Purdue University Duc Viet Le

CS 59000SA

Assignment 3

Problem 1. (**Decision Tree**) Let `IS` be the random variable denote if the student is interest in security or not, `CERIAS` be the random variable denotes if the student is in cerias or not, `CS555` denote if student took CS555, `CS526` denote if student took CS526.

1. Information Gain:

   (a) First we need to compute the entropy before split:

   $$H(\texttt{IS}) = -\sum_{i\in\{Y,N\}} p_i \log(p_i) = 0.9709 \tag{1}$$

   (b) For each of attribute, we compute:

   - Cerias student:

   $$\begin{aligned} H(\mathsf{IS}|\mathsf{CERIAS}) &= \frac{H(\mathsf{IS}|\mathsf{CERIAS}=Y) + H(\mathsf{IS}|\mathsf{CERIAS}=N)}{2} \\ &= \frac{-4/5\log(4/5) - 1/5\log(1/5) - 3/5\log(3/5) - 2/5\log(2/5)}{2} = 0.8464 \end{aligned} \tag{2}$$

   - CS555:

   $$\begin{aligned} H(\mathsf{IS}|\mathsf{CS555}) &= \frac{H(\mathsf{IS}|\mathsf{CS555}=Y) + H(\mathsf{IS}|\mathsf{CS555}=N)}{2} \\ &= \frac{-5/6\log(5/6) - 1/6\log(1/6) - 3/4\log(3/4) - 1/4\log(1/4)}{2} = 0.7306 \end{aligned} \tag{3}$$

   - CS526:

   $$\begin{aligned} H(\mathsf{IS}|\mathsf{CS526}) &= \frac{H(\mathsf{IS}|\mathsf{CS526}=Y) + H(\mathsf{IS}|\mathsf{CS526}=N)}{2} \\ &= \frac{-5/6\log(5/6) - 1/6\log(1/6) - 3/4\log(3/4) - 1/4\log(1/4)}{2} = 0.7306 \end{aligned} \tag{4}$$

   (c) Therefore, we can pick either $\mathsf{CS526}$ or $\mathsf{CS555}$ to split because it maximizes the information gain. I chose $\mathsf{CS555}$. We compute entropy for left subtree and right subtree:

   $$\begin{aligned} H_{left}(\mathsf{IS}|\mathsf{CS555}=N) &= -3/4\log(3/4) - 1/4\log(1/4) = 0.8112 \\ H_{righ}(\mathsf{IS}|\mathsf{CS555}=Y) &= -5/6\log(5/6) - 1/6\log(1/6) = 0.6500 \end{aligned} \tag{5}$$

   On the left subtree, for each of the last 2 attributes, we compute:

   - Cerias student:

   $$\begin{aligned} H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}) &= \frac{H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}=Y) + H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}=N)}{2} \\ &= \frac{-1/2\log(1/2) - 1/2\log(1/2) - 0\log(0) - 1\log(1)}{2} = 0.5 \end{aligned} \tag{6}$$

- CS526:

$$\begin{aligned} H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}) &= \frac{H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}=Y)+H(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}=N)}{2} \\ &= \frac{-1/2\log(1/2)-1/2\log(1/2)-0\log(0)-1\log(1)}{2}=0.5 \end{aligned} \tag{7}$$

Therefore, on the left subtree, it doesn't matter which attribute is chosen. On the right substree, for each of the last 2 attributes, we compute:

- Cerias student:

$$\begin{aligned} H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}) &= \frac{H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}=Y)+H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}=N)}{2} \\ &= \frac{-1/2\log(1/2)-1/2\log(1/2)-0\log(0)-1\log(1)}{2}=0.5 \end{aligned} \tag{8}$$

- CS526:

$$\begin{aligned} H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}) &= \frac{H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}=Y)+H(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}=N)}{2} \\ &= \frac{-1\log(1)-0\log(0)-1/3\log(1/3)-2/3\log(2/3)}{2}=0.4591 \end{aligned} \tag{9}$$

Therefore, on the right subtree, we used CERIAS to split.

Therefore we have the following decision tree:

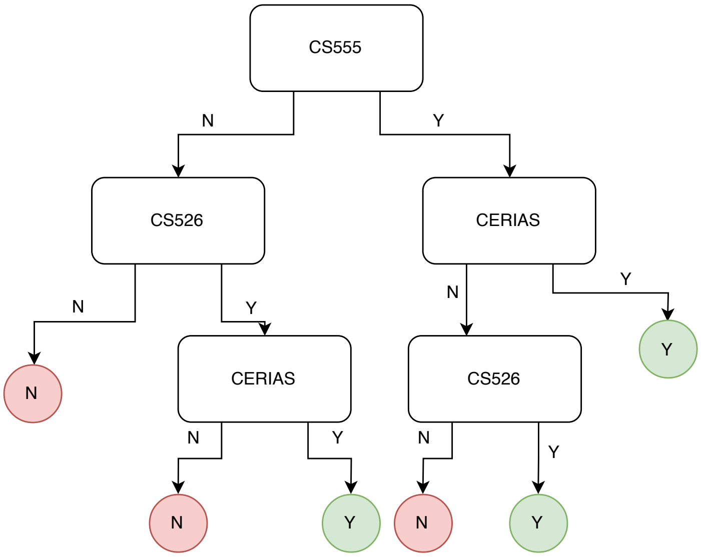


Figure 1: Decision Tree based on information gain

2. Gini Impurity:

   (a) First we need to compute the gini impurity before split:

$$Gini(\texttt{IS}) = 1 - p_y^2 - p_n^2 = 0.48 \tag{10}$$

(b) For each of attribute, we compute:

- Cerias student:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CERIAS}) &= \frac{Gini(\mathsf{IS}|\mathsf{CERIAS}=Y)+Gini(\mathsf{IS}|\mathsf{CERIAS}=N)}{2} \\ &= \frac{2-(4/5)^2-1/5^2-(3/5)^2-(2/5)^2}{2} = .3999 \end{aligned} \tag{11}$$

- CS555:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS555}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS555}=Y)+Gini(\mathsf{IS}|\mathsf{CS555}=N)}{2} \\ &= \frac{2-(5/6)^2-(1/6)^2-(3/4)^2-(1/4)^2}{2} = 0.3263 \end{aligned} \tag{12}$$

- CS526:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS526}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS526}=Y)+Gini(\mathsf{IS}|\mathsf{CS526}=N)}{2} \\ &= \frac{2-(5/6)^2-(1/6)^2-(3/4)^2-(1/4)^2}{2} = 0.3263 \end{aligned} \tag{13}$$

(c) Therefore, we can pick either CS526 or CS555 to split because it maximizes the gini difference. I chose CS555. We compute entropy for left subtree and right subtree:

On the left subtree, for each of the last 2 attributes, we compute:

- Cerias student:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}=Y)+Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CERIAS}=N)}{2} \\ &= \frac{2-1/2^2-1/2^2-1-0}{2} = 0.25 \end{aligned} \tag{14}$$

- CS526:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}=Y)+Gini(\mathsf{IS}|\mathsf{CS555}=N,\mathsf{CS526}=N)}{2} \\ &= \frac{2-1/2^2-1/2^2-1-0}{2} = 0.25 \end{aligned} \tag{15}$$

Therefore, on the left subtree, it doesn't matter which attribute is chosen.

On the right substree, for each of the last 2 attributes, we compute:

- Cerias student:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}=Y)+Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CERIAS}=N)}{2} \\ &= \frac{2-1/2^2-1/2^2-1-0}{2} = 0.25 \end{aligned} \tag{16}$$

- CS526:

$$\begin{aligned} Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}) &= \frac{Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}=Y)+Gini(\mathsf{IS}|\mathsf{CS555}=Y,\mathsf{CS526}=N)}{2} \\ &= \frac{1-1/3^2-(2/3)^2}{2} = 0.2222 \end{aligned} \tag{17}$$

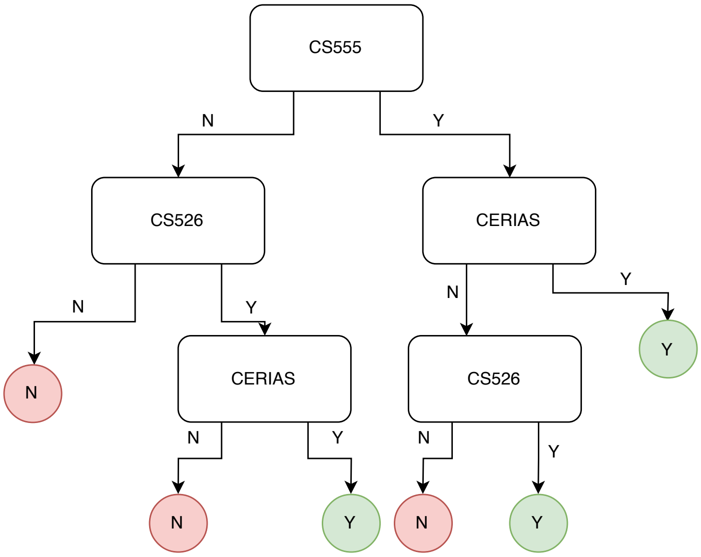


Figure 2: Decision Tree based on Gini impurity

Therefore, on the right subtree, we used CERIAS to split.

(d) Thus, this will give us the same decision as information gain.

Problem 2. (**SVM**) Let x_d denotes the number of derivation calculated, x_c denotes the number of line of codes written, - denotes cs student, + denotes math student. From the graph, we have:

x_d	x_c	label
1	2	+
3	3	+
3	5	+
5	1	-
7	2	-

Table 1: Data points

- Compute the decision boundary.

Ans: We know that the decision boundary is the line that seperate the data points into part. In otherword, it has the form of:

$$w_1 x_d + w_2 x_c + b = 0 \tag{18}$$

and the label is determined as:

$$\mathsf{sign}(w_1 x_d + w_2 x_c + b) \tag{19}$$

In other word, we need to find (w_1, w_2, b) that give negative number for all math data point and positive number for all CS data points. We can simply solve the following equations:

$$\begin{aligned} w_1 + 2w_2 + b &= 1 \\ 3w_1 + 3w_2 + b &= 1 \\ 5w_1 + 1w_2 + b &= -1 \end{aligned} \tag{20}$$

we get:

$$(w_1, w_2, b) = (-1/3, 2/3, 0) \tag{21}$$

Thus, all training data point satisfy the classifier.

- What is the width margin?

Ans: Using the equation from the class we know the width margin is:

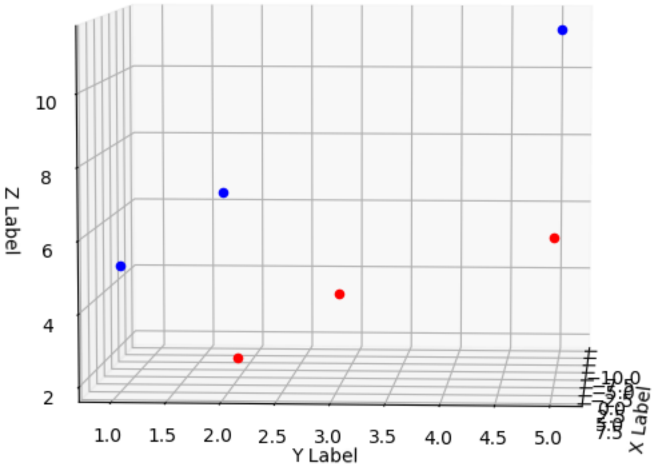
$$\rho = \frac{2}{||w||} = \frac{2}{\sqrt{1/3^2 + (2/3)^2}} = 2.68 \tag{22}$$

- Bob received a funny data point $(-10, 5)$ from a CS student. Help Bob to build a kernel so the new SVM can classify all the data points.

Ans.:

x_d	x_c	$\sqrt{x_d^2 + x_c^2}$	label
1	2	$\sqrt{5}$	+
3	3	$\sqrt{18}$	+
3	5	$\sqrt{34}$	+
5	1	$\sqrt{26}$	–
7	2	$\sqrt{53}$	–
-10	5	$\sqrt{125}$	–

If we plot these data point into 3-d space we get:



As we can see, the data point now can be seperated by a plane. Thus, out mapping is:

$$\phi(x_d, x_c) = (x_d, x_c, \sqrt{x_d^2 + x_c^2}) \tag{23}$$

The kernel is:

$$\begin{aligned} K(\mathbf{x_1}, \mathbf{x_2}) &= \mathbf{x}_1 \cdot \mathbf{x}_2 + ||\mathbf{x_1}|| \times ||\mathbf{x_2}|| \\ &= \phi(\mathbf{x_1}) \cdot \phi(\mathbf{x_2}) \end{aligned} \tag{24}$$

In this equation $\mathbf{x_1} \cdot \mathbf{x_2}$ denotes the dot product of 2 vectors.

Problem 3. (**Perceptron Algorithm**) The Perceptron learning algorithm can be parameterized by a learning rate α. If the current weight vector w classifies an instance $(\mathbf{x}, 1)$ as -1, we do $\mathbf{w}+ = \alpha\mathbf{x}$. If $(\mathbf{x}, -1)$ is classified incorrectly as 1, we do $\mathbf{w}- = \alpha\mathbf{x}$.

3.1) Short answers:

- Assume Perceptron algorithm fails to converge on a training data D. Does adding more training data help Perceptron to converge? Justify your answer.

  Ans: No. I think the reason is because that the algorithm is not able to find the hyperplane that seperates the current training data. Adding more training data will not help the algorithm to find the hyperplane. However, in this case, one should apply the Kernel trick or allow slack variables.

- What problem will result from using a learning rate that's too large, and how can one detect that problem? Justify your answer.

  Ans: learning rate thats too large: the decision boundary will fluctuate a lot. And so one way to detect this problem is to look at the training errors. If the training errors fluctuate without decreasing then the learning rate is too large

- What problem will result from using a learning rate that's too small, and how can one detect that problem? Justify your answer.

  Ans: it will take a lot of time for perceptron to converge (assuming the data is linearly separable). Also, according to the slide, it can be very slow if two input are highly correlated. One way to detect this problem is to look at the training errors. if the training errors decrease very slowly then probably the learning rate is too small

3.2) Consider the following question about training phrase of Perceptron algorithm with training data (2 dimensional vectors) and their label.

- $(0, 1), 0$
- $(1, 1), 1$
- $(1, 0), 1$

Assume the bias is 0, threshold value is 0.5, learning factor is 0.3 and the initial weight vector is $\mathbf{w} = (0, 0)$

- $(0, 1), 0$:

  $$(0, 1) \cdot \mathbf{w} = (0, 1) \cdot (0, 0) = 0 \leq 0.5$$

  ouputs 0 which is correct.

- $(1, 1), 1$:

  $$(1, 1) \cdot \mathbf{w} = (1, 1) \cdot (0, 0) = 0 \leq 0.5$$

  ouputs 0 which is not correct. We update:

  $$\mathbf{w} = \mathbf{w} + \alpha\mathbf{x} = (0, 0) + .3(1, 1) = (.3, .3) \tag{25}$$

- $(1, 0), 1$:

  $$(1, 0) \cdot \mathbf{w} = (1, 0) \cdot (.3, .3) = .3 \leq 0.5$$

  ouputs 0 which is not correct. We update:

  $$\mathbf{w} = \mathbf{w} + \alpha\mathbf{x} = (.3, .3) + .3(1, 0) = (.6, .3) \tag{26}$$

- After the first 3 iteration, we have $\mathbf{w} = (.6, .3)$. We try to predict:

  $$\begin{aligned} \mathbf{x_1} \cdot \mathbf{w} &= (0, 0) \cdot (.6, .3) = 0 \leq .5 \text{ output } 0 \\ \mathbf{x_2} \cdot \mathbf{w} &= (1, 1) \cdot (.6, .3) = 0.9 > .5 \text{ output } 1 \\ \mathbf{x_2} \cdot \mathbf{w} &= (1, 0) \cdot (.6, .3) = 0.6 > .5 \text{ output } 1 \end{aligned} \tag{27}$$

  Therefore, $\mathbf{w}$ did accurately label all 3 samples.