COMP5318 - Machine Learning and Data Mining

Assignment 2

Due: 8 November 2019 11:00PM

This assignment is to be completed in groups of 3 students. It is worth 15% of your total mark. Your groups can be different from Assignment 1 and you have to **register your groups in Canvas**.

1. Objective

The objective of this assignment is to apply machine learning and data mining methods to solve a real problem. You should compare at least three techniques with at least one, not taught in this course (e.g. AdaBoost, Random forest, XGBoost, ADTree, etc.).

2. Instructions

2.1 Datasets

In this assignment, you can choose one of the following datasets:

- CIFAR-100, classification, https://www.cs.toronto.edu/~kriz/cifar.html
- The Chars74K, classification, http://www.ee.surrey.ac.uk/CVSSP/demos/chars74k/
- Chess Positions, classification, https://www.kaggle.com/koryakinp/chess-positions
- Forest Fires, regression, https://archive.ics.uci.edu/ml/datasets/Forest+Fires
- New York Stock Exchange, regression, https://www.kaggle.com/dgawlik/nyse
- Incident management process enriched event log, regression, https://archive.ics.uci.edu/ml/datasets/Incident+management+process+enriched+event+log

Note that if the datasets are too big to run, you can consider doing some pre-processing of the datasets or use part of them to train. However, they should be clearly explained in your report.

2.2 Assignment tasks

a) Choose a dataset from the list above.

b) Try different Machine Learning methods (at least 3) and compare their performance. At least one of the techniques you use should **NOT** have already been covered in the course material. You should experiment and clearly discuss your design decision to help you achieve a higher performance and speed. The design options should consider the following aspects:

- Choosing an appropriate model and its complexity

- Using pre-processing techniques on the datasets (e.g. clustering, feature extraction, etc.)
- Computer infrastructure (e.g. parallelizing, speeding-up your code, etc.)
- Ease of prototyping (e.g. implementation approach, choice of algorithms and libraries)

c) You are expected to fine tune each algorithm and explain why one approach outperforms the others.

d) Since you are expected to use more complex models that have not been discussed in the lectures, you can use most external open source libraries such as: scikit-learn, pandas, Keras, Tensorflow, PyTorch, Theano, Caffe2, or their equivalent in Python 3 to write your own classifiers. Should you require to use any other external libraries, please post on Piazza for confirmation.

e) **You are only allowed to use Python 3 on Jupyter Notebook in this assignment.**

3. Report

The report must be organised in a similar way to research papers, and include the following:

- In the **abstract**, succinctly describe the rest of your report.
- The **introduction** section should present the dataset that you chose, discuss its relevance in diverse applications, and give an overview of the methods you used.
- You are expected to include a section on the **previous work**, explaining successful techniques utilised on the same or similar datasets and how they are different to yours.
- The next section should discuss the **methods** you have adopted. Explain the theory behind each of them and discuss your design choices. This part should at least include pre-processing approaches and machine learning techniques used.
- The **experiment** section displays results and comparisons for the implemented algorithms. Include runtime, hardware and software specifications of the computer that you used for performance evaluations. You are then expected to include meaningful comments on the results of your experiments, and reflect on your design choices.
- In **conclusion**, sum up your results and provide suggestion for meaningful future work.
- The **references** section includes all references cited in your report, formatted in a consistent way.

3.1 Evaluation metrics

You should compare the algorithms with a 10-fold cross validation exercise.

Classification task: When evaluating different classifiers, include accuracy, precision, recall and confusion matrix.

Regression task: For regression problems, include Mean Square Error (MSE) and Negative Log Likelihood for the predictions (NLL):

$$N\,N\,L = -\log p(y_*|D, x_*) \; = \frac{1}{2}\log(2\pi\sigma_*^2) + \frac{\left(y_* - \bar{f}(x_*)\right)^2}{2\sigma_*^2}$$

where y_* is the actual value to be predicted, D is the training dataset, x_* is a query point, and $\bar{f}(x_*)$ and σ_*^2 are the prediction mean and variance respectively.

3.2 Report layout

Please follow the format of the MS-Word report template provided.

Length: Ideally 10 to 15 pages up to a maximum of 25 pages with [-10] penalty for each additional page after 25.

4. Submission

The report and code are due for submission by **8 November 2019, 11:00 PM**.

4.1 Proceed to Canvas and upload all files separately, as follows:

a) Report (a PDF file)

The report should include your group ID and each member's details (student ID and name).

You must include an appendix that provides detailed steps on how to successfully run your code, including any external libraries installation required to be able to execute your code.

b) Code (.ipynb files)

Your code should be written as one or more **.ipynb files**. You should separate the code file containing the algorithm and parameters that yield the best result from all the other algorithms, so in this case there would be 2 code files to submit.

Another alternative is to have one code file for each method / algorithm, i.e. 3 code files for 3 algorithms, 1 file for each one.

Note: Do **NOT** submit the dataset. In case your model takes significant time to train, you should submit the trained model as well for evaluation.

c) Code (PDF files of .ipynb code)

Every .ipynb code file must be saved as a PDF document and included in your submission e.g. if there are 2 .ipynb code files, you should also submit 2 PDF documents, one for each corresponding .ipynb file.

4.2 Only one student in your group needs to submit all the files and they must be named using your group ID separated by underscores e.g.

- group1_report.pdf
- group1_best_algorithm1.ipynb
- group1_other_algorithms.ipynb

- group1_best_algorithm1.pdf
- group1_other_algorithms.pdf

4.3 Your submission should include report and all the code files. A plagiarism checker will be used.

4.4 Clearly provide instructions on how to run your code in the appendix of the report.

4.5 Provide hyperlinks of the datasets you used, any external open-source libraries you used for the experiments and analysis, and versions of the libraries e.g. PyTorch 1.2.

4.6 Indicate the contribution of each group member. The contribution will be taken into consideration for adjusting the mark of each member accordingly.

4.7 A penalty of MINUS 20 (twenty) percent per each day after the due date. The maximum delay is 5 (five) days, after which the assignment submission will no longer be accepted.

4.8 The rubric is available in Canvas. Please review it carefully.

4.9 Remember, the due date to submit your deliverables in Canvas is **8 November 2019, 11:00PM**.