Norwegian University of Science and Technology

TMR4345 — **Project Assignment - ML & AI**

MARINE COMPUTER SCIENCE LAB (MARIN DATALAB)

Prateek Gupta

Spring 2023

1 Introduction

With the advancement of sensor technologies and big data handling tools, the usage of data-driven methodologies for large businesses as well as individual needs is becoming very common. Such data-driven methodologies are generally based on some machine learning (ML) and/or artificial intelligence (AI) techniques. The current project is designed so that the participants would become familiar with some of these methods. The aim of the project is not only to help the participants get started with data-driven methods, but also to make them understand the basic assumptions taken before such methodologies are employed, so that they would use such methods responsibly in future.

The project comprises of several tasks, which are listed in the following section. Although it is recommended to use python and Jupyter notebooks[1] here, the participants have the freedom to choose any programming language or framework to produce the results required by each task. The final report can be submitted in the form of a single or few jupyter notebooks or a PDF file with several code files. In the latter case, the PDF file should contain the information regarding which code file is concerned with each corresponding task. The details regarding the evaluation, submission and deadline are provided in sections 3, 4 and 5, respectively.

2 Tasks

The project is divided into several tasks, listed in the following subsections. All tasks are mandatory for all the participants. The distribution of score (or grade) is as per the percentage values provided in the task titles, which adds up to a 100%.

[1] `https://jupyter.org/`

2.1 Task 1: Data Exploration & Visualization (20%)

The ship data provided for the project should be explored and visualized to understand the data variables. Perform following tasks for each data variable:

a) Present the statistical properties, like mean, standard deviation, minimum, maximum, etc. of each data variable.

b) Create a histogram for each data variable, marked with its mean, median and mode. Is the data symmetrically distributed?

c) Does any of the data variables contain any abnormal, invalid or unrealistic values? If yes, then remove them from the data before carrying-out the next task (2.2).

d) Plot the time-series for each data variable and mark the erroneous samples identified in the above step. Explain briefly why these samples are erroneous.

2.2 Task 2: Data Analysis & Correlation Study (30%)

The data must be analyzed to make sure that it is good enough to be used for the data-driven methods, which are to be implemented in the following task. Following analysis must be performed here:

a) Calculate Pearson's correlation coefficient for each combination of data variables. Present the correlation coefficient values as a heat map[2] (with annotations or coefficient values).

b) Create cross-plots, i.e., plots with different variables on x and y axes, for each combination of data variables. What type of correlation exists between each pair of data variables? Answer with following keywords: strong or weak; positive or negative; linear or non-linear.

c) Divide the data variables into dependent and independent. Write the functional relationship[3] between the dependent and independent variables based on the results from the correlation study as well as your domain knowledge.

d) Is the data good enough for regression analysis? Answer briefly.

e) Can you suggest and, if possible, derive some additional variables which can help improve the results from regression modeling? Feel free to use your domain knowledge as well as the above results to corroborate your suggestions.

[2]See several examples here.

[3]In the form of $y = f(x_1,x_2)$. Do not provide any detailed equations.

2.3 Task 3: Regression Modeling (30%)

Provided the data is good enough, it is possible to establish regression models to depict the functional relationships presented in the above task. Select the most appropriate functional relationship, using as many data variables as possible, from the above task, and carry-out the following tasks. The participants are allowed to choose any set of regression algorithms or methods for the following tasks, but the selected algorithms must provide presentable results.

a) Divide the data into training, validation and test set. The proportion of data to be used for validation and testing as well as the data division scheme (random or continuous sections) has to be decided by the participant, and a brief justification should be provided to support the adopted scheme.

b) Standardize the data using the mean and standard deviation of the training data.

c) Create a linear regression model based on the selected functional relationship. Train the model using the data, and calculate its predictive performance based on goodness of fit parameters, including but may not be limited to root mean square error (RMSE), coefficient of determination (also denoted as R-squared or R2 score) and mean absolute error (MAE). Also, create the following plots using the calibrated model:

   - Original vs predicted values.
   - Time-series of original and predicted values in the same plot (with different markers).
   - Standardized[4] regressions coefficient.

d) Create a non-linear regression model to perform the above task. Calculate the goodness of fit and create the plots using the new model. The hyperparameters of the model (if any) should either be optimized or set to appropriate values such that the results are presentable.

e) Compare the results from the linear and non-linear regression models. Is the linear approach good enough to perform the task? Is it possible to further improve the results from the linear model? Write a short note.

f) Plot the regression residuals against all the data variables (in separate plots or subplots) used to create the above models. Do you see any pattern in the residuals with respect to any data variable? What does these patterns (if any) signify? Answer briefly.

[4]Scaled by the standard deviation of the corresponding data variable.

2.4 Task 4: Text Recognition (20%)

Text recognition using ML can be done using a model trained on one of the publicly available image databases. For the current task, MNIST database[5] is used to train a model to recognize handwritten digits (0 to 9). The database contains 60,000 and 10,000 images of handwritten digits for training and testing, respectively. Perform the following sub-tasks using the MNIST database:

a) Download the database, and visualize some sample images along with their digit labels. Also, present the table of pixel values for at least 1 sample image. What do these pixel values represent?

b) Calibrate a ML model on the training set, and calculate its accuracy on both the training and test sets.

c) Present a few cases (at least 1) where the model fails to predict the correct digit along with some case where the model prediction is correct.

3 Evaluation

The score (or part of the grade) corresponding to each task is provided by the percentage values in each task title. The final submission will be evaluated based on the following criteria (presented in the diminishing order of importance):

- Correctness of result values, code implementation and answers to theoretical questions.
- Presentation of plots, i.e., providing proper x and y labels, titles, legends, captions, etc. is recommended.
- Organization of code and report. It is important to properly mark or mention which part of code (or code file) and report is presented in response to which task.
- Short and concise language. Use of unnecessary verbose language may be penalized.
- Organization of plots. It is recommended to combine several plots as subplots in the same plot window.

4 Submission Requirements

The following is expected to be submitted by each participant:

[5] http://yann.lecun.com/exdb/mnist/

- **Code** with proper comments and implementation in any appropriate programming language. The code should be organized into modules and/or functions, clearly separating the code related to each task. To perform repetitive tasks, it is highly recommended to use `for` loops instead to copy-pasting the code several times, and reusing the same modules and/or functions at different places would also be appreciated.
- **Report** in PDF format including the answers, explanations, plots, results, etc. for each task. It is possible to submit the report with code in one or few Jupyter notebooks[6]. In this case, a separate PDF report file is not required to be submitted.

5 Deadline

All submissions are to be uploaded on the assignment submission link in the blackboard page of the course on or before the deadline, i.e., 1^{st} May 2023 mid night (23:59 hrs).

[6] `https://jupyter.org/`