_c1 <= 0.857143

| | | | cmp\_by <= 0: FALSE (14.03)

| | | | cmp\_by > 0: TRUE (14.01/0.01)

| | | cmp\_fname\_c1 > 0.857143: TRUE (1986.71)

Number of Leaves : 14

Size of the tree : 27

Time taken to build model: 9.28 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.37 seconds

=== Summary ===

Correctly Classified Instances 574905 99.9986 %

Incorrectly Classified Instances 8 0.0014 %

Kappa statistic 0.9981

Mean absolute error 0

Root mean squared error 0.0037

Relative absolute error 0.3972 %

Root relative squared error 6.1483 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.996 0.000 1.000 0.996 0.998 0.998 1.000 0.999 TRUE

1.000 0.004 1.000 1.000 1.000 0.998 1.000 1.000 FALSE

Weighted Avg. 1.000 0.004 1.000 1.000 1.000 0.998 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2085 8 | a = TRUE

0 572820 | b = FALSE

### Block 5

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_5

Instances: 574914

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.888889: FALSE (500866.49)

| cmp\_lname\_c1 > 0.888889

| | cmp\_fname\_c1 <= 0.8: FALSE (69693.94/1.0)

| | cmp\_fname\_c1 > 0.8

| | | cmp\_by <= 0: FALSE (1034.37/1.0)

| | | cmp\_by > 0

| | | | cmp\_bm <= 0: FALSE (30.03)

| | | | cmp\_bm > 0: TRUE (88.07/2.08)

cmp\_plz > 0

| cmp\_bd <= 0: FALSE (951.75/4.3)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.777778

| | | cmp\_by <= 0: FALSE (179.24)

| | | cmp\_by > 0

| | | | cmp\_fname\_c1 <= 0.75: FALSE (56.33/1.0)

| | | | cmp\_fname\_c1 > 0.75

| | | | | cmp\_bm <= 0: FALSE (5.03)

| | | | | cmp\_bm > 0: TRUE (9.0/1.0)

| | cmp\_lname\_c1 > 0.777778: TRUE (1999.74/7.03)

Number of Leaves : 11

Size of the tree : 21

Time taken to build model: 13.42 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.48 seconds

=== Summary ===

Correctly Classified Instances 574897 99.997 %

Incorrectly Classified Instances 17 0.003 %

Kappa statistic 0.9959

Mean absolute error 0.0001

Root mean squared error 0.0054

Relative absolute error 0.8232 %

Root relative squared error 8.9943 %

Total Number of Instances 574914

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.997 0.000 0.995 0.997 0.996 0.996 1.000 0.995 TRUE

1.000 0.003 1.000 1.000 1.000 0.996 1.000 1.000 FALSE

Weighted Avg. 1.000 0.003 1.000 1.000 1.000 0.996 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2087 7 | a = TRUE

10 572810 | b = FALSE

### Block 6

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_6

Instances: 574914

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.888889: FALSE (500544.84/2.0)

| cmp\_lname\_c1 > 0.888889

| | cmp\_fname\_c1 <= 0.714286: FALSE (69917.83/1.96)

| | cmp\_fname\_c1 > 0.714286

| | | cmp\_bd <= 0: FALSE (1184.43/1.0)

| | | cmp\_bd > 0

| | | | cmp\_by <= 0: FALSE (39.08/2.0)

| | | | cmp\_by > 0: TRUE (64.13/1.1)

cmp\_plz > 0

| cmp\_bd <= 0

| | cmp\_lname\_c1 <= 0.928571: FALSE (674.54)

| | cmp\_lname\_c1 > 0.928571

| | | cmp\_fname\_c1 <= 0.875: FALSE (237.5)

| | | cmp\_fname\_c1 > 0.875

| | | | cmp\_by <= 0

| | | | | id\_1 <= 12692: TRUE (3.01/1.01)

| | | | | id\_1 > 12692: FALSE (6.0)

| | | | cmp\_by > 0: TRUE (5.0/1.0)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.727273

| | | cmp\_by <= 0: FALSE (159.18)

| | | cmp\_by > 0

| | | | cmp\_fname\_c1 <= 0.777778: FALSE (49.41)

| | | | cmp\_fname\_c1 > 0.777778

| | | | | cmp\_bm <= 0: FALSE (4.03)

| | | | | cmp\_bm > 0: TRUE (5.0)

| | cmp\_lname\_c1 > 0.727273

| | | cmp\_fname\_c1 <= 0.888889

| | | | cmp\_by <= 0: FALSE (8.03)

| | | | cmp\_by > 0: TRUE (17.0)

| | | cmp\_fname\_c1 > 0.888889: TRUE (1995.01)

Number of Leaves : 17

Size of the tree : 33

Time taken to build model: 9.08 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.39 seconds

=== Summary ===

Correctly Classified Instances 574904 99.9983 %

Incorrectly Classified Instances 10 0.0017 %

Kappa statistic 0.9976

Mean absolute error 0

Root mean squared error 0.004

Relative absolute error 0.4512 %

Root relative squared error 6.6718 %

Total Number of Instances 574914

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.997 0.000 0.999 0.997 0.998 0.998 0.999 0.999 TRUE

1.000 0.003 1.000 1.000 1.000 0.998 0.999 1.000 FALSE

Weighted Avg. 1.000 0.003 1.000 1.000 1.000 0.998 0.999 1.000

=== Confusion Matrix ===

a b <-- classified as

2086 7 | a = TRUE

3 572818 | b = FALSE

### Block 7

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_7

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.857143: FALSE (496572.18)

| cmp\_lname\_c1 > 0.857143

| | cmp\_fname\_c1 <= 0.909091: FALSE (73968.01/1.0)

| | cmp\_fname\_c1 > 0.909091

| | | cmp\_bd <= 0: FALSE (1038.34)

| | | cmp\_bd > 0

| | | | cmp\_bm <= 0: FALSE (27.03)

| | | | cmp\_bm > 0: TRUE (99.99/2.0)

cmp\_plz > 0

| cmp\_bd <= 0: FALSE (967.75/3.0)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.714286: FALSE (242.66/2.0)

| | cmp\_lname\_c1 > 0.714286

| | | cmp\_fname\_c1 <= 0.875

| | | | cmp\_by <= 0: FALSE (7.02)

| | | | cmp\_by > 0: TRUE (21.02/1.01)

| | | cmp\_fname\_c1 > 0.875: TRUE (1969.0)

Number of Leaves : 10

Size of the tree : 19

Time taken to build model: 22.32 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.36 seconds

=== Summary ===

Correctly Classified Instances 574904 99.9984 %

Incorrectly Classified Instances 9 0.0016 %

Kappa statistic 0.9978

Mean absolute error 0

Root mean squared error 0.0039

Relative absolute error 0.4263 %

Root relative squared error 6.5279 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.997 0.000 0.999 0.997 0.998 0.998 1.000 0.999 TRUE

1.000 0.003 1.000 1.000 1.000 0.998 1.000 1.000 FALSE

Weighted Avg. 1.000 0.003 1.000 1.000 1.000 0.998 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2087 6 | a = TRUE

3 572817 | b = FALSE

### Block 8

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_8

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.857143: FALSE (496000.34)

| cmp\_lname\_c1 > 0.857143

| | cmp\_fname\_c1 <= 0.833333: FALSE (74517.87/0.98)

| | cmp\_fname\_c1 > 0.833333

| | | cmp\_bd <= 0: FALSE (1118.48/1.0)

| | | cmp\_bd > 0

| | | | cmp\_by <= 0: FALSE (27.05)

| | | | cmp\_by > 0: TRUE (95.98/3.0)

cmp\_plz > 0

| cmp\_bd <= 0: FALSE (935.91/3.3)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.75

| | | cmp\_bm <= 0: FALSE (161.05)

| | | cmp\_bm > 0

| | | | cmp\_fname\_c1 <= 0.666667: FALSE (50.44/1.0)

| | | | cmp\_fname\_c1 > 0.666667

| | | | | cmp\_by <= 0: FALSE (7.11/1.0)

| | | | | cmp\_by > 0: TRUE (4.01/0.01)

| | cmp\_lname\_c1 > 0.75: TRUE (1994.76/6.02)

Number of Leaves : 11

Size of the tree : 21

Time taken to build model: 13.51 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.34 seconds

=== Summary ===

Correctly Classified Instances 574897 99.9972 %

Incorrectly Classified Instances 16 0.0028 %

Kappa statistic 0.9962

Mean absolute error 0.0001

Root mean squared error 0.0052

Relative absolute error 0.7684 %

Root relative squared error 8.6967 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.997 0.000 0.996 0.997 0.996 0.996 1.000 0.996 TRUE

1.000 0.003 1.000 1.000 1.000 0.996 1.000 1.000 FALSE

Weighted Avg. 1.000 0.003 1.000 1.000 1.000 0.996 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2086 7 | a = TRUE

9 572811 | b = FALSE

### Block 9

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_9

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.875: FALSE (498757.43/1.0)

| cmp\_lname\_c1 > 0.875

| | cmp\_fname\_c1 <= 0.857143: FALSE (71690.93)

| | cmp\_fname\_c1 > 0.857143

| | | cmp\_by <= 0: FALSE (1099.41/1.0)

| | | cmp\_by > 0

| | | | cmp\_bd <= 0: FALSE (41.02)

| | | | cmp\_bd > 0: TRUE (107.97/1.0)

cmp\_plz > 0

| cmp\_bd <= 0

| | cmp\_lname\_c1 <= 0.928571: FALSE (748.91)

| | cmp\_lname\_c1 > 0.928571

| | | cmp\_fname\_c1 <= 0.4: FALSE (236.89)

| | | cmp\_fname\_c1 > 0.4

| | | | cmp\_bm <= 0: FALSE (11.22/0.19)

| | | | cmp\_bm > 0: TRUE (7.12/1.0)

| cmp\_bd > 0

| | cmp\_lname\_c1 <= 0.75

| | | cmp\_by <= 0: FALSE (165.27)

| | | cmp\_by > 0

| | | | cmp\_fname\_c1 <= 0.4: FALSE (50.35)

| | | | cmp\_fname\_c1 > 0.4

| | | | | cmp\_bm <= 0: FALSE (3.02)

| | | | | cmp\_bm > 0: TRUE (6.04/1.04)

| | cmp\_lname\_c1 > 0.75

| | | cmp\_fname\_c1 <= 0.857143

| | | | cmp\_bm <= 0: FALSE (13.37)

| | | | cmp\_bm > 0: TRUE (12.34/1.33)

| | | cmp\_fname\_c1 > 0.857143: TRUE (1961.71)

Number of Leaves : 16

Size of the tree : 31

Time taken to build model: 18.48 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.58 seconds

=== Summary ===

Correctly Classified Instances 574907 99.999 %

Incorrectly Classified Instances 6 0.001 %

Kappa statistic 0.9986

Mean absolute error 0

Root mean squared error 0.0032

Relative absolute error 0.2919 %

Root relative squared error 5.2448 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.999 0.000 0.998 0.999 0.999 0.999 1.000 0.999 TRUE

1.000 0.001 1.000 1.000 1.000 0.999 1.000 1.000 FALSE

Weighted Avg. 1.000 0.001 1.000 1.000 1.000 0.999 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

2091 2 | a = TRUE

4 572816 | b = FALSE

### Block 10

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: block\_10

Instances: 574913

Attributes: 12

id\_1

id\_2

cmp\_fname\_c1

cmp\_fname\_c2

cmp\_lname\_c1

cmp\_lname\_c2

cmp\_sex

cmp\_bd

cmp\_bm

cmp\_by

cmp\_plz

is\_match

Test mode: evaluate on training data

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

J48 pruned tree

------------------

cmp\_plz <= 0

| cmp\_lname\_c1 <= 0.888889: FALSE (500355.5/2.0)

| cmp\_lname\_c1 > 0.888889

| | cmp\_fname\_c1 <= 0.833333: FALSE (70229.85)

| | cmp\_fname\_c1 > 0.833333

| | | cmp\_bd <= 0: FALSE (1069.46/1.0)

| | | cmp\_bd > 0

| | | | cmp\_by <= 0

| | | | | cmp\_bm <= 0: FALSE (27.02)

| | | | | cmp\_bm > 0

| | | | | | id\_1 <= 9572: TRUE (2.0)

| | | | | | id\_1 > 9572: FALSE (7.01)

| | | | cmp\_by > 0: TRUE (91.07/0.08)

cmp\_plz > 0

| cmp\_by <= 0

| | cmp\_bd <= 0: FALSE (707.19/1.23)

| | cmp\_bd > 0

| | | cmp\_bm <= 0: FALSE (173.26/0.06)

| | | cmp\_bm > 0

| | | | cmp\_lname\_c1 <= 0.727273: FALSE (15.1)

| | | | cmp\_lname\_c1 > 0.727273

| | | | | cmp\_fname\_c1 <= 0.714286: FALSE (3.0)

| | | | | cmp\_fname\_c1 > 0.714286: TRUE (7.01)

| cmp\_by > 0

| | cmp\_lname\_c1 <= 0.666667

| | | cmp\_bd <= 0: FALSE (185.24)

| | | cmp\_bd > 0

| | | | cmp\_fname\_c1 <= 0.75: FALSE (40.51)

| | | | cmp\_fname\_c1 > 0.75

| | | | | cmp\_bm <= 0: FALSE (4.02)

| | | | | cmp\_bm > 0: TRUE (5.01/0.01)

| | cmp\_lname\_c1 > 0.666667: TRUE (1990.75/7.03)

Number of Leaves : 17

Size of the tree : 33

Time taken to build model: 9.35 seconds

=== Evaluation on training set ===

Time taken to test model on training data: 0.4 seconds

=== Summary ===

Correctly Classified Instances 574902 99.9981 %

Incorrectly Classified Instances 11 0.0019 %

Kappa statistic 0.9974

Mean absolute error 0

Root mean squared error 0.0044

Relative absolute error 0.545 %

Root relative squared error 7.2824 %

Total Number of Instances 574913

=== Detailed Accuracy By Class ===

TP Rate FP Rate Precision Recall F-Measure MCC ROC Area PRC Area Class

0.998 0.000 0.997 0.998 0.997 0.997 1.000 0.996 TRUE

1.000 0.002 1.000 1.000 1.000 0.997 1.000 1.000 FALSE

Weighted Avg. 1.000 0.002 1.000 1.000 1.000 0.997 1.000 1.000

=== Confusion Matrix ===

a b <-- classified as

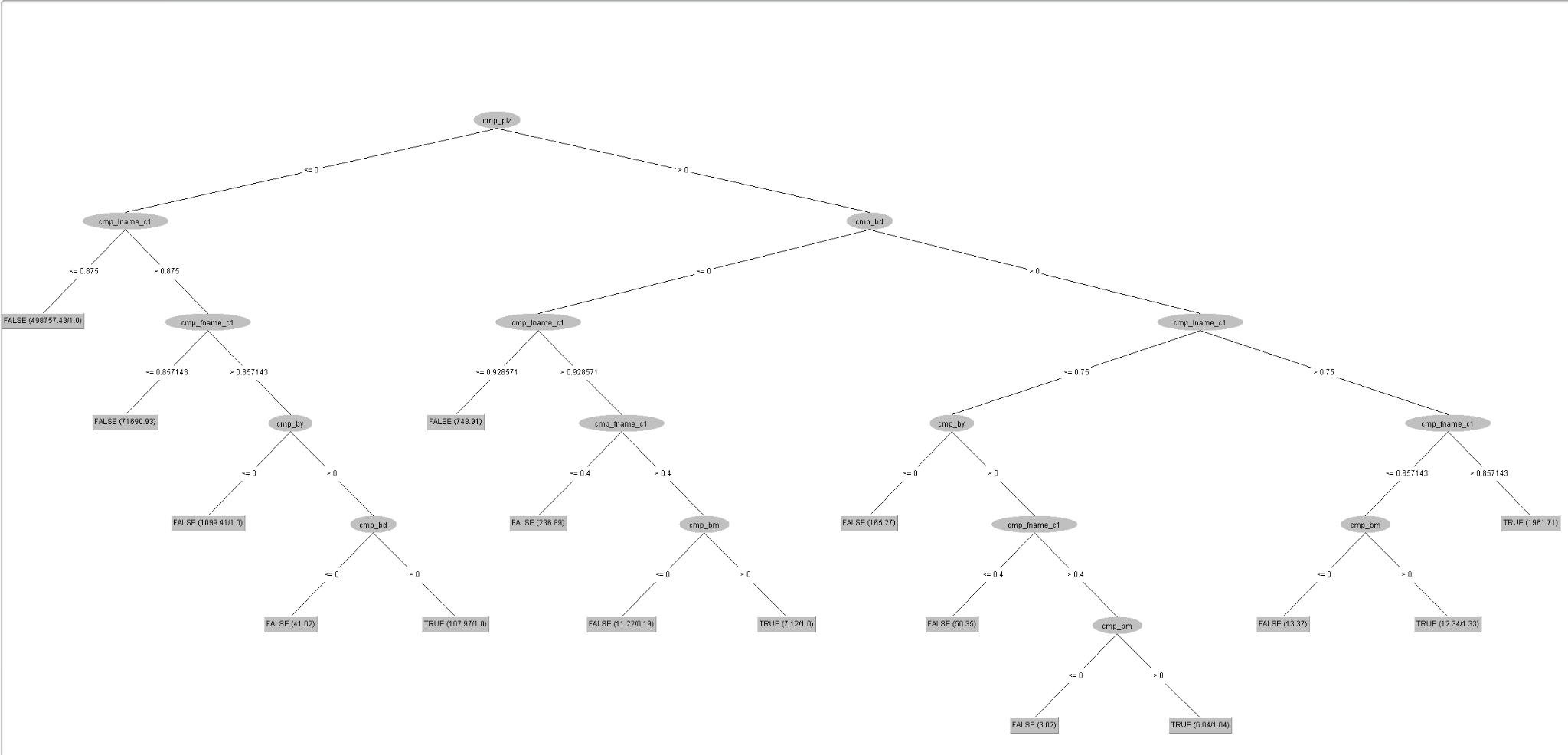
2089 4 | a = TRUE

7 572813 | b = FALSE

## What did I learn:

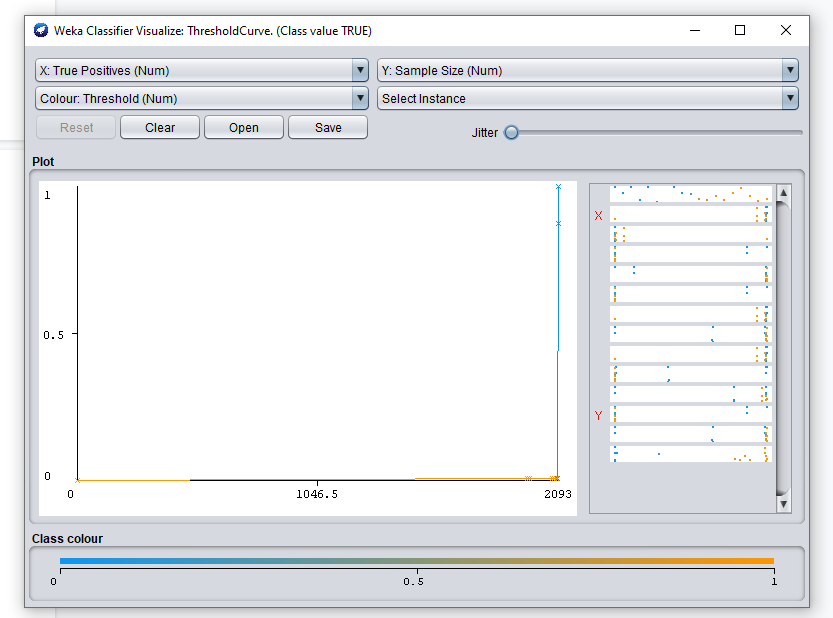
Each block has around 574,913 instances

The best decision tree came from block 9:



With almost a 99.99% classification accuracy. That being said, I tried to research for a while how to apply the created decision tree to another block. Sadly I wasn’t capable of doing so. Since this week only allowed us to do one page of PPT for our ME, I decided the top was far more than enough (given the computer runtime and me using knowledge flow incorrectly).

For personal reasons, I then tried to mess with the visualizing the threshold curve on the decision tree produced from block 9. After clicking through each x and y option, I came to the realization that I still have no idea what it means. Even doing a graph where sample size is y and true positive rate is x, I still have no idea what the graph demonstrates.



What I do know is jitter adds more random noise to the data, so its more interpretable

