

2627 - Machine Learning Algorithms and Practice

Assignment 1

The goal in this assignment is to craft a classic machine learning solution for an object recognition task. Each object is a 28x28 pixel image. You will get these images as 'flattened' 784-dimensional vectors, each tagged with a label (+1 or -1).

Data Sources: The training images (`Xtrain.csv`), training labels (`Ytrain.csv`), and unlabelled test images (`Xtest.csv`) are available on the [module webpage](#). Load the files using `np.loadtxt`; they are whitespace-separated and contain no headers.

- Training Data (`Xtrain`): Rows of images for you to train your model.
- Training Labels (`Ytrain`): The label of each image.
- Test Data (`Xtest`): More rows of images for you to test your model.

Submission Guide: You must submit a single Jupyter (IPython) Notebook containing all code, figures, and written answers. Your notebook must include the following and is to be submitted using Tabula.:

1. Your name and student ID at the top. For all experiments involving randomness (e.g. data shuffling or cross-validation), use the numeric part of your student ID as the random seed (e.g. u1234567 to 1234567) and report results using this seed.
2. A declaration at the beginning stating whether you used AI tools (e.g. ChatGPT), and in at most two lines, the purpose for which they were used. The use of such tools is permitted provided it is declared and complies with Warwick's academic integrity principles. All submitted work must be your own; the use of unacknowledged external work (including AI-generated content) will be treated as a serious breach of academic integrity and will be severely penalised in accordance with University regulations. Submissions showing inconsistencies between code, results, and explanations, or raising concerns about authorship or understanding, may be selected for a short follow-up viva in which the student will be asked to explain and defend their work and final marking will be based on that.
3. All code, outputs, figures, and explanations required to answer the questions.
4. All cells executed in order, with outputs visible, so that results can be verified.
5. A clear summary table comparing the performance metrics of the models you evaluated.
6. Code restricted to the following libraries: numpy, pandas, scipy, sklearn. If additional libraries are used, installation commands (e.g. `!pip install ...`) must be included and justified.
7. Sufficient inline comments and explanations to make your reasoning clear.
8. In addition, you must submit a separate prediction file for the test data: A single-column CSV file containing the prediction score for each example in `Xtest`, in the original order. The file must be named using your student ID (e.g. u100011.csv).

Marking Criteria

- a. Correctness and completeness of implementation: 20-30%
- b. Reasoning, interpretation, and diagnostic analysis: 40-50%
- c. Falsification, robustness analysis, or insightful extensions: 20%

Question No. 1: (Exploring data) [10% Marks]

Start by loading the training and test data. Once you have it ready, let's explore with these questions:

i. Dataset Overview

- How many examples of each class are in the training set? And in the test set?
- Does this distribution of positive and negative examples signify any potential issues in terms of design of the machine learning solution and its evaluation? If so, please explain.

ii. Visual Data Exploration

- Pick 10 random objects from each class in the training data and display them using `plt.matshow`. Reshape the flattened 28x28 arrays for this. What patterns or characteristics do you notice?
- Do the same for 10 random objects from the test set. Are there any peculiarities in the data that might challenge your classifier's ability to generalize?

iii. Choosing the Right Metric

Which performance metric would be best for this task (accuracy, AUC-ROC, AUC-PR, F1, Matthews correlation coefficient, mean squared error etc.)? Define each metric and discuss your reasoning for this choice.

iv. Benchmarking a Random Classifier

Imagine a classifier that produces a random prediction score in the range $[-1, +1]$ for a given input example. What metrics (AUC-ROC, AUC-PR, F1, Matthews correlation coefficient, mean squared error etc.) would you expect it to achieve on both the training and test datasets? Show this through a coding experiment.

v. Benchmarking a "Positive" Classifier

Imagine a classifier that produces a positive label (+1) for any given input example. What metrics (AUC-ROC, AUC-PR, F1, Matthews correlation coefficient, mean squared error etc.) would you expect it to achieve on both the training and test datasets? Show this through a coding experiment.

Question No. 2: (Nearest Neighbor Classifier) [10% Marks]

Perform 5-fold stratified cross-validation (https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.StratifiedKFold.html) over the training dataset using a k-nearest neighbour (kNN) classifier and answer the following questions:

- Can two images that look very similar to a human be far apart under Euclidean distance? Construct or find an example.
- Start with a $k = 5$ nearest neighbour classifier. **Define and calculate** the accuracy, balanced accuracy, AUC-ROC, AUC-PR, F1 and Matthews Correlation Coefficient for each fold using this classifier? Show code to demonstrate the results. Calculate the average and standard deviation for each metric across all folds and show these in a single table. As the KNN classifier in sklearn does not support `decision_function`, be sure to understand and use the `predict_proba` function for AUC-ROC and AUC-PR calculations or plotting.
- Plot the ROC and PR curves for one fold. What are your observations about the ROC and PR curves? What part of the ROC curve is more important for this problem and why?
- At what value of k would kNN become equivalent to a trivial classifier? Why?

- v. Identify one training example that is consistently misclassified across folds. What does this tell you about the dataset rather than the model?

Question No. 3: [20% Marks] Cross-validation of SVM and RFs

Use 5-fold stratified cross-validation over training data to choose an optimal classifier between: SVMs (linear, polynomial kernels and Radial Basis Function Kernels) and Random Forest Classifiers. Be sure to tune the hyperparameters of each classifier type (C and kernel type and kernel hyper-parameters for SVMs, the number of trees, depth of trees etc. for the Random Forests etc). Report the cross validation results (mean and standard deviation of accuracy, balanced accuracy, AUC-ROC and AUC-PR across fold) of your best model. You may look into grid search as well as ways of pre-processing data (<https://scikit-learn.org/stable/modules/preprocessing.html>) (e.g., mean-standard deviation or standard scaling or min-max scaling).

- i. Write your strategy for selecting the optimal classifier. Show code to demonstrate the results for each classifier.
- ii. Show the comparison of these classifiers in a single consolidated table.
- iii. Plot the ROC curves of all classifiers on the same axes for easy comparison.
- iv. Plot the PR curves of all classifier on the same axes for comparison.
- v. Write your observations about the ROC and PR curves. Why might two classifiers have almost identical ROC curves but very different PR curves? If you were forced to deploy only one model without retraining, which curve would you trust most and why?

Question No. 4 [20% Marks] PCA

- i. Plot the scree graph of PCA and find the number of dimensions that explain 95% variance in the training set. Then reduce the number of dimensions of the training data using PCA to 2 and plot a scatter plot of the training data showing examples of each class in a different color. What are your observations about the data based on these (scree and scatter) plots?
- ii. Reduce the number of dimensions of the training and test data together using PCA to 2 and plot a scatter plot of the training and test data showing examples of each set in a different color (or marker style). What are your observations about the data based on this plot? What would it imply if test points project outside the convex hull of training points in PCA space?
- iii. Reduce the number of dimensions of the data using PCA and perform classification. You may want to select different principal components for the classification (not necessarily the first few). What is the (optimal) cross-validation performance of a Kernelized SVM classification with PCA? Remember to perform hyperparameter optimization!
- iv. Plot the first at least 10 PCA basis vectors as 28×28 images using plt.matshow. Which PCA components are easiest for a human to interpret visually? Are these the same components that best separate the classes? Why or why not?
- v. By applying controlled transformations to the training data and refitting PCA, identify which principal component basis vectors are most affected by (i) uniform brightness increase, (ii) addition of random noise, (iii) randomisation of labels, (iv) horizontal translation, and (v) rotation, and justify your conclusions using visual and quantitative evidence.

Question No. 5 Another classification problem [20% Marks]

- Define a binary classification task where each example is labelled by its origin: training set (−1) or test set (+1). Using 5-fold stratified cross-validation, train a classifier to solve this task and report the mean and standard deviation of the AUC-ROC.
- Interpret the resulting AUC-ROC value as a measure of dataset shift. What does a value close to 0.5, moderately above 0.5, or close to 1.0 imply about the relationship between the training and test sets?
- Identify which features or transformations of the data contribute most to separating training and test examples, and provide evidence to support your conclusion.
- Apply data augmentations (random noise and random rotations) to the training data and repeat the experiment. Analyse how and why the AUC-ROC changes, and what this reveals about the nature of the shift.
- Explain how the presence of such a train–test distinction would affect your confidence in the evaluation of classifiers in earlier questions, and describe at least one principled strategy to reduce or eliminate this issue.

Question No. 6 Optimal Pipeline [20% Marks]

Using evidence and insights from Questions 1-5, design a complete end-to-end classification pipeline for this task. Your pipeline may include any preprocessing, representation learning, model selection, calibration, and post-processing steps you consider appropriate, but must use only the provided data.

You must:

- Clearly describe the structure of your pipeline and justify each design choice in terms of the empirical findings from earlier questions, not generic best practices.
- Identify at least one plausible alternative pipeline that a competent practitioner might choose, explain why it is reasonable, and justify why you did not select it.
- Perform at least one stress test (e.g., perturbations, reduced data, altered preprocessing, or metric sensitivity) and analyse how robust your pipeline is to this change.
- Report and submit the prediction scores produced by your final pipeline on the test set as a single-column file (named using your student ID, e.g., u100011.csv) in the same order as Xtest.
- Explicitly state the main assumption your pipeline relies on, and discuss how violating this assumption would affect your conclusions.

Your marks will prioritise coherence, robustness, and defensibility of the pipeline, rather than absolute test-set performance.