Amirkabir University of Technology
(Tehran Polytechnic)

Assignment 3
From Density Estimation to Sample Classification

Homeworks Guidelines and Policies

- ***What you must hand in.*** It is expected that the students submit an assignment report (HW3_[student_id].pdf) as well as required source codes (.m or .py) into an archive file (HW3_[student_id].zip).
- ***Pay attention to problem types.*** Some problems are required to be solved *by hand* (shown by the icon), and some need to be implemented (shown by the icon).
  Please don't use implementation tools when it is asked to solve the problem by hand, otherwise you'll be penalized and lose some points.
- ***Don't bother typing!*** You are free to solve by-hand problems on a paper and include picture of them in your report. Here, cleanness and readability are of high importance. Images should also have appropriate quality.
- ***Reports are critical.*** Your work will be evaluated mostly by the quality of your report. Don't forget to explain what you have done, and provide enough discussions when it's needed.
- ***Appearance matters!*** In each homework, 5 points (out of a possible 100) belongs to compactness, expressiveness and neatness of your report and codes.
- ***Python is also allowable.*** By default, we assume you implement your codes in MATLAB. If you're using Python, you have to use equivalent functions when it is asked to use specific MATLAB functions.
- ***Be neat and tidy!*** Your codes must be separated for each question, and for each part. For example, you have to create a separate .m file for part b. of question 3. Please name it like p3b.m.
- ***Use bonus points to improve your score.*** Problems with bonus points are marked by the icon. These problems usually include uncovered related topics or those that are only mentioned briefly in the class.
- ***Moodle access is essential.*** Make sure you have access to Moodle because that's where all assignments as well as course announcements are posted on. Homework submissions are also done through Moodle.

- ***Assignment Deadline.*** Please submit your work **before the end of January 17th**.
- ***Delay policy.*** During the semester, students are given 7 free late days which they can use them in their own ways. Afterwards there will be a 25% penalty for every late day, and no more than three late days will be accepted.
- ***Collaboration policy.*** We encourage students to work together, share their findings and utilize all the resources available. However you are not allowed to share codes/answers or use works from the past semesters. Violators will receive a zero for that particular problem.
- ***Any questions?*** If there is any question, please don't hesitate to contact me through **ali.the.special@gmail.com**.

1. Is Oxford/AstraZeneca Vaccine Safe? (12 Pts.)

Keywords: *Density Estimation, Non-Parametric Methods, Histogram, Parzen Windows, Kernel Density Estimation*

Amid sharp increase in Coronavirus cases in the UK and worrying news of spreading a new variant of the virus in the country, the government gave the green light to Oxford/AstraZeneca vaccine. Unlike Pfizer-BioNTech and Moderna, this vaccine requires a more flexible storage condition (in regular fridge temperature) and is distributed with a much lower fee. Similar to other vaccines, it also needs two doses.

In their early stages of testing, the researchers in the Oxford University decided to investigate the impact of injection on different blood measurements. Table below summarizes two of these measurements recorded from 12 volunteers three days after receiving the vaccine. These measurements are as follows:

Figure 1 Boris Johnson meets a COVID-19 vaccine. The U.K. became the first country in the world to authorize the Oxford/AstraZeneca vaccine for use.

- *Systolic Blood Pressure (SBP)* which is reported in mmHg and categorized based on its mean value to "very low-normal" (<120 mmHg), "low-normal" (between 120-130 mmHg), "high-normal" (between 130-140 mmHg), "high" (between 140-150 mmHg) and "very high" (>150 mmHg).
- *Blood pH* which in human body takes values between 7.35 and 7.45.

Volunteer	SBP (mmHg)	Blood pH
1	134.11 ± 5.67	7.38 ± 0.01
2	129.53 ± 3.44	7.43 ± 0.00
3	142.81 ± 4.92	7.42 ± 0.01
4	130.26 ± 4.88	7.40 ± 0.02
5	118.43 ± 2.83	7.38 ± 0.02
6	144.40 ± 2.51	7.36 ± 0.00
7	126.20 ± 4.37	7.41 ± 0.03
8	137.05 ± 1.17	7.40 ± 0.01
9	114.63 ± 7.21	7.39 ± 0.02
10	151.42 ± 3.10	7.37 ± 0.01
11	132.29 ± 8.67	7.41 ± 0.00
12	164.05 ± 6.12	7.40 ± 0.01

Your task is to use **Density Estimation** methods to estimate the probability of SBP category and blood pH value of a receiver after being injected with the vaccine.

a. Draw a histogram for both measurements using appropriate bins centres with the width of one.
b. Sketch the Parzen window estimate of the two unknown density functions for $h_1=1$ and $h_2=3$.
c. Sketch the 3-NN estimate of the density functions.
d. Use a Gaussian kernel to estimate the probability of the following states:
   - SBP = 131.
   - SBP falls into a safe range, or 125 < SBP < 145.
   - pH = 7.42.
   - Blood becomes too acidy, or pH < 7.38.

Note 1: Write down any assumption you made.

Note 2: You are free to use any calculation tool. However, your solution must be precise and complete.

2. Beyond Density Estimation: k-NN is Jack of All Trades! (15 Pts.)

Keywords: *Classification Problem, Regression Problem, Density Estimation, Non-Parametric Methods, K-Nearest Neighbors*

In this problem, we are going to see how **Density Estimation** methods can be used to perform typical machine learning tasks like classification and regression.

First, consider the following two toy datasets consisting of three samples with two input features along with their associated labels:

$$D_1 = \{((0,0),1),((2,2),2),((4,0),3)\}$$
$$D_2 = \{((0,0),1),((1,1),1),((-1,1),2)\}$$

a. Consider the samples in D_1 and draw the decision boundaries for a 1-NN classifier with respect to Euclidean distance. Write down the equations of the decision boundaries.
b. Now repeat the previous part considering a new variant of Euclidean distance d_M. Given two vectors $v_1 = (x_1, y_2)$ and $v_2 = (x_2, y_2)$, $d_M(v_1,v_2)$ is defined as:

$$d_M(v_1,v_2) = \sqrt{\frac{1}{2}(x_1 - x_2)^2 + (y_1 - y_2)^2}$$

c. Repeat part (a) for samples in D_2.
d. Repeat part (b) for samples in D_2.

Now consider the dataset given in Figure 2, part (a).

e. Draw the decision boundary of a k-NN classifier that separates the red points from the blue points. Set $k = 1$ and $k = 3$.
f. Determine the labels of the following points using the above classifiers: (1.3, 4), (1.5, 0.4), (1.4, 3.5).

Finally, consider a regression problem in which the goal is to apply k-NN in order to predict the outcome of a test sample given its feature. Samples are illustrated in Figure 2, part (b).

g. Find the training error when $k = 1$ and $k = 3$.
h. Given the following test samples, predict their corresponding outcomes using 1-NN and 3-NN models: (0.75, ?), (1.75, ?), (2.25, ?)
i. Plot the regression line considering 1-NN and 3-NN models.

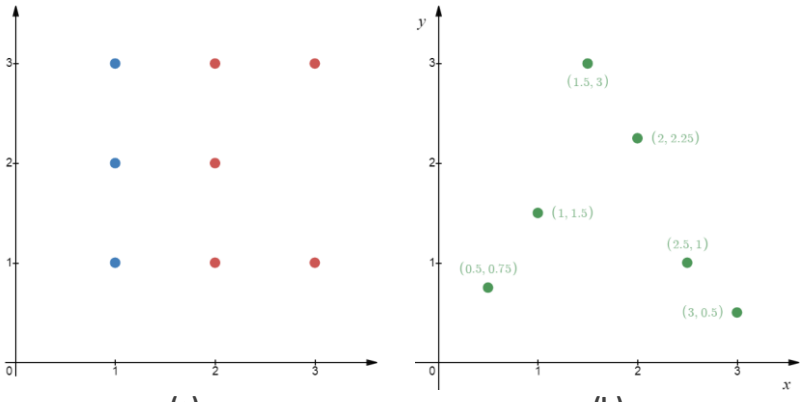


Figure 2 Datasets provided for this problem (a) classification (b) regression

Note: The diagrams in Figure 2 are given as image files, so that you can use them in your answer.

3. Never Get Fooled Again: Fake Bill Detection System (14 Pts.)

Keywords: *Classification Problem, K-Nearest Neighbors, Voronoi Diagram*

United States Department of Treasury has recently estimated that there are roughly $70 million fake bills in circulation, which is approximately 1 note in counterfeits for every 10,000 in genuine currency. It's not often easy for ordinary people to distinguish between fake and real bills merely by their appearance. However, various anti-counterfeiting features have been embedded into a $100 bill which has made the task of detecting forged bills easier for the experts (Figure 3). It has also been shown that images taken from fake and real bills have different values in some of their features.

In this problem, you are going to use **K-Nearest Neighbors** algorithm in order to predict whether a given banknote is authentic or not. You are provided with a dataset containing 1,372 observations with four input variables (features) and one output variable. The variable names are as follows:

- *Feature 1:* Variance of Wavelet transformed image
- *Feature 2:* Skewness of Wavelet transformed image
- *Feature 3:* Kurtosis of Wavelet transformed image
- *Feature 4:* Entropy of image
- *Output*: 0 for authentic, 1 for inauthentic

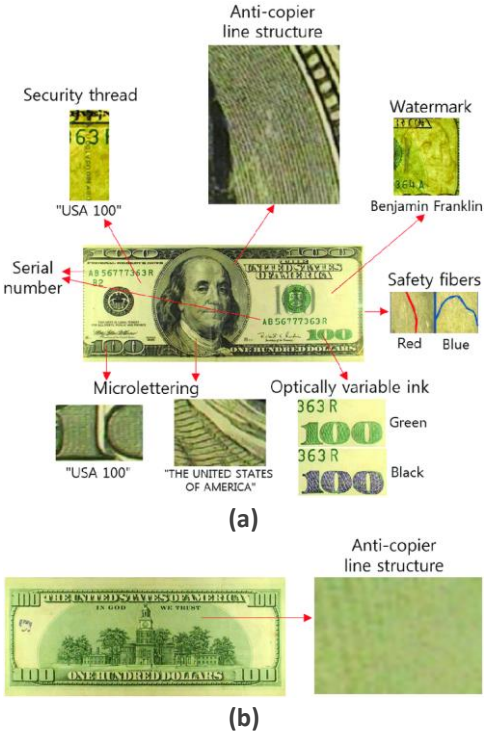


Figure 3 Depiction of anti-counterfeiting features in a new $100 bill (a) Front side (b) Back side

Consider the first 500 samples as your training set, and the remaining as your test set. Let's also assume we want to detect fake banknotes in the test set using only two of the above features.

a. Which of the two features are more suitable for a 1-NN classifier?
b. Plot the Voronoi diagram for these two features assuming a 1-NN classifier.
c. Repeat part (a) for a 3-NN classifier.
d. Repeat part (b) for a 3-NN classifier.

4. Help Trump Fight Against Fake News (14 Pts.)

Keywords: *Classification Problem, K-Nearest Neighbors, Distance Metrics*

Since the start of his presidency in 2016, Donald Trump has always used a particular term to address those news agencies he thought were against him: *fake news*. In some ways he was right, as there were over 100 incorrect articles and rumours detected before and after the 2016 United Stated presidential election. The truth is, fake news has become so prevalent over the past few years that the International Federation of Library Associations and Institutions (IFLA) has published a guide to assist people in recognizing news of this type.

Figure 4 Donald Trump accuses a majority of US news services of publishing fake news.

The goal of this problem is to implement a system based on k-NN Classifier capable of detecting fake news merely by scanning the news headline. You will use a dataset of 1298 "fake news" and 1968 "real news" headlines. The data is attached with this assignment file. It is also recommended that you implement this problem in Python for its relevant libraries. Also you are free to use k-NN built-in implementations.

a. Implement a function `prepare_data()` which loads the data and preprocesses it using a `CountVectorizer` (look here). It must then split the dataset randomly into 70% training, 15% validation and 15% test samples.
b. Implement another function `knn_model_selection()` which utilizes a k-NN classifier to separate real news from the fake ones. Set the values of *k* to be between 1 and 20, and find the training and validation errors. Display a plot of training and validation accuracy for different values of *k*. Also determine the best model based on your calculations.
c. Repeat the previous part with the distance metric set to cosine. Why might the cosine metric perform better than the Euclidean metric here?
Hint: Consider the set {'dog', 'CR7, 'dog dog dog'}.

Recommended Python libraries and functions: `sklearn`, `KNeighborsClassifier()`

5. A Glance At the World of Linear Discriminant Analysis (12 Pts.)

Keywords: *Classification Problem, Supervised Learning, Linear Discriminant Analysis (LDA), Fisher Criterion, Minimum-Squared Error (MSE), Neural Networks, Support Vector Machines, Perceptron Algorithm*

Linear Discriminant Analysis (**LDA**) is a method in machine learning which tries to find a linear combination of features that separate two or more classes of objects. A **Linear Discriminant Function** divides the feature space by a hyperplane decision surface, where the orientation of such a surface is determined by the **Weights** vector (i.e. normal vector w), and the location of the surface is determined by the **Bias** (w_0).

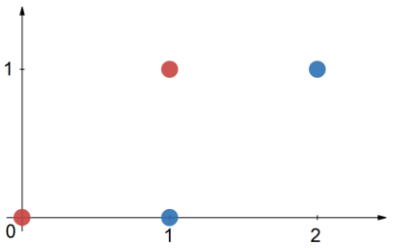
In this problem, the goal is to get you more familiar with **LDA** as well as the **Fisher Criterion**. You will also examine more linear discriminant functions by implementing simple **Neural Networks** and **Support Vector Machines**.

First, assume a two-category classification problem, where there are two sets of data with normal distribution such that:

$$P_1 = 0.3P_2, \qquad \mu_1 = \begin{bmatrix} -3 \\ 0 \end{bmatrix}, \mu_2 = \begin{bmatrix} 3 \\ 0 \end{bmatrix}, \qquad \Sigma_1 = \begin{bmatrix} 1.5 & 1 \\ 1 & 1.5 \end{bmatrix}, \Sigma_2 = \begin{bmatrix} 1.5 & -1 \\ -1 & 1.5 \end{bmatrix}$$

a. Find the linear discriminant function which maximises the Fisher criterion.
b. Minimise the error by adjusting the threshold.
c. Use the linear discriminant function to classify the point $x = [-0.5, 0.25]^T$.
d. Generate 500 samples for each class (1000 in total) and sketch the separating line obtained in part (a).

Next, consider a 2-class classification problem as below:



e. Find the line separating the classes using the Perceptron algorithm in its reward and punishment form, with $\rho = 1$ and $\omega(0) = [0, 0]^T$.

Note: You may use the diagram image of the Figure which is attached to this homework.

Finally, consider the dataset in the following table:

Class	Point
ω_1	(0.25, 0.25)
ω_1	(0.25, 0.75)
ω_1	(0.5, 0.75)
ω_1	(0.75, 0.25)
ω_2	(1, 1)
ω_2	(1.25, 0.75)
ω_2	(1.75, 0.5)
ω_2	(2, 0)

f. Plot the sample points, and try to construct the weight vector for the optimal hyperplane and the optimal margin by inspection.
g. Determine the support vectors.
h. Find the solution in the dual space by finding the Lagrange multipliers α_i. Compare the result with part (f).

6. Categorizing Different Antarctic Penguin Species (20+5 Pts.)

Keywords: *Classification Problem, Neural Networks, Weights Update, Gradient Descent Algorithm, Newton's Algorithm, Perceptron Algorithm, Pocket Algorithm*

There are several species of Penguins in Antarctica. Among them, *Chinstrap*, *Gentoo* and *Adélie* has attracted the attention of Dr. Kristen Gorman, marine biologist from the University of Alaska Fairbanks, and her research team. They collected a dataset of 344 cases along with their physical attributes and their habitats. In this problem, you're going to investigate how **Neural Networks** can handle the problem of classifying these penguins into their correct species using their bill shape.

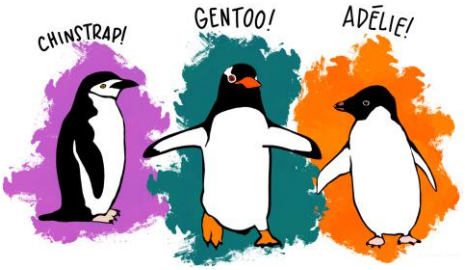


Figure 5 Three species of Antarctic penguins that have been taken into consideration in the research

Please load `penguins.csv`. Among different features of the penguins, keep "bill length" and "bill depth" and ignore the others. These features are sufficient to fairly separate each class of penguins from the others, Figure 5 and 6. Use the first 300 samples as the training set, and the remaining as the test set.

a. Apply basic gradient descent and Newton's algorithm to the samples in *Gentoo* and *Adélie* categories. Set $\eta(k) = 0.1$. Display the criterion function as function of the iteration number. Also display the data distribution as well as the decision boundaries, and report the prediction accuracy on test samples.
b. Display the variation of convergence time versus learning rate. What is the minimum learning rate that fails to lead convergence?
c. Estimate the total number of mathematical operations in the two algorithms.

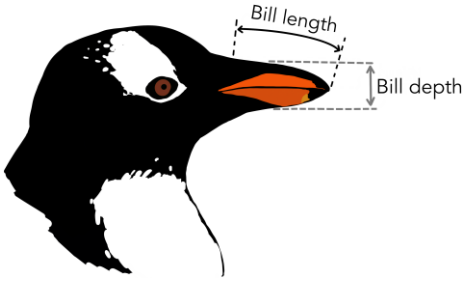


Figure 6 Most Antarctic penguins can be distinguished by the shape of their peaks, namely the length and the depth of the bill.

d. Repeat the previous parts for the samples in *Chinstrap* and *Gentoo* categories. What are your observations?
e. Starting with $a = 0$, apply the Perceptron algorithm to the samples from *Gentoo* and *Adélie*. Report the number of iterations needed for convergence. Also display the decision boundary on the samples distribution and report the prediction accuracy on test samples.
f. Repeat the previous part for samples from *Chinstrap* and *Gentoo*.
g. Comment on the difference between the iterations needed for convergence in the two cases.
h. Write a pocket algorithm to be employed with Perceptron algorithm, and apply it to the samples from *Gentoo* and *Adélie*. How often are the pocket weights updated?
i. Repeat the previous part for samples from *Chinstrap* and *Gentoo*.

7. Some Explanatory Questions (8 Pts.)

Please answer the following questions as clear as possible:

a. In Parzen window density estimation, is there a method to find an appropriate value for bandwidth? Explain.
b. Are k-NN and minimum distance classifiers related? If yes, how? If no, why?
c. Is k-NN decision boundary always linear? Justify your answers with examples.
d. What is the relationship between the values of k and the smoothness of the decision boundary of a k-NN classifier?
e. Consider a 2-class classification problem in which there are samples of both classes in the training set. Is it possible for a 1-NN classifier to always assign a specific label to all test samples? If yes, give an example. If no, explain.
f. Explain a drawback of a k-NN classifier, and suggest a modification to the method in order to avoid it.
g. Under what circumstances do you recommend using SVM instead of MLP? What about the reverse? Which problems do both SVM and MLP suffer to solve?
h. In order to apply backpropagation algorithm on a neural network, which topological condition should it have? Explain.

Good Luck!
Ali Abbasi