Project 2

Nick Greenquist Alex Hedges

May 5, 2017

A total of 210 experiments were run.

1 Observation of CSV Results

Dataset	AdaBoost.NC	C4.5	Chi-RW	FURIA	GAssist-Intervalar	GFS-GCCL	Ripper
abalone	10.6	75.2	0.1	24.2	25.3	24.9	46.2
australian	99.5	93.3	91.3	89.3	90.7	75.5	92.4
balance	49.0	89.9	91.4	88.7	87.0	88.6	60.1
bupa	88.9	85.9	59.9	78.3	79.3	60.0	86.0
car	90.4	95.8	97.7	97.1	90.7	70.0	94.7
chess	99.6	99.6	100.0	99.7	97.6	0.0	99.9
coil2000	95.9	94.2	97.5	94.1	94.1	46.0	94.4
contraceptive	47.8	72.5	51.9	55.9	57.8	43.8	63.2
ecoli	65.0	91.7	75.8	90.4	83.0	67.9	92.6
flare	53.5	78.6	45.1	75.7	77.0	31.1	75.4
german	100.0	84.8	99.4	76.8	81.6	70.4	86.6
glass	66.6	93.7	66.0	86.3	71.4	68.5	90.8
haberman	79.4	75.9	74.3	76.0	82.1	73.9	56.9
heart	93.0	91.6	97.2	88.9	92.1	89.4	90.2
ionosphere	99.8	98.7	97.7	97.2	98.3	92.6	97.6
iris	66.7	98.0	93.8	98.0	98.2	95.8	99.3
led7digit	14.3	77.1	74.9	76.3	69.3	77.6	54.1
lymphography	56.1	92.3	100.0	93.2	95.0	73.3	95.0
monk-2	100.0	100.0	97.2	100.0	98.7	97.2	100.0
newthyroid	30.2	98.4	86.0	99.5	96.8	87.4	99.5
pima	91.9	83.8	75.6	79.2	81.1	69.9	85.0
ring	99.9	98.7	58.3	99.0	95.3	90.5	95.5
saheart	91.8	78.6	78.5	75.3	81.6	68.1	86.3
splice	47.9	96.2	99.9	99.5	91.8	0.0	97.0
thyroid	7.4	99.9	93.0	99.9	97.9	92.6	99.8
tic-tac-toe	97.1	93.0	100.0	99.7	96.8	65.3	99.8
titanic	78.6	79.1	78.3	78.5	79.0	77.6	64.2
vehicle	48.4	90.6	65.9	79.8	67.4	62.0	87.9
vowel	17.9	97.1	55.3	96.4	43.2	60.2	94.2
wine	73.0	98.9	98.8	99.3	99.2	97.7	99.8

Table 1: Training Results

Dataset	AdaBoost.NC	C4.5	Chi-RW	FURIA	GAssist-Intervalar	GFS-GCCL	Ripper
abalone	9.3	20.4	0.0	21.7	24.1	24.1	23.5
australian	86.5	85.2	79.9	85.4	85.4	73.6	84.2
balance	41.4	76.8	89.6	82.9	81.0	81.9	50.9
bupa	70.2	67.0	57.9	68.4	63.8	58.5	59.3
car	87.4	91.5	77.8	93.0	90.3	70.0	89.9
chess	99.4	99.4	0.0	99.5	97.6	0.0	99.3
coil2000	93.6	93.9	75.1	94.0	94.0	46.6	93.0
contraceptive	39.8	52.7	39.9	54.2	53.2	43.4	51.3
ecoli	61.6	79.5	72.0	79.8	77.7	65.5	74.7
flare	53.5	74.3	38.5	74.8	73.2	31.1	67.6
german	65.9	71.1	19.6	73.8	73.2	70.2	66.1
glass	54.9	67.4	60.0	70.5	66.2	63.2	66.3
haberman	72.5	73.2	73.2	72.2	71.2	73.2	46.7
heart	79.6	78.1	51.9	81.5	80.4	78.9	76.3
ionosphere	92.6	90.9	65.5	91.8	86.6	88.3	86.1
iris	66.0	96.0	92.7	95.3	93.3	95.3	94.7
led7digit	13.4	71.0	63.6	71.4	64.8	71.0	48.8
lymphography	54.2	74.3	12.3	79.3	82.5	67.2	77.8
monk-2	100.0	100.0	42.9	100.0	98.0	97.3	100.0
newthyroid	28.8	92.1	84.7	96.3	91.7	86.1	94.5
pima	74.1	74.2	72.5	73.8	76.3	68.6	70.8
ring	94.7	90.6	55.8	93.6	93.8	91.1	87.7
saheart	67.8	68.4	72.7	70.1	68.2	65.4	57.8
splice	46.0	94.1	10.5	95.0	90.6	0.0	93.4
thyroid	7.4	99.6	92.0	99.7	97.8	92.6	99.5
tic-tac-toe	88.6	84.5	0.0	98.1	94.1	65.3	97.6
titanic	78.2	79.1	78.3	78.4	78.8	77.6	63.6
vehicle	45.6	74.7	60.8	70.2	62.7	57.2	68.7
vowel	17.3	81.5	49.9	80.1	39.0	54.6	76.5
wine	71.4	94.9	93.8	97.2	90.4	91.0	93.2

Table 2: Testing Results

1.1 Train vs. Test

Overall, train had better results than test, with only 6 experiments having test results equal to training results and 10 experiments having test results better than training results. This is almost always expected in classification because algorithms are usually much stronger classifying data it has seen before.

1.2 Dataset Observations

Certain datasets were classified very well while others were not. Most fit somewhere in the middle. A few datasets should be discussed. The first is abalone, which had horrible classification testing results from all algorithms. For training, results were awful from all algorithms except C4.5. Looking at the data set, there are many different features. Our theory is that there are too many features and the unneeded ones are interfering with the results. C4.5 must be an algorithm that handles features in a unique way compared to the other algorithms that all had poor results.

coil2000 was a dataset that caused its experiments to have drastically larger runtimes than all other datasets. coil2000 has over 80 attributes for each input. Surprisingly, coil2000 actually had some of the best results for both train and test. This leads to a hypothesis that having numerous inputs does not eventually lead to worse results. With abalone,we first thought it was the number of attributes that led to poor performance (it was the second slowest dataset to run experiments on). However, we can now change our idea to be that unneeded features are the problem.

1.3 Algorithm Observations

All algorithms had a mean performance better than 65% on training and 55% on testing.

For training, 2 results were 0s and 7 were 100s, showing perfectly incorrect and correct classifications, respectively. For testing, 4 results were 0s and 4 were 100s. We took the difference between the training and test results for the same experiment. Only 2 were 100s (which shows extreme overfitting), and 6 were 0s (showing no performance difference).

AdaBoost.NC was by far the algorithm that took the longest to run experiments with. Therefore, one would think that its results would be accurate. Unfortunately, AdaBoost.NC consistently had poor performance on many datasets compared to other algorithms. AdaBoost.NC had many results below 30%. These results would be less shocking if all the other algorithms returned similar numbers; however, AdaBoost.NC would consistently classify datasets with accuracy below 60% while the other algorithms consistently classified results with accuracy over 70%.

2 Statistical Analysis

2.1 Friedman 1xN

2.1.1 Train 1xN

Algorithm	Ranking
AdaBoost.NC	4.4
C4.5	2.75
Chi-RW	4.1667
FURIA	3.6667
GAssist-Intervalar	3.9
GFS-GCCL	5.9667
Ripper	3.15

Table 3: Average training rankings of the algorithms (Friedman)

i	algorithm	$z = (R_0 - R_i)/SE$	p	Holm
6	GFS-GCCL	5.766978	0	0.008333
5	AdaBoost.NC	2.958191	0.003095	0.01
4	Chi-RW	2.539861	0.01109	0.0125
3	GAssist-Intervalar	2.061769	0.03923	0.016667
2	FURIA	1.643439	0.100292	0.025
1	Ripper	0.717137	0.473289	0.05

Table 4: Post Hoc comparison Table for training with $\alpha = 0.05$ (FRIEDMAN)

1. Average Rankings
   (a) C4.5 is ranked lowest.
   (b) GFS-GCCL is ranked highest.
2. Post-Hoc Comparison (Holm)
   (a) C4.5 is withheld from this list as per Holms procedure to compare other algorithms to.
   (b) Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.016667.
   (c) The null hypothesis for GFS-GCCL, AdaBoost-NC, and Chi-RW can be rejected, as they have p-values ≤ 0.016667. However, the null hypotheses for FURIA, Ripper, and GAssist-Intervalar cannot be rejected.

2.1.2 Test 1xN

Algorithm	Ranking
AdaBoost.NC	4.9
C4.5	2.8
Chi-RW	5.5167
FURIA	1.8833
GAssist-Intervalar	3.4167
GFS-GCCL	5.0667
Ripper	4.4167

Table 5: Average testing rankings of the algorithms (Friedman)

1. Average Rankings
   (a) FURIA is ranked lowest, which means it was the best algorithm.

i	algorithm	$z = (R_0 - R_i)/SE$	p	Holm
6	Chi-RW	6.513996	0	0.008333
5	GFS-GCCL	5.707217	0	0.01
4	AdaBoost.NC	5.408409	0	0.0125
3	Ripper	4.541869	0.000006	0.016667
2	GAssist-Intervalar	2.749026	0.005977	0.025
1	C4.5	1.643439	0.100292	0.05

Table 6: Post Hoc comparison Table for testing with $\alpha = 0.05$ (FRIEDMAN)

i. We can look back at the results table and scan FURIAs results compared other algorithms and confirm it performed the best across all algorithms.

(b) Chi-RW is ranked with the highest value; therefore, it performed the worst.

i. Looking back at the results table, we can confirm that on many datasets, the results for Chi-RW are very poor.

2. Post-Hoc Comparison (Holm)

(a) FURIA is withheld from this list as per Holms procedure to compare other algorithms to.

(b) Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.05.

(c) Therefore, the null hypotheses for Chi-RW, CFS-GCCL, AdaBoost.NC, Ripper, and GAssist-Intervalar can be rejected, demonstrating that FURIA is a statistically significant improvement over them. However, the null hypothesis for C4.5 could not be rejected.

2.1.3 Conclusions on Friedman 1xN

1. Chi-RW, GFS-GCCL, and AdaBoost-NC are all poor performing algorithms. Looking back at the CSV results table, we can confirm that these three algorithms are indeed poor. Many percentages are below 50% and some are below 20%.

2. FURIA, although removed from post-hoc comparison, is ranked as the best algorithm. Again, we can see how well it performed in the CSV.

3. Ripper performed well in train but rather poorly in test by comparison.

   (a) Ripper is ranked second best in training but fourth best in testing.

   (b) This leads to a hypothesis about some algorithms performing better with between train and test.

4. C4.5 is a strong performer both with statistical analysis and when naively looking at the raw CSV.

    (a) C4.5 is also ranked best in train and second best in test.

    (b) C4.5 is a decision tree algorithm. These algorithms are generally simpler to construct. Since C4.5 performed extremely well in this analysis, one could safely use it to construct an effective classification tool while also staying generally simple (versus more complex classification methods).

5. This statistical analysis calculated algorithm performance well, as many of its findings match up with what is apparent in the raw CSV data.

2.2 Friedman NxN

2.2.1 Train NxN

i	algorithms	$z = (R_0 - R_i)/SE$	p	Holm	Shaffer
21	C4.5 vs. GFS-GCCL	5.766978	0	0.002381	0.002381
20	GFS-GCCL vs. Ripper	5.049841	0	0.0025	0.003333
19	FURIA vs. GFS-GCCL	4.123539	0.000037	0.002632	0.003333
18	GAssist-Intervalar vs. GFS-GCCL	3.705209	0.000211	0.002778	0.003333
17	Chi-RW vs. GFS-GCCL	3.227117	0.00125	0.002941	0.003333
16	AdaBoost.NC vs. C4.5	2.958191	0.003095	0.003125	0.003333
15	AdaBoost.NC vs. GFS-GCCL	2.808787	0.004973	0.003333	0.003333
14	C4.5 vs. Chi-RW	2.539861	0.01109	0.003571	0.003571
13	AdaBoost.NC vs. Ripper	2.241054	0.025023	0.003846	0.003846
12	C4.5 vs. GAssist-Intervalar	2.061769	0.03923	0.004167	0.004167
11	Chi-RW vs. Ripper	1.822724	0.068345	0.004545	0.004545
10	C4.5 vs. FURIA	1.643439	0.100292	0.005	0.005
9	GAssist-Intervalar vs. Ripper	1.344632	0.178744	0.005556	0.005556
8	AdaBoost.NC vs. FURIA	1.314751	0.188593	0.00625	0.00625
7	FURIA vs. Ripper	0.926302	0.354289	0.007143	0.007143
6	AdaBoost.NC vs. GAssist-Intervalar	0.896421	0.370028	0.008333	0.008333
5	Chi-RW vs. FURIA	0.896421	0.370028	0.01	0.01
4	C4.5 vs. Ripper	0.717137	0.473289	0.0125	0.0125
3	Chi-RW vs. GAssist-Intervalar	0.478091	0.632585	0.016667	0.016667
2	AdaBoost.NC vs. Chi-RW	0.41833	0.675706	0.025	0.025
1	FURIA vs. GAssist-Intervalar	0.41833	0.675706	0.05	0.05

Table 7: P-values Table for training with $\alpha = 0.05$

1. Average Rankings

    (a) The average rankings of each algorithm are exactly the same as in Friedman train 1xN.

2. Post-Hoc Comparisons

    (a) $\alpha = 0.05$

        i. Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.003333.

i	algorithms	$z = (R_0 - R_i)/SE$	p	Holm	Shaffer
21	C4.5 vs. GFS-GCCL	5.766978	0	0.004762	0.004762
20	GFS-GCCL vs. Ripper	5.049841	0	0.005	0.006667
19	FURIA vs. GFS-GCCL	4.123539	0.000037	0.005263	0.006667
18	GAssist-Intervalar vs. GFS-GCCL	3.705209	0.000211	0.005556	0.006667
17	Chi-RW vs. GFS-GCCL	3.227117	0.00125	0.005882	0.006667
16	AdaBoost.NC vs. C4.5	2.958191	0.003095	0.00625	0.006667
15	AdaBoost.NC vs. GFS-GCCL	2.808787	0.004973	0.006667	0.006667
14	C4.5 vs. Chi-RW	2.539861	0.01109	0.007143	0.009091
13	AdaBoost.NC vs. Ripper	2.241054	0.025023	0.007692	0.009091
12	C4.5 vs. GAssist-Intervalar	2.061769	0.03923	0.008333	0.009091
11	Chi-RW vs. Ripper	1.822724	0.068345	0.009091	0.009091
10	C4.5 vs. FURIA	1.643439	0.100292	0.01	0.01
9	GAssist-Intervalar vs. Ripper	1.344632	0.178744	0.011111	0.011111
8	AdaBoost.NC vs. FURIA	1.314751	0.188593	0.0125	0.0125
7	FURIA vs. Ripper	0.926302	0.354289	0.014286	0.014286
6	AdaBoost.NC vs. GAssist-Intervalar	0.896421	0.370028	0.016667	0.016667
5	Chi-RW vs. FURIA	0.896421	0.370028	0.02	0.02
4	C4.5 vs. Ripper	0.717137	0.473289	0.025	0.025
3	Chi-RW vs. GAssist-Intervalar	0.478091	0.632585	0.033333	0.033333
2	AdaBoost.NC vs. Chi-RW	0.41833	0.675706	0.05	0.05
1	FURIA vs. GAssist-Intervalar	0.41833	0.675706	0.1	0.1

Table 8: P-values Table for training with $\alpha = 0.10$

ii. AdaBoost.NC vs. C4.5 and above can be rejected.

iii. Shaffer's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.002381.

iv. Chi-RW vs. GFS-GCCL and above can be rejected.

(b) $\alpha = 0.10$

i. Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.007143.

ii. AdaBoost.NC vs. GFS-GCCL and above can be rejected.

iii. Shaffer's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.004762.

iv. AdaBoost.NC vs. C4.5 and above can be rejected.

2.2.2 Test NxN

1. Average Rankings

   (a) The average rankings of each algorithm are exactly the same as in Friedman test 1xN.

2. Post-Hoc Comparisons

   (a) $\alpha = 0.05$

   i. Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.004545.

   ii. C4.5 vs. Ripper and above can be rejected.

i	algorithms	$z = (R_0 - R_i)/SE$	p	Holm	Shaffer
21	Chi-RW vs. FURIA	6.513996	0	0.002381	0.002381
20	FURIA vs. GFS-GCCL	5.707217	0	0.0025	0.003333
19	AdaBoost.NC vs. FURIA	5.408409	0	0.002632	0.003333
18	C4.5 vs. Chi-RW	4.870557	0.000001	0.002778	0.003333
17	FURIA vs. Ripper	4.541869	0.000006	0.002941	0.003333
16	C4.5 vs. GFS-GCCL	4.063777	0.000048	0.003125	0.003333
15	Chi-RW vs. GAssist-Intervalar	3.76497	0.000167	0.003333	0.003333
14	AdaBoost.NC vs. C4.5	3.76497	0.000167	0.003571	0.004545
13	GAssist-Intervalar vs. GFS-GCCL	2.958191	0.003095	0.003846	0.004545
12	C4.5 vs. Ripper	2.898429	0.00375	0.004167	0.004545
11	FURIA vs. GAssist-Intervalar	2.749026	0.005977	0.004545	0.004545
10	AdaBoost.NC vs. GAssist-Intervalar	2.659384	0.007828	0.005	0.005
9	Chi-RW vs. Ripper	1.972127	0.048595	0.005556	0.005556
8	GAssist-Intervalar vs. Ripper	1.792843	0.072998	0.00625	0.00625
7	C4.5 vs. FURIA	1.643439	0.100292	0.007143	0.007143
6	GFS-GCCL vs. Ripper	1.165348	0.243878	0.008333	0.008333
5	AdaBoost.NC vs. Chi-RW	1.105586	0.268906	0.01	0.01
4	C4.5 vs. GAssist-Intervalar	1.105586	0.268906	0.0125	0.0125
3	AdaBoost.NC vs. Ripper	0.866541	0.386194	0.016667	0.016667
2	Chi-RW vs. GFS-GCCL	0.806779	0.419794	0.025	0.025
1	AdaBoost.NC vs. GFS-GCCL	0.298807	0.765087	0.05	0.05

Table 9: P-values Table for testing with $\alpha = 0.05$

i	algorithms	$z = (R_0 - R_i)/SE$	p	Holm	Shaffer
21	Chi-RW vs. FURIA	6.513996	0	0.004762	0.004762
20	FURIA vs. GFS-GCCL	5.707217	0	0.005	0.006667
19	AdaBoost.NC vs. FURIA	5.408409	0	0.005263	0.006667
18	C4.5 vs. Chi-RW	4.870557	0.000001	0.005556	0.006667
17	FURIA vs. Ripper	4.541869	0.000006	0.005882	0.006667
16	C4.5 vs. GFS-GCCL	4.063777	0.000048	0.00625	0.006667
15	Chi-RW vs. GAssist-Intervalar	3.76497	0.000167	0.006667	0.006667
14	AdaBoost.NC vs. C4.5	3.76497	0.000167	0.007143	0.009091
13	GAssist-Intervalar vs. GFS-GCCL	2.958191	0.003095	0.007692	0.009091
12	C4.5 vs. Ripper	2.898429	0.00375	0.008333	0.009091
11	FURIA vs. GAssist-Intervalar	2.749026	0.005977	0.009091	0.009091
10	AdaBoost.NC vs. GAssist-Intervalar	2.659384	0.007828	0.01	0.01
9	Chi-RW vs. Ripper	1.972127	0.048595	0.011111	0.011111
8	GAssist-Intervalar vs. Ripper	1.792843	0.072998	0.0125	0.0125
7	C4.5 vs. FURIA	1.643439	0.100292	0.014286	0.014286
6	GFS-GCCL vs. Ripper	1.165348	0.243878	0.016667	0.016667
5	AdaBoost.NC vs. Chi-RW	1.105586	0.268906	0.02	0.02
4	C4.5 vs. GAssist-Intervalar	1.105586	0.268906	0.025	0.025
3	AdaBoost.NC vs. Ripper	0.866541	0.386194	0.033333	0.033333
2	Chi-RW vs. GFS-GCCL	0.806779	0.419794	0.05	0.05
1	AdaBoost.NC vs. GFS-GCCL	0.298807	0.765087	0.1	0.1

Table 10: P-values Table for testing with $\alpha = 0.10$

iii. Shaffer's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.002381.

iv. AdaBoost.NC vs. C4.5 and above can be rejected.

(b) $\alpha = 0.10$

i. Holm's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.011111.

ii. C4.5 vs. FURIA and above can be rejected.

iii. Shaffer's procedure rejects those hypotheses that have an unadjusted p-value ≤ 0.004762.

iv. C4.5 vs. Ripper and above can be rejected.

2.2.3 Conclusions on Friedman NxN

1. One could view the comparison results table as a matrix (as in the statistical paper provided). Because it is purposeless to compare an algorithm against itself, each result table shows $\binom{7}{2} = 21$ comparisons.

2. GFS-GCCL, AdaBoost-NC, and Chi-RW consistently perform poorly in both training and testing.

3. FURIA and C4.5 consistently perform well in both training and testing.

4. In train and test, Shaffer has slightly or equal higher p-values than Holm. However, when adjusted, Holm then has slightly higher p-values than Shaffer.

5. Results follow a pairwise pattern of performance based on the strength of the compared algorithms.

   (a) The good algorithms perform statistically significantly better than the bad ones.

   (b) The good algorithms do not perform statistically significantly better than other good algorithms.

   (c) The bad algorithms do not perform statistically significantly better than other bad algorithms.

   (d) For example, in test, Chi-RW vs. FURIA is the highest ranked comparison, and FURIA is the highest ranked algorithm individually and Chi-RW is the lowest ranked algorithm.

   (e) We see the same exact behavior in train where C4.5 is the highest ranked algorithm and GFS-GCCL is the lowest. When compared, they are the highest ranked comparison.