U.PORTO FEUP FACULDADE DE ENGENHARIA UNIVERSIDADE DO PORTO	**L.EIC – BSc/First Degree in Informatics and Computing Engineering** **Artificial Intelligence**	**2024/2025** **(3rd Year)** **2nd Semester**

Assignment No. 2

Supervised Learning

Theme

The second practical assignment of IART focuses on the application of **Supervised Learning** techniques in the context of classification problems. The goal is to develop and evaluate machine learning models capable of learning from labeled data to make accurate predictions regarding a specific target variable (or concept).

The assignment should begin with a simple **Exploratory Data Analysis** to understand the dataset's characteristics, including class distribution, data types, missing values, and basic statistical summaries for each attribute. This analysis informs potential **data preprocessing steps**, such as handling missing data, encoding categorical variables, normalization, and feature selection.

Students must implement and compare **at least three supervised learning algorithms**, such as **Decision Trees, Neural Networks, k-Nearest Neighbors (k-NN), Support Vector Machines (SVM)**, or others.

Model performance should be assessed using standard metrics, including: **Accuracy, Precision, Recall, F1-core, Confusion Matrixes, Training and testing times.**

The complete **machine learning pipeline** should be followed: Data loading and preprocessing; Problem definition and target identification; Model selection and parameter tuning; Model training and testing; Evaluation and comparison of results

All experiments and results should be documented with the support of **tables and visualizations** (e.g., using Seaborn or Matplotlib) to clearly illustrate the comparative performance of the models.

Students may also develop other analysis such as **Learning Curves, ROC Curves** (Receiver Operating Characteristic Curve), **Training vs. Validation Error Plots** (Learning Curve), among several other analysis.

Python is strongly recommended as the programming language due to the availability of powerful libraries like **Pandas, NumPy, Scikit-learn**, and **Matplotlib/Seaborn**, which simplify and enhance the development of machine learning pipelines.

Programming Language/Libraries

Although students are free to choose any programming language or development environment—such as Python, C++, Java, or C#—**Python is strongly recommended** due to its extensive ecosystem of libraries and tools specifically designed for machine learning and data analysis.

To ensure consistency with the methodologies taught during the course and to simplify development, students are encouraged to use the following Python libraries:

- **Pandas** – for data manipulation and preprocessing
- **NumPy/SciPy** – for numerical computations
- **Scikit-learn** – for implementing machine learning models and evaluation metrics
- **Matplotlib/Seaborn** – for data visualization and presentation of results

Other tools or libraries may be used **with prior approval from the course instructors**, provided they are well justified in terms of functionality or performance. In all cases, the code must be **well-structured, clearly commented**, and **easily reproducible**.

Groups

Students must carry out this assignment in **groups of three**. In exceptional cases, groups of **two students** may be allowed, but this should be justified and approved by the instructors.

To ensure effective collaboration and facilitate logistical coordination, **each group should consist of students from the same practical class**. Groups composed of students from different classes are discouraged due to the additional difficulties in scheduling and monitoring progress.

All group members must be **present at the checkpoint session and the final presentation/demonstration**. Active and balanced participation from all members is expected throughout the development of the assignment.

Checkpoint

Each group must submit a brief **progress presentation** in PDF format (**maximum 5 slides + title slide**) via Moodle. This presentation will be discussed during class with the instructor to assess the current status of the project**, validate the approach and provide guidance** for its continuation.

The checkpoint presentation should include the following elements:

1. **Problem Definition** – Clear description of the supervised learning problem being addressed and the target variable(s).
2. **Related Work** – Summary of relevant studies, datasets, or codebases, with proper references (e.g., research articles, official documentation, GitHub repositories).
3. **Methodology** – Description of the tools, libraries, algorithms, and evaluation metrics selected for the assignment.
4. **Work Progress** – Brief report on the implementation carried out so far (e.g., data analysis, preprocessing, initial models).

The checkpoint is intended to **validate the approach** chosen by the group and ensure that the project is progressing in a well-structured and coherent manner. All team members must be present during the session to discuss the work with the instructor.

Final Delivery

Each group must submit **two files** via Moodle:

1. A **presentation** in PDF format (maximum **10 slides**) summarizing the work developed.
2. The **complete source code**, in a jupyter notebook format, including a well-documented **README** file with instructions on how to execute the program.

The final submission must include the following elements:

- A clear description of the **problem addressed**, including the definition of the target variable and dataset used.
- Details on **data preprocessing**, including any cleaning, encoding, normalization, or feature selection techniques applied.
- Description of the **machine learning models implemented**, including the algorithms chosen, their parameters, and justifications for their selection.
- **Evaluation and comparison of models** using appropriate metrics (e.g., accuracy, precision, recall, F1-score, confusion matrix) and training/testing time.
- **Visual elements** such as tables, plots, and confusion matrices to support the comparison and analysis of results (preferably using Matplotlib or Seaborn).

Students should submit their code as a **Jupyter Notebook**, which is recommended for its clarity and integration of code, visualizations, and explanations.

Based on the final presentation, each group will give a **10-minute live demonstration** of their project, either during the practical class or in a special session scheduled by the instructors. All group members must participate in the presentation.

Datasets

All problems should be treated as classification problems (defining appropriate classes) and not as regression problems:

A) Employee Promotion Dataset

B) All UFC Fight Outcome

C) Fraud Detection in E-Commerce Dataset

D) Exploring Mental Health Data

E) Binary Prediction of Poisonous Mushrooms

F) Binary Classification of Insurance Cross Selling

G) Classification with an Academic Success Dataset

H) Binary Prediction of Smoker Status using Bio-Signals

I) Multi-Class Prediction of Cirrhosis Outcomes

J) Binary Classification with a Bank Churn Dataset

K) Backpack Prediction Challenge

L) Loan Approval Prediction

M) Steel Plate Defect Prediction

N) Multi-Class Prediction of Obesity Risk

O) Predict Students' Dropout and Academic Success

P) Binary Classification with a Software Defects Dataset

Q) Diabetes Prediction Dataset