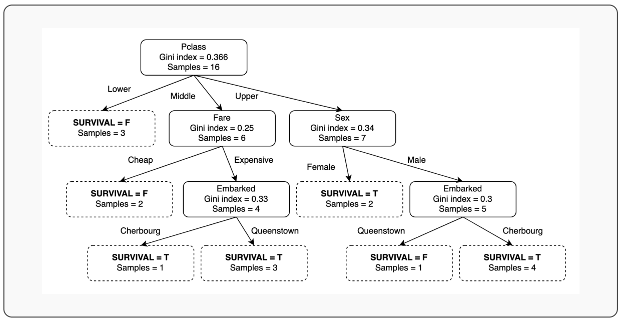
Homework 2

Automated Learning and Data Analysis
Team 1
Madison Garity, *mngarity*
Nithika Aduri, *naduri*
Rahul Yedida, *ryedida*

March 1, 2021

1 Solution to Problem 1a

1.1 Summary



1.2 Work

We have, Gini impurity is given by

$$Gini = 1 - \sum p_i^2$$

The data has 16 rows, split equally between the two classes. Therefore, the Gini index is maximum, equal to 0.5.

We can split the data using four attributes. In each case, we compute the Gini index.

(a) **Pclass:** See Table 1.

Pclass	**T**	**F**
Lower	0	3
Middle	3	3
Upper	5	2

Table 1: Gini index on split by pclass.

The Gini impurity for “Lower” is $1-\left(\frac{0}{9}+\frac{9}{9}\right)=0$. For “Middle”, the Gini impurity is 0.5, since it is an equal split. For “Upper”, the Gini impurity is $1-\left(\frac{25}{49}+\frac{4}{49}\right)=0.408$.

The Gini index, then, is $\frac{3}{16}\cdot 0+\frac{6}{16}\cdot\frac{1}{2}+\frac{7}{16}\cdot 0.408 =$ **0.366**.

(b) **Sex:** See Table 2. This split is not very convincing, since the Gini impurities are all 0.5. Therefore, the Gini index of this will be higher, i.e., *worse* than Pclass, and this can therefore not be the split criterion. We do not need to compute its Gini index.

Sex	**T**	**F**	**Total**
Female	3	3	6
Male	5	5	10

Table 2: Gini impurity computation of Sex.

(c) **Embarked:** See Table 3.

Embarked	**T**	**F**	**Total**
Cherbourg	5	3	8
Queenstown	3	5	8

Table 3: Gini impurity computation of Embarked.

The Gini impurity for “Cherbourg” is $1-\left(\frac{25}{64}+\frac{9}{64}\right)=0.47$. This is the same for Queenstown. The Gini index, therefore, since the classes are balanced, is also **0.47**.

(d) **Fare:** See Table 4.

Fare	**T**	**F**	**Total**
Cheap	1	5	6
Expensive	7	3	10

Table 4: Gini impurity computation of Fare.

The Gini impurity for “Cheap” is $1-\left(\frac{1}{36}+\frac{25}{36}\right)=0.28$, and for “Expensive” is $1-\left(\frac{49}{100}+\frac{9}{100}\right)=0.42$. Therefore, the Gini index is $\frac{6}{16}\cdot 0.28+\frac{10}{16}\cdot 0.42=0.3666$.

Because the lowest Gini index is for Pclass, we split based on that. For the samples whose Pclass attribute value is “Lower”, we have a leaf node, F.

For Pclass = Middle, we have 2 possible splits (since all the samples have Sex = Female, there is no possible gain from splitting on it):

(a) **Embarked:** See Table 5.

Embarked	**T**	**F**
Cherbourg	1	2
Queenstown	2	1

Table 5: Gini impurity computation of Embarked.

Cherbourg and Queenstown have Gini impurity 0.44, so the Gini index is 0.44.

(b) **Fare:** See Table 6.

Fare	**T**	**F**
Cheap	0	2
Expensive	3	1

Table 6: Gini impurity computation of Fare.

The Gini impurities for Cheap and Expensive are 0 and 0.375 respectively. Therefore, the Gini index is 0.25, and we split based on Fare.

This yields a leaf node for Fare = Cheap; for Expensive, we can again only split based on Embarked, yielding two leaf nodes.

For Pclass = Upper, we can split on 3 attributes:

(a) **Sex:** See Table 7

Sex	**T**	**F**
Male	3	2
Female	2	0

Table 7: Gini impurity computation of Sex.

These have Gini impurities 0.48 and 0 respectively. The Gini index is therefore 0.34.

(b) **Embarked:** See Table 8.

These have Gini impurities 0.32 and 0.5 respectively. Therefore the Gini index is 0.37.

(c) **Fare:** See Table 9.

These have Gini impurities 0.32 and 0.5. Therefore, the Gini index is 0.37, and we split based on Sex.

Finally, for the last subset, i.e., Pclass = Upper and Sex = Male, we can split by Fare or Embarked. Splitting by Fare leads to 2:1 and 1:1 splits, yielding a Gini index 0.47; splitting by Embarked yields 3:1 and 1:0 splits, yielding a Gini index of 0.3.

Embarked	**T**	**F**
Cherbourg	4	1
Queenstown	1	1

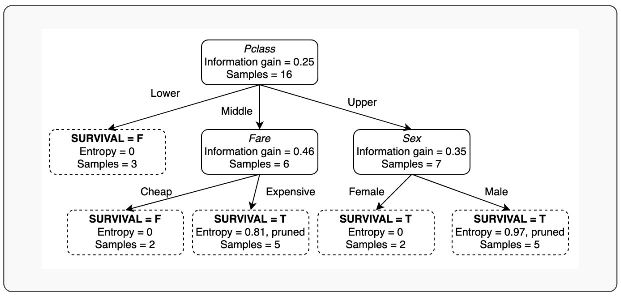
Table 8: Gini impurity computation of Embarked.

Fare	**T**	**F**
Cheap	1	1
Expensive	4	1

Table 9: Gini impurity computation of Fare.

2 Solution to Problem 1b

2.1 Summary



2.2 Work

Information gain is given by

$$I = H(N) - \sum_{i=1}^{k} \frac{|C_i|}{|N|} \cdot H(C_i)$$

for a parent node N and children C_i.

At the root node, the classes are equally split, so the entropy is 1.

(a) **Pclass:** See Table 10.

Pclass	**T**	**F**	**Total**
Lower	0	3	3
Middle	3	3	6
Upper	5	2	7

Table 10: Entropy computation of Pclass.

For Pclass = Lower, the entropy is 0. For Pclass = Middle, the entropy = 1. For Pclass = Upper, the entropy is 0.86. Therefore, the information gain is $1 - \left(0 + \frac{6}{16} \cdot 1 + \frac{7}{16} \cdot 0.86\right) = 0.249$.

(b) **Sex:** See Table 11.

This has the most even split, so the information gain is 0.

Sex	T	F	Total
Female	3	3	6
Male	3	3	10

Table 11: Entropy computation of Sex.

Embarked	T	F	Total
Cherbourg	5	3	8
Queenstown	3	5	8

Table 12: Entropy computation of Embarked.

(c) **Embarked:** See Table 12.

The entropy for each is $-\frac{5}{8}\log_2\frac{5}{8} - \frac{3}{8}\log_2\frac{3}{8} = 0.95$. The information gain, therefore, is $1 - 0.95 = 0.05$.

(d) **Fare:** See Table 13.

Fare	T	F	Total
Cheap	1	5	6
Expensive	7	3	10

Table 13: Entropy computation of Fare.

The entropy for Cheap is $-\frac{1}{6}\log_2\frac{1}{6} - \frac{5}{6}\log_2\frac{5}{6} = 0.65$. The entropy for expensive is $-\frac{7}{10}\log_2\frac{7}{10} - \frac{3}{10}\log_2\frac{3}{10} = 0.88$. Therefore the information gain is $1 - \left(\frac{6}{16}\cdot 0.65 + \frac{10}{16}\cdot 0.88\right) = 0.21$.

Therefore, we split by Pclass. Pclass = Lower is a leaf node whose output is F. For Pclass = Middle, the first two columns (Sex and Embarked) yield 2-1 splits for both classes; therefore, the information gain for these is the same. Because there are 3 samples of each class, the parent here has entropy 1. This gives us the information gain: $1 - \left(\frac{3}{6}\cdot 0.92 + \frac{3}{6}\cdot 0.92\right) = 0.08$ where 0.92 is the entropy of a 2-1 split (since $-\frac{2}{3}\log_2\frac{2}{3} - \frac{1}{3}\log_2\frac{1}{3} = 0.92$). For the split based on Fare, see Table 14.

Clearly, Cheap has entropy 0, while Expensive has entropy $-\frac{3}{4}\log_2\frac{3}{4} - \frac{1}{4}\log_2\frac{1}{4} = 0.81$. Therefore, the information gain from this split is $1 - \frac{4}{6}\cdot 0.81 = 0.46$. Therefore, we split by Fare.

For Pclass = Upper, splitting by Sex gives us Table 15.

These rows have entropy values 0 and $-\frac{3}{5}\log_2\frac{3}{5} - \frac{2}{5}\log_2\frac{2}{5} = 0.97$. Therefore, the information gain is $1 - \frac{5}{7}\cdot 0.91 = 0.35$.

Splitting by Embarked gives us Table 16.

Queenstown's entropy is 1, and Cherbourg's is 0.72. This gives the information gain $1 - \frac{2}{7}\cdot 1 - \frac{5}{7}\cdot 0.72 = 0.2$. Since Fare has the same distribution of classes, it has the same information gain, and therefore we split by Sex. On doing so, we get a leaf node for females.

Fare	T	F	Total
Cheap	0	2	2
Expensive	3	1	4

Table 14: Entropy computation of Fare.

Sex	T	F	Total
Male	2	0	2
Female	3	2	5

Table 15: Entropy computation of Sex.

At this stage, we have reached depth = 2, so we stop. The tree is shown in Figure 1.

3 Solution to Problem 1c

3.1 Summary

The trees differ for one specific subtree. An example that would be classified differently would have Pclass = Upper, Sex = Male, and Embarked = Queenstown.

3.2 Work

The trees differ only in a minor way, i.e., for samples with Pclass = Upper, Sex = Male. The tree based on Gini index splits it further, but the tree based on information gain, having been pruned, does not split it further. Therefore, an example with Pclass = Upper, Sex = Male, and Embarked = Queenstown would have different predictions by these two trees.

4 Solution to Problem 1d

On the training set, the tree based on Gini index performs better (accuracy = 0.875 vs 0.8125). On a test set, we cannot say, since we do not have access to the test set; however, we can optimistically say the tree

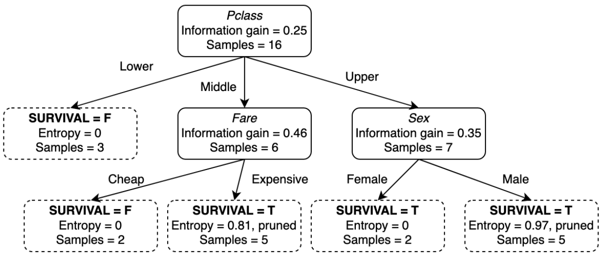


Figure 1: Decision tree for 1b

Embarked	T	F	Total
Cherbourg	4	1	5
Queenstown	1	1	2

Table 16: Entropy computation of Embarked.

based on Gini index will likely score better in a test set.

5 Solution to Problem 2

5.1 Summary

(a) Accuracy = 0.81, precision = 0.83, recall = 0.71, F-1 = 0.77, error rate = 0.38.

(b) Optimistic training error before and after pruning is 0.32 and 0.43 respectively, therefore we should not prune the tree.

(c) Pessimistic training error before pruning is 0.436, so we should prune the tree.

(d) Error rate = 0.25; the original tree was overfitting.

5.2 Work

(a) Table 17 shows the confusion matrix.

	N	Y
N	8	2
Y	1	5

Table 17: Confusion matrix

Therefore, the accuracy is 0.8125 (or, 81.25%), the precision is 0.833 (83.3%), the recall is 0.714 (71.4%), and the F1-score is 0.769 (76.9%). The table is 6Y 10N, so the error rate is $\frac{6}{16} = 0.375$.

(b) The optimistic training error is $\frac{9}{28} = 0.32$. If we prune the node, we get Y12 N16, making the optimistic training error $\frac{12}{28} > \frac{9}{28}$. Therefore, the tree should not be pruned.

(c) The pessimistic training error is $\frac{9+0.8\cdot 4}{28} = \frac{12.2}{28} = 0.436$. Therefore, in this case, the tree should be pruned.

(d) The confusion matrix is shown in Table 18. This yields a total of 12N 4Y, so the error rate is 0.25. Since the test error is lower post-pruning, the original tree was overfitting.

N	Y	
N	8	4
Y	1	3

Table 18: Caption

6 Solution to Problem 3

6.1 Summary

(a)

```
[[ 0.  ,  2.85,  4.51,  3.36,  7.67,  6.46,  3.83,  6.01,  2.97],
 [ 2.85,  0.  ,  4.75,  0.88,  7.63,  3.84,  3.72,  3.23,  2.64],
 [ 4.51,  4.75,  0.  ,  4.25,  3.16,  5.68,  7.9 ,  5.97,  2.12],
 [ 3.36,  0.88,  4.25,  0.  ,  6.96,  3.12,  4.6 ,  2.67,  2.22],
 [ 7.67,  7.63,  3.16,  6.96,  0.  ,  7.23, 11.01,  7.88,  5.12],
 [ 6.46,  3.84,  5.68,  3.12,  7.23,  0.  ,  7.13,  1.  ,  4.42],
 [ 3.83,  3.72,  7.9 ,  4.6 , 11.01,  7.13,  0.  ,  6.26,  5.9 ],
 [ 6.01,  3.23,  5.97,  2.67,  7.88,  1.  ,  6.26,  0.  ,  4.41],
 [ 2.97,  2.64,  2.12,  2.22,  5.12,  4.42,  5.9 ,  4.41,  0.  ]]
```

(b) (i) 0.5

(ii) 0.33

(iii) 0.33

(c) Holdout

6.2 Work

(a) Using a program, the distance matrix is

```
[[ 0.  ,  2.85,  4.51,  3.36,  7.67,  6.46,  3.83,  6.01,  2.97],
 [ 2.85,  0.  ,  4.75,  0.88,  7.63,  3.84,  3.72,  3.23,  2.64],
 [ 4.51,  4.75,  0.  ,  4.25,  3.16,  5.68,  7.9 ,  5.97,  2.12],
 [ 3.36,  0.88,  4.25,  0.  ,  6.96,  3.12,  4.6 ,  2.67,  2.22],
 [ 7.67,  7.63,  3.16,  6.96,  0.  ,  7.23, 11.01,  7.88,  5.12],
 [ 6.46,  3.84,  5.68,  3.12,  7.23,  0.  ,  7.13,  1.  ,  4.42],
 [ 3.83,  3.72,  7.9 ,  4.6 , 11.01,  7.13,  0.  ,  6.26,  5.9 ],
 [ 6.01,  3.23,  5.97,  2.67,  7.88,  1.  ,  6.26,  0.  ,  4.41],
 [ 2.97,  2.64,  2.12,  2.22,  5.12,  4.42,  5.9 ,  4.41,  0.  ]]
```

(b) We have $k = 1$.

(i) For the last four rows, we scan the last four rows of the distance matrix (until we hit the sixth column). The nearest neighbors to the last four rows are samples 4 (dist = 3.12), 2 (dist = 3.72), 4 (dist = 2.67), and 3 (dist = 2.12). These have classes +, -, +, - respectively. However, the actual labels are -, +, +, -. Therefore, the confusion matrix is

```
[[ 1.   1.
   1.   1. ]]
```

The accuracy is 0.5.

(ii) We do this in 3 splits. First, we train on samples 1-6 and test on 7-9. For this, the nearest neighbors to samples 7-9 (from samples 1-6) are 2 (dist = 2.67), 6 (dist = 1), and 3 (dist = 2.12). This yields all three predictions -. The confusion matrix is

```
[[ 1.    0.
   2.    0.  ]]
```

The testing accuracy is therefore 0.33.

For the second fold, we train on 1-3 and 7-9, and test on 4-6. The nearest samples are 9 (dist = 2.22), 3 (dist = 3.16), and 8 (dist = 1). These have classes -, -, + respectively. The confusion matrix is

```
[[ 1.   1.
   1.   0. ]]
```

Therefore, the testing accuracy is 0.33.

For the third fold, we train on samples 4-9 and test on 1-3. The nearest samples are 9 (dist = 2.97), 4 (dist = 0.88), and 5 (dist = 3.16). These have classes -, +, - respectively. The confusion matrix is

```
[[ 1.   1.
   1.   0. ]]
```

Therefore the test accuracy is 0.33. This gives our overall accuracy = $\frac{0.33+0.33+0.33}{3} = 0.33$.

(iii) For leave-one-out cross-validation, each sample becomes its own fold. For brevity, we used `np.argmin` to compute the nearest neighbors of each fold. These are, respectively, 2, 4, 9, 2, 3, 8, 2, 6, 3. These have classes -, +, -, -, -, +, -, -, -. These predictions are only true for samples 3, 5, and 9, so the test accuracy is 0.33.

(c) The validation accuracy is 0 for the holdout method. The other two approaches are both k-fold cross-validation methods (with LOOCV being a special case with $k = n$), and use an average across all the folds. Therefore, it is impossible for the validation accuracy to be 0, since the validation accuracy scores on all the folds would have to be zero.

7 Solution to Problem 4

7.1 Summary

(a) C0

(b) C0

(c) C1

(d) C0

7.2 Work

We have,

$$
\begin{aligned}
P(Class = C0) &= 0.5 \\
P(Class = C1) &= 0.5 \\
P(Gender = M|Class = C0) &= 0.6 \\
P(Gender = F|Class = C0) &= 0.4 \\
P(Gender = M|Class = C1) &= 0.4 \\
P(Gender = F|Class = C1) &= 0.6 \\
P(CarType = Luxury|Class = C0) &= 0.6 \\
P(CarType = Sports|Class = C0) &= 0.4 \\
P(CarType = Luxury|Class = C1) &= 0.6 \\
P(CarType = Sports|Class = C1) &= 0.4 \\
P(AgeGroup = G1|Class = C0) &= 0.6 \\
P(AgeGroup = G2|Class = C0) &= 0.4 \\
P(AgeGroup = G1|Class = C1) &= 0.8 \\
P(AgeGroup = G2|Class = C1) &= 0.2
\end{aligned}
$$

(a) For Gender = M, Car Type = Luxury, and Age Group = G1, we compute the class-conditional probabilities as follows. We use S to denote the sample.

$$
\begin{aligned}
P(Class = C0|S) \propto\ & P(Gender = M|Class = C0) \cdot \\
& P(CarType = Luxury|Class = C0) \cdot \\
& P(AgeGroup = G1|Class = C0) \cdot \\
& P(Class = C0) \\
& = 0.6 \cdot 0.6 \cdot 0.6 \cdot 0.5 \\
P(Class = C1|S) \propto\ & P(Gender = M|Class = C1) \\
& \cdot P(CarType = Luxury|Class = C1) \\
& \cdot P(AgeGroup = G1|Class = C1) \\
& \cdot P(Class = C0) \\
& = 0.4 \cdot 0.6 \cdot 0.8 \cdot 0.5 \\
& < P(Class = C0|S)
\end{aligned}
$$

Therefore we predict class **C0** for this sample.

(b) For Gender = M, Car Type = Sports, Age Group = G2, we have:

$$\begin{aligned}
P(Class = C0|S) &\propto P(Gender = M|Class = C0) \\
&\cdot P(CarType = Sports|Class = C0) \\
&\cdot P(AgeGroup = G2|Class = C0) \\
&\cdot P(Class = C0) \\
&= 0.6 \cdot 0.4 \cdot 0.4 \cdot 0.5 \\
P(Class = C1|S) &\propto P(Gender = M|Class = C1) \\
&\cdot P(CarType = Sports|Class = C1) \\
&\cdot P(AgeGroup = G2|Class = C1) \\
&\cdot P(Class = C0) \\
&= 0.4 \cdot 0.4 \cdot 0.2 \cdot 0.5 \\
&< P(Class = C0|S)
\end{aligned}$$

Therefore we predict class **C0** for this sample.

(c) For Gender = F, Car Type = Sports, Age Group = G1, we have:

$$\begin{aligned}
P(Class = C0|S) &\propto P(Gender = F|Class = C0) \\
&\cdot P(CarType = Sports|Class = C0) \\
&\cdot P(AgeGroup = G1|Class = C0) \\
&\cdot P(Class = C0) \\
&= 0.4 \cdot 0.4 \cdot 0.6 \cdot 0.5 \\
P(Class = C1|S) &\propto P(Gender = F|Class = C1) \\
&\cdot P(CarType = Sports|Class = C1) \\
&\cdot P(AgeGroup = G1|Class = C1) \\
&\cdot P(Class = C0) \\
&= 0.6 \cdot 0.4 \cdot 0.8 \cdot 0.5 \\
&> P(Class = C0|S)
\end{aligned}$$

Therefore we predict class **C1** for this sample.

(d) For Gender = F, Car Type = Luxury, and Age Group = G2, we have:

$$\begin{aligned}
P(Class = C0|S) &\propto P(Gender = F|Class = C0) \\
&\cdot P(CarType = Luxury|Class = C0) \\
&\cdot P(AgeGroup = G2|Class = C0) \\
&\cdot P(Class = C0) \\
&= 0.4 \cdot 0.6 \cdot 0.4 \cdot 0.5 \\
P(Class = C1|S) &\propto P(Gender = F|Class = C1) \\
&\cdot P(CarType = Luxury|Class = C1) \\
&\cdot P(AgeGroup = G2|Class = C1) \\
&\cdot P(Class = C0) \\
&= 0.6 \cdot 0.6 \cdot 0.2 \cdot 0.5 \\
&< P(Class = C0|S)
\end{aligned}$$

Therefore, we predict class **C0** for this sample.