Machine Learning
Programming Assignment I

Ujjwal Sharma, Athanasios Efthymiou and dr. Stevan Rudinac

The following assignment will test your understanding of topics covered in the first two weeks of the course. This assignment will count towards your grade and should be submitted through Canvas by **17.11.2022** at **08:59 (CET)**. You must submit this assignment in teams of 3 or 4. You can get at most 10 points for this assignment, which is 10% of your final grade.

Instructions

- Alongside the code for your experiments, you are also required to present a report summarizing the observations and results of each of the experiments. You can use text and graphs/plots (`matplotlib`) for these reports. You should place these report blocks within the Jupyter Notebook in separate text cells. Plots can be appropriately placed near the text explanation. Your final submission should be a single Jupyter Notebook with code and report blocks.
- While it is perfectly acceptable to brainstorm and discuss solutions with other colleagues, please do not copy code. We will check all submissions for code similarity with each other and with openly-available solutions on the web. Submissions with high similarity will be summarily rejected and no points will be awarded.
- Please ensure that all code blocks are functional before you finalize your submission. Points will NOT be awarded for exercises where code blocks are non-functional.

Submission

You can submit your solutions within a Jupyter Notebook (*.ipynb). To test the code we will use Anaconda Python (3.9). Please state the names and student ids of the authors at the top of the submitted file.

1 Data

In the `data.zip` file, you will find two accompanying files:

- A data file named `data.csv`. This is the dataset for Programming Assignment 1.
- A companion document `desc.txt` with a brief explanation of the features present in the data. Please read this document carefully before starting this assignment.

The accompanying data file contains data on the number of bikes rented each hour from the Seoul Bike-Sharing system [Sathishkumar et al., 2020, VE and Cho, 2020]. This data consists of 9 *features* and a single continuous-valued label `"Rented Bike Count"` that contains the count of bikes rented each hour. All other columns (except the date) are independent variables/features. You will find the `pandas` library extremely helpful for working with this data. For each experiment, you are also required to split the data into train, validation (if needed) and test splits. Choose an appropriate split ratio.

✉ u.sharma@uva.nl, a.efthymiou@uva.nl, s.rudinac@uva.nl

Before you start fitting models on this data, please analyze the data using Pandas. `DataFrame` methods like `info` and `describe` can provide helpful summaries on the structure and statistics of the data. As a preprocessing step, you are asked to analyze the data and perform the following operations:

1. Convert all features to an appropriate data type. Please read the description and ascertain what these data types should be based on the nature of the information contained in them.
2. Use appropriate plots to demonstrate the distribution of the 9 features and the target variable.

2 Regression

In this week, you've been introduced to the *Regression* task which models relationships between a continuous *dependent* variable and multiple *independent* variables. In this assignment, you will use regression models to predict the number of bikes rented each hour (`Rented Bike Count`). Each of your models will use 9 features or independent variables to predict the target variable.

2.1 Model Fitting

For this assignment, you will implement the following regressors:

1. An Ordinary Least-Squares linear regression model.
2. A Ridge regression model that adds L2 regularization to the Ordinary Least-Squares model.
3. A Lasso regression model that adds L1 regularization to the Ordinary Least-Squares model.

You are asked to perform the following tasks:

1. Fit the *Ordinary Least-Squares* model to the data. Once completed, report the *Mean Squared Error*, *Mean Absolute Error* and the R^2 coefficient of determination.
2. Fit the *Ridge* and *Lasso* regression models to the data. To find an optimal value for the `alpha` hyperparameter, use the scikit-learn grid search functionality in `sklearn.model_selection.GridSearchCV`. Only the training (and validation; if needed) set should be used for the grid search. You will need to compute the optimal `alpha` separately for both models. Report the best `alpha` value from your search. Please provide learning curves and plots to illustrate the effect of the choice of `alpha` on model performance.

Tip:

1. All models required for this assignment can be found in `sklearn.linear_model`.
2. For grid search over hyperparameters, you are advised to consult the `sklearn` documentation to check the default value for that hyperparameter and devise a suitable search strategy.

Grading

Experiment	Points
OLS Regression	2
Lasso Regression	2
Ridge Regression	2
Grid Search and Cross Validation	2
Report and Code Quality	2

References

[Sathishkumar et al., 2020] Sathishkumar, V., Park, J., and Cho, Y. (2020). Using data mining techniques for bike sharing demand prediction in metropolitan city. *Computer Communications*, 153:353–366.

[VE and Cho, 2020] VE, S. and Cho, Y. (2020). A rule-based model for seoul bike sharing demand prediction using weather data. *European Journal of Remote Sensing*, pages 1–18.