Assignment Specification: Study, Implement and Present a Machine Learning Model, Presentation and Peer Review

1 Introduction

The aim of Assignments A2 and A3 is to strengthen both your practical proficiency in machine learning and your grasp of the technical intricacies underlying the learning process. You will be expected to:

- perform a machine learning task, and
- present your project outcomes, demonstrating readiness to engage with related questions.

2 Specification of A2: Study, Implement and Present a Machine Learning Model

There are generally two categories of acceptable tasks for this assignment, based on the primary type of effort required. This classification is intended to guide your focus during the project. While you may define your effort as a mixed type, you will not be penalised for crossing boundaries between categories.

Option 1: Study a fundamental machine learning model. The emphasis here is on theoretical foundations, including concepts and techniques such as hypothesis spaces, learning algorithms, and loss function design. To complete a project of this type, you must demonstrate a solid understanding of the technical details of a specific machine learning model. Your work should be grounded in the learning theory framework introduced in the subject.

Option 2: Build a machine learning system to address a practical challenge. The focus in this type is on applying your understanding of machine learning systems to real-world problems. *Alignment* and *practicality* are key. You are expected to design a learning-based system that solves a task involving real-world data—such as predicting stock prices or recognising patterns in images or text. The problem should be formulated in a computable way, and the learning objective (loss function) should be both computationally effective and practically meaningful.

Submission

Regardless of the category, an acceptable submission must include a computer program implementing the learning model, executable on a cloud-based computational platform.

Specifically, you will submit:

1. A computer program (link to a cloud-based implementation, see below)
2. A project journal (PDF), due 5 Oct 2025, 11:59 pm(Canvas upload), including link to a cloud-based implementation (e.g. `https://colab.research.google.com/`). The project must be self-contained, including environment setup and data downloading and pre-processing. If you are seeking to make a resource-intensive project, you must ensure that your accessor can run it without additional setup (it may involve hiring facilities for a short period of time).
3. (optional) Presentation materials (e.g., slides or Jupyter notebooks)

If only the journal document is submitted, a single PDF file upload is sufficient. If additional materials are provided, they should be uploaded as a ZIP file within the limit of Canvas file size restrictions.

Use of AI tools is permitted with documentation and critical review in the journal.

Quality Assessment

The project will be assessed based on the following criteria:

- **Definition of the Learning Task:** Specify input/output for both training and deployment phases.
- **Knowledge of Model and Algorithm:** Explain key components and theory-to-code connection.
- **Evaluation and Improvement:** Define model behavior, loss function, and differences from practical objectives.

See later section 4 for detailed criteria.

3 Specification of A3: Presentation and Peer Review

In this assessment task, you will present your project to your peers and simultaneously act as an assessor for your classmates' projects. For each project you assess, you will evaluate the quality based on three criteria outlined below (refer to the A2 specification and the attached detailed criteria guidelines for more details).

In your presentation, you are expected to:

- explain the key components of the machine learning model;
- describe how these components are implemented as functions or objects in your code;
- discuss the main technical challenges encountered;
- evaluate the benefits and limitations of the chosen model.

You will rank each criterion as "Poor," "Fair," or "Excellent" and provide comments to justify your rankings. Additionally, you will receive rankings and feedback from your peers and a tutor for your own project.

3.1 Timeframe of Presentations

The peer presentation and assessment process comprises three stages. You will be assigned to a group of six students (see your group page on Canvas), and each group will present during an allotted time slot.

A. Preparation (about one week before presentation)

By this time, your group members submission materials, the Zoom meeting link, and your presentation time slot will be available on Canvas.

Action Items:

- Review the peer reports you have received.
- Prepare preliminary evaluations based on the A2 criteria—it’ s normal to have questions at this stage.
- Note any unclear aspects and prepare questions for the presentation session.
- Do not discuss your evaluations with peers before the presentation. The tutor will moderate to ensure impartiality.

B. Presentation Day

On presentation day:

- Join the designated Zoom session at your scheduled time.
- Verify your identity (student ID or equivalent).
- Deliver your A2 project presentation and respond to questions from peers and the tutor.
- Provide constructive feedback and ask questions during other presentations.

Important Notes:

- Clear, effective communication is essential. Refer to the assessment criteria below.
- Within 24 hours of your presentation, post your observations and feedback (comments only, NOT scores/rankings) on your group page by replying to threads posted by the presentation chair. Again, do **NOT** post evaluation grades. Timely completion of this step contributes to your A3 grade.

C. Post-Presentation and Grading

After presentations:

- Review the peer and tutor feedback on your project.
- Submit your final peer evaluations as a PDF on Canvas AFTER 24 hours (this is to allow other students to post their comments and fulfil A3-criteria). A template will be available but not required—just include all requested information.

Procedure of the Presentation Sessions

Schedule

- **Duration:** Each presenter has 15 minutes during the peer-review, which includes a 5-minute presentation and a 10-minute question-and-answer period.
- **Overtime Policy:** Presentations exceeding 5 minutes will be considered as overtime and will result in reduction of A3 grade (see criteria below). The session chair will end the presentation at 7 minutes if necessary, in order to ensure meaningful Q&A time within the schedule.

Q&A Session

- **Purpose:** Peers will ask questions to:
    1. Clarify relevant points about the project.
    2. Assess how the project aligns with the A2 criteria.
    3. Determine whether the author demonstrates genuine effort, understanding, and critical thinking.
- **Duration:** This session will last approximately 10 minutes, depending on the number of questions and presentation length.
- **Activities for All Students:**
    - **Take Notes:** Document key points and insights from the presentation and Q&A to adjust your tentative evaluations of the A2 criteria. This is the best time to identify whether the project reflects the author's own understanding and effort or merely compiles external materials.
    - **Assess Presentation Skills:** Evaluate the presenter's ability to communicate effectively and handle questions (refer to A3 Criteria below).
    - **Observe Peer Engagement:** Note the activity levels of other students, such as asking relevant questions and providing constructive feedback.
- **Timing:** The Q&A session ends 15 minutes after the presentation starts. In case of an overtime presentation, the Q&A session will end 17 minutes after the presentation starts, including any overtime.

3.2 Submission

This is a recap of 3.1.C. There are two submission components for A3:

- **Peer Evaluation:** Within 24 hours after the presentation provide feedback on your peers' projects in group discussions on Canvas.
- **Final Peer Evaluation:** After 24 hours of the presentation, you will submit your peer evaluations as a PDF on Canvas.

3.3 Quality Assessment

Your A3 grade will be based on the following criteria:

- **Clarity of Presentation:** Clear structure and effective visual communication that helps demonstrate your understanding of the solution, as well as the project's meeting the A2 criteria.
- **Question Handling:** Demonstration of sufficient familiarity with the project to answer questions effectively.
- **Peer Interaction:** Active participation in peer-review discussions, contributing to the discussion by asking relevant questions and providing constructive feedback.

The criteria are detailed below (see section 5).

Clarifications

- The program must be self-contained, including environment setup and data processing
- Offline demo must match an online archived version (e.g., GitHub, Colab)
- Option 1 requires algorithm implementation, not just use of libraries
- Option 2 must use real, non-trivial datasets (no toy datasets like Iris or Titanic)
- Proper validation scheme is required

4 Criteria for A2

A: Task Definition You need to briefly explain the objective of your task. More importantly, you must clearly define the interface of information exchange between your system and the environment or user: what constitutes valid input, and what the expected output format is.

A higher standard involves convincingly articulating how the input received by the system and the output it produces are related to the defined task.

The Table 1 below illustrates three stands of descriptions for system **input/output**—Acceptable (A), Excellent, and Poor:

IO	**Acceptable** (A)	**Excellent**	**Poor**
I	Image data represented as a 2D matrix of pixels, each with 3 color channels.	[A] + The data is preprocessed: pixel values in each image are normalized to the range [0,1] based on their per-image min/max. The task is image classification. The images contain photographs of handwritten digits from 0–9. Most images satisfy this expectation, but not all.	Image
I	A string, e.g., "to be or not to be".	[A] + Input strings are typically 50–200 characters long, with a maximum of 512. The task is sentiment classification. Input strings are sentences in some language. A rule-based filter selects English sentences, but may allow nonsensical text. Many inputs contain statements expressing judgments (e.g., product reviews), but clarity is not guaranteed.	User review
I	A 500-dimensional vector of real numbers	[A] + Each vector represents the past 500 days of closing prices for a specific stock. Missing data is filled with the last available value. The closing price is defined as the final transaction price per [cite].	Stock price information
O	A real-valued score: the higher the value, the more likely the model thinks the input is positive.	[A] + Let z be the output; the probability that the sample is positive is $1/(1+e^{-z})$.	Classification
O	The likelihood and bounding box of a target object around 196 reference locations in an image.	[A] + Each detection is a 205-dimensional vector: the first value indicates objectness; the next 4 give bounding box size, aspect ratio, and x/y offset; the final 200 values represent class probabilities.	Object detection result

Table 1: Specification of Task Input and Output

B: Explanation of Model, Algorithm, Key Components, and System Structure You need to demonstrate a deep understanding of the core computational procedures and components of the machine learning model you studied or used. Emphasize how theoretical algorithms are mapped to implementation in code and how different parts of your system work together to achieve the task.

- **System Overview:** Provide a high-level summary of the data flow or computation pipeline (e.g., data preprocessing, training routine, and model deployment). Keep it concise yet complete. Show understanding of ML fundamentals (for Option 1) or of system design and feasibility (for Option 2).
- **Key Components:** Explain the design choices, implementation, and motivation behind core modules (e.g., data-model alignment, model architecture, optimizer).
- **Theory-to-Code:** Describe how mathematical formulas are implemented as functions or classes in code—such as loss functions, forward/backward propagation.
- **Implementation Details and Challenges:** Discuss framework choices, hyperparameter tuning, performance optimizations, and their impact on model behavior.

The Table 2 below provides examples of **component-level descriptions**—Acceptable (A), Excellent, and Poor:

Component	**Acceptable** (A)	**Excellent**	**Poor**
H	The model is a 3-layer fully connected neural network with output dimensions 128, 128, and 10.	[A] + The network uses ReLU activations; skip connections exist between the two 128-d layers. Parameters are initialized via Xavier normal distribution.	Neural network
H	Decision tree using information gain maximization, where information gain is computed based on entropy ...	[A] + Tree structure defined recursively as ...; key logic for decision traversal is ...; line X of the code implements recursive return; no explicit loss is defined during training (information gain is used).	Decision tree
L	Loss function is mean squared error (MSE), $\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$; same metric used during testing.	[A] + Outliers caused problems with variance; I experimented with absolute error. However, gradient issues arose with $\lvert x \rvert$ near 0 ...	Mean squared error
A	Used gradient descent: $\theta_{t+1} = \theta_t - \eta \nabla L(\theta_t)$, where η is the learning rate.	[A] + Used Adam optimizer with update rule: $\theta_{t+1} = \theta_t - \eta \frac{m_t}{\sqrt{v_t}+\epsilon}$. m_t, v_t are moment estimates; ϵ prevents divide-by-zero. Compared Adam and vanilla GD in experiments. Adam was implemented from scratch ...	Gradient descent

Table 2: Specification of Model Components

C: Model Evaluation and Refinement You must clearly articulate the model's intended behavior for your selected task. Revisit your choice of loss function—if it does not directly match the task's real objective, you should explain why this mismatch is tolerated during model construction (this discussion may also appear under section B).

In addition to the training loss, report any extra evaluation metrics used to ensure your model meets the task's requirements. Consider the following aspects:

- **Expected Model Behavior:** Describe the desired metrics (e.g., accuracy, recall, F1 score, MSE) and present comparisons on training/validation/test sets, with confidence intervals or statistical tests.
- **Loss Function vs. Task Objective:** Define the loss function and explain how it may differ from the true business goal (e.g., user satisfaction, IoU).
- **Discrepancy Analysis:** If loss does not fully capture the task objective, identify examples where this matters. Supplement with additional metrics.
- **Reflections and Future Work:** Critically assess your implementation and learning outcomes; outline potential improvements or next steps.

Table 3 illustrates **evaluation-level descriptions**—Acceptable (A), Excellent, and Poor:

Acceptable (A)	**Excellent**	**Poor**
Classifier achieves 0.88 accuracy on the test set. 5-fold cross-validation yields a mean accuracy of 0.86 with std = 0.02.	[A] + Train/val/test split was 70/20/10. Model achieved 0.87 accuracy on validation and 0.88 on test. Hyperparameter sweeps gave accuracies between 0.85–0.90, std = 0.03. Confusion matrix shows class A has recall 0.65, others >0.80. (Identifies future improvement.) No cross-validation due to compute cost. Next step: …	Model trained successfully

Table 3: Specification of Model Evaluation

5 Criteria Details for A3

5.1 Effective Communication

- **Clarity and Conciseness:** Present your project clearly and concisely, focusing on how it addresses the A2 criteria (refer to the A2 specification and attached grading guidelines).
- **Demo of Techniques:** Ensure that your presentation highlights a few key components at the technical level, clearly linked to the A2 criteria and reflecting *your efforts and understanding*. In contrast, a common example of *low-quality* project implementation and presentation is simply listing AI-generated content and explaining, like a map walkthrough, how it corresponds to the A2 criteria without demonstrating genuine work or insight.
- **Time Management:**
  - **Standard Duration:** 5 minutes.
  - **Overtime Policy:** Presentations exceeding 5 minutes but not surpassing 7 minutes are discouraged and will incur an overtime penalty -1 per minute (rounded up) — e.g. a 6:34 presentation will incur a -2 penalty.
  - **Enforcement:** The session chair will terminate the presentation at 7 minutes. The presenter will receive a -2 penalty, and also responsible for un-fulfilled presentation of A2, i.e. it is likely to cost A2 grade in the case of serious overtime.

5.2 Handling Questions

- **Response Quality:** Answer questions accurately and concisely, demonstrating a clear understanding of the topic.
- **Criterion Weighting:**
  - **Correctness:** 50% based on the accuracy and relevance of your responses.
  - **Conciseness:** 50% based on the brevity and clarity of your answers.
- **Expected Responses:**
  - **Adequate Response:** Providing a direct and accurate answer, even if it means acknowledging uncertainty (e.g., "I understand the question, but I need to research the exact details."), will receive 50% credit.
  - **Inadequate Response:** Offering lengthy, vague, or irrelevant answers that do not address the question will receive 0% credit.
- **Examples:**
  - *Question:* "You mention adding regularisation penalising complexity and claim it will help learning. Why?"
  - **Good Response 1:** "More complex models are more able to fit to data, including data containing unwanted relationships, i.e., noise. Enforcing simpler models reduces the influence of such data."

- **Good Response 2:** "I don ' t know. ChatGPT states so. I can ask for references to check the correctness or find out the origin of this idea."
- **Poor Response 1:** "Yes, the regularization technique we implemented helps prevent overfitting by adding a penalty for more complex models, aligning with our goal of enhancing generalization."
  (This is a tautological restatement of the question and lacks substantive reasoning.)
- **Poor Response 2:** "That's an interesting question. I think it relates to the model bias..."
  (This is an irrelevant discussion trying to avoid answering the question.)

5.3 Active and Constructive Peer Grading

- **Engagement:** Actively participate in the peer-grading process by thoughtfully evaluating your classmates' presentations.
- **Constructive Feedback:** Provide constructive and actionable feedback, highlighting both strengths and areas for improvement. Feedback must be provided in the group discussion page on Canvas within 24 hours after the end of the presentation session.
- **Criteria Alignment:** Ensure that your feedback is aligned with the established A2/A3 criteria, focusing on aspects such as communication effectiveness, question handling, and overall presentation skills.

5.4 Suggestions for Effective Communication

There are no unique optimal way to present your project, but here are some suggestions to help you prepare a concise and effective presentation:

- **Prepare the Presentation Concisely:** Since everyone in the session is part of the same class and already familiar with the context, one effective approach is to "head-first diving" into the core of your project. You may open directly with clarifying the inputs, the desired outputs, and how to evaluate discrepancies between the model outputs and the expected results. With the basic elements established, you can then proceed to explain the task and solution, focusing on the most relevant details.

  For example, you can start with:

  > The input data are of $3 \times 32 \times 32$ matrices, each entry is an integer in the range of 0 to 255, representing RGB values of image pixels. The output is a vector of 10 real numbers. That corresponds 10 classes, the number represents the model's confidence in the corresponding class. I studied how to compare the model's output with the expected output, which is one label, such as "cat", "dog", "car", etc.

  A counter-example of this approach can be:

> Image classification is an important application of machine learning. According to research by Commonwealth Scientific and Industrial Research Organisation (CSIRO), the related industry is worth $2.5 billion ...

- **Use Few Slides and Prepare Your Desktop and Project:** Use a few slides to highlight key points, such as the task, solution, and evaluation. Avoid over-decorating your slides. Focus on the essential aspects of your project that you want to convey to the audience.

  Practice your presentation to handle switching between slides and your implementation code smoothly. (This is means to find a good desktop arrangement, so you can present information in different application windows without losing focus – full screen model is not recommended, because it makes it difficult for audience to see your code and slides at the same time.)

  Another useful tip is to ensure that your project is ready on the cloud platform (e.g., Google Colab, Jupyter Notebook) before the presentation. Your project usually exists as an instance of a virtual machine. Each time the clould-based virtual machine is started, it will take some time to activate the execution kernel. Also, as required by the A2 specification, the project must be self-contained, including environment setup and data downloading and pre-processing. You may need to install libraries and download and pre-process datasets before actually demonstrating your project. Therefore, it is recommended to pre-run your project short time before the presentation. So the demo can fit in your total 15-minute presentation and Q&A time.

- **Avoid Exploratory Data Analysis (EDA):** Overloading your presentation with exploratory data analysis (EDA), and fancy-looking statistics generated by AI tools does NOT add value to either A2 or A3. On the contrary, it can distract from the main focus of your project.

  Worse, it can lead to a situation where you are unable to answer questions about the materials that were not directly related to your project but only for cosmetic purposes of your slides — unfortunately, you are responsible for all the content of your slides, regardless what they are meant for. Unexplained materials would lead to an assessment of misusing AI and damage your project grade.

- **Allocate Time to Key Points:** Dedicate about 2 minutes to highlight a particular aspect of your project that you truly invested effort in studying. Remember, half or more of your A3 grade is about convincing others that the project is YOUR work. It's essential to let reviewers firmly believe you invented it, designed it, and understand it deeply.

- **Engagement in Q&A:** For particularly interesting points in your project, leave a few "baits-for-questions" during your presentation. It's important to communicate how the project fits the criteria, and having questions from the audience can be more effective than making statements yourself.

  When answering questions, try to be concise and to the point, as discussed above in the examples. Again, "I don't know" is valid, informative (you are rewarding the asker for a critical question) and point-awarding. "White-house-press-secretary" style wastes time and may cost you points.

6 Useful Guide and Examples

Example 1 Task: Implementation and Study of a Simple Linear Regression Model. (Option-1)

Video Recording: `https://tinyurl.com/36sn3enm`

The presenter is "P3". From 27:30 to 43:00 (You are encouraged to watch other presentations to get an idea of how the assignment, the procedure remains similar).

Example Report (ChatGPT Conversation and Project Implementation Included) `https://tinyurl.com/4u4bjv7m`

Grade: A2 > 43/50 (HD Level), A3 > 17/20 (HD Level)

Example 2 Task: Building Medical Image Analytics Model (Option-2).

Video Recording: `https://tinyurl.com/bdz3xyt8`

The presenter is "P2". Starting from 17:30.

Example Report link (Contains Project Implementation Link) `https://tinyurl.com/5n85fzf8`

Grade: A2 > 40/50 (D-HD Level), A3 > 17/20 (HD Level)

7 A Weekly Plan to Build a Machine Learning Project and Journal Writing

7.1 Setup a Machine Learning Project

- Choose a focus: (1) study a model/method; or (2) address a practical application.
- Select a machine learning model (or a model from a previous project) as the object of study.
  - Understand the model's input and output format.
  - With AI assistance, try writing a "dummy" function that matches the signature of your selected model—meaning the input/output *types* match. For example, for a sentiment classification task, you might write:

    ```
    def classify_sentiment(text_sequence: List[str]) -> str:
        # find "happy" in text_sequence
        if "happy" in text_sequence:
            ret = "positive"
        else:
            ret = "negative"
        return ret
    ```

  - Clearly describe your required data format to the AI, search for widely accepted datasets, and download one locally.
  - Ask AI to help you write a data loading and preprocessing module that loads the dataset into memory and converts it to your expected format.
  - Test your dummy function with this data, even if the results are poor.

7.2 Understand the ML Framework: Hypothesis, Loss, Optimizer

- Ask AI to build a proper ML model to replace your dummy function.
- Map your understanding of the ML framework concepts—hypothesis, loss function, optimizer—onto the components of the model created by AI.
- Try identifying the code blocks that correspond to these components. Even if there's no one-to-one mapping, clarify your thinking and note any questions.

7.3 Study the Loss (Criterion)

- Consider your task (either from Section 1 or an updated idea). In the model AI helped you implement, are the training and evaluation criteria the same? If not, is the difference justified? If yes, is the chosen criterion appropriate? Can you change one of them while keeping the pipeline consistent?
- Does the evaluation criterion truly reflect the goals of the task? If not, propose and test a modification. Try to understand compromises in the original design.
- Even if the current evaluation criterion is reasonable, attempt to propose and justify an alternative.
- Try applying your new criterion during training. Is it feasible? Why or why not? Reflect on this.

7.4 Information-Theoretic Loss

- Consider this setup: your model solves a multi-class classification task (e.g., three or more categories):

$$\text{Data } X \xrightarrow{\text{Model } f} \text{Prediction } \hat{Q} \xleftrightarrow{\text{Criterion } L} \text{Multi-Class Label } Y \tag{1}$$

  What is the output format of Q? (Hint: it should contain one value per class.) Consider an example output $[q_1, q_2, q_3]$.

  What distinguishes the multi-class criterion $L^{\text{Multi-Class}}$ from, say, a multi-output regression criterion $L^{\text{Multi-Real}}$? (Hint: $L^{\text{Multi-Real}} = L(q_1) + L(q_2) + \dots$ assumes independent targets; a change in q_1 has fixed impact regardless of q_2, q_3 values. But in $L^{\text{Multi-Class}}$, a change in q_1 affects loss depending on q_2, q_3 due to the coupling required by interpreting f's output as a probability distribution.)

- Ask AI to help you implement the details of a multi-class classification loss criterion.
- Information theory provides tools for reasoning about the structure of probability distributions.
- Ask AI to explain how this loss implementation relates to information-theoretic notions like cross entropy and KL divergence. (Warning: AI may produce flawed logic—use critical thinking and document your questions.)

7.5 Model Family

- Describe your ML model as a *family of functions*. For the function family:
  - What are the input/output formats of the functions?
  - Are they parametrized? If so, what is the parameter space? If not, justify your view (you can use concrete examples).
  - Randomly sample 100 functions from this family.
- Evaluate their performance on a dataset (toy or real). Should you use training/test data? (Hint: is train/test distinction meaningful here?)
- Pick one function and describe its similarity/difference with the other 99. Who is closer to whom?

7.6 Model Family (continued)

- Continue with your 100 sampled functions. For a dataset with 10,000 examples, ask AI to compute a $100 \times 10{,}000$ matrix representing the prediction behavior.
- Analyze this matrix: look at inter-function differences. (Hint: relate to the previous question on proximity.)
- *Expand* your model family to increase diversity of prediction behavior over the dataset (not necessarily performance).
- *Reduce* your model family to increase concentration of prediction behavior.

7.7 Model Training

- Reflecting on the above—"random function," "100 samples," etc.—now try: i) randomly pick a function f; ii) randomly pick 100 nearby functions. Define and justify what "nearby" means.
- Select one of these 100 as f_{New} such that it performs better on your task.
- Given $f \to f_{\text{New}}$, ask AI to implement an iterative training process using repeated `Naive Update`.
- Ask AI to improve your `Naive Update` into a proper *optimizer*.
- Now you have all components of a typical ML pipeline. Implement the complete training process.

7.8 Project Construction and Reflection

- Consider replacing your model with a neural network. Ask AI for modern alternatives suited to your task.
- Compare your implementation with the modern solution. Identify technically interesting gaps or improvements.

- If you're working with neural networks, consider using the PyTorch stack. Ask AI to:
  - Modularize your codebase and build a standard project folder.
  - Help you write the final report section summarizing the purpose, method, results, and reflection.
  - Prepare a 5-minute presentation: slides, speaker notes, even a demo video or script. As long as **you** take responsibility for the Q&A, you'll retain your credibility.

You journal should be in a submission-ready state by now.

7.9 Project Promotion

- Extend your project as a foundation for future study, career, or commercialization.
- Ask AI to help you create a GitHub repository and upload your project.
- Ask AI to help you build a project website and suggest cloud deployment options.
- Ask AI to help you write a project summary along with either: (i) a research roadmap; (ii) a funding/business plan; or both.

8 Miscellaneous Admin Matters

8.1 Project Workload

The project is expected to take about 20–40 hours of work, including off-class work, research, design, implementation, journal writing, and presentation preparation. The workload may vary depending on the complexity of the project, the depth of theoretical understanding required, and the extent of practical implementation.

The substantial work devoted to the project can be varying in nature depending on the project type (Option 1 or Option 2). In Option 1, the workload may mainly involve theoretical research, algorithm design, and implementation of the machine learning model. Hence in this case, it is NOT allowed to use off-the-shelf tools and claim the implementation as your own work. In Option 2, the workload may involve data collection, pre-processing real-world datasets, perform research and make design decisions to select the appropriate machine learning model, and finally, training, evaluating. In Option 2, it is allowed to use off-the-shelf tools and libraries, because the focus is on data and model alignment, solving real-world problems.

Below is a counter-example of a project that is NOT acceptable as either Option 1 (direct use of off-shelf tools) or Option 2 (tested on a toy, not real-world dataset).

Counter-example

```
from sklearn import LinearRegression
from sklearn.model_selection import train_test_split
from sklearn.datasets import load_iris

iris_data = load_iris()
X = iris_data.data
y = iris_data.target
X_train, X_test, y_train, y_test = train_test_split(X, y,
    test_size=0.2, random_state=42)
model = LinearRegression()
model.fit(X_train, y_train)
y_pred = model.predict(X_test)
# draw evaluation metrics, such as accuracy, precision, recall, etc.
```

8.2 Use of AI Tools

The use of AI tools is permitted in this project, provided that you document their use and critically review the generated content in your journal. The journal document will NOT be checked for plagiarism, but it is NOT being graded independently either. One must be graded by presenting the study and implementation of the project. The submission of the journal alone will NOT be considered as completing A2.

8.3 Extension and Special Considerations

A2 submissions may be extended by up to three days. You can apply for an extension via the Assignment Extension page on Canvas.

Because the A3 peer viva exam is organised in multiple student groups, we cannot grant ad hoc extensions for A3. Since A3 depends on A2, we are also unable to extend A2 by more than three days. Please do not submit such requests to the Coordinator, as they will not be answered.

If you do not attend A3, you will receive no marks for neither A2 or A3 in this course.

If, for *legitimate and unavoidable reasons*, you cannot attend A3, then due to the size of this course we cannot offer a make-up exam in 2025. Instead, we will conduct individual interviews in the first half of 2026, which will significantly delay the release of your grades.

For "legitimate and unavoidable reasons", you must provide written evidence a professional or institution, such as a medical certificate or court notice. Some non-qualifying issues are listed (but not limited to)

- Work commitments do NOT qualify if you are a full-time student.
- Family commitments do NOT qualify (See "Minor Impacts" below to find solution to alleviate the impact).
- Personal commitments, such as travel, holidays, or other personal engagements, do NOT qualify.

- Technical issues, such as computer failure, software malfunction, or internet connectivity problems, do NOT qualify — unless a major outage is reported by the service provider, such as UTS IT, Microsoft, or Google.

Minor impacts: Most commonly, students may encounter minor issues that can affect your ability in the last few days prior to A2 or A3. If that is the case, you are advised to submit your project as is, participate the presentation as scheduled. After the presentation, you can seek evidence from a professional, and submit a request for special consideration to assigned staff (to be announced later). If your case is verified AND your record of *milestone submissions* (See section 7) of A2 is complete (which means you have been working on the project consistently, not just at the last minute — hence the last-minute mishap has limited and managable impact), your A2 and A3 grades will be adjusted accordingly.

If you are affected by a disability or health condition that may impact your ability to complete the project, you are encouraged to contact the Academic Liaison Officers (ALOs) in the Faculty of Engineering and Information Technology (FEIT) for support and to discuss possible learning adjustments. Two Academic Liaison Officers (ALOs) in the Faculty responsible under the UTS Student Rules for approving learning adjustments for students:

Undergraduate Students – Timothy Boye (FEIT.Undergrad.ALO@uts.edu.au)

Postgraduate Students – Nham Tran (FEIT.Postgrad.ALO@uts.edu.au)

UTS is required under the Disability Discrimination Act 1992 to make reasonable adjustments to accommodate students with disabilities in order to avoid direct or indirect discrimination.