TASK 1B

Regression (graded)

You are in the group Invitepls consisting of lrasmus (lrasmus@student.ethz.ch (mailto://[u'lrasmus@student.ethz.ch'])) and nmojado (nmojado@student.ethz.ch (mailto://[u'nmojado@student.ethz.ch'])).

1. READ THE TASK DESCRIPTION | 2. SUBMIT SOLUTIONS | 3. HAND IN FINAL SOLUTION

1. TASK DESCRIPTION

This task is about **linear regression**: given an input vector **x**, your goal is to predict a value **y** as a **linear** function of a set of feature transformations, $\phi(\mathbf{x})$.

DATA DESCRIPTION

Download handout (/static/task1b_ow9d3s.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

Each line in train.csv is one data instance indexed by an Id. It consists of one float for y and 5 floats for the vector x1-x5:

```
Id,y,x1,x2,x3,x4,x5
0,-5.522113571291328,1.764052345967664,0.4001572083672233,0.9787379841057392,2.240893
...
```

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

```
1
2
...
```

The submission format is explained in the section below.

FEATURES DESCRIPTION

You are required to use the following features (in the following order) to make your predictions:

- Linear

$$\phi_1(\mathbf{x}) = x_1,\ \phi_2(\mathbf{x}) = x_2,\ \phi_3(\mathbf{x}) = x_3,\ \phi_4(\mathbf{x}) = x_4,\ \phi_5(\mathbf{x}) = x_5,$$

- Quadratic

$$\phi_6(\mathbf{x}) = x_1^2,\ \phi_7(\mathbf{x}) = x_2^2,\ \phi_8(\mathbf{x}) = x_3^2,\ \phi_9(\mathbf{x}) = x_4^2,\ \phi_{10}(\mathbf{x}) = x_5^2,$$

- Exponential

$$\phi_{11}(\mathbf{x}) = e^{x_1},\ \phi_{12}(\mathbf{x}) = e^{x_2},\ \phi_{13}(\mathbf{x}) = e^{x_3},\ \phi_{14}(\mathbf{x}) = e^{x_4},\ \phi_{15}(\mathbf{x}) = e^{x_5}$$

- Cosine

$$\phi_{16}(\mathbf{x}) = \cos(x_1),\ \phi_{17}(\mathbf{x}) = \cos(x_2),\ \phi_{18}(\mathbf{x}) = \cos(x_3),\ \phi_{19}(\mathbf{x}) = \cos(x_4),\ \phi_{20}(\mathbf{x}) = \cos(x_5$$

- Constant

$$\phi_{21}(\mathbf{x}) = 1$$

where we indicate the whole input vector with **x** and we use x_i to denote its i^{th} component.

Your predictions are calculated as a linear function of the features above according to the following formula:

$$\hat{y} = w_1\phi_1(\mathbf{x}) + w_2\phi_2(\mathbf{x}) + \cdots w_{21}\phi_{21}(\mathbf{x})$$

SUBMISSION FORMAT

You are required to submit the weight of your linear predictor in a .csv file.

The file should contain 21 lines containing a float each. The i-th line indicates the i-th weight of your linear predictor:

```
1
2
...
```

The evaluation on this task depends on the accuracy of your predictor on a test data set that we do not release.

Notice that, to compute your prediction on the test data, the raw features are transformed according to the transformations introduced in the previous section. This means that the first entry of your weight vector is multiplied by $\phi_1(\mathbf{x})$, the second entry is multiplied by $\phi_2(\mathbf{x})$ and so on. As a consequence, it is important to submit the weight vector in the **correct order**.

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

EVALUATION

The evaluation metric for this task is the **Root Mean Squared Error** which is the square root of the mean/average of the square of all of the error.

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

How to compute it in Python:

```
from sklearn.metrics import mean_squared_error
RMSE = mean_squared_error(y, y_pred)**0.5
```

GRADING

We provide you with **one test set** for which you have to compute predictions. We have partitioned this test set into two parts and use it to compute a *public* and a *private* score for each submission. You only receive feedback about your performance on the public part in the form of the public score, while the private leaderboard remains secret. The purpose of this

division is to prevent overfitting to the public score. Your model should generalize well to the private part of the test set.

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 three baselines (easy, medium and hard). Your final grade depends on the public score and the private score (weighted equally), on your submitted code and on a properly-written description of your approach. The following **non-binding** guidance provides you with an idea on what is expected to obtain a certain grade: If you hand in a properly-written description, your source code is runnable and reproduces your predictions, and your submission performs better than the easy baseline, you may expect a grade exceeding a 4. If it further beats the medium baseline, you may expect that the grade will exceed a 5. If in addition your submission performs equal to or better than the hard baseline, you may expect a 6. If you do not hand in a properly-written description of your approach, you may obtain zero points regardless of how well your submission performs.

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

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.

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).

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. For example, you could read in the dataset as:

```
import pandas as pd
df_train = pd.read_csv('train.csv')
```

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 PIAZZA AS SOON AS HAVE A QUESTION?

This is highly discouraged. 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.

If you still consider that you should contact the TAs, you can post a **private** question on Piazza. Remember that collaboration with other teams is prohibited.

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.