MS4S10 COURSEWORK 1: 2020/21

Submission deadline: **23:59pm Tuesday 16th February 2021.**

Contribution to module: **50%**

Your report should be submitted via Blackboard assessment, as a pdf file containing the assessment cover sheet.

You are required to produce a formal report summarising your results for the 4 tasks (A, B, C and D) detailed below. You are required to apply the machine learning algorithms seen in the first part of this module along with the pre-processing steps outlined. You can apply any machine learning algorithm or pre-processing steps appropriate for the required tasks. This report should be created in Jupyter Notebook using appropriate code snippets and visualisation.

Before the deadline, students should also email moizzah.asif@southwales.ac.uk a zip file containing the Jupyter Notebook (report's html/pdf + code's .ipynb) of your analyses. The file name and email subject should be your student number.

Introduction

The learning analytics research group at the Knowledge Media institute, The Open University, has provided a publicly accessible dataset called **Open University Learning Analytics dataset.**

The dataset contains data on students' undertaking 7 selected modules and their interactions with the university's VLE (Virtual Learning Environment). So, the data has modules, students and VLE interaction records in separate csv files. It comprises of 32,593 students and 22 courses and their modules. This dataset has records of these students' interactions with the university's VLE (behaviour) and their assessment results (performance).

The data has been presented in a certain schema via tables. The tables are connected with each other using unique identifiers. These relationships are established in the Entity Relationship (ER) diagram as shown in Figure 1.

The tables shown in ER diagram are each populated in a separate csv file. The following section provides description of each table (csv file).

Data Description

course.csv

File contains the list of all available modules and their presentations. The columns are:

1. *code_module* – code name of the module, which serves as the identifier.
2. *code_presentation* – code name of the presentation. It consists of the year and "B" for the presentation starting in February and "J" for the presentation starting in October.
3. *length* - length of the module-presentation in days.

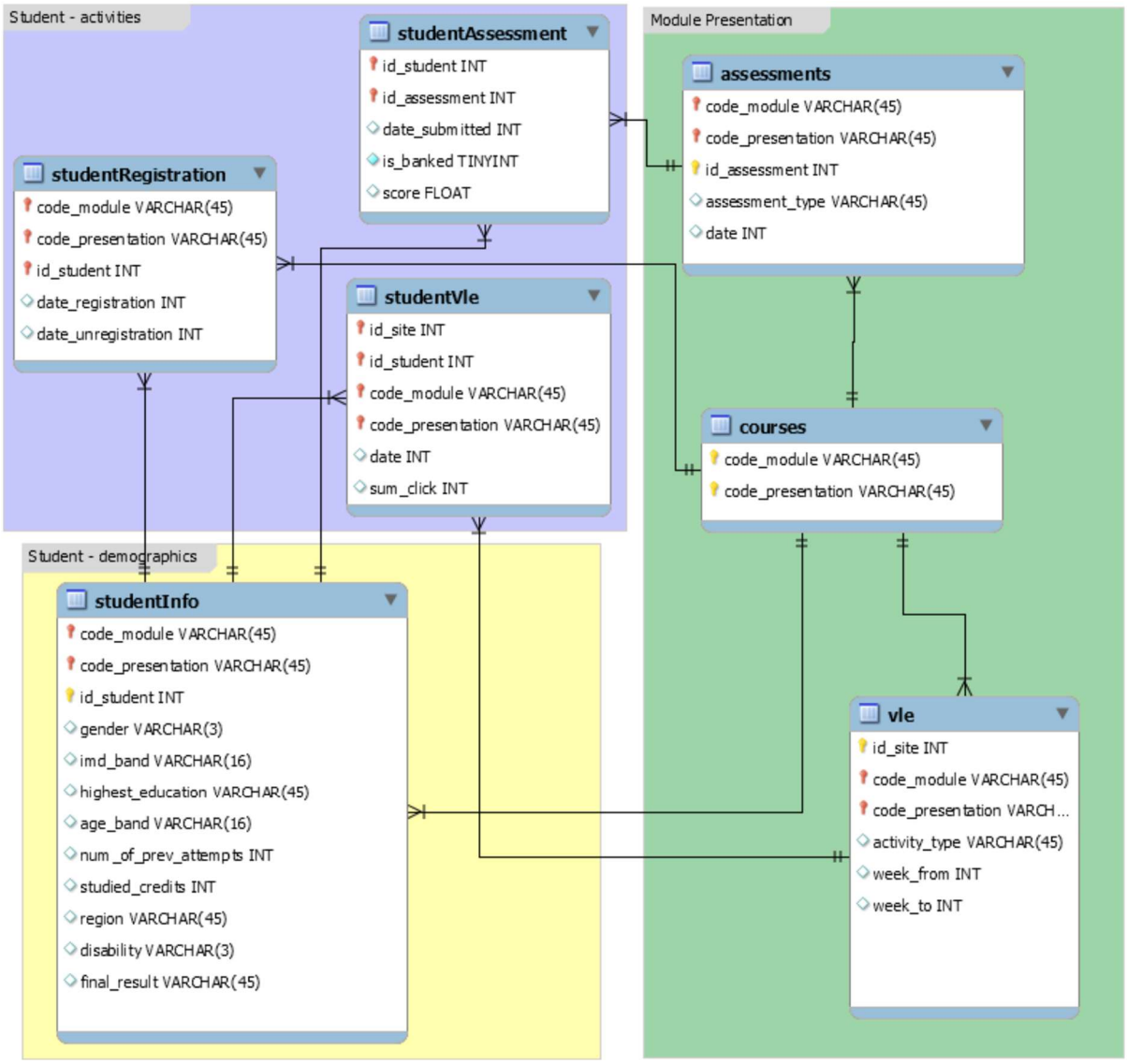

Figure 1 OULAD Entity Relationship Diagram

assessments.csv

This file contains information about assessments in module-presentations. Usually, every presentation has a number of assessments followed by the final exam. CSV contains columns:

1. *code_module* – identification code of the module, to which the assessment belongs.
2. *code_presentation* - identification code of the presentation, to which the assessment belongs.
3. *id_assessment* – identification number of the assessment.
4. *assessment_type* – type of assessment. Three types of assessments exist: Tutor Marked Assessment (TMA), Computer Marked Assessment (CMA) and Final Exam (Exam).
5. *date* – information about the final submission date of the assessment calculated as the number of days since the start of the module-presentation. The starting date of the presentation has number 0 (zero).
6. *weight* - weight of the assessment in %. Typically, Exams are treated separately and have the weight 100%; the sum of all other assessments is 100%.

If the information about the final exam date is missing, it is at the end of the last presentation week.

vle.csv

The csv file contains information about the available materials in the VLE. Typically, these are html pages, pdf files, etc. Students have access to these materials online and their interactions with the materials are recorded. The vle.csv file contains the following columns:

1. *id_site* – an identification number of the material.
2. *code_module* – an identification code for module.
3. *code_presentation* - the identification code of presentation.
4. activity_type – the role associated with the module material.
5. *week_from* – the week from which the material is planned to be used.
6. *week_to* – week until which the material is planned to be used.

studentInfo.csv

This file contains demographic information about the students together with their results. File contains the following columns:

1. *code_module* – an identification code for a module on which the student is registered.
2. *code_presentation* - the identification code of the presentation during which the student is registered on the module.
3. *id_student* – a unique identification number for the student.
4. *gender* – the student's gender.
5. *region* – identifies the geographic region, where the student lived while taking the module-presentation.

6. *highest_education* – highest student education level on entry to the module presentation.
7. *imd_band* – specifies the Index of Multiple Depravation band of the place where the student lived during the module-presentation.
8. *age_band* – band of the student's age.
9. *num_of_prev_attempts* – the number times the student has attempted this module.
10. *studied_credits* – the total number of credits for the modules the student is currently studying.
11. *disability* – indicates whether the student has declared a disability.
12. *final_result* – student's final result in the module-presentation.

studentRegistration.csv

This file contains information about the time when the student registered for the module presentation. For students who unregistered the date of unregistration is also recorded. File contains five columns:

1. *code_module* – an identification code for a module.
2. *code_presentation* - the identification code of the presentation.
3. *id_student* – a unique identification number for the student.
4. *date_registration* – the date of student's registration on the module presentation, this is the number of days measured relative to the start of the module-presentation (e.g. the negative value -30 means that the student registered to module presentation 30 days before it started).
5. *date_unregistration* – date of student un-registration from the module presentation, this is the number of days measured relative to the start of the module-presentation. Students, who completed the course have this field empty. Students who unregistered have Withdrawal as the value of the final_result column in the studentInfo.csv file.

studentAssessment.csv

This file contains the results of students' assessments. If the student does not submit the assessment, no result is recorded. The final exam submissions is missing, if the result of the assessments is not stored in the system. This file contains the following columns:

1. *id_assessment* – the identification number of the assessment.
2. *id_student* – a unique identification number for the student.
3. *date_submitted* – the date of student submission, measured as the number of days since the start of the module presentation.
4. *is_banked* – a status flag indicating that the assessment result has been transferred from a previous presentation.
5. *score* – the student's score in this assessment. The range is from 0 to 100. The score lower than 40 is interpreted as Fail. The marks are in the range from 0 to 100.

studentVle.csv

The studentVle.csv file contains information about each student's interactions with the materials in the VLE. This file contains the following columns:

1. *code_module* – an identification code for a module.
2. *code_presentation* - the identification code of the module presentation.
3. *id_student* – a unique identification number for the student.
4. *id_site* - an identification number for the VLE material.
5. *date* – the date of student's interaction with the material measured as the number of days since the start of the module-presentation.
6. *sum_click* – the number of times a student interacts with the material in that day.

Aim of the coursework

For this coursework you are expected to perform all of the following *tasks* to predict which students are suspected to fail or withdraw from a module.

Task A (20%)

Within Task A, you should look to:

1. conduct exploratory analysis of the dataset,
2. use pre-processing techniques to modify or produce new tables (set of features) suitable for testing machine learning models deployed in later tasks.

Useful hints

The data is spread into 7 tables and encapsulates plenty of interesting insights. Try to discover some insights which inform the next steps of the coursework.

Based on the insights, you should be able to apply feature engineering techniques, such as feature extraction and selection.

There should be a trail of informed decisions throughout your coursework and recording these tasks forms a major part of that trail as such, you should provide informative comments throughout your code.

Task B (20%)

Using the features/attributes provided within the data and those extracted and selected in **Task A**, conduct an unsupervised analysis, and interpret the produced groups/clusters in relation to the aim of the coursework.

Apply a minimum of two types of algorithms same/similar to those as discussed in the lectures to create unsupervised models. Compare and interpret the results, while making use of relevant and appropriate evaluation metrics.

Useful hints

It is not necessary that every interpretation has to be made with regards to a pre-determined target variable in preparation of a supervised learning task. You may as well interpret the results and uncover trends, and hidden groups which may not very well be linked to the final result but can lead to other directions.

To justify the decisions and choices you have made, it is vital to support and reflect on the process of choosing final models' parameters and evaluation metrics.

Task C (20%)

Find an optimal supervised learning model to predict a target variable which helps you achieve the aim of the coursework. You are advised to look at this prediction as both a regression and a classification problem and present the best models for both the categories.

Useful hints

Explore a variety of machine learning algorithms, ranging from probabilistic, tree based (ex: CART, Random forest and etc) to advanced algorithms such as support vector machines.

Using suitable evaluation measures, helps interpreting the models.

Linking the exploratory data analysis with feature importance can be a pretty impressive way of concluding the coursework.

Task D (20%)

Provide well written paragraph/s in your Jupyter notebook addressing the following subtasks:

Subtask 1. At the end of each of task A, B and C's code in Jupyter note book:

summarise and reflect on the task. Use this as an opportunity to present and support your decisions and choices which may or may not have been reflected in your python scripts including comment.

(You are encouraged to use visualisations and tables)

Subtask 2. Introduce the purpose of your Jupyter notebook and outline the structure of the notebook. State which python libraries/packages you have used and why so. Also highlight the final supervised and unsupervised models from Task B and C with their main results and outcomes.

Subtask 3. Add a section at the end of your notebook which draws final conclusions, recommendation from all your tasks. It is imperative to provide the limitations of the work done when providing any recommendations on further analysis.

Useful hints

GENERAL

Avoid grammatical errors and write coherent and full sentences in each section.

For each task: use titles, and headings. They should be meaningful and indicative of the main coursework task you are attempting. You can use distinctive numbering system for headings, sub headings and etc.

Figures/visualisation and tables should be numbered and labelled in a meaningful way and referenced accordingly in any text.

All visualisations should have meaningful main titles, correctly labelled axes. The axes titles and coordinates should be readable as they are, without requiring any zoom in or out.

If you are citing lecture notes or any external piece of work, cite them in text and add references to them in a separate section at the end of your notebook.

Note: Be clear and concise.

Demonstration and Presentation (20%)

Finally, following the submission deadline, you will be asked to spend 10-15 minutes to present your coursework using your programme scripts and visualisations to the Course Team.

Further details on presentation logistics will be communicated just after the deadline of the first 80% assessment of this coursework.

The demonstration is used to test that you:

a) understand your code and

b) can explain in detail the algorithms utilised.

Useful hints

A well-integrated Task D can be very useful while you are demonstrating.

Marking Guidelines

	80-100	70-79	60-69	50-59	40-49	30-39	0-29
	Exceptional First	**First**	**Upper 2nd**	**Lower 2nd**	**Third**	**Narrow Fail**	**Fail**
Analysis and Methods outline	Professional outline of analysis and methods used.	Detailed purpose of analysis and methods provided.	Adequate outline of analysis and methods provided.	Outline of analysis and methods provided but with some flaws.	Simple outline of analysis and methods provided, but lacking key detail.	Inadequate outline of analysis and methods provided.	No outline of analysis or methods provided.
Data pre-processing	Sophisticated pre-processing of data.	Comprehensive pre-processing of data.	Adequate pre-processing of data.	Pre-processing of data is attempted but with some flaws.	Limited pre-processing of data.	Inadequate pre-processing of data.	No pre-processing of data.
Key results and correctness of content	Unanticipated results and implementations presented. Appropriate, substantial, correct and sophisticated nature.	Comprehensive results and implementations, presented and employed well. Appropriate, substantial and correct.	Expected results and implementations presented. All appropriate, largely correct, with few flaws.	Not all expected results and implementations presented. All appropriate, largely correct, with few flaws	Few or simple results and implementations presented. Much appropriate material, but flawed.	Seriously flawed results or no implementation. Appropriate but seriously flawed material.	No results or implementation. Incorrect or inappropriate content.
Conclusions	Deep and critical understanding provided.	Thorough understanding shown.	Good understanding shown.	Key concepts generally understood.	Some evidence of understanding.	Little of superficial understanding shown.	No evidence of understanding.
Report	Like a publishable report, virtually error-free.	Like a publishable report with isolated minor errors.	Can be followed easily with very few errors.	Can be followed easily with some weaknesses.	Can be followed with difficulty.	Poor structure or containing significant errors.	Unstructured and with many errors.
Demonstration	Able to execute and explain the program clearly. Demonstrates an excellent level of understanding and explanation.	Able to execute and explain the program with no aid or guidance. Has a good level of understanding.	Able to explain the program with minor errors. Has a good but not complete understanding of the code.	Able to explain the program with some errors. Has some understanding of the code.	Able to explain the program with major errors. Makes an effort to explain the program but unable to do so.	Unable to explain in any detail the program that has been created Little awareness of the tasks or sections of code.	Unable to explain the program at all. No awareness of the purpose / location of any set tasks or sections of code.