laSalle ENG
Universitat Ramon Llull

Practical assignment 2
Data processing

Subject: Data Mining
Deadline: 09 December 2022
Year: 2022/2023

Optimization, pre-processing and Instance-Based learning (IBL) with scikit-learn

PART 0: MOTIVATION AND OBJECTIVES

Motivation

Scikit-learn is by far one of the pillars for machine learning in Python as it allows us to build machine learning models as well as providing utility functions for data pre-processing such as dimensionality reduction, search, optimization and applied IBL.

Objectives

a) Learn to master the scikit-learn Python package.
b) Know the different implementations of the algorithm IBL example-based learning.
c) Get skills with selection, normalization, pre-processing and searching attributes.
d) Work with the concept of "learning optimization".

Instructions

Read all instructions in this section thoroughly.

- Try to understand the python script skeleton (more explanations below).
- Find the comment “Introduce something here”.

Formatting: Your solution must be implemented in python, following the precise instructions included in Part 2. The programming exercise ask you to add code and analyse the results. Your analysis must be clear and well-argued and upload along with your Python file (or Jupyter notebook). The submission will be evaluated according the answers to questions asked and result analysis.

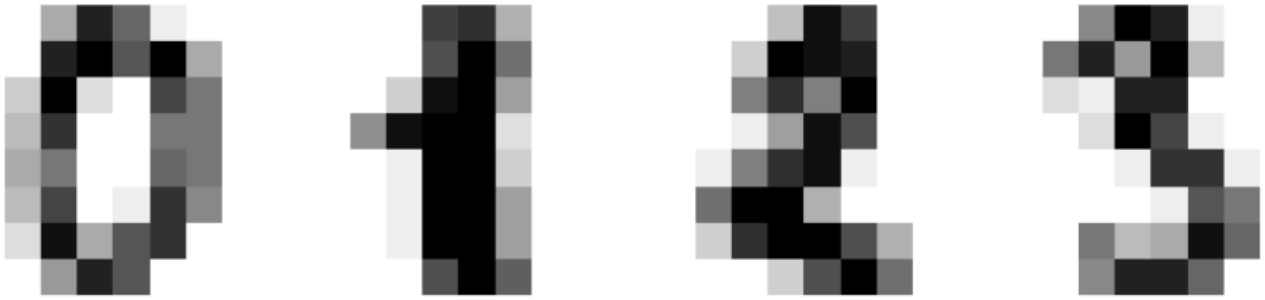
PART I: ENVIRONMENT

Before starting the exercise, you will need to make certain that you are working on a computer with particular software:

- Python 3.8.x (https://www.python.org/downloads)
- numpy (https://www.numpy.org/)
- scikit-learn (https://www.scikit-learn.org/)
- Python editor (PyCharm, Atom, Spyder, Notepad++, Jupyter).
- Necessary resources for the assignment can be download from eStudy.

PART II: DATASET

The sklearn digits dataset is made up of set of 8x8 pixel images of digits. Each image is of a hand-written digit in grayscale values. Here some samples:



PART III: INSTRUCTIONS

Follow the following steps:

- **Data analysis:** After loading the dataset, try to do an initial data exploration. For example, figure out the number of samples, shape, target variable, visualization of an array, etc. Read the dataset information by calling the DESCR attribute. You can visually check the contents of the arrays using the matplotlib library (https://matplotlib.org/). Also, try to do a statistical description of the dataset using basic measures as standard deviation, number of samples for classes, etc. Results can be added to the report taking a terminal screenshot or plots.

- **Pre-processing:**
  - Split the data in data train (75%) and data test (25%) (Help: data-split). Why is the purpose of splitting data?
  - Normalize the data to zero mean and unit variance (z-score normalization) (help: data-normalization). What is data normalization and why is it important?
  - PCA analysis comparison (train data): by (i) PCA (help: decomposition-pca), and (ii) SVD (help: decomposition-svd). What is the difference in results between SVD and PCA? What is the number of principal components for each method?
- **Learner: kNN classification:**

  After selecting the principal components by 95% of variance explained:
  - Select the appropriate k value using a 10-fold cross validation (help: k-fold).
  - Train your k-NN classifier using the best k value and predict the labels of the test data. Your classifier should allow as input the training data matrix, the training labels, the instance to be classified, the value of K (help: knn-classifier).
  - Create an evaluation function to measure the performance of your classifier. This function will call the classifier function in part a on all the test instances and in each case compares the actual test class label to the predicted class label. It should take as input the training data, the training labels, the test instances, the labels for test instances, and the value of K. Show a final classification report (help: metrics)
  - In order to compare accuracy values for different numbers of neighbours, do an analysis over various values of “k” for the k-NN classifier, including the optimal k. You can use the accuracy metric (hint: use a loop and print). Present the result as graphs with K in the x-axis and the evaluation metric (accuracy) on the y-axis.

- **Additional information:**

  This practical assignment will be completed in groups of a **maximum of two people** and only one of them must to submit the code to the eStudy, in a single compressed file (a zip file is strongly recommended) named "surname_partner1_surname_partner2", which will contain the code and your individual report. Reports are individual, so the other partner must to submit only their individual report.

  Only reports in PDF format are accepted. The report also should include any analysis, discussions, additional figures or explanations of the results.