Assignment 2
Decisions, decisions, decisions!

Homeworks Guidelines and Policies

- ***What you must hand in.*** It is expected that the students submit an assignment report (HW2_[student_id].pdf) as well as required source codes (.m or .py) into an archive file (HW2_[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 do not use implementation tools when it is asked to solve the problem by hand, otherwise you will be penalized and lose some points.
- ***Don't bother typing!*** You are free to solve by-hand problems on a paper and include their pictures 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. Do not forget to explain your answers clearly, and provide enough discussions when needed.
- ***Appearance matters!*** In each homework, 5 points (out of a possible 100) belong 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 are using Python, you have to use the 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, which must be named '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 is where all assignments as well as course announcements are posted. Homework submissions are also made through Moodle.

- ***Assignment Deadline.*** Please submit your work **before the end of November 26th**.
- ***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 do not hesitate to contact us through the following email addresses: **ali.the.special@gmail.com** and **ebp.mohsen@gmail.com**.

1. Image Segmentation Through Bayes Decision Rule (15 Pts.)

Keywords: *Classification Problem, Bayes Decision Rule, Image Segmentation, CIELAB Color Space*

Bayes Decision Rule is an incredibly simple approach to the problem of pattern classification. It assumes the ideal case in which the probability structure underlying each class is known perfectly. While this assumption rarely occurs in practice, it allows us to determine the optimal classifier against which we can compare all other classifiers.

Now, there is an image with three main regions with different textures and colors; ‘figure’, ‘map’, and ‘hair’. You are given a dataset which contains the color values of 15 randomly-picked pixels of the image in RGB and CIELAB color spaces. The goal is to make use of them to perform **Image Segmentation**, and find a Bayesian classifier capable of assigning these three labels to each arbitrary pixel of the image. Here, you are only allowed to use two features.

Figure 1 The goal is to determine the region of an arbitrary pixel using its color values. There are three classes; ‘figure’ (green), ‘map’ (black), and ‘hair’ (brown).

Pixel	Label	R	G	B	L	a	b
1	Figure	0	97	94	36	-26	-6
2	Figure	205	252	255	96	-15	-7
3	Figure	146	236	211	87	-32	3
4	Figure	39	120	114	46	-26	-4
5	Figure	186	245	235	92	-21	-2
6	Map	0	16	32	4	-1	-12
7	Map	5	12	45	1	6	-19
8	Map	1	0	31	1	5	-17
9	Map	3	0	32	1	6	-17
10	Map	9	1	29	2	10	-13
11	Hair	114	23	25	25	40	25
12	Hair	196	67	3	47	51	58
13	Hair	151	46	19	36	44	41
14	Hair	133	43	24	32	38	33
15	Hair	93	11	24	19	36	17

a. By visual inspection using 2D feature space, evaluate which two features are the most suited.

b. Design a classifier using the Bayes rule by considering the two features you picked in the previous part. The data are assumed to have Gaussian distributions with the same covariance matrix $\Sigma = \mathbf{I}_2$. Find the general form of the discriminant function.

c. Classify the following pixels using the functions you obtained in the previous part.

Pixel	R	G	B	L	a	b
1	95	7	27	19	38	15
2	0	0	28	1	4	-14
3	0	90	89	34	-24	-6
4	35	3	33	5	19	-12
5	113	207	183	77	-34	3

d. Express some of the challenges this system encounters.

Note: You are allowed to use calculator or any other calculation tools such as MATLAB or Python. However, you are not allowed to solve this problem by mere ‘programming’.

2. MLE Applications in Different Fields (18 Pts.)

Keywords: *Parameter Estimation, Maximum Likelihood Estimation, Probability Mass Function*

Maximum Likelihood Estimation (**MLE**) is a **Parameter Estimation** method which tries to find the parameter values of a statistical model that maximise the **Likelihood Function**, given the observations. The resultant is called **Maximum Likelihood Estimate**, abbreviated as **MLE**. Here, we are going to practice MLE in several different sub-problems, each related to a certain field.

First, lets investigate the application of MLE in social networks. Let n be the number of all members in a social network, labeled as $U_1,U_2,\ldots,U_n$. We define R_{ij} as the relationship between two users i and j, so that

$$R_{ij} = \begin{cases} 1 & \text{if } i \text{ and } j \text{ are friend} \\ 0 & \text{otherwise} \end{cases}$$

Furthermore, suppose that the 'friendliness' of the users are the same, i.e., $R_{ij} \sim Ber(p)$ in all cases where $i \neq j$. We aim to make use of maximum likelihood to estimate p. To do so, we make a survey and ask each person how many friends they have. Assuming $n = 7$, and our responses are the following:

U_1: 3 friends, U_2: 4 friends, U_3: 3 friends, U_4: 5 friends, U_4: 2 friends, U_6: 3 friends, U_7: 2 friends

a. Calculate the ML estimate of p.

Next, we will explore the application of MLE in computational imaging. The electronic sensors in digital cameras often generate an intensity value at a certain pixel position which is proportional to the amount of light they receive at the particular location. In ideal condition, if the illumination and the object are fixed, one can take several pictures and get exactly the same digital images. In real world, however, the arrival of photons should be modelled as a Poisson process, hence the intensity values spread across a range of values following a Poisson distribution despite the fact that their average across multiple photos might be λ.

b. Prove that this Poisson distribution would have a parameter λ, i.e., $X \sim Poisson(\lambda)$.
c. In contrast to conventional electronic sensors, the single-photon image sensor is designed for low-light imaging and only gives a binary output. Here, the output is 1 if it detects the presence of at least one photon, and 0 if it does not. In another word,

$$Y = \begin{cases} 1 & X \geq 1 \\ 0 & X = 0 \end{cases}$$

Calculate the PMF of Y.
d. Assume $Z_1,Z_2,\ldots,Z_n$ is a sequence of independent *i.i.d.* Bernoulli random variables where $P(Z_i = 1) = \theta$. Show that the PMF can be written as $p_{z_i}(z_i;\theta) = \theta^{z_i}(1-\theta)^{1-z_i}$, and fine the MLE of θ.
e. We now want to estimate the 'true' intensity value from the binary signals in the single-photo image sensor. Combining the above, show that we can estimate λ through the following equation if $y_1,y_2,\ldots,y_n$ are the observations of the binary values corresponding to a certain pixel location:

$$\hat{\lambda}_n = -\log\left(1 - \frac{1}{n}\sum_{i=1}^{n} y_i\right)$$

Finally, assume that a specific gene appears in one of two 'alleles', which are labeled as B and b. The probability that it comes as b is p. Moreover, a diploid genotype is comprised of two genes, hence it can be BB with probability p^2, Bb with probability $2p(1\text{-}p)$, and bb with probability $(1\text{-}p)^2$.

f. If in a population, x_1 people have genotype BB, x_2 people have genotype Bb, and x_3 people have genotype bb, Find the MLE of p.

3. One Step Further: Maximum A Posteriori Estimation (16 Pts.)

Keywords: *Parameter Estimation, Maximum Likelihood Estimation, Maximum A Posteriori (MAP) Estimation, Posterior Distribution, Prior Distribution*

Another well-known method for **Parameter Estimation** is **Maximum A Posteriori Estimation**, where the estimate of the unknown quantity equals the **Mode** of the **Posterior Distribution**. **MAP** estimation is nearly identical to **MLE**, where the only difference is the inclusion of **Prior Distribution** in MAP. In other words, the likelihood is now weighted with some weight coming from the prior. Therefore, MAP estimation can be interpreted as a **Regularization** of ML estimation.

In this problem, you are to devise ML and MAP estimators for a simple model of an uncalibrated sensor. X Is a random variable which ranges over the real numbers and determines the sensor output. Assume that when tested over a range of environments, the sensor outputs are uniformly distributed on some unknown interval $[0, \theta]$, such that

$$p(x\,|\,\theta) = \begin{cases} \dfrac{1}{\theta} & 0 \le x \le \theta \\ 0 & otherwise \end{cases} = \frac{1}{\theta}\mathbf{I}_{0,\theta}(x)$$

Where $\mathbf{I}_{0,\theta}(x)$ denotes an indicator function which equals 1 when $0 \le x \le \theta$, and 0 otherwise. This distribution is denoted by $X \sim \mathrm{U}(0, \theta)$. It is desirable to infer θ in order to characterise the sensor's sensitivity.

Consider n *i.i.d.* observations $x_1, x_2, \ldots, x_n$, where $X_i \sim \mathrm{U}(0, \theta)$.

a. Find the likelihood function $p(x|\theta)$.
b. What is the maximum likelihood estimator for θ?
c. Express an informal proof that the obtained estimator is actually the ML estimator.

Now suppose that the following prior distribution has been put on the parameter θ:

$$p(\theta) = \alpha\beta^{\alpha}\theta^{-\alpha-1}\mathbf{I}_{\beta,\infty}(\theta)$$

This distribution – known as a *Pareto* distribution – is denoted by $\theta \sim \mathrm{Pareto}(\alpha, \beta)$.

d. Considering the following hyperparameter choices, plot the three prior probability densities corresponding to them:

$(\alpha, \beta) = (0.1, 0.1)$ $\quad$ $(\alpha, \beta) = (0.1, 2.0)$ $\quad$ $(\alpha, \beta) = (2.0, 1.0)$

e. If n uniformly distributed observations $X_i \sim \mathrm{U}(0, \theta)$ are observed, such that $\theta \sim \mathrm{Pareto}(\alpha, \beta)$, find the posterior distribution $p(\theta\,|\,x)$.
f. Is the obtained result in the previous part a member of any standard family?
g. Find the corresponding MAP estimator of θ for the derived posterior in part e.

h. How does MAP estimator you derived in the previous part compare to the ML estimator?

i. The **Quadratic Loss** could be defined as $L(\theta,\hat{\theta})=(\theta-\hat{\theta})^2$. For the posterior derived in part e., what estimator of θ minimises the posterior expected quadratic loss?

j. Let $x=(0.6,1.1,1.9)$ be our observations. Find the posterior distribution of θ for each of the priors given in part d., and plot the corresponding posterior densities.

k. In the previous part, what would be the MAP estimator for each of the hyperparameter choices?

l. In part j., what estimator minimises the quadratic loss for each of the hyperparameter choices?

4. MDC for OCR: An Attempt to Classify Farsi Digits (12 Pts.)

Keywords: *Classification Problem, Minimum Distance Classifier, Optical Character Recognition*

A **Minimum Distance Classifier** attempts to classify an unlabelled sample to a class which minimise the distance between the sample and the class in multi-feature space. As minimising distance is a measure for maximising similarity, **MDC** actually assigns data to its most similar category.

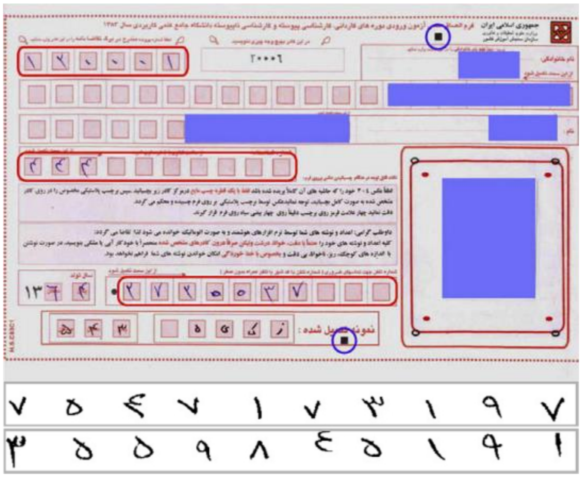

Figure 2 The images in the dataset are scanned from thousands of registration forms of Karshenasi Arshad entrance exam in the year 2005.

While **MDC** might look too basic, it works pretty well in some problems. One of them could be **Optical Character Recognition** (**OCR**), where the goal is to distinguish handwritten or printed text characters inside digital images of documents. Here, we aim to apply this technique to the problem of Farsi digits classification. We use a dataset named Hoda, which contains 102353 samples of digits written by candidates of Karshenasi Arshad entrance exam in their registration forms, Figure 2. You are given a shorter version of this dataset in which the images are binary.

a. Use the training set to calculate the prototype of each class. Display the results.

b. Now use the test samples to evaluate your MDC classifier. Report the error, and display five erroneous predictions.

5. Face Detection Problem: How Does Bayesian Decision Theory Handle It?! (24 Pts.)

Keywords: *Classification Problem, Bayes Decision Rule, Confusion Matrix, Bayes Error, ROC Curve, Face Detection, RG Chromaticity*

So far you have probably got familiar enough with the **Bayesian Decision Theory** to know what to expect from it. If the decision problem is posed in probabilistic terms and all relevant probability values are known, **BDT** allows to take optimal decisions that minimise errors by choosing the least risky class. Although in many practical classification problems, these conditions are not fulfilled and therefore **BDT** won't be effective, there are still some applications where it may come in handy.

Face Detection (not to be mistaken with **Face Recognition**) is the process of identifying and locating human faces in digital images and videos. It is often the first step in many face-related machine vision applications such as face recognition, emotion detection, gender detection, etc. Now we are going to find out how **BDT** deals with this problem in practice.

You are given a customised dataset divided into two 'train and 'test' sections. The train set consists of 50 face images alongside the corresponding face masks. These binary masks indicate face pixels with white (or binary value 1) and non-face pixels with black (or binary value 0).

a. First assume images in the training set. Considering two classes for each pixel, 'face' and 'non-face', use the provided masks to find the class priors.
b. We want to model the class-conditional probability density of each class using a univariate Gaussian. Find the mean and variance of both class-conditional densities.
c. Apply your classifier on the test images and display the results. Also report the test error using the given masks.
d. Compute a confusion matrix for your classifier.
e. Calculate the Bayes error.
f. Draw a ROC curve to visualise the performance of your classifier.
g. Comment on the above results. In what circumstances has your classifier failed, and why? What do you think the advantages and disadvantages of such a Bayesian face detector are? Suggest at least two modifications to improve it.

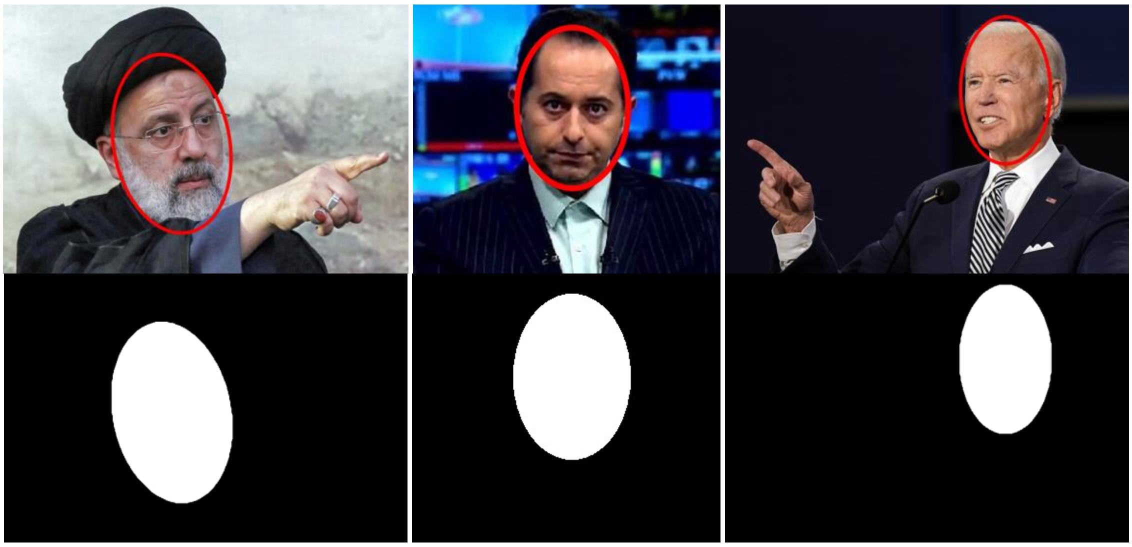

Figure 3 Samples in the given dataset are 'unconstrained', i.e. there is no limitation on the number of faces in an image and their illumination, pose, occlusion and facial expressions. Top: original images with the location of face(s) highlighted. Bottom: corresponding masks, which can be used to specify the pixels inside face regions.

Hint: RGB is not the best color representation for characterising skin-color, because it represents not only color but also brightness. Therefore, represent skin-color in the chromatic space which is defined as follows:

$$r = \frac{R}{(R+G+B)}, \quad g = \frac{G}{(R+G+B)}, \quad b = \frac{B}{(R+G+B)}$$

6. Some Explanatory Questions (10 Pts.)

Please answer the following questions as clear as possible:

a. Is the result of the Bayes decision rule unique? Explain.
b. How would a Gaussian-based classifier act upon an XOR classification problem?
c. How do you extend Bayes decision rule for a three-class classification problem? Explain and – if necessary – include calculations.
d. How does selecting different distance functions affect MDC classification result? Support your answer with simple examples in 2D feature space.
e. How does ML and MAP estimation change when the data is not i.i.d.?

Good Luck!
Ali Abbasi, Mohsen Ebadpour