Data Mining and Machine Learning (Group Coursework)

F20DL/F21DL (2023/2024)

In this group coursework, you and your team will work on a data mining and machine learning project using GitHub as your collaborative platform.

The project will cover various aspects of data mining and machine learning, including basic concepts, generative models, discriminative learning, and practical application of these techniques.

The project is designed to give you hands-on experience with real-world data and machine learning algorithms while also teaching you how to effectively collaborate using GitHub for version control, documentation, code reviews, regular commits, testing, and data cleaning.

Coursework Objectives

1. Gain a solid grasp of the fundamental concepts in data mining and machine learning, including datasets, dealing with missing data, classification, and supervised vs. unsupervised learning.
2. Explore generative models such as naïve Bayes, probabilistic graphical models, and cluster analysis (including k-means clustering and the EM algorithm).
3. Implement discriminative learning methods such as linear regression, decision tree learning, perceptron, and advanced models like multi-layer perceptron and deep learning architectures.
4. Develop practical skills by working on real-world datasets, conducting data preprocessing, implementing machine learning models, and evaluating their performance.
5. Learn how to effectively collaborate with your team members using GitHub for version control, code sharing, and project management. Your coursework documentation and evidence should be hosted on GitHub.

Summary of the Group Project

- Each group will select real-world datasets (at least one and at most three) suitable for data mining and machine learning analysis (see resources/links at the end of this document for details).
- The group will pitch the project title, idea and theme.
- The project should incorporate both generative and discriminative tasks, showcasing various algorithms discussed in the syllabus that are relevant to the chosen dataset.
- The project must include key components such as data preprocessing, model training, performance evaluation, and data visualization.
- Teams should provide a comprehensive documentation detailing their approach, results, discussions, and analysis. All documentation, commentary, and code must be hosted on GitHub (e.g., README/Jupyter Notebook files/Documentation).
- The grading is based on the marking rubric (back of this document)

Group size: 4-5 people in each group.
Effort: approx. 40+ hours per student
Credit: 40% of overall mark
Coursework timeline: 18th September 2023 (Coursework Released)
Assessment: **Continuous throughout course (attendance, discussions, checks, week 3-4 each group must pitch their project title, idea and theme to ensure it meets the coursework criteria)**
Deadline Submission: 15:30 (local time) on 27th November 2023

Coursework Overview

The group project at hand involves selecting and analyzing 1–3 unique datasets while dividing the project into five distinct phases. The project's overarching goal is to explore, analyze, and apply machine learning techniques to these datasets, contributing valuable insights and solutions to relevant research questions. The project emphasizes collaboration, data exploration, and the application of various machine learning methods, including clustering, decision trees, and neural networks.

Datasets

The first step is to select **up to three distinct datasets** (these should be unique to your group project). These datasets will serve as the foundation for the entire project, guiding the research direction and analysis.

It is absolutely imperative you ensure that your dataset is able to complete the course requirements (see below). You must take into consideration dataset complexities and size when planning your project.

Storing your dataset

GitHub supports large files but you might need to do some tweaking for this. We highly recommend using Google and asking for help if you struggle getting it set up.

The maximum file size allowed on GitHub is 100MiB, and anything above 50MiB receives a warning as it can impact performance. You can find more information on this here: https://docs.github.com/en/repositories/working-with-files/managing-large-files/about-large-files-on-github#file-size-limits

If your dataset contains files larger than 100MiB, you can use Git Large File Storage (Git LFS). The maximum file size allowed through Git LFS is 2GB.[1] GitHub has good documenation on how to use and configure and collaborate with Git LFS and we highly recommend them. You can find them here: https://docs.github.com/en/repositories/working-with-files/managing-large-files

If you are unable to commit your data to GitHub and are unable to use LFS, you should clearly include scripts which will automatically download your data and put it into a place that is expected. For reproducibility, you should try to automate as much of this as you can.

Useful Links to find datasets

The following list has some pointers to places where you might get some inspiration for data mining challenges together with associated data such as evaluation criteria and comparative performance data

- https://huggingface.co/datasets — consolidation of publicly available image, text, and audio datasets
- https://www.kaggle.com - source of lots of different data mining competitions.
- http://www.drivendata.org - source of lots of different data mining competitions with an emphasis on saving the world.
- http://multimediaeval.org/datasets/ - a range of data and evaluation criteria for different types of data mining problems involving multimedia and multi-modal data.
- http://www.kdnuggets.com/competitions/past-competitions.html - list of past data mining competitions; data and evaluation criteria is likely to be available for many of these.
- http://webscope.sandbox.yahoo.com - publicly available research datasets from Yahoo!
- http://www.kdd.org/kdd-cup - KDD Cup is an annual data mining competitbrief_template.pdf ion run by ACM SIG KDD; datasets, evaluation criteria, and info previous winners are available (note that the most recent competitions are actually hosted on kaggle.com).

[1] We are currently trying to increase this to 4GB, but at the time of writing it is 2GB.

Coursework Requirements

Your group project must use your chosen datasets and complete the following requirements (R1–R5):

R1. Project Topic, Direction, and Questions
In this phase, the group will define a topic and a set of clear questions and objectives based on the selected theme/datasets. Identify the specific problems or hypotheses to be addressed using the datasets (week 3–4: pitch the idea in lab to confirm the datasets/questions/direction).

R2. Data Analysis and Exploration
Comprehensive data preprocessing and cleaning to ensure data quality. Exploratory data analysis (EDA) to gain insights into the datasets. Visualization techniques to present data patterns and trends.

R3. Clustering
Implement an appropriate clustering algorithm to show aspects of the data with similar characteristics. Evaluate the performance of clustering algorithms. Interpret the results and discuss their implications.

R4. Decision Trees
Apply decision tree algorithms to build predictive models. Evaluate the decision tree models in terms of accuracy and interpretability. Discuss the practical applications of decision trees in the context of the selected datasets/topic.

R5. Neural Networks and CNN
Implement neural network models, including convolutional neural networks (CNN), for tasks such as classification or regression. Train and fine-tune the neural network models. Assess the performance of neural network models and compare them with other techniques.

Throughout this group coursework, the project team will work collaboratively to navigate through the various requirement phases, from selecting appropriate datasets to applying advanced machine learning techniques. The project's outcomes will contribute to the understanding of the selected datasets, provide solutions to research questions, and demonstrate proficiency in data analysis and machine learning. This structured approach will enable the group to make meaningful contributions to the field of data science and machine learning.

Each group is responsible for project planning, task allocation among team members, and ensuring efficient communication. Regular interactions with instructors and peer code reviews are encouraged to ensure the success of the group projects. Additionally, ethical considerations regarding data usage and privacy must be strictly adhered to when working with real-world datasets.

Finally, it is highly recommended that your dataset satisfies all the above course requirements. If you are unsure whether your choice will satisfy the above requirements, you should ask as soon as possible.

Submission

At the end of the project (submission deadline), you should '.zip' your entire repository and upload it on CANVAS. Your final project should demonstrate a clear understanding of the covered concepts and techniques in data mining and machine learning.

Each member is expected to contribute to the project with regular commits including:

- Your project must be managed and maintained on GitHub (e.g., documentation, wiki, scripts, ..).
- You must use Jupyter notebooks to evaluate/log your tests/results.
- Your project should be fully reproducible (e.g., README should explain the steps).
- Each member is expected to contribute to the project to the deliverables of the course through regular commits (weekly to the repository) — including informative comments/details/code reviews.

- All the project work (including documentation) should be developed and managed on the repository (committed/updated weekly).
- The project repository should conform with good practices and open standard (e.g., ReadMe, structure/folders, consistent naming conventions, ...)

Deliverables

There are two deliverables for this coursework:

D1. Short group project presentation/discussion in week 12. Each team will have 5-10 minutes to present their work and describe approaches they have taken. This will be followed by a few minutes of questions and discussion. This will probably be arranged via a booked session, where groups will be allocated a time.

D2. Teams must submit a project repository as a single zip by **15:30 on 27th November 2023** on CANVAS. Group submissions in mind; each team should nominate a team leader to make the submission on behalf of their team.

Important notes:

- If you fail to complete the GitHub repository or attend the oral discussion then you get **0 marks**
- **Regular attendance in labs is required by all team members** (actively demonstrate and present work during scheduled sessions)
- **Any data or resources used for the project must be accessible** (e.g., downloadable/run/test any submission on another machine)
- You should also **provide a markdown file that highlights all the progress and intermediate steps that you have taken during the coursework.** The markdown file must contain the following (a template is given in the github repo):
    - The questions, objectives, and hypotheses that you are trying to answer in the coursework
    - A brief description of the datasets that you are using
    - Section for each requirement that show the experimental design, comments and conclusions based on the results

If you have any questions or queries about the assessment, please do not hesitate to contact the Edinburgh teaching team: DongDong Chen (d.chen@hw.ac.uk), Ben Kenwright (b.kenwright@hw.ac.uk)

Late Submissions

Late submissions will be subject to the normal penalties as defined in the late coursework policy. The University recognises that, on occasion, students may be unable to submit coursework on the submission date or be unable to present their work on the submission date. In these cases, the University's Submission of Coursework Policy outlines are:

- No individual extensions are permitted under any circumstances.
- Standard 30% deduction from the mark awarded (maximum of five working days).
- In the case where a student submits coursework up to five working days late, and the student has valid mitigating circumstances, the mitigating circumstances policy will apply, and appropriate mitigation will be applied.
- Any coursework submitted after five working days of the set submission date shall be automatically awarded a no grade with no formative feedback provided.

Please contact your Personal Tutor or Counsellor if you are unable to meet the deadlines or need information for Mitigating Circumstances or Temporal Suspensions of Studies.

Plagiarism

In any coursework, it is imperative that students uphold the principles of academic integrity and ethical scholarship. Plagiarism, the act of presenting someone else's ideas, work, or words as our own without proper acknowledgment, is strictly prohibited. We are committed to producing original and authentic content, and any external sources, whether they be ideas, data, text, images, or any other material, must be appropriately referenced and acknowledged using the prescribed citation style. This not only ensures the credibility of our work but also demonstrates our respect for the intellectual contributions of others. Together, let us maintain the highest standards of honesty and integrity throughout our collaborative efforts.

https://www.hw.ac.uk/uk/students/studies/examinations/plagiarism.htm

Resources/Useful Links

The following list has some pointers to places where you might get some inspiration for data mining challenges together with associated data such as evaluation criteria and comparative performance data:

Getting started with Python

We recommend using conda to create a virtual environment (with Python 3.11) for the project.

- Installation: https://docs.conda.io/projects/conda/en/latest/user-guide/install/index.html
- Managing your environment: https://conda.io/projects/conda/en/latest/user-guide/tasks/manage-environments.html#activating-an-environment

Framework recommendations for the course

We recommend the frameworks used in the labs, however if you want to use others like PyTorch and HuggingFace you should feel free to do so. Remember, documentation is your best friend.

GitHub Repository

The repository for the project will be managed through **GitHub classroom**. See CANVAS for more details and ask for help if you have any issues.

Ensure that the repository is well-structured, organized, and follows industry best practices. Here's a suggested structure:

```
- data/                 # Store your datasets here
- notebooks/            # Jupyter notebooks for data analysis and modeling
- scripts/              # Python scripts for data preprocessing and machine learning
- documentation/        # Project documentation, including weekly updates
  - week1/
  - week2/
  - ...
- README.md             # Project overview and documentation index
```

When creating your repository, we recommend following the suggested structure as it should work for most people. However, this is *just a suggestion and if you have a better organisational structure, you should follow whatever is best for your project.*

It is extremely important that you keep a copy of your raw data separate to your preprocessed data. This is to aid with reproducibility for any other researchers. Your raw data should get committed into the repository once you have decided on your datasets and then not touched. If you need to change the raw data for any reason, you should clearly document your reasons within a commit and/or a pull request so that it can be tracked.

Week 3 – Pitch Your Project Title/Idea/Theme/Brief

Teams must share their project title/theme/idea with lab helpers in Week 3 (information must also be on the GitHub ReadMe page). In addition to describing the proposed project, it must include the details for the datasets.

You should also provide a project plan and timeline on GitHub (deliverables/stages/milestones). You will be able to discuss your project during the lab sessions. Each team should present their ideas/project direction including describing initial thoughts on which approaches they are going to use (or are thinking of taking). This will take place during the scheduled on-campus lab sessions.

Distributing Workload

While we encourage collaboration between teams, we also want to ensure that the workload for the deliverables is distributed evenly. If there are any issues regarding the management of the workload you can flag them up and we will track the status of the project and engagement in the repository.

Weekly Updates

You should create a separate section in your documentation to show your project's progress. Each week, your team should provide updates on the project's status, including references and details of what you've accomplished during that week. Be sure to include the following in your weekly updates:

- Week X: Summary of what was achieved during the week.
- References: Any external resources, papers, or datasets used during the week.
- Challenges: Any challenges faced and how you overcame them.
- Next Steps: Plans for the next week.

Failure to demonstrate progressive updates may result in a grade penalty.

Marking Rubric

		75-100%	50-75%	25-50%	0-25%
Topic/ Questions	15%	The topic is exceptionally well-defined, highly motivated, and sets the stage for significant contributions to the field. The questions are exceptionally challenging, innovative, insightful, and highly novel, demonstrating a deep understanding of the subject matter.	The topic is well-defined and moderately motivated, but there is room for improvement. The questions are challenging and somewhat innovative, showing potential for generating interesting insights. Some aspects of the questions may be novel, but not all questions meet this criterion.	The topic is somewhat clear but lacks strong motivation. The questions are somewhat challenging but lack innovation and fail to fully inspire interest or insight. Some effort is made to address novel aspects, but they are not well-defined.	The topic is unclear, vague, or lacks motivation. The questions are trivial, lack innovation, and fail to inspire interest or insight. There is no indication of any novel or challenging aspects.
Data	5%	The datasets are well-documented, including clear information on their source, collection methods, preprocessing steps, and data format. Detailed instructions and scripts are provided for acquiring, cleaning, and preparing the datasets, ensuring straightforward reproduction. Any gaps or missing data are clearly identified and strategies for resolution are documented (accounted for in the scripts).	The datasets are reasonably documented, with information on their source, collection methods, preprocessing steps, and data format, but some details may be missing. Instructions and scripts are provided for acquiring, cleaning, and preparing the datasets, but there are gaps in clarity or completeness. Gaps or missing data are identified, and some strategies for resolution are documented in the scripts, but they may not cover all scenarios.	The datasets have minimal documentation with some information on their source and data format. Partial instructions and scripts are provided for acquiring, cleaning, or preparing the datasets, but they lack detail. Some gaps or missing data are mentioned, but no clear strategies for resolution are documented.	The datasets lack documentation regarding their source, collection methods, preprocessing steps, and data format. No instructions or scripts are provided for acquiring, cleaning, or preparing the datasets. Gaps or missing data are not identified or addressed.
Algorithms/Analysis	15%	The choice of algorithms/analysis is highly appropriate, advanced, and well-suited to the different areas and questions. Complete in all areas, thorough, and exceptionally well-executed. Offers a wealth of informative insights, demonstrating a deep understanding of the subject matter.	The choice of algorithms/analysis is appropriate for the questions and shows a good level of sophistication. Comprehensive, covering all essential areas and relevant data. Provides valuable insights and contributes to a better understanding of the problem.	The choice of algorithms/analysis is somewhat appropriate but lacks depth or completeness. Covers some essential components but is not comprehensive. Some insights but does not fully address the questions or is limited in its informativeness.	The choice of algorithms/analysis is overly simplistic or inappropriate for the course. Incomplete, lacking key components, or relevant data. Does not contribute meaningful insights.
Results	25%	Results are highly relevant, explicitly tied to the analysis, and well-connected to the broader context. Visualizations are exceptional, conveying information correctly and effectively with clear, comprehensive reference information. Visualizations are well-labelled and enhance the overall understanding of the results.	Results are relevant and explicitly tied to the analysis and the broader context. Visualizations convey information correctly with adequate and appropriate reference information. While the visualizations are generally effective, there may be room for improvement in terms of clarity or detail.	Results are relevant but partially correct or partially complete. Visualizations convey some information but lack context for interpretation. Visualizations may be present, but their labels and reference information need improvement to enhance clarity.	Results are missing, incorrect, or not based on the analysis conducted. There is an inappropriate choice of visualizations or a complete absence of visual representations. Visualizations, if present, are poorly labelled or missing reference information, making them difficult to interpret.
Readability	10%	The code is very well-organized, providing an excellent reading experience. There is no unused or irrelevant code, ensuring a clean and focused codebase. Variable and function names are exceptionally clear and have a strong relationship to their purpose in the code, making it easy to understand.	The code is well-organized, making it relatively easy to follow the logic. There is little to no unused or irrelevant code that distracts from the main code. Variable and function names are clear and have a discernible relationship to their purpose in the code, enhancing readability.	The code is reasonably organized, but there is still some room for improvement. Unused or irrelevant code has been largely removed or isolated from the main project files. Variable and function names are generally meaningful and somewhat helpful for understanding.	The code is messy and poorly organized, making it difficult to follow the logic. There is a significant amount of unused or irrelevant code that distracts when reading.

					Variable and function names are unclear and do not help understand the code's purpose.
Code Reviews	5%	There is extensive evidence that group members are consistently providing constructive feedback on each other's code. Code reviews are a well-integrated part of the development process, resulting in substantial improvements in code quality. The feedback process is highly effective, fostering collaboration and significantly enhancing the overall quality of the codebase.	There is evidence that group members are regularly giving constructive feedback on each other's code. Code reviews are conducted regularly, leading to noticeable improvements in code quality. The feedback provided contributes to better coding practices and understanding among group members.	There is some evidence that group members are giving constructive feedback on each other's code. Code reviews occur sporadically and have a moderate impact on code quality. Some code improvements can be attributed to the feedback received, but it is not consistent.	There is little to no evidence that group members are giving constructive feedback on each other's code. Code reviews are infrequent or superficial, with minimal impact on code quality. The quality of the code remains largely unchanged despite the presence of group members.
Reproducibility	15%	Results, algorithms, or scripts in the project directory correctly load data and generate all results and figures as presented in the documentation. Detailed and comprehensive instructions are provided for reproducing results, including any extra steps or data validation against sources, with links or downloads clearly specified. Test cases are well-documented, including input data, expected outcomes, and clear, concise instructions for running tests. Reproduction is straightforward, transparent, and aligns perfectly with the documented results.	Results, algorithms, or scripts in the project directory can be reproduced with moderate effort. Detailed instructions are provided for reproducing results, including any extra steps or data validation against sources. Test cases are documented reasonably well, with clear instructions for running tests, but some minor improvements are possible.	Results, algorithms, or scripts in the project directory can be partially reproduced with some difficulty. There are some details for extra steps required to reproduce results, but they may be incomplete or unclear. Test cases are mentioned, but the documentation lacks some essential details, making it challenging to run tests effectively.	The code in the project directory fails to run or produce any results. There are no clear instructions or documentation for reproducing the results or algorithms. There is no evidence of any effort to ensure reproducibility.
Documentation	10%	The documentation in the repository is exceptionally well-structured, regularly updated, and meticulously maintained. It is detailed, correct, complete, concise, and well presented. The explanations and resources are of a high standard for clarity, correctness, and persuasiveness and is regularly refreshed to ensure relevance and accuracy.	The documentation in the repository is detailed, correct, complete, and concise. It presents a compelling argument or rationale for the topic with a well-structured approach. While it is excellent overall, there might be occasional minor issues with clarity.	The documentation in the repository is generally correct and complete, addressing the topic adequately. It presents a convincing argument or rationale for the topic, but there may be some minor gaps or room for improvement in clarity. The structure of the explanation is decent, but it may lack elegance or sophistication.	The documentation in the repository provided is illogical, incorrect, or incoherent, making it challenging to follow. It lacks a clear structure and fails to convey the intended message. The explanation does not provide a convincing argument or rationale for the topic.