

LAB 2 – Supervised Learning

Requirements

- Python + Keras (Tensorflow recommended) + Jupyter Notebook

Methodology

1. Obtain a unique dataset from [1]. Should two or more students submit work using the same dataset, marking will be done as detailed in the 'Overlap' section below. A Google sheet is provided to ensure no overlap[2].
2. Load the dataset you've selected, performing any conversions which may be necessary. These (and following) steps should all be done in a Jupyter notebook, which will be part of your submission.
3. Define a training, testing, and validation data set.
4. Create and train a variety of supervised learning techniques. Parameters should be chosen well.
5. Observe any difference in performance and/or training speed. You will need to tabulate these results.

Report

- Your report should present your results and your analysis of those results.
- You should also justify your choice of parameters.
- There is no page limit, but be sensible. Aim for 4-6 pages (less is better than more).
- Be sure to state clearly which dataset you selected.

Submission

- Your submission comprises of ONE (1) Jupyter notebook and ONE (1) PDF document (Word files not accepted).
- Submission deadline is exactly 7 days after your lab finishes (5.30pm on the next Friday after your lab). Submission is through a Google Form[3].

Overlap

- If N students have overlap, the highest mark (X) between the N reports will be chosen then divided by N. Each of the N students will receive X/N marks.

Selecting a Data Set

- Make sure to select a data set with significant (≥ 100) instances, non-trivial (≥ 3) number of attributes, and for classification task. Some data sets may prove harder than others, so feel free to change to another option if this is the case.

[1] – UCI Machine Learning Repository: Data Sets <https://archive.ics.uci.edu/ml/datasets.php>

[2] – <https://tinyurl.com/AI2019SupervisedCoordination>

[3] – <https://tinyurl.com/AI2019SupervisedSubmission>

LAB 2 – Supervised Learning Marking Rubric

Graded Components Weightage

- Presentation and Formatting (15%)
- Results (15%)
- Analysis and Justification (40%)
- Code Quality (30%)

Presentation and Formatting Rubrics

- 0 Unreadable report.
- 1 Difficult to read, with obvious errors in formatting, grammar etc.
- 2 Acceptable, with some errors in formatting, grammar etc.
- 3 Good readability, appropriate use of graphics/tables. Minimal grammatical and formatting errors.
- 4 Outstanding presentation and formatting, no errors at all.

Results Rubrics

- 0 Not reported.
- 1 Inaccurate or incomplete results.
- 2 Basic results reported.
- 3 Results reported well, with thought given to organizing and summarizing data appropriately.
- 4 Reporting of results is impeccable, summary is easily viewable at a glance.

Analysis and Justification Rubrics

- 0 Not provided.
- 1 Perfunctory analysis and/or justification, off-topic or nonsensical.
- 2 Brief (but correct) analysis or justification provided.
- 3 Good analysis and justification which clearly provides rationale/reasoning.
- 4 Very good analysis and justification which convinces the reader.

Code Quality Rubrics

- 0 Not submitted.
- 1 Very poor code (no cells, hard to read etc.) or provided code does not work.
- 2 Working code.
- 3 Code is well organised and commented.
- 4 Code is easy to read because it is very well organised, showing proper planning.

Tabulation of Marks

Each graded component receives a mark based on the above rubrics. This assigned mark N is then divided by the maximum mark for the rubric M and multiplied by the weightage W. So the sum of your report marks S will be:-

$$S = \sum_{i=1}^n \frac{N_i}{M_i} \times W_i$$