Information on the Programming Project
Machine Learning PS (SS 2023)

PS Team 2023 Jakob Hollenstein Sayantan Auddy David Drexel Danielle McKenney

Read the following information carefully!

I. Overview of the Project

A. First things first

Start by reading this description thoroughly. Then, before you start working on the dataset, thoroughly read through the report template and start discussing what the questions in the template require you to do.

It goes without saying that the rules of academic honesty apply. In short, this means doing your own work and not cheating, not presenting the work of others as your own and not submitting previously submitted work.

You are not allowed to collaborate on the report outside your designated groups.

B. Problem selection and assignment

In this project your task is to solve a problem using machine learning methods. There are three different problems with respective datasets. The datasets will be assigned so that all of the problems are covered evenly. You can and should state your preference and we will try to accommodate that. The datasets are available alongside notebook stubs on the jupyterhub. The notebook stubs show how to load the datasets and at the end these are also used to test your methods/your solution. Some basic information about the datasets is also provided in the notebooks.

C. Report & Implementation

To solve the machine learning problem *principled* experimentation is necessary, i.e. you will have to try different methods and evaluate which work best. This experimentation, and your results, must be described in the report. Use the provided **report template**. The report template contains questions. Your report must answer those questions because we grade your answers in the report. The report must be two pages (three pages for the 3 person team) in length; see the report-template for details.

D. Comparing methods

First you will need to investigate the dataset to decide what the problem is about and what methods can and should be applied.

Next, you will need to try *different* methods. Two methods are *different* if they are related to different PS topics, i.e. logistic regression vs. SVM vs. decision trees are three different classification methods. SVM with a gaussian kernel vs. SVM with a quadratic kernel are *not* different methods. Two neural network architectures are only considered different methods when their architectures are substantially different, e.g. fully-connected vs. convolutional network.

You have to experiment with different methods (at least two methods; for teams with three people: at least three different methods). You can re-use anything from the PS notebooks, and any methods that are **already installed and available on our jupyterhub** (i.e. sklearn, torch).

Note that you will have to understand the method you use, be able to explain how it works, and most importantly explain what you used. Somebody else has to be able to replicate your results from the description – without knowing the content of the PS (but you can assume knowledge of the VO).

Obviously the description is easier for methods that we have covered. If the method is different from what we covered in the VO, you need to explain how it is different, i.e. how we can get from the methods we already know to the method you used.

Your goal is to solve the machine learning problem as adequately as possible. In the previous year, there was more focus on the implementation of the models, for example: no sklearn. This year, models from sklearn are also allowed you should focus more on finding a good model and following a principled way of achieving this (e.g., Slide set “5 Overfitting and Generalization” and look for “Generalization Error vs. Model Complexity”).

For each method, different settings are possible, for example different kernels, different regularizers. Finding the right settings in a principled way is essential for machine learning.

3 person team: Since there are more people in this team, there needs to be even more thorough experimentation and thus a longer report.

E. Model size limitation

You need to be able to store and load your model (weights/parameters) from a file so that it can be evaluated without retraining. The maximum allowed size for your stored model is **50MB**. Hence, you should try to build a model that is as small as possible while being accurate enough for your dataset.

II. Grading

The grading is based on the report. The template contains questions and your report has to provide answers to those questions. We will evaluate your report based on how principled your approach is and how well you are able to answer the questions. As a hint: if you read the report as if it were coming from someone else, would you understand what had been done? Could you reproduce what is described in the report? Does the approach convince you?

In addition, you have to a.) provide a version of your model that runs on the jupyterhub to evaluate the *test* dataset and b.) provide a runnable version of your training scripts as a zipfile on OLAT. (See *Submission*).

III. Misc

- Two years ago we had a leaderboard for bonus points and to compare the methods. This is **not** part of the project this year. However, the code is still in place and in some parts of the code you will see reference to the “leaderboard” test functions. The test data is still accessible through these functions but now it is transparent to you; before the test data was hidden.
- The dataset is already provided on the hub. **Do not upload any datasets**.

IV. Submission

At the end of the project, that is until the deadline **2023-07-05**, you need to submit:

1) The **report PDF** on **OLAT**
2) The runnable training scripts and evaluation scripts as **ZIP** on **OLAT**
3) The working test script on **JupyterHub** as a **Notebook Submission**
4) Your code must be complete and runnable.
5) Check that you do not submit a **model that is too big**
6) Do not train on the submitted **test**-notebook on jupyter-hub! You can create a separate training notebook, or you can use a flag that disables the training (and that is set so it disables training by default!)
7) Do not change the name of the notebook file! Do not replace the notebook! (Because it will be used for testing your model on the test set.)
8) We will *deduct 50% of your report points* if your code/notebook submissions not comply with the requirements stated above.