Portfolio-Exam

This is the description (7 pages) of the portfolio exam for the Data Science module MADS-MMS – Mathematics and Multivariate Statistics. The main idea of the portfolio is to conduct comprehensive experiments in several steps on **a single real-world dataset of your choice** (cf. Section 3), using clustering and dimensionality reduction methodology (details follow in the tasks). The project consists of three stages: The pitch, the draft, and the final.

1 Submission

There are two separate submissions:

Submission 1 Submit your pitch. Submit a single PDF-Document called `pitch.pdf` (see Section 2.1). Deadline: **Apr. 17, 2025**.

Submission 2 Compressed into one **ZIP**-file, submit

- a Jupyter-Notebook of the draft stage (Section 2.3) called `draft.ipynb` and a PDF export of that notebook called `draft.pdf`,
- a Jupyter-Notebook of the final stage (Section 2.4) called `final.ipynb` and a PDF export of that notebook called `final.pdf`,
- an additional notebook called `assembly.ipynb` containing the work of the assembly stage (Section 2.2) **if** necessary,
- a data folder **if** your data is not simply available as a single file online, and
- a resources folder **if** you have additional resources (e.g. images).

Deadline: **May 18, 2025**.

Upload your results to **Moodle before 11:59 pm (23:59 o'clock German time)** on the respective day of the deadline!

2 The Portfolio Stages

The portfolio exam consists of a data science project in three mandatory stages: pitch, draft and final. An optional assembly stage is only required if your dataset is not just loaded from a file or downloaded, but created by you (e.g. by combining different datasets into one, or downloading data from an API). In that case, the dataset is created and saved in the assembly stage, and then loaded in the draft and final stage.

2.1 The Pitch

The deliverable of the first stage is a presentation in PDF format, named `pitch.pdf`, containing 3-5 slides (hard limits) including a title card. The pitch will NOT actually be presented, thus the slides must be self-explanatory!

1. Present a real or a realistic fictitious situation, in which you are the data scientist trying to convince a customer to greenlight your project.
2. Propose to explore a **non-trivial dataset** through **clustering** experiments (as described in Section 4) to solve a meaningful problem.
3. Present and explain the a) goal of your experiments, b) the **specific** value (monetary or otherwise) that you expect to generate if those experiments are successfull, and c) the dataset and its suitability for the project.

The proposed dataset must be used in the following stages to approach the pitched project.

2.2 The Data Assembly Stage

This stage is only required if your dataset needs a longer creation process (e.g. by combining different datasets into one, or assembling data from API calls). **If your dataset is only a single file that can be loaded, you can skip this stage!**
The deliverable of this stage is an executable Jupyter notebook called `assembly.ipynb` in which the end result is the complete unpreprocessed dataset in the form that it is going to be used in the subsequent stages. This dataset should be stored here and loaded in the notebooks of the draft and final stage. The dataset must be included in the second submission.

2.3 The Draft

In a single executable Jupyter notebook named `draft.ipynb`, conduct multiple analyses and create preliminary results along the Tasks 1–6 (Section 4) using the pitched dataset. Results should be interpreted and conclusions must be drawn!

2.4 The Final Version

The deliverable in this phase is a single executable jupyter notebook named `final.ipynb` that is created within a template provided in the last lecture of the MMS module. The final version contains all experiments along Tasks 1–8 (Section 4), focussing, extending, and strengthening the work from the draft. For that purpose, you will have to select and rearrange parts of the draft to fit into the template. Some experiments may have to be extended, other shortened or omitted completely. The final version should be a concise, executable notebook that conveys the most important information to the customer in a plausible story, containing exaclty the relevant code, interpretation, discussion and conclusions.

- **The template must be used for the submission. It must not be rearranged nor modified beyond adding content where indicated.**

- The final version must be a fully executed Jupyter Notebook, containing all necessary code, results, and interpretation!
- Mind the restrictions, e.g., for the maximum number of bullet points or analyses.
- Since the final version is a new notebook, it naturally requires to re-run all experiments! Plan the time to do that!
- The final version is not just an extract of the draft version. It will be necessary to adapt and extends experiments and outputs to the requirements of the final version template.

3 Rules and Conditions

Programming Language Use Python and (where possible) scikit-learn for the submission.

Text Language Choose either English or German for all textual content.

Dataset The dataset may be chosen according to the following three requirements:

- Use **real** data from real applications/sources, no artificial datasets!
- Do **not** use datasets from the lectures. The document `lecture_datasets.pdf` from March 02, 2025 contains a list of datasets that are used in the modules MMS, ML, and DL. **None** of these datasets nor derivatives are allowed for this exam.
- The data must be available: There are two possible availability scenarios:
  - It is available online with proper license. In that case indicate in your notebook where the data can be downloaded.
  - You (legally!) obtain a dataset and share it with me via Moodle. Such data should not come with any form of NDA or other obligations. If you are not sure about these criteria regarding the dataset you consider, please contact me before you invest too much time in the experiments. If you write your own code to acquire data (e.g. querying an API), create a **separate notebook** for that purpose only and submit everything in a zip file.

No Teamwork This exam is supposed to be done during the self study time of the module. Students are allowed to exchange ideas. However, this is **NOT a teamwork exercise**. Every student must derive and write up their own solutions in their own words and programming style.

Code from other Sources You may reuse all the code from this module's lectures and exercises. Copying (and adapting) from other sources is allowed in small quantities – e.g. a function from stackoverflow. Quote the respective source. WARNING: Copying code in large quantities will be treated as intent to deceive and result in a score of zero points.

4 Tasks

Task 1 – The Data

Load and present the dataset. This is the raw data, that you will work on in the subsequent tasks.

- Name the variable of the dataset `raw_data`.
- Explain the dataset itself (e.g., what the features represent, units, . . .).

- Explain how the dataset is suitable for the pitched project.
- Keep the conditions for selecting datasets in mind!

Task 2 – Initial Data Analysis

Conduct an initial data analysis.

- Present basic relevant quantities of the data that inform the reader about the dataset.

Task 3 – Preprocessing

Check the data and determine if and which preprocessing is necessary or useful. Bring the dataset into the form that you need for the experiments.

- Clean the data.
- If necessary: create, transform, select features.
- If necessary: select/exclude data instances.
- After this task the dataset should be final with respect to the included features and instances.
- If you did change the dataset in this task, do not forget to present updated relevant quantities of the result, like in Task 2.

Name the variable that holds the final preprocessed version of the dataset `data`.

Task 4 – Exploratory Data Analysis: Statistical Properties

Use the preprocessed dataset (`data`) from Task 3 to compute interesting statistics, distributions, and feature relations that are relevant to the pitched task.

Draft Stage: Investigate different directions. Conduct at least three different analyses. Include all aspects that are relevant for the task at hand.

Final Stage: Pick exactly three highlights from the draft stage and present them. Pick those that have the highest relevance for your task.

Task 5 – Dimensionality Reduction

Use PCA to reduce the dataset `data`. Call the result `data_pca`.

- Analyze, discuss, and visualize relevant properties of the PCA transformation for your dataset and choose a reasonable number of components based on these observations.
- Apply the selected PCA transformation to your dataset.
- Should your conclusion of the analysis be, that no reasonable PCA transformation suggests itself, **then**
  - state and explain that clearly and
  - reduce the dataset anyway, by removing one component.

Task 6 – Exploratory Data Analysis: Clustering

Use clustering to identify different groups of instances in your dataset. Conduct the following steps **separately on both** `data` **and** `data_pca`.

- Use one clustering method from **each** of the following algorithm families: a) k-Means/k-Means++, b) DBSCAN/OPTICS, and c) HAC from the lectures to analyze clustering structure in the data.
- Where it is possible to choose a distance function, pick the one that seems the most promising for your task at hand. Explain! A separate evaluation of other functions for distance is not required for the exam.

Draft Stage Conduct **preliminary** experiments with these three algorithms, systematically try different hyperparameter combinations, create different clusterings. Discuss their merit (bullet points).

Final Stage For the final stage, you will be provided with additional information in the template.

- In the final version template, one of the three clustering algorithm families will be specified, further called A. Similarly, **two** clustering evaluation measures E_1 and E_2 will be specified.
- Familiarize yourself with the two clustering evaluation measures.
- Use the specified algorithm A in different parametrizations and both, E_1 and E_2, to conduct **comprehensive** clustering analyses and determine the most suitable clustering.
- Discuss the agreement between E_1 and E_2. If they suggest different parametrizations then you must make a choice and explain it.
- Analyze the quality and properties of the clustering (using the methodology from the lectures) as far as possible. (Interpretation of features in clusters is not yet necessary, this follows later in Task 8).

Task 7 – Clustering Comparison – Final Stage Only

Compare the two clusterings from Task 6 with respect to their number of clusters, number of noise points (if such were identified by Algorithm A), and their clustering evaluation results (using E_1 and E_2). Finally, use a table to describe and discuss the relationship between the two clusterings with respect to the data instances.

Task 8 – Conclusions and Future Work – Final Stage Only

Please address the following points separately and in that order.

1. Pick one of the clusterings from Task 6 as your final result and explain your choice.
2. Discuss the features of the various clusters. Use tables or visuals to communicate the analysis. Explain the results.
3. Summarize the achieved results. Compare your results to the expected or desired outcomes in the original plan (pitch).

4. Explain the generated value! How does the final clustering help the organization specifically? Recommend a course of action based on the results for the organization in your pitch.
5. Reflect on limitations and possible pitfalls of using these results.
6. Propose ideas for future work (a short sketch or enumeration of ideas is sufficient, no further experiments). The ideas should not be too general (e.g., "try further algorithms") but be specific to the project (e.g., "try Algorithm X, as because of Property Y, it might work specifically well on this dataset").
7. Critically discuss the employed methodology (your choices as well as the choices given in these tasks and in the template)! What could or even should have been done differently?
8. Critically reflect the original task you pitched. In hindsight, were the goals realistic? What could have been changed at the time of the pitch?

5 Expectations

Your submissions will be graded at the end of the term. That means, you will not receive separate feedback after each phase. The maximum number of points that can be reached is 100.

5.1 The Pitch – 15 points

The pitch basically provides the setting for the experiments in the subsequent phases. Criteria include: the description of a meaningful task with clear value to the customer, a convincing presentation, a reasonably chosen, promising dataset.

5.2 The Draft and the Final Version – 85 points

The main focus is on the final version. Of the draft version, it is required that the experiments work and that the draft contains more experiments (e.g. more data analysis than presented in the final version). The final version should address all tasks and will be assessed along two dimensions: experiments and presentation.

5.2.1 Experiments

When grading your experiments, I consider technical soundness, completeness, and fit to the tasks. I expect (among others):

1. The task is a non-trivial data science task.
2. Your code is executable and yields reasonable and reliably replicable results.
3. All cells of the notebook have been executed.
4. The choices of methodology (different approaches, dataset specific choices, settings of hyperparameters, etc.) make sense and are reasonably explained.
5. You have experimented with a reasonable selection of hyperparameters (just one selection is not sufficient).
6. You have addressed the influence of random choices and used appropriate steps to mitigate it.

5.2.2 Presentation

The presentation in the final notebook **must respect the template** and the requirements there! When grading the presentation, I will put myself into the position of your project's customer. I expect (among others)

1. that the code is structured (DRY principle, organized imports, ...) and commented and free of errors or distracting outputs (e.g., no debug messages).
2. that the code is documented using appropriate means of Jupyter.
3. that results are presented in helpful numbers, in tables, and customized diagrams which are referenced and **interpreted** in the text.
4. that diagrams are easy to understand (appropriate colors, tics, scaling, labelling, legends, ...).
5. that the text is easy to understand (short, concise sentences, proper references to (previous) results, diagrams, ...).
6. that I am guided through the different parts of the experiments and told what the purpose of upcoming code blocks will be.
7. that the **visualizations are actually visible in the uploaded notebook**! For example, there are known issues with `plotly` graphics (and other libraries), which only show in notebooks when executed directly, but not afterwards, unless explicitly configured.

6 Advice

- Depending on the project and dataset you choose, some of the portfolio tasks may be more or less extensive. E.g. if your dataset is already cleaned up, preprocessing might be very quick. However, in other situations you might want to make use of various preprocessing steps and iteratively refine the dataset before you run algorithms on it.
- When selecting a dataset for clustering, it is often easier if it contains many numerical features, rather than many categorical ones.
- In one of the tasks, you will have to use dimensionality reduction. Therefore best choose a dataset that does not have too few dimensions to begin with!
- Your notebook should contain MUCH MORE TEXT, introduction, comments, etc. than the notebooks we use to demonstrate methods in the lectures or the exercises!
- Before you submit, re-read the exam description and make sure that all points are addressed.
- Before you submit, re-run your notebook(s) and check that the results are still as expected.
- Keep in mind that this is NOT your Master Thesis. The experiments should be comprehensive and created and conducted solely by you. However, limit your work to the portfolio tasks, solving ONE problem – even though along the way you might recognize other interesting angles to follow up on.