Project 7: Decision Trees & Random Forest

Paul Beggs
GitHub Link

November 5, 2025

1 Two-Letter Analysis

For the two-letter dataset, the learners performed very similarly to that of KNN. Both learners had near perfect accuracy (Appendix A), which I attribute to the letters being relatively distinct from each other. This can be seen especially clearly with the decision tree's visualization (Figure 1), but less so with random forest's (Figure 4). This is because each tree in the random forest is trained on a different random sample of the data, and at every split, the tree is only allowed to consider a small subset of the total pixels. These trees culminate in a feature rich union of all 30 rule sets.

2 Three-Letter Analysis

When the learners used the three-letter dataset, they performed worse than KNN, but we can see that the decision tree's results are consistent with that of random forest's (Appendix B). Again, we see that not many features are needed to distinguish between the three letters for decision trees (Figure 2), but again the random forest's is full of features (Figure 5).

3 Eight-Letter Analysis

Finally, for the eight-letter dataset, we see a stark contrast between all learners: KNN does the best, followed by decision trees (Table 5), and in dead last, random forest (Table 6). It's honestly impressive how poorly the random forest learner does. If you look at Figure 6, you'll see that for the 'T,' 'V,' 'Z,' and 'L' labels there is only one or two on pixels that differentiate one letter from another, thus leading to mislabeling of letters.

4 Conclusion

Through this analysis, there is a trend of relatively high performance for few letters, but as we increase the amount of letters in the data set, that performance diminishes. We see that the decision trees perform well, and this is because at every split, it scans all pixels and is guaranteed to find one perfect feature (if it exists) that gives the highest information gain. This is in contrast to the random forest learner consisting of trees who are forced to pick a

feature that looks good from their random sample. This leads me to believe that we would be better off only employing random forest for images that are feature rich. It should not be used for images that consist of binary pixels, and more so for images that have full RGB values, for example.

Appendix A: Two Letter Dataset

	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	4	4	100%	3	3	100%	5	4	80%	7	6	86%
H	6	6	100%	7	7	100%	5	5	100%	2	2	100%

Table 1: Decision Tree 4-Way Cross Validation Results With Two Letters

	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	6	5	83%	5	5	100%	6	6	100%	2	2	100%
H	4	4	100%	5	5	100%	4	4	100%	7	6	86%

Table 2: Random Forest 4-Way Cross Validation Results With Two Letters

Appendix B: Three Letter Dataset

	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	6	5	83%	4	4	100%	4	3	75%	5	5	100%
E	6	4	67%	5	4	80%	6	6	100%	3	3	100%
H	3	3	100%	6	6	100%	5	5	100%	6	5	83%

Table 3: Decision Tree 4-Way Cross Validation Results With Three Letters

	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	6	6	100%	4	3	75%	3	3	100%	6	6	100%
E	4	4	100%	5	5	100%	4	4	100%	7	6	86%
H	5	5	100%	6	5	83%	8	7	88%	1	1	100%

Table 4: Random Forest 4-Way Cross Validation Results With Three Letters

Appendix C: Eight Letter Dataset

	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	5	3	60%	5	0	0%	5	3	60%	5	5	100%
E	8	6	75%	5	0	0%	2	0	0%	5	2	40%
H	6	3	50%	3	2	67%	7	5	71%	4	3	75%
L	3	2	67%	4	3	75%	7	3	43%	6	6	100%
Q	7	6	86%	4	0	0%	4	4	100%	5	3	60%
T	6	6	100%	4	3	75%	5	3	60%	5	5	100%
V	1	1	100%	11	3	27%	2	2	100%	6	6	100%
Z	4	4	100%	4	3	75%	8	6	75%	4	3	75%

Table 5: Decision Tree 4-Way Cross Validation Results With Eight Letters

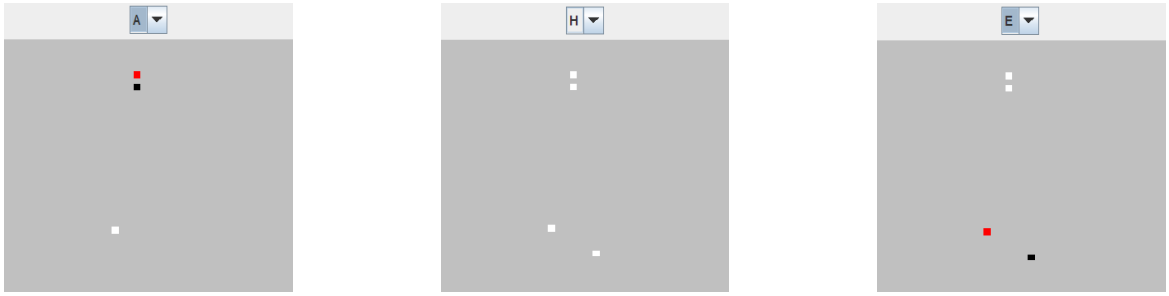
	1st Partition			2nd Partition			3rd Partition			4th Partition		
Lbl.	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**	**Ex.**	**Cor.**	**%**
A	6	4	67%	6	6	100%	3	3	100%	5	1	20%
E	4	0	0%	6	0	0%	6	0	0%	4	3	75%
H	6	3	50%	4	3	75%	7	4	57%	3	3	100%
L	7	0	0%	5	1	20%	4	0	0%	4	1	25%
Q	2	2	100%	6	0	0%	6	0	0%	6	0	0%
T	5	0	0%	4	4	100%	4	0	0%	7	0	0%
V	4	0	0%	5	0	0%	6	0	0%	5	0	0%
Z	6	0	0%	4	0	0%	4	0	0%	6	0	0%

Table 6: Random Forest 4-Way Cross Validation Results With Eight Letters

Appendix D: Decision Tree Visualizations

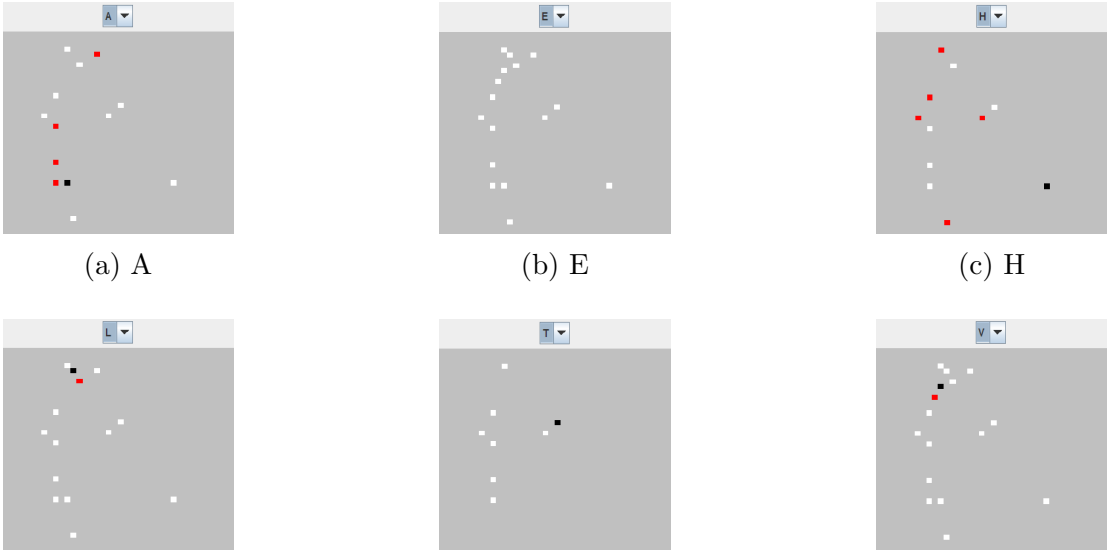
(a) A (b) H

Figure 1: Two-Letter Visualizations



(a) A (b) H (c) E

Figure 2: Three-Letter Visualizations



(a) A (b) E (c) H

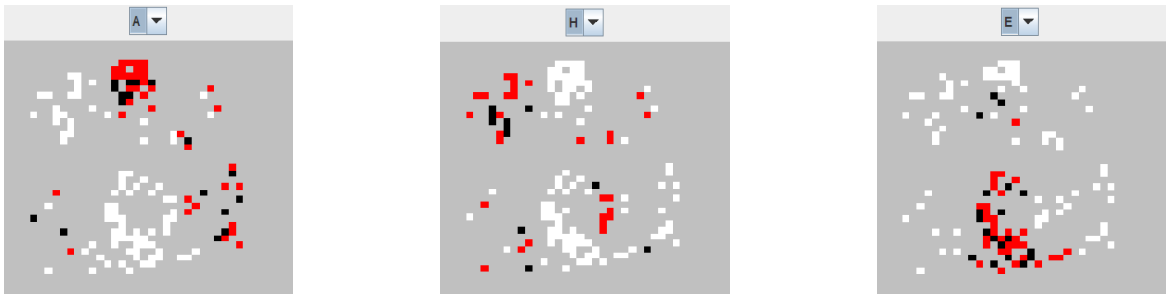
(d) L (e) T (f) V

Figure 3: Eight-Letter Visualizations

Appendix E: Random Forest Visualizations

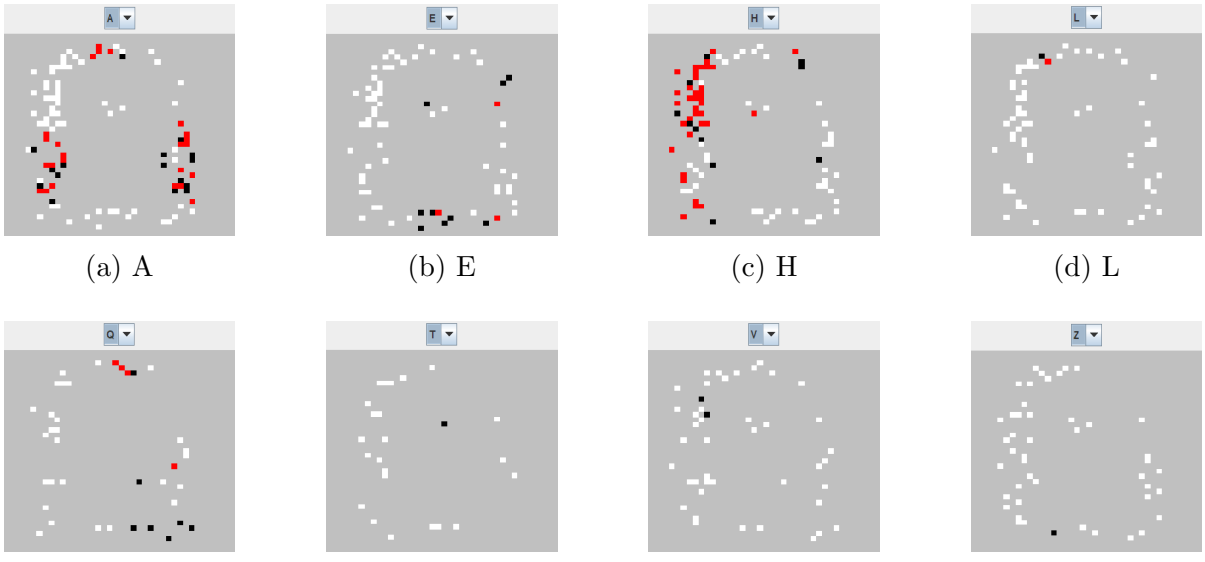
(a) A (b) H

Figure 4: Two-Letter Visualizations



(a) A (b) H (c) E

Figure 5: Three-Letter Visualizations



(a) A (b) E (c) H (d) L

(e) Q (f) T (g) V (h) Z

Figure 6: Eight-Letter Visualizations