

INFO284 Machine Learning Exam, spring 2026

Delivery date: April 30th 2025, 14:00

Format: 1) Jupyter notebook (ipynb-file) containing runnable Python code, documentation and reflections on the process and result;

2) a pdf file created from the Jupyter notebook with all Python code executed and models built.

Word limit: The total text parts should not be more than 3000 words. There are no limits on Python code size, as well as the number of figures, tables, and graphs.

Groups: for the assignment, you need to form a group of 3-4 people. It is **strongly recommended** that everyone in the group attends the same seminar slot. Having smaller groups is **not allowed** unless you get an official permission. If there are reasons that preclude you from joining a group, please write to the course coordinator and the seminar coordinator ASAP.

Delivery system: you submit your assignment as a group on Inspera. Do not forget to **anonymise** the documents, i.e. remove all the names.

Feedback sessions: we will run three (3) feedback sessions, where you can upload your current version of the assignment and get feedback from TAs during your seminar sessions. This is voluntary. Deadlines and links for uploading your work-in-progress are on MittUiB.

Submission information

You shall deliver the assignment in the form of a **well commented Jupyter notebook**. This code needs to run on the original data set, so any preprocessing you choose to do needs to be programmed in Python and included in the notebook. The code shall in the end return the results of your experiments with your chosen models. As a backup, you will also deliver **the same notebook** that you have run on your computer as a **pdf file**.

‘Well commented notebook’ means that you need to explain your choices and decisions based on the given data. Moreover, you need to provide your assessment of performance of your trained models and the reasons for such a performance.

For example, you need to explain

- Important and relevant properties of the data
- Your preprocessing steps. For example: your process of feature selection and its results, your choices when it comes to dimension reduction (why/why not/which method/why that method) etc.
- Based on what you have learnt from the data, why do you think that your models are best-suited for the task
- Why the particular parameters of a model that you use work best
- How you control over- and underfitting
- Your choice of evaluation methods. Which metrics did you choose and why? Additionally, you need to give an explanation based on your intuition about why given methods perform better or worse on the given task.

Provide the **list of libraries** you use in the form of a *requirements.txt* in the format used by pip.

Finally, as a concluding comment in the Jupyter notebook, you need to write a summary of your results, and discuss consequences of such results.

As the choice of evaluation is up to you, a high evaluation score is not necessary for any grade, including an A. Low scores need to be explained sufficiently, and an attempt to create a performant model must be clear from the documentation you hand in. The spirit of the task is not to create a performant model, but to showcase an understanding of relevant techniques and an ability to apply them sufficiently in practice.

Note: The datasets may be too large for your personal computer. You are encouraged to work around this by, for example, sampling a part of the data or using Google Collab. Indicate this issue in your submitted document.

Final note: The data is prepared for this course and are shared with you in confidence that you do not share them in any way but use them only for the purpose of this exam. Moreover, use of **external code** (from, e.g., stackexchange) should be clearly highlighted. The same goes for the use of **AI tools** like ChatGPT. Please, mark clearly, from where you adapt the code, if you do so.

Task I: Sentiment Analysis

On MittUiB you will find a compressed (zip) *reviews.csv* file that contains reviews, scores, and other information.

Your tasks are:

- a) **Exploratory data analysis and preprocessing:** Gain a sufficient understanding of the data for model development and perform data cleaning and feature engineering steps if you find it necessary. You may make any changes to your dataset; however, you must attempt to give a reason as to why you find a given transformation necessary.
- b) **Models:** Build four (4) machine learning models for predicting the assigned review score based on the review's content. Sentiment analysis is recommended. **One of the models needs to be a neural network (e.g. LSTM).** You are welcome to train models that are not covered in the course. Evaluate the performance of the built models using the appropriate evaluation metrics.

Task II: Convolutional neural networks

The data set for this task can be found here <https://shorturl.at/uDV6m>. It contains images created by people and images generated by AI.

You are supposed to

- a) **Train a convolutional neural network** as a binary classifier determining whether a given image was created by a person or generated by AI. In order to create such a classifier, it is recommended to start by finetuning a preexisting CNN. You must explain why the given model is the best choice for the task. You are also welcome to build a new architecture from scratch; however, you will need to identify and overcome the challenges posed by this decision in order to receive a good grade. If you have to chose a suboptimal model due to resource limitations, note what the optimal choice would have been and why yours is the best given your limitations. Usage of free online resources such as Google Collab is highly recommended.

b) Find **5 new images** (or create those yourself), show how you would use your model to classify the new images, and test the performance of your model on those images.

Some links to information about pre-trained CNN:

- <https://medium.com/towards-data-science/transfer-learning-from-pre-trained-models-f2393f124751>
- <https://medium.com/@mikhailenko/instructions-for-transfer-learning-with-pre-trained-cnns-203ddafe01>
- BOOK: F. Chollet. Deep Learning with Python. Ch. 5.3 Using a pretrained convnet. p.143-159