U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO	L.EIC – BSc/Licenciatura Degree in Informatics and Computing Engineering **Artificial Intelligence**	2021/2022 (3rd Year) 2nd Semester
TEACHERS: Luís Paulo Reis, Henrique Lopes Cardoso, Ana Paula Rocha, Nuno Guimarães		

Assignment No. 2

Supervised Learning

Reinforcement Learning

Natural Language Processing

Theme

IART's second practical assignment consists in the application of machine learning models and algorithms related to one of three possible topics.

Topic 1: Supervised Learning

For supervised learning problems, the idea is to learn how to classify examples in terms of the concept under analysis. An initial exploratory data analysis should be carried out (class distribution, values per attribute, and so on). Different learning algorithms should be employed and compared using appropriate evaluation metrics (performance during learning, confusion matrix, precision, recall, accuracy, F1 measure) and the time spent to train/test the models.

Supervised learning includes the following steps: dataset analysis to check for the need for data pre-processing, identification of the target concept, definition of the training and test sets, selection and parameterization of the learning algorithms to employ, and evaluation of the learning process (in particular on the test set). At least 3 supervised learning (classification) algorithms should be employed (Decision Trees, Neural Networks, K-NN, SVM, …) but more may be employed and compared using the Scikit-Learn Python library and considering the characteristics of the dataset. Results should be compared using tables or plots (e.g., using Seaborn or Matplotlib libraries).

Students may also find similar datasets and apply the same approach to them or improve the dataset with information gathered by other means comparing the results achieved in all the applications.

Topic 2: Reinforcement Learning

For Reinforcement Learning problems, the puzzles/games used in the first assignment are to be reused. Students should start working with reduced versions of the puzzles/games and solving simple learning problems and then build progressively on their achievements in terms of learning agents capable of solving the simplified puzzle or playing the simplified game. All puzzles and games should be simplified in order to be possible to solve them with Reinforcement Learning. Students should start with a very basic task in a very small board (like, for example, learning to move a piece to the opposite side of a 4x4 board) and experiment solving it with several algorithms before analysing more complex tasks and environments.

At least two reinforcement learning algorithms should be used (Q-Learning, SARSA, DQN, Proximal Policy Optimization – PPO, Soft Actor-Critic – SAC, etc.) and their different parameterizations. The learning process should be evaluated throughout time and visualized in appropriate plots, showing the reward obtained throughout the time. Students may use directly the RL algorithms using libraries or implementing the source code, may model the puzzle or game as an environment in Open AI Gym and/or use ML-Agents for Unity.

Topic 3: Natural Language Processing

For NLP problems, the aim is to process textual data, employing diverse techniques to transform them into appropriate datasets that can then be addressed using supervised learning algorithms. An initial exploratory data analysis should be carried out (class distribution, word distribution per class based on TF-IDF, and so on). Different pre-processing and feature engineering techniques should be tried out. The employed machine learning algorithms should be tested and compared (performance on the test set, confusion matrix, precision, recall, accuracy, F1 measure, time spent to train/test the models).

At least 3 machine learning algorithms should be employed (Naïve Bayes, Decision Trees, Neural Networks, K-NN, SVM, …), matching them with the different ways of generating the dataset.

Programming Language/Libraries

Any programming language and development system can be used, including, at the language level, Python, C++, Java, C#, among others. However, it is strongly advised to use the Python language due to the availability of very strong machine learning libraries for this language. Although you may use any library or tool specific for developing supervised machine learning models (after validating it with the course teachers), it is highly advisable that the libraries used are the ones lectured on the course such as pandas, NumPy/SciPy, Scikit-learn and Matplotlib/Seaborn. You may use reinforcement learning systems/libraries such as Open AI Gym or ML-Agents. To use a different system/library you should validate it with the course teachers. You may use any library appropriate for Natural Language Processing (NLP). However, if the libraries used are not the ones suggested in the course (NLTK, Stanza and scikit-learn), you should validate it with the course teachers. The suggested tools or systems are the following: (i) IPython, Jupyter Notebook and scikit-learn for supervised learning; (ii) Open AI Gym for reinforcement learning; (iii) IPython, Jupyter Notebook, NLTK, Stanza and scikit-learn for NLP.

Groups

Groups must be composed of 3 students (exceptionally 2). Individual groups or groups composed of 4 students are not accepted. Groups can be composed of students attending the same practical class (although exceptions are possible). All students must be present in the checkpoint sessions and presentation/demonstration of the work. Groups composed of students from different classes are discouraged, given the logistic difficulties of performing work that this can cause.

Checkpoint

Each group must submit in Moodle a brief presentation (max. 5 slides), in PDF format, which will be used in the class to analyse, together with the teacher, the progress of the work. The presentation should contain: (1) specification of the work to be performed (definition of the machine learning problem to address); (2) related work with references to works found in a bibliographic search (articles, web pages and/or source code); (3) description of the tools and algorithms to use in the assignment; and (4) implementation work already carried out.

Final Delivery

Each group must submit in Moodle two files: a presentation (max. 10 slides), in PDF format, and the implemented code, properly commented, including a "readme" file with instructions on how to compile, run and use the program. The code and comments may be submitted as a complete Jupyter Notebook. Based on the submitted presentation, students must carry out a demonstration (about 10 minutes) of the work, in the practical class, or in another period to be designated by the teachers of the course.

The file with the final presentation should include, in addition to the aforementioned for the checkpoint, details on data pre-processing, the developed models and their evaluation and comparison, using appropriate graphical elements (tables, plots, etc.).

Suggested Problems

Topic 1: Supervised Learning

1A) Credit Card Fraud

1B) Top Hits Spotify from 2000-2019

1C) Sample Telco Customer Churn Dataset

1D) Students' dropout and academic success

Topic 2: Reinforcement Learning

2.1A) Unequal Length Mazes – Simple Puzzles

2.1B) Chess Snake Puzzles – Simple Puzzles

2.1C) Exactly 1 Mazes – Simple Puzzles

2.1D) Robot Mazes – Simple Puzzles

2.2B) Gekitai – Small Board, Simpler Games

2.2D) Splinter – Small Board, Simpler Games

2.2E) Three Dragons – Small Board, Simpler Games

2.2F) Lines of Action – Small Board, Simpler Games

Topic 3: Natural Language Processing

3A) Emotions

3B) Phishing Website HTML Classification

3C) Yelp Reviews for SA fine-grained 5 classes CSV

3D) Amazon Reviews for SA fine-grained 5 clases CSV