Manchester Metropolitan University

Department of Computing and Mathematics

ASSESSMENT COVER SHEET 2023/24

Unit Code and Title:	6G7V0015 Machine Learning Concepts
Assessment Set By:	L Gerber
Assessment ID:	1CWK100
Assessment Weighting:	100%
Assessment Title:	Machine Learning Project
Type:	Individual
Hand-In Deadline:	See Moodle
Hand-In Format and Mechanism:	Moodle (.zip file with .pdf and .ipynb)

Learning outcomes being assessed:

LO1 Critically analyse a given problem and build predictive models to gain insights from real-world data.

LO2 Report on predictive outcomes and critically assess strengths and drawbacks of various predictive modelling techniques.

LO3 Apply a wide range of transferable skills and attributes applicable to industry and research.

Note: it is your responsibility to make sure that your work is complete and available for marking by the deadline. Make sure that you have followed the submission instructions carefully, and your work is submitted in the correct format, using the correct hand-in mechanism (e.g., Moodle upload). If submitting via Moodle, you are advised to check your work after upload, to make sure it has uploaded properly. If submitting via OneDrive, ensure that your tutors have access to the work. Do not alter your work after the deadline. You should make at least one full backup copy of your work.

Penalties for late submission

The timeliness of submissions is strictly monitored and enforced.

All coursework has a late submission window of 7 calendar days, but any work submitted within the late window will be capped at 50%, unless you have an agreed extension. Work submitted after the 7-day late window will be capped at zero unless you have an agreed extension. See ‘Assessment Mitigation’ below for further information on extensions.

Please note that individual tutors are unable to grant extensions to assessments.

Assessment Mitigation

If there is a valid reason why you are unable to submit your assessment by the deadline you may apply for assessment mitigation. There are two types of mitigation you can apply for via the unit area on Moodle (in the 'Assessments' block on the right-hand side of the page):

- **Self-certification**: does **not** require you to submit evidence. It allows you to add a short extension to a deadline. This is not available for event-based assessments such as in-class tests, presentations, interviews, etc. You can apply for this extension during the assessment weeks, and the request must be made **before** the submission deadline.

- **Evidenced extensions:** requires you to provide independent evidence of a situation which has impacted you. Allows you to apply for a longer extension and is available for event-based assessment such as in-class test, presentations, interviews, etc. For event-based assessments, the normal outcome is that the assessment will be deferred to the Summer resit period.

Further information about Assessment Mitigation is available on the dedicated Assessments page: https://www.mmu.ac.uk/student-life/course/assessments#ai-69991-0

Plagiarism

Plagiarism is the unacknowledged representation of another person's work, or use of their ideas, as one's own. Manchester Metropolitan University takes care to detect plagiarism, employs plagiarism detection software, and imposes severe penalties, as outlined in the Student Code of Conduct and Regulations for Postgraduate Programmes. Poor referencing or submitting the wrong assignment may still be treated as plagiarism. If in doubt, seek advice from your tutor.

As part of a plagiarism check, you may be asked to attend a meeting with the Unit Leader, or another member of the unit delivery team, where you will be asked to explain your work (e.g. explain the code in a programming assignment). If you are called to one of these meetings, it is very important that you attend.

If you are unable to upload your work to Moodle

If you have problems submitting your work through Moodle, you can email it to the Assessment Team's Contingency Submission Inbox using the email address submit@mmu.ac.uk. You should say in your email which unit the work is for, and provide the name of the Unit Leader. The Assessment team will then forward your work to the appropriate person. If you use this submission method, your work must be emailed **before the published deadline**, or it will be logged as a late submission. Alternatively, you can save your work into a single zip folder then upload the zip folder to your university OneDrive and submit a Word document to Moodle which includes a link to the folder. **It is your responsibility to make sure you share the OneDrive folder with the Unit Leader, or it will not be possible to mark your work.**

Assessment Regulations

For further information see Assessment Regulations for Undergraduate/Postgraduate Programmes of Study on the Student Life web pages.

Formative Feedback:	Formative feedback on provisional work will be given primarily verbally in lab sessions.
Summative Feedback:	Summative feedback is provided with a breakdown of marks for the assessment criteria and a general feedback document via Moodle.

Coursework Specification (1CWK100)

6G7V0015 Machine Learning Concepts

Luciano Gerber

23/24, Semester 1

Overview

In this assignment students have the opportunity to consolidate their learning in the Machine Learning Concepts unit by creating a **working solution** for a **real-world, industry-based machine learning problem**. It involves typical tasks such as data **understanding**, **exploration**, **cleaning**, and **preparation**; and machine learning model **selection**, **tuning**, **fitting**, **evaluation**, and **analysis**.

More specifically, you will work with a **Car Sale Adverts dataset** provided by **AutoTrader (AT)**, one of our industry partners. The dataset contains an anonymised collection of adverts with information on vehicles such as brand, type, colour, mileage, as well as the selling price. Your main task is to produce a **regression** model for **predicting the selling price** given the characteristics of the cars in the historical data given. This type of task is similar to that addressed by AT's machine learning engineers and data scientists in the implementation of the price indicator feature in AT's website.

You are expected to produce, as **deliverable**, a **brief, structured report** containing **snippets of code** and **explanations** that address the set of machine learning tasks defined here. Also, you must hand in a single **fully-documented**, **reproducible Python notebook** with the **rough work** that forms that basis of your report.

Obtaining and Exploring the Dataset

Please note the following **license information** before downloading and using the dataset:

- *this dataset provided by AutoTrader is to be used only by students enrolled on 6G7V0015 (23/24). Students can place it in their personal storage spaces (e.g., MMU OneDrive) for carrying out their practical data science and machine learning work (e.g., with Jupyter Lab). Please do not redistribute the dataset.*

It is available in our Moodle area here as a `.csv` file, and it has around 400K rows.

Machine Learning Tasks to Perform

In order to address the main task of producing a regression model for price prediction, you need to implement and run the following tasks that are typical of a Machine Learning pipeline, using our usual environment (i.e., Python ecosystem/notebooks):

1. **Data/Domain Understanding and Exploration** (e.g., load the data, sample observations, check correct parsing of data, identify quantitative and qualitative features, analyse data distributions (e.g., range, centrality, dispersion, shape), identify good predictors, process the data for visualisation and exploration).
2. **Data Processing for Machine Learning** (e.g., detect and deal with noise (i.e., erroneous values), missing values, and outliers; subset, reshape, and engineer features for machine learning purposes; categorically-encode, rescale data; split data into predictors and target; obtain train/validation/test folds).
3. **Model Building** (e.g., choose suitable algorithm(s), fit and tune models; grid-search, rank, and select model(s) on based on evaluation metrics and under/overfit trade-off).
4. **Model Evaluation and Analysis** (e.g., evaluate selected model(s) with score/loss metrics with cross-validation, analyse actual vs predicted plot, gain and discuss insights based on feature importance, analyse individual predictions and distribution of scores/losses together with predictors; for an interpretable model, briefly discuss about its structure).

Marking Criteria

The **assessment criteria** is based on the University's PGT Assessment Criteria and stepped marking, and includes aspects of code/explanations such as:

- clarity, conciseness, style, correctness;
- usefulness, challenge, creativity, initiative;
- efficiency, reusability, and generality.

Report Components and Weights

Please structure your report with the components below including, for each, a small but representative and interesting portion of your work in completing the tasks above. These should consist of a relevant fragment of code, a corresponding output (e.g., table, plot), and a brief explanation (one or two paragraphs) of findings. Next to each subcomponent below you will see a range (e.g., for *Dealing with Missing Values, Outliers, and Noise*, it is *2-3*) which indicates the suggest number of selecteed, interesting examples to report on.

Component	Weight	Limit
1. **Data/Domain Understanding and Exploration**	**30%**	
1.1. Meaning and Type of Features; Analysis of Distributions (3-4)	10%	(1 page)
1.2. Analysis of Predictive Power of Features (2-3)	10%	(1 page)
1.3. Data Processing for Data Exploration and Visualisation (2-3)	10%	(1 page)
2. **Data Processing for Machine Learning**	**20%**	
2.1. Dealing with Missing Values, Outliers, and Noise (2-3)	10%	(1 page)
2.2. Feature Engineering, Data Transformations, Feature Selection (2-3)	10%	(1 page)
3. **Model Building**	**20%**	
3.1. Algorithm Selection, Model Instantiation and Configuration (3)	10%	(1 page)
3.2. Grid Search, and Model Ranking and Selection (3)	10%	(1 page)
4. **Model Evaluation and Analysis**	**30%**	
4.1. Coarse-Grained Evaluation/Analysis (3) (e.g., with model scores)	10%	(1 page)
4.2. Feature Importance (2-4)	10%	(1 page)
4.3. Fine-Grained Evaluation (2-4) (e.g., with instance-level errors)	10%	(1 page)

In model building and evaluation/analysis, it is **required** that three main classes of machine learning models covered in the unit - **kNN**, **Decision Tree**, and **Linear regressors** - are employed and reported on.

Submission

Your submission should be a `.zip` file containing (1) a PDF for your report and (2) a `.ipynb` **documented**, **reproducible Python notebook** as **rough work**. There is no need to submit the dataset, but it is expected that one is be able to reproduce your work on the copy of the dataset available on Moodle. You will find the submission link in our unit's Moodle area. Please be careful **not to leave it to the last minute**; internet issues and similar are unlikely to count as exceptional factors.

Your report should contain your name and student ID. It does not need introduction, conclusion, or extra prose; simply focus on the structure shown above and add corresponding sections with content as suggested.

How to Pass This Assignment

One pathway (narrow/deep) to obtaining at least a pass in the assignment would be to focus on and do a reasonable job on components 1 and 2 and a genuine start at 3. Say that you obtain 72% for (1), 72% for (2), 52% for (3), and 22% for (4) - that would result in a combined mark of 53%.

Another (broad/shallow) is to have a genuine, but limited attempt at each component. One possible scenario would be 62% for (1), 62% for (2), 52% for (3), and 52% for (4) - that would give you 57%.

Regulations and Code of Conduct

Please make sure you are familiar with the Taught Postgraduate Assessment Regulations. As mentioned in the induction week, the pass mark is 50%, and one has a single reassessment opportunity (capped at 50%). If there are mitigating factors, please visit and follow the guidance at Exceptional Factors, such as that on self-certification.

Importantly, please also make sure you are fully aware of the regulations on Academic Misconduct, particularly if this is your first experience with the UK's higher-education system.

One important aspect to highlight is that this is an **individual assignment**, and the **submission has to be your own**. It is absolutely fine for people to work in groups and collaborate. It is also fine to **be inspired** by existing code snippets created by colleagues, contributors at StackOverflow, and **ChatGPT**. But, importantly, you have to **own it** in the end. That is, **you must be able to explain, customise, and apply it** in a similar, but separate context; also, **cite/reference** the sources. If in doubt, question yourself whether what the help you are receiving is "advice" or simply "do-it-for-me"; or whether you are simply "copy-and-pasting" or "owning" a code snipped.

Some examples of scenarios of when things **are not fine** are:

- *the representation of another person's work, without acknowledgement of the source, as one's own*: Say, student A wishes to help and shares their solution to the assignment with student B. The latter submits what was shared as their own work (fully or partially, identically or with little modification). This characterises **collusion** and would implicate both A and B.
- *the use of third parties and/or websites to attempt to buy assessments or answers to questions set*: this characterises **contract cheating**.

All cases of Academic Misconduct (e.g., plagiarism) will be reported to and investigated by Student Case Management Team.

Vivas

We will hold vivas (i.e., presentations with Q/A) for selected submissions - for example, when we need further clarification on the work described, when we have not been able to fully align the rough work with the report, when students fail to be in the labs carrying out the provisional work, or when we simply need to establish that the work submitted is the student's own. The performance on the viva will be used to confirm or adjust the provisional marks obtained on the submitted work.

Tips

- For **reducing computational time** and **cognitive load**, specially during prototyping/experimental phases, you might want to:
  - Avoid plotting a large number of individual data points; you could turn to heatmaps/hexbins instead of scatterplots, for example.
  - Take/work with a smaller sample of the dataset.
- When dealing with high-cardinality categorical features, at initial stages, you might want to reduce the number of categories before encoding, plotting, among others (say, keep the most frequent ones).
- An iterative approach with gradual expansion of scope and complexity is always recommended. For example, you might restrict your analysis initially to subsets of the data to features and regions of the data that are easier to make sense of.
- Favour things that can be automated and have a greater impact in the quality of the work. We wouldn't like to see you spending an inordinate amount of time in data cleaning, for example.