

Data Mining Coursework (IS53023C^2022-23^1)

Opened: Tuesday, 14 February 2023, 12:00 AM

Due: Monday, 20 March 2023, 6:00 PM

Data Mining Assignment

This assignment represents 100% of the Data Mining module's mark. It is composed of Part 1 which is worth 40 marks, and Part 2 which is worth 60 marks. You can work in a team of 2 students for this assignment. One student per team will be chosen by the team as being the team leader – who will be in charge of coordinating the team's work, and of submitting the assignment in their account on VLE on behalf of all the team.

PART 1:

This task is based on the Sonar real data seen previously in class. Several objects which can be rock or metal cylinders are scanned on different angles and under different conditions, with sonar signals. 60 measurements are recorded per columns for each object (one record per object) and these are the predictors called A1, A2, ..., A60. The label associated with each record contains the letter "R" if the object is a rock and "M" if it is metal cylinder, and this is the outcome variable called Class.

Two datasets are provided to you: a training dataset in the sonar_train.csv file, and a test dataset in the sonar_test.csv file.

a) You are required to write a Python code implementing the simplest Nearest Neighbour algorithm (that is, using just 1 neighbour), with the Minkowski distance, both discussed in lecture of week 1. Your code will read the power q appearing in the Minkowski distance, and will classify each record from the test dataset based on the training dataset. Remember, to classify a record from the test set you need to find its nearest neighbour in the training set (this is the one which minimizes the distance to the test set record); take the class of the nearest neighbour as the predicted class for the test set record. After classifying all the records in the test set, your code needs to calculate and



display the accuracy, recall, precision, and F1 measure with respect to the class "M" (which is assumed to be the positive class), of the predictions on the test dataset. Run your code to produce results first for Manhattan distance and then for Euclidian distance, which are particular cases of Minkowski distance ($q=1$, and $q=2$, see lecture week 1).

b) Run your code for the power q as a positive integer number from 1 to 20 and display the accuracy, recall, precision, and F1 measure on the test set in a chart. Which value of q leads to the best accuracy on the test set?

The code, comments, explanations and results will be provided in a Jupyter notebook called Part1.

Note that in this task you are not to apply a library for the nearest neighbour algorithm, but you are required to compute the distances, find the nearest neighbour, and so code yourself this simple algorithm.

PART 2:

This task is based on a real credit risk data, and is to predict a response variable Y which represents a credit card default payment (Yes = 1, No = 0), using the 23 input variables as follows:

X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.

X2: Gender (1 = male; 2 = female).

X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).

X4: Marital status (1 = married; 2 = single; 3 = others).

X5: Age (year).

X6 - X11: History of past payment. One tracked the past monthly payment records (from April to September, 2005) as follows: X6 = the repayment status in September, 2005; X7 = the repayment status in August, 2005; . . .; X11 = the repayment status in April, 2005. The measurement scale for the repayment status is: -1 = pay duly; 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.

X12-X17: Amount of bill statement (NT dollar). X12 = amount of bill statement in September, 2005; X13 = amount of bill statement in August, 2005; . . .; X17 = amount of bill statement in April, 2005.

X18-X23: Amount of previous payment (NT dollar). X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .; X23 = amount paid in April, 2005.


Two datasets are provided to you: a training dataset in the creditdefault_train.csv file, and a test dataset in the creditdefault_test.csv file.

Using Python and any relevant libraries, you are required to build the best predictive model by tuning models using cross validation on the training dataset with each of the following algorithms discussed in this module: [k-Nearest Neighbours](#), [Decision Trees](#), [Random Forest](#), [Bagging](#), [AdaBoost](#), and [SVM](#). Out of the models tuned with the above algorithms, select the best model and clearly justify your choice, and evaluate its performances on the test set.

The coding, comments and explanations will be provided in your Python Jupyter notebook called *Part2*, which should include also the results. Moreover, for each algorithm mentioned above, include 1 chart in the notebook illustrating how accuracy of the models vary when you vary the values of one numeric hyperparameter only (at your choice).

Note regarding working in a team or individually, and what you need to submit:

- **You can work and submit in a team of 2 students** - in which case you should choose a team leader. As a team you should work on all the tasks. Include the names and student numbers of both of the team members on top of each notebooks Part 1 and Part 2, and indicate who is the team leader. The team leader must perform the submission from their account (hence only once) for both students.
- **Or you can work also work and submit alone** for this coursework. In this case you must tackle only point (a) in Part 1, and only 3 out of the 6 algorithms mentioned in Part 2 (at your choice, but **choose 3 only**). Include your name and student number on top of the notebooks Part 1 and Part 2, followed by the mention "**I worked and submitted alone**"

 creditdefault test.csv	28 February 2022, 6:49 PM
 creditdefault train.csv	28 February 2022, 6:49 PM
 sonar test.csv	28 February 2022, 6:49 PM
 sonar train.csv	28 February 2022, 6:49 PM

Submission status

Submission status	Submitted for grading
Grading status	Graded
Time remaining	Assignment was submitted 2 hours 19 mins early



Last modified Monday, 20 March 2023, 3:40 PM

File submissions



[part 1.ipynb](#)

20 March 2023, 3:40 PM



[part 2.ipynb](#)

20 March 2023, 3:40 PM

Submission comments

▶ [Comments \(0\)](#)

Feedback

Grade 70 / 100

Graded on Thursday, 11 May 2023, 9:33 AM

Graded by  Henry Musto

Feedback comments**Part 1 : 35/40**

Distances used: Euclidean, Manhattan

Marks broken down:

20 marks for the correct code;

10 marks for ...

◀ [How you are assessed](#)

Jump to...

[Late submissions and deferrals](#) ▶