Assignment 2

Start Assignment

- Due 1 Mar by 20:00
- Points 40
- Submitting a file upload
- Available 1 Feb at 20:00 - 5 Mar at 20:00

Assignment Name	AI-based Project for cybersecurity (AI4Cyber)
Assignment Description	This assignment aims to guide students through the development of a complete machine learning project, focusing on real-world cybersecurity applications that involve both regression and classification tasks. The assignment challenges students to apply their technical skills to practical cybersecurity scenarios. Students are expected to acquire the following skills from the assignment: - **Data Collection and Processing**: Gathering and preprocessing data from open-source websites, ensuring that the data is clean, normalized, and ready for analysis. - **Model Selection and Evaluation**: Selecting appropriate machine learning models based on the problem context, and evaluating their performance using relevant metrics such as accuracy, precision, recall, and F1 score. - **Technical Implementation**: Gaining hands-on experience with Python and key libraries like pandas, scikit-learn, and matplotlib for data handling, modeling, and visualization. - **Report Writing and Presentation**: Writing a structured report that clearly communicates the project objectives, methodologies, findings, and conclusions, supported by appropriate visualizations and references. Students have already chosen the following cybersecurity-related topic in the assignment one: 1. **Misinformation Detection on Social Media**: Detect misinformation on social media platforms. Students will collect real-world social media data, extract useful features (like text content and user behaviour), and train models to identify false or misleading posts. The goal is to help improve online safety and address a key cybersecurity challenge in today's digital world. 2. **Spam or Malware Detection**: Detect spam messages or potential malware threats. Students will work with datasets containing emails, messages, or files, extract relevant features (such as text patterns or file properties), and build models to classify harmful content. The project aims to help protect users from digital threats and improve cybersecurity in communication systems. 3. **Detecting Software Vulnerabilities**: Detect security vulnerabilities in software code. Students will analyse source code to identify patterns that may lead to bugs or exploits, and train models to automatically flag risky code. The goal is to help developers find and fix vulnerabilities early, improving the overall security of software systems.

	4. **Network Traffic Classification for Anomaly Detection**: Classify network traffic and detect unusual or potentially malicious activity. Students will analyse network data, extract features such as packet size and timing, and train models to distinguish normal behaviour from anomalies. The goal is to support early detection of cyberattacks and improve network security. 5. **Cybersecurity threat detection in banking**: This covers a range of cybersecurity problems in the banking sector. Students are expected to do some research to choose a suitable cybersecurity problem (e.g. fraud detection, identify theft, etc.) to solve using AI engineering. The goal is to improve the system security in this important industry sector.
Weight	40% of your total marks for the unit
Due Date	Due on Sunday AEST 11:59 pm 01/03/2026 (Week 8) **Extended due date: AEST 11:59 pm 08/03/2026 (Week 9)**
Submission	• A report : ◦ Size: up to 3000 words ◦ Format: pdf • A zip file: ◦ Datasets that you have already processed. (JSON/CSV and so on) ◦ Readme file (pdf/md): tell us how to run your code ◦ Your source code （including the final models you used） • Meeting Minutes ◦ Format: pdf • Contribution Form ◦ Format: pdf
Late Penalties	10% deduction of the available mark per calendar day or part thereof for up to one week. Submissions that are late more than 7 calendar days after the due date will receive a mark of zero (0) and no assessment feedback will be provided.

1 Project Description:

The project description provides an overview of the background and essential requirements for the cybersecurity-related projects.

This project involves developing a comprehensive machine-learning solution that integrates project management, design elements, and technical implementation to address real-world cybersecurity challenges. Students will work in teams to complete the project in three phases: creating a detailed project management plan, implementing a machine learning model, and developing a dynamic website to showcase their results.

The Machine Learning Web Application aims to deliver an interactive platform for users to engage with machine learning models and visualize data insights. The primary goal is to demonstrate practical machine learning applications in real-world cybersecurity-related scenarios, enhancing user interaction and understanding of the underlying models.

Core Functional Requirements for the whole project:

- Select a project topic and find relevant datasets suitable for the chosen topic.
- Investigate and analyse the topic, using machine learning techniques for prediction, attribution, or classification to gain a better understanding and response to the cybersecurity-related topic.
- Develop a website application that allows users to interact with trained models and provides corresponding responses based on user inputs.
- Include data visualization features to help users understand the dataset.

(Note that we have highlighted any information related to Assignment 2.)

2 Submission

You must submit your assignment via the assignment submission link (i.e., "Assignment 2 Submission") on the Canvas site by the deadline specified in Section 1 (Due on Sunday AEST 11:59 pm 05/10/2025 (Week 8).

- No hard copy submission is required for this assignment.
- You are required to submit your assignment named with your group name. For example, if your group name is "Session01-Group 1", you would submit the files named "session-xx-group-1-AssignmentName.pdf".
- Do not include any unnecessary files in this folder
- Note that marks will be deducted if this requirement is not strictly complied
- No submission is accepted via email.

Detailed Requirement:

- A report:
  - Size: up to 3000 words
  - Format: pdf
- A zip file:
  - Datasets that you have already processed (JSON/CSV and so on)
  - Readme file (pdf/md): tell us how to run your code
  - Your source code (including the final models you used)
- Meeting Minutes
- Contribution Form

********Only one student per group is required to submit all the assignment documents on behalf of the group.********

3 Overview of the tasks:

- Use the **topic** you selected for the project.

- Clearly define the **intended users** of your project. The user might be the elderly, young children, or whoever you like, although the choice should make sense for the data and topic of choice.
- **Data Collection**: Use open-source websites to collect relevant data. Ensure the data is appropriate for the selected topic and intended users.
    - **Note**: Using only one simple dataset will earn you a basic score. To achieve a high distinction (HD) grade, you need to find and use additional suitable datasets. Your final grade will be based on how well the additional datasets enhance your work, earning you points above the basic score.
- **Data Processing**: Clean and preprocess the collected data. This may include handling missing values, normalizing data, and feature extraction to prepare the data for analysis.
- **Data Analysis**: Analyze the preprocessed data to uncover patterns and insights. Utilize statistical methods and visualizations to better understand the data and inform the selection of the machine learning model.
- **Model Selection**: Choose an appropriate machine learning model based on the data and the problem you are addressing. Consider models such as regression, classification, clustering, etc.
    - **Note**: In this project, students are required to use at least two types of machine learning methods (classification, clustering, regression) out of the three. Students may use two machine learning methods of the same type, but they must provide strong justifications. Without compelling reasons, they risk receiving lower marks due to the low complexity. To earn additional HD points, students should conduct extra machine learning experiments beyond the existing two algorithms, based on their project goals. For example, if a student implements two different clustering methods and one regression method, we will award additional points above the basic score based on the implementation and effectiveness of the code.
- **Model Evaluation**: Evaluate the chosen model's performance using relevant metrics. This could include accuracy, precision, recall, F1 score, or other suitable evaluation criteria.
- **Implementation in Python**: Execute all tasks using Python, leveraging libraries such as pandas, scikit-learn, and matplotlib for data handling, modeling, and visualization.
- **Report**: Compare the evaluation metrics and present the findings in a comprehensive report. The report should include visualizations, a detailed explanation of the process, and an interpretation of the results.
- Note that the core functionalities in Assignment 2 are highlighted in orange font.

4 Deliverables

Your submission should contain the following files:

1. Report: Write a report of up to 3000 words (excluding coverpage, table of contents, bibliography, and appendix) that consists of the following sections:

- **Project title**
    - Title of your machine learning project. This can be included in the cover page.
- **Your group identities**
    - Your group name and ID, student names and IDs, tutor's name. This can be included on the cover page.
- **Introduction**
    - A precise and succinct description of what motivations you wanted your machine learning project to solve, and who the intended user is.
- **Problem Framing**

    - Accurately defining the problem is essential. Outline the specific challenge you aim to address, the limitations of existing solutions, and why a machine-learning approach is suitable.
- **Data Collection**
    - Detail the sources and methods used to gather your dataset. Describe any specific criteria or tools used to collect the data, ensuring it is relevant and sufficient for your analysis. Mention any challenges encountered during data collection and how they were addressed.
- **Data Processing**
    - Outline the steps taken to clean and preprocess the collected data. This may include handling missing values, normalization, feature engineering, and transforming the data into a format suitable for machine learning algorithms. Describe the processes of joining and merging datasets, ensuring consistency and relevance.
- **Machine Learning Model Selection**
    - Describe the criteria used to select the appropriate machine-learning models for your project. Discuss the algorithms considered, the rationale behind your choices, and how they align with the problem's framing. Include any preliminary tests or comparisons conducted to determine the best-performing models.
- **Implementation**
    - **Technical Implementation**
        - This section contains a high-level description of your implementation, including libraries used, references to external code sources such as templates, and reasons for any differences between your final decisions. You should briefly explain the reasons why your project was challenging (e.g., extensive wrangling was required)
    - **Implementation Evaluation**
        - Evaluate the effectiveness of your implementation by comparing the results against your initial objectives and performance metrics. Discuss the performance of the selected machine learning models, including any validation and testing procedures. Highlight any unexpected outcomes, and provide insights into how well the implementation addresses the problem. Tests or comparisons are conducted to determine the best-performing models.
- **Conclusion**
    - Summarize the key findings and outcomes of your project. Reflect on how effectively your machine learning solution addressed the initial problem and met the objectives. Discuss any significant insights gained, the implications of your results, and potential areas for future work or improvements. Highlight the overall contribution of your project to the field and its potential impact on the intended users.
- **Bibliography**
    - Appropriate references of all resources that have influenced your work in Harvard style.
- **Appendix**
    - Additional files can be added to this part if you have them.

If possible, avoid using a single screenshot of the entire page since the resolution might be low; instead, crop and explain individual sections of the page. It is also recommended that you export your PDF using a local word processor (e.g., Microsoft Word), as exporting your document as a PDF directly from Google Docs will result in low-quality images. Make sure you can read and understand the PDF document and its images at A4 size without requiring further enlargement.

2. A zip file:

- **Datasets that you are already processed (JSON/CSV and so on):**
  - The dataset must only include the data used in your final version machine model.
  - **Note**: We will provide a basic dataset for each project topic, attached in this specification. Each dataset includes a dataset file and a README file with explanations about the dataset.
  - **Basic_Datasets.zip (https://swinburne.instructure.com/courses/71633/files/40836685?wrap=1)** ↓ **(https://swinburne.instructure.com/courses/71633/files/40836685/download?download_frd=1)**
- **Readme file (pdf/md)**:
  - explain how to configure your project environment using conda commands, how to perform further data processing based on your prepared training dataset, how to train your model, and how to use your model for prediction.
- **Source code**:
  - any code related to your assignment 2.

3. Meeting Minutes: Students are required to hold at least one meeting each week since the team's inception and submit all meeting minutes, along with other deliverables, as part of the assignment. You can use the attached meeting minutes as they are or as a reference to create your own. **Meeting Minutes Example.docx (https://swinburne.instructure.com/courses/71633/files/40836565?wrap=1)** ↓ **(https://swinburne.instructure.com/courses/71633/files/40836565/download?download_frd=1)**

4. Contribution Form: A form includes sections for the personal information of each team member, details of the contribution, and other additional information. You can download the form **Group Assessment Contribution Form.docx (https://swinburne.instructure.com/courses/71633/files/40836967?wrap=1)** ↓ **(https://swinburne.instructure.com/courses/71633/files/40836967/download?download_frd=1)**

Important Notes:

- Please be careful to ensure you do not publicly post anything which includes your reasoning, logic, or any part of your work to the Canvas discussion, doing so violates Swinburne plagiarism/ collusion rules and has significant academic penalties. Use email to your allocated tutor to raise questions that may reveal part of your reasoning or solution.
- In this assessment, you must **NOT** use generative artificial intelligence (AI) to generate any materials or content related to the assessment task.
- According to the feedback from students, the team has updated the contribution form. All the team members are required to discuss and sign together within the group before submitting. This can solve the issue of student form submissions being overwritten.

5 Marking Criteria

You must acknowledge all statements and information taken from other sources and adhere to the guidelines published regarding plagiarism. All ideas and material taken from references must be cited within the report itself and a full reference list and bibliography (if appropriate) must be provided at the end of the report. Diagrams and/or tables may be used if you think this will strengthen your arguments. Remember that diagrams and tables adapted from other sources must be cited (***Harvard*** style) as well.

COS30049 Assignment 2 Rubric

Criteria	Ratings				Pts
Report: 1) Explanation of Data Collection and Processing Operations	**5 Pts** **-** Comprehensive and clear explanation of data collection methods and processing steps, including rationale for chosen methods.	**3 Pts** **-** Adequate explanation with minor gaps in detail or rationale.	**1 Pts** **-** Limited explanation with significant gaps in detail or rationale.	**0 Pts** **-** Explanation is missing or inadequate.	5 pts
Report: 2) Explanation of Data Analysis	**2 to >1.0 Pts** **-** Detailed and clear explanation of data analysis methods and interpretation of results.	**1 to >0.0 Pts** **-** Basic explanation with some detail, but lacks depth or clarity.	**0 Pts** **-** Explanation is missing or insufficient.		2 pts
Report: 3) Justification of Model Selection	**2 Pts** **-** Thorough justification for model choice, including consideration of alternatives and alignment with project goals.	**1 Pts** **-** Limited justification with significant gaps in reasoning or insufficient consideration of alternatives.	**0 Pts** **No marks** Justification is missing or poorly articulated.		2 pts
Report: 4) Quality of Writing and Logical Structure	**1 to >0.0 Pts** **-** Exceptionally well-written with a clear and logical structure, no grammatical or spelling errors, and a smooth flow between sections.	**0 Pts** **-** Poorly written with frequent grammatical or spelling errors; lacks logical structure and coherence.			1 pts
Report: 5) Academic Referencing and Citations	**1 Pts** **Full marks** All sources are correctly cited using the Harvard style; a comprehensive and accurate reference list is included.	**0 Pts** **-** Significant errors in citation or missing references.			1 pts
Report: 6) Completeness	**1 Pts** **-** All required sections are included and fully completed with appropriate detail.	**0 Pts** **-** Several sections are missing or incomplete; significant gaps in content.			1 pts

Criteria	Ratings			Pts
Source code execution: 1) Core Functionalities: Core features and functionalities of the project are successfully realized and perform as intended (Please refer to Assignment 2 - orange font)	**10 Pts** **-** Data is thoroughly collected from reliable sources, cleaned, normalized, and prepared for analysis. The process is well-documented, with clear explanations of the methods used, including justifications for each step. Models are selected based on a comprehensive evaluation of the problem context, and the performance is thoroughly evaluated using appropriate metrics such as accuracy, precision, recall, and F1 score. The reasoning behind the model choice is clearly articulated and well-supported.	**8 Pts** **-** Data is collected and processed appropriately, but there may be minor gaps in documentation or justifications for the methods used. Data is generally clean and ready for analysis. Models are selected appropriately, but the evaluation may lack depth in some areas. The rationale for model selection is provided but may have minor gaps.	**4 Pts** **-** Data is collected and processed with significant gaps in documentation or justifications. Some steps in data preparation may be incomplete or unclear. Model selection is basic, with minimal evaluation and justification. The rationale for the choice is weak or unclear.	10 pts
Source code execution: 2) Datasets	**4 to >3.0 Pts** **-** The dataset is complete and highly relevant to the final machine learning model, containing only the necessary data with no extraneous information. The dataset is provided in a clear, accessible format (jsonl, json, or csv), with proper organization and no errors, making it easy to load and use.	**3 to >1.0 Pts** **-** The dataset has some relevant data, but also includes significant irrelevant data or lacks key components. The dataset is provided in the correct format, but with noticeable formatting errors or poor organization, making it somewhat difficult to use.	**1 to >0 Pts** **-** The dataset is largely incomplete or irrelevant to the final model, with little to no documentation of preprocessing steps. The dataset format is incorrect or contains significant errors, making it very difficult to use.	4 pts
Source code execution: 3) readme file	**2 to >1.0 Pts** **-** Comprehensive and Clear Instructions: The readme file provides detailed and clear instructions on how to configure the project environment using commands. Every step is explained thoroughly, making it easy for users to	**1 to >0 Pts** **-** Incomplete or Confusing Instructions: The instructions provided are either incomplete or confusing, with significant gaps in the explanation of environment setup, data processing, model training, or prediction. Insufficient Detail: The		2 pts

Criteria	Ratings			Pts
	replicate the environment setup. The guide is easy to follow and clearly connected to the dataset and model. Thorough Model Training Instructions: The process of training the model is described in detail, including all commands and parameters needed. The explanation ensures that anyone can replicate the training process with the provided dataset. Clear Prediction Guide: Instructions on how to use the model for prediction are straightforward, with clear examples or commands provided. Professional Presentation: The readme file is well-organized, formatted correctly (in either pdf or markdown), and free from errors, with a logical flow and professional appearance.	readme lacks critical details, making it difficult for users to replicate the project setup or understand the processes involved. Many commands or steps are missing or incorrect. Poor Presentation: The readme file is poorly organized, with significant errors or formatting issues that hinder readability and usability.		
Source code execution 4) Comments: Proper code comments inside the file	**2 to >1.0 Pts** **-** The source code is thoroughly commented. All significant sections of the code have clear and informative comments that explain the purpose and functionality of the code. The comments enhance the readability and maintainability of the code, making it easy for others to understand and modify.	**1 to >0 Pts** **No marks** The code has minimal comments, with many sections lacking explanations. The comments provided may be too brief, vague, or inconsistent, leading to difficulties in understanding the code's functionality.		2 pts
Source code execution, 5) Code Structure: Clean and well structured source code and program files	**2 to >1.0 Pts** **-** The source code is clean, well-organized, and follows best practices for code structure. The program files are logically organized, with a clear hierarchy, and are easy to navigate. The code is modular, with functions or classes clearly separated, making it maintainable and scalable.	**1 to >0.0 Pts** **-** The source code is generally organized but may have some issues in structure or organization. The program files are mostly well-structured but might have some redundancies or lack clear separation between different functions or modules.	**0 Pts** **No marks** The source code is poorly structured, with a disorganized layout and lack of clear organization. The program files may be difficult to navigate, with unclear or inconsistent structuring, making the code hard to maintain or understand.	2 pts

Criteria	Ratings		Pts
Additional datasets Use additional datasets to reach the project goal.	**4 to >2.0 Pts** **-** Utilize a large number of additional datasets to support machine learning algorithms, while correctly and appropriately applying data processing methods for preprocessing and analysis (including visualization). The datasets align with the project goals and demonstrate a significant amount of work.	**2 to >0 Pts** **-** Utilize a moderate or small amount of additional datasets, but the data has weak relevance to the machine learning tasks and project goals. Data processing and analysis are minimal or contain errors, resulting in a relatively low additional workload.	4 pts
Additional machine learning implementations Add additional machine learning implementations on top of the basic two types, based on the project goals (the types of implementations can be repeated).	**4 to >2.0 Pts** **-** Apply appropriate machine learning methods based on the project goals and data, with well-tuned parameters. The code is clear and correct, and the model performs excellently, contributing effectively to the project's objectives.	**2 to >0 Pts** **-** The chosen machine learning methods are not entirely appropriate, and there are flaws in the code implementation. The report lacks sufficiently detailed explanations and descriptions, parameter tuning needs improvement, and the model's effectiveness is limited or only weakly aligned with the project goals.	4 pts
			Total points: 40