ML I supervised learning : project

NICOLAS LE HIR
nicolaslehir@gmail.com

TABLE DES MATIÈRES

INTRODUCTION

All processing should be made with python3.

A pdf report is expected in order to present your work. There is no length constraint on the report, you do not need to write more than necessary. The goal of writing a report is that you understand what you did with the project (an also that I understand more easily too and can give you some useful feedback).

If you use notebooks, you may also write your explanations in markdown inside the notebook, instead of writing a pdf file.

The 5 parts of the project are independent.

1 PART 1 : ARTIFICIAL DATASET GENERATION

The goal of this exercise is to work with statistical notions such as mean, standard deviation, and correlation.

Write a file named **artificial_dataset.py** that generates a numerical dataset with 300 datapoints (i.e. lines) and at least 6 columns and saves it to a csv file named **artificial_dataset.csv**.

The columns must satisfy the following requirements :
— they must all have a different mean
— they must all have a different standard deviation (English for "écart type")
— at least one column should contain integers.
— at least one column should contain floats.

— one column must have a mean close to 2.5.
— some columns must be positively correlated.
— some columns must be negatively correlated.
— some columns must have a correlation close to 0.

2 PART 2 : DEFINITION OF A METRIC

A dataset representing a population is stored in **dataset.csv** inside the **project/ex_2_metric/** folder.

Define a **metric** in this dataset, which means define a **dissimilarity** between the samples, by taking into account all their features (columns of the dataset).

Some features are numerical and others are categorical, hence you can not use a standard euclidean metric, and you need to define a custom metric, like we did in the **code/metrics/hybrid_data/** exercise during the course. Compute the mean dissimilarity and the standard deviation of the dissimilarity distribution that you obtain, and save the dissimilarity matrix to a file (e.g. a npy file).

Importantly, you must define and explain which features are more important with this metric, since you have to balance the contribution of all the features. Your metric should be meaningful in the sense that not all feature values should induce the same contribution to the dissimilarity : the music style "technical death metal" is closer to "metal" than it is to "classical".

3 PART 3 : PREDICTION OF THE WINNER OF A NBA GAME (CLASSIFICATION)

We would like to predict the winner of a Basketball game, as a function of the data gathered at half-time.

The dataset is stored in **project/ex_3_classification_NBA/** :
— The inputs x representing the features are stored in **inputs.npy**.
— The labels y are stored in **labels.npy**. If the home team wins, the label is 1, -1 otherwise.

You are free to choose the classification method. **However**, it is required that you explain and discuss your approach in your report. For instance, you could discuss :
— the performance of several methods and models that you tried.
— the choice of the hyperparameters and the method to tune them.
— the optimization procedure.

Your objective should be to obtain a mean accuracy superior than 0.85 on a test set or as a cross validation score.
https://scikit-learn.org/stable/modules/model_evaluation.html#accuracy-score
https://scikit-learn.org/stable/modules/cross_validation.html
Several methods might work, including some methods that we have not explicitely studied in the class. Do not hesitate to try such methods.

4 PART 4 : PREDICTION OF THE AMOUNT OF ELECTRICITY PRODUCED (REGRESSION)

We would like to predict the amount of electricity produced by a windfarm, as a function of the information gathered in a number of physical sensors (e.g. speed of the wind, temperature, ...).

The dataset is stored in **project/ex_4 _regression_windfarm/** :

— The inputs x are stored in **inputs.npy**.

— The labels y are stored in **labels.npy**

The instructions are the same as in 3.

Your objective should be to obtain a R2 score superior to 0.85 on a test set or as a cross validation score.

https://fr.wikipedia.org/wiki/Coefficient_de_d%C3%A9termination

https://scikit-learn.org/stable/modules/generated/sklearn.metrics.r2_score.html

Several methods might work, including some methods that we have not explicitely studied in the class. Do not hesitate to try such methods.

5 PART 5 : APPLICATION OF SUPERVISED LEARNING

Pick a dataset and perform a supervised learning on it. Ideally, your algorithm should answer an interesting question about the dataset. The supervised learning can then be either a **classification** or a **regression**.

You are free to choose the dataset within the following constraints :

— several hundreds of lines

— at least 6 attributes (columns), the first being a unique id

— some features may be categorical (non quantitative).

If necessary, you can tweak an existing dataset in order to artificially make it possible to apply analysis ans visualization techniques. Example resources to find datasets :

— Link 1

— Link 2

— Link 2

— Link 4

You could start with a general analysis of the dataset, with for instance a file **analysis.py** that studies :

— histograms of quantitative variables with a comment on important statistical aspects, such as **means** , **standard deviations** , etc.

— A study of potential **outliers**

— Correlation matrices (maybe not for all variables)

— Any interesting analysis : if you have categorical data, with categories are represented most ? To what extent ?

If the dataset is very large you may also extract a random sample of the dataset to build histogram or compute correlations. You can discuss whether the randomness of the sample has an important influence on the analysis result (this will depend on the dataset).

Whether it is a classification or a regression, you must provide an **evaluation** of your processing. For supervised learning, this could be an average squared error, coefficient of determination (R2 score), etc (https://scikit-learn.org/stable/modules/model_evaluation.html).

Short docstrings in the python files will be appreciated, at least at the beginning of each file.

In our report, you could include for instance :

— general informations on the dataset found in the analysis file.

— a potential comparison between several algorithm / models that you explored, if relevant

— a presentation of the method used to tune the algorithms (choice of hyperparameters, cross validation, etc).

— a short discussion of the results

Feel free to add useful visualizations for each step of your processing.

6 CODE STYLE

Add a short docstring at the top of each file and in functions, if relevant.
You are encouraged to use **type hints**.
https://docs.python.org/3/library/typing.html
http://mypy-lang.org/

7 THIRD-PARTY LIBRARIES

You may use third-party libraries, but need to slightly explain their usage in your report (choice of hyperparameters, etc.)

8 ORGANISATION

Number of students per group : 3.

Deadline for submitting the project :

— 1st session (October 13th, 14th) : November 13th.

— 2nd session (January 5th, 6th) : February 5th.

The project should be shared through a github repo with contributions from all students. Please briefly indicate how work was divided between students (each student must have contributions to the repository).

Each exercise should be in its own folder.

If you used third-party libraries, please include a **requirements.txt** file in order to facilitate installations for my tests.

https://pip.pypa.io/en/stable/user_guide/#requirements-files

You can reach me be email if you have questions.