U.PORTO FACULDADE DE CIÊNCIAS UNIVERSIDADE DO PORTO U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO	**First Degree in Artificial Intelligence and Data Science** **Elements of Artificial Intelligence and Data Science**	**2022/2023** **1st Year** **2nd Semester**
TEACHERS: Luís Paulo Reis, Pedro Ferreira, David Aparício		

Assignment No. 2

Data exploration and enrichment for supervised classification

Theme

The second practical work consists in exploratory data analysis and the application of supervised learning models for classification. Optionally, the project may also consist of data collection and preparation.

Students may choose to use:

- A video games dataset (~6000 videogames). This dataset requires no data collection so the focus will be on data exploration, feature engineering, and classification.
- A media dataset, containing movies, music albums, and tv shows. This dataset is not given in tabular format and thus needs to be pre-processed (and optional data collection). Thus, data cleaning is particularly valued, as well as data exploration, feature engineering, and classification.

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 including class distribution, values per attribute, feature pre-processing (imputation of missing values, scaling, etc.), feature engineering (e.g building new features or removing redundant features) and other tasks considered relevant. Different learning algorithms should be employed and compared using appropriate evaluation metrics (performance during learning, confusion matrix, precision, recall, accuracy). Optional: Depending on the data set, you might first need to perform data collection and format your data in the form of a table.

Be aware that your dataset may contain redundant dependent and/or dependent variables. You may need to perform a critical assessment during the feature selection/engineering phase.

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).

Two supervised learning algorithms should be employed: Decision Trees and K-NN 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).

Programming Language/Libraries

The programs should be developed using Python language due to the availability of very strong machine learning libraries for this language. 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. The final result should be i) a python script to be run in the command line or ii) a jupyter notebook.

Groups

Groups must be composed of 2 students. Groups should be composed of students from the same practical class. All students should be present in the checkpoint sessions and presentation/demonstration of the work. The establishment of groups composed of students from different classes is not advised, given the logistic difficulties of performing work that this can cause and is only accepted in exceptional conditions.

Checkpoint

Each group must submit in Moodle a brief presentation (max. 5 slides), in PDF format, which will be used in the class to analyze, 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 or a python script. 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 preprocessing, the developed models and their evaluation and comparison, using appropriate graphical elements (tables, plots, etc.).

Annex - Datasets Description

Video games

1. This dataset is mostly curated and formatted already.
2. Main challenges:
    a. Explore features.
    b. Create new features.
    c. Train a supervised classifier.

Classification task: Predict if a video game has good or bad user reviews. The dataset consists of ~6000 videogames. The goal is to predict if the average users gave the video game a bad, mediocre, good, or great score. The dataset contains the following features:

- Identifier: **Id**
- Categorical: **Name**
- Categorical: **Category** (e.g., main game, expansion)
- Numerical: **Number of DLCs**
- Numerical: **Number of expansions**
- Numerical: **Release year**
- Numerical: **Follows** (number of people following a game on the IGDB website)
- Boolean: **In a franchise** (e.g., "Star Wars Racer" → True since it belongs to the Star Wars franchise)
- String: **Genre** (e.g., "Action, Sport")
- String: **Platform** (e.g., "Xbox, PC")
- String: **Companies** (e.g., "Electronic Arts, EA Canada)

- Numerical: **Average users score** (0 to 100)
- Categorical: **Average users rating** (bad, mediocre, good or great). Each class represents ~25% of the data.
- Numerical: **Number of reviews by users**
- String: **Summary**

Media

1. Dataset needs some data pre-processing (and, optionally, data collection).
2. Main challenges:
    a. (Data collection)
    b. Data pre-processing.
    c. Explore features.
    d. Create new features.
    e. Train a supervised classifier (lower focus than for the video games dataset).

Classification task: Predict if the movie/album/tv-show has good or bad critic reviews. The goal is to predict if the critics gave the movie/album/tv/show a bad, mediocre, good, or great score. The dataset contains the following features:

Preparing the dataset (suggested)

1. Load `movies.json` into a pandas dataframe.
2. Adapt `extract_movies_info.py` to obtain information about tv shows and music albums, e.g., create a `extract_musicalbums_info.py` and a `extract_tvshows_info.py`. In `movies.txt`, `albums.txt`, and `tv-shows.txt` you already have a list of examples that you can use to query MetaCriticAPI using the scripts.
3. Merge the three datasets. Take into account that the datasets don't have the same information (add NAs, or try to find features that could be equivalent across datasets).
4. Adapt the problem to a classification task by converting the critic score into four classes: bad, mediocre, good, excellent.
5. Create new features, explore them.

Note that you need to create the appropriate query to extract data from the MetaCritic API. See an example below of a query to extract some information about a music album. Use the API playground to make sure that your queries are correct.

```
query {
  album(input: {
    artist: "Kendrick Lamar",
    album: "DAMN"
  }) {
    criticScore
    releaseDate
       genres
    numOfCriticReviews
  }
}
```

Feel free to (i) add more movies, tv shows, music albums, to (ii) focus only on the medium that you are more interested in, (iii) to collect other features from the internet (e.g., the screenwriter of the movie, if it was based on a book, etc).