Assignment #5

Pouya Akbarzadeh
CS412, Data Mining
April 20, 2021

Problem 1

a. Gain ratio is used to reduce bias by accounting for the size and number of branches of a decision tree while making choice.

$$GainRatio(S,A) = \frac{Gain(S,A)}{SplitInformation(S,A)}$$

Above, we can see the mathematical notation of Gain Ratio. Gain, information gain, is the difference between the original information requirements and the new requirements.

$$Gain(A) = Info(D) - Info_A(D).$$

Where Gain(A) would be representing how much would be gained by branching on A. Furthermore Info(D) and $Info_A(D)$ can be defined as

$$Info_A(D) = \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times Info(D_j). \qquad Info(D) = -\sum_{i=1}^{m} p_i \log_2(p_i)$$

In these equations p_i is the probability that an arbitrary tuple in D belongs to a class C_i and is estimated by the following

$$|C_{i,D}|/|D|$$

The log base 2 is there due to information being encoded in bits. To put it simply Info(D) is the average amount of information need to identify the class label of a tuple or entropy of D. In $Info_A(D)$ we can see the $|D_j|/|D|$ ratio is setting a weigh for the j^{th} partition.

Now that we have the numerator explained, let's look at the denominator. The equation here is very similar to what we saw for $info_A(D)$ and info(D). What we see is the information generated by splitting the training data set of D into v partitions. Where v is the outcome of test attribute A.

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times \log_2\left(\frac{|D_j|}{|D|}\right).$$

Now to put it all together, we can say that Gain Ratio is the measure of the information with respect to classification that is acquire based on the same partitioning.

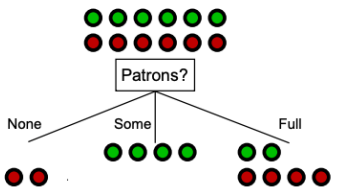
Gini impurity measures the likelihood of a randomly picked datapoint from our data set be incorrectly classified by randomly classifying it according to the class of distribution in the dataset.

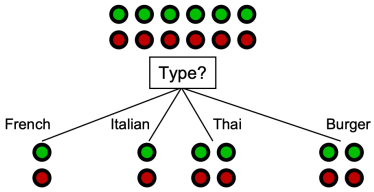
$$\sum_{k \neq i} p_k = 1 - p_i \qquad Gini(D) = 1 - \sum_{i=1}^{m} p_i^2,$$

Now that we have defined the two, we should not that they do have a relationship. That relationship can be summarized by stating that they both measures aim to quantify to what extent the impurity will be reduced if we split the current node based on given/needed attributes. They slightly differ because Information gain is measured by the average amount of information need to identify the class label of a tuple while Gini impurity is related to misclassification by how likely a randomly chosen tuple will be misclassified if it were to be assigned to a random class label.

To add on, if the tree is strictly binary it is usually driven by the attribute selection measures. One of these attribute selections measures that can result in a binary tree is Gini impurity. However, information gain does not.

b.





$GI_{NONE} = 1 - [(\frac{2}{2+0})^2 + \left(\frac{0}{0+2}\right)^2] \rightarrow 0$

$GI_{SOME} = 1 - [(\frac{4}{4+0})^2 + \left(\frac{0}{0+4}\right)^2] \rightarrow 0$

$GI_{FULL} = 1 - [(\frac{2}{2+4})^2 + \left(\frac{4}{4+2}\right)^2] \rightarrow 0.444$

$GI_{FRENCH} = 1 - [(\frac{1}{1+1})^2 + \left(\frac{1}{1+1}\right)^2] \rightarrow 0.5$

$GI_{ITALIAN} = 1 - [(\frac{1}{1+1})^2 + \left(\frac{1}{1+1}\right)^2] \rightarrow 0.5$

$GI_{THAI} = 1 - [(\frac{2}{2+2})^2 + \left(\frac{2}{2+2}\right)^2] \rightarrow 0.5$

$GI_{BURGER} = 1 - [(\frac{2}{2+2})^2 + \left(\frac{2}{2+2}\right)^2] \rightarrow 0.5$

Partons GI = $\frac{6}{12} * (0.444) = .222$

Type GI = $\frac{2}{12} * 0.5 + \frac{2}{12} * 0.5 + \frac{4}{12} * 0.5 + \frac{4}{12} * 0.5 = 0.5$

c. We will first look at patrons. In the image above we can conclude that entropy is 1. This is due to the following calculation $-\left(\frac{6}{12}\right) log_2\left(\frac{6}{12}\right) + -\left(\frac{6}{12}\right) log_2\left(\frac{6}{12}\right)$ which results in 0.5 + 0.5 = 1.

We then will form the following table

	Yes	No	Total
Full	2	4	6
Some	4	0	4
None	0	2	2

We then go ahead and calculate the E(Full), E(Some), and E(None) respectively.

E(Full) = $-\left(\frac{1}{3}\right) log_2\left(\frac{1}{3}\right) - (\frac{2}{3}) log_2\left(\frac{2}{3}\right) = 0.918$

E(Some) = $-\left(\frac{6}{12}\right) log_2\left(\frac{6}{12}\right) - (\frac{6}{12}) log_2\left(\frac{6}{12}\right) = 0$

E(None) = $\mathbf{0} - 1\, log_2(1) = 0$

Info = $0.918 * \frac{6}{12} = 0.459$

Gain = 1 – 0.459 = 0.541

Split Info = $-\left(\frac{2}{12}\right) log_2\left(\frac{2}{12}\right) - \left(\frac{4}{12}\right) log_2\left(\frac{4}{12}\right) - \left(\frac{6}{12}\right) log_2\left(\frac{6}{12}\right) = 1.459$

Gain Ratio = Gain/SplitInfo = 0.3708

Just like the part for patron E(S) is 1.

	Yes	No	Total
French	1	1	2
Italian	1	1	2
Thai	2	2	4
Burger	2	2	4

E(French), E(Italian), E(Thai), E(Burger) = $-\frac{1}{2} log_2\left(\frac{1}{2}\right) - \frac{1}{2} log_2\left(\frac{1}{2}\right) = 1$

Info = $1\left(\frac{2}{12}\right) + 1\left(\frac{2}{12}\right) + 1\left(\frac{4}{12}\right) + 1\left(\frac{4}{12}\right) = 1$

Gain = 0

Gain Ratio = 0

Problem 2

To properly answer these questions we need to explore Bayes' Theorem.

We know that $P(A|B) = \frac{P(A \cap B)}{P(B)}$ and $P(B|A) = \frac{P(B \cap A)}{P(A)}$. We can see that the numerator $P(A \cap B)$ and $P(B \cap A)$ are equivalent. We can re-write $P(B|A) = \frac{P(A|B)*P(B)}{P(A)}$. This is equation is Bayes' Theorem.

a.

We can assume the following based on the given information.

$P(C_1|h_1) = 0\%$ $P(C_1|h_2) = 50\%$ $P(C_1|h_3) = 100\%$

$P(C_1) = P(C_1|h_1) * P(h_1) + P(C_1|h_2) * P(h_2) + P(C_1|h_3) * P(h_3) =$

$= 0 * \frac{1}{4} + \frac{1}{2} * \frac{1}{2} + 1 * \frac{1}{4} = ½$

Based on Bayes we can say $P(B|A) = \frac{P(A|B)*P(B)}{P(A)}$

Thus, we will calculate as seen in the following:

$$P(h_1|C_1) = \frac{P(C1|h1) * P(h1)}{P(C1)} = \frac{0*\frac{1}{4}}{\frac{1}{2}} = 0 = 0\%$$

$$P(h_2|C_1) = \frac{P(C1|h2) * P(h2)}{P(C1)} = \frac{\frac{1}{2}*\frac{1}{2}}{\frac{1}{2}} = \frac{1}{2} = 50\%$$

$$P(h_3|C_1) = \frac{P(C1|h3) * P(h3)}{P(C1)} = \frac{1*\frac{1}{4}}{\frac{1}{2}} = \frac{1}{2} = 50\%$$

b.

$P(C_2) = 1 - P(C_1) = \frac{1}{2}$

$$P(h_1|C_2) = \frac{P(C2|h1) * P(h1)}{P(C2)} = \frac{1*\frac{1}{4}}{\frac{1}{2}} = \frac{1}{2} = 50\%$$

$$P(h_2|C_2) = \frac{P(C2|h2) * P(h2)}{P(C2)} = \frac{\frac{1}{2}*\frac{1}{2}}{\frac{1}{2}} = \frac{1}{2} = 50\%$$

$$P(h_3|C_2) = \frac{P(C2|h3) * P(h3)}{P(C2)} = \frac{0*\frac{1}{4}}{\frac{1}{2}} = 0 = 0\%$$

$P(h_1|C_1, C_2) = P(h_1|C_1) * P(h_1|C_2) = 0 * \frac{1}{2} = 0 = 0\%$

$P(h_2|C_1, C_2) = P(h_2|C_1) * P(h_2|C_2) = \frac{1}{2} * \frac{1}{2} = .25 = 25\%$

$P(h_3|C_1, C_2) = P(h_3|C_1) * P(h_3|C_2) = \frac{1}{2} * 0 = 0 = 0\%$

Problem 3

a. We can assume two variables (A&B) are independent if P(A^B) = P(A)*P(B). Further more we can assume two variables are conditionally independent given C. If P(A, B | C) = P(A|C) * P(B|C). Keeping that in mind, we are drawing the tables in following ways.

Size	Good	Bad	P(G)	P(B)
Small	1	3	¼	½
Large	3	3	¾	½

Color	Good	Bad	P(G)	P(B)
Red	4	1	1	1/6
Green	0	5	0/4	5/6

Shape	Good	Bad	P(G)	P(B)
Circle	3	2	¾	1/3
Irregular	1	4	¼	2/3

		P(G) & P(B)
Good	4	2/5
Bad	6	3/5
Total	10	100%

We have 3 independent parameters. Shape, Color, and Size. For those 3 parameters it could be one of the options.

b.

Small = 40% Large = 60%
Red = 50% Green = 50%
Circle = 50% Irregular = 50%

c.

To answer this part we can use what we derived in question 2. So we know that we have 3 parameters. So for something to be "bad" the probability is the following.
P(Bad | Small, Red, Circle) = P(Small | Bad) * P(Red | Bad) * P(Circle | Bad) * P(No)
= 0.5 * 0.2 * 0.33333 * 0.6 = 0.16

Same thing, instead we switch "bad" for "good".
P(Good | Small, Red, Circle) = P(Small | Good) * P(Red | Good) * P(Circle | Good) * P(Good) * 1/p(x)

= 0.25 * 1.00 * 0.75 * 0.4 = 0.075

Based on our calculation it is clear that Good | X is bigger than Bad | X thus the class picked will be Yes. Assuming all calculation above are correct.

Problem 4

a. Hefty amount of decision trees is created by sampling data. What makes a difference between random forests and decision tree is the fact that each node is split by the best of random subset of variables instead of the best of all the variables. So each individual is classified by each tree and the most common outcome is used for the classification. **The more features (m) we have, the more the impurity of the split we have. The deeper the tree (higher value of d) the more data we will be able to observe.**

b. Boot strapping allows each of our decision trees in a random forest to be unique. Knowing that that probability of each sample being selected is p() = 1/n. Thus probability of not choosing it is just subtracting one from it. (1 - p()). Knowing data amount is big, we can then assume that our probability of not being chosen will be exponential thus having a form similar to $\left(1-\left(\frac{1}{n}\right)\right)^{n}$ into e^{-1} which is around 0.368. We now subtract 1, 1-0.368 = 0.632. Thus we can say that we have 63.2% of n to be the expected number of unique samples or a subset of the original set of n samples.

c. I agree with the professor. Here is why, having m=1 will allow us to decrease the number of attributes per tree. Doing that allows us to decrease the correlation between the trees and increasing the overall strength of the forest. Furthermore, I referenced mljar.com and there I found that "we can measure how each feature decrease the impurity of the split (the feature with highest decrease is selected for internal node). For each feature we can collect how on average it decreases the impurity". Therefore, we can conclude that by decreasing it and observing each feature independently we can better see its impacts on our data. Therefore, I once again state that I am in agreement with Professor.

Problem 5 (Extra Credit).

		Prediction		
		Spam	Not Spam	Total
Truth	Spam	2588	412	3000
	Not Spam	46	6954	7000
	Total	2634	7366	10000

a. Sensitivity = $\frac{TP}{TP+FN} = \frac{2588}{2588+46} = 0.9825$

b. Specificity = $\frac{TN}{TN+FP} = \frac{6954}{6954+412} = 0.9440$

c. Accuracy = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{2588+6954}{2588+6954+412+46} = 0.9542$

d. Precision = $\frac{TP}{TP+FP} = \frac{2588}{2588+412} = 0.8626$

e. Recall = $\frac{TP}{TP+FN} = \frac{2588}{2588+46} = 0.9825$

f. F1 = $\frac{2*Precission*Recall}{Precission+Recall} = \frac{2*0.8626*0.9825}{0.8626+0.9825} = 0.9186$

g. To answer this question I would like to first explain my personal experience with e-mail and email filtering. I have had a much better experience digging for a confirmation email in my spam folder rather than getting notifications and cluttering my inbox with spam. THUS, I believe that high recall, low precision is best. This means that we will have more emails flagged as spam (false positives), but much smaller number of spams (false negatives). Thus, I think high recall, low precision would be the most effective.