Machine Learning and Data Mining (COMM055)
Coursework 2021
(Revised version due to Covid-19 pandemic)

Submission[1]	Your Project Plan	Week 6	Tue
	Final Report & All Files	Week 13	Tue
Weighting[2]	**100%** (no exam this year) Your final mark is your coursework mark consisting of different parts as described in Section 5.		

Important Note: This document is subject to revisions/updates. You will be informed of any variations to the above dates/times should variation become necessary/unavoidable. Please check SurreyLearn for the latest version of this document.

Extensions, Late Submissions and Academic Integrity: Coursework will be routinely checked for academic misconduct. Please refer to your Student Handbook and the advice given in SurreyLearn on cheating, plagiarism and collusion and make sure that you understand the regulations. If you are in any doubt, please seek advice from your Module Leader or Personal Tutor. Students are reminded of the University policy on late submission of coursework as outlined in your Student Handbook and on the policy for mitigating or extenuating circumstances.

1 Introduction

In this coursework you will use the knowledge and skills you have acquired over the Machine Learning and Data Mining (MLDM) module to explore datasets of your choice, apply ML algorithms and DM techniques, and evaluate and discuss the results and insights. This also provides an opportunity for you to practice the whole process of data mining (with special

[1] See Section 7.
[2] See Section 5.

emphasis on model development, evaluation and comparisons) and creating your own ML/DM pipeline (using Python), working in a group and subsequently reporting your project and findings.

1.1 Form a Group

To form your group, please complete the online form here:
https://forms.gle/KiBCrZ2uo2ejdr7u8

If you are not part of any group, you still need to complete this form and you will be assigned to a group. Each group must consist of 4 or 5 people. Any smaller or larger group is only allowed in special circumstances and this must be agreed and confirmed by the module leader.

1.2 Choose Datasets

Your group must choose **a minimum of 2 datasets** for your project. You can choose suitable datasets for your project from repositories such as:

- https://www.kaggle.com/
- https://archive.ics.uci.edu/ml/datasets.php
- http://aws.amazon.com/datasets
- http://catalog.data.gov/dataset
- http://data.un.org/
- …

See also the lists at:

- https://github.com/awesomedata/awesome-public-datasets
- https://github.com/openml/OpenML/wiki/Data-Repositories
- https://www.kdnuggets.com/datasets/index.html
- https://www.nature.com/sdata/policies/repositories
- https://toolbox.google.com/datasetsearch

Using more datasets gives you more opportunities for comparing and evaluating different algorithms and techniques. You should select diverse/different datasets that allow you to demonstrate the strength and weakness of the algorithms and techniques that you are using for model creation and evaluation. Datasets on these repositories may have been prepared to a certain extent but you may still demonstrate your understanding of the data mining steps (including meaningful data understanding and data exploration and visualisation) and if possible, show what additional ways you have processed the data.

Your group must select **at least 4 different learning algorithms** from the 10 different machine learning approaches mentioned in Section 3.3. Your selection must include at least one of the learning algorithms which require special inference engine (i.e. Logic-based and relational learning) or special environment (i.e. Reinforcement learning). I.e. **your selection must include at least one algorithm from group (8) or (10).**

If you selected relational machine learning you can still choose datasets from the repositories mentioned above. But you can also use datasets which are already prepared for relational machine learning such as:

- https://github.com/joschout/RelationalDatasets
- https://www.doc.ic.ac.uk/~shm/Datasets
- https://github.com/JoseCSantos/GILPS/tree/master/datasets

These datasets are also ready to be used with other machine learning algorithms.

If you selected reinforcement learning then you can use an environment from the OpenAI Gym (instead of a dataset) similar to what you have seen in the labs.

2 Learning Objectives

- To deploy ML/DM on real-world datasets.
- Select the most appropriate ML/DM techniques for a given problem and provide a well-reasoned rationale for the choice of solution
- Understand and effectively use variety of ML algorithms and DM techniques
- Discuss the benefits / drawbacks of different approaches and evaluate and compare them using different performance metrics on different type of data
- Design and evaluate ML/DM techniques and tools to discover new relations and insights for a given problem
- To learn how to work in a team composed of people with diverse background and skills (and personalities).

3 Requirements

You are to demonstrate your ability in devising a specific analytical approach using a ML/DM pipeline implemented in Python based upon your chosen dataset and problem outline. The data mining steps must follow the CRISP-DM[3] methodology and include the following four stages of work (**with special emphasis on model development, evaluation and comparisons**):

3.1 Project Definition

You must understand the datasets of your choice and give a problem statement for each dataset. It is useful to identify and formulate questions (or hypotheses) that you want to address. You must make sure that each dataset can be used to answer the questions/hypotheses. Some points to consider:

- What are the major questions you wish to answer for each dataset?
- What ML/DM techniques do you plan to use?
- What performance metrics do you want to use to compare different algorithms?
- How do you plan to distribute your workload within your team?

[3] See Cross-Industry Standard Process for Data Mining (CRISP-DM) manual: ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf

3.2 Data Preparation, Pre-processing, Integration and Exploration

This stage is about knowing the data and preparing for analysis. It may involve:

- Data loading
- Data cleaning
- Data integration
- Variable transformation
- Dimensionality reduction
- Derivation of new variables

Data exploration tasks are typically performed to ensure the validity of the data preparation work and gain initial understanding before model development and evaluation. The dataset to be investigated must be prepared (e.g. in Python/Jupyter) to be used by your subsequent choice of modelling algorithms and evaluation metrics. Examples of tasks are:

1. Determine the variables (fields) to be used
2. Determine if a variable is a numeric or binary
3. Apply scaling if needed
4. Determine outliers in real numbers
5. Determine if the dates are reasonable
6. Determine the data subjects (fields) that will be included in your dataset, what is the inclusion criteria. Similarly, determine the data subjects (fields) that will be excluded from your study and decide the exclusion criteria.

If you are applying reinforcement learning to a problem, this stage involves setting up the environment including the reward function.

3.3 Model Development and Evaluation

This stage typically requires repeated iterations of modelling that involve:

- The design of modelling tasks
- The selection of learning algorithms (at least 4)
- The choice of the algorithm parameters (parameter tuning)
- The evaluation of the learned models using (at least 2) performance measures/metrics
- The collection of results

Please note that you need to select **at least 4 different learning algorithms** from different machine learning approaches such as: (1) Decision trees, (2) Neural nets (inc. Perceptron, MLP, etc), (3) Bayesian learning (inc. NB, BNs, etc.), (4) Instance-Based Learning (inc. kNN), (5) Support Vector Machines and (6) Clustering (inc. K-means, Mixture Models, etc), (7) Rule-based learning, (8) Logic-based and relational learning (inc. ILP, MIL, etc), (9) Evolutionary machine learning (inc. LCS, etc), (10) Reinforcement learning (inc. QL, DRL, etc), etc. Your selection must include at least one of the learning algorithms which require special inference engine (i.e. Logic-based and relational learning) or special environment (i.e. Reinforcement learning). I.e. **your selection must include at least one algorithm from group (8) or (10).**

Please note that the 4 different algorithms must be selected from different groups, e.g. Perceptron and MLP cannot be considered as 2 out of 4 (though it is recommended to use and compare as many algorithms as possible). The evaluation of the modelling results is critical to success and an important step. The resulting models (e.g. classifiers) must be evaluated **using at least 2 performance measures/metrics**, e.g. predictive accuracy using cross-validation, ROC curves, etc for classification problems and mean-squared error (MSE), mean absolute error (MAE), etc for regression problems, and so on. **The models should be also evaluated based on their interpretability/explainability, their ability to generalise, interestingness or significance.** The design of the tasks must be sound. Appropriate parameter settings must be used and justified. A systematic approach for parameter setting is preferred, but inevitably there will be elements of experimental trial and error.

3.4 Result Collection and Interpretation

Results must be compiled from the output of your Python pipeline. You must analyse and discuss the results and your interpretation may confirm or reject any initial hypothesis or indicate newly discovered trends/patterns in the data.

4 Deliverables

As a group, you will create and deliver:

1. Planning report
2. Implementation
3. Final report

4.1 Planning Report

The Planning report will not be assessed. This is for you to demonstrate that you understand what you are being asked to do and how you plan to go about doing it, as well as for the course instructors to ensure that you are on the correct path and that your choice of dataset is reasonable. Feedback may be provided based on the content of the plan.
The written document must include:

1.	Group name & members	State your group name and list your team members
2.	Datasets	What are the problems / datasets you think you will be using?
3.	Learning algorithms	What algorithms / approaches do you think you will be using, e.g. a selection of at least 4 from (1) Decision trees, (2) Neural nets (inc. Perceptron, MLP, etc), (3) Bayesian learning (inc. NB, BNs, etc.), (4) Instance-Based Learning (inc. kNN), (5) Support Vector Machines and (6) Clustering (inc. K-means, Mixture Models, etc), (7) Rule-based learning, (8) Logic-based and relational learning (inc. ILP, MIL, etc), (9) Evolutionary machine learning (inc. LCS, etc), (10) Reinforcement learning (inc. QL, DRL, etc), etc and including at least one from (8) or (10)
4.	Expectations	What are you expected results, e.g. algorithm ‘A’ works better on dataset D_1 and algorithm ‘B’ works better on dataset D_2, etc.

5.	Project plan	Include a realistic schedule on a weekly basis. You must specify the names of those in your group that are allocated to each identified task.

4.2 Implementation

Your group is required to implement a ML/DM pipeline in Python. This includes code for data preparation, data exploration and visualisation, model construction (e.g. learning a classifier) and evaluation (e.g. cross-validation) as well as the code for visualising the results. This implements your design to meet the requirements in 3.2 to 3.4. For relational learning you can use other languages and tools outside Python (e.g. Prolog) for model development and evaluation (3.3).

Your code must process your chosen datasets, run your pre-processing and chosen ML algorithms, evaluate and output results and provide visualisation as appropriate:

- You can use existing machine learning libraries (e.g. sk-learn)
- You can include your own appropriately named functions implementation in Python code.
- You must include handling of the dataset, such as retrieving data and the storing and output of results.
- Your application must run within the Python environment without critically stopping or aborting.
- You are required to use the same version of Python as in the labs.
- Source code is provided on SurreyLearn for each individual lab sessions that gives you example implementations of ML/DM pipelines in Python. You are welcome to use these Python codes.
- In general, when in doubt, provide a simple reference (at least as a courtesy) where you have taken and used any code from an external source – even if you have amended this. You will need to judge if you believe the code to be "sufficiently different" to the source so that you do not need to reference. If the code is generic or "obvious" then you may consider this unnecessary.
- Remember to include comments to document your code and choice of libraries.
- Your code should output tables and any plots/graphs/visualisations as necessary. These can be exported and used in your Final report where appropriate.

You need to be aware that you may not get a good mark if you only demonstrate a minimum level of skill by extensive use of Python code copied from other sources.

4.3 Final report

Your group will produce a concise final report using the paper template attached to this document. You should follow the rules mentioned in the appendix regarding the size and format of the final report. Your report should be also succinctly written, making use of tables and charts where appropriate. You are to:

	Provide	See
1.	Provide a project definition (see 3.1). It will contain an overview of your chosen datasets, including a data dictionary. You will describe your pipeline design using the design patterns given throughout the labs, your choices and assumptions and detail the main functions of your application, etc.	3.1

	Provide	See
2.	Explain your choice of data preparation and exploration approaches and each of the steps, make sure you justify your choices adequately.	3.2
3.	Document how you choose appropriate machine learning algorithms and justify your approach, along with your assumptions.	3.3
4.	Provide the summarised results from your Python code in tables/charts as appropriate for a technical audience. This must include a technical evaluation of your different models.	3.4

4.4 Contributions section

Your group must collectively agree on everyone's contributions to the project and your final report should include a 'Contributions' section where each group member writes a sentence summarising their contributions to the project.

5 Coursework Marking

The marking scheme assumes that each group member is contributing to the project with different degrees of commitment, and therefore they may have different final marks. Your final mark is your Coursework Mark which consisting of different parts (see mark descriptions below) including your Individual Commitment. Your individual commitment will be assessed by:

(a) Your group members confirming your contributions to the project. This will be based on the 'Contributions' section in your final report and follow up enquiries (if needed).
(b) The academic overseeing the assessment. The individual commitment will be also judged based on the student's engagement in the lab sessions evident by the lab exercises submitted to SurreyLearn.

NOTE: Please note that if based on evidence, it is clear that an individual in a group has not contributed at all, then they will be considered as "not submitting" coursework and will get 0 mark for the coursework (even if they attended and submitted all lab exercises).

Below are mark descriptors for each part of coursework.

5.1 Mark Descriptors

The mark descriptors for this module are aligned with those required by the University of Surrey, "Code of practice for assessment and feedback". The following are the mark descriptors for each part of the coursework:

Project Definition [total 5 marks]	
70-100	Well-articulated objectives demonstrating a good problem understanding, clearly described assumptions and hypotheses, with described measurable outcome.
60-69	Clearly defined objectives, good understanding of the problem, but the strategies of which are not entirely clear.
50-59	Implied objectives, vague understanding of the problem. Implied strategies to measure the objectives.

40-49	Implied objectives, lack of understanding of the problem, *ad hoc* strategies for measuring the objectives.
0-39	Objectives were not stated or clearly articulated. No means of measuring how objectives are achieved. Mis-represented or misunderstood the Coursework objectives.
Data Preparation [total 5 marks]	
70-100	Well-justified strategies with clear objectives when preparing the data, including a clear plan of measuring data quality.
60-69	Well-structured steps in preparing the data, with somewhat clear objective but did not have a plan for measuring data quality.
50-59	Well-structured steps in preparing the data, but the objectives for data preparation are mostly implied. No clear means of assessing data quality.
40-49	No clear steps in preparing the data, implied objectives for data preparation. No clear means of assessing data quality.
0-39	Data preparation is *ad hoc*, the objective of which is not clearly stated; and data quality not measured. Demonstrates little or no awareness of data preparation approaches.
Model Development & Evaluation [total 30 marks]	
70-100	Clearly articulated choice of modelling methods (at least 4), considering both pros and cons of each possible methods, clear evaluation methodology and explanation of evaluation criteria which is non-arbitrary. Excellent choice of (at least 2) performance measures/metrics. The models are also evaluated based on their interpretability/explainability, their ability to generalise, interestingness or significance.
60-69	Well-justified modelling methods and a good explanation of evaluation criteria with appropriate performance measures/metrics but implied evaluation methodology and unclear choice of parameter values.
50-59	Well-justified modelling methods but lack of justification for the evaluation criteria, poor or limited use of performance measures/metrics for the unbalanced dataset, unclear choices of parameters used.
40-49	Somewhat justified modelling methods, no explanation of choice of evaluation criteria, unclear or non-robust use of dataset for evaluation or parameters.
0-39	Unclear justification of the modelling methods, lack of evaluation criteria, no clear partitioning of dataset for evaluation, poor or incorrect use of threshold choice or parameters.
Data Visualisation, Result Collection, Interpretation and Critical Assessment [total 20 marks]	
70-100	Excellent use of charts and diagrams, with clear labels, legends and crisp explanation, demonstrating a full understanding of the figures; correct interpretation of the figures, leading to potentially actionable conclusions.
60-69	Suitable use of charts and diagrams, but labels and legends are not always clear. There is evidence of good understanding of the figures, with reasonable interpretation but it remains unclear how the findings can be used.

50-59	The charts used are acceptable but alternative means of presentation could have been used; labels and legends could have been made clearer. There is some evidence of understanding. It remains difficult to identify actionable conclusions.
40-49	The charts used are inappropriate; labels and legends could have been made clearer. There is some evidence of understanding but the results of which are difficult to interpret. No actionable conclusions can be identified.
0-39	Inappropriate choice of visualisation, no clear labels, insufficient explanation or lack of understanding about the figures being produced; inappropriate or incorrect interpretation the figures; not sufficiently critical in assessing the result.

Quality of Final Report [total 5 marks]	
70-100	Excellent structure and a concise report; making use of clear formatting and consistent referencing. Use of well-defined and simple tables, charts (including defined confusion matrix) and diagrams used to enhance understanding, discussion and actionable conclusions.
60-69	Well-structured report with a crisp message but the format not always consistent and some topics discussed could have been made clearer. Tables, charts and diagrams do not always support or enhance the readers understanding.
50-59	The report is of acceptable quality but could have been organised better; so too the messages. Some sections could have been made clearer.
40-49	The report was not well organised. The message was not convincing or clear.
0-39	No evidence of any substantial effort. There is an impression that the report and work was disorganised. The message was not convincing or clear.

Quality of Implementation [total 5 marks]	
70-100	Excellent structured code making use of appropriate algorithms from machine learning libraries or own written functions and clear commenting throughout. Demonstrates an understanding of the algorithms and use of libraries. The code runs without needing modification with no fatal errors/aborting to achieve all the coursework objectives. Where appropriate, intermediate results are output. The final results/tables/charts are clear. Multiple machine learning algorithms (at least 4) are implemented and their parameters determined within the code.
60-69	Excellent structured code making use of appropriate algorithms from machine learning libraries or own written functions but some use is unclear or inconsistent. The code runs without needing modification with no fatal errors/aborting to achieve most of the coursework objectives. The final results/tables/charts are not always output or could have been made clearer. Some commenting in the code. The parameters for ML algorithms might have been more clearly set.
50-59	The code runs but may have required some modification to do so without fatal errors/aborting. The code implements most of the coursework objectives. The use of appropriate algorithms is unclear or inconsistent. The final results/tables/charts are not always output or need to be clearer. Some commenting in the code. The parameters for ML algorithms might have been more clearly set.

40-49	The code does not run and requires significant modification to do so. The code implements some of the coursework objectives. Inappropriate or unclear use of machine learning algorithms. Output of results are erratic or unclear. Limited or no appropriate commenting in the code. Large use of cut/paste code from other sources.
0-40	No evidence of any substantial effort. The code does not run and has not been tested; it will require extensive modification to run. Very few coursework objects have been implemented. Poor and disorganised structure of code.
Individual Commitment [total 30 marks]	
75-100	The student was actively engaged in the project and lab exercises, attended and contributed to most of project meetings and lab sessions and properly completed the tasks assigned to them.
50-75	The student attended some of project meetings and lab sessions and did not properly complete the tasks assigned to them (and did not seek help to resolve the problems).
25-50	The student was not engaged in the project and did not properly complete the tasks assigned to them (and did not seek help to resolve the problems).
0-25	The student was not engaged in the project and did not have significant contribution.

The marks are summarised:

Item	Marks (%)
Project Definition	5
Data Preparation	5
Model Development & Evaluation	30
Data Visualisation, Result Collection, Interpretation and Critical Assessment	20
Quality of Final Report	5
Quality of Implementation	5
Individual Commitment	30
Total Mark	100

6 What to Submit

All submissions are through SurreyLearn.

Submission	Format	Description	Submit
Plan Report	doc, docx or PDF	Your Planning Report	As a single document
Final Report	doc, docx or PDF	Your Final Report	You are required to submit these files as one ZIP file to be named: project_name_MLDM.zip
Implementation	Python or Jupyter (one or more files)	Data/setups/all source files	
Set up document	doc, docx or PDF	A short file that describes how to set up your	

	implementation and any prerequisites	

7 When to Submit

What to submit	Submission Deadline	Date due for Feedback	Marks
Plan report	Tue Week 6 by 4:00pm	TBC	Formative
Final report and all source files	Tue Week 13 by 4:00pm	TBC	See section 5

8 Final Report Template (TBC)