CSCI 5521: Introduction to Machine Learning (Spring 2022)[1]

Homework 1

Due date: Feb 16, 2022 11:59pm

1. (**30 points**) Find the Maximum Likelihood Estimation (MLE) of $\boldsymbol{\theta}$ in the following probabilistic density functions. In each case, consider a random sample of size n. Show your calculation:

   (a) $f(x|\theta) = \frac{x}{\theta^2} \exp\{\frac{-x^2}{2\theta^2}\}, x \geq 0$

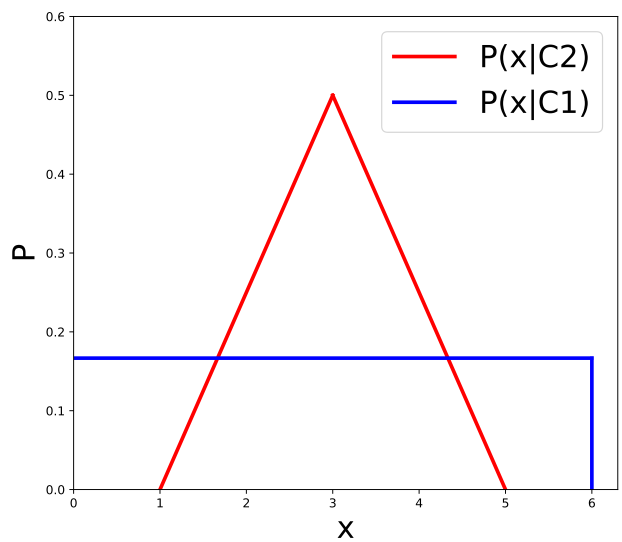
   (b) $f(x|\alpha, \beta, \theta) = \alpha\theta^{-\alpha\beta}x^{\beta} \exp\{-(\frac{x}{\theta})^{\beta}\}, x \geq 0, \alpha > 0, \beta > 0, \theta > 0$

   (c) $f(x|\theta) = \frac{1}{\theta}, 0 \leq x \leq \theta, \theta > 0$ (Hint: You can draw the likelihood function)

2. (**30 points**) We want to build a pattern classifier with continuous attribute using Bayes' Theorem. The object to be classified has one feature, x in the range $0 \leq x < 6$. The conditional probability density functions for each class are listed below:

$$P(x|C_1) = \begin{cases} \frac{1}{6} & if\ 0 \leq x < 6 \\ 0 & otherwise \end{cases}$$

$$P(x|C_2) = \begin{cases} \frac{1}{4}(x-1) & if\ 1 \leq x < 3 \\ \frac{1}{4}(5-x) & if\ 3 \leq x < 5 \\ 0 & otherwise \end{cases}$$



   (a) Assuming equal priors, $P(C_1) = P(C_2) = 0.5$, classify an object with the attribute value $x = 2.5$.

   (b) Assuming unequal priors, $P(C_1) = 0.7, P(C_2) = 0.3$, classify an object with the attribute value $x = 4$.

[1] Instructor: Catherine Qi Zhao. TA: Prithvi Raj Botcha, Shi Chen, Suzie Hoops, James Yang, Yifeng Zhang. Email: csci5521.s2022@gmail.com

(c) Consider a decision function $\phi(x)$ of the form $\phi(x) = (|x-3|) - \alpha$ with one free parameter α in the range $0 \leq \alpha \leq 2$. You classify a given input x as class 2 if and only if $\phi(x) < 0$, or equivalently $3 - \alpha < x < 3 + \alpha$, otherwise you choose x as class 1. Assume equal priors, $P(C_1) = P(C_2) = 0.5$, what is the optimal decision boundary - that is, what is the value of α which minimizes the probability of misclassification? What is the resulting probability of misclassification with this optimal value for α? (Hint: take advantage of the symmetry around x = 3.)

3. (**40 points**) In this programming exercise you will implement three multivariate Gaussian classifiers, with different assumptions as follows:

   - Assume S_1 and S_2 are learned independently (learned from the data from each class).
   - Assume $S_1 = S_2$ (learned from the data from both classes).
   - Assume $S_1 = S_2$ (learned from the data from both classes), and the covariance is a diagonal matrix.

   What is the discriminant function in each case? Show in your report and briefly explain.

   For each assumption, your program should fit two Gaussian distributions to the 2-class training data in `training_data.txt` to learn m_1, m_2, S_1 and S_2 (S_1 and S_2 refer to the same variable for the second assumption). Then, you use this model to classify the test data in `test_data.txt` by comparing log $P(C_i|x)$ for each class C_i, with $P(C_1) = 0.3$ and $P(C_2) = 0.7$. Each of the data files contains a matrix $M \in \mathbb{R}^{N\times 9}$ with N samples, the first 8 columns include the features (*i.e.* $x \in \mathbb{R}^8$) used for classifying the samples while the last column stores the corresponding class labels (*i.e.* $r \in \{1, 2\}$).

   Report the confusion matrix on the test set for each assumption. Briefly explain the results.

   We have provided the skeleton code `MyDiscriminant.py` for implementing the classifiers. It is written in a *scikit-learn* convention, where you have a *fit* function for model training and a *predict* function for generating predictions on given samples. Use Python class `GaussianDiscriminant` for implementing the multivariate Gaussian classifiers under the first two assumptions, and `GaussianDiscriminant_Diagonal` for the third one. To verify your implementation, call the main function `hw1.py`, which automatically generates the confusion matrix for each classifier. Note that you do not need to modify this file.

Submission

- **Things to submit:**

1. hw1_sol.pdf: a document containing all your answers for the written questions (including those in problem 3).

2. `MyDiscriminant.py`: a Python source file containing two python classes for Problem 3, *i.e.,* `GaussianDiscriminant` and `GaussianDiscriminant_Diagonal`. Use the skeleton file `MyDiscriminant.py` found with the data on the class web site, and fill in the missing parts. For each class object, the *fit* function should take the training features and labels as inputs, and update the model parameters. The *predict* function should take the test features as inputs and return the predictions.

- **Submit**: All material must be submitted electronically via Gradescope. **Note that There are two entries for the assignment, *i.e.,* Hw1-Written (for hw1_sol.pdf) and Hw1-Programming (for a zipped file containing the Python code), please submit your files accordingly.** We will grade the assignment with vanilla Python, and code submitted as iPython notebooks will not be graded.