Congressional Voting

1.) ZeroR: The target concept is to predict whether the instance (congressperson) is a democrat or not based on the attributes which are voting records. It took 1/100th of a second to build the model, and the ZeroR algorithm was 61% accurate, and 38% inaccurate over 435 instances.

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

ZeroR predicts class value: democrat

Time taken to build model: 0.01 seconds

=== Summary ===

Correctly Classified Instances 267 61.3793 %

Incorrectly Classified Instances 168 38.6207 %

Kappa statistic 0

Mean absolute error 0.4742

Root mean squared error 0.4869

Relative absolute error 100 %

Root relative squared error 100 %

Total Number of Instances 435

2.) Bayes – BayesNet: Looks like the BayesNet algorithm built a tree structure to test the party of each instance. This was far more accurate than the ZeroR algorithm, at 90% accurate, and 9.8% inaccurate results. It also took less time to build the model at 1/500th of a second vs 1/100th of a second.

Time taken to build model: 0.05 seconds

=== Summary ===

Correctly Classified Instances 392 90.1149 %

Incorrectly Classified Instances 43 9.8851 %

Kappa statistic 0.7949

Mean absolute error 0.1005

Root mean squared error 0.3012

Relative absolute error 21.199 %

Root relative squared error 61.8661 %

Total Number of Instances 435

3.) Function – Logistic: I kept some of the training set because it seems to show how important each attribute is to predicting the party. For instance, yes votes for duty free exports and synfuels corp cutback seem to suggest that democrats voted largely for these issues. Which makes sense because voting yes for no tax on exports and what I assume is reduced oil corp taxes aligns more strongly with republican voters. This algorithm was even more accurate than the BayesNet algorithm, at 96% accuracy and almost 4% inaccuracy. It also took less time to build the model than BayesNet at 1/700th of a second.

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

Odds Ratios...

Class

Variable democrat

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

handicapped-infants=y 1.0348

water-project-cost-sharing=y 1.8034

adoption-of-the-budget-resolution=y 27.9402

physician-fee-freeze=y 0.0008

el-salvador-aid=y 0.0951

religious-groups-in-schools=y 1.3812

anti-satellite-test-ban=y 0.1871

aid-to-nicaraguan-contras=y 0.0915

mx-missile=y 4.5037

immigration=y 0.0889

synfuels-corporation-cutback=y 53.1713

education-spending=y 0.5792

superfund-right-to-sue=y 0.6712

crime=y 0.9418

duty-free-exports=y 10.3873

export-administration-act-south-africa=y 0.4943

Time taken to build model: 0.07 seconds

=== Summary ===

Correctly Classified Instances 418 96.092 %

Incorrectly Classified Instances 17 3.908 %

Kappa statistic 0.9177

Mean absolute error 0.0496

Root mean squared error 0.1751

Relative absolute error 10.4549 %

Root relative squared error 35.9553 %

Total Number of Instances 435

4.) Meta – Bagging: This algorithm was highly accurate as well, at almost 96% accurate and just over 4% inaccurate. So it is less accurate than the Logistic algorithm, it also took longer to build the model at almost 1/5th of a second. I kept the training set section because what I believe is that it runs the model 10 times (iterations) against a base iteration, and then averages the results? That would explain why it took a bit longer.

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

Bagging with 10 iterations and base learner

Time taken to build model: 0.16 seconds

=== Summary ===

Correctly Classified Instances 416 95.6322 %

Incorrectly Classified Instances 19 4.3678 %

Kappa statistic 0.9082

Mean absolute error 0.0698

Root mean squared error 0.1739

Relative absolute error 14.7117 %

Root relative squared error 35.7124 %

Total Number of Instances 435

5.) Rules – OneR: This model took a bit less time than the Bagging model at 1/10th of a second, and interestingly had the exact same accuracy at 95.6322% and inaccuracy at 4.3678%. So these two so far are the most accurate albeit slowest models.

Time taken to build model: 0.01 seconds

=== Summary ===

Correctly Classified Instances 416 95.6322 %

Incorrectly Classified Instances 19 4.3678 %

Kappa statistic 0.9088

Mean absolute error 0.0437

Root mean squared error 0.209

Relative absolute error 9.21 %

Root relative squared error 42.9237 %

Total Number of Instances 435

6.) Trees - Random Forest: So this model took the longest of all the algorithms used, at just over 1/3 of a second. It was also the second most accurate, tying with the logistic function at 96.092% accurate and 3.908 inaccurate. It also seemed to use a similar technique to the Meta Bagging algorithm by using what I assume is a base case, against 100 iterations instead of the 10 iterations that the Meta Bagging model used.

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

Bagging with 100 iterations and base learner

Time taken to build model: 0.35 seconds

=== Summary ===

Correctly Classified Instances 418 96.092 %

Incorrectly Classified Instances 17 3.908 %

Kappa statistic 0.9175

Mean absolute error 0.0714

Root mean squared error 0.1742

Relative absolute error 15.0587 %

Root relative squared error 35.7776 %

Total Number of Instances 435

Summary: The most accurate model I used was the Logistic function algorithm, at just over 96% accuracy, and the least accurate was the ZeroR algorithm at 61% accuracy. The fastest model was also the most accurate, at 1/700th of a second. The Logistic function was the ‘best’ model used, and for me it was also the easiest to understand because of the weighted attribute values. It seems to suggest that certain attribute values are more important than others in predicting the congresspersons party. For instance the “synfuels-corporation-cutback” attribute has a weighted value of 53.1713, versus ‘physician-fee-freeze’ at 0.0008 which seems to suggest that voting yes on oil corporation tax cutbacks was a more frequent indicator of republicans (which is probably true), than freezing physician fees, which at close to 0 suggests that this was an issue that was probably 50/50 and not a strong indicator of either party.