


 FACULDADE DE CIÊNCIAS UNIVERSIDADE DO PORTO  FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO	First Degree in Artificial Intelligence and Data Science Elements of Artificial Intelligence and Data Science	2023/2024 1 st Year 2 nd Semester
TEACHERS: Miriam Santos, Pedro Ferreira, Luís Paulo Reis		

Assignment No. 2

Data exploration and enrichment for supervised classification

Theme

The second practical assignment consists in the development of a full data science pipeline, from exploratory data analysis and data preprocessing to the application of supervised learning techniques for classification and their respective performance evaluation. Optionally, the project may also consider the exploration of additional techniques such as clustering, missing or imbalanced data handling, among others, to improve the system's performance.

The Hepatocellular Carcinoma Dataset

In this project, the goal is to address a real data science use case from data cleaning and feature assessment to visual inspection and communication of results, using the Hepatocellular Carcinoma (HCC) dataset. The HCC dataset was collected at the Coimbra Hospital and University Center (CHUC) in Portugal and contains real clinical data of patients diagnosed with HCC. The main goal of this project is to develop a machine learning pipeline capable of determining the survivability of patients at 1 year after diagnosis (e.g., “lives” or “dies”). To address this project, students should focus on each step of a standard data science pipeline and explore suitable solutions to develop an efficient machine learning solution:

- **Data Exploration:** An initial exploratory data analysis should be carried out including examining feature types, number of features/records, class distribution, values per attribute, etc., and highlighting feature inconsistencies such as missing values, outliers, underrepresented concepts, irrelevant features, etc. The analysis can and should be supported with visualization techniques.
- **Data Preprocessing:** This refers to feature pre-processing (e.g., imputation of missing values, data transformation, data scaling, etc.) and feature engineering (e.g., building new features or removing redundant features) and other tasks considered relevant.
- **Data Modeling (Supervised Learning):** Supervised learning includes the 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). Decision Trees and KNN (e.g., using Scikit-learn) should be used to build classification models. Other classifiers are optional and considered as “extra elements”.
- **Data Evaluation:** Classification results should be compared across different evaluation metrics (performance during learning, confusion matrix, ROC/AUC, precision, recall, accuracy) using a standard train/test split. Results should be compared using tables and plots (e.g., using Seaborn or Matplotlib libraries). Other partitioning methods are considered “extra elements”.
- **Interpretation of Results:** This involves extracting meaningful insights from the obtained results: explain the behavior of the models, drawing conclusions about the effectiveness of the chosen algorithms and preprocessing techniques, providing recommendations for future analysis, investigating discrepancies of unexpected findings, etc.

Extra Elements:

The incorporation of elements beyond the core requirements of the project are given a bonus of 10%. These elements can either focus on technical implementation of data science software or refer to subsidiary tasks along the experimental setup. Some suggestions are as follows:

- Exploring several techniques for missing data imputation and their effect in classification performance (e.g., sensitivity/specificity results) and imputation quality (e.g., RMSE).
- In case of class imbalance, assessing the impact of data balancing techniques, either through data down-sampling of the most frequent class or through imbalanced data techniques (e.g., SMOTE, ADASYN) and compare their impact on the final performance results.
- Using additional partitioning methods (e.g., holdout, cross-validation, leave-one-out, etc.) to determine how they impact the classification results.
- Experimenting with additional algorithms and hyperparameters to optimize model performance.
- Deploying the solution to an external source, such as creating a Streamlit app to create an interactive web application for the project.

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 main 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 3 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 PDF with a brief presentation (**maximum 5 slides**) that 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 the following deliverables:

- **A PDF presentation (max. 10 slides)** with details on the final work: data preprocessing, the developed models and their evaluation and comparison, using appropriate graphical elements (tables, plots, etc.).
- **The implemented code**, properly commented, submitted as a complete Jupyter Notebook or as a python script.
- **A README.md** file with instructions on how to run and use the program, basic documentation and package dependencies.
- **A link to the [GitHub](#) repository** where the project should be pushed.

Based on the presentation, students must carry out a demonstration (**no more than 10 minutes**) of the work, in the practical class, or in another period to be designated by the teachers of the course. **The deadline for the submission is May 21, and the defenses of the project will take place on May 22 and 23.**

Evaluation

The project will be evaluated regarding the following expected outcomes:

- **Checkpoint (5%):** Student involvement, progress, plan, and management of the project's timelines;
- **Presentation and Demo Quality (10%):** Quality of presentation delivered by the students, domain over the projects goals, methodology, results, and conclusion;
- **Code Quality and Domain (10%):** Quality of the technical implementation and documentation;
- **Data Characterization (10%):** Understanding variable/feature types and overall dataset and feature characteristics;
- **Data Quality Assessment and Data Preprocessing (20%):** Exploring feature selection and engineering methods, identifying data quality issues in the data and applying suitable techniques to handle them effectively;
- **Data Visualization (15%):** Visually exploring the data and producing meaningful insights from the chosen visualizations (e.g., bar plots, histograms, heatmaps, correlation matrices, dendrograms, etc.)
- **Supervised Learning (20%):** Exploring Decision Trees and KNN results and interpreting performance metrics;
- **Critical Thinking and Communication of Results (10%):** Examining the information yielded by the data analysis, and sharing insights with stakeholders with different backgrounds and knowledge of the problem (e.g., data scientists, business analysts, and domain experts);
- **Extra Elements (10%):** Any creative methods that go beyond the scope of the project, either theoretical (e.g., exploring clustering solutions, other classifiers, distance metrics, missing data, imbalance data) or practical (e.g., developing a Streamlit application to showcase the project results).

Bibliography

1. EASL Clinical Practice Guidelines: Management of hepatocellular carcinoma. *Journal of Hepatology — The home of liver research*. <https://socgastro.org.br/novo/wp-content/uploads/2021/01/easl-easl-guidelines-management-of-hepatocellular-carcinoma.pdf>
2. Santos, M. et al. "A new cluster-based oversampling method for improving survival prediction of hepatocellular carcinoma patients." *Journal of biomedical informatics* 58 (2015): 49–59. <https://www.sciencedirect.com/science/article/pii/S1532046415002063>
3. Chicco, D. et al. "Computational intelligence identifies alkaline phosphatase (ALP), alpha-fetoprotein (AFP), and hemoglobin levels as most predictive survival factors for hepatocellular carcinoma." *Health Informatics Journal* 27.1 (2021). <https://journals.sagepub.com/doi/10.1177/1460458220984205>