TASK 1A

Cross-validation for Ridge Regression (graded)

You are in the group Siuuuu consisting of angarcia (angarcia@student.ethz.ch (mailto://angarcia@student.ethz.ch)).

1. READ THE TASK DESCRIPTION

2. SUBMIT SOLUTIONS

3. HAND IN FINAL SOLUTION

1. TASK DESCRIPTION

This task is about using **cross-validation** for **ridge regression**. Remember that ridge regression can be formulated as the following optimization problem:

$$\min_{\mathbf{w}} \sum_{i=1}^{n} (y_i - \mathbf{w}^T \mathbf{x}_i)^2 + \lambda ||\mathbf{w}||_2^2$$

where y_i is the label of the ith datapoint, $\mathbf{x}_i$ is the vector of features for the ith datapoint and λ is a hyperparameter.

Consider the following set of regularization parameters:

λ_1	λ_2	λ_3	λ_4	λ_5
0.1	1	10	100	200

Your task is to perform **10-fold cross-validation** with **ridge regression** for each value of λ given above and report the **Root Mean Squared Error (RMSE) averaged over the 10 test folds**. In other words, for each λ, you should train a ridge regression 10 times leaving out a different fold each time, and report the average of the RMSEs on the left-out folds. More background on K-fold cross-validation and some hints on useful library functions can be found here (https://scikit-learn.org/stable/modules/cross_validation.html) in the scikit-learn user guide. You can assume that each datapoint in your dataset is sampled independently from the same distribution (iid), so section 3.1.2.1 of the user guide should be particularly relevant.

The RMSE is defined as

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2}$$

where y_i are the ground truth labels and $\hat{y}_i$ estimations made by your regressor. You should perform the linear regression on the original features, i.e., you **should not perform any feature transformation or scaling**. Note that the goal here is to test your understanding by accurately performing the described cross-validation of the machine learning method we've described (linear regression with ridge). This is different to future tasks, where the goal will normally be to have the most accurate predictions.

DATA DESCRIPTION

Download handout (/static/task1a_do4bq81me.zip)

In the handout for this project, you will find the the following files:

- **train.csv** - the training set

- **sample.csv** - a sample submission file in the correct format
- **template_solution.py** - a template file that will guide you through the implementation of the solution

Each line in train.csv represents one datapoint by its label y, and its features x1 to x13:

```
y,x1,x2,x3,x4,x5,x6,x7,x8,x9,x10,x11,x12,x13
22.6,0.06724,0.0,3.24,0.0,0.46,6.333,17.2,5.2146,4.0,430.0,16.9,375.21,7.34
...
```

For your convenience, we further provide a sample submission file:

```
13.1
9.6
6.0
14.5
22
```

We provide a template solution file that suggests a structure for how you can solve the task, by filling in the TODOs in the skeleton code. It is not mandatory to use this solution template but it is recommended since it should make getting started on the task easier. While it is not mandatory, we encourage you to implement ridge regression from scratch, to get a deeper understanding of the course material.

SUBMISSION FORMAT

The submission file format should be identical to the format of sample.csv, i.e., it should have exactly 5 lines, each containing a floating point number that represents the average RMSE scores obtained for $\lambda_1, \lambda_2, \lambda_3, \lambda_4, \lambda_5$ (in this order).

Please keep in mind that, as a group, you have a limited number of submissions as stated on the submissions page.

EVALUATION

Your submission is evaluated based on the following formula:

$$\text{score}(\mathbf{v}) = 100 \cdot \sum_{i=1}^{5} \frac{|v_i - v_i^*|}{v_i^*}$$

where $\mathbf{v} = \{v_1, \ldots, v_5\}$ are your submitted values, $\mathbf{v}^* = \{v_1^*, \ldots, v_5^*\}$ is the reference solution. Your goal is to minimize this score i.e. bring your estimate as close as possible to our ground truth value.

GRADING

Task 1a is **graded differently to other tasks** . In all other tasks, we ask you to solve a machine learning problem and we score you based upon the prediction accuracy of your model. In this task, instead we ask you to simply replicate fitting a model and estimating its performance via cross validation under the instructions given above. You achieve a better score if your estimate of the RMSE is closer to that of the reference solution. You achieve a **pass (6.0)** if you achieve a better score than the single baseline, and a **fail (2.0)** otherwise. When handing in the task, you need to select which of your submissions will get graded and provide a short description of your approach. This has to be done **individually by each member** of the team. We will then compare your selected submission to our baseline. In addition, for the pass/fail decision, we consider the code and the description of your solution that you submitted. The following **non-binding** guidance provides you with an idea on what is expected to pass the project: If you hand in a properly-written description, your source code is runnable and reproduces your submitted csv, and your submission performs better than the baseline, you can expect to have passed the assignment.

⚠ Make sure that you properly hand in the task, otherwise you may obtain zero points for this task.

PLAGIARISM

The use of open-source libraries is allowed and encouraged. However, we do not allow copying the work of other groups / students outside the group (including work produced by students in previous versions of this course). Publishing project solutions online is not allowed and use of solutions from previous years in any capacity is considered plagiarism. Among the code and the reports, including those of previous years, we search for similar solutions / reports in order to detect plagiarism. Use of GPT3 Copilot or similar code/language generation tools in any capacity for writing code or reports will be considered and treated as plagiarism in the context of this course. Basic code autocompletion such as those used in the default setup of Sublime Text 3 are permitted. If we find strong evidence for plagiarism, we reserve the right to let the respective students or the entire group fail in the IML 2023 course and take further disciplinary actions. By submitting the solution, you agree to abide by the plagirism guidelines of IML 2023.

FREQUENTLY ASKED QUESTIONS

⊙ WHICH PROGRAMMING LANGUAGE AM I SUPPOSED TO USE? WHAT TOOLS AM I ALLOWED TO USE?

You are free to choose any programming language and use any software library. However, **we strongly encourage you to use Python**. You can use publicly available code, but you should specify the source as a comment in your code.

⊙ WHAT TO DO IF I CAN'T RUN THE CODE/SETUP AN ENVIRONMENT ON MY PC?

If you are having trouble running your solution locally, consider using the ETH Euler cluster to run your solution. Please follow the Euler guide (/static/euler-guide.md). The setup time of using the cluster means that this option is only worth doing if you really can't run your solution locally.

⦿ AM I ALLOWED TO USE MODELS THAT WERE NOT TAUGHT IN THE CLASS?

Yes. Nevertheless, the baselines were designed to be solvable based on the material taught in the class up to the second week of each task.

⦿ IN WHAT FORMAT SHOULD I SUBMIT THE CODE?

You can submit it as a single file (main.py, etc.; you can compress multiple files into a .zip) having max. size of 1 MB. If you submit a zip, please make sure to name your main file as *main.py* (possibly with other extension corresponding to your chosen programming language).

⦿ IN WHAT FORMAT SHOULD I SUBMIT THE REPORT?

The handin page of the submission server contains a simple textbox in which you should insert your report. It should consist of a couple of sentences explaining the main ideas and concepts of your solution. Every student writes and submits the report independently.

⦿ WILL YOU CHECK / RUN MY CODE?

We will check your code and compare it with other submissions. We also reserve the right to run your code. Please make sure that your code is runnable and your predictions are reproducible (fix the random seeds, etc.). Provide a readme if necessary (e.g., for installing additional libraries).

⦿ SHOULD I INCLUDE THE DATA IN THE SUBMISSION?

No. You can assume the data will be available under the path that you specify in your code.

⦿ CAN YOU HELP ME SOLVE THE TASK? CAN YOU GIVE ME A HINT?

As the tasks are a graded part of the class, **we cannot help you solve them**. However, feel free to ask general questions about the course material during or after the exercise sessions.

⦿ CAN YOU GIVE ME A DEADLINE EXTENSION?

⚠ We do not grant any deadline extensions!

⦿ CAN I POST ON MOODLE AS SOON AS HAVE A QUESTION?

This is highly discouraged. Remember that collaboration with other teams is prohibited. Instead,

- Read the details of the task thoroughly.
- Review the frequently asked questions.
- If there is another team that solved the task, spend more time thinking.
- Discuss it with your team-mates.

⦿ WHEN WILL I RECEIVE THE PRIVATE SCORES? AND THE PROJECT GRADES?

We will publish the private scores, and corresponding grades before the exam the latest.