1 Problem

Commonly for all subsections of this problem, I would like to note down two equations that hold for binary classifier:

$$p(Y = 0|X = x) + p(Y = 1|X = x) = 1 \tag{1.1}$$

$$\begin{aligned} p(Y \neq 0|X = x) = p(Y = 1|X = x) \\ p(Y \neq 1|X = x) = p(Y = 0|X = x) \end{aligned} \tag{1.2}$$

While the equation 1.1 follows from the total/marginal probability, 1.2 follows from the definition of binary classifier.

1.1 Solution

From the given definition for f^*, we know that $f^*(x) = 1$ only when $p(Y = 1|X = x) \geq \frac{1}{2}$. Combining this with equation 1.1, we can rewrite it as:

$$f^*(x) = \left\{ \begin{array}{l} 1, \text{if } p(Y = 0|X = x) \leq \frac{1}{2} \\ 0, \text{otherwise} \end{array} \right\}$$

and noting that this is already in indicator function form with the first condition being the indicator variable, we can write:

$$f^*(x) = \mathbb{1}[p(Y = 0|X = x \leq \frac{1}{2}]$$

1.2 Solution

From the definition of indicator function:

$$\mathbb{1}[f(x) = 0] = \left\{ \begin{array}{l} 1, \text{if } f(x) = 0 \\ 0, \text{otherwise} \end{array} \right\}$$

Note that if $f(x) \neq 0$, it must be that $f(x) = 1$ as f is a binary classifier. Given these are the only possibilities, We will show that the given equality for probability of error holds for both the cases where $f(x) = 0$ and $f(x) = 1$ separately and that should prove the equation.

For any x where $f(x) = 0$, the RHS of equation becomes:

$$\begin{aligned} &(1 - 2P(Y = 0|X = x))\mathbb{1}[f(x) = 0] + P(Y = 0|X = x) \\ &\quad = (1 - 2P(Y = 0|X = x)) + P(Y = 0|X = x) \text{ ...since } \mathbb{1}[f(x) = 0] = 1 \\ &\quad = 1 - P(Y = 0|X = x) \\ &\quad = P(Y = 1|X = x) \text{ ...from equation 1.1} \\ &\quad = P(Y \neq 0|X = x) \text{ ...from equation 1.2} \\ &\quad = P(Y \neq f(X)|X = x) \text{ ...since } f(x) = 0 \end{aligned}$$

Now, let's consider the case where $f(x) = 1$:

$$\begin{aligned}(1 - 2P(Y = 0|X = x))\mathbb{1}[f(x) = 0] + P(Y = 0|X = x)&\\ &= P(Y = 0|X = x) \text{ ...since } \mathbb{1}[f(x) = 0] = 0\\ &= P(Y \neq 1|X = x) \text{ ...from equation 1.2}\\ &= P(Y \neq f(X)|X = x) \text{ ...since } f(x) = 1\end{aligned}$$

Together, the above two complete the proof of the RHS expression for $P(Y \neq f(X)|X = x)$.

1.3 Solution

Following the expression for $P(Y \neq f(X)|X = x)$ from Problem 1.2, we have:

$$\begin{aligned}P(Y \neq f(X)|X = x) - P(Y \neq f^*(X)|X = x)&\\ &= (1 - 2P(Y = 0|X = x))\{\mathbb{1}[f(x) = 0] - \mathbb{1}[f^*(x) = 0]\}\\ &= (T_1)(T_2)\end{aligned}$$

where $T_1 = (1 - 2P(Y = 0|X = x))$ and $T_2 = \mathbb{1}[f(x) = 0] - \mathbb{1}[f^*(x) = 0]$.

Consider the scenario where $P(Y = 0|X = x) \leq \frac{1}{2}$; In this case, $T_1 \geq 0$. Also, from the definition of f^* (from Solution 1.1), we know that $f^*(x) = 1$ i.e., $\mathbb{1}[f^*(x) = 0] = 0$. Since the value of an indicator function is either 0 or 1 which are the only possible values for $\mathbb{1}[f(x) = 0]$, we can see that $T_2 \geq 0$ as well. Following this, the product $(T_1)(T_2)$ will also be non-negative.

Now let's switch to the other scenario i.e., $P(Y = 0|X = x) > \frac{1}{2}$; In this case, $T_1 < 0$. And from the definition of f^*, we have $f^*(x) = 0$ giving us $\mathbb{1}[f^*(x) = 0] = 1$. And in either the case where $\mathbb{1}[f(x) = 0]$ is 0 or 1, we can see that $T_2 \leq 0$. Together, we can see that the product $(T_1)(T_2) \geq 0$.

Combining both, we have for any binary classifier f:

$$P(Y \neq f(X)|X = x) - P(Y \neq f^*(X)|X = x) \geq 0.$$

1.4 Solution

This one follows from the definition of risk and the proof we established in Solution 1.3. From the definition of classifier risk, we have for any binary classifier f:

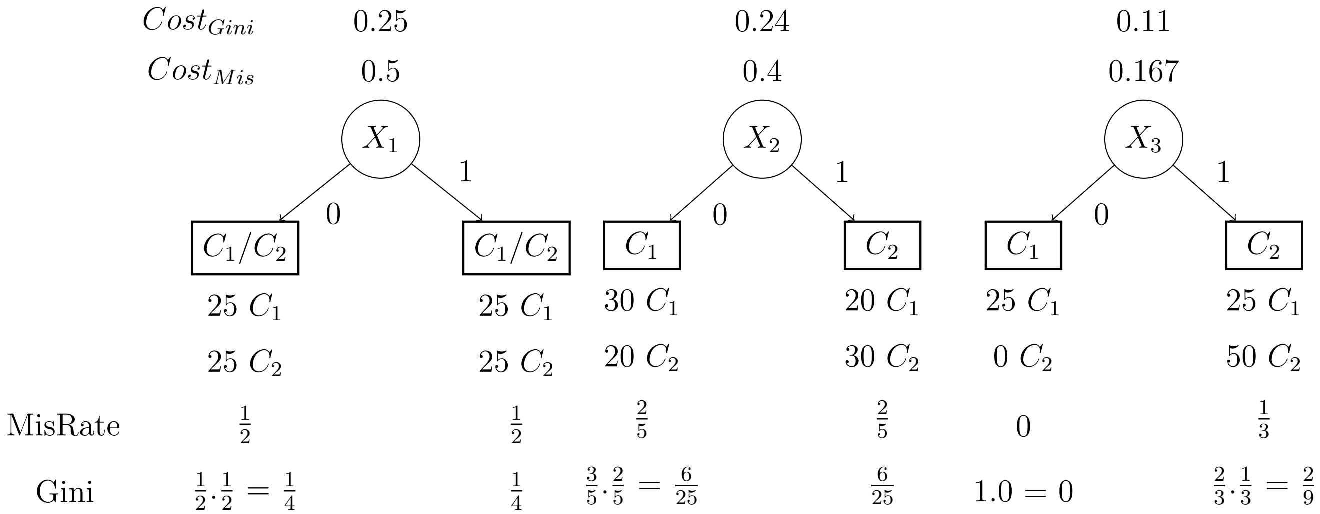
$$\begin{aligned}R(f) - R(f^*) &= \mathbb{E}_{x\sim P(x)}[P(f(X) \neq Y|X = x)] - \mathbb{E}_{x\sim P(x)}[P(f^*(X) \neq Y|X = x)]\\ &= \mathbb{E}_{x\sim P(x)}[P(f(X) \neq Y|X = x) - P(f^*(X) \neq Y|X = x)]\end{aligned}$$

From Solution 1.3, each term inside the expectation is ≥ 0. So the whole expression is ≥ 0. So for any binary classifier, $R(f) - R(f^*) \geq 0$ i.e., $R(f^*) \leq R(f)$.

2 Problem

2.1 Solution

Below diagram shows the cost of splitting for each input variable with mis-classification rate or Gini index as the impurity measure.



Each input variable is shown in a circle and the branches are indicated with the value of input variable for decision. Each box represents the classification based on given data (when there is a tie we show both the classes, the choice is arbitrary). The row below is the number of items which belong to C1 in that branch and the row below is the number of items that belong to C2. The row below that is the mis-classification rate for that specific leaf node / branch (#mis-classified/#total). The bottom row is showing the impurity measure based on Gini index for the same leaf nodes. The cost of each input node is shown on top of each circle in corresponding rows for Cost based on Gini index as impurity measure and mis-classification rate as impurity measure.

2.2 Solution

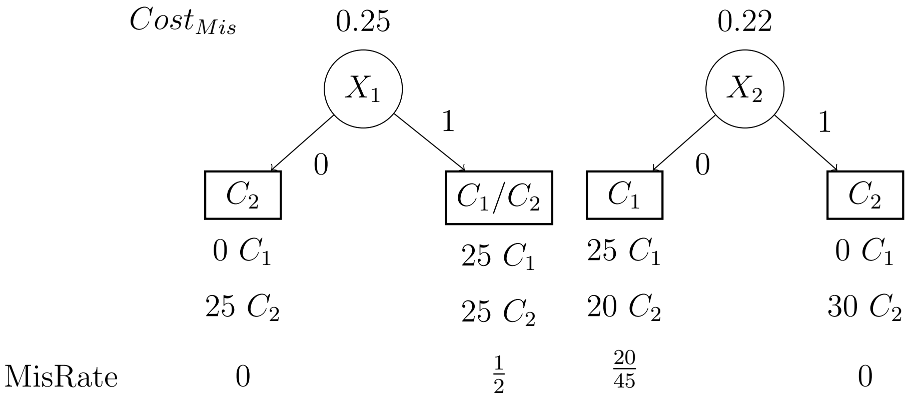
Since we want to minimize the cost, from the above diagram we want to split first on the input variable with least cost i.e., X_3 with $Cost_{Mis}$ being 0.167.

2.3 Solution

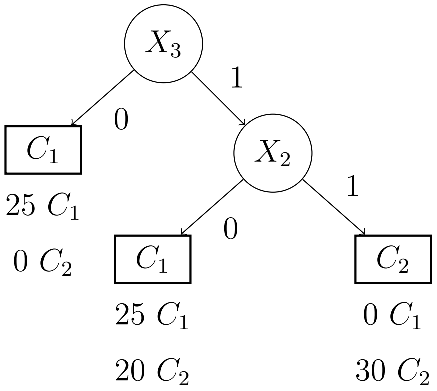
Given we already split first based on X_3, the left branch is a pure leaf node because all items belong to one class. We only need to further split the right node. Selecting the rows only for the right node (i.e., all rows where $X_3 = 1$) from the input data :

X_1	X_2	C_1	C_2
0	0	0	20
0	1	0	5
1	0	25	0
1	1	0	25

We need to repeat the same process like above for X_1 and X_2 with the data in the above table to determine the node with the least cost. The below diagram shows the mis-classification rates and cost based on the above table:



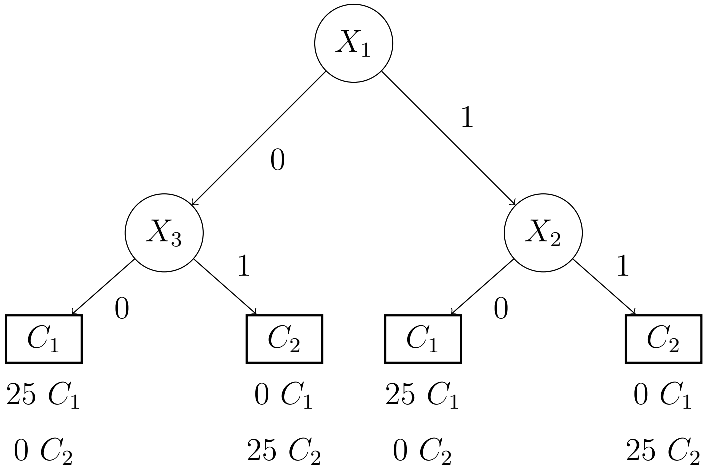
Based on the above, since the variable X_2 is a lower cost split, we should use that variable for 2^{nd} level. The complete decision tree will look like this:



And from the above diagram we can see that 20 instances are still mis-classified.

2.4 Solution

If we instead split on X_1 first, we need to follow the similar procedure for both the left and right branches (because neither of them are pure). I am showing the final tree obtained by using mis-classification rate as criteria for splitting the nodes:



Interestingly, no instances are mis-classified in this model.

2.5 Solution

On the given data, the decision tree in Solution 2.4 clearly performs better as there are no mis-classifications. This is clearly an example problem where greedy choice did not yield the optimal decision tree (atleast for the training data).

One thing to note is, Solution 2.3 is better in terms of depth of tree. All instances where $X_3 = 0$ are classified with only one attribute check. Compared to that, all instances require 2 attribute checks for classification in Solution 2.4.

3 Problem

3.1 Solution

The below equation shows the expression for $P(C|X)$:

$$\begin{aligned} P(C|X) &= \frac{P(X|C)P(C)}{P(X)} \\ &= \frac{P([X_1; X_2; X_3]|C)P(C)}{P[X_1; X_2; X_3]} \text{ ...expanding } X \\ &= \frac{P(X_1|C)P(X_2|C)P(X_3|C)P(C)}{P[X_1; X_2; X_3]} \text{ ...conditional independency assumption of naive bayes} \\ &= \frac{P(X_1|C)P(X_2|C)P(X_3|C)P(C)}{\sum_{C \in (C_1, C_2)} P(X_1|C)P(X_2|C)P(X_3|C)P(C)} \text{ ...marginal probability} \end{aligned}$$

3.2 Solution

From the given data, we have:

$$
\begin{aligned}
P(C_1) = P(C_2) &= \frac{50}{100} = \frac{1}{2} \\
P(X_1 = 0|C = C_1) &= \frac{25}{50} = \frac{1}{2};\ P(X_1 = 1|C = C_1) = \frac{1}{2} \\
P(X_2 = 0|C = C_1) &= \frac{30}{50} = \frac{3}{5};\ P(X_2 = 1|C = C_1) = \frac{2}{5} \\
P(X_3 = 0|C = C_1) &= \frac{25}{50} = \frac{1}{2};\ P(X_3 = 1|C = C_1) = \frac{1}{2} \\
P(X_1 = 0|C = C_2) &= \frac{25}{50} = \frac{1}{2};\ P(X_1 = 1|C = C_2) = \frac{1}{2} \\
P(X_2 = 0|C = C_2) &= \frac{20}{50} = \frac{2}{5};\ P(X_2 = 1|C = C_2) = \frac{3}{5} \\
P(X_3 = 0|C = C_2) &= \frac{0}{50} = 0;\ P(X_3 = 1|C = C_2) = 1
\end{aligned}
$$

3.3 Solution

The below table shows each possible input and the most probable label for that input using naive bayes model. For columns showing class probabilities for C_1 and C_2, we are only showing the values to which each probability is proportional to (this ignores the common denominators in both the expression for total probability of X and also the values of input feature probabilities for a given class making sure all the denominators are same.)

X_1	X_2	X_3	$P(C_1\|X) \approx$	$P(C_2\|X) \approx$	Predicted Label	Mis-classified instances
0	0	0	3	0	C_1	0
0	0	1	3	4	C_2	0
0	1	0	2	0	C_1	0
0	1	1	2	6	C_2	0
1	0	0	3	0	C_1	0
1	0	1	3	4	C_2	25
1	1	0	2	0	C_1	0
1	1	1	2	6	C_2	0

From the table, we can see that total mis-classification error is $\frac{25}{100} = 0.25$ or 25%.

3.4 Solution

The given data has an interesting pattern. If we treat X_1, X_2, X_3 as switches with On representing 1 and Off representing 0, the data is such that when $X_1 = 0$, the value of X_3

determines the instance class (C_1 for $X_3 = 0$ otherwise C_2). Similarly when $X_1 = 1$, the value of X_2 determines the instance class (C_1 for $X_2 = 0$ otherwise C_2). In other words, X_1 is the switch that decides which of X_2, X_3 is the operating switch for deciding the class. So there is a strong correlation between given input features and that's why the naive bayes assumption that input features are independent given a class label is inaccurate.

So we know that both X_2, X_3 are correlated to X_1. Indeed, it's probably a reasonable assumption that given X_1 and the class label, both X_2 and X_3 are independent. With this, we have to re-write the equation for probability of class as:

$$\begin{aligned} P(C|X) &= \frac{P(X|C)P(C)}{P(X)} \\ &= \frac{P([X_1; X_2; X_3]|C)P(C)}{P[X_1; X_2; X_3]} \text{ ...expanding } X \\ &= \frac{P(X_1|C)P(X_2|C, X_1)P(X_3|C, X_1)P(C)}{P[X_1; X_2; X_3]} \text{ ...based on correlations between input features} \\ &= \frac{P(X_1|C)P(X_2|C, X_1)P(X_3|C, X_1)P(C)}{\sum_{C \in (C_1, C_2)} P(X_1|C)P(X_2|C, X_1)P(X_3|C, X_1)P(C)} \text{ ...marginal probability} \end{aligned}$$

When we use the above equations, we can see that when calculating the probabilities, for a given input value, for the wrong class, one of the probabilities will be zero (For the inputs where $X_1 = 0$, the value $P(X_3|C, X_1)$ will be zero for the wrong class, similarly for the inputs where $X_1 = 1$, the value $P(X_2|C, X_1)$ will be zero for the wrong class). Hence the expected class will have higher probability and will be predicted by the model as the target class. So we will not mis-classify any of the training data and accuracy will be 100%.