**Major Course Output 3** **Release Date:** October 27, 2023
**Comparison of Machine Learning Models for Classification** **Due Date:** November 29, 2023 (Wed), 11:59 PM

# 1 Project Description

In this project, students will apply the concepts and theories they have learned in the course to develop machine-learning based classifiers for a task of their choice. Through this project, students should be able to demonstrate the following learning outcomes:

- **LO4.** Collaboratively experiment how machine learning can be used to classify objects using real-world datasets.
- **LO5.** Articulate ideas and present results in correct technical written and oral English.

# 2 Project Specifications

This section contains the specifications for this major course output.

## 2.1 Overview

The goal of this project is for the students to compare the performance of two machine learning models for a classification task of their choice. You will start by picking a dataset and then preprocessing it. After that, you will identify at least one classification task which can be solved by two different machine learning models trained from that dataset. You will then analyze the results of the chosen models on the classification task, and compare and contrast their performance. In this project, getting a good performance with the trained model is not a requirement. However, you must demonstrate a deeper-than-surface-level understanding of how they work and why they behave that way in the context of the dataset that was used.

## 2.2 Dataset

For this project, you can pick any dataset from Kaggle. Kaggle is an online data platform containing several public datasets that can be used for machine learning. In choosing a dataset, consider the following:

1. Choose a dataset that your group finds interesting or relevant to a real-world problem or curiosity.
2. Do not choose toy datasets (datasets that are not based on the real-world and are just created to illustrate the performance of a certain machine learning model).
3. Choose a dataset that has at least 500 instances.
4. Choose a dataset that contains appropriate features and labels for a classification task.

Many datasets in Kaggle contain associated code written by other people. These codes often contain machine learning models trained on these datasets. You may use these as inspiration, but please do **not** copy these codes directly for the project. Although many datasets in Kaggle are designed to predict or classify a specific variable, please be reminded that you are free to deviate from this suggestion. Focus on something that truly interests you as a group.

## 2.3 Jupyter Notebook

For this project, the main deliverable is a Jupyter Notebook containing all codes and findings from the project. Jupyter Notebook is an interactive computational environment for creating notebook documents, allowing for the combination of Python code and desciptive cells containing explanations. You can follow the steps in this link to install Jupyter Notebook in your machine. Alternatively, you may also use Google Colabatory, an online version of the Notebook.

### 2.3.1 General Guidelines

Please observe the following guidelines when writing the Notebook:

1. The Notebook should run properly when the cells are run from top to bottom.
2. Do not include unnecessary and experimental codes in the final submission.
3. For every step performed, include markdown cells with a descriptive explanation.
4. The Notebook should be readable and easy to understand.

#### 2.3.2 Preprocessing

You must perform the necessary preprocessing steps on your chosen dataset, including but not limited to: handling of missing values, making sure that the values are in the correct format, and normalization of variables. Markdown cells should be included to describe the procedures that were applied and the rationale behind them.

#### 2.3.3 Definition of Classification Task

The Notebook must include a section where the classification task is formally defined. This section should clearly indicate the features and the label for the machine learning task. It should also explain the group's rationale for choosing this type of classification task and why is it interesting or relevant for them.

#### 2.3.4 Implementation and Evaluation of Classification Models

You must apply **at least two machine learning models** for the same classification task that you defined. Thereafter, you must make a comparative analysis of the performance of the models using the appropriate metrics and techniques. Please ensure that every step you perform is concisely documented within the Notebook, including the rationale for specific decisions made throughout the process.

To build, train, and run the models, you can use Scikit-learn, a free Python machine learning library that offers implementations of several machine learning models and algorithms. You can download scikit-learn here. The user guide can be found here, and the documentation can be found here. You are also allowed to use other external libraries for machine learning if you wish.

Libraries such as Scikit-learn abstract the low-level implementations of machine learning models. However, please be reminded that you are still expected to have an understand of how the models work underneath. While you can these libraries to implement the models, you are expected to make reasonable choices in preparing the training and testing framework, setting up the models, and the analysis of the results. For example, you need to justify how you set up your training and testing data, and how you interpreted the results.

## 3 Deliverables

There is only one deliverable for this major course output: a zip file containing the source files for the project. The source files should include:

1. The Jupyter Notebook (`.ipynb` file) containing the source files as well as the **documentation** about the project thorugh the included markdown cells
2. The dataset/s that was used for the project. If the dataset is too large to attach, provide a link to it in the Notebook instead.
3. A `contributions.txt` file containing the list of contributions for each group member.

The file name of the zip file should be: `mco3_<surname 1>_<surname 2>_<surname 3>_<surname 4>.zip` with the surnames in alphabetical order.

## 4 Academic Honesty

Honesty policy applies. Please take note that you are **NOT** allowed to borrow and/or copy-and-paste in full or in part any existing related program code from the internet or other sources (such as printed materials like books, or source codes by other people that are not online). **Violating this policy is a serious offense in De La Salle University and will result in a grade of 0.0 for the whole course**.

Please remember that the point of this project is for all members to learn something and increase their appreciation of the concepts covered in class. Each member is expected to be able to explain the different aspects of their submitted work, whether part of their contributions or not. **Failure to do this will be interpreted as a failure of the learning goals, and will result in a grade of 0 for that member**.