# Homework #4: KNN, Naïve Bayes, and Banknote Authentication

In this assignment, you have two tasks to complete using the *Banknote Authentication* dataset:

1) Use your ML / DM tool of choice (e.g., KNIME) to experiment with the **K-Nearest Neighbor** and **Naive Bayes** classifiers on the *training* dataset (i.e., banknote_train.csv). Submit a short description (with screenshots) explaining your experimentation. E.g.,
   - What algorithm parameters did you try?
   - Did you use **K-fold cross-validation**?
   - What were the results (accuracy, precision, and recall)?

2) Once you determine the "best" algorithm and parameters (e.g., K), implement the classification workflow in **Python** to classify the *test* dataset (i.e., banknote_test.csv). Submit your Python code.
   - You can *hard-code* input files "**banknote_train.csv**" and "**banknote_test.csv**"
   - You must write the *class predictions* to a file named **<lastname>.csv**
     i. Please include each *row* of test data with class label in your output

## EXAMPLES (task 2 *only*)

```
UNIX> ls *.csv
banknote_test.csv   banknote_train.csv

UNIX> head -n 1 banknote_test.csv
4.2586,11.2962,-4.0943,-4.3457

//use python script(s) to predict banknote test data
UNIX> python KNN_workflow.py

UNIX> python NB_workflow.py

UNIX> ls *.csv
banknote_test.csv   rubin_knn.csv
banknote_train.csv  rubin_nb.csv

//examine the first prediction made
UNIX> head -n 1 rubin_knn.csv
4.2586,11.2962,-4.0943,-4.3457,0.0000

UNIX> head -n 1 rubin_nb.csv
4.2586,11.2962,-4.0943,-4.3457,0.0000
```

**<u>HINTS</u>**

- Task #1
  - K-folds cross validation
  - accuracy, prediction, recall
  - KNIME specific:
    - CSV Reader
    - Number To String
    - X-partitioner
    - X-aggregator
    - Predictor
    - Learner
    - Scorer

- Task #2
  - Python **scikit-learn** module
  - numpy module
  - numpy genfromtxt( … )
  - numpy savetxt( … )