DeepL
Subscribe to DeepL Pro to translate larger documents.
Visit www.DeepL.com/pro for more information.

Task 1. Supervised Learning: Classification

Study datasets of the UCI repository and the Kaggle platform

Introduction

The aim of this work is to study and optimize classifiers on datasets. Each group in the lab will study two datasets, one from the UCI repository and one from the Kaggle platform.

The training and optimization of classifiers in the UCI dataset will be done exclusively with the skicit-learn functions while in the Kaggle dataset you will have to use an optimization library (whichever you want).

There are 14 different UCI datasets (U01-U14) and 14 different Kaggle datasets (K01-K14). Each team has a unique combination of U-K datasets. You can find which datasets codes correspond to the number your team has in helios in the Teams - Datasets table.

To find out which dataset corresponds to the UCI code consult the UCI Datasets table.

To find out which dataset corresponds to the Kaggle code consult the Kaggle Datasets table.

Part 1. UCI dataset (40%)

Introduction and overview

Import the dataset from the text file into your notebook. Then in markdown cells write the basic information about it:

- Brief presentation of the dataset i.e. what is the problem it describes.
- Did you have to make any modifications to the plain text files for the import; if so, what are they?
- Give the number of samples and attributes, and the type of all attributes. Are there unordered features and what are they?
- Are there headings? Line numbering?
- What are the class labels and on which column are they located?
- Are there any missing values? How many samples have missing values and what is their percentage of the total?
- What is the number of classes and their sample percentages of the total? If we consider a dataset to be unbalanced if any one class is 1.5 times more frequent than another (60%-40% in binary datasets) estimate whether the dataset is balanced or not.

Preparation

- Divide the dataset into a train set and a test set with 30% of the samples in the test set. If your datasets are by description already split into train and test, as long as the percentages are close to 70-30 you can use them as they are. If they are not, consolidate train and test and proceed to the same split.
- If there are missing values, manage them and justify them.
- Manage any categorical and/or unordered features and justify.

Classification

Classifiers

At the UCI we will study the classifiers

- dummy,
- Gaussian Naive Bayes (GNB),
- KNeirestNeighbors (kNN),
- and Logistic Regression (LR).

Metrics

The optimisation and presentation of the results should be done separately for each of the two metrics:

- accuracy, and
- F1-score (macro in multiclass problems).

Cross-validation scheme

For all experiments you will use 10-fold cross-validation.

Out-of-the-box performance

First we will see how the classifiers behave without any optimization (out-of-the-box) and with all parameters set to default values.

Train all estimators with a simple fit on the entire training set and calculate their performance on the test set for the two metrics.

Briefly and comparatively present their indulgence:

1. in a markdown table, and
2. in bar plot comparison,

and comment on their performance.

Optimization

For all classifiers, optimize their performance through the procedures

- pretreatment, definition of
- pipelines, and
- finding optimal ultrameters by grid search with cross-validation

For the best model of each classifier, train it on the entire train set and evaluate its performance on the test set. In addition, for the best models, record the train and test times.

Results and conclusions

Briefly and comparatively present their indulgence:

1. in a markdown table where, in addition to the two metrics, their change with respect to out-of-the-box as well as both times will be included, and
2. in a bar plot comparison that includes the change (without the times).

Comment on their overall performance as well as the change from out-of-the-box performance.

For the best and worst classifier (excluding dummies) in terms of correctness print the confusion tables graphically (e.g. seaborn) and comment.

Which classifier do you finally recommend for this problem and why? Can you give any explanation for its good performance on the problem, absolutely and/or relative to the others (except for the dummies)?

Part 2. Kaggle dataset (60%)

Overall objective

In the second part of the assignment you are asked to study a dataset from Kaggle. In general, the selected Kaggle datasets are larger to much larger than those from UCI.

You will find that you are given more freedom of choice on how to find the optimal models for your dataset. The overall goal in this part is threefold:

- Optimize the classifiers to achieve the best possible performance with the right methodology across all options.
- Be able to describe in a brief and meaningful way your choices when experimenting.
- Present your conclusions in a complete and eloquent way.

Import of the dataset

Kaggle

Since they are Kaggle datasets the simplest way is to work in Kaggle and that is the recommended way.

Just go to "Code", create a new notebook, "Add data", search for the dataset by name, and "Add" it.

Then just run the first ready cell and it will show you the path for all the files in the dataset.

Colab

To work in Colab you need to import the data from Kaggle. For this you need an API key from Kaggle. Additionally, Colab has the disadvantage that, unlike Kaggle, after 30 minutes of no activity it deletes all data in the "/content" folder (the vm location). There is however the option to mount your Google Drive and have persistancy.

Follow the guide "Downloading Kaggle datasets directly into Google Colab" which shows step-by-step how to achieve the above.

Classifiers

In the case of the Kaggle dataset you will work with two classifiers

- Mylti-Layer Perceptron (MLP), and
- Support Vector Machines (SVM).

Overview

Use the Kaggle descriptions and the data itself to understand the dataset and the task. Make sure you have insight into all the files in the dataset, if it has more than one.

Provide in markdown cells the basic information about the dataset, as you did in the UCI dataset. You can include any other comments you think are important if you want.

Metrics

Select the metric(s) you will work with and justify your choice.

Train-test split and CV shape

If you use cross-validation via your library, you choose the train-test rate and cross-validation scheme. Justify your choice. If cross-validation is not used or a variation is used, describe the procedure and justify any choices.

Out-of-the-box performance

First we will see how the classifiers behave without any optimization (out-of-the-box) and with all parameters set to default values.

Train all estimators with a simple fit on the entire training set and calculate their performance on the test set for the two metrics.

Briefly and comparatively present their indulgence:

1. in a markdown table, and
2. in bar plot comparison,

and comment on their performance including the dummies as baseline.

Optimization

For the two classifiers, optimize their performance through the preprocessing procedures,

-

- definition of pipelines, and
- finding optimal ultrameters by grid search with cross-validation

For model selection you will use functions from sklearn and necessarily an optimization library of your choice (Ray, Optuna, other).

Work with each classifier individually to optimise it. As you experiment, note your choices and conclusions, as well as any other elements you consider important (e.g. pipelines, successive hyperparameter ranges, etc.) so that you can then describe the process.

Documentation of the procedure

Provide a concise and concise description of the entire process you followed to arrive from the out-of-the-box to the optimal model for each classifier. We should be able to understand at each step or stage what you did and why you did it, always briefly and qualitatively.

Presentation of results

Present in detail the final performance evaluation of the two classifiers individually and comparatively.

Use for presentation and explanations all the tools available such as tables, graphs and natural comments.

Focus on your most important observations and analyse them thoroughly. Don't include something if what it shows we have already seen qualitatively and is simply repeated with minor quantitative differences.

Conclusions

Explain what is the final classifier model you propose for your dataset and why. Here you can also talk about individual choices if based on different criteria one model or the other is superior.