Introduction to Machine Learning (67577)
Hackathon 2022

Teachers: Dr. Matan Gavish, Dr. Gabriel Stanovsky

TAs: Avihu Dekel, Dan Derazne, Gilad Green, Gili Lior, Michael Tomer

Tzars: Nitay Alon, Yuval Lahav, Eitan Wagner

Second Semester, 2022

Contents

1 General Instructions

Welcome to the IML Hackathon 2022! Please read carefully the entire document before selecting a challenge to work on.

- On **Wednesday 01.06.2022 7pm** we all meet in the auditorium for the reveal of this years challenges. The hackathon ends on **Friday 03.06.2022 at 11am**.
- During the hackathon you will be working in teams of **three or four students only**.
- The instructions and data for both challenges will be available on the course Moodle.
- After reading through the entire document select **one** of the challenges and solve it as best as you can.
- **We will provide you with a set of test examples for which you will submit your predictions**.
- In addition to your predictions, your will submit your code. To ensure we are able to run your code without errors we require all submissions to define and use a virtual environment. Submit the environment's `requirements.txt` file. Instructions on the virtualenv and the freeze command are available on Moodle. The virtual environment will allow you to install packages via `pip install <package_name>` and use packages that are not installed on the CS computers.
- Solutions must be written in Python3. To use Python 3 on CS computers, use python3 instead of python.
- Submission of your solution is done through the course Moodle. Submit a single zip file named `id_number.zip`, where `id_number` is a 9-digits ID of one team member. The file should include:
  - A folder named `task1` or `task2`, depending on the challenge you choose.
    This folder should include:
    - `predictions.csv` – your prediction file.
    - Your code files.
    - Any supplementary files you used (trained models etc.).
    - `requirements.txt` — the required packages for your virtual environment, as described above.
  - `README.txt` — contains a file list and a brief description of each file.
  - `USERS.txt` — contains the logins and IDs of all the team members. Use one line per student, in the format login, ID.
  - `project.pdf` -– a written description of your project, up to 2 pages long, as a PDF file. Explain your solution, describe your work process, and do your best to justify any design decisions you have made. Feel free to include supporting figures if you want.
- Make sure to write good, clean and documented code. Our ability to understand what you are performing and why is important for receiving a high grade.
- We advise to begin with finding a basic solution that works. Then try to improve it as much as you can in the given time.

1.1 Evaluation

For both challenges your grade will be determined by two factors (in descending order of importance):

1. The quality of your prediction on the test set. This will be based on a ranking between the performance of all participants, where any team that will produce a model that has some

reasonable performance ('reasonable' depends on which task you chose) will receive a grade of 80 and the top ranking teams will receive a grade of 100, and there will be some distribution between these two extremes.

2. The quality and depth of your written description of your work that you wrote in the PDF. We will grade your written description based on the following:
   - Did you describe the dataset, and any challenging characteristics it has?
   - Did you describe (briefly) the data cleaning and preprocessing?
   - Did you describe the considerations that guided your design of learning systems?
   - Did you describe (briefly) various methods you tried and the results you obtained?
   - Did you describe the learning system you ended up using?
   - Did you provide a prediction (and explanation) of the generalization error you expect your system to have?

Good Luck!

1.2 Verify Before Submission

IMPORTANT: Our tests have **zero tolerance** hence you have to follow the requested formats. Submission that fail to comply with the requested format will receive zero points.

- You are obligated to submit files in the defined file folder structure as below.
- Output data has to be in the matching format.
- Your code should only call files present in the submitted folder. When doing so use a relative path. For example : `"weights.txt"` is good while `"C:\Users\name\IML\weights.txt"` is bad.
- You are required to submit a `requirements.txt` file.
- Your code cannot throw any exception or error when running.
- Your code has to finish prediction (including any necessary loading or initialization) in a time frame of 5 minutes when running on the CS computers.
- After zipping your project, it should have a maximum size of 20 MB.
- Submit only **one** of the challenges (see directory structure).
- Submit the zip file through the Moodle of only **one** of the team members.
- You must use the following directory structure in your submission zip file:

File or directory	**Description**
`<student-id>.zip`	Submission zip file
├── `USERS`	
├── `README`	
├── `project.pdf`	
├── `task1`	***Only if submitting task 1***
│ ├── `predictions.csv .3`	
│ │ `requirements.txt`	
│ ├── `regerssion.py`	
│ └── *`<other source files>`*	
└── `task2`	***Only if submitting task 2***
├── `predictions.csv .3`	
│ `requirements.txt`	
├── `classifier.py`	
└── *`<other source files>`*	