# Assignment 2: Logistic Regression with Bagging

Apply logistic regression to this [Bank Note Authentication](https://archive.ics.uci.edu/ml/datasets/banknote+authentication#) dataset. A **CSV** file is also included in the code base. Your goal is to predict whether a banknote is original.

* Class 0 (Negative class): Fake
* Class 1 (Positive class): Original

Please find the necessary boilerplate codes here > [Assignment 2 Code Base](https://drive.google.com/drive/folders/16UlUJBMk5a9onECFW2mK-d6-sIUozK47?usp=sharing)

There are missing parts in the code that you need to implement. You have two tasks.

## Task 1: Simple Logistic Regression

Refer to **run\_logistic\_regression.py**. That code should handle data reading, splitting, training, and performance reporting.

## Task 2: Logistic Regression with Bagging

Refer to **run\_logistic\_regression\_with\_bagging.py**. That code should handle data reading, splitting, training, and performance reporting. For ensembling, you need to train classifiers on different samples from the **training** dataset. The **BaggingClassifier** class should be able to generate samples and train separate classifiers on its own.

## Additional Information

* You **must** use NumPy and Pandas.
* Using Sklearn, Scikit, Keras, Tensorflow, PyTorch - or other frameworks that include these implementations is prohibited.
* Please refer to [Precision and recall - Wikipedia](https://en.wikipedia.org/wiki/Precision_and_recall#Definition_(classification_context)) for the performance metrics
* Write the code in such a way that you can quickly incorporate a new dataset (with a different number of samples and attributes)
* You don’t have to follow the design pattern by heart. You are allowed some creative independence.

Contact us if you find any discrepancies.

**Deadline 17 December (Saturday), 12:05 AM**