MACHINE LEARNING FROM DATA
Fall 2018

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 Matlab toolbox "Statistics and Machine Learning" for classification
- generate Gaussian datasets and estimate their parameters
- use MAP linear and quadratic classifiers on the generated datasets

2. Instructions

Getting the material:

- Download and uncompress the file ML_Lab0.zip

Handling your work:

- Answer the questions in the document Lab1_report_yourname.pdf
- Save the report and Matlab code (if necessary) in a folder, and upload the compressed folder (zip, rar).

3. Introduction and previous study

Read Matlab documentation to understand the use of the function `fitcdiscr.m` for classification: see: http://es.mathworks.com/help/stats/fitcdiscr.html.
See how to Create a model , how to visualize the boundaries, and how to make predictions, in particular for more than two classes.
http://es.mathworks.com/help/stats/creating-discriminant-analysis-model.html
http://es.mathworks.com/help/stats/create-and-visualize-discriminant-analysis-classifier.html
http://es.mathworks.com/help/stats/prediction-using-discriminant-analysis-models.html

Q1: Find 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

4. Generation of Gaussian datasets

4.1. Analysis of ROC curves

Use Matlab script `lab1_gauss3.m`

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

Q2. Read the script `lab1_gauss3.m,` analyse the code, identify the most relevant parts. Briefly describe each part.

Note: using Matlab help you can get more information regarding the functions used in the script.

Run `lab1_gauss3.m` for the following four cases:

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

Observe the error probability, confusion matrices and ROC curves obtained by the following classifiers:

- **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).

Q3. 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. Discuss the results.

Q4. Include in the report the confusion matrices obtained for SNR=-10db and SNR=-3dB. Discuss the results.

Q5. Include in the report the ROC curves obtained for SNR=-10db and SNR=-3dB. Discuss the results.

Q6. Compute the Mahalanobis distance between the two classes using `mahal.m`. Explain why the result differs depending on how this function is called (depending on the order of the parameters).

4.2. Eigenvalues of the covariance matrix and cluster shape

Now use Matlab script `lab1_QPSK.m`
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):

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

Q7. Read the script `lab1_QPSK.m,` analyse the code, identify the most relevant parts.

Run `lab1_QPSK.m` using SNR=10dB for the following cases

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

Q8. 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.

Q9. Repeat the previous analysis (Q8) for $\rho = 0{,}5$. Compare the metrics for the two classifiers and discuss the results.

Note that not all the boundaries between pairs of classes are displayed. The plot only shows boundaries between class 1 and all the other classes

Q10. Compare and discuss the results obtained in Q8 and Q9

<u>QPSK and different covariance matrices (case 3)</u>

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

Edit the script `lab1_QPSK.m` 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: $\rho = +0.8$

Q11. 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.

Q12. Complete the table with the theoretical eigenvalues obtained in Q1, and the eigenvalues computed using the class covariance matrices with the function

```
eigenvalues=eig(squeeze(M_covar(:,:,c)))
```

Where M_covar is the variable that contains the covariance matrices.
Notice that when the SNR changes, the eigenvalues change proportionally. Therefore, it is not necessary to compute theoretical and empirical eigenvalues for different SNR values, you can just compute them for one single SNR value.

Q13. 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 <u>of class 1 </u>by a large number (for example sigma(1)=30). Compute the classification error, scatter plots and boundaries between class 1 and the other classes for the linear and the quadratic classifiers. Observe that in this case the quadratic discriminant outperforms the linear one.

Q14. Include error probabilities, scatter plots and decision boundaries. Compare the performance of the classifier and justify the results.