CSCI 5521: Introduction to Machine Learning (Spring 2020)[1]

Homework 1

Questions

1. (**30 points**) Consider the class of K-interval classifier I_K in $\mathbb{R}$ which is specified by K intervals $[a_1, b_1]$, $[a_2, b_2]$,...,$[a_K, b_K]$ and labels any example positive iff it lies inside any of the K intervals.

   (a) What is the VC dimension of I_1 denoted by VC(I_1)? Prove your answer. You need to show why the classifiers can shatter VC(I_1) data points but not VC(I_1)+1 data points.

   (b) What is the VC dimension of I_2 denoted by VC(I_2)? Prove your answer. You need to show why the classifiers can shatter VC(I_2) data points but not VC(I_2)+1 data points.

   (c) What is the VC dimension of I_K denoted by VC(I_K)? Prove your answer. You need to show why the classifiers can shatter VC(I_K) data points but not VC(I_K)+1 data points.

2. (**30 points**) Find the Maximum Likelihood Estimation (MLE) for the following pdf. In each case, consider a random sample of size n. Show your calculation:

   (a) $f(x|\theta) = \frac{1}{\theta}e^{-\frac{x}{\theta}}$, $x > 0, \theta > 0$

   (b) $f(x|\theta) = 2\theta x^{2\theta-1}$, $0 < x \leq 1$, $0 < \theta < \infty$

   (c) $f(x|\theta) = \frac{1}{2\theta}$, $0 \leq x \leq 2\theta$ (Hint: You can draw the likelihood function and pick a θ based on all the data points.)

3. (**30 points**) Let $P(x|C)$ denote a Bernoulli density function for a class $C \in \{C_1, C_2\}$ and $P(C)$ denote the prior,

   (a) Given the priors $P(C_1)$ and $P(C_2)$, and the Bernoulli densities specified by $p_1 \equiv p(x = 0|C_1)$ and $p_2 \equiv p(x = 0|C_2)$, derive the classification rules for classifying a sample x into C_1 and C_2 based on the posteriors $P(C_1|x)$ and $P(C_2|x)$. (Hint: give rules for classifying $x = 0$ and $x = 1$.)

   (b) Consider D-dimensional independent Bernoulli densities specified by $p_{ij} \equiv p(x_j = 0|C_i)$ for $i = 1, 2$ and $j = 1, 2, \ldots, D$. Derive the classification rules for classifying a sample x into C_1 and C_2. It is sufficient to give your rule as a function of x.

[1] Instructor: Rui Kuang (kuang@umn.edu). TAs: Tianci Song (song0309@umn.edu) and Ruyuan Wan (wanxx199@umn.edu).

(c) Follow the definition in 3(b) and assume $D = 2$, $p_{11} = 0.6$, $p_{12} = 0.1$, $p_{21} = 0.6$, and $p_{22} = 0.9$. For three different priors ($P(C_1) = 0.2, 0.6, 0.8$ and $P(C_2) = 1 - P(C_1)$), calculate the posterior probabilities $P(C_1|x)$ and $P(C_2|x)$. (Hint: Calculate the probabilities for all possible samples $(x_1, x_2) \in \{(0,0), (0,1), (1,0), (1,1)\}$).

4. (**30 points**) Using your answers from Question 3 and the provided training, validation, and test datasets, write a Python program to calculate the maximum likelihood estimation on the training set. Consider the prior defined as

$$P(C_1|\sigma) = \frac{1}{1 + e^{-\sigma}} \tag{1}$$

and $P(C_2) = 1 - P(C_1)$. Using the learned Bernoulli distributions and the given prior function, classify the samples in the validation set using your classification rules for $\sigma = -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5$. Finally, choose the best prior (the one that gives the lowest error rate on the validation set) and use it to classify the samples in the test set. Print to the Python console (either in terminal or PyCharm) a table of error rate of each prior on the validation set and the error rate using the best prior on the test set.

Instructions

- **Programming Questions:** All programming questions must be written in Python, no other programming languages will be accepted. Only Numpy can be used in this assignment. The code must be able to be executed from the terminal command prompt on the cselabs machines. Each function must take the inputs in the order specified and display the textual output via the Python console (either in terminal or PyCharm). For each part, you can submit additional files/functions (as needed) which will be used by the main functions specified below. Put comments in your code so that one can follow the key parts and steps. **Please follow the rules strictly. If we cannot run your code, you will receive no credit.**

  - **Question 3:**

    * Training function in Bayes_learning.py: *Bayes_Learning*(training_data , validation_data). The function returns the outputs (p1: learned Bernoulli parameters of the first class, p2: learned Bernoulli parameters of the second class, pc1: best prior of the first class, pc2: best prior of the second class). It must also print to the terminal (sprintf) a table of error rates of all priors.
    * Testing function in Bayes_testing.py: *Bayes_Testing*(test_data, p1: the learned Bernoulli paramter of the first class, p2: the learned Bernoulli paramter of the second class, pc1: the learned prior of the first class, pc2: the learned prior of the second class). The function must print to the Python console (either in terminal or PyCharm) the error rate on the test dataset.

  - **Error rate:** Error rate is the percentage of wrongly classified data points divided by the total number of classified data points.

- **Report:** Solutions to Questions 1, 2 and 3 must be included in a report. The table of error rates on the validation set and the error rate on the test set for Question 4 must also be included in the report.

- **Things to submit:**

  1. hw1_sol.pdf: A document which contains the report with solutions to all questions. Scanned answer sheets need to be clean and 100% legible.
  2. Bayes_Learning.py and Bayes_Testing.py: Code for Question 4.
  3. Any other files, except the data, which are necessary for your code.

- **Submit**: All material must be submitted electronically via Canvas.