In the name of GOD.



Sharif University of Technology

Machine Learning

Spring 2025

Hamid R. Rabiee, Zahra Dehghanian

Homework 4 | Deadline: 1404/02/07

1. (35) Let $\{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$ be a binary classification dataset where $y^{(i)} \in \{-1, +1\}$, and assume all inputs are distinct and satisfy the condition:

$$\|x^{(i)} - x^{(j)}\| \geq \epsilon > 0 \quad \text{for all } i \neq j.$$

Consider training a Support Vector Machine (SVM) using the Gaussian (RBF) kernel:

$$K(x, z) = \exp\left(-\frac{\|x - z\|^2}{\tau^2}\right).$$

The SVM decision function in its dual form is given by:

$$f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} K(x^{(i)}, x) + b$$

(a) Show that there exists a value of $\tau > 0$ such that the SVM achieves zero training error.
Hint: Use the dual form of the SVM and analyze the behavior of the kernel matrix as $\tau \to 0$.

(b) Suppose $\alpha_i = 1$ for all i, and the bias term is set to $b = 0$. Derive an explicit upper bound on τ in terms of ϵ and n that guarantees zero training error.

Solution:

(a) We use the dual form of the SVM:

$$f(x) = \sum_{i=1}^{n} \alpha_i y^{(i)} K(x^{(i)}, x) + b.$$

To ensure perfect classification, we require:

$$\text{sign}(f(x^{(i)})) = y^{(i)} \quad \text{for all } i.$$

Since the Gaussian kernel satisfies:

$$K(x^{(i)}, x^{(j)}) = \exp\left(-\frac{\|x^{(i)} - x^{(j)}\|^2}{\tau^2}\right),$$

we note the following:

- $K(x^{(i)}, x^{(i)}) = 1$

- For $i \neq j$, $\|x^{(i)} - x^{(j)}\| \geq \epsilon$, so:

$$K(x^{(i)}, x^{(j)}) \leq \exp\left(-\frac{\epsilon^2}{\tau^2}\right) \to 0 \quad \text{as } \tau \to 0$$

Thus, the kernel matrix approaches the identity matrix. This implies that each data point becomes increasingly orthogonal in the feature space induced by the kernel, so the classifier can focus on each point individually with minimal interference from others. In this regime, the optimization essentially assigns weight to each point independently, enabling exact classification.

If we set $\alpha_i = 1$ and $b = 0$, then:

$$f(x^{(i)}) = y^{(i)} + \sum_{j \neq i} y^{(j)} \exp\left(-\frac{\|x^{(j)} - x^{(i)}\|^2}{\tau^2}\right)$$

The second term becomes arbitrarily small as $\tau \to 0$, so:

$$f(x^{(i)})y^{(i)} \approx 1 + (\text{a negligible additive term}) > 0$$

Therefore, the sign of the prediction matches $y^{(i)}$, and zero training error is achieved for sufficiently small τ.

(b) Assuming $\alpha_i = 1$ and $b = 0$, the decision function at a training point is:

$$f(x^{(i)}) = y^{(i)} + \sum_{j \neq i} y^{(j)} \exp\left(-\frac{\|x^{(j)} - x^{(i)}\|^2}{\tau^2}\right)$$

To ensure correct classification:

$$\text{sign}(f(x^{(i)})) = y^{(i)} \iff f(x^{(i)})y^{(i)} > 0$$

Compute:

$$f(x^{(i)})y^{(i)} = 1 + \sum_{j \neq i} y^{(i)}y^{(j)} \exp\left(-\frac{\|x^{(j)} - x^{(i)}\|^2}{\tau^2}\right)$$

In the worst case, all $y^{(j)} = -y^{(i)}$, so each product $y^{(i)}y^{(j)} = -1$. Then:

$$f(x^{(i)})y^{(i)} \geq 1 - \sum_{j \neq i} \exp\left(-\frac{\|x^{(j)} - x^{(i)}\|^2}{\tau^2}\right)$$

Using $\|x^{(i)} - x^{(j)}\| \geq \epsilon$, each term is at most $\exp\left(-\frac{\epsilon^2}{\tau^2}\right)$. There are $n - 1$ such terms, so:

$$f(x^{(i)})y^{(i)} \geq 1 - (n-1)\exp\left(-\frac{\epsilon^2}{\tau^2}\right)$$

To ensure the right-hand side is positive:

$$(n-1)\exp\left(-\frac{\epsilon^2}{\tau^2}\right) < 1 \quad \Rightarrow \quad \exp\left(-\frac{\epsilon^2}{\tau^2}\right) < \frac{1}{n-1}$$

Taking logarithms:

$$-\frac{\epsilon^2}{\tau^2} < \ln\left(\frac{1}{n-1}\right) = -\ln(n-1) \quad \Rightarrow \quad \frac{\epsilon^2}{\tau^2} > \ln(n-1)$$

Solving for τ:

$$\tau < \frac{\epsilon}{\sqrt{\ln(n-1)}}$$

Final result:

$$\tau < \frac{\epsilon}{\sqrt{\ln(n-1)}}$$

This bound on τ guarantees that the classifier with $\alpha_i = 1$, $b = 0$, and a Gaussian kernel will correctly classify all training points.

2. (15) a) Compare the Gini and Entropy criteria in terms of their value ranges. Based on their mathematical definitions, what do you think are the advantages and disadvantages of using each compared to the other?

   b) Train a decision tree using the Gini criterion on the following data points with input variables x_1, x_2, and target variable y. Then, predict the value of y for the new data point ($x_1 = 1.5$, $x_2 = 1.5$). Does the result change if the Entropy criterion is used instead?

x_1	x_2	y
1.5	2.0	1
3.0	2.8	0
1.2	0.7	1
4.2	2.1	1
1.5	1.8	0
0.9	1.2	0
3.1	2.1	1
2.7	2.3	1
1.0	0.8	0
2.8	1.9	1

Table 1: Training data

Solution:

1. Since for any p_i, we have $0 < p_i \leq 1$, and $\log(p_i) \leq 0$, therefore the result of $p_i \log(p_i)$ will be less than or equal to 0, and thus the entropy will always be a non-negative value.

$$H(S) = -\sum_{i=1}^{C} p(x_i) \log p(x_i) = \sum_{i=1}^{C} p(x_i)(-\log(x_i)) = \sum_{i=1}^{C} p(x_i) \left(\log\left(\frac{1}{p(x_i)}\right)\right)$$

Since $\frac{d^2 \log(x)}{dx^2} = -\frac{1}{x^2}$ and is a concave function, we have:

$$\sum_{i=1}^{C} p(x_i) \log\left(\frac{1}{p(x_i)}\right) \leq \log\left(\sum_{i=1}^{C} p(x_i) \cdot \frac{1}{p(x_i)}\right) = \log(C)$$

Hence,

$$0 \leq \text{Entropy} \leq \log(C)$$

We know the maximum value of the function is when all class probabilities are equal:

$$-\sum_{i=1}^{C} \frac{1}{C} \cdot \log\left(\frac{1}{C}\right) = -\log\left(\frac{1}{C}\right) = \log(C)$$

$$\text{Gini} = 1 - \sum_{i=1}^{C} p(x_i)^2$$

Given:

$$\sum p(x_i) = 1 \quad \text{and} \quad 0 \leq p(x_i) \leq 1 \Rightarrow p(x_i)^2 \leq p(x_i) \Rightarrow \sum p(x_i)^2 \leq \sum p(x_i) = 1$$

So,

$$\min \text{Gini} = 1 - \max \sum_{i=1}^{C} p(x_i)^2 = 0$$

This happens when one of the p_i's is 1, and the rest are 0.

$$\max \text{Gini} = 1 - \min \sum_{i=1}^{C} p(x_i)^2$$

Using the Cauchy-Schwarz inequality:

$$\left(\sum_{i=1}^{C} p(x_i)^2\right) \cdot \left(\sum_{i=1}^{C} 1\right) \geq \left(\sum_{i=1}^{C} p(x_i)\right)^2 \Rightarrow \sum_{i=1}^{C} p(x_i)^2 \geq \frac{1}{C}$$

So,

$$\max \text{Gini} = 1 - \frac{1}{C} = \frac{C-1}{C}$$

Hence,

$$0 \leq \text{Gini} \leq \frac{C-1}{C}$$

Given the definitions of Entropy and Gini, and the range of their values for different class probabilities, Entropy emphasizes the distinction among the classes more sensitively than Gini. However, due to the presence of the logarithm, Entropy has higher computational cost compared to Gini. . If the classes are balanced, the result is likely similar, but when the classes are highly imbalanced, entropy is usually more informative.

2. The detailed solution is shown in the figures below.

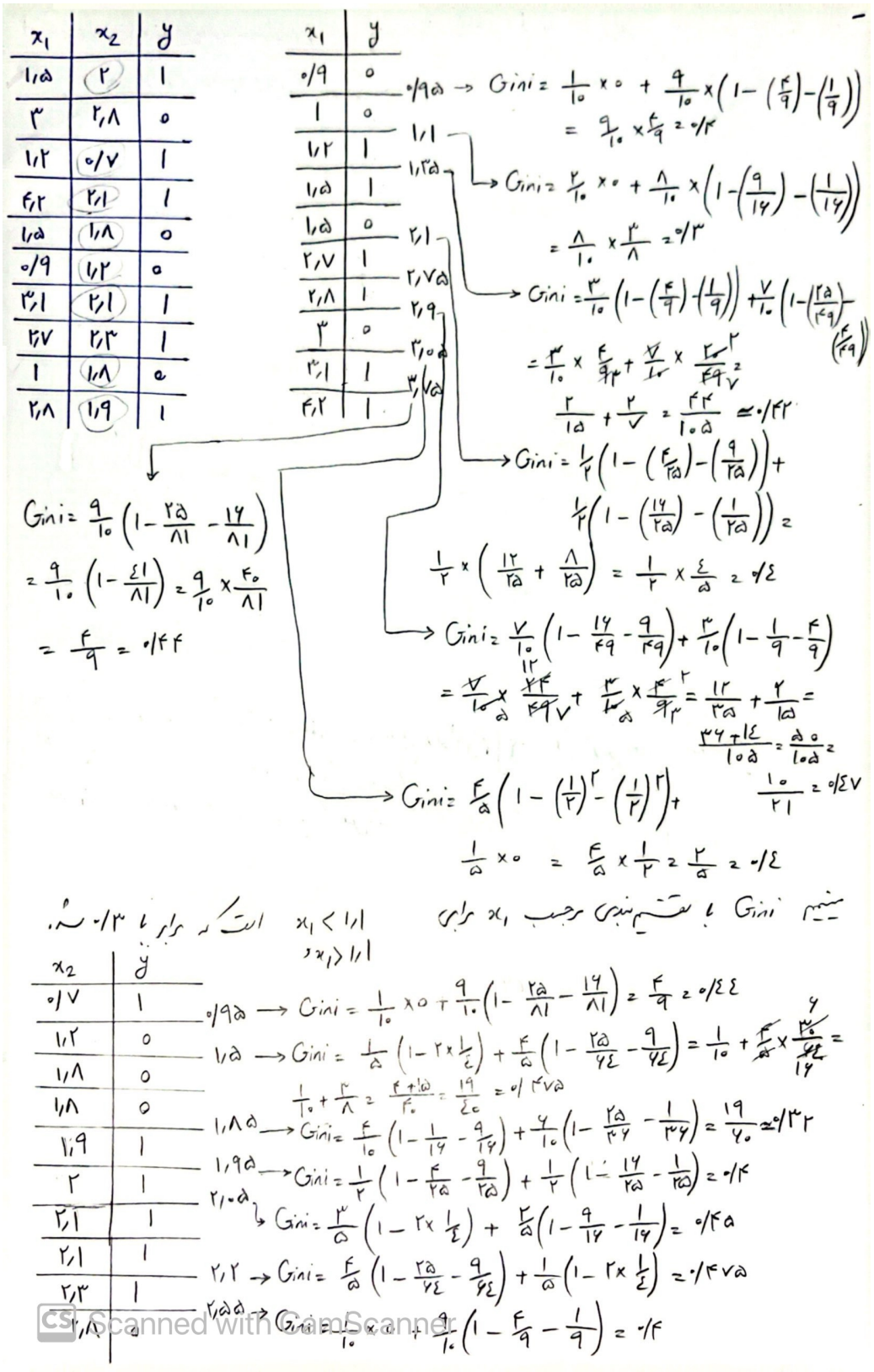


Figure 1: Step-by-step Gini index computation

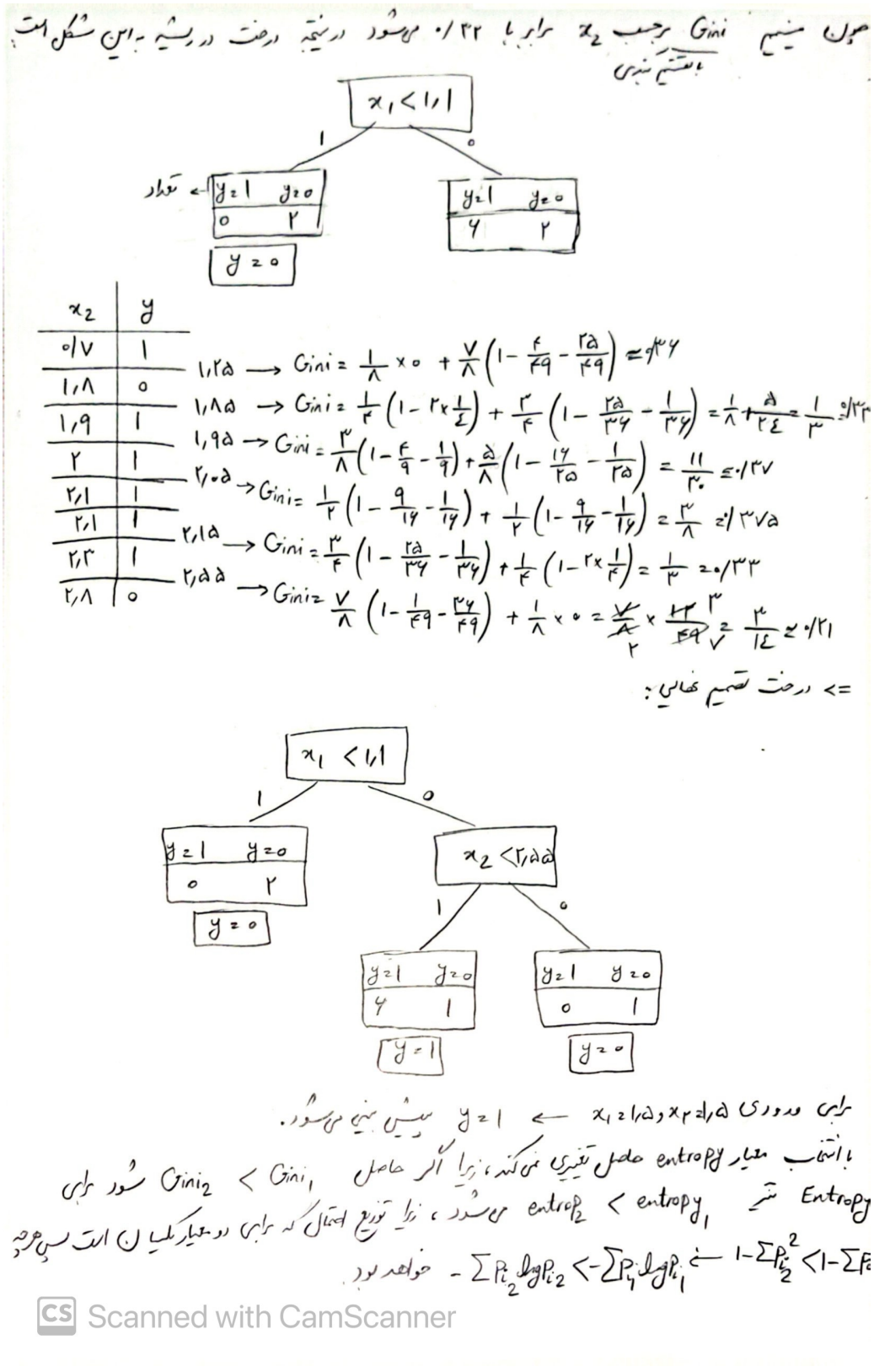


Figure 2: Decision Tree

3. (25) In a small town where communication relies solely on sending encoded strings made up of lowercase English letters (a–z), two friends, Sam and Tina, want to check how similar their encoded messages are. They define a special function, called the **"Secret Code Kernel"**, denoted by $K(s,t)$, which measures the *number of common subsequences* between Sam's message s and Tina's message t.

   Formally,

   $$K(s,t) = \text{The number of equal subsequences of } s \text{ and } t.$$

   For instance, if Sam sends "abc" and Tina replies with "cbc", they calculate $K(s,t) = 4$, since they both share:

   (a) The letter 'c' (once in each string),

   (b) Another 'c' from a different position match,

   (c) The letter 'b', and

   (d) The sequence 'bc'.

   Your task: Prove that this kernel function is valid (i.e., it satisfies the properties of a kernel), and construct the corresponding feature mapping $\Phi(x)$ that represents each string in a suitable feature space.

   Solution:
   To count the number of equal subsequences of s and t, we first define the function $\text{count}_s(x)$ as follows:

   $$\text{count}_s(x) = \text{the number of times the sequence } x \text{ has appeared in } s$$

   By that definition, if we only consider the subsequence x, the number of equal subsequences of s and t where the subsequence equals x is $\text{count}_s(x) \times \text{count}_t(x)$.

   So if we choose k as the max possible length for the strings, and consider Σ as the alphabet, then we can sum the above expression to get $K(s,t)$. So we conclude:

   $$K(s,t) = \sum_{x \in \Sigma^k} \text{count}_s(x) \times \text{count}_t(x)$$

   Thus, if we define $\Phi(s)$ as $[\text{count}_s(x)]_{x \in \Sigma^k}$, then it is obvious that $K(s,t) = \Phi^T(s)\Phi(t)$. So we conclude that $K(s,t)$ is a valid kernel function.

4. (25) Complete the attached notebook. You are permitted to use chatbots or other resources for assistance, but you must ensure that you fully understand the code and implementations. During the online sessions, you may be asked to explain specific functions, code lines, and your overall approach. Be prepared to demonstrate your understanding in detail.