MACHINE LEARNING FROM DATA

Lab Session 1 – MAP and Gaussian data
Classification criteria based on maximizing posterior probability

1. Goal

The goal of this session is to

- become familiar with Python libraries NumPy, Matplotlib, Seaborn, Scikit-learn
- generate Gaussian datasets and estimate their parameters
- use MAP linear and quadratic classifiers on the generated datasets

2. Instructions

- Download and uncompress the file **Mlearn_Lab1.zip**
- Answer the questions in the document **Mlearn_Lab1_report_team_surnames.docx .**

3. Introduction and previous study

Read ScikitLearn documentation about linear and quadratic discriminant analysis https://scikit-learn.org/stable/modules/lda_qda.html

Q1: Compute the eigenvalues of the matrix $\mathbf{C} = \frac{\sigma^2}{2}\begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}$ as a function of the parameters ρ and σ^2

(edit equations or solve by hand and scan and insert an image with the solution)

4. Generation of Gaussian datasets

4.1. Analysis of ROC curves

Open Colab Notebook **Mlearn_lab1_1.ipynb**

In this section we will use vectors of dimension d=3 (variable n_feat in the script) and c=2 classes (variable n_classes). The three features are statistically independent, as described by their covariance matrices, following Case 1 studied in class (see course slides, topic 2.1).

The following formula describes de model followed by the samples corresponding to each of the two classes:

$$f\left(\mathbf{x}\middle|\omega_c\right) = N(\mathbf{m}_c, \mathbf{C}); \quad c = 1,2 \qquad \mathbf{C} = \tfrac{1}{d}\sigma^2\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}; \quad d = 3$$

where $SNR = 10\log_{10}\left(\frac{average\ energy}{s^2}\right) = 10\log_{10}\left(\frac{\mathbf{m}_c^T\mathbf{m}_c}{s^2}\right)$

Read the notebook, analyse the code, identify the most relevant parts.

Run the code four times, using different SNR values. Consider the following four cases:

SNR= 3, 0, -3, -10 dB.

Observe the error probability, confusion matrices and ROC curves obtained by the following classifiers **on the test set**:

- **Linear (LC)**: Linear decision boundaries. By default, it assumes that class prior probabilities are the class relative frequencies (number of elements in the class divided by the total number of elements)
- **Quadratic (QC)**: Quadratic decision boundaries. By default, it assumes class prior probabilities are class relative frequencies, and estimates different covariance matrices (one matrix per class).

Q2. Create a table including error probabilities obtained by the linear classifier (LC) and error probabilities obtained by the quadratic classifier (QC), for each SNR value on the test set. Discuss the results.

Q3. Include in the report the confusion matrices obtained for SNR=-10db and SNR=-3dB on the test set. Discuss the results.

Q4. Include in the report the ROC curves obtained for SNR=-10db and SNR=-3dB on the test set. Discuss the results.

Q5. Compute the Mahalanobis distance between the two classes on the test set for SNR= 3, 0, -3,-10 dB. Compare the results. Explain why the result differs depending on the order of the parameters.

4.2. Eigenvalues of the covariance matrix and cluster shape

Now use Colab notebook **Mlearn_lab1_2.ipynb**.
In this section we will work with the QPSK modulation. Therefore, feature vectors have dimension d=2 and there are four classes or symbols (c=4).

QPSK and covariances of all classes identical but arbitrary (case 2)

Initially all the classes share the same covariance matrix, which is not diagonal (that is the case 2 explained in class, see slides):

$$f\left(\mathbf{x}|\omega_c\right) = N(\mathbf{m}_c, \mathbf{C}); \quad c = 1,..,4; \qquad \mathbf{C} = \tfrac{1}{d}\sigma^2 \begin{pmatrix} 1 & \rho \\ \rho & 1 \end{pmatrix}; \quad d = 2$$

Read the notebook **Mlearn_lab1_2.ipynb ,** analyse the code, identify the most relevant parts.

Run the code, using SNR=10dB for the following cases

- parameter $\rho = 0$
- parameter $\rho = 0.5$

Q6. Include the scatter plot, decision boundary, confusion matrices and error probabilities obtained using the linear classifier (LC) and the quadratic classifier (QC) for $\rho = 0$. Compare the metrics for the two classifiers and discuss the results.

Q7. Repeat the previous analysis (Q8) for $\rho = 0{,}5$. Compare the metrics for the two classifiers and discuss the results.

Q8. Compare and discuss the results obtained in Q6 and Q7

QPSK and different covariance matrices (case 3)

Now, each class has a different covariance matrix. This is the case 3 explained in class.

Edit the notebook Mlearn_lab1_2.ipynb to generate the QPSK modulation using SNR = +5 dB and SNR = +10 dB, where classes (or symbols) are generated using the following parameters

- Symbol 1: $\rho = +0.5$
- Symbol 2: $\rho = 0$
- Symbol 3: $\rho = -0.5$
- Symbol 4: $\rho = +0.8$

Q9. Include the error probabilities obtained using the linear classifier (LC) and the quadratic classifier (QC) for SNR = +5 dB and +10 dB. Compare the metrics for the two classifiers and discuss the results.

Q10. Complete the table with the theoretical eigenvalues using the formula obtained when answering Q1, and the eigenvalues computed using the **sample** data covariance matrices (the covariance matirx computed from the randomly generated data) . In order to compute eigenvalues using the sample data, add the code to compute the eigenvalues of each covariance matrix; use `scipy.linalg.eigvals` (for just one SNR value)
Notice that when the SNR changes, the eigenvalues change proportionally. Therefore, it is not necessary to compute eigenvalues for different SNR values, you can just compute them for one single SNR value.

Q11. Include scatter plots for the linear and quadratic classifiers using SNR= +5 dB and SNR= +10 dB. Relate the shape of the clusters with the eigenvalues of the covariance matrices.

For SNR = 10 dB, multiply the covariance matrix of class 1 by a large number (for example sigma(0)*=30). Compute the classification error, scatter plots and boundaries for the linear and the quadratic classifiers. Observe that in this case the quadratic discriminant outperforms the linear one.

Q12. Include error probabilities, scatter plots and decision boundaries. Compare the performance of the classifier and justify the results. Include in your answer the new value of sigma[0].