

CS6530 Assignment 4 Report

Brandon Rodriguez

June 8, 2020

Part 1

See "a4.r" file, for descriptions of what I think the provided code is doing.

Note that, even after googling, I could not determine what "**Species~,iris**" does, which is a segment of code used in 3 out of 4 of the provided code examples.

Taking an educated guess, I think this is the equivalent of "**iris[['Species']]**" or "**iris\$Species**", except that it seems to automatically format the data in some way such that rpart accepts it?

Either way, it seems important for the code to function, and appears to indicate the attribute to use as the classifier label result.

Part 2.1 - Balance Dataset

2.1) Decision Tree Results

Predictions (Confusion Table):

		Orig		
		B	L	R
Predic	B	0	0	0
	L	7	106	6
	R	7	10	89

Stat Summary:

Total Accuracy: 0.8667

	B	L	R
Balanced Accuracy	0.5000	0.8973	0.9030
Sensitivity/Recall	0.0000	0.9138	0.9368
Specificity	1.0000	0.8807	0.8692
Precision	NaN	0.8908	0.8396

2.1) Forest Results

Predictions (Confusion Table):

		Orig		
		B	L	R
Predic	B	14	0	0
	L	0	116	0
	R	0	0	95

Stat Summary:

Total Accuracy: 1.0

	B	L	R
Balanced Accuracy	1.0	1.0	1.0
Sensitivity/Recall	1.0	1.0	1.0
Specificity	1.0	1.0	1.0
Precision	1.0	1.0	1.0

Part 2.2 - Nursery Dataset

2.2) Decison Tree Results

Predictions (Confusion Table):

		Orig				
		not_rec	priority	rec	spec_prior	very_rec
Predic	not_rec	1623	0	0	0	0
	priority	0	1284	0	147	123
	rec	0	0	0	0	0
	spec_prior	0	373	0	1410	0
	very_rec	0	0	0	0	0

Stat Summary:

Total Accuracy: 0.8704

	not_rec	priority	rec	spec_prior	very_rec
Balanced Accuracy	1.0000	0.8466	N/A	0.8980	0.5000
Sensitivity/Recall	1.0000	0.7749	N/A	0.9056	0.0000
Specificity	1.0000	0.9183	1.0000	0.8904	1.0000
Precision	1.0000	0.8263	N/A	0.7908	NaN

2.2) Forest Results

Predictions (Confusion Table):

		Orig				
		not_rec	priority	rec	spec_prior	very_rec
Predic	not_rec	1623	0	0	0	0
	priority	0	1644	0	4	22
	rec	0	0	0	0	0
	spec_prior	0	13	0	1553	0
	very_rec	0	0	0	0	101

Stat Summary:

Total Accuracy: 0.9921

	not_rec	priority	rec	spec_prior	ver_rec
Balanced Accuracy	1.0000	0.9921	N/A	0.9968	0.9106
Sensitivity/Recall	1.0000	0.9922	N/A	0.9974	0.8211
Specificity	1.0000	0.9921	1.0000	0.9962	1.0000
Precision	1.0000	0.9844	N/A	0.9917	1.0000

Part 2.3 - LED Dataset

2.3) Decison Tree Results

Predictions (Confusion Table):

		Orig									
		0	1	2	3	4	5	6	7	8	9
Predic	0	129	1	0	0	0	0	0	1	16	1
	1	0	135	0	0	14	0	0	20	0	0
	2	3	3	142	9	0	0	0	2	5	1
	3	14	19	12	148	1	16	9	15	15	17
	4	0	16	0	1	144	3	0	0	1	15
	5	4	0	2	2	15	134	11	2	2	21
	6	11	1	1	0	0	15	148	0	15	2
	7	6	11	18	14	3	1	0	147	0	3
	8	16	0	14	3	11	3	17	0	153	16
	9	22	1	1	16	18	19	5	11	16	141

Stat Summary:

Total Accuracy: 0.7105

	0	1	2	3	4
Balanced Accuracy	0.8093	0.8516	0.8673	0.8508	0.8395
Sensitivity/Recall	0.6293	0.7219	0.7474	0.7668	0.6990
Specificity	0.9894	0.9812	0.9873	0.9347	0.9799
Precision	0.8716	0.7988	0.8606	0.5564	0.8000
	5	6	7	8	9
Balanced Accuracy	0.8345	0.7789	0.7424	0.6861	0.6498
Sensitivity/Recall	0.7016	0.7789	0.7424	0.6861	0.6498
Specificity	0.9674	0.9751	0.9689	0.9550	0.9389
Precision	0.6943	0.7668	0.7241	0.6567	0.5640

2.3) Forest Results

Predictions (Confusion Table):

		Orig									
		0	1	2	3	4	5	6	7	8	9
Predic	0	142	0	1	3	0	0	2	4	18	1
	1	1	167	0	7	24	1	0	22	0	2
	2	3	0	149	2	0	0	0	0	5	0
	3	1	5	21	147	1	13	1	12	13	18
	4	0	2	0	0	135	1	1	0	4	0
	5	4	0	2	2	3	133	11	0	2	21
	6	10	1	3	0	0	18	154	0	16	2
	7	8	11	2	12	5	2	0	158	0	4
	8	16	0	11	3	4	3	16	0	148	16
	9	20	1	1	17	34	20	5	2	17	153

Stat Summary:

Total Accuracy: 0.7430

	0	1	2	3	4
Balanced Accuracy	0.8383	0.9308	0.8893	0.8573	0.8254
Sensitivity/Recall	0.6927	0.8930	0.7842	0.7617	0.6553
Specificity	0.9838	0.9686	0.9945	0.9530	0.9955
Precision	0.8304	0.7455	0.9371	0.6336	0.9441
	5	6	7	8	9
Balanced Accuracy	0.8357	0.8915	0.8868	0.8124	0.8197
Sensitivity/Recall	0.6963	0.8105	0.7980	0.6637	0.7051
Specificity	0.9751	0.9724	0.9756	0.9612	0.9344
Precision	0.7472	0.7549	0.7822	0.6820	0.5667

Part 2 Summary

In all instances of Part 2, the Forest (ensemble method) seemed all around more accurate than the single Decision Tree method.

In part 2.1, Forest even increased the accuracy to 100%, which I think is pretty impressive. The Forest in Part 2.2 increased accuracy to nearly 100%, with minimal errors.

Meanwhile, the Forest in Part 2.3 only increased accuracy by a few percent, but it was still an overall improvement. I think this was likely due to the inherent noise of the dataset, and the Forest method most likely internalizing some of that noise.