1 Decision Tree, written

1. Decision Tree construction

   (a) attributes: x1, x2, x3, x4//
   entropy of whole dataset: H(y) = $-(2/7)log_2(2/7)-(5/7)log_2(5/7)$ = .863 // //
   entropy of x1 = 0: p = 1, n = 4, H(x1=0) = $-(1/5) * log_2(1/5) - (4/5) * log_2(4/5)$ = .722
   entropy of x1 = 1 : p = 1, n =1, H(x1=1) = $-(1/2) * log_2(1/2) - (1/2) * log_2(1/2)$ = 1
   Information gain of x1 = .863 - ((5/7)*.722 + (2/7)*1)= .062

   entropy of x2 = 0: p = 1, n = 2, H(x2=0) = $-(1/3) * log_2(1/3) - (2/3) * log_2(2/3)$ = .918
   entropy of x2 = 1 : p = 4, n =0, H(x2=1) = 0
   Information gain of x2 = .863 - ((3/7)*.918 + (2/7)*0)= .47

   entropy of x3 = 0: p = 3, n = 1, H(x3=0) = $-(3/4) * log_2(3/4) - (3/4) * log_2(3/4)$ = .811
   entropy of x3 = 1 : p = 2, n =1, H(x3=1) = $-(2/3) * log_2(2/3) - (1/3) * log_2(1/3)$ = .918
   Information gain of x3 = .863 - ((4/7)*.811 + (3/7)*.918)= .006

   entropy of x4 = 0: p = 0, n = 4, H(x3=0) = $-(0/4) * log_2(0/4) - (4/4) * log_2(4/4)$ = 0
   entropy of x4 = 1 : p = 2, n =1, H(x3=1) = $-(2/3) * log_2(2/3) - (1/3) * log_2(1/3)$ = .918
   Information gain of x3 = .863 - ((4/7)*.0 + (3/7)*.918)= .469

   Information gain highest at x2 or x4, I chose x2. split table into:

x1	x2	x3	x4	y
0	0	1	0	0
0	0	1	1	1
1	0	0	1	1

Table 1: Subset where $x_2 = 0$

   entropy of table x2 = 0: H(y) = $-(1/3)log_2(1/3)-(2/3)log_2(2/3)$ = .918 // //

x1	x2	x3	x4	y
0	1	0	0	0
0	1	1	0	0
1	1	0	0	0
0	1	0	1	0

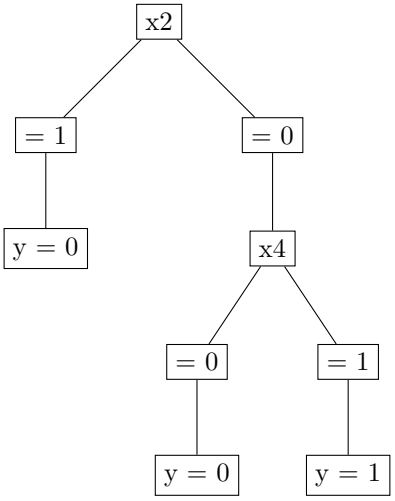
Table 2: Subset where $x_2 = 1$

entropy of x1 = 0: p = 1, n = 1, H(x1=0) = $-(1/2) * log_2(1/2) - (1/2) * log_2(1/2) = 1$
entropy of x1 = 1 : p = 1, n =0, H(x1=1) = $-(1/1) * log_2(1/1) - (0/1) * log_2(0/1) = 0$
Information gain of x1 = .918 - ((2/3)*1 + (1/3)*0)= .251

entropy of x3 = 0: p = 1, n = 0, H(x3=0) = $-(1/1) * log_2(1/1) - (0/1) * log_2(0/1) = 0$
entropy of x3 = 1 : p = 1, n =1, H(x3=1) = $-(2/3) * log_2(2/3) - (1/3) * log_2(1/3) = 1$
Information gain of x3 = .918 - ((1/3)*.0 + (2/3)*.1)= .851

entropy of x4 = 0: p = 0, n = 1, H(x3=0) = $-(1/1) * log_2(1/1) - (0/1) * log_2(0/1) = 0$
entropy of x4 = 1 : p = 2, n =0, H(x3=1) = $-(2/2) * log_2(2/2) - (0/2) * log_2(0/2) = 0$
Information gain of x3 = .918 - ((1/3)*.0 + (2/3)*0)= .918

information gain highest at x4, finished.
conjunction rules:
if x2 = 1, y = 0 if x2 = 0 if x4 = 0, y = 0 if x4 = 1, y = 1



(b) $y = (\overline{x2} \wedge x4)$

2. tennis

(a) majority error of dataset: $5/14 = .357$

Majority error Outlook node: sunny: p=2, n=3 — overcast: p=4, n=0 — rainy: p=3, n=2 information gain of outlook: $5/14 - ((5/14) * (2/5) + (4/14) * (0/4) + (5/14) * (2/5)) = .071$

Majority error temp node: hot: p=2, n=2 — mild: p=4, n=2 — cool: p=3, n=1 information gain of outlook: $5/14 - ((4/14) * (2/4) + (6/14) * (2/6) + (4/14) * (1/4)) = 0$

Majority error Humidity node: high: p=3, n=4 — normal: p=6, n=1 — low: p=0, n=0 information gain of outlook: $5/14 - ((7/14) * (3/7) + (7/14) * (1/7) + 0) = .071$

Majority error wind node: strong: p=3, n=3 — weak: p=6, n=2 information gain of outlook: $5/14 - ((6/14) * (3/6) + (8/14) * (2/8)) = 0$

information gain highest at outlook or humidity, I choose outlook to split on outlook table for outlook = sunny:

Outlook	Temperature	Humidity	Wind	Play Tennis
S	H	H	W	-
S	H	H	S	-
S	M	H	W	-
S	C	N	W	+
S	M	N	S	+

Table 3: Table of data where Outlook is Sunny

table for outlook = overcast:

Outlook	Temperature	Humidity	Wind	Play Tennis
O	H	H	W	+
O	C	N	S	+
O	H	H	W	+
O	M	H	S	+

Table 4: Table of data where Outlook is Overcast

table for outlook = rainy

Outlook	Temperature	Humidity	Wind	Play Tennis
R	M	H	W	+
R	C	N	W	+
R	C	N	S	-
R	M	N	W	+
R	M	H	S	-

Table 5: Table of data where Outlook is Rainy

majority error of sunny dataset: $2/5 = .4$

Majority error temp node: hot: p=0, n=2 — mild: p=1, n=1 — cool: p=1, n=0 information gain of outlook: $2/5 - ((2/5) * (0/4) + (2/5) * (1/2) + (1/5) * (0/1)) = .2$

Majority error Humidity node: high: p=0, n=3 — normal: p=2, n=0 — low: p=0, n=0 information gain of outlook: $2/5 - ((3/5) *$

$(0/3) + (2/5) * (0/2) + 0) = .2$

Majority error wind node: strong: p=1, n=1 — weak: p=1, n=2 information gain of outlook: $2/5 - ((2/15) * (1/2) + (3/5) * (1/3)) = .133$

split on humidity to complete this side.

For outlook rainy:

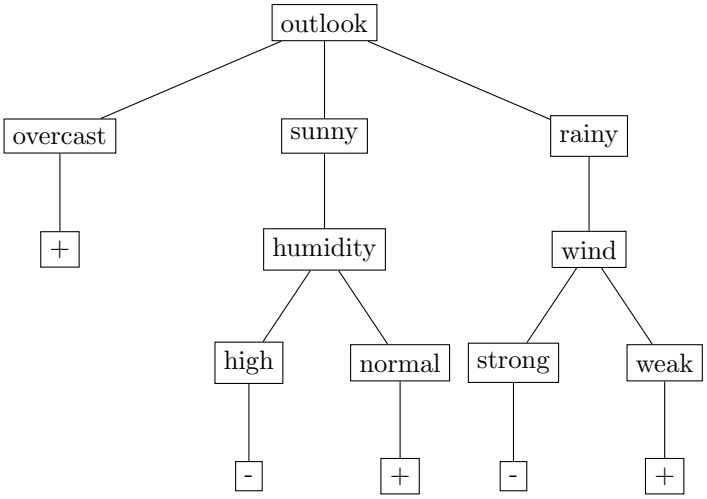
majority error of rainy dataset: $2/5 = .4$

Majority error temp node: mild: p=2, n=1 — cool: p=1, n=1 information gain of outlook: $2/5 - ((3/5) * (1/3) + (2/5) * (1/2)) = 0$

Majority error Humidity node: high: p=1, n=1 — normal: p=2, n=1 — low: p=0, n=0 information gain of outlook: $2/5 - ((2/5) * (1/2) + (3/5) * (1/3) + 0) = 0$

Majority error wind node: strong: p=0, n=2 — weak: p=3, n=0 information gain of outlook: $2/5 - 0 = .133$

split rain table on wind to complete this side.



(b) Gini index of dataset: $1 - (5/14)^2 - (9/14)^2 = .459$

Gini index Outlook node: sunny: p=2, n=3 — overcast: p=4, n=0 — rainy: p=3, n=2 information gain of outlook: $.459 - (5/14)(1 - (2/5)^2 - (3/5)^2) - (4/14)(1 - (4/4)^2 - (0/4)^2) - (5/14)(1 - (3/5)^2 - (2/5)^2) = .116$

Gini index temp node: hot: p=2, n=2 — mild: p=4, n=2 — cool: p=3, n=1 information gain of temp: $.459 - (5/14)(1 - (2/4)^2 - (2/4)^2) - (6/14)(1 - (4/6)^2 - (2/6)^2) - (4/14)(1 - (3/4)^2 - (2/4)^2) = .036$

Gini index Humidity node: high: p=3, n=4 — normal: p=6, n=1 — low: p=0, n=0 information gain of humidity: $.459 - (7/14)(1 - (3/7)^2 - (4/7)^2) - (7/14)(1 - (6/7)^2 - (1/7)^2) = .092$

Gini index wind node: strong: p=3, n=3 — weak: p=6, n=2 information gain of wind: $.459 - (6/14)(1 - (3/6)^2 - (3/6)^2) - (8/14)(1 - (6/8)^2 - (2/8)^2) = .030$

information gain highest at outlook:

Outlook	Temperature	Humidity	Wind	Play Tennis
S	H	H	W	-
S	H	H	S	-
S	M	H	W	-
S	C	N	W	+
S	M	N	S	+

Table 6: Table of data where Outlook is Sunny

Outlook	Temperature	Humidity	Wind	Play Tennis
O	H	H	W	+
O	C	N	S	+
O	H	H	W	+
O	M	H	S	+

Table 7: Table of data where Outlook is Overcast

Outlook	Temperature	Humidity	Wind	Play Tennis
R	M	H	W	+
R	C	N	W	+
R	C	N	S	-
R	M	N	W	+
R	M	H	S	-

Table 8: Table of data where Outlook is Rainy

overcast table already completed. sunny table: Gini index of sunny dataset: $1-(2/5)^2-(3/15)^2=.8$

Gini index temp node: hot: p=0, n=2 — mild: p=1, n=1 — cool: p=1, n=0 information gain of temp: $.8-(2/5)(1-(0/2)^2-(2/2)^2)-(2/5)(1-(1/2)^2-(1/2)^2)-(1/5)(1-(1/1)^2-(0/1)^2)=.6$

Gini index Humidity node: high: p=0, n=3 — normal: p=2, n=0 information gain of outlook: $.8-(3/15)(1-(0/3)^2-(3/3)^2)-(2/5)(1-(2/2)^2-(0/2)^2)=.8$

Gini index wind node: strong: p=1, n=1 — weak: p=1, n=2 information gain of outlook: $.8-(2/5)(1-(1/2)^2-(1/2)^2)-(3/5)(1-(1/3)^2-(2/3)^2)=.333$

split on humidity. completes sunny table.

for rainy table:

Gini index of sunny dataset: $1-(3/5)^2-(2/15)^2=.8$

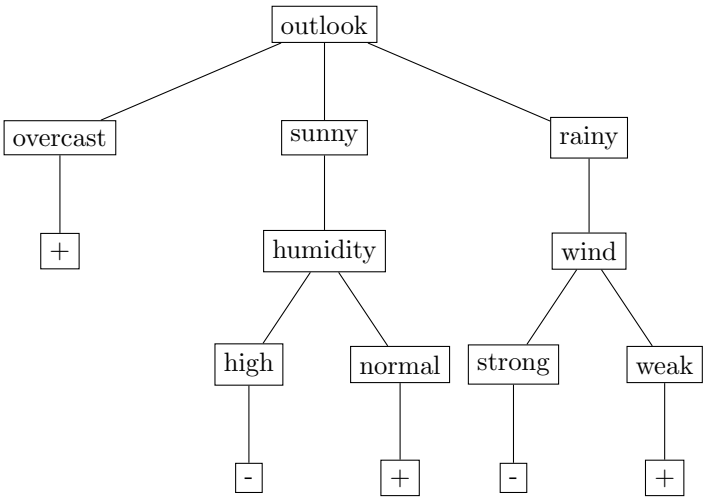
Gini index temp node: mild: p=2, n=1 — cool: p=1, n=1 information gain of temp: $.8-(3/5)(1-(2/3)^2-(2/3)^2)-(2/5)(1-(1/2)^2-(1/2)^2)=.533$

Gini index Humidity node: high: p=1, n=1 — normal: p=2, n=1 information gain of outlook: $.8-(2/5)(1-(1/2)^2-(1/2)^2)-(3/5)(1-(2/3)^2-(1/3)^2)=.333$

Gini index wind node: strong: p=0, n=2 — weak: p=3, n=0 infor-

mation gain of outlook: $.8 - (2/5)(1 - (0/2)^2 - (2/2)^2) - (3/5)(1 - (3/3)^2 - (0/3)^2) = .8$

split on wind. completes rainy table



(c) there is no difference in the tables created here, but I chose to split first by outlook for the majority error tree, even though it could have also been split on humidity. These tables could look different depending on the if the purity came out slightly differently. however, these all happened to be the same.

3. missing, mild, normal, weak, +

(a) most common value in table is sunny or rainy. I choose rainy becuase there is a rainy entry with mild temp, normal humidity, and weak wind already. I'm going to use majority error to compute information gain.

majority error of dataset: $5/15 = .333$

Majority error Outlook node: sunny: p=2, n=3 — overcast: p=4, n=0 — rainy: p=4, n=2 information gain of outlook: $5/15 - ((5/15) * (2/5) + (4/15) * (0/4) + (6/15) * (2/6)) = .066$

Majority error temp node: hot: p=2, n=2 — mild: p=5, n=2 — cool: p=3, n=1 information gain of outlook: $5/15 - ((4/15) * (2/4) + (7/15) * (2/7) + (4/15) * (1/4)) = 0$

Majority error Humidity node: high: p=3, n=4 — normal: p=7, n=1 — low: p=0, n=0 information gain of outlook: $5/15 - ((7/15) * (3/7) + (8/15) * (1/8) + 0) = .066$

Majority error wind node: strong: p=3, n=3 — weak: p=7, n=2 information gain of outlook: $5/15 - ((6/15) * (3/6) + (9/15) * (2/9)) = 0$

(b) among + plays, outlook is the most common value.
majority error of dataset: $5/15 = .333$

Majority error Outlook node: sunny: p=2, n=3 — overcast: p=5, n=0 — rainy: p=4, n=2 information gain of outlook: $5/15 - ((5/15) * (2/5) + (5/15) * (0/5) + (6/15) * (2/6)) = .066$

Majority error temp node: hot: p=2, n=2 — mild: p=5, n=2 — cool: p=3, n=1 information gain of outlook: $5/15 - ((4/15) * (2/4) + (7/15) * (2/7) + (4/15) * (1/4)) = 0$

Majority error Humidity node: high: p=3, n=4 — normal: p=7, n=1 — low: p=0, n=0 information gain of outlook: $5/15 - ((7/15) * (3/7) + (8/15) * (1/8) + 0) = .066$

Majority error wind node: strong: p=3, n=3 — weak: p=7, n=2 information gain of outlook: $5/15 - ((6/15) * (3/6) + (9/15) * (2/9)) = 0$

(c) 5/14: sunny, 4/14: overcast, 5/14: rain
majority error of dataset: $5/15 = .333$

Majority error Outlook node: sunny: $p = 2\frac{5}{14}, n = 3 | overcast : p = 4\frac{4}{14}, n = 0 | rainy : p = 3\frac{5}{14}, n = 2$ information gain of outlook: $5/15 - (((75/14)/15) * (2/(75/14)) + ((75/14)/15) * (0/(74/14)) +$

$((75/14)/15) * (2/(75/14))) = .066$

Majority error temp node: hot: p=2, n=2 — mild: p=5, n=2 — cool: p=3, n=1 information gain of outlook: $5/15 - ((4/15) * (2/4) + (7/15) * (2/7) + (4/15) * (1/4)) = 0$

Majority error Humidity node: high: p=3, n=4 — normal: p=7, n=1 — low: p=0, n=0 information gain of outlook: $5/15 - ((7/15) * (3/7) + (8/15) * (1/8) + 0) = .066$

Majority error wind node: strong: p=3, n=3 — weak: p=7, n=2 information gain of outlook: $5/15 - ((6/15) * (3/6) + (9/15) * (2/9)) = 0$

value	Outlook	Temperature	Humidity	Wind	Play Tennis
1	S	H	H	W	-
1	S	H	H	S	-
1	S	M	H	W	-
1	S	C	N	W	+
1	S	M	N	S	+
5/14	S	M	H	W	+

Table 9: Table of data where Outlook is Sunny

(d) split on outlook

value	Outlook	Temperature	Humidity	Wind	Play Tennis
1	O	H	H	W	+
1	O	C	N	S	+
1	O	H	H	W	+
1	O	M	H	S	+
4/14	O	M	H	W	+

Table 10: Table of data where Outlook is Overcast

for sunny table :
majority error of sunny dataset: $(33/14)/(75/14) = .44$

Majority error temp node: $hot : p = 0, n = 2 | mild : p = 1\frac{5}{14}, n = 1 | cool : p = 1, n = 0$ information gain of outlook: $.44 - ((2/(75/14) *$

value	Outlook	Temperature	Humidity	Wind	Play Tennis
1	R	M	H	W	+
1	R	C	N	W	+
1	R	C	N	S	-
1	R	M	N	W	+
1	R	M	H	S	-
5/14	R	M	H	W	+

Table 11: Table of data where Outlook is Rainy

$(0/4) + ((33/14)/(75/14)) * ((19/14)/2) + (1/(75/14)) * (0)) = .141$

Majority error Humidity node: $high : p = \frac{5}{14}, n = 3 | normal : p = 2, n = 0$ information gain of outlook: $.44 - ((3/(75/14)) * ((5/14)/3) + (2/(75/14)) * (0/2) + 0) = .373$

Majority error wind node: $strong : p = 1, n = 1 | weak : p = 1\frac{5}{14}, n = 2 information gain ino foutlook$:.44 - ((2/(75/14))*(1/2) + (3/(75/14))*((19/14)/(47/14)) = .027

split on humidity to complete this side, as close as it can be.

For outlook rainy:

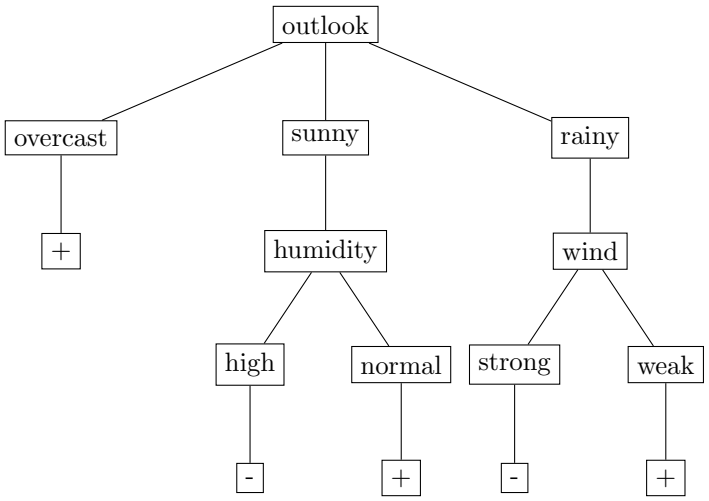
majority error of rainy dataset: $2/(75/14) = .373$

Majority error temp node: $mild : p = 2\frac{5}{14}, n = 1 | cool : p = 1, n = 1$ information gain of outlook: $.373 - ((3/(75/14)) * (1/(47/14)) + (2/(75/14)) * (1/2)) = .02$

Majority error Humidity node: $high : p = 1\frac{5}{14}, n = 1 | normal : p = 2, n = 1 | low : p = 0, n = 0$ information gain of outlook: $.373 - (((33/14)/(75/14)) * (1/(33/14)) + (3/(75/14)) * (1/3) + 0) = 0$

Majority error wind node: $strong : p = 0, n = 2 | weak : p = 3\frac{5}{14}, n = 0$ information gain of outlook: $.373 - 0 = .373$

split on wind to complete rainy side.



4. because entropy can never be above 1, the summation of the information portions times entropy will never be above the original information value. that is: $\sum_{vEValues(A)} \frac{S_v}{S} * entropy(S_v) \leq \sum_{vEValues(A)} \frac{S_v}{S}$ and because $\sum_{vEValues(A)} \frac{S_v}{S} = entropy(S)$, the gain can at lowest be Entropy(s) - Entropy(s)

5.

2 Decision Tree Code questions

1. Car Decision tree Results

Max Depth	Training Accuracy	Testing Accuracy
1	0.698	0.703297
2	0.778	0.777473
3	0.817	0.806319
4	0.914	0.855769
5	0.959	0.858516
6	1.000	0.839286

Results for Purity Equation: Entropy

Max Depth	Training Accuracy	Testing Accuracy
1	0.698	0.703297
2	0.701	0.690934
3	0.756	0.725275
4	0.808	0.721154
5	0.903	0.732143
6	1.000	0.686813

Results for Purity Equation: Majority Error

Max Depth	Training Accuracy	Testing Accuracy
1	0.698	0.703297
2	0.778	0.777473
3	0.822	0.818681
4	0.906	0.865385
5	0.957	0.869505
6	1.000	0.842033

Results for Purity Equation: Gini Index

1.c)from the tables, we can see that the training accuracy increases until it gets 100% correct while the training accuracy stops and begins to drop with increasing max depth.

2. Bank Decision Tree results

   (a) teat unknown as an attribute

Max Depth	Training Accuracy	Testing Accuracy
1	0.8808	0.8752
2	0.8900	0.8832
3	0.8912	0.8830
4	0.9046	0.8582
5	0.9148	0.8374
6	0.9242	0.8260
7	0.9294	0.8164
8	0.9358	0.8020
9	0.9432	0.7996
10	0.9496	0.7956
11	0.9540	0.7912
12	0.9614	0.7862
13	0.9666	0.7844
14	0.9722	0.7826
15	0.9730	0.7832
16	0.9750	0.7826

Training and Testing Accuracy for Entropy Purity Equation

Max Depth	Training Accuracy	Testing Accuracy
1	0.8912	0.8834
2	0.8912	0.8862
3	0.8970	0.8808
4	0.9064	0.8662
5	0.9188	0.8584
6	0.9284	0.8476
7	0.9364	0.8284
8	0.9434	0.8176
9	0.9504	0.8032
10	0.9558	0.7918
11	0.9598	0.7828
12	0.9638	0.7744
13	0.9678	0.7700
14	0.9712	0.7658
15	0.9734	0.7644
16	0.9756	0.7616

Training and Testing Accuracy for Majority Error Purity Equation

Max Depth	Training Accuracy	Testing Accuracy
1	0.8912	0.8834
2	0.8894	0.8842
3	0.8934	0.8740
4	0.9002	0.8532
5	0.9078	0.8370
6	0.9140	0.8250
7	0.9216	0.8128
8	0.9302	0.8024
9	0.9394	0.7978
10	0.9470	0.7888
11	0.9546	0.7834
12	0.9606	0.7792
13	0.9648	0.7768
14	0.9708	0.7738
15	0.9730	0.7740
16	0.9756	0.7740

Training and Testing Accuracy for Gini Index Purity Equation

(b) replace unknown with most common value from that column

Max Depth	Training Accuracy	Testing Accuracy
1	0.8808	0.8752
2	0.8900	0.5578
3	0.8902	0.5520
4	0.8998	0.4956
5	0.9086	0.4900
6	0.9170	0.4870
7	0.9218	0.4824
8	0.9286	0.4818
9	0.9340	0.4808
10	0.9406	0.4792
11	0.9476	0.4778
12	0.9532	0.4774
13	0.9606	0.4766
14	0.9660	0.4758
15	0.9690	0.4760
16	0.9702	0.4758

Accuracy for Entropy Purity Equation

Max Depth	Training Accuracy	Testing Accuracy
1	0.8912	0.1484
2	0.8886	0.1514
3	0.8942	0.1456
4	0.9058	0.1360
5	0.9150	0.1326
6	0.9238	0.1274
7	0.9302	0.1266
8	0.9352	0.1248
9	0.9412	0.1246
10	0.9478	0.1232
11	0.9530	0.1220
12	0.9566	0.1204
13	0.9606	0.1194
14	0.9644	0.1188
15	0.9672	0.1176
16	0.9704	0.1176

Accuracy for Majority Error Purity Equation

Max Depth	Training Accuracy	Testing Accuracy
1	0.8912	0.1484
2	0.8934	0.1536
3	0.8912	0.1444
4	0.8998	0.1292
5	0.9086	0.1244
6	0.9174	0.1226
7	0.9230	0.1196
8	0.9290	0.1188
9	0.9344	0.1186
10	0.9408	0.1186
11	0.9484	0.1182
12	0.9538	0.1178
13	0.9610	0.1166
14	0.9662	0.1160
15	0.9694	0.1162
16	0.9708	0.1160

Accuracy for Gini Index Purity Equation

(c) from the tables, we can see that testing accuracy maxes out around depth 1 or 2, then immediately overfits. replacing unkown with the most common value seems to massively ruin the testing accuracy, especially when not using entropy