

Recognizing handwritten digits

INF283 - Project - 2

(Project 2 deadline: October 26th, 23.59).

Deliver here: <https://mitt.uib.no/courses/12791/assignments/15298>

Projects are a compulsory part of the course. You will need to get at least 50% of score to pass this project. This project contributes a total of 17 points to the final grade. You need to upload your answer to MittUiB.no/assignments before 23.59 on the October 26th. It will then be graded, and the points will be added to your total grade score.

Recognising handwritten digits:

In this project, we practice our machine learning skills in a scenario that mimics real-world machine learning projects. The goal is to produce a classifier that predicts labels of handwritten digits.

Grading will be based on the following qualities:

- Experimental design (quality of the classifier, design choices, model selection and evaluation)
- Correctness (your answers/code are correct and clear, **still important**)
- Clarity of code (documentation, logical formatting, **still important**)
- Reporting (thoroughness and clarity of the report, **more important than last project**)

Deliverables:

1. a PDF report containing an explanation of your approach and design choices to help us understand how your particular implementation works. You can include snippets of code in the PDF to elaborate any point you are trying make.
2. a zip file of your code. We may want to run your code if we feel necessary to confirm that it works the way it should.

Please include a README.txt file in your zip file that explains how we should run your code. In case you have multiple files in your code directory, you must mention in the README.txt file which file is the main file that we need to run to execute your entire pipeline.

Note: There are over 100 students at the course so to make grading easier we ask you to ensure that your deliverables obey the following instructions (a small point deduction may be made if they deviate from this):

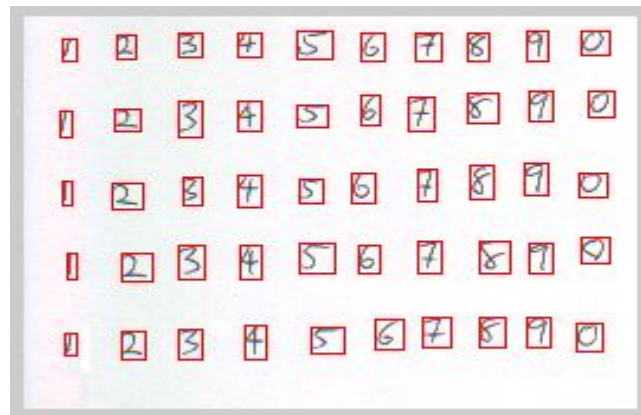
1. Deliver 2 files (a pdf and a zip), not one zip file with a pdf inside
2. Name your zip file with firstname_lastname.zip
3. Don't include the data set in your zip file that you submit on MittUiB.
4. Do not rename the two CSV files (handwritten_digits_images.csv and handwritten_digits_labels.csv) when using them in your code

5. Use relative path of these two CSV files in your code. Do not use absolute paths in your code. A perfect approach would be place these two CSV file in a directory one level up relative to your code directory, and then use the relative paths to these two files in your code.

Digit recognizer

Scenario:

You are working for a small company that provides machine learning solutions for its customers.



The postal office is developing an AI system to automatically deliver mail. As a part of the system, they need a computer program that recognises handwritten digits. Your company is providing this program and as a machine learning expert, you have been asked to develop such a program.

Task:

Download the MNIST dataset from MittUiB that contains letters for your classifier to train on.

Write code that produces a classifier

Write a report that describes what you have done


1.Data Format

The data is in the files `handwritten_digits_images.csv` and `handwritten_digits_labels.csv`. The images have the shape (70000, 784), where each row represents a 28x28 pixel grayscale image (28*28=784). Each pixel has a value of 0–255 (white to black). The images can be reshaped to (70000, 28, 28), in Python (Numpy) with


```
x_data = x_data.reshape(x_data.shape[0], 28, 28)
```

The labels have the shape (70000,), where each row is the label for a corresponding image (labels are 0-9).

Training: 7 Training: 9 Training: 1 Training: 1 Training: 3 Training: 4

A row of six handwritten digits: 7, 9, 1, 1, 3, and 4. The digits are written in a casual, slightly slanted style with varying line thickness.

Prediction: 3 Prediction: 1 Prediction: 7 Prediction: 9 Prediction: 6 Prediction: 4

A row of six handwritten digits: 3, 1, 7, 9, 5, and 4. These digits are more stylized and rounded than the ones in the first row.

2. Code

The goal is to produce a classifier that predicts the labels of handwritten digits as well as possible (It is up to you to decide a reasonable way to measure “goodness” of the solution).

This is not an implementation project and thus you are free to use libraries such as sklearn and keras.

You should try at least 3 different types of classifiers before choosing the final one. Note that showing effort to optimize performance will affect the grade positively.

It is important that your results are reproducible. That is, your customer should be able to verify your claims (Meaning that they should be able to easily run your code and get exactly the same numbers that you give in your report). Thus, you should write an automated test pipeline that runs all of your tests (given enough time). That is, training and assessment of all models and hyperparameters.

Note: read about random seed to understand how to get reproducible pipelines.

3. Report

The report should consist of two parts (in the same pdf):

- a. an executive summary
- b. a technical report.

a.

The executive summary is meant for the business managers of your customer and gives a short, non-technical overview of your project. You should also argue whether or not the machine learning approach is appropriate for this task based on your results.

b.

The technical report that tells what you have actually done and why. It should contain detailed information of your design choices and experimental design. Technical report should contain at least the following information:

- Preprocessing steps
- Candidate algorithms and choice of candidate hyperparameters (and why were the others left out?)
- Performance measure
- Model selection schemes
- What is your final classifier and how does it work. Justify why is it the best choice.
- How well it is expected to perform in production (on unseen data). Justify your estimate.
- Given more resources (time or computing resources), how would you improve your solution?

Use plots and figures whenever appropriate.

Very original work and good plots affect the grade positively.

As the report is a part of a course and our main goal is learning, it is also ok to report failed experiments. Especially, it is appreciated if you can explain why things didn't work as you initially expected.