Exercise 1: Classification

VU Machine Learning
Summer Semester 2021

Goals

In this exercise, you will perform classification using multiple algorithms on a number of different datasets. The goal is to experiment with these different combinations and settings, and go through the whole machine learning process from getting your data, preparing it, making your predictions, to evaluating and comparing the results, discussing them and drawing final conclusions. You will experience how the size of your dataset affects the runtime and how different preprocessing strategies and parameters impact the performance of classifiers. You will learn how to measure and compare these changes, and thus improve the overall performance. You should investigate such changes, provide comparisons of your results and analyze them, to be able to discuss your findings.
You will also participate in our own Kaggle Competition within this course, in which you might earn bonus points for the exercise.

Groups

Groups of exactly 3 students

- It is beneficial to remain in the groups from Exercise 0, but you may switch
- **Work as a team**

Deliverables

Submission package

- 10-15 page report (including tables & diagrams)
- Code & scripts

Presentation

- Presentation dates will be published in the upcoming weeks
  In total, you will present 2 exercises out of 3; either exercise 1 or 2, AND exercise 3.

Tools

Generally, you shall use APIs to enable repeatable, scalable experiments. Tools you can use include:

- Python / scikit
- WEKA (http://www.cs.waikato.ac.nz/ml/weka/) - use the API, not GUI
- R (http://www.r-project.org/) – advanced & powerful software – if you know R already, or you want to learn it
- Rapid Miner
- MATLAB

Exercise

Datasets:

You will use 4 datasets:

- 2 from our own Kaggle competition
- 1 from Exercise 0 (unless you encounter issues)
- 1 that you still need to find

IMPORTANT: You must choose diverse datasets, with different properties

- number of samples – small vs. large
- number of dimensions – low vs. high dimensional
- number of classes – few vs. many classes
- pre-processing needed…

Classifiers:

You will choose 3 different classifiers from different types of learning algorithms
So in the end you will have 4 × 3 combinations of datasets × classifiers

Task:

Perform necessary steps as presented in the lecture:
Importing the data
Data Exploration and Preprocessing (missing values, outliers, scaling, encoding, etc.)
Carry out the **classification**:

- Run classifiers, and **Experiment** with:
  - Different classifiers and your datasets
  - Different parameter settings (= several results per classifier per dataset, not only random/best)

Evaluate and analyse the **performance** (primarily effectiveness, but also provide basic details on efficiency):

- Choose **suitable, multiple performance measures**
- Make **valid** comparisons (among the classifiers, across your datasets, parameters, preprocessing effects…)
- Can you identify any patterns/trends?
  - Which methods work well and which did not, is there e.g. one method outperforming the others on all datasets?
  - How do the results change when preprocessing strategies change? How sensitive is an algorithm to parameter settings?
  - Are there differences across the datasets? Design your experiments so that you can investigate the influence of single parameters.

Perform **significance testing** against at least one **baseline**

- Make sure you are comparing results that **can** be compared to each other!

Compare **holdout** to **cross-validation**

- Pay attention to your splits and settings
  Are there differences? Why? In which metrics? What could have caused it?
- Compare/document changes in **runtime** behavior with the changing e.g. dataset size

Summarize your results in **tables, figures**!
Document your **findings, issues:**

Write a report
Make a presentation

Upload your best results to **Kaggle competition** (more information below)

You do not need to implement the algorithms, rely on libraries/modules
- Code just for loading data, pre-processing, running configurations, processing/aggregating results, ...

Pointers for your project

Apply the knowledge from the lectures
Document the whole process
Carefully design your experiments

- work out your **experiment design as a group**

Important points:
- Explain your choice of **datasets**, introduce them, their characteristics
- Briefly describe the **preprocessing** steps and argue why you chose them
  Evaluate their **impact** on the results (mainly scaling)
- Explain your choice of **classifiers**, describe their characteristics
- Argue on your choice of **performance measures**
  Think and find multiple, suitable measures, argue why you chose them (why are they necessary, what do they measure/tell us about the performance), and if they are sufficient
- In the report, include a paragraph briefly describing the steps you took to **ensure** that the **performance of the classifiers can be compared** (think if the comparison makes sense & research what needs to be fulfilled in order to e.g. compare the performance of multiple classifiers on one dataset, how to compare the impact of parameter changes etc.)
- Discuss your experimental results, compare them using **tables** and **figures**
- Try to pay attention to clarity and readability of your tables and visualizations (scale, legend, axis labels, etc.)
- Provide an **aggregated** comparison of your results as well - i.e. a big table of the best settings and results for all combinations

Presentation

Presentation dates will be published in the upcoming weeks
- 10 <= minutes
- Present findings specific to your project
  Interesting findings, issues, conclusions
  Usefulness of the algorithms for your datasets
  Comparison of classifiers

Keep the overview of datasets, algorithms brief, Do not repeat materials from the lectures, No code in your presentation slides

Report

10-15 pages long
Full report of your work, document the whole process

- Datasets, preprocessing, algorithms, experiments, explanation of your choices, arguments, comparisons, analysis, discussion, tables, figures, conclusion.....
- Do not include code in the report (only in the your submission package)

Competition

We will use Kaggle In-class (https://inclass.kaggle.com) for a competition
Submission requires a simple CSV file

- for each sample in the test set: <id>,<predicted class>

Pick two of the datasets provided in TUWEL
Number of uploads to Kaggle per day is limited - start early!

- That way you also have early feedback on your results compared to other groups!
  First try locally what works, only then upload to Kaggle
- 15% Bonus points for the top 3 teams!
  +5% Bonus if you also have your notebook running in Kaggle
  See https://www.kaggle.com/notebooks
  Keep it private, share it with mayer@ifs.tuwien.ac.at