Homework Assignment 4

CSE 251A: Introduction to Machine Learning

Due: May 30th, 2023, 9:30am (Pacific Time)

Instructions: Please answer the questions below, attach your code in the document, and insert figures to create **a single PDF file**. You may search information online but you will need to write code/find solutions to answer the questions yourself.

Grade: ____ out of 120 points

1 (40 points) Naïve Bayes

In this question, we would like to build a Naïve Bayes model for a classification task. Assume there is a classification dataset $S = \{(\mathbf{x}^{(i)}, y^{(i)}), i = 1, ..., 8\}$ where each data point $(\mathbf{x}, y)$ contains a feature vector $\mathbf{x} = (x_1, x_2, x_3); x_1, x_2, x_3 \in \{0, 1\}$ and a ground-truth label $y \in \{0, 1\}$. The dataset S can be read from the table below:

i	x_1	x_2	x_3	y
1	0	0	1	1
2	0	1	1	1
3	1	1	0	1
4	0	0	1	1
5	0	1	0	0
6	1	1	0	0
7	1	0	0	0
8	0	0	1	0

In Naïve Bayes model, we use random variable $X_i \in \{0, 1\}$ to represent i-th dimension of the feature vector $\mathbf{x}$, and random variable $Y \in \{0, 1\}$ to represent the class label y. Thus, we can estimate probabilities $P(Y)$, $P(X_i|Y)$ and $P(X_i, Y)$ by counting data points in dataset S, for example:

$$
\begin{aligned}
P(Y = 1) &= \frac{\#\{\text{data points with } y = 1\}}{\#\{\text{all data points}\}} = \frac{4}{8} = 0.5 \\
P(X_1 = 1|Y = 0) &= \frac{\#\{\text{data points with } x_1 = 1 \text{ and } y = 0\}}{\#\{\text{data points with } y = 0\}} = \frac{2}{4} = 0.5 \\
P(X_1 = 1, Y = 1) &= P(X_1 = 1|Y = 1)P(Y = 1) \\
&= \frac{\#\{\text{data points with } x_1 = 1 \text{ and } y = 1\}}{\#\{\text{all data points}\}} = \frac{1}{8} = 0.125
\end{aligned}
$$

It is noteworthy that **only** probabilities $P(Y)$, $P(X_i|Y)$ and $P(X_i, Y)$ can be **directly** estimated from dataset S in Naïve Bayes model. Other joint probabilities (e.g. $P(X_1, X_2)$ and $P(X_1, X_2, X_3)$) should **not** be estimated by directly counting the data points.

Next, we can use the probabilities $P(Y)$ and $P(X_i|Y)$ to build our Naïve Bayes model for classification: For a feature vector $\mathbf{x} = (x_1, x_2, x_3)$, we can estimate the probability $P(Y = y|X_1 = x_1, X_2 = x_2, X_3 = x_3)$ with the **conditional independence assumptions**:

$$
\begin{aligned}
P(Y = y|X_1 = x_1, X_2 = x_2, X_3 = x_3) &= \frac{P(X_1 = x_1, X_2 = x_2, X_3 = x_3, Y = y)}{P(X_1 = x_1, X_2 = x_2, X_3 = x_3)} \\
&= \frac{P(X_1 = x_1, X_2 = x_2, X_3 = x_3|Y = y)P(Y = y)}{P(X_1 = x_1, X_2 = x_2, X_3 = x_3)} \\
&= \frac{\left(\prod_{i=1}^{3} P(X_i = x_i|Y = y)\right)P(Y = y)}{P(X_1 = x_1, X_2 = x_2, X_3 = x_3)}
\end{aligned}
$$

where the joint probability $P(X_1 = x_1, X_2 = x_2, X_3 = x_3)$ can be calculated as:

$$
\begin{aligned}
P(X_1 = x_1, X_2 = x_2, X_3 = x_3) &= \sum_{y=0}^{1} P(X_1 = x_1, X_2 = x_2, X_3 = x_3, Y = y) \\
&= \sum_{y=0}^{1} \Big(P(X_1 = x_1, X_2 = x_2, X_3 = x_3|Y = y)P(Y = y)\Big) \\
&= \sum_{y=0}^{1} \left(\left(\prod_{i=1}^{3} P(X_i = x_i|Y = y)\right)P(Y = y)\right)
\end{aligned}
$$

Finally, if we find:

$$
P(Y = 1|X_1 = x_1, X_2 = x_2, X_3 = x_3) > P(Y = 0|X_1 = x_1, X_2 = x_2, X_3 = x_3)
$$

then we can predict the class of feature vector $\mathbf{x} = (x_1, x_2, x_3)$ to be 1, otherwise 0. It is noteworthy that although conditional independence assumptions are made in Naïve Bayes model, $P(Y = 1|X_1 = x_1, X_2 = x_2, X_3 = x_3) + P(Y = 0|X_1 = x_1, X_2 = x_2, X_3 = x_3)$ should **still be** 1.

1. (15 pts) Please estimate the following probabilities:

$$
(1)\ P(X_1 = 1, Y = 0), \quad (2)\ P(Y = 0), \quad (3)\ P(X_1 = 1|Y = 1).
$$

   Note that these probabilities can be directly estimated by counting from dataset S.

2. (18 pts) Please calculate the probability $P(Y = 1|X_1 = 1, X_2 = 1, X_3 = 0)$ in Naïve Bayes model using conditional independence assumptions.

3. (7 pts) Please calculate the probability $P(Y = 0|X_1 = 1, X_2 = 1, X_3 = 0)$ in Naïve Bayes model and predict the class of feature vector $\mathbf{x} = (1, 1, 0)$.

2 (40 points) Decision Tree

In this question, we would like to create a decision tree model for a binary classification task. Assume there is a classification dataset $T = \{(\mathbf{x}^{(i)}, y^{(i)}), i = 1, ..., 5\}$ where each data point $(\mathbf{x}, y)$ contains a feature vector $\mathbf{x} = (x_1, x_2) \in \mathbb{R}^2$ and a ground-truth label $y \in \{0, 1\}$. The dataset T can be read from the table below:

i	x_1	x_2	y
1	1.0	2.0	1
2	2.0	2.0	1
3	3.0	2.0	0
4	2.0	3.0	0
5	1.0	3.0	0

To build the decision tree model, we use a simplified CART algorithm, which is a recursive procedure as follows:

- Initialize a root node with dataset T and set it as current node.
- Start a procedure for current node:
  - **Step 1**: Assume the dataset in current node is T_{cur}. Check if all data points in T_{cur} are in the same class:
    - If it is true, set current node as a *leaf node* to predict the common class in T_{cur}, and then terminate *current* procedure.
    - If it is false, continue the procedure.
  - **Step 2**: Traverse all possible splitting rules. Each splitting rule is represented by a vector (j, t), which compares feature x_j and threshold t to split the dataset T_{cur} into two subsets T_1, T_2:

    $$T_1 = \{(\mathbf{x}, y) \in T_{\text{cur}} \text{ where } x_j \le t\},$$
    $$T_2 = \{(\mathbf{x}, y) \in T_{\text{cur}} \text{ where } x_j > t\}.$$

    We will traverse the rules over all feature dimensions $j \in \{0, 1\}$ and thresholds $t \in \{x_j | (\mathbf{x}, y) \in T_{\text{cur}}\}$.
  - **Step 3**: Decide the best splitting rule. The best splitting rule (j^*, t^*) minimizes the weighted sum of Gini indices of T_1, T_2:

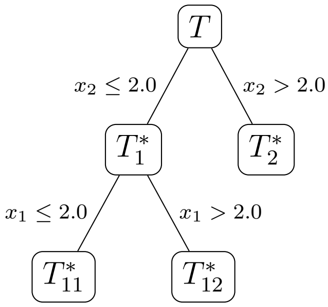
    $$(j^*, t^*) = \arg\min_{j,t} \frac{|T_1|\text{Gini}(T_1) + |T_2|\text{Gini}(T_2)}{|T_1| + |T_2|}$$

    where the Gini$(\cdot)$ is defined as:

    $$\text{Gini}(T_i) = 1 - \sum_{y=0}^{1} P(Y = y)^2,$$

    $$P(Y = y) = \frac{\#\{\text{data points with label } y \text{ in } T_i\}}{\#\{\text{data points in } T_i\}}.$$
  - **Step 4**: We split the dataset T_{cur} into two subsets T_1^*, T_2^* following the best splitting rule (j^*, t^*). Then we set current node as a *branch* node and create child nodes with the subsets T_1^*, T_2^* respectively. For each child node, start from **Step 1** again recursively.

If we run the above decision tree building procedure on dataset T and find the generated tree is shown below:



Please answer the questions:

1. (16 pts) Calculate the subsets $T_1^*, T_2^*, T_{11}^*, T_{12}^*$ using the given decision tree.

2. (12 pts) Calculate $\text{Gini}(T_1^*)$ and $\text{Gini}(T_2^*)$.

3. (12 pts) With the given tree, we can predict the class of a feature vector $\mathbf{x} = (x_1, x_2)$:

   - Start from the root node of the tree:
     - **Step 1**: If current node is a *branch* node, we evaluate conditions on branch edges with $\mathbf{x}$, choose the satisfied branch to go through, and repeat **Step 1**.
     - **Step 2**: If current node is a *leaf* node, the common class of the subset in the leaf node will be used as prediction.

   Please predict the following feature vectors using the given tree:

   (1) $\mathbf{x} = (2, 1)$,

   (2) $\mathbf{x} = (3, 1)$,

   (3) $\mathbf{x} = (3, 3)$.

4. (**Bonus Question, 10 pts extra**) In this question, you need to implement the decision tree algorithm. Please download the Jupyter notebook `HW4_Decision_Tree.ipynb` and fill in the blanks. Note that since the same dataset T is used in the notebook, you can use the code to check if your previous answers are correct or not. Please attach your **code** and **results** in Gradescope submission.

3 (20 points) Bagging and Boosting

Assume we obtain T linear classifiers $\{h_t, t = 1, ..., T\}$ where each classifier $h : \mathbb{R}^2 \to \{+1, -1\}$ predicts the class $\hat{y} \in \{+1, -1\}$ with given feature vector $\mathbf{x} = (x_1, x_2)$ as follows:

$$\hat{y} = h(\mathbf{x}) = \text{sign}(w_1 x_1 + w_2 x_2 + b) \qquad \text{where} \qquad \text{sign}(a) = \begin{cases} +1 & \text{if } a \geq 0, \\ -1 & \text{if } a < 0. \end{cases}$$

where $w_1, w_2, b \in \mathbb{R}$ are the parameters.

- In a bagging model H_{bagging} of the T linear classifiers, we calculate the average prediction using classifiers $\{h_t\}$, and then use it to predict the class $\hat{y}_{\text{bagging}}$:

$$\hat{y}_{\text{bagging}} = H_{\text{bagging}}(\mathbf{x}) = \text{sign}\Big(\frac{1}{T}\sum_{t=1}^{T} h_t(\mathbf{x})\Big)$$

- In a boosting model H_{boosting} of the T linear classifiers, we calculate the weighted sum of predictions using classifiers $\{h_t\}$, and then use it to predict the class $\hat{y}_{\text{boosting}}$:

$$\hat{y}_{\text{boosting}} = H_{\text{boosting}}(\mathbf{x}) = \text{sign}\Big(\sum_{t=1}^{T} \alpha_t h_t(\mathbf{x})\Big)$$

  where $\{\alpha_t, t = 1, ..., T\}$ are the weight coefficients.

In this problem, suppose we have 3 linear classifiers (i.e. $T = 3$):

$$h_1(\mathbf{x}) = \text{sign}(x_1 + x_2 + 1), \quad h_2(\mathbf{x}) = \text{sign}(x_1 - x_2), \quad h_3(\mathbf{x}) = \text{sign}(x_1 - 2x_2 + 1).$$

Please answer the questions below:

1. (10 pts) Please calculate the $\hat{y}_{\text{bagging}}$ of feature vector $\mathbf{x} = (1, 2)$ using bagging on these three classifiers.

2. (10 pts) Please calculate the $\hat{y}_{\text{boosting}}$ of feature vector $\mathbf{x} = (1, 2)$ using boosting on these three classifiers. The weight coefficients are $\alpha_1 = 0.8,\ \alpha_2 = 0.2,\ \alpha_3 = 0.3$.

4 (20 points, Open Question) Overfitting of Bagging

Following the last question, suppose in the general case, we train T linear classifiers, each trained on a randomly sampled subset of the training data (assume dataset size is N, subset size is N_p). Is this model more prone to overfitting than the original model? (Could we overfit our final model by increasing N_p? Could we overfit our final model by increasing T?) Explain your reasoning.