Report

M-MLR-900

(Alexandre Guichet, Alexis Auriac, Benjamin Feller)

Machine Learning - Supervised Learning

January 30, 2023

Introduction

We did part 1, 2, 3, and 4 but did not have time to do part 5.
Division of labor:

- Alexandre: part 1
- Benjamin: part 2
- Alexis: part 3 and 4, writing report

Part 1

Problem 1

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 or to a numpy array in a binary python file.

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.

Solution.

See `exercise_1/artificial_dataset.py` for the code.

The generated data can be found in `exercise_1/data.npy`, it contains 300 dataponts with 6 columns.

Let's go over each point mentioned in the subject one by one.

All columns must have a different mean

- column 1: 2.58
- column 2: 0.898
- column 3: 1.686
- column 4: 4.707
- column 5: 0.349
- column 6: 10.115

All columns must have a different standard deviation

- column 1: 1.752
- column 2: 0.827

- column 3: 1.998
- column 4: 1.889
- column 5: 2.430
- column 6: 1.352

At least one column should contain integers
Column 1 contains integers.
At least one column should contain floats
All columns except column 1 contain floats.
One column must have a mean close to 2.5
Column 1 has a mean of 2.58.
Columns correlations
Using `numpy.corrcoef` to get a correlation matrix for column 4, 5, and 6 we get this:

$$\begin{pmatrix} 1. & -0.5595593 & 0.66913029 \\ -0.5595593 & 1. & 0.03168539 \\ 0.66913029 & 0.03168539 & 1. \end{pmatrix}$$

Column 4 is **negatively correlated** with column 5.
Column 4 is **positively correlated** with column 6.
Column 5 is **has a correlation close to 0** with column 6.

Part 2

Problem 2

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".

Solution.