

# Bayesian Classifier

## Assignment 1

Prof. Panos Trahanias trahania@csd.uoc.gr  
T.A. Emmanouil Sylligardos sylligardos@csd.uoc.gr  
T.A. Despina - Ekaterini Argiropoulos despargy@csd.uoc.gr

**Deadline Friday 23:59, 3rd of March 2023**

### Abstract

In this assignment the task is to familiarize with Bayesian Decision Theory. Assignment's 1 task is the implementation of a Bayesian classifier and its evaluation on the given Wine dataset. Please **use** the *'parameters.txt'* file for any variables/information that you might need.

**Note:** This is a personal assignment and should be pursued individually and without use of any computerised AI facilities. Note that this will be automatically checked for every delivered assignment. The assignment should be implemented entirely in Google Colaboratory following the delivered instructions below.

### Question A: The model (40/100)

1. Implement a function that computes the prior probabilities for each class.
2. Compute the posterior probabilities for each sample.
  - (a) Implement a function that takes as input one sample (one point in the 2-dimensional space), a mean and a covariance matrix and returns the likelihood of that point belonging to the class whose distribution is described by the given mean and covariance matrix.
  - (b) Implement a function that computes the posterior probabilities for each sample.
3. Save and print the predicted class for each sample, according to the posterior probabilities. What's the accuracy of your classifier?

### Question B: Conditional Risk (30/100)

1. Compute the conditional risk for every sample.
2. Save and print the predicted class for each sample, in order to minimize the conditional risk. What's the accuracy now? What do you notice and why do you think this is the case?

### Question C: Plot Results (30/100)

1. Create a scatter plot that also shows the distribution of each class and the decision boundaries of your classifier.

### \* 10% Bonus for Pythonic implementations (see tutorial 1)

### Deliverable

This assignment should be implemented entirely in Google Colaboratory. Google's notebook allows you to combine executable Python scripts with rich text in a single document. Your deliverable should be a single .ipynb file along with its corresponding .py file (both can be easily exported from Google Colaboratory). Every single question should be implemented in a single code block. Code blocks should be clearly and shortly explained (you may use the text boxes for that goal). Use **only** library functions for matrix operations and plots.