Assessment Information Coursework 2023-24 v1.0

Module name: Machine Learning with Applications in Finance

Module code: COMP0198

Module leader name(s): Nikhil Vadgama, Carolyn Phelan

Teaching assistant: Walter Hernandez

Academic year: 2023/24

Term 1, 2 or 3: Term 1

Type of assessment: Coursework assignment

Nature of assessment: Individual

Content of this Assessment Brief

Section A: Core information

This assessment is marked out of:	120 marks
% weighting of this assessment within the total module mark	80%
Word count/number of pages - maximum	4000 words and 15 pages (this only applies to the written report). There is no standard penalty for using fewer words/pages, although a significantly shorter report is unlikely to contain sufficient detail to gain a good mark.
Determining word count impacted by Turnitin	• After submission to Turnitin, the Turnitin recorded word count is usually higher than the word count in a Word/pdf document. • Where the assessment brief specifies a maximum word count, on the front cover of your submission, record the number of words as recorded in your Word document. • It is the Word/pdf document word count which will be considered in marking, NOT the Turnitin word count.
Footnotes, appendices, tables, figures, diagrams, charts included in/excluded from word count/page length?	Any footnotes and appendices are included in the word count and page limit.
Bibliographies and reference lists included in/excluded from word count?	The title page, table of contents, bibliography and jupyter notebook (if uploaded), are excluded from the word count and page limit.
Penalty for exceeding specified word count/page length?	• Where there is a specified word count/page length, and this is exceeded, there is a penalty: 10 percentage points deduction, capped at 40% for Levels 4,5, 6, and 50% for Level 7. Refer to Academic Manual Section 3: Module Assessment - 3.13 Word Counts. • No penalty applies when there is no specified word count/page length.
Requirements for/use of references	• The main object of the coursework is to assess whether you have engaged fully with module teaching material and any supporting material which is more widely available such as books and papers. • You must therefore cite sources appropriately and go into detail about how they are relevant to the work you are doing. It is not sufficient to say, for example, that SVMs are described in lecture 5. You should be specific about the particular slides which you are referring to. This is also the

	case where you are using example code which has been shown in the tutorial classes. • You may draw upon various sources, including, illustratively, journal articles, other textbooks, and industry reports. • As appropriate, you may draw upon course materials – lecture slides, notes, handouts, readings, textbook(s) - you engaged with in studying this module. • Unless citing content specifically (e.g. a quote from a book), you should not be copying word for word from lecture slides, notes, handouts, readings, or textbook(s). • You should capture, articulate and communicate your views, thoughts and learning in your own words. • If you provide quotes from any lecture slides, notes, handouts, readings, or textbook(s), you should cite them and provide references in the usual way. • Be aware that a number of academic misconduct checks, including the use of Turnitin, are available to your module leader. • If required/where appropriate, UCL Academic Misconduct penalties may be applied (see immediately below).
Artificial Intelligence (AI) category	• AI may be used in an **assistive** role which means you are permitted to use AI tools to support the development of specific skills required for the assessment. You should refer to the UCL guidance on acknowledging use of AI and referencing AI. Failure to correctly reference use of AI in assessments may result in you being reported via the Academic Misconduct procedure. Refer to the section of the UCL Assessment success guide on Engaging with AI in your education and assessment.
Academic misconduct (including plagiarism)	• Academic integrity is paramount. • It is expected that your submission and content will be your work with no academic misconduct. • Academic Misconduct is any action or attempted action, including collusion with other students, that may result in a student obtaining an unfair academic advantage. There are severe penalties for Academic Misconduct, including exclusion from UCL where appropriate and required. • Refer to Academic Manual Section 9: Student Academic Misconduct Procedure - 9.2 Definitions.
Submission date	**Friday 15th December 2023**
Submission time	**16:00 (UK time)**
Penalty for late submission?	Yes. Standard UCL penalties apply. Students should refer to https://www.ucl.ac.uk/academic-manual/chapters/chapter-4-assessment-framework-taught-programmes/section-3-module-assessment#3.12

Submitting your assignment	The assignment MUST be submitted to the module submission link located within this module's Moodle 'Assessment' tab by the specified deadline. Your submission must include: 1. Your written report submitted as one pdf file (with two sections, one for each prediction task)
Anonymity of identity. Normally, all submissions are anonymous unless the nature of the submission is such that anonymity is not appropriate, illustratively as in presentations or where minutes of group meetings are required as part of a group work submission	• Anonymity is required. • Your name should NOT appear anywhere on your submission.
Return and status of marked assignments	• All assignments aim to be returned within 4 working weeks from the submission date as per UCL guidelines. Still, we will endeavour to return it earlier than this if possible. In certain circumstances, it may take longer than the specified period. • Assessments are subject to appropriate double marking/scrutiny and internal quality inspection. All results, when first published, are provisional until the relevant External Examiner and the Examination Board confirm them. • No appeals regarding your published mark are available until the Examination Board confirms them. UCL regulations specify that academic judgment applied within the marking process cannot be challenged.

Section B: Coursework Brief and Requirements

We will be working with a dataset of credit card information to predict credit card payment default probability and limit balances (both a classification and regression task).

The dataset contains information on default payments, demographic factors, credit data, history of payment and bill statements of credit card customers in Taiwan from April 2005 to September 2005.

There are 25 variables and 30,000 rows of data.

The variables have the following meaning:

- X0: ID of the customer
- X1: Amount of the given credit (NT dollar): it includes individual consumer credit and family (supplementary) credit.
- X2: Gender (1 = male; 2 = female).
- X3: Education (1 = graduate school; 2 = university; 3 = high school; 4 = others).
- X4: Marital status (1 = married; 2 = single; 3 = others).
- X5: Age (in years).
- X6 - X11: History of past payments. The past monthly payment records (from April to September 2005) are as follows: X6 = the repayment status in September 2005; X7 = the repayment status in August 2005; . . .; X11 = the repayment status in April 2005. The measurement scale for the repayment status is: -2 = No consumption; -1 = pay duly; 0 = use of revolving credit (some repayment); 1 = payment delay for one month; 2 = payment delay for two months; . . .; 8 = payment delay for eight months; 9 = payment delay for nine months and above.
- X12-X17: Amount of bill statement (NT dollar). X12 = amount of bill statement in September 2005; X13 = amount of bill statement in August 2005; . . .; X17 = amount of bill statement in April 2005.
- X18-X23: Amount of previous payment (NT dollar). X18 = amount paid in September 2005; X19 = amount paid in August 2005; . . .; X23 = amount paid in April 2005.
- X24: The default on payment in the next month (Yes =1; No = 0)

Please note that the dataset has some features where not all the possible values are explicitly described above. Where you encounter this, please make sensible assumptions about what to do in these cases.

Your tasks are as follows:

1. Build a classification model that predicts whether or not a customer will default on their next payment
    a. Use Logistic Regression, SVM and a Random Forest model
2. Build a regression model that predicts a customer's limit balance if they were a new customer to the bank with only X0, X2-5 available to you as data.
    a. Use Linear Regression, Random Forest and an ANN model

Your answer should be submitted as one report with two sections (in prose and in one pdf file) of no more than fifteen (15) pages and 4000 (four-thousand) words in total (in total across both sections). Each prediction task should include at least the following information and give details of what you have done and why:

1. General analysis of the problem you are solving and what steps you will take to optimise your model [10 marks]
    a. Discussion of the features
    b. Discussion of the number of features
    c. Discussion of the feasibility of the task
    d. Discussion of the approach you will take
2. Data exploration and feature engineering [10 marks]
    a. Distributions of data
    b. Outliers
    c. Correlations
    d. Scaling
    e. Transformations
    f. Data balance
    g. New features that could be generated
3. Model training [10 marks]
    a. Performance measures
    b. Training/testing split
    c. Cross-validation
    d. Bias and Variance
4. Model optimisation [10 marks]
    a. Hyperparameter tuning
5. Conclusions [10 marks]
    a. Compare and contrast your algorithms' performance, identifying the best model and why?
    b. What additional steps could you take to improve your models?

10 marks are also available for being concise and for the general presentation and layout of the report and code, including using the seed value (for each report).

Within the report (as a single pdf file), please add the code used to solve for each part. This should be annotated with a brief explanation of your steps. This is important because marks for each section will be considered across the written component of the report and your code where appropriate.

Wherever there is a random component in your modelling, please use a random seed value of 33 (thirty-three).

Explicitly state why you have finally chosen whichever parameters for a model and hyperparameters when tuned.

The total number of marks available is 120 (60 for each prediction task).

The dataset is available at: https://archive.ics.uci.edu/ml/machine-learning-databases/00350/default%20of%20credit%20card%20clients.xls

You may wish to read and look at a paper that also analysed this dataset which is available at: https://www.sciencedirect.com/science/article/pii/S0957417407006719

Section C: Assessment criteria

Within each section of this coursework, you may be assessed on the following aspects as applicable and appropriate to this particular assessment. You should thus consider these aspects when fulfilling the requirements of each section:

- The accuracy of any calculations;
- The strengths and quality of your overall analysis and evaluation;
- Appropriate use of relevant theoretical models, concepts and frameworks;
- The rationale and evidence that you provide in support of your arguments;
- The credibility and viability of the evidenced conclusions/recommendations/plans of action you put forward;
- The code you use and structure;
- Structure and coherence of your considerations and reports;
- With all of the above, we are looking for evidence that you have engaged fully with the taught materials and the supporting sources of information. Therefore, we ask that you refer to the taught materials (and other sources) where relevant to the work that you are doing. This should be specific to particular aspects of the material: for example, it is not sufficient to say that SVMs are taught in lecture 5. Rather the particular aspects of SVMs that you are working with should be linked to the place that they are discussed in the slides and other supporting materials;
- We are keen to see code snippets linked to particular aspects of the question included in the report. Note that they should be specific to the issue that you are describing, for example in Q2.b above you should include just the lines that are specifically linked to the reduction of outliers.
- As and where required, relevant and appropriate, any references should use either the Harvard OR Vancouver referencing system (see References, Citations and Avoiding Plagiarism)
- Academic judgement regarding the blend of scope, thrust and communication of ideas, contentions, evidence, knowledge, arguments, conclusions.
- Each part has requirements with allocated marks, maximum word count limits/page limits and where applicable, templates that are required to be used.

You are advised to refer to the UCL Assessment Criteria Guidelines, located at https://www.ucl.ac.uk/teaching-learning/sites/teaching-learning/files/migrated-files/UCL_Assessment_Criteria_Guide.pdf

Please note that a mark of 70 is in the distinction/1st class zone, meaning that the work is excellent, and excellence needs to be rare to be meaningful. Academic marking scales are counter-intuitive to many as they have 30 marks available above the threshold of excellence. This is so that rare, very rare and exceptionally rare degrees of outstanding performance can be recognised and distinguished. It does not mean that 30 marks have been subtracted for errors. The marking scheme that will be used is also available below.

Grade	No marks (0)	Very poor (1)	Poor (2)	Needs more work (2.25)	Satisfactory (2.5)	Very Satisfactory (2.75)
%	0%	20%	40%	45%	50%	55%
Writing (75%)	Either required content is missing, or zero has been awarded for irregularity(ies).	It appears that what was required has been misunderstood. The arguments are either copied from the lecture notes with no analysis or not related to the problems that need to be solved.	Significant errors/omissions/ unclear points. Much more work is needed to develop and support assertions.	Generally, arguments need to be strengthened much more. Some points on the obvious issues but quite a few errors and omissions. The writing is a very general treatment of the area and only partially relates to the problem.	A few good arguments on the obvious issues, but weak on relevance to the problems and models being used and support from theories/concepts learnt. The writing is very general and although it relates to the problem it does not specifically indicate how the solution relates to the taught and supporting content.	Arguments are generally along the right lines but a little weak in places. Scope to increase depth and breadth of analysis in many instances. Quite a few key points were missed. The writing is quite general, and it only relates to the problem in a few places. In most places it does not specifically indicate how the solution relates to the taught and supporting content.
Code (25%)	No code has been written	The code that has been produced misses the brief entirely and does not address the questions.	Some of the code produced addresses the problem, but major chunks are missing and major issues in getting the code to run.	Some of the points have been addressed in the code, but there are still some missing points, and the code needs minor adjustments to run.	Most of the points have been addressed in the code, but there are still some missing points, or the code needs minor adjustments to run.	The code is operational and addresses the problem. However, the programming style is very poor, with major repetitions and inefficiencies. Some of the built-in python functions have been used in a non-optimal way.

Grade	Moderate (3)	Good (3.25)	Very good (3.5)	Excellent (3.75)	Outstanding (4)	Publishable standard (5)
%	60%	65%	70%	75%	80%	100%
Writing (75%)	A solid analysis that covers many of the key reasons why and how methods are used and algorithms applied, although there is scope for further analysis in some places. A few key points were missed. The writing directly relates to the problem in most places. It is fairly specific in most places as to how the solution relates to the taught and supporting content. However, this is missing in a few places.	A good solid analysis which covers the key issues of how and why decisions are taken in your solution. There is some scope to increase the breadth and depth of analysis in a few places. The writing directly relates to the problem in almost all. It is fairly specific in most places as to how the solution relates to the taught and supporting content. However, this is missing in a few very minor places.	Very good points, with well-supported arguments which relate clearly to decisions made to solve the problem. The writing directly relates to the problem in all places. It is very specific in as to how the solution relates to the taught and supporting content.	Excellent arguments. Well-justified assertions that demonstrate depth and breadth of insight on how to solve the problems. The writing directly relates to the problem in all places. It is very specific in as to how the solution relates to the taught and supporting content and it is clear that the student has engaged with the majority of the course content.	Outstanding work. Demonstrates originality, breadth and depth of thought and attention to detail. Industry-standard work. Close to being publishable work. The writing directly relates to the problem in all places. It is very specific in as to how the solution relates to the taught and supporting content and it is clear that the student has engaged with all of the course content.	Exceptional work. The depth and breadth of points made are of publishable standard. The writing directly relates to the problem in all places. It is very specific in as to how the solution relates to the taught and supporting content and it is clear that the student has engaged with all of the course content.
Code (25%)	The code is operational and addresses the problem. Some attempts have been made to optimise the code, such as using functions, minimising loops, etc. The built-in python functions are used satisfactorily.	The code is operational and addresses the problem. Significant efforts have been made to optimise the code using functions, minimisation of loops, etc. A more sophisticated understanding of the built-in Python functions is shown	The code is operational and addresses the problem. The code is very well optimised within the procedural programming paradigm. A deep and detailed understanding of the built-in Python functions is shown	The code is operational and addresses the problem. The programming style is more sophisticated and shows a knowledge of the Object orientated paradigm. A deep and detailed understanding of the built-in Python functions is shown	The code is operational and addresses the problem. The programming style is more sophisticated and is well structured within the Object orientated paradigm. A deep and detailed understanding of the built-in Python functions is shown. Possibly some test functions are included.	The code is operational and addresses the problem. The programming style is at the level of a professional developer.

This and any following pages are left intentionally blank.