

# General Instructions

This is an individual examination, you are not allowed to receive help from your peers. You are free to use internet, and the blackboard including usage of existing code.

Note: you may need to perform further steps to do the required work, e.g. data loading, pre-processing, shuffling, post-processing etc.

**Deliverable:**

1. Code: and a one-page report.
2. Short Report: submitted along with code (If appropriate, use tables and/or graphs to summmarise your data)

example

|  |  |
| --- | --- |
| **Parameter value** | **Result** |
|  |  |
|  |  |
|  |  |

**Submission of the Deliverable:**

1. Zip the code, and the short report in a zip file named: LastnameFirstname.zip
2. Zipped File to be submitted to the blackboard

**Q1-Clustering-50%**

Access the following data set: yeast.data. The class labels are at the last column, the first column has the IDs of the data, the rest of the columns are the attributes. (The data set is about biology)

Code

1. Load the data (Hint: the data are separated by one or more spaces, you may need to use the: sep=”\s+” in Pandas).
2. Use the above data set without the class labels.
3. Apply two different clustering algorithms (the choice is yours); discover the optimal clustering parameter(s)
4. Present the clustering results: number of data items per cluster, quality of clustering (e.g.. use plots and/or prints)
5. Now try to preprocess the data with PCA, what do you get as a result with respect to clustering?

Report

1. One paragraph: which evaluation criterion did you use, how did you discover the best value of the parameters, as well as how the results of clustering differ after PCA.

**Q2-Classification-50%**

Consider the dataset: Australian-im.data. The last column denotes the class of each instance: there are two classes the 0 and 1. The rest of the columns are data attributes. (The dataset is about approval of credit cards).

Code

1. Load the data (Hint: the data are separated by one or more spaces, you may need to use the: sep=”\s+” in Pandas.)
2. Rank the input features in terms of relation (importance) to the attribute (or class) we wish to predict.
3. Train a NaiveBayes, a Nearest Neighbor and a Decision Tree and a Support Vector Machine. For Nearest Neighbor and the Support Vector Machine try two parameters for each model. Use all input features
4. Evaluation: Display the Confusion matrices, ROC curves, and macro F1 scores

Report

1. One paragraph: What do the ROC curves tell about this problem. For which class do we get the worst prediction? Why do you think is that? What remedy can you propose? Can you test the remedy?