# TP 4 Data Mining: LDA Linear Discriminant Analysis

Tuesday 23$^{\text{rd}}$ April, 2024
**deadline**: Monday 13$^{\text{th}}$ May, 2024, 23:59
**Obligatory**

## 1 Introduction

The goal of this TP is to understand, implement and apply the Fisher Linear Discriminant Analysis method for classification and dimensionality reduction. You will be asked to implement the algorithm using two different method: a standard one based on Rayleigh quotients and an alternative one based on gradient descent. In this TP you are going to fill a few missing functions in the python scripts to implement the exercises that we ask. First of all, read and understand the given python scripts. To test your implementation, you have to run the main main lda.ipynb notebook. Here you will also need to write some code (it is mentioned where) to run the LDA algorithm on different datasets and tasks. For the rest of the exercises, the code is given and it works if the missing functions are implemented.

## 2 Exercises

The LDA class in lda.py contains the base class with some general methods for applying LDA. Read the comments carefully and implement the requested methods. Notice that the _calculate_discriminants() method will not be yet implemented at this point. This is because we will then build two separate implementations of this method, in different child classes. Once everything in lda.py is complete, we can look at the two distinct implementations from lda_rayleigh.py and lda_gd.py. In both these files, you will only be asked to implement the method to calculate the discriminant for the specific implementation.

In the LDARayleigh class, you will implement the function using the standard method that employs Rayleigh quotients. Here you will have to compute the scatter matrices (you should have already implemented the method for that in lda.py) and then use them to compute the discriminants.

In the LDAGD class (inside lda_gd.py), you will implement the discriminant using a less standard method based on gradient descent. The pseudocode is provided in $https://doi.org/10.1007/s11063-007-9056-7$.

Again, the only additional method that is required here is _calculate_discriminants() one. Everything else should have already be implemented in the LDA base class.

**Note:**

- Explain in your own words the GD algorithm provided in this pseudocode.
- Once everything is implemented correctly, you should be able run the main_lda.ipynb notebook from start to finish without any error or inconsistency. You cannot modify the given functions.
- Write a function named compute_accuracy(y_true, y_pred) in the utils.py script. The function takes as arguments the true and the predicted class labels and returns the accuracy. Use only numpy.

Once your implementation is ready you will work with the Iris dataset and the Breast cancer dataset

## 3 Questions

Comment all your results, plots, remarks etc and also answer the following questions:

- Why, in some cases, would LDA be preferable to PCA as a dimensionality reduction technique?
- Perform a grid search on the number of iterations for the gradient descent algorithm, and report the test accuracy for the different values (at least 4). How the number of the iterations influences the accuracy?
- Compare the test accuracy of the 2 approaches.

## 4 General Instructions

Submit a formal report in .pdf format answering the previous questions and necessary guidance for understanding your code.

The code should be individual, well-written and with detailed comments to explain each step. Avoid the for loops and if statements using the nymPy library. Your code should be generic and you should use the given functions. For example, the code for the *iris data* should be applicable, to the *Digits data* without any modification.

The work should be submitted before the end of the deadline. The work will still be able to be submitted with a 10% minus per late day.

## Reference

- Sharma, A., & Paliwal, K. K. (2008). A Gradient Linear Discriminant Analysis for Small Sample Sized Problem. *Neural Processing Letters*, *27*(1), 17–24. DOI: 10.1007/s11063-007-9056-7