Software Engineering Stage 6 (Year 12) – teacher support resource

**Software automation**

# Teacher support resource

**Teacher note:** this resource has been designed to facilitate the ready conversion into a student booklet by removing the answers within the response windows. Teacher notes can be deleted before distributing to students. This booklet should be submitted as the documentation component of the assessment task in part or in whole.

Student name:

Class:

Teacher:

Contents

[Teacher support resource 1](#_Toc193810279)

[Unit overview 3](#_Toc193810280)

[Teaching advice on Software automation 7](#_Toc193810281)

[Task description 8](#_Toc193810282)

[Teaching advice 10](#_Toc193810283)

[Submission details 11](#_Toc193810284)

[Component A 11](#_Toc193810285)

[Component B 11](#_Toc193810286)

[Steps to success 13](#_Toc193810287)

[What is the teacher looking for? 15](#_Toc193810288)

[Marking guidelines 16](#_Toc193810289)

[Student-facing rubric 18](#_Toc193810290)

[Glossary 23](#_Toc193810291)

[NESA glossary keywords 28](#_Toc193810292)

[Software automation 29](#_Toc193810293)

[Algorithms in machine learning 29](#_Toc193810294)

[Programming for automation 94](#_Toc193810295)

[Significance and impact of ML and AI 100](#_Toc193810296)

[Appendix 117](#_Toc193810297)

[References 148](#_Toc193810298)

[Legislation 156](#_Toc193810299)

# Unit overview

Teachers are provided options to differentiate content depending upon students’ experience. At a minimum, students complete the activities within this teacher support resource (TSR). They are guided through the development of a linear regression algorithm from the [Software Engineering course specifications](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview#software-engineering-course-specifications-software_engineering_11_12_2022) (Course specifications) document to create a mark estimation program. A simple solution is provided in this document ([Appendix 1](#_Appendix_1_:)) and sample code is available in the **Software Automation** folder under the **Files** tab of the [Software Engineering channel](https://schoolsnsw.sharepoint.com/:f:/r/sites/TASNSWStatewideStaffroom/Shared%20Documents/13.%20%F0%9F%A7%91%E2%80%8D%F0%9F%92%BB%20Software%20Engineering%2011-12/Secure%20Software%20Architecture?csf=1&web=1&e=6nkPE3) via the [TAS Statewide Staffroom](https://teams.microsoft.com/l/team/19%3acd41312b69a14cd38a7c429ffd90493a%40thread.tacv2/conversations?groupId=cd5a04e1-7742-47dd-b141-9519486d9e00&tenantId=05a0e69a-418a-47c1-9c25-9387261bf991). The Python code for a ‘middle-level’ solution is also provided in this channel and considerations for a ‘high-level’ solution discussed. There are at least 3 options in delivering this task:

1. A summative assessment if it fits within their existing assessment schedule (**Note**: the [assessment schedule](https://education.nsw.gov.au/teaching-and-learning/curriculum/tas/tas-curriculum-resources-7-12/tas-11-12-curriculum-resources/assessment-schedule-software-engineering-year-12) published by the department did not include this as one of the assessment tasks and suggested content be formally assessed in the trial examination). A formal assessment task is included in this document for teachers choosing this option. This is also available in the **Software Automation** folder under the **Files** tab of the [Software Engineering channel](https://schoolsnsw.sharepoint.com/:f:/r/sites/TASNSWStatewideStaffroom/Shared%20Documents/13.%20%F0%9F%A7%91%E2%80%8D%F0%9F%92%BB%20Software%20Engineering%2011-12/Secure%20Software%20Architecture?csf=1&web=1&e=6nkPE3) via the TAS Statewide Staffroom.
2. A summative assessment of the task that integrates Software automation, Programming for the web, Secure software architecture and Software engineering project into a yearlong progressive web app (PWA) solution. Students could create a web-based markbook with Structured Query Language (SQL) lite (SQLite) database and predictive capabilities. Other web-based catalogue type systems could include the machine learning aspects of this unit.
3. A formative assessment of a solution to the problem as a practical group-based project to deliver syllabus content within the software automation focus area and programming for automation topic. In this scenario students would improve upon the existing ‘low-code’ solution.

All files, comma-separated value (CSV) markbooks and Python sample code are available to teachers in the **Software Automation** folder under the **Files** tab of the [Software Engineering channel](https://schoolsnsw.sharepoint.com/:f:/r/sites/TASNSWStatewideStaffroom/Shared%20Documents/13.%20%F0%9F%A7%91%E2%80%8D%F0%9F%92%BB%20Software%20Engineering%2011-12/Secure%20Software%20Architecture?csf=1&web=1&e=6nkPE3) via the TAS State wide staffroom.

This unit guides students through the Software automationfocus area. The content for this focus area is delivered as theory required for, and integrated into, the development of a mark estimation program. Students are provided with the following scenario.

A student named Alex has missed a maths exam.

Her teacher needs to estimate Alex's score and has been told that machine learning can predict a student's mark on a missed assessment by analysing historical data, identifying patterns and using these patterns to make informed predictions.

Alex’s maths teacher has also been told that you are studying machine learning and may be able to help.

Students consider various machine learning algorithms and models to create a solution. They are guided through a simple application of a linear regression algorithm using Python code (by importing NumPy and scikit-learn machine learning frameworks). The use of these modules imported into Python is assumed by their reference in the code sample from page 28 of the [Course specifications](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview#software-engineering-course-specifications-software_engineering_11_12_2022).

During Weeks 1 to 2, students complete activities including cloze passage, note making on relevant videos and class discussions about the fundamental concepts within the Software automation focus area. They are introduced to and become familiar with the subject specific terminology of this focus area including machine learning, DevOps, robotic process automation (RPA) and business process automation (BPA). Students participate in a jigsaw activity which provides a deeper dive into these concepts and facilitates the construction of class notes and a study guide. These groups apply their understanding to consider a school-based scenario ‘The school library upgrade with machine learning’. Students consider how to apply the MLOps stages from the Course specifications to ‘The school library upgrade with machine learning learning’ scenario.Students complete activities 1 to 9 of this TSR.

Students are introduced to the coding task for this unit, the mark estimation program, referencing linear regression from the Course specifications.

In Weeks 3 to 4, students list all school-based systems that could be automated with software. This list includes a mark estimation program. Other school-based systems could be automated as a project for this unit. Students distinguish between artificial intelligence (AI) and machine learning (ML) by completing activities 10 to 11 of this TSR. Students explore models of training ML through close investigation of [Google’s Teachable Machine](https://teachablemachine.withgoogle.com/). Students complete activities 12 to 17 of the TSR and investigate common applications of key ML algorithms. They are given a quiz (Activity 18) to check their understanding.

For their project, students apply their understanding of [How to choose the Right Machine Learning Algorithm (23:04)](https://youtu.be/uh6iYQEHyyI?si=LxFMXBKQdXJxft6c)to their mark estimation task. Students complete Activity 19.

During Weeks 5 to 6, students investigate case studies that demonstrate the common applications of key ML algorithms. They form research teams and present their findings to the class including a brief explanation of how their algorithms work. These presentations form part of class study notes. Students research models used by software engineers to design and analyse ML, including decision trees and neural networks. Students differentiate between ‘traditional’ decision trees used in computing from their use in ML. They annotate the neural network diagram from the [Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) and develop their understanding of the training and execution cycle. Students complete the TSR activities 20 to 32.

In Weeks 7 to 9, students describe types of algorithms associated with ML by investigating [Linear Regression (with Desmos) (9:25)](https://youtu.be/_ldsNWOHHBk?si=_2LRIjC2VZJkd9LA). Students apply linear regression using Python code by importing [NumPy](https://numpy.org/doc/stable/) and [scikit-learn](https://scikit-learn.org/stable/getting_started.html) machine learning frameworks. Students work through the mark estimation program including a class discussion on the best approach to take. They consider the social and ethical issues that arise with this predictive system. Students consider other real-world examples of linear regression ML algorithms. Students complete activities on linear regression, polynomial regression and logistic regression and investigate the Robodebt case study. Students explore how patterns in human behaviour influence ML and AI software development. They investigate the effect of human and dataset source bias in the development of ML and AI solutions.

Students complete the activities in the TSR from 32 to 57.

In Week 10, students present their mark estimation project, submit the code and documentation and complete activities within the TSR.

This teacher support resource (TSR) provides activities that align with the content of the syllabus to ensure content is covered. After the initial linear delivery of this unit, teachers will identify opportunities to integrate theory and practice within single lessons. The Software automation task is developed in the classroom under the supervision of the teacher using explicit teaching methods outlined in this TSR.

## Teaching advice on Software automation

**Why is it important?**

The Software automation focus area provides opportunities for students to extend their knowledge and understanding of programming and its diverse applications. Students are provided with opportunities to design and implement code as they apply object-oriented programming knowledge and design thinking to solve problems. Learning in this focus area emphasises the fields of machine learning (ML) and artificial intelligence (AI). This includes some of the fundamental knowledge and skills required to program for emerging technologies and the significance and impact of ML and AI on the individual, society and the environment.

**Additional advice**

The practical learning experiences should apply the skills associated with developing and evaluating the success of ML systems and the code that drives them. This highlights opportunities for students to incorporate software automation into their Software engineering project.

## Task description

A student named Alex missed a maths exam.

Her teacher needs to predict Alex's score and has been told that machine learning can predict a student's mark on a missed assessment by analysing historical data, identifying patterns and using these patterns to make informed predictions.

Alex’s maths teacher has also been told that you are studying machine learning and may be able to help.

Use Python with the NumPy and scikit-learn modules to calculate Alex’s potential score.

The product will be a program to predict student grades for missed assessments.

You are to:

* set up the work environment
* set up the training data
* plan the UI
* code initial functionality
* plan the estimation algorithm and overall system structure
* code remaining functionality
* test and evaluate.

**Student support material**

* [scikit-learn](https://scikit-learn.org/)
* [Matplotlib](mat)
* [NumPy](https://numpy.org/)

**Outcomes being assessed**:

A student:

* justifies methods used to plan, develop and engineer software solutions **SE-12-01**
* applies structural elements to develop programming code **SE-12-02**
* evaluates practices to safety and securely collect, use and store data **SE-12-04**
* justifies the selection and use of tools and resources to design, develop, manage and evaluate software **SE-12-06**
* tests and evaluates language structures to refine code **SE-12-08**
* applies methods to manage and document the development of a software project   
  **SE-12-09**

[Software Engineering 11–12 Syllabus](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview) © NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2022.

**Suggested weighting**: 30%

**Teacher note:** the [assessment schedule](https://education.nsw.gov.au/teaching-and-learning/curriculum/tas/tas-curriculum-resources-7-12/tas-11-12-curriculum-resources/assessment-schedule-software-engineering-year-12) published by the department did not include this as one of the assessment tasks and instead suggested content be formally assessed in the trial examination. Teachers will need to align this project with their school assessment schedule if they are to use this as a summative assessment task.

## Teaching advice

Estimating the mark can be done through several levels of complexity.

A simple implementation would consider a single dimension, for example, a mark estimated through looking at the students marks in other assessments only.

A better solution would consider the estimate by looking at two dimensions and looking at what other students got for that assessment as well.

Higher-level solutions may be able to analyse the data and see if a linear or polynomial regression model is best for the specific data. For example, some classes may see marks that progress in a linear fashion however other classes may improve over time with marks instead fitting on a curve.

Other considerations include the use of standard deviations, z scores and the use of averages of other tasks. Teachers should lead a class discussion introducing the problem statement and how to best approach the task to ensure fairness. Students should be motivated by the nature of this process as it applies to their high school experience. The discussion should include the conditions for being able to estimate a mark (decision tree) including:

* + Are there enough students present to estimate the mark effectively?
  + Are there not too many gaps in the data?
  + Are the marks consistent enough to estimate the mark effectively?
  + Have enough assessment tasks been completed?

# Submission details

Students are to submit 3 components – Component A, B and C.

## Component A

Documentation of the project including:

* IPO tables
* diagrams
* decision tree
* intrinsic documentation
* supporting information
* evidence of testing
* social and ethical issues.

See the [Course specifications](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview#software-engineering-course-specifications-software_engineering_11_12_2022) for guidelines on how to present some project documentation elements.

The new syllabus no longer contains reference to ‘IPO tables’. They were a part of the old SDD syllabus. IPO tables will not be required from students in the Software Engineering HSC exam. They do, however, provide a conceptual framework to better understand the problem. Teachers may decide to ask their students to use another format (for example, a brief textual explanation) instead of IPO tables.

## Component B

A zip file of any implemented solution or sample code including use of training data, consideration of overall structure, OOP layout and evidence of usability considerations.

Component C

A presentation to the class who are playing the role of the teacher client with questions and answers including justification of the model.

## Steps to success

Table 1 – assessment preparation schedule

|  |  |
| --- | --- |
| ****Steps**** | ****What I need to do**** |
| Set up the work environment | 1. Install and test Python3 and a relevant integrated development environment (IDE).  * pip3 install NumPy and scikit-learn * if using a graphic user interface (GUI), students may want to also have TKinter or PyGame installed as well. |
| Set up training data | 1. Obtain or mock up a CSV file with marks for a class. There should be at least 4 tasks in the markbook and a reasonable number of students. |
| Plan the UI | 1. Create a storyboard for the pages that will be present in the system. user interface (UI) may be graphical or a command-line interface (CLI). |
| Code initial functionality | 1. Code the means for importing and displaying the marks held within the CSV markbook file. 2. Code a means for specifying which student and which assessment task the estimated grade should be calculated for. |
| Plan estimation algorithm and overall system structure | 1. Decide whether a linear, polynomial or logistic regression model is most appropriate and provide justification. 2. Create a decision tree outlining the conditions for a missing assessment mark to be reliably calculated. 3. Create an Input, Process, Output (IPO) table with a general strategy. 4. Use a diagram (for example, a class diagram or structure chart) to express the structure of your product or processing. Provide justification for your choice of diagram. |
| Code remaining functionality | 1. Using an agile development approach and object-oriented paradigm (OOP), build out the rest of the functionality. |
| Test and evaluate | 1. Provide a report demonstrating that you have tested your product, recording any syntax, logic and runtime errors you identify. 2. Provide an outline of whether the system is using artificial intelligence (AI) or machine learning (ML). 3. Assess any social and ethical impacts of the system and any dangers of bias in the system. |

## What is the teacher looking for?

This task will require students to create a system to predict marks for students for missed assessment tasks.

The system will incorporate:

* the creation of a relevant UI (either graphical or CLI)
* the ability to read data from a file (stored as CSV).

The mark estimation system is to have the following functionality:

* Allow for current marks to be displayed with missing marks clearly labelled.
* Have an effective means for the user to identify which marks are to be estimated.
* Present the estimated mark with supporting information (for example, task average).
* Allow the markbook to be exported (as csv) with estimated marks incorporated.

The testing and evaluation report will incorporate:

* evidence of thorough testing
* thoughtful discussion and evaluation of the social and ethical issues.

## Marking guidelines

Table 2 – assessment marking guidelines

|  |  |
| --- | --- |
| ****Grade**** | ****Marking guideline descriptors**** |
| ****A**** | The student:   * develops highly effective and clean code utilising effective programming methodology * provides comprehensive planning documentation with broad and strong training data * demonstrates an extensive understanding of designing and creating effective and accessible UIs * creates a solution that effectively estimates marks taking into account several factors * provides detailed information supporting their estimate * demonstrates an extensive ability to test and document their system and its impacts. |
| ****B**** | The student:   * develops effective and clean code using effective programming methodology * provides effective planning documentation with relevant training data * demonstrates a thorough understanding of designing and creating clean and accessible UIs * creates a solution that provides a reasonable estimated mark using more than one factor * provides more than one piece of information supporting their estimate * demonstrates a thorough ability to test and document their system and its impacts. |
| ****C**** | The student:   * develops sound code using some characteristics of effective programming methodology * provides valid planning documentation with relevant training data * demonstrates a sound understanding of designing and creating useable and accessible UIs * creates a solution that provides an estimated mark using a single factor * provides a relevant piece of information supporting their estimate * demonstrates a sound ability to test and document their system and its impacts. |
| ****D**** | The student:   * develops basic code using effective programming methodology * provides limited planning documentation with some training data * demonstrates a basic understanding of designing and creating UIs * organises some data within the database and accesses attempts to access it with SQL * demonstrates an attempt to test and document their system. |
| ****E**** | The student:   * identifies the requirements for the documentation and production of the system * demonstrates some understanding of designing and creating UIs. |

## Student-facing rubric

Table 3 – rubric for Software automation

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Criteria | Outstanding | High | Sound | Basic | Limited |
| Mark | (5 marks) | (4 marks) | (3 marks) | (2 marks) | (1 marks) |
| Project management |  |  |  |  |  |
| Training data |  |  | Useful training data is created that will allow the system to be tested from a variety of perspectives. | Adequate training data is present that allows for the system to be tested. | An attempt has been made to create relevant training data. |
| IPO tables | IPO tables are presented correctly and outline a sophisticated plan for the processing in a concise manner. | IPO tables are presented correctly and outline a well thought out plan for the processing. | IPO tables are mostly presented correctly and represent the intended logic in a readable manner. | An attempt has been made to produce IPO tables representing some processing relevant to the scenario. | An attempt has been made to produce IPO tables. |
| Class diagram | A relevant diagram has been presented correctly that outlines a thorough and concise structure for the code. | A relevant diagram has been presented correctly that outlines a well thought out and clean structure for the code. | A diagram is presented, mostly correctly, that outlines the intended structure for the code. | An attempt to model the intended structure of the program through a diagram. The diagram contains some errors. | An attempt has been made to use a diagram relevant to the system. |
| Justification of model |  |  | A relevant regression model has been selected and justified well. | A relevant regression model has been selected but the justification could be stronger. | An attempt has been made to link a regression model to the scenario. |
| Decision tree |  | A valid decision tree presented correctly and cleanly with well thought through criteria. | A valid decision tree, presented mostly correctly, with valid criteria. | A decision tree is presented with some criteria relevant to the scenario. | An attempt has been made to create a decision tree. |
| Programming |  |  |  |  |  |
| Overall structure | Code is structured cleanly and neatly making for easy-to-follow code that is maintainable and extensible. Comments are used effectively to help convey the structure. | Code is structured in an easy-to-follow manner with a logical structure. Comments are used appropriately to convey the overall structure. | Code is mostly organised with a general structure that maps to the processing. Comments are present, helping to guide the reader. | There is some general structure to the code. Comments are present outlining the purpose for some sections of code. | An attempt has been made to structure code neatly. |
| OOP layout | A solid breakdown of code into classes making effective use of methods, attributes and OOP concepts. | A logical breakdown of most code into classes making good use of methods, attributes and OOP concepts. | Classes are used for a reasonable amount of the code with relevant use of methods, attributes and OOP concepts. | Classes are used in some areas of the code, but their usage could be improved. | An attempt has been made to use classes in the code. |
| Intrinsic documentation |  |  | Effective use of intrinsic documentation across all code that is present. | Mostly effective use of intrinsic documentation adds to the readability and maintainability of the code. | An attempt has been made to use intrinsic documentation techniques. |
| Overall functionality |  |  |  |  |  |
| Evidence of usability considerations |  | Submission demonstrates a sophisticated UI that is clean and unambiguous. | Submission shows good consideration of usability. Interaction and presentation of data is sound. | Submission shows some consideration of usability. Interaction with the system works but there is room for improvement. | An attempt has been made to incorporate usability in the submission |
| Importing data |  | Evidence of data wrangling by importing correctly and identifying and managing invalid data. | Some evidence of data wrangling during import. Some invalid data is identified and indicated in some manner. | Little evidence of data wrangling during import. Data is imported though it may not be reliable. Invalid data is not identified. | An attempt has been made to import data from a file. |
| Estimating mark | A highly effective mechanism is created that estimates marks using several factors. | An effective mechanism is created that estimates marks using more than one factor. | A mechanism that creates a reasonable estimate based upon a single factor. | A mechanism that creates an estimate though the estimate may not be that accurate. | An attempt has been made to estimate a mark. |
| Supporting information |  | Several pieces of useful supporting information are presented in an effective manner. | Some valid data is presented in a useable manner to support the estimate. | Some data relevant to the scenario is present. | An attempt has been made to show a piece of data. |
| Testing and evaluation |  |  |  |  |  |
| Testing | Extensive testing is documented within the process diary and each task specification is addressed. | Thorough testing is documented within the process diary and most task specifications are addressed. | Some testing is documented within the process diary and task specifications are mentioned. | Testing is mentioned within the process diary and task specifications may be included. | An attempt has been made to test the solution against the task's specifications. |
| Social and ethical issues | A thorough analysis of the potential impacts of the product from a variety of perspectives. | A detailed analysis of the potential impacts of the product from several perspectives. | A sound analysis of the potential impacts of the product. | Some analysis of social and ethical issues relating to the product. | An attempt has been made to discuss some issues relevant to the product. |
|  |  |  |  | **Total** | **/60** |

# Glossary

Teachers may choose to cut up this table and provide a matching exercise for teams of students as a classroom challenge. Alternatively, they can empty the definition column and have students research the meaning and complete the table.

Teachers use professional judgement to adjust these definitions for their focus and context.

The following terms will gather more meaning as students work through this booklet.

Table 4 – glossary

|  |  |
| --- | --- |
| Term | Definition |
| Algorithms | A set of rules or instructions designed to perform a specific task or solve a problem; in machine learning, algorithms are used to make predictions from data. |
| Artificial intelligence (AI) | The simulation of human intelligence processes by machines, especially computer systems, including learning, reasoning and self-correction. |
| Belief systems | The set of principles and values that shape an individual's or group's worldview, influencing how they interact with technology and automation. |
| Business process automation (BPA) | The process of automating complex business processes and functions beyond just individual tasks, often involving multiple systems and stakeholders. |
| Cultural protocols | The established norms, values and practices within specific cultural groups that influence how technology is perceived and used. |
| Data analysis and forecasting | The process of inspecting, cleansing, transforming and modelling data to discover useful information for decision-making, often used for predicting future outcomes. |
| Decision trees | A decision support tool that uses a tree-like model of decisions and their possible consequences; widely used for classification and regression tasks. |
| DevOps | A set of practices that combines software development (Dev) and IT operations (Ops) to shorten the development lifecycle and improve software quality. |
| Economy and distribution of wealth | The influence of automation and AI on economic structures, including job displacement, income inequality and wealth distribution. |
| Human and dataset source bias | The biases that arise from human decisions in data collection and the inherent biases within the datasets used to train ML and AI models, affecting their fairness and accuracy. |
| Image recognition | The ability of a computer or software to identify and process images, typically using machine learning algorithms to classify and analyse visual data. |
| K-nearest neighbour (KNN) | A simple, non-parametric machine learning algorithm used for classification and regression by identifying the 'k' closest training examples in the feature space. |
| Linear regression | A statistical method that models the relationship between a dependent variable and one or more independent variables by fitting a linear equation to the observed data. |
| Logistic regression | A statistical method used for binary classification that models the probability of a categorical dependent variable based on one or more predictor variables. |
| Machine learning (ML) | A subset of artificial intelligence (AI) that enables systems to learn from data, improve their performance over time and make predictions without being explicitly programmed. |
| Machine learning regression models | Models used in machine learning to predict numeric values based on input features. These models learn from historical data to make predictions. |
| Nature and skills required for employment | The changing requirements for skills and job roles as automation and AI technologies evolve, impacting workforce development and training. |
| Neural network models | Computational models inspired by the human brain, consisting of interconnected nodes (neurons) that process input data and learn to make predictions through training. |
| Neural networks | A series of algorithms that mimic the operations of a human brain to recognise relationships in data; commonly used in deep learning to solve complex problems. |
| Object-oriented programming (OOP) | A programming paradigm based on the concept of ‘objects’, which can contain data and code: data in the form of fields (attributes) and code in the form of procedures (methods). |
| Patterns in human behaviour | Recognisable trends in how humans act or respond in various situations, which can influence the design and functionality of AI systems. |
| Patterns related to acute stress response | The identifiable reactions and behaviours exhibited by individuals during high-stress situations, which can inform the design of supportive AI systems. |
| People with disability | The focus on how automation and AI can affect individuals with disabilities, ensuring that technologies are inclusive and accessible. |
| Polynomial regression | A form of regression analysis that models the relationship between the independent variable and the dependent variable as an nth degree polynomial, capturing non-linear trends. |
| Production efficiency, waste, and the environment | The effects of automation on operational efficiency, resource usage and environmental sustainability in production processes. |
| Programming for automation | The process of writing code to create automated tasks and processes, often using various programming languages and tools to enhance efficiency and productivity. |
| Psychological responses | The mental and emotional reactions of individuals to stimuli, which can affect how they interact with technology and how systems are designed to support them. |
| Reinforcement learning | A type of machine learning where an agent learns to make decisions by taking actions in an environment to maximise cumulative reward, based on feedback from its actions. |
| Robotic process automation (RPA) | Technology that uses software robots to automate repetitive and rule-based tasks typically performed by humans in business processes. |
| Safety of workers | The consideration of how automation and AI technologies impact the health, safety and job security of workers in various industries. |
| Semi-supervised learning | A hybrid approach that combines supervised and unsupervised learning, using a small amount of labelled data alongside a larger amount of unlabelled data. |
| Significance and impact of ML and AI | The importance and effects of machine learning and artificial intelligence technologies on various aspects of life, including individuals, society and the environment. |
| Software automation | The use of technology to perform tasks with minimal human intervention, improving efficiency and consistency in processes. |
| Supervised learning | A type of machine learning where the model is trained on a labelled dataset, meaning that the algorithm learns from input-output pairs. |
| Unsupervised learning | A type of machine learning where the model is trained on data without labelled responses, allowing the algorithm to identify patterns and relationships in the data. |
| Virtual personal assistants | AI systems that assist users in performing tasks through voice or text commands, utilising natural language processing and machine learning. |

**Teacher note:** for students learning English as an additional language or dialect (EAL/D), the completed glossary can be provided so that they have additional time to understand the key terms with bilingual dictionaries. The glossary can be provided to students in their preferred communication mode. Teachers may consider using [semantic wave (PDF 422 KB)](https://static.raspberrypi.org/files/curriculum/quickreads/6-Pedagogy_Summary_Semantic_Waves_V3_2023.pdf) strategies to explore the meaning of these concepts.

## NESA glossary keywords

NESA keywords can be used in the syllabus and in the Higher School Certificate examination. Familiarisation with these keywords can assist in understanding how to write and respond to questions.

Table 5 – NESA glossary keywords

|  |  |
| --- | --- |
| Key term | Definition |
| Apply | Use, utilise, employ in a particular situation. |
| Describe | Provide characteristics and features. |
| Explain | Relate cause and effect; make the relationships between things evident; provide why and/or how. |
| Investigate | Plan, inquire into and draw conclusions about. |

[NESA: Glossary of key words](https://educationstandards.nsw.edu.au/wps/portal/nesa/11-12/hsc/hsc-student-guide/glossary-keywords)

**Teacher note:** ‘develop’, ‘explore’, ‘select’ and ‘verify’ are verbs used in this topic though not listed in NESA’s Glossary of keywords.

Table 6 – key terms and definitions

|  |  |
| --- | --- |
| Key term | Definition |
| Develop | Grow or cause to grow, improve or advance in skills, knowledge or processes over time. |
| Explore | Investigate or examine something in detail, with curiosity and discovery. |
| Select | Carefully choose or pick out from a group based on specific criteria or preferences. |
| Verify | Confirm the accuracy or truth of something, often through investigation or evidence. |

# Software automation

Software automation refers to the use of technology to perform tasks and processes with minimal human intervention.

Software automation uses algorithms, machine learning (ML) and automation techniques such as DevOps, robotic process automation (RPA) and business process automation (BPA) to enhance efficiency, accuracy and productivity in software development and operational processes.

## Algorithms in machine learning

### Investigate how machine learning (ML) supports automation through the use of DevOps, robotic process automation (RPA) and business process automation (BPA)

**Teacher note**: students familiarise themselves with subject-specific terminology used in the Software Engineering course. Activities to help students understand these terms include research, cloze passages, predicting upcoming sections of text read by the teacher and engaging with videos provided by industry or academia while actively taking notes. In the digital examination Higher School Certificate, students will be required to respond to stimulus including video. Teachers can facilitate the development of this skill by playing videos with closed captions and by printing copies of the video transcripts to read, highlight and underline to reinforce the understanding acquired from the videos. The activities in this TSR should be delivered alongside the demonstration and discussion of the variety of ways to make notes, helping students to become independent learners. Where students need assistance to build understanding on unfamiliar content, teachers could recite the sample answers provided and require students to build their own notes with these responses. Application of the concepts and terminology via practical activities will consolidate understanding. Regular quizzes of the last lesson’s work will enhance understanding.

#### What is machine learning (ML)

**Activity 1**: students watch [Machine Learning Explained in 100 Seconds (2:34)](https://youtu.be/PeMlggyqz0Y?si=5eMQP6PfhILW8ZeS) and [What is Machine Learning? (2:19)](https://youtu.be/f_uwKZIAeM0?si=eEgbCgIv-8LbAHcX)before responding to the questions below.

1. Provide a definition of machine learning in the space provided.

|  |
| --- |
| **Machine learning (ML)** is a branch of artificial intelligence (AI) that focuses on developing algorithms and models that enable computers to learn from data and make predictions or decisions without being explicitly programmed for each task. It’s about teaching machines to recognise patterns and improve their performance over time based on experience. Instead of writing specific instructions for every task, we provide the computer with data and let it figure out how to perform the task on its own. |

1. Provide an explanation of how machine learning works using the keywords: **data**, **algorithms**, **training** and **testing** in the space provided below.

|  |
| --- |
| The foundation of machine learning is data. This could be anything from numbers and text to images and audio. The more relevant data you have, the better the machine can learn. Machine learning algorithms are mathematical procedures that the computer uses to learn from data. Different algorithms are suited for different types of tasks. During the training phase, the machine learns from a dataset by finding patterns and relationships within the data. It uses this information to make predictions or decisions. After training, the model is tested with new data to evaluate its performance. This step checks how well the model can generalise its learning to unseen data. |

1. Provide a description of the main types of machine learning in the space provided.

|  |
| --- |
| **Supervised learning**: a model is trained on a labelled dataset, meaning that the data includes both the input and the correct output (label). The model learns to predict the output based on the input data. For example, predicting house prices based on features like size and location.  **Unsupervised learning**: here, the model is trained on data without labelled outputs. It tries to find patterns or groupings in the data on its own. For example, clustering customers based on purchasing behaviour without prior labels.  **Reinforcement learning**: in this approach, the model learns by interacting with an environment. After a series of steps, it receives feedback in the form of rewards or penalties. Using this feedback, the model is updated. This process of interacting with the environment and updating the model based on feedback is repeated many times until the model makes optimal decisions. Reinforcement learning is often used in scenarios like gameplaying or robotics. |

1. Provide a description of where machine learning is applied in the space provided.

|  |
| --- |
| * Image recognition: ML algorithms can identify objects, people and scenes in images, such as those used in facial recognition systems. * Natural language processing (NLP): this allows computers to understand and interpret human language, enabling applications like chatbots and language translation. * Recommendation systems: services like Netflix and Amazon use ML to recommend movies or products based on user preferences and behaviours. * Fraud detection: financial institutions use ML to analyse transactions and detect suspicious activity that may indicate fraud. |

1. Identify some of the challenges in using machine learning in the space provided.

|  |
| --- |
| * Quality of data: the success of an ML model heavily depends on the quality and quantity of the data used for training. * Overfitting: this occurs when a model learns the training data too well, including noise and outliers, making it perform poorly on new data. * Bias: if the training data is biased or unrepresentative, the model can perpetuate those biases in its predictions or decisions. |

**Activity 2**: students complete the following cloze passage summary of machine learning: Fill in the blanks with the appropriate words from the list below:

* technology
* data
* algorithms
* decisions
* explicitly
* experience
* transform
* industries
* applications

Machine learning is a powerful \_\_\_\_\_\_\_\_\_\_ that enables computers to learn from \_\_\_\_\_\_\_\_\_\_ and make \_\_\_\_\_\_\_\_\_\_ without being \_\_\_\_\_\_\_\_\_\_ programmed for every task. By understanding \_\_\_\_\_\_\_\_\_\_, applying \_\_\_\_\_\_\_\_\_\_ and continuously improving through \_\_\_\_\_\_\_\_\_\_, ML has the potential to \_\_\_\_\_\_\_\_\_\_ various \_\_\_\_\_\_\_\_\_\_ and \_\_\_\_\_\_\_\_\_\_.

**Answer**

Machine learning is a powerful technology that enables computers to learn from data and make decisions without being explicitly programmed for every task. By understanding data, applying algorithms and continuously improving through experience, ML has the potential to transform various industries and applications.

**Teacher note**:

The following jigsaw activities enable students to distinguish between DevOps, business process automation (BPA), robotic process automation (RPA) and MLOps. These are closely related concepts within process improvement and automation but serve different purposes and have distinct characteristics.

DevOps is a set of [practices](https://www.atlassian.com/devops/what-is-devops/devops-best-practices), [tools](https://www.atlassian.com/devops/devops-tools/choose-devops-tools) and a [cultural philosophy](https://www.atlassian.com/devops/what-is-devops/devops-culture) that automate and integrate the processes between software development and IT teams. It emphasises team empowerment, cross-team communication and collaboration, and technology automation.

BPA refers to the use of technology to automate complex business processes and functions beyond just individual tasks. It involves the integration and optimisation of various processes within an organisation to improve efficiency, reduce costs and enhance productivity.

RPA is a specific form of automation that uses software robots (or ‘bots’) to automate repetitive, rule-based tasks typically performed by humans. It focuses on automating individual tasks within existing workflows, often interacting with user interfaces in the same way that a human would. RPA can be considered a subset of BPA. BPA encompasses broader process automation initiatives involving strategic changes and integrated systems. RPA specifically focuses on automating individual tasks within those processes.

RPA is a tool within a BPA framework, allowing organisations to achieve automation of specific tasks as part of a larger effort to optimise and automate business processes. By automating repetitive tasks with RPA, businesses can free up human resources to focus on more strategic activities and process improvements.

MLOps (machine learning automation through DevOps) is the automated process of designing, training and deploying machine learning models. It borrows many of the same principles and practices used in DevOps, bringing together the teams involved in developing machine learning models and the operational teams involved in deploying and supporting the models in production. Upon completion of the jigsaw activities, students apply their knowledge to the school library scenario. Other scenarios are provided for discussion.

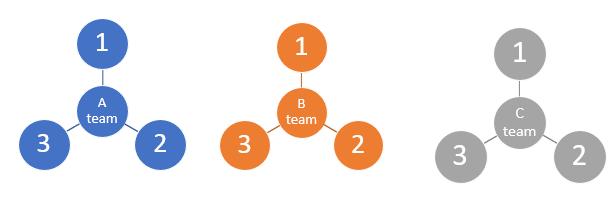
**Activity 3**: Jigsaw

Jigsaw activities are an effective way to build collaboration and cover large amounts of content in an efficient and engaging way. Jigsaw activities necessitate students work alongside classmates they may not otherwise work with. This is both closer to real-world industry practice and builds collegiality in the classroom. Teachers are advised to change teams in each jigsaw activity to maximise interaction between different students and approximate industry practice. This example uses 9 students. Groups can be smaller (2 students each) or bigger (4 students each) or approximated to any other multiples based on the size of the class.

**Step 1:**

The class is separated into teams. Each team has 3 students. These teams are labelled A, B, C and so on. The teacher numbers each member of the team: 1, 2 or 3.

Figure 1 – create teams



**Step 2:**

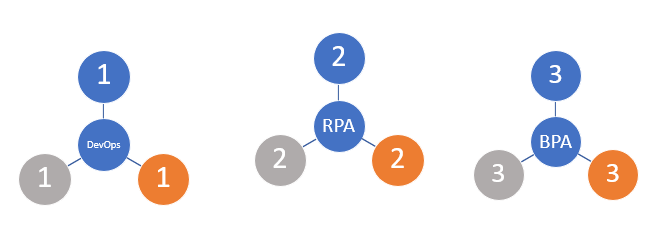
All ones then form a group responsible for researching DevOps.

All twos then form a group responsible for researching robotic process automation (RPA).

All the threes then form a group responsible for researching business process automation (BPA).

Each group brainstorms and researches their topic including a definition (What is it?) and key concepts (how it works).

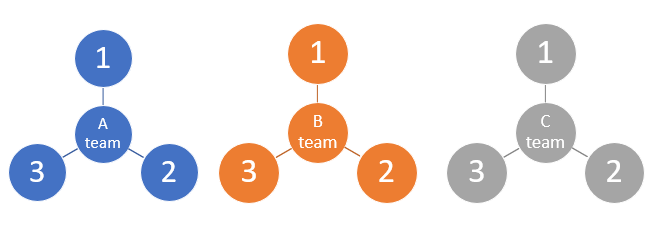
Figure 2 – create groups



**Step 3:**

Groups reform into teams (A, B, C and so on) to combine each of their findings into a single presentation that summarises: ‘how machine learning (ML) supports automation through the use of DevOps, robotic process automation (RPA) and business process automation (BPA)’.

Figure 3 – re-form teams



**Step 4:**

Teams apply their understanding to a scenario they are familiar with, for example ‘The school library upgrade with machine learning’. Teams compile research notes into a slide deck for course notes.

**Activity 3.1**: DevOp group watch [What is DevOps in Simple English? (7:06)](https://youtu.be/_I94-tJlovg?si=ouWHJoACaO-anDzm) and [What is DevOps? REALLY understand it | DevOps vs SRE (35:32)](https://youtu.be/0yWAtQ6wYNM?si=bDoj0ZWVbIFanaJ3).

The world-leading DevOps software tool is a Sydney-based software company called Jira by Atlassian.

1. DevOp group – provide a definition of DevOp in the space provided below.

|  |
| --- |
| DevOps is a set of practices and cultural philosophies that aim to improve collaboration and communication between software development (Dev) and IT operations (Ops) teams.  The goal of DevOps is to shorten the development lifecycle, increase the frequency of software releases and improve the quality and reliability of the software being delivered. |

1. DevOp group – identify the key concepts of DevOps in the space provided below.

|  |
| --- |
| Traditional software development often involves separate teams working in silos. Developers create software, while operations teams manage the infrastructure and deployment. In DevOps, these teams work together throughout the entire software development process. This **collaboration** leads to better understanding, shared goals and quicker problem-solving. DevOps emphasises **automation** of repetitive tasks such as code testing, integration, deployment and monitoring. This reduces human error, speeds up processes and allows teams to focus on more complex tasks. Developers frequently integrate their code changes into a **shared repository**. Automated tests are run to ensure that new code does not break existing functionality. After passing the tests, code changes are **automatically deployed** to production (the live environment) with minimal manual intervention. This allows for faster delivery of new features and bug fixes. DevOps encourages continuous **monitoring** of applications and infrastructure. This helps teams identify issues quickly and gather feedback on performance and user experience. **Feedback** loops enable teams to make informed decisions and continuously improve their software and processes. A DevOps **culture** promotes **experimentation**, learning from failures and adapting based on feedback. Teams are encouraged to take risks and innovate without the fear of repercussions from mistakes. This culture fosters a growth mindset, where teams continuously improve their skills and processes. |

1. DevOp group – describe the benefits of DevOps in the space provided below.

|  |
| --- |
| Teams can release software updates and new features more **quickly**, responding to user needs and market demands. Automation and continuous testing **reduce bugs** and improve the overall **quality** of software. Teams **collaborate** more effectively, leading to improved communication and shared responsibility for the software. **Automation** of repetitive tasks allows teams to focus on more valuable work, reducing time spent on manual processes. |

1. DevOp group – provide a summary of DevOps for course notes and a slide deck in the space provided below.

|  |
| --- |
| DevOps is about breaking down barriers between development and operations teams to create a more efficient, collaborative and automated process for building, testing and deploying software. By embracing DevOps practices, organisations can deliver high-quality software faster and respond better to user needs and changes in the market. It’s a way of thinking that prioritises communication, automation and continuous improvement! |

#### What is RPA?

**Activity 3.2**: RPA group – watch and take notes on the following videos:

* [RPA in 5 Minutes | What Is RPA -– Robotic Process Automation? (5:34)](https://youtu.be/9URSbTOE4YI?si=OFIy6WNFsrAYkSp4)
* [Introduction to Robotic Process Automation (RPA) (4:39)](https://youtu.be/VKsHgNMsZqo?si=QkTH3Dma4hyBaDi7)
* [What is RPA (Robotic Process Automation)? | Automation Anywhere (7:45)](https://youtu.be/61K7JeJixnU?si=WoLzQSRFErT9jNKE)

1. RPA group – provide a definition of robotic process automation (RPA) in the space provided below.

|  |
| --- |
| Robotic process automation (RPA)is a technology that allows software robots (or ‘bots’) to automate repetitive and rule-based tasks that are typically performed by humans on a computer. RPA is designed to help businesses improve efficiency, reduce errors and free up human workers to focus on more complex and value-added activities. These software robots mimic human actions in digital systems. They can interact with applications and systems just like a human would, performing tasks such as data entry, data extraction and processing transactions. |

1. RPA group – provide an explanation of how robotic process automation (RPA) works in the space provided below.

|  |
| --- |
| RPA systems develop action lists by watching the user perform tasks in the application's graphical user interface (GUI) and then perform the automation by repeating those tasks directly in the GUI. The important difference is that RPA bots are not pre-programmed with what actions to do but learn by observing what a user does. It can perform tasks like:   * logging into applications: just like a human user would * navigating through screens: clicking buttons, entering data and making selections * copying and pasting data: moving information from one application to another * reading and interpreting data: extracting information from documents or databases * generating reports: compiling data and creating summaries. |

1. RPA group – provide detail on the benefits of robotic process automation (RPA) in the space provided below.

**Teacher note**: this question should be posed using the [5 whys](https://app.education.nsw.gov.au/digital-learning-selector/LearningActivity/Card/638) strategy to extract more thoughtful responses.

|  |
| --- |
| Bots can work 24/7 without breaks, completing tasks much faster than humans, increasing efficiency. By following consistent rules, RPA minimises human errors that can occur during manual data entry or processing. Automating repetitive tasks can reduce labour costs and allow employees to focus on more strategic work. RPA can be easily scaled up or down based on the organisation's needs, making it flexible for changing workloads. |

1. RPA group – provide common cases where robotic process automation (RPA) is used in the space provided below.

|  |
| --- |
| * Automating the entry of data into databases or spreadsheets * Extracting information from invoices and entering it into accounting systems * Automating responses to common customer inquiries through chatbots * Compiling data from various sources and generating reports automatically |

1. RPA group – suggest a process that you (or your teacher) find frustrating that could be automated in the space provided below.

|  |
| --- |
| For teachers marking rolls, writing reports, filling in other administrative paperwork could be frustrating. Likewise, students may identify administrative type routines that are frustrating in the school or casual work lives. |

1. RPA group – provide a summary of RPA for course notes and a slide deck in the space provided below.

|  |
| --- |
| Robotic process automation (RPA) is a powerful technology that automates routine and repetitive tasks performed by humans on computers. By using software robots, RPA can enhance efficiency, reduce errors and allow employees to focus on more complex and valuable work. As businesses increasingly adopt RPA, it plays a crucial role in streamlining operations and improving productivity across various industries. RPA represents an exciting intersection of technology and business, showcasing how automation can transform the way work is done. |

#### What is BPA?

**Activity 3.3:** BPA group – watch and take notes on the following videos:

* [Why You Need to Automate Processes in Business & Life (11:27)](https://youtu.be/TparimIBWtU?si=oOD7Wb1jwpHvyKFl)
* [What is Business Process Automation? Pros, Cons, Mysths & Tips (3:51)](https://www.youtube.com/watch?v=MM20QXEGdAc)

1. BPA group – provide a definition of business process automation (BPA) in the space provided below.

|  |
| --- |
| BPA refers to the use of software and technology to automate workflows, processes, and tasks within an organisation. It involves the integration of applications, restructuring of labour resources, and the use of digital tools to improve business performance. |

1. BPA group – provide an explanation of how BPA works in the space provided below.

|  |
| --- |
| BPA involves several steps. Understanding and mapping out existing business processes to identify areas for automation, then designing automated workflows using software tools that define how tasks will be executed. Various systems and applications are connected to work together seamlessly. Automated processes are continuously monitored to ensure they are functioning correctly and making adjustments as needed. |

1. BPA group – provide detail on the benefits of BPA. This question should be posed using the 5 whys strategy to extract more thoughtful responses in the space provided below.

|  |
| --- |
| By automating processes, tasks can be completed faster and with fewer errors. This allows employees to focus on more strategic and creative activities rather than repetitive work. Automating processes can lead to significant cost reductions by minimising labour costs and reducing the time spent on manual tasks. Automated processes are more consistent than human-performed tasks, which can lead to fewer mistakes and higher quality output. BPA can help ensure that processes adhere to regulations and standards by maintaining consistent workflows and automatically logging actions taken. |

1. BPA group – provide common cases where BPA is used in the space provided below.

|  |
| --- |
| Automating the collection, approval and payment of invoices can streamline the accounts payable process, reduce errors and speed up payments. Automating the onboarding process for new employees, including filling out forms, training schedules and equipment provisioning, can enhance the experience and ensure consistency. Using chatbots to handle common customer inquiries can free up human agents to deal with more complex issues. Automating email marketing campaigns, social media posts and customer segmentation can enhance marketing efficiency and effectiveness. |

1. BPA group – provide a summary of BPA for course notes and a slide deck in the space provided below.

|  |
| --- |
| Business process automation (BPA) involves using technology to streamline and automate complex business processes to improve efficiency, reduce costs, and enhance accuracy. By automating workflows, organisations can free up human resources for more strategic tasks, ensure consistent quality, and respond more quickly to changes. BPA represents an exciting area of technology that combines process improvement, software development and business strategy to drive efficiency in organisations. |

**Teacher** **note:** Activities 4 to 10:

Teams will be required to refer to NESA’s Course specifications (machine learning automation through DevOps) for the Software Engineering 11–12 Syllabus and apply what they have learned about DevOps to the following scenario.

**The school library upgrade with machine learning**

A school community wants its library to improve services and make things easier for students and staff. The library has many processes, like checking out books, managing inventory and handling requests for new books. Improvements can be made through software automation of these processes.

**Activity 4**: students complete the cloze passage below to describe the following scenario: ‘The school library upgrade with machine learning’.

**Word list**: DevOps, BPA, RPA, ML

**Scenario**

A school community wants its library to improve services and make things easier for students and staff. The library has many processes, include checking out books, managing inventory and handling requests for new books. The principal and school executive decide to form a special team called the ‘Library improvement team’ to work on making the library more efficient.

This team includes librarians, IT staff and students. They use a practice called \_\_\_\_\_, which combines development (the people who create new systems) and operations (the people who manage them). The team works together to create a new online system for checking out books. They meet regularly to share ideas, solve problems and make sure the system is user-friendly. This collaboration helps them quickly improve the library's services based on feedback from students and staff.

The library improvement team notices that checking out books and managing inventory takes a lot of time and effort. To solve this, they decide to implement \_\_\_\_.

The team creates an automated system that allows students to check out books online. When a student checks out a book, the system automatically updates the inventory, sends a notification to the librarian and records the transaction. This automation simplifies the process, reduces waiting time and makes it easier for everyone.

Even with the new automated system there are still some repetitive tasks that take up a lot of time, like entering data about new books into the system. The team decide to use \_\_\_\_ to help with these specific tasks. The team uses \_\_\_\_ to create a ‘robot’ that automatically enters data about new books into the library system from a spreadsheet provided by publishers. Instead of a librarian having to do this manually, the robot completes the task quickly and accurately. This allows librarians to focus on helping students instead of doing repetitive data entry.

To further enhance the library's services, the library improvement team decides to use \_\_\_\_. They want to make the library more responsive to student needs and preferences. The team gathers data on which books are borrowed most often, what genres students enjoy and how frequently certain books are requested. They use this data to train a machine learning model that can predict which books students are likely to want in the future. The ML model creates a recommendation system that suggests books to students based on their previous borrowing history and the preferences of similar users. When students log into the library system, they see personalised recommendations, making it easier for them to find books they will enjoy. Additionally, the ML model helps librarians manage inventory by predicting which books are likely to be checked out soon. This allows the library to adjust its stock and ensure popular titles are always available.

**Teacher note**: compared to the use of MLOps in big business this example may appear trivial however it allows students to apply their understanding to a scenario with which they are familiar.

**Activity 5**: teachers cut up the table to separate the term from the description.

Students match the term with the description based on the scenario.

Table 7 – school library upgrade – terms and descriptions

|  |  |
| --- | --- |
| Term | Description |
| DevOps | Fosters collaboration among different roles to create better systems. |
| BPA | Automates entire processes to save time and improve efficiency. |
| RPA | Automates specific, repetitive tasks to free up human resources for more valuable work. |
| ML | Provides insights and recommendations that enhance user experience and improve inventory management. |

**Activity 6**: using colour coding, students identify the 3 stages of MLOps used in the ‘The school library upgrade with machine learning’ scenario.

MLOps is the automated process of designing, training and deploying machine learning models. It borrows many of the same principles and practices used in DevOps, bringing together the teams involved in developing machine learning models and the operational teams involved in deploying and supporting the models in production.(NESA 2023)

Figure 4 – MLOPs cycle

Diagram depicting the interaction between stages of the machine learning operations cycle.



#### © NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2023.

#### The 3 stages of MLOps

**Design**

* Defining the business problem to be solved
* Refactoring the business problem into a machine learning problem
* Defining success metrics
* Researching available data

**Model development**

* Data wrangling\*
* Feature engineering
* Model training
* Model testing and validation

**Operations**

* Model deployment
* Supporting operations/use
* Monitoring model performance

**Extension**: students applying MLOps to their project consider the [levels of MLOps](https://github.com/TempeHS/Practical-Application-of-NESA-Software-Engineering-MLOps?tab=readme-ov-file).

**Activity 7**: students identify the MLOps stages in the ‘The school library upgrade with machine learning’ scenario.

Table 8 – Design stage

|  |  |
| --- | --- |
| **Define the problem to be solved** | The library wants to improve services for students and staff, specifically in checking out books, managing inventory, and handling requests for new books. |
| **Reframe the problem into a machine learning problem** | The team refines the problem into a machine learning context by identifying the need to predict which books students will likely want in the future based on their borrowing history and preferences. |
| **Define measures of success** | Success metrics could include the accuracy of book recommendations, user satisfaction scores, and the reduction in the time taken for librarians to manage inventory and check out processes. |
| **Research available data** | The team gathers data on borrowing patterns, genres preferred by students, frequency of requests for new books, and other relevant data that can help train the ML model. |

Table 9 – Model development stage

|  |  |
| --- | --- |
| **Wrangle the data** | The team cleans and prepares the data collected from various sources, such as library records and user feedback, ensuring it is suitable for analysis.  Data wrangling involves removing null and duplicate data and normalising the data to improve the speed and accuracy of the fit. |
| **Engineer new features** | The team creates new features from the data, such as categorising books by genre, identifying peak borrowing times, and generating user profiles based on borrowing history. They calculate the average count of borrows in each genre to improve the accuracy of recommender system |
| **Train the model** | Using the prepared data, the team trains a machine learning model to predict which books students are likely to check out based on their past behaviour and preferences. |
| **Test and validate the model** | The team tests the model using a validation dataset to ensure its predictions are accurate and reliable. They may adjust model parameters based on its performance in this phase. |

Table 10 – Operation stage

|  |  |
| --- | --- |
| **Implement the model** | The trained machine learning model is integrated into the library's online system, allowing it to make real-time recommendations to students when they log in. |
| **Support operations and use** | The library improvement team provides support for using the system, ensuring that librarians and students can easily access and benefit from the recommendations. |
| **Monitor the performance of the model:** | Using the prepared data, the team trains a machine learning model to predict which books students are likely to check out based on their past behaviour and preferences. |
| **Test and validate the model** | The team continuously monitors the model's performance by tracking user engagement with recommendations, feedback from students, and adjustments to inventory based on predictions. If the model's accuracy declines or user satisfaction decreases, they will investigate and refine the model as needed. |

**Activity 8**: students visit [What is MLOps?](What%20is%20MLOps?) to contrast and compare MLOps and DevOps.

**Contrast**

Table 11 – key differences between MLOps and DevOps

|  |  |
| --- | --- |
| MLOps vs DevOps | ****Differences**** |
| ****Scope**** | **DevOps focuses on the software development life cycle.**  **MLOps focuses on the ML life cycle.** |
| ****Complexity**** | **ML models are often more complex than traditional software applications, requiring specialized tools and techniques for development and deployment.** |
| ****Data**** | **ML models rely on data for training and inference, which introduces additional challenges for managing and processing data.** |
| ****Regulation**** | **ML models may be subject to regulatory requirements, which can impact the development and deployment process.** |

**Compare**

|  |
| --- |
| MLOps and DevOps share some common principles, such as the importance of collaboration, automation and continuous improvement. |

**Activity 9**: students list and describe other scenarios from school-based systems and processes that could use DevOps, robotic process automation (RPA), business process automation (BPA) and machine learning (ML) to support automation in their school environment.

Students complete Table 12 on the following page.

**Teacher note**: students may consider developing one of these scenarios into a solution for the Software engineering project.

Table 12 – automation in school-based systems and processes matrix

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Scenarios | DevOps | BPA | RPA | ML |
| Student enrolment process | A cross-functional team collaborates to develop and deploy an online enrolment system that integrates with existing student databases. | Automate the enrolment process by allowing students to fill out forms online, which then automatically updates the school’s database and sends confirmation emails. | Use bots to extract data from paper forms or email submissions and input it into the enrolment system. | Implement predictive analytics to forecast enrolment numbers based on historical data, helping the school plan for resources accordingly. |
| Grade management system | Teams work together to continuously improve a digital grading system that integrates with learning management software. | Automate the process of grade calculations and reporting, allowing teachers to focus on instruction instead of paperwork. | Use bots to gather grades from different sources (like tests, assignments, and projects) and compile them into a single report automatically. | Develop algorithms to identify trends in student performance and provide insights for teachers to offer targeted support. |
| Attendance tracking | Continuous collaboration between administration and IT to enhance attendance tracking software. | Automate attendance recording through an online system where teachers can mark attendance digitally. | Use bots to process attendance data, generating reports and notifying parents of absences automatically. | Analyse attendance patterns to identify students at risk of dropping out and flag them for intervention. |
| Event planning and coordination | A team collaborates to develop a school event management platform that integrates with calendars and communication tools. | Automate the process of scheduling events, sending invites and managing RSVPs through an online portal. | Use bots to compile event feedback forms and generate summary reports for future planning. | Analyse past event data to determine which types of events are most popular with students and adjust future planning accordingly. |
| Homework and assignment submissions | Teachers and IT staff collaborate to improve a homework submission platform that integrates with grading systems. | Automate the homework submission process, allowing students to upload assignments directly online. | Implement bots to check for plagiarism or format issues in submitted assignments and notify teachers of any concerns. | Use machine learning to analyse assignment submissions and provide personalised feedback to students based on common errors. |
| Parent-teacher communication | Teams work together to build and maintain a communication platform for parents and teachers. | Automate the scheduling of parent-teacher conferences and send reminders to parents automatically. | Use bots to gather reports on student performance and send them to parents through email or a dedicated app. | Analyse communication patterns to identify when parents are most responsive and optimise communication strategies accordingly. |
| Resource allocation and budgeting | Finance and administration teams collaborate to create a budgeting tool that integrates with school finance systems. | Automate the process of tracking resource usage (like classroom supplies or technology) and generating budget reports. | Use bots to collect data from multiple sources (for example, purchase orders, invoices) and compile it for financial reporting. | Implement predictive analytics to forecast future resource needs based on past spending and usage trends. |
| Other |  |  |  |  |

### Distinguish between artificial intelligence (AI) and ML

**Activity 10**: students watch [AI vs Machine Learning (5:48)](https://youtu.be/4RixMPF4xis?si=wN3-nvW4uIZQ_Z0J) and make notes on the key differences between AI and ML.

1. Summarise the key concepts in the video in the space provided below.

|  |
| --- |
| **Definitions**  Artificial intelligence (AI) is the capability of a machine to mimic human intelligence and perform tasks that typically require human intelligence, such as reasoning, discovering new information, and inferring data from various sources.  Machine learning (ML) is a subset of AI that focuses on the ability of machines to learn from data and make predictions or decisions without being explicitly programmed.  **The relationship between AI and ML**  **AI vs ML**: instead of thinking of AI and ML as opposing concepts, they should be seen as interconnected.  **Subset vs superset**: machine learning is a subset of AI, meaning that all ML is AI, but not all AI is ML. AI encompasses a broader range of technologies and applications beyond ML.  **Types of machine learning**   * Supervised learning involves training a model on a labelled dataset where the human oversight provides guidance through labels. * Unsupervised learning is where the model learns from data without explicit labels, discovering patterns or relationships on its own. * Deep learning (DL) is a subfield of ML that uses neural networks with multiple layers (deep networks) to model complex relationships in data. While deep learning can provide valuable insights, it may lack transparency in how its derived. |

**Activity 11**: create a diagram to explain the relationship between AI and ML.

|  |
| --- |
| Sample answer explaining the relationship between AI and ML. |

**Activity 12**: students visit [Difference Between Machine Learning and Artificial Intelligence](https://www.geeksforgeeks.org/difference-between-machine-learning-and-artificial-intelligence/) and [Artificial intelligence (AI) vs. machine learning (ML)](https://cloud.google.com/learn/artificial-intelligence-vs-machine-learning) to research the difference between AI and ML. They are presented with an incomplete table (in bold) and are to suggest a distinguishing feature for the other technology.

Classifications of areas and subsets that comprise AI can also be found at [AI4K12](https://ai4k12.org/).

Table 13 – artificial intelligence versus machine learning

|  |  |
| --- | --- |
| Artificial intelligence | Machine learning |
| **A technology which enables a machine to simulate human behaviour.** | A subset of AI which allows a machine to automatically learn from past data without programming explicitly. |
| The goal is to make a smart computer system, like humans, to solve complex problems. | **The goal is to allow machines to learn from data so that they can give accurate output.** |
| **Intelligent systems to perform a task like a human.** | Teaching machines with data to perform a particular task and give an accurate result. |
| Machine learning is a subset of AI, deep learning is a subset of ML. | **Deep learning is a main subset of machine learning.** |
| **has a very wide range of scope** | has a limited scope |
| **is concerned about maximizing the chances of success** | concerned about accuracy and patterns |
| Applications include Siri, customer support using chatbots, expert system, online game playing, intelligent humanoid robot and so on. | **Applications include online recommender system, Google search algorithms, Facebook auto friend tagging suggestions and so on.** |
| **Two types:** general (strong) and narrow (weak) | Three types: **supervised learning, unsupervised learning and reinforcement learning.** |
| includes learning, reasoning and self-correction. | **includes learning and self-correction when introduced with new data.** |
| **Deals with structured, semi-structured and unstructured data.** | Deals with structured and semi-structured data. |

### Explore models of training ML

**Including: supervised learning, unsupervised learning, semi-supervised learning, reinforcement learning**

[Google’s Teachable Machine](https://teachablemachine.withgoogle.com/) uses a form of **supervised learning** for training machine learning (ML) models. This approach is particularly user-friendly and designed to allow users, even those without extensive programming or ML expertise, to create models easily.

It is an engaging entry point for students to consider the power and potential of machine learning while considering how it compares with and differs from the models included in the syllabus.

**Teacher note**: for students that have enjoyed the Year 11 Mechatronics unit there are many real-world projects that use (for example) an Arduino and Servos along with the teachable machine to sort based on image recognition. Students could consider this a model to controlling stock with gates or sorting seeds for planting.

Students may be inspired by [Google labs](https://labs.google/) or [Experiments with Google](https://experiments.withgoogle.com/experiments).

Google’s Teachable Machine uses **transfer learning** to allow users to train machine learning models in an accessible and efficient manner. By building upon pre-trained models and providing a simple interface for training, it empowers users to create custom ML models for various applications without requiring deep technical knowledge in machine learning.

**Activity 13**: ****[Google’s Teachable Machine](https://teachablemachine.withgoogle.com/) (group activity)

**Step 1**: students form teams of 3.

**Step 2**: students watch [Teachable Machine Tutorial 1: Gather (2:15)](https://youtu.be/DFBbSTvtpy4?si=u9ts10yqtWJlDVZw).

**Step 3**: students open[Teachable Machine](https://teachablemachine.withgoogle.com/).

**Step 4**: teams gather and group examples from the webcam, microphone or upload images into classes or categories, that they want the computer to learn.

* One student uses images
* One student uses sounds
* One student uses poses

**Step 5**: students watch [Teachable machine Tutorial 2: Train (0:54)](https://youtu.be/CO67EQ0ZWgA?si=T5T-6R9zvjZgOEaR).

**Step 6**: students watch[Teachable machine Tutorial 3: Export (0:59)](https://youtu.be/n-zeeRLBgd0?si=XL_NzJYncJFgL8Xe).

**Step 7**: students improve their models by further training and investigating ‘under the hood’.

**Activity 14**: students watch and take notes on the following videos.

**Teacher note**: these videos could also be delivered as homework using a [flipped classroom](https://www.teaching.unsw.edu.au/flipped-classroom) approach.

* Supervised learning
* [Supervised Learning: Crash Course AI #2 (15:22)](https://youtu.be/4qVRBYAdLAo?si=2K6fEBcCKNkXhrkG)
* [Machine Learning Basics: Supervised v Unsupervised (6:12)](https://youtu.be/3fsy2oheRdg?si=kM2SEvwHRiTTd2Pi)
* [Supervised & Unsupervised Machine Learning (11:45)](https://youtu.be/wvODQqb3D_8?si=ucI9X_TTeE1w7zCI)
* Unsupervised learning
* [Unsupervised Learning: Crash Course AI #6 (12:34)](https://youtu.be/JnnaDNNb380?si=kI9ALARb8iQSBdOW)
* Semi-supervised learning
* Reinforcement learning
* [Reinforcement Learning: Crash Course AI #9 (11:27)](https://youtu.be/nIgIv4IfJ6s?si=NmeVph0T6iJBAYpY)

**Extension**: [A Machine Learning Primer: How to Build an ML Model (13:47)](https://youtu.be/Vx2DpMgplEM?si=S6P_XTjlGwfHUv3t)

**Activity 15**: students work through Google’s foundational course on machine learning by completing the [Introduction to Machine Learning](https://developers.google.com/machine-learning/intro-to-ml).

**Activity 16**: students apply their understanding from the videos and Google’s [Introduction to Machine Learning](https://developers.google.com/machine-learning/intro-to-ml) to complete the following table, describing each of the learning models.

Table 14 – learning models

|  |  |  |
| --- | --- | --- |
| Learning model | Definition | Example |
| Supervised | The model is trained on a labelled dataset, meaning that each training example comes with the correct output (label). The model learns to map inputs to outputs based on this labelled data. | Image classification, where you might train a model to recognise pictures of cats and dogs using a dataset where each image is labelled as ‘cat’ or ‘dog’. |
| Unsupervised | The model is trained on data without any labels. The goal is to identify patterns or groupings in the data without knowing the correct outputs beforehand. | Clustering customers based on purchasing behaviour is a typical application of unsupervised learning, where the model tries to group similar customers together without predefined categories. |
| Semi-supervised | A middle ground between supervised and unsupervised learning. It involves using a small amount of labelled data alongside a larger amount of unlabelled data. This approach helps improve model performance while reducing the need for extensive labelled datasets. | You might have a few labelled images of cats and dogs but a large collection of unlabelled images. The model can learn from both the labelled and unlabelled data to improve its classification abilities. |
| Reinforcement | A type of learning where an agent learns to make decisions by interacting with an environment. The agent receives rewards or penalties based on its actions and learns to maximise the cumulative reward over time. | Training an AI to play a game. The AI learns which moves are beneficial (rewarding) and which are detrimental (penalising) based on the outcomes of its actions. |

**Activity 17**: students summarise the completed Table 14 above.

|  |
| --- |
| Google Teachable Machine employs a supervised learning approach through transfer learning. This is different from unsupervised learning, semi-supervised learning and reinforcement learning based on how data is used, and on the goals of the learning. Each type of learning has its own purpose and way of working, but they are all part of machine learning. |

**Extension**: role-play. Students are placed into 4 teams.

1. **Supervised learning:** learning from labelled data to predict outcomes.

One student acts as the ‘teacher’ (providing labels) and others as ‘data points’ (labelled examples). The teacher shows a set of fruits (for example, apples, oranges) with labels. Students must learn to identify these fruits based on the provided labels and then classify new, unlabelled fruits correctly.

1. **Unsupervised learning:** finding patterns or groupings in unlabelled data. Students will act as ‘data points’ that must work together to find patterns without guidance. Students are given a mix of different shapes (for example, circles, triangles, squares) and must group them based on similarities without any guidance.
2. **Semi-supervised learning:** combining a small amount of labelled data with a large amount of unlabelled data. Some students will be ‘labelled data points’ and others will be ‘unlabelled data points’, collaborating to learn and predict. Provide a few labelled fruits and many unlabelled ones. Students will discuss how they can use the labelled examples to make educated guesses about the unlabelled ones.
3. **Reinforcement learning:** learning through trial and error, receiving feedback from actions taken. One student is the ‘agent’ (making decisions), while others act as the ‘environment’ (providing rewards or penalties based on the agent's actions). The agent tries to navigate a maze (drawn on the floor) to find a treasure while the environment gives feedback (claps for right turns, boos for wrong turns).

Each team presents their role-play scenario, explaining how their activity demonstrated their assigned learning type.

### Investigate common applications of key ML algorithms

**Including:** **data analysis and forecasting, virtual personal assistants,** **image recognition**

**Activity 18**: quiz to check for students understanding.

1. Define ‘variable’ in the context of machine learning.

|  |
| --- |
| In a machine learning context, a **variable** is a characteristic, property or feature that can change and is used to represent data.  Variables are fundamental to understanding how machine learning models work because they provide the information that the models use to learn and make predictions. |

1. What types of variables are used in machine learning? Provide examples.

|  |
| --- |
| Features are independent input variables used by the machine learning model to make predictions. For example, in a dataset about houses, features might include the size of the house (in square metres), the number of bedrooms, the location and the age of the house. Each of these characteristics helps the model understand what influences the price of a house.  Targets are dependentoutput variables that the model is trying to predict. The target variable is what you want to learn about based on the features. For example, with the house dataset, the target variable could be the price of the house. The model will use the features to predict this price.  Labels are previously known true values of a target variable. They are used during training and testing of supervised ML systems. |

1. Imagine you are building a machine learning model to predict whether a student will pass or fail a test based on their study habits. What would be the features and target variable?

|  |
| --- |
| Features could include study hours per week, attendance rate, previous test scores and participation in study groups. The target would bePass or Fail (this could be represented as 1 for ‘Pass’ and 0 for ‘Fail’)  In this example, the **features** provide the information the model uses to make predictions about the student's performance. The **target** variableis the outcome we want to predict based on those features. |

1. Choosing the right features and understanding their relationships with the target variable is crucial for building effective machine learning models. Identify the features required for your assessment task.

A student named Alex missed a maths exam.

Her teacher needs to predict Alex's score and has been told that machine learning can predict a student's mark on a missed assessment by analysing historical data, identifying patterns and using these patterns to make informed predictions.

Alex’s maths teacher has also been told that you are studying machine learning and may be able to help.

**Homework**: students watch [How to choose the Right Machine Learning Algorithm (23:04)](https://youtu.be/uh6iYQEHyyI?si=LxFMXBKQdXJxft6c).

**Activity 19**: complete the decision tree (21:49 of video) to choose the right machine learning algorithm.

|  |
| --- |
| **Sample student decision tree to help choose the right machine learning algorithm.** |

**Activity 20**: students research the common applications of key ML algorithms using the below resources.

* [10 Top Machine Learning Algorithms & Their Use-Cases](https://www.datacamp.com/blog/top-machine-learning-use-cases-and-algorithms)
* [Exploring 11 popular machine learning algorithms](https://www.elastic.co/blog/popular-ml-algorithms)

**Activity 21**: students form 3 teams to investigate common applications of key ML algorithms:

* Team 1 investigates data analysis and forecasting
* Team 2 investigates virtual personal assistants
* Team 3 investigates image recognition.

Each team explores a case study and identifies algorithms used.

Each team presents their findings to the class including a brief explanation of how their algorithms work. These presentations should form part of the class study notes.

**Teacher note**: the algorithms described in these case studies go beyond the scope of the Software Engineering Syllabus and require only the simplest description. Teachers ensure that a prerequisite of these activities includes watching and discussing [Understanding Neural Networks and AI (9:20)](https://youtu.be/NMZ0Tgc2jFQ?si=wNDPpenrEcnO0-8E) and [Neural Networks Explained in 5 minutes (4:31)](https://youtu.be/jmmW0F0biz0?si=kex5OwsVdWc1ksKQ) with reference to page 29 of the [Software Engineering course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF). These concepts are covered in greater detail when students research models used by software engineers to design and analyse ML.

The algorithms refenced in these case studies have been included to inspire students with the understanding that this is a rapidly changing field.

**Case study** **– data analysis and forecasting**

A medium-sized company in the retail industry have employed your team to investigate how key machine learning algorithms can be applied in data analysis and forecasting to improve inventory management, sales predictions and customer insights in a retail environment. ‘RetailCo’ specialises in clothing and accessories. They have multiple locations and an online store and face challenges in inventory management, understanding customer preferences and accurately forecasting sales. To address these challenges, RetailCo have employed your team to implement machine learning algorithms to analyse data and improve operational efficiency.

Your team have advised RetailCo to use a [Random Forest](https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/)algorithm to analyse historical sales data, seasonality and trends to optimise inventory levels across its stores. The model will take into account various factors, such as local events, holidays and promotions.You’ll usehistorical sales data from the past 3 years collected and pre-processed. Features are created, including day of the week, month, promotional events and weather conditions. The Random Forest model is trained to predict the optimal stock levels for each product category in each store.

You also adviseRetailCo to use [K-Means Clustering](https://www.geeksforgeeks.org/k-means-clustering-introduction/) to segment its customer base based on purchasing behaviour, preferences and demographics to enable targeted marketing campaigns. You’ll use customer purchase data, including purchase frequency, average transaction value and product categories. K-Means clustering is applied to identify distinct customer segments.

**Case study – virtual personal assistants**

‘SmartAssistant’ is a leading technology company developing a virtual personal assistant application. It aims to streamline daily tasks for users by using advanced machine learning techniques. With an increasing number of users relying on VPAs for managing schedules, answering questions, and controlling smart home devices, SmartAssistant have approached your team to investigate and explain how key machine learning algorithms are applied in virtual personal assistants (VPAs) to enhance user experience, improve task accuracy, and provide personalised recommendations.

Your team have advised SmartAssistant to use [Deep Neural Networks (DNNs) (5:31)](https://youtu.be/ZBnHoVMeeJE?si=7IGU4CHmzmmEZ99S) for [voice recognition](https://spotintelligence.com/2024/01/31/speech-recognition/) to accurately transcribe user speech into text, enabling effective interaction. The DNN model was trained on diverse audio samples covering various accents, dialects, and background noises. Feature extraction techniques were employed to identify phonemes and words within the audio signals. Continuous learning was applied to adapt to users' unique speech patterns.

Your team have advised SmartAssistant to use [Collaborative filtering](https://developers.google.com/machine-learning/recommendation/collaborative/basics) to provide personalised recommendations for music, news, reminders and tasks based on user preferences. Your team collect user interaction data, including songs played, articles read, and tasks completed, to create user profiles. A user-item interaction matrix is developed and collaborative filtering algorithms identified similarities among users to suggest content. The system continuously updates recommendations based on ongoing user interactions.

Your team also advise SmartAssistant to employ reinforcement learning to improve its contextual understanding, allowing it to manage multi-turn conversations. This system will learn from user feedback and adapt its responses based on the context of ongoing conversations. It used reward-based system where positive user feedback increases the likelihood of similar responses in future interactions. The model is tested in various scenarios to refine its ability to manage complex queries.

**Case study –** **image recognition**

‘HealthVision’ is a medical technology company, developing an advanced image recognition system aimed at assisting radiologists in diagnosing diseases from medical images such as X-rays, MRIs and CT scans. They have employed your team to investigate and explain how key machine learning algorithms are applied in image recognition to enhance diagnostic accuracy and efficiency in medical imaging, especially in detecting diseases from radiology images. They need to reduce diagnostic errors and improve the speed of analysis, which is crucial for timely patient care.

Your team have advised HealthVision to use [convolutional neural networks (CNNs)](https://www.ibm.com/think/topics/convolutional-neural-networks) to classify medical images into different categories, such as identifying healthy tissues versus diseased tissues. Your team access a large dataset of labelled medical images, including examples of healthy and diseased cases. The CNN architecture is designed with multiple convolutional layers to automatically extract relevant features from the images. The model is trained using a supervised learning approach, adjusting parameters based on the classification accuracy during training.

To detect specific anomalies within medical images, such as tumours or lesions your team suggests the system implement [Faster R-CNN](https://viso.ai/deep-learning/faster-r-cnn-2/) The training dataset is marked up to indicate the locations of key features within the images. The Faster R-CNN model is trained to generate bounding boxes around detected irregularities and classify them. Your team recommend HealthVision employ [U-Net](https://towardsdatascience.com/understanding-u-net-61276b10f360), a convolutional network architecture designed for biomedical image segmentation, that outlines structures within medical images. This model is trained on images with pixel-level annotations to identify different structures, such as organs or lesions. To speed up the training process and improve accuracy, your team have advised HealthVision to use transfer learning with pre-trained CNN Models (for example, [VGG16, ResNet](https://geekpython.in/practical-examination-of-4-deep-learning-models)) on [ImageNet.](https://viso.ai/deep-learning/imagenet/) The pre-trained models are fine-tuned on the medical imaging dataset, allowing the model to use knowledge gained from a large, diverse dataset. The top layers of the pre-trained networks were replaced with custom layers specific to the medical image classification tasks. The model was trained on a smaller dataset of medical images, significantly reducing the time required for training.

### Research models used by software engineers to design and analyse ML

**Including: decision trees, neural networks**

#### Decision trees

**Teacher note:** decision trees are a versatile and widely used model in both traditional computing and modern machine learning (ML). While decision trees in traditional computing and machine learning share a common tree structure and decision-making framework, their applications, implementations and capabilities differ. Traditional decision trees are rule-based and static, making them suitable for straightforward decision-making processes. Decision trees in machine learning are dynamic models that learn from data, enabling them to handle complex tasks and adapt over time. Understanding these differences is essential for software engineering students when choosing the appropriate approach for their specific problems. Teachers are advised to explicitly differentiate the application of decision trees via a comparison of their use in traditional computing versus their use in machine learning by software engineers.

“A decision tree is a diagram that represents all possible combinations of decisions and their resulting actions. Branches are shown to describe the eventual action depending on the condition at the time. Each decision path will lead to either another decision or a final action.” (NESA 2023)

**Activity 22**: students trace through the following decision trees to understand the logic that determines the smart temperature system and whether or not to purchase an SUV. Students discuss the types of variables used and their use in algorithm design. They should also consider whether the variables (features) are of equal importance in determining the target.

The following decision tree shows the rules in controlling the temperature system within a ‘smart’ house.

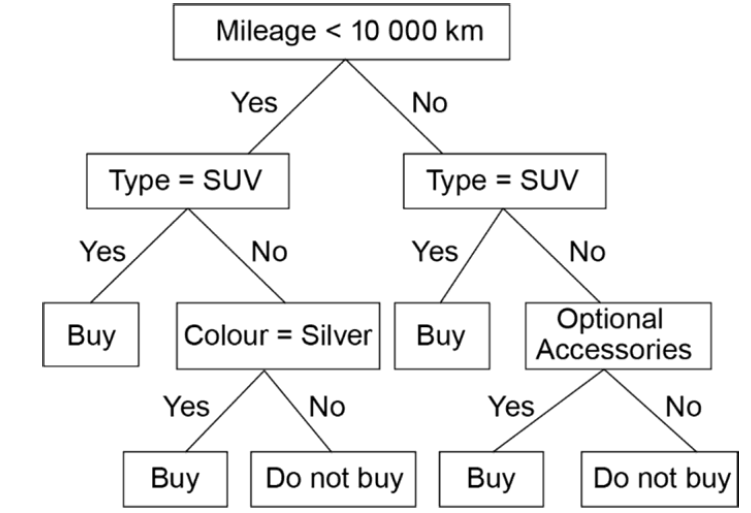
Figure 5 – decision trees



© NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2023.

The following diagram shows another way to represent a decision tree.

Figure 6 – decision tree example



© NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2023.

Students complete the [Familiarisation Questions (PDF 467 KB)](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/04e72951-9e6f-473a-a6ca-f857d22eaff6/software-engineering-hsc-marking-guidelines.PDF) on decision trees.

**Activity 23**: students partner up. One student reads the following comparison aloud and the other makes notes. Students summarise these comparisons into a table.

**Decision trees in traditional computing**

In traditional computing, decision trees are often used as a flowchart-like structure that helps in decision-making processes based on a series of questions or criteria. They serve as decision-making tools, guiding users through a series of options to arrive at a conclusion or action based on predefined rules. Traditional decision trees usually rely on explicit rules set by domain experts. Each branch of the tree represents a decision point based on specific conditions. Once built, these trees do not adapt or learn from new data. They are static and based on fixed logic, which can limit their effectiveness in dynamic environments. Traditional decision trees are often simple and easy to understand but may struggle with complex decision-making scenarios where multiple criteria interact in non-linear ways. They are often used in business scenarios to guide users through processes, such as troubleshooting or customer support queries, where a series of yes/no questions lead to a resolution.

**Decision trees in machine learning**

In machine learning, decision trees are a form of predictive modelling that uses a tree structure to represent decisions and their possible consequences (outcomes). They aim to model complex relationships in data, making predictions or classifications based on input features. Machine learning decision trees are trained on historical data, meaning they learn patterns and relationships from the data rather than relying on predefined rules. These trees can adapt to new data when retrained, allowing them to improve their predictions over time. This dynamic aspect is crucial for handling real-world data variability. Machine learning decision trees can handle complex interactions between features and can model non-linear relationships, making them more powerful for predictive tasks. Techniques such as pruning are applied to prevent overfitting, ensuring that the decision tree generalises well to unseen data. Used in various fields such as finance for credit scoring, healthcare for patient diagnosis, and marketing for customer segmentation. They help in making predictions based on patterns in historical data.

**Activity 24**: students summarise the comparison into the following table.

Table 15 – traditional computing versus machine learning

|  |  |  |
| --- | --- | --- |
| Aspect | Traditional computing | Machine learning |
| Purpose | decision support based on predefined rules | predictive modelling based on learned patterns |
| Implementation | static, rule-based systems | dynamic, data-driven learning |
| Complexity | limited handling of complex scenarios | capable of modelling complex relationships |
| Adaptability | does not adapt to new information | learns and improves with new data |
| Use cases | business processes, troubleshooting | predictive analytics, classification tasks |

**Activity 25**: students watch: [How Do Decision Trees Work (Simple Explanation) – Learning and Training Process (31:43)](https://youtu.be/xDWZzD4TPO0?si=BHWYan9KAbDw-TdC) and make notes.

**Teacher note:** the presenter in [How Do Decision Trees Work (Simple Explanation) – Learning and Training Process (31:43)](https://youtu.be/xDWZzD4TPO0?si=BHWYan9KAbDw-TdC) discusses decision trees as ‘greedy algorithms’. A greedy algorithm looks for the most immediate benefit without considering the bigger picture. At each step, the algorithm selects the option that seems the best at that moment making a ‘greedy choice’. Once a choice is made, it cannot be changed later. The algorithm does not backtrack. Greedy algorithms are often faster and simpler than other methods because they don't explore all possible options. Students can best understand this with a simple example:

**Problem:** you have coins of denominations 1, 5, and 10, and you want to make 28 cents using the fewest coins.

**Step-by-step process:** start with 28 cents. Use the largest coin (10 cents).

Take two 10 cent coins. Remaining amount = 28 − 20 = 8 cents. Use the largest coin (5 cents): Take one 5 cent coin. Remaining amount = 8 − 5 = 3 cents.

Use the largest coin (1 cent): Take three 1 cent coins. Remaining amount = 3 − 3 = 0 cents.

**Coins used:**

2 (10 cents) +1 (5 cents) + 3 (1 cent) = 6 coins total.

Students code the solution using a language of their choice:

function greedyCoinChange(amount, coins):

numCoins = 0

for each coin in coins (from largest to smallest):

while amount >= coin:

amount = amount - coin

numCoins = numCoins + 1

return numCoins

**Activity 26**: students watch: [Decision Tree: Important things to know (4:23)](https://youtu.be/JcI5E2Ng6r4?si=-zF45pVBeRZ3Va1m) and make notes.

**Teacher note:** in the presentation about decision trees, the term ‘non-parametric supervised learning’ is used. This means that the model learns to make predictions without needing to follow a specific shape or structure for the data. Instead of using fixed rules, the model changes based on the data it sees. Decision trees work by breaking the data into parts, or branches, based on different features to help make predictions. They don't assume how the features are related to the outcomes, which is why they are called non-parametric. Each point in the tree is a question about a feature and the paths lead to different results depending on the answers.

**Activity 27**: based upon their notes to the videos, students answer the following questions.

1. **What is a decision tree?**

|  |
| --- |
| A decision tree is a flowchart-like structure used for making decisions based on data.  It consists of nodes (representing features or attributes), branches (representing decision rules) and leaves (representing final outcomes or class labels). Decision trees help software engineers and data scientists make predictions or classifications by breaking down complex decisions into simpler, more manageable steps. |

1. **How do decision trees work? (include data collection, building the tree and making prediction)**

|  |
| --- |
| Data is gathered that contains various features (attributes) and their corresponding outcomes (labels). For example, if you're trying to predict whether someone will buy a product, the features could include age, income, and previous purchase history. To create the decision tree the algorithm evaluates and chooses the feature that provides the most information to split the data into different classes. Based on the chosen feature, the data is split into subsets. Each subset corresponds to different values of the feature (for example, age < 30, age ≥ 30). The algorithm repeats the process and continues to choose the best features and split the subsets until certain conditions are met, such as reaching a predefined depth or having a minimum number of samples in each leaf. Once the decision tree is built, it can be used to make predictions. When a new data point (for example, a new customer) is encountered, the decision tree is traversed from the root node, following the branches according to the values of the features until a leaf node is reached. The outcome associated with that leaf node is the prediction (for example, yes or no for buying the product). |

1. **Explain why software engineers use decision trees.**

|  |
| --- |
| Decision trees:   * are easy to understand and interpret. The visual structure allows anyone, even those without a technical background, to see how decisions are made. This makes them a popular choice for explaining models to stakeholders. * can work with both numerical and categorical data, making them versatile for various applications. They can capture complex, non-linear relationships between features, which simple linear models might miss. * can help identify which features are most important for making predictions. This insight can guide software engineers in feature selection and data preprocessing. |

1. **Extension**: explain the limitations of decision trees (including overfitting, instability and bias).

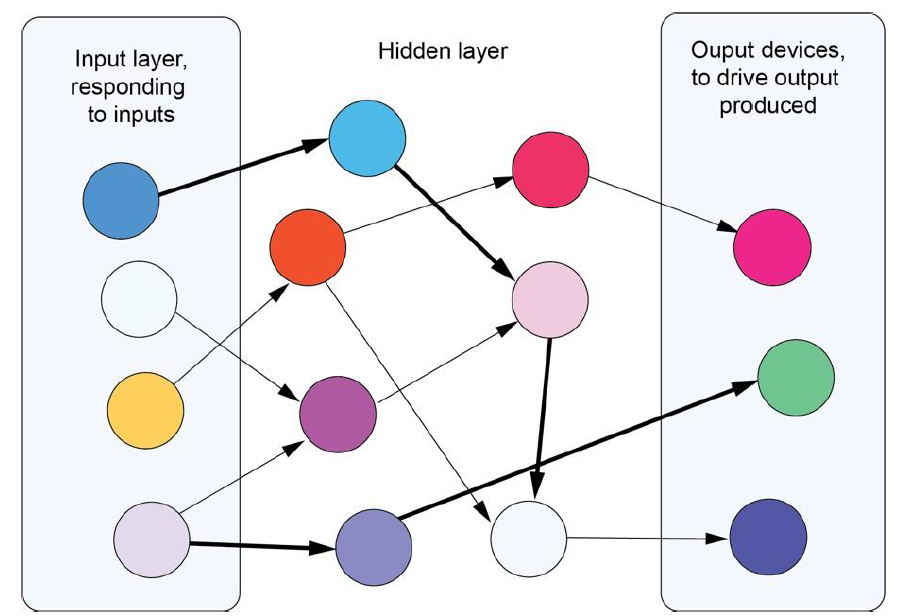
|  |
| --- |
| Decision trees can easily become too complex, capturing noise in the data instead of the underlying pattern. This can lead to poor performance on new, unseen data. To combat **overfitting**, techniques like pruning (removing branches that have little importance) can be applied. Small changes in the data can lead to different trees being generated, making them **less stable** compared to other algorithms. Decision trees can be **biased** towards features with more levels or categories. For example, if one feature has many unique values, it may dominate the decision-making process. |

#### Neural networks

Students should attempt to summarise their understanding of neural networks.

“Neural networks were designed to mimic the processing inside the human brain. They consist of a series of interconnected nodes (artificial neurones). Each neurone can accept a binary input signal and potentially output another signal to connected nodes”. (NESA 2023)

Figure 7 – neural networks



© NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2023.

**Activity 28**: students annotate the neural network diagram from the [Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) with the following:

* **Input layer:** this is the first layer, where data is fed into the network. Each neuron in this layer represents a feature of the input data (for example, pixel values for images).
* **Hidden layers:** these layers are situated between the input and output layers. They perform computations and transformations on the input data. A neural network can have one or multiple hidden layers and the more layers it has, the deeper the network (hence the term ‘deep learning’).
* **Output layer:** this layer produces the final result or prediction. Each neuron in this layer corresponds to a possible output class (for example, identifying whether an image is of a cat or a dog).

**Activity 29**: students watch: [Neural Networks Explained in 5 minutes (4:31)](https://youtu.be/jmmW0F0biz0?si=X7pD0sFIpAmFPhQf) and answer the following question.

What are the **5 key concepts** explained in 5 minutes?

|  |
| --- |
| 1. Neural networks are composed of node layers. There is an input node layer, a hidden layer, and an output layer. These neural networks reflect the behaviour of the human brain, allowing computer programs to recognise patterns and solve common problems in the fields of AI and deep learning. They should be described as an artificial neural network, or an ANN, to distinguish it from the very un-artificial neural network that is operating in our heads. |
| 1. Each node, or artificial neuron, is its own linear regression model. Linear regression is a mathematical model that's used to predict future events.   The weights of the connections between the nodes determine how much influence each input has on the output.  Each node is composed of input data, weights, a bias, or a threshold, and then an output.  Data is passed from one layer in the neural network to the next in what is known as a feed forward network. |
| 1. Considers what a single node in our neural network might look like to decide if we should go surfing. The decision to go or not is our predicted outcome or known as our ‘yhat’. There are 3 factors influencing our decision. Each decision gets assigned a weight based on its importance on a scale of 0 to 5. For example, waves are important, so this gets a score of 5. Crowds are not so important, so would get a score of 2. Sharks, get a score of 4. These values are plugged into the formula to get the desired output. yhat = of 1 – ‘we're going surfing’. If the weights or the threshold are adjusted, a different outcome is achieved. |
| 1. Neural networks rely on training data to learn and improve their accuracy over time.   Labelled datasets leverage supervised learning to train the algorithm. As models are trained, accuracy is evaluated using something called a cost function. The goal is to minimise the cost function to ensure the correctness of fit for any given observation, and that happens as the model adjusts its weights and biases to fit the training data set, through what's known as gradient descent, allowing the model to determine the direction to take to reduce errors, that is, to minimise the cost function. |
| 1. There are multiple types of neural networks beyond the feed forward neural network than in this video. For example, there are convolutional neural networks, known as CNNs, which have a unique architecture that is well suited for identifying patterns like image recognition and there are recurrent neural networks, or RNNs, which are identified by their feedback loops. RNNs are primarily leveraged using time series data to make predictions about future events like sales forecasting. |

Students should understand neural networks well enough to explain to a younger sibling. For example, [What Is An Artificial Neural Network And Why Do We Need It?](https://kids.frontiersin.org/articles/10.3389/frym.2021.560631)

**Activity 30**: students watch the following videos and make notes.

* [Neural Networks and Deep Learning: Crash Course AI #3 (12:22)](https://www.youtube.com/watch?v=oV3ZY6tJiA0)
* [But what is a neural network? | Deep learning chapter 1 (18:39)](https://youtu.be/aircAruvnKk?si=CpWDf7YTPdA0qYBF).

**Homework extension**: students watch [A Neural Network Primer (19:13)](https://youtu.be/_56bfCu02ZE?si=-okFKtww3E5PHRh7).

**Activity 31**: students use their notes from the videos and the Course specifications to answer the following questions.

**How do neural networks work?**

**Training cycle**

Internal weightings and threshold values for each node are determined in the initial training cycle for each neural network. The system is exposed to a series of inputs with known responses. Linear regression with backward chaining is used to iteratively determine the set of unique values required for output. Regular exposure to the training cycle results in improved accuracy and pattern matching.

**Execution cycle**

In the diagram, signal strength between nodes with the strongest weightings are thicker representing a higher priority in determining the final output. The execution cycle follows the training cycle and utilises the internal values developed during the training cycle to determine the output. (NESA 2023)

1. What is a neural network?

|  |
| --- |
| A neural network is a computational model inspired by the way biological brains work. It consists of layers of interconnected nodes (or neurons) that process information. Neural networks are particularly useful in machine learning (ML) for tasks such as image recognition, natural language processing and more. They can learn complex patterns from data and make predictions or classifications. |

1. How do neural networks work?

|  |
| --- |
| **Training cycle**  The training cycle involves processes that help the neural network learn from data.  The steps adjust the network's weights to minimise the error in its predictions.  After producing an output, the neural network compares it to the actual target value using a loss function. The loss function quantifies how far off the prediction is from the actual result. To improve accuracy, the network uses a process called backpropagation.  It calculates the gradient of the loss function with respect to each weight in the network.  The weights are then adjusted in the opposite direction of the gradient to minimise the loss. This process is repeated through multiple iterations (epochs) until the network learns to make accurate predictions. |
| **Execution cycle**  The execution cycle involves processes that occur when the neural network is used to make predictions on new data. These steps do not adjust the network's weights but rather use the trained weights to produce an output. When an input is fed into the network, it undergoes forward propagation. Each neuron computes a weighted sum of its inputs. A weight is a numerical value that represents the importance of a particular input. The weighted sum is passed through an activation function (like ReLU or Sigmoid) that introduces non-linearity, allowing the network to learn complex patterns. The output from one layer becomes the input for the next layer, continuing until the final output layer is reached. |

1. What are some of the limitations of neural networks?

|  |
| --- |
| Neural networks typically require a large amount of data to train effectively. Insufficient data can lead to poor performance and overfitting. Training deep neural networks can be resource-intensive, requiring significant computational power (often needing GPUs) and time. Neural networks are often considered ‘black boxes’ because their internal workings can be difficult to interpret. This lack of transparency can be a drawback in applications where understanding the decision-making process is essential. There are many hyperparameters (like learning rate, number of layers and number of neurons per layer) that need to be carefully tuned to achieve optimal performance. Finding the right configuration is challenging. |

**Homework**: students watch [Machine Learning Essentials: From Linear Regression to Neural Networks | Companion Podcast (24:36)](https://youtu.be/UIJ7r74v4AM?si=MDYanCXoe16v_6K6).

**Extension**: students work through [Neural networks | Machine Learning | Google for Developers](https://developers.google.com/machine-learning/crash-course/neural-networks).

### Describe types of algorithms associated with ML

**Including: linear regression, logistic regression, K-nearest neighbour**

#### Linear regression

Linear regression algorithms are used to predict values in a continuous range, such as integers. These regression algorithms are used for machine learning.

Linear regression is a foundational algorithm in machine learning that models the relationship between independent and dependent variables. Its simplicity, interpretability and efficiency make it a popular choice for a wide range of applications.

Understanding linear regression is crucial for anyone starting out in machine learning, as it lays the groundwork for more advanced techniques and models.

Students may recall from Year 9 and 10 Mathematics exploration of linear relationships including linear regression equations: [Linear regression | Machine Learning | Google for Developers](https://developers.google.com/machine-learning/crash-course/linear-regression).

As an engaging refresher of these concepts and how they apply to regression algorithms students can apply datasets (including those for their assessment task) to a [Desmos calculator.](https://www.desmos.com/calculator)

**Activity 32**: students watch [Linear Regression (with Desmos) (9:25)](https://youtu.be/_ldsNWOHHBk?si=_2LRIjC2VZJkd9LA) and apply the data from the 2 features of the [Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) (p 28) to Desmos to better understand the linear regression equation.

Figure 8 – linear regression using Desmos

A screen shot of an online Desmos graph calculator.



Students As Extension Homework students complete the [Google crash course module that introduces linear regression.](https://developers.google.com/machine-learning/crash-course/linear-regression)

**Linear Regression in Machine Learning**

**Linear Regression** is one of the simplest and most widely used algorithms in machine learning for predictive modelling. It is a statistical method that models the relationship between a dependent variable (also known as the target variable) and one or more independent variables (also known as features or predictors). The goal of linear regression is to find the best-fitting linear relationship that can be used to make predictions.

Image created using [Desmos](https://www.desmos.com/?lang=en) and is licensed under the [Desmos Terms of Service](https://www.desmos.com/terms?lang=en).

**Extension**: students watch [How to Calculate a Simple Linear Regression by Hand (7:02)](https://youtu.be/YC0bvIxR6t4?si=pkSIEVqt2stPexsf) and use the x, y values provided by the Course specification example.

**Activity 33**: students apply linear regression using Python code by importing [NumPy](https://numpy.org/doc/stable/) and [scikit-learn](https://scikit-learn.org/stable/getting_started.html) machine learning frameworks. They install these modules using:

* pip3 install NumPy
* pip3 install scikit-learn

Students [PRIMM](https://teachcomputing.org/blog/using-primm-to-structure-programming-lessons/) the following code and according to the [Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) may expect an answer of >>3. They modify line 7 to: y\_prediction = model.predict([[4.5]]) and may expect an answer of >>3.5.

1. import numpy as np
2. from sklearn.linear\_model import linear\_regression
3. x = np.array([[2], [4], [6], [8], [10], [12], [14], [16]])
4. y = np.array([1, 3, 5, 7, 9, 11, 13, 15])
5. model = linear\_regression()
6. model.fit(x, y)
7. y\_prediction = model.predict([[4]])
8. print(f"predicted value is: {y\_prediction}")

Students should see that y\_prediction = model.predict([[4]]) doesn't produce exactly 3, even though 3 is the value in the target data y mapped to the input 4 in the training data. Students can confirm results by calculating the slope and intercept of the line: y=mx+b.

This problem illustrates the need for data splitting to separate training data from test data.

**Activity 34**: students propose reasons why results differ in the space provided below.

|  |
| --- |
| The model.predict([[4]]) gives 1.5 instead of 3 because the model finds the best-fitting line that minimises the overall error across all data points. The line it fits is a compromise to best represent all the data points, not just the exact pairs in the training data. The prediction for x=4 is based on this best-fit line, not a direct lookup of the training data.  y\_prediction = model.predict([[4.5]]) produces 1.75 because the linear regression model fits a line that best represents the overall trend in the data, rather than memorising the exact values. This is why the prediction is based on the learned relationship, which aims to minimise the error across all training data points. |

**Extension**: students consider how to modify the code so that it maps directly to existing training data or predicts the value where no exact values are known.

This solution could include the use of dictionaries in Python.

##### Practical task

**Teacher note:**

Teachers may choose options to implement this task.

Summative assessment of the following task if it fits within their assessment schedule (**Note**: the [assessment schedule](https://education.nsw.gov.au/teaching-and-learning/curriculum/tas/tas-curriculum-resources-7-12/tas-11-12-curriculum-resources/assessment-schedule-software-engineering-year-12) published by the department did not include this as one of the assessment tasks and suggested content be formally assessed in the trial examination).

**OR**

Summative assessment of the task that integrates Software automation, Programming for the web and Secure software architecture into a year-long progressive web app (PWA) project. Students could create a web-based markbook with SQLite database and predictive capabilities.

**OR**

Formative assessment of a solution to the problem as a practical group-based project to deliver syllabus content within the Software automation focus area and Programming for the web topic.

A comprehensive support guide for these options can be found in the Appendix section of this document. All files, CSV, markbooks and Python code is available to teachers in the **Software Automation** folder under the **Files** tab of the [Software Engineering channel](https://schoolsnsw.sharepoint.com/:f:/r/sites/TASNSWStatewideStaffroom/Shared%20Documents/13.%20%F0%9F%A7%91%E2%80%8D%F0%9F%92%BB%20Software%20Engineering%2011-12/Secure%20Software%20Architecture?csf=1&web=1&e=6nkPE3) via the TAS Statewide Staffroom.

##### Scenario problem statement

A student named Alex missed a maths exam.

Her teacher needs to predict Alex's score and has been told that machine learning can predict a student's mark on a missed assessment by analysing historical data, identifying patterns and using these patterns to make informed predictions.

Alex’s maths teacher has also been told that you are studying machine learning and may be able to help.

**Activity 35**: teacher-led discussion and questions.

What approach should students take?

Estimating the mark can be done through several levels of complexity.

A simple implementation would consider a single dimension. For example, a mark estimated through looking at the students marks in other assessments only.

A better solution would consider the estimate by looking at two dimensions and looking at what other students got for that assessment as well.

The best solution may be able to analyse the data and see if a linear or polynomial regression model is best for the specific data. For example, some classes may see marks that progress in a linear fashion however other classes may improve over time with marks instead fitting on a curve.

What about using standard deviations and z scores? For example, using averages of other tasks.

What concerns do students have about fairness and accuracy?

**Activity 36**: students answer the following questions.

1. What are some of the social and ethical issues in predicting a student's mark on a missed assessment?

|  |
| --- |
| * Students are impacted by the system so its output must be reliable or enough information given to allow the teacher to reliably assess the validity of the estimated mark. * Will the system lead to teachers just putting in estimated marks for missed assessments rather than encouraging students to catch up on missed assessments? * Is there the potential for bias in the system considering the types of students that may frequently miss assessment tasks (for example, lower performing students)? * Do estimates tend higher or lower for certain ability levels? * Should estimates only be completed at the end when all assessment tasks are in? * The system could be used to detect unusual marks (abnormally high or low) which could indicate, for instance, possible cheating. For example, the system may have the ability to create an estimate for each assessment for each student and compare to their actual mark highlighting any that deviate too far. * Other concerns? |

1. What are some of the datasets the model might analyse?

|  |
| --- |
| * Alex's previous maths scores (for example, 80%, 85%, 90%) * Her attendance record (for example, was she present for most classes?) * Her participation in homework or projects (for example, did he consistently complete assignments?) * The average score of classmates who took the exam (to see if the exam was particularly hard or easy).   What other datasets could be used? |

1. What are the conditions for being able to estimate a mark (decision tree)?

|  |
| --- |
| * Are there enough students present to estimate the mark effectively? * Are there not too many gaps in the data? * Are the marks consistent enough to estimate the mark effectively? * Have enough assessment tasks been completed? |

1. How could bias impact Alex's predicted grade?

|  |
| --- |
| Students may consider factors including whether Alex may be stronger in this content area. |

1. How might students’ knowledge that this is the model used effect their motivation?

|  |
| --- |
| Students may consider scenarios where they might skip a difficult task and get an equal result. |

1. How could the teacher reduce human and dataset source bias? (This could be answered in conjunction with Activity 55 of the TSR).

|  |
| --- |
| A range of student responses are applicable here. Students should consider how they would mitigate against bias in the design and development process. |

##### How to start?

**Frame the problem**

**Activity 37**: students revise [How to choose the Right Machine Learning Algorithm (23:04)](https://youtu.be/uh6iYQEHyyI?si=LxFMXBKQdXJxft6c).

[Introduction to Machine Learning Problem Framing](https://developers.google.com/machine-learning/problem-framing) teaches you how to determine if machine learning (ML) is a good approach for a problem and explains how to outline an ML solution.

**Steps**

1. Collect data

Gather historical data on students' assessment scores, including those who completed the task and those who missed it. This data might include: previous assessment scores (for similar tasks), attendance records, participation in class, study habits or hours spent studying, other relevant factors like demographics or learning styles.

1. Wrangling data

Clean the data by handling missing values, removing duplicates and normalising the data if needed. This ensures that the model can learn effectively from the data. Create a dataset that includes features (input variables) that might help in predicting the missed task's mark. For example, if a student usually scores 75% on similar assessments, this could be a feature used in the model. [Code samples of data wrangling](https://github.com/TempeHS/Practical-Application-of-NESA-Software-Engineering-MLOps/blob/main/2.Model_Development/2.1.Data_Wrangling/2.1.1.data_preview.ipynb).

1. Choose a machine learning model

Select a suitable machine learning algorithm for the task:

* Linear regression: useful for predicting continuous values based on relationships between variables.
* Decision trees: these can be used to make predictions based on decision rules inferred from the data.
* Random Forest: an ensemble method that combines multiple decision trees to improve prediction accuracy.

1. Train the model

Split the data into a training set and a test set. The training set is used to teach the model by showing it examples of past students' scores and the factors that influenced those scores. The model learns patterns in the data, such as how previous scores correlate with various factors (like attendance or study habits).

1. Test the model

After training, evaluate the model’s performance using the test set, which contains data the model has not seen before. This helps determine how accurately the model can predict scores. Metrics like mean absolute error (MAE) or mean square error (MSE) can be used to measure the model's accuracy.

1. Make predictions

Once the model is trained and validated, it can be used to predict the mark for a student who missed the assessment. You would input the relevant features for that student (for example, their past scores, attendance and so on.) into the model, and it would output a predicted score.

1. Continuously improve

As more data becomes available (for example, scores from future assessments) the model can be updated and retrained to improve its predictions.

**Activity 38**: students annotate the code with the following descriptions. These may be read out by the teacher or provided as a class discussion and quiz.

1. **import numpy as np**

Line 1 imports the numpy library for numerical computations in Python. It is commonly used for working with arrays and matrices of numbers. The name np is used to make it easier to reference the library's functions.

1. **from sklearn.linear\_model import linear\_regression**

Line 2 imports the linear\_regression class from the sklearn.linear\_model module. Scikit-learn (or sklearn) is a popular machine learning library in Python that provides simple and efficient tools for data analysis and modelling. The linear\_regression class is used to create and train linear regression models.

1. **x = np.array([[2], [4], [6], [8], [10], [12], [14], [16]])**

Line 3 creates an array x using the numpy library. The array x contains the input data (also called features) for the linear regression model. Each value is enclosed in double square brackets to make it a 2D array with one column. This format is required by the linear\_regression model.

1. **y = np.array([1, 3, 5, 7, 9, 11, 13, 15])**

Line 4 creates an array y using the numpy library. The array y contains the target data (also called labels) corresponding to the input data in x.

This is a 1D array where each value is the output for the corresponding input value in x.

1. **model = linear\_regression()**

Line 5 creates an instance of the linear\_regression class and assigns it to the variable model. This object will be used to fit the linear regression model to the input data and make predictions.

1. **model.fit(x, y)**

Line 6 trains (or fits) the linear regression model using the input data x and the target data y. The fit method finds the best-fitting line that minimises the difference between the predicted and actual target values.

1. **y\_prediction = model.predict([[4]])**

Line 7 uses the trained model to make a prediction for input value 4. The predict method takes a 2D array as input (hence the double square brackets) and returns the predicted target value. The result is stored in the variable y\_prediction.

1. **print(f"predicted value is: {y\_prediction}")**

Line 8 prints the predicted target value for the input 4. The f before the string indicates that this is an f-string, which allows embedding expressions inside curly braces {}. The value of y\_prediction is inserted into the string and displayed.

**Extension**: students can improve their understanding by applying the Course specifications x, y data from the linear regression to this guided sample of [How to Calculate a Simple Linear Regression by Hand (7:02)](https://youtu.be/YC0bvIxR6t4?si=IVxWdXrJz_7HOtSm).

****Students experiment with code samples of different expansions of the Course specifications They visit [Tempe Github](https://github.com/TempeHS/NESA_Course_Specifications_Linear_Regression) to import CSV, SQLite DB as well as model export/import.

**Activity 39**: students summarise the features of the linear regression algorithm by completing the description in the table below.

Table 16 – features of the linear regression algorithm

|  |  |
| --- | --- |
| Code | Description |
| import libraries | The code imports necessary libraries and classes (numpy and linear\_regression). |
| create data arrays | It creates arrays for input data (x) and target data (y). |
| initialise model | It initialises a linear regression model. |
| train model | It trains the model using the input and target data |
| make prediction | It makes a prediction for a new input value |
| print result | It prints the predicted value |

**Activity 40**: students refer Table 16 to provide a written summary of what the code does in the space provided below.

|  |
| --- |
| The code imports necessary libraries and classes. It creates input and output data arrays using numpy. It initialises and trains a linear regression model using the input and output data. It creates a test input value and makes a prediction using the trained model. It prints the predicted output value. |

**Activity 41**: students complete the following table on the uses of linear regression in ML.

Table 17 – uses of linear regression in machine learning

|  |  |  |  |
| --- | --- | --- | --- |
| Context | Example | Description | How |
| Real estate | **Predicting house prices** | Predicting the price of a house based on various features, such as its size (square metres), number of bedrooms, number of bathrooms, location and age of the property. | Analyse past sales data and the corresponding features of houses to train a linear regression model in estimating the selling price of a new property based on its characteristics. |
| Retail | **Sales forecasting** | Forecasting future sales based on historical sales data, marketing expenditures, seasonality and economic indicators. | Train a model on past sales data and related features (like advertising spend or holiday seasons), businesses to predict future sales trends. |
| Healthcare | **Predicting patient outcomes** | Predicting patient outcomes based on various health metrics, such as age, weight, blood pressure and cholesterol levels. | Predict a patient's recovery time after surgery based on preoperative health indicators. This information can assist planning post-operative care. |
| Education | **Predicting student performance** | Predicting student performance based on factors such as study hours, attendance rates and previous test scores. | Analyse historical data to identify which factors most significantly influence student grades. This allows educators to intervene early with students who may need additional support. |
| Environmental Science | **Predicting air quality index (AQI)** | Predicting air quality levels based on environmental factors such as humidity, temperature, wind speed and pollution sources. | Model historical air quality data against features, to forecast pollution levels, which helps in public health alerts and policymaking. |
| Marketing | **Customer lifetime value (CLV) prediction** | Estimating the lifetime value of a customer based on factors such as purchase frequency, average transaction value and customer demographics. | Understand expected CLV, to make informed decisions about marketing budgets, customer acquisition strategies and retention efforts. |
| Finance | **Stock price prediction** | Predicting stock prices based on historical stock prices, trading volume and economic indicators. | Analyse historical stock data along with relevant economic factors, to make more informed decisions about buying or selling stocks. |

#### Logistic regression

**Teacher note**: logistic regression is a foundational supervised machine learning technique for binary classification tasks, making it a suitable starting point for handling categorical data.

**Activity 42**: students watch [Machine Learning and Logistic Regression (5:12)](https://youtu.be/AX-ZEC-71DI?si=EbNnhhtctyPN0Piw).

Students summarise the video into key concepts of logistic regression for categorical data.

1. What is the purpose of logistic regression?

|  |
| --- |
| Used to generate predictions for categorical data (for example, colour, gender, job role) rather than numerical continuous data (for example, height, weight).  Ideal for binary classification problems, determining the probability of data entries belonging to one of two categories. For example: classifying animals as ‘cat’ or ‘not a cat’ based on distinguishing features (for example, 4 legs, whiskers, claws). |

1. How is data represented?

|  |
| --- |
| Categorical features are represented as binary values (0 for false, 1 for true).  Example data entries:  Animal 1: 0 (not 4 legs), 0 (no whiskers), 0 (no claws), 0 (not a cat)  Animal 2: 1 (4 legs), 0 (no whiskers), 0 (no claws), 0 (not a cat)  Animal 3: 1 (4 legs), 1 (whiskers), 1 (claws), 1 (is a cat) |

1. How does logistic regression work?

|  |
| --- |
| It uses the **sigmoid function** (logistic function) to map input features to a probability between 0 and 1. The sigmoid function is defined as:[f(x) = \frac{1}{1 + e^{-x}}]  The graph of the sigmoid function is an S-shaped curve, indicating that outputs are constrained between 0 and 1. A threshold (for example, 0.5) is set to classify probabilities:  Probabilities ≥ 0.5 are classified as ‘cat’ (1). Probabilities < 0.5 are classified as ‘not a cat’ (0).  The closer the predicted value is to 1, the higher the likelihood of being classified as a cat.  Each data entry receives a prediction of either 0 (not a cat) or 1 (cat).  The goal is to accurately predict the categories based on the input features, ideally matching the true labels (for example, predicting 0, 0, 1 for the example dataset). |

**Homework** students watch [Machine Learning Crash Course: Logistic Regression (3:24)](https://youtu.be/72AHKztZN44?si=aAca23dTQw-fNUT7).

**Extension**: students complete the Google crash course module that introduces [Logistic Regression](https://developers.google.com/machine-learning/crash-course/logistic-regression).

#### K-nearest neighbour

**Activity 43**: students research K-nearest neighbour.

**Teacher note**: the K-nearest neighbours (KNN) algorithm is a simple and popular method used in machine learning to classify data. It works by looking at the data points that are closest to a new point to decide which group it belongs to.

Students research [What is the K-Nearest Neighbor (KNN) Algorithm? (8:00)](https://youtu.be/b6uHw7QW_n4?si=iOuVtFRyJziZ_p6v) and [What is the k-nearest neighbors (KNN) algorithm?](https://www.ibm.com/topics/knn) Students summarise their findings from the research in the space provide below.

|  |
| --- |
| KNN is a supervised learning algorithm that uses labelled data (data with known answers) to make predictions. It is non-parametric: which means it doesn't make any assumptions about the underlying data distribution. When you need to classify a new data point, KNN finds the 'k' closest points (neighbours) around it. Then, it looks at the groups of these neighbours and assigns the new point to the group that most of its neighbours belong to. Although KNN can be used to predict continuous values (regression), it is most commonly used for classification tasks, where the goal is to assign data points to a specific category based on their features. KNN assumes that similar data points are near each other. It’s a bit like finding out which group a new student belongs to by looking at the groups of the students sitting closest to them. |

## Programming for automation

### Design, develop and apply ML regression models using an OOP to predict numeric values

**Including:** **linear regression, polynomial regression, logistic regression**

In this section students work on the assessment task for this focus area. They then modify the linear regression sample code provided in the Course specifications by completing the code activities in the Appendix section of this document. Students can familiarise themselves with the [models here](https://github.com/TempeHS/Machine_Learning_OOP_Implementation_Examples).

[Linear regression](#_Linear_regression) has been described in previous activities.

**Activity 44**: students watch [Linear Regression in 2 minutes (2:33)](https://youtu.be/CtsRRUddV2s?si=CaVpplsHSTLK1xOP) and note (from 1:28) on using Python.

#### Polynomial regression

Polynomial regression is an extension of simple linear regression that allows modelling of relationships between variables that are not just a straight line. Instead of fitting a straight line to the data, polynomial regression fits a curve, which can capture more complex patterns. In summary, polynomial regression is a powerful tool for modelling relationships that are not linear. It's important to choose the degree of the polynomial carefully to avoid overfitting. Understanding polynomial regression helps apply more flexible models to various data scenarios to improve predictions.

**Activity 45**: teacher-led discussion on the limitations of linear regression algorithms and scenarios where polynomial regression can provide more accuracy than linear regression. Students complete the table on the following page.

Table 18 – examples for use of polynomial regression

|  |  |
| --- | --- |
| Example | Description |
| 1. Plant growth | The relationship between the age of a plant and its height is non-linear. Rapid initial growth that slows down as the plant matures. |
| 1. Car performance | The relationship between car speed and fuel consumption is often non-linear, with fuel efficiency decreasing at higher speeds after initially improving. |
| 1. House prices | The price of a house relative to its size can be non-linear, where larger homes may have disproportionately higher prices due to various factors. |
| 1. Temperature changes | The daily temperature pattern, rising in the morning and peaking in the afternoon before dropping, can be better modelled with a polynomial regression. |
| 1. Sales over time | The sales of a new product often follow a non-linear trend, starting slow, then growing rapidly and finally plateauing as the market saturates. |
| 1. Marketing campaign effects | The impact of advertising spend on sales typically shows diminishing returns, which polynomial regression can model effectively. |
| 1. Human growth patterns | The age versus height relationship in children is non-linear, showing rapid growth in infancy that slows down later in childhood. |
| 1. Physics experiments | The relationship between the angle of a projectile and its range follows a parabolic path, requiring polynomial regression for accurate modelling. |
| 1. Economic indicators | The relationship between interest rates and investment levels can show diminishing returns, which can be captured more accurately with polynomial regression. |
| 1. Health data | The dosage of medication versus its effectiveness can be non-linear, where too little or too much leads to lower effectiveness, best modelled with polynomial regression. |
| 1. Other |  |

**Activity 46**: students refer to the table above to answer the following questions.

1. Why use polynomial regression for plant growth?

|  |
| --- |
| Imagine you have data about the height of plants over time. If you plotted the height of the plants against time, you might see that they grow quickly at first and then slow down as they mature. A linear regression model would fit a straight line to this data, which might not accurately represent the growth pattern. By using polynomial regression, you could fit a curve that starts steep and gradually flattens out, capturing the true growth pattern of the plants. |

1. Why use polynomial regression more generally?

|  |
| --- |
| Many real-world phenomena do not follow a linear trend. For example, the relationship between the speed of a car and the distance it travels might curve rather than remain straight. By using polynomial terms, we can create a model that better fits the data, capturing the nuances and complexities that a straight line might miss. |

1. What is different between linear relationships and polynomial relationships?

|  |
| --- |
| In linear regression, we assume that the relationship between the independent variable (input) and the dependent variable (output) can be described by a straight line  In polynomial regression curves are allowed for by adding powers of the independent variable. |

1. What does ‘the degree of the polynomial’ mean?

|  |
| --- |
| The degree of the polynomial (the highest power of x*x* in the equation) determines how flexible the curve is. A higher degree can fit the data more closely but may also lead to overfitting, where the model captures noise in the data rather than the underlying trend. For example: a degree of 1 is a straight line (linear). A degree of 2 is a parabola (quadratic). A degree of 3 is an S-shaped curve (cubic). A balance must be found between fitting the data well and maintaining a model that generalises to new, unseen data. |

1. How does polynomial regression work?

|  |
| --- |
| Similar to linear regression, you start with your data, which consists of input features (independent variables) and output values (dependent variable). The input features are transformed into polynomial features. For example, if you have a single input x*x*, you would create new features like x2*x*2,x3*x*3, and so on. Then use these polynomial features to fit a regression model, just like you would with linear regression. Once the model is trained, you can use it to predict output values for new input data. |

**Extension**: students watch [Polynomial Regression for Machine Learning (15:53)](https://youtu.be/Q8UfQ_aoCeQ?si=75dRe0GrI8aLQeUR).

**Activity 47**: students modify their low.py solution in the assessment task from using linear regression into one that uses polynomial regression by completing the code in the Appendix section of this document.

**Logistic regression**

[Logistic regression](#_Logistic_regression) has been described in previous activities. Students may attempt to modify their assessment task using linear regression into one that uses logistic regression. Sample code is available in the TAS Statewide Staffroom in the Software Engineering channel. Students attempt to transform the low.py solution from using linear regression to using logistical regression. Students will need to make several modifications such as:

* + - * changing the linear regression model to logistic regression
      * adjusting the target variable (marks) to be binary
      * assuming a binary classification scenario (for example, ‘pass’ or ‘fail’ based on a passing mark threshold)
      * modifying the prediction logic accordingly including the assumption that a mark of 50 or above indicates a ‘pass’ (1) and below 50 indicates a ‘fail’ (0).

### Apply neural network models using an OOP to make predictions

Students should have attained a basic understanding of neural networks from previous activities. They consider and discuss [How can you apply neural networks to education?](https://www.linkedin.com/advice/1/how-can-you-apply-neural-networks-education)

Students consider how they would scale up their mark estimation program into an agent or larger system that uses a neural network. They consider the steps involved including:

* + - 1. Importing a neural network library such as Keras or TensorFlow to create and train the model (instead of using sklearn for linear regression).
      2. Defining the neural network architecture by creating appropriate layers (input, hidden and output layers).
      3. Putting together the model by specifying the optimiser, loss function and metrics.
      4. Training the model by fitting it using the training data (input features and target values).
      5. Making predictions using the trained model to predict marks based on the input task.

**Extension and/or homework**: students may wish to engage with the following online courses and activities.

* [How to build a Neural Network from scratch](https://www.freecodecamp.org/news/building-a-neural-network-from-scratch/)
* [Create a Simple Neural Network in Python from Scratch (14:14)](https://youtu.be/kft1AJ9WVDk?si=YK-e_el7vgFJ6EHF)
* [Python AI: How to Build a Neural Network & Make Predictions (4:09)](https://realpython.com/videos/build-neural-network-ai-overview/)
* [Neural network playground](https://nnplayground.com/)

## Significance and impact of ML and AI

### Assess the impact of automation on the individual, society and the environment

**Including: safety of workers, people with disability, the nature and skills required for employment, production efficiency, waste and the environment, the economy and distribution of wealth**

**Activity 48**: teacher-led discussion on the widespread impact of automation and how it affects individuals, society and the environment.

**Activity 49:** students respond to questions raised in the discussion to decide whether effects are positive or negative and complete Table 19 below.

**Extension**: this content could be covered as a class debate with the topic reframed as Automation has significant negative impact upon individuals, society and the environment and should be stopped.

Table 19 – impact of automation on individuals, society and the environment

|  |  |  |
| --- | --- | --- |
| Impact | Positive impact | Negative impact |
| Safety of workers | Automation can enhance workplace safety by doing dangerous tasks such as heavy lifting, operating machinery or performing hazardous material handling. Robots and automated systems can operate in environments that are unsafe for human workers, reducing the risk of accidents and injuries. | Increased automation may lead to job displacement. Workers in roles that are easily automated may face unemployment or the need to transition to new roles. This can lead to stress and anxiety regarding job security, affecting mental health and overall wellbeing. |
| People with disability | Automation and assistive technologies can significantly improve the quality of life for people with disabilities offering greater independence through smart home technologies, adaptive devices and mobility aids, making daily activities more manageable. | If automation leads to reduced job opportunities in certain sectors, individuals with disabilities may find it harder to secure employment. If not designed for inclusivity, automated systems may inadvertently marginalise those who are unable to use standard interfaces or technologies. |
| The nature and skills required for employment | Workers may need to pursue upskilling and reskilling initiatives to remain relevant in the job market. Educational institutions and employers will need to adapt to provide training in digital literacy, programming and critical thinking. | Widening the skills gap, where those with access to education and training thrive, while others may be left behind. Socioeconomic disparities can affect who can take advantage of new opportunities. |
| Production efficiency, waste and the environment | Automation can lead to improved production efficiency, allowing companies to produce goods faster and at a lower cost. This should translate into lower prices for consumers and increased competitiveness for businesses. Automated systems can optimise resource use, reducing waste through better inventory management and precise manufacturing processes. For example, automation in agriculture can lead to more efficient water and fertiliser usage. | Increased production capacity may lead to higher consumption rates and environmental degradation if not managed sustainably. Automation could also contribute to increased energy consumption if systems are not designed with energy efficiency in mind. |
| The economy and distribution of wealth | Automation could drive economic growth by increasing productivity and enabling new business models. It may create new markets and opportunities for innovation. | The benefits of automation may not be evenly distributed. Wealth generated may go to business owners and shareholders rather than workers making income inequality and social tensions worse. The labour market become polarised, with high-skill, high-wage jobs growing while low-skill, low-wage jobs decline. This can create a divide between those who can use automation to their advantage and those who cannot. |

**Activity 50**: students summarise the content of the table or participate in a debate.

|  |
| --- |
| The impact of automation is a complex interplay of benefits and challenges. While it has the potential to enhance safety, efficiency and economic growth, it also poses risks related to job displacement, inequality and environmental sustainability. To maximise the positive effects of automation while mitigating the negative ones, it is crucial for policymakers, educators and businesses to collaborate in creating a future that prioritises inclusivity, skill development and sustainable practices. This involves investing in education and training, promoting equitable access to technology and ensuring that the benefits of automation are shared broadly across society. |

#### Case study: Robodebt

The ‘Robodebt’ scheme, officially known as the ‘Online Compliance Intervention’ (OCI), was a program implemented by the Australian Government's Department of Human Services (now Services Australia) to identify and recover alleged overpayments of welfare benefits. The program used data matching and automated processes to compare income reported by welfare recipients against income reported to the Australian Taxation Office (ATO).

**Activity 51**: students research the Robodebt scheme and respond to the following questions.

The following links are to articles from the Information Age.

* [Robodebt was an AI ethics disaster](https://ia.acs.org.au/article/2021/robodebt-was-an-ai-ethics-disaster.html)
* [‘Failure of government’: Public servants breached code over Robodebt](https://ia.acs.org.au/article/2024/-failure-of-government---public-servants-breached-code-over-robo.html)
* ['Crude and cruel': shameful Robodebt findings laid bare](https://ia.acs.org.au/article/2023/crude-and-cruel-shameful-robodebt-findings-laid-bare.html)

1. Suggest the machine learning algorithms that were most likely used in the ‘Online Compliance Intervention’

|  |
| --- |
| Specific details about the exact algorithms used in the Robodebt scheme are not publicly disclosed. There are however several machine learning and data analysis techniques are commonly associated with similar data matching and compliance intervention programs. Algorithms and techniques that could be related to the Robodebt process include:   * Linear regression to model the relationship between reported income and welfare payments, potentially identifying discrepancies. * Logistic regression to classify cases as likely overpayments or not based on various predictors. * Data matching algorithms to compare and match datasets from different sources (for example, ATO income data and welfare recipient reports) to identify inconsistencies. * Techniques like fuzzy matchingto handle variations in data entry. * Anomaly detection techniques to identify outliers in reported income compared to expected patterns, which may suggest errors or discrepancies. * Machine learning models such as support vector machines (SVM) or Isolation Forests can be trained to detect anomalies in income reporting. * Decision tree algorithms could be used to create rules for identifying cases that warrant further investigation based on various features of the income data. * Random Forestscould be used to classify cases into categories (for example, likely overpayment, likely correct payment) based on historical data. * Natural language progression (NLP) to process free-text comments or notes from caseworkers, to extract relevant information. |

1. Summarise the ethical concerns raised by the Robodebt scheme.

|  |
| --- |
| The Robodebt program faced significant criticism and legal challenges due to its reliance on automated data matching and the lack of procedural fairness. Many recipients were incorrectly pursued for debts that they did not owe, leading to harsh financial and emotional consequences. This highlights the importance of ethical considerations in the use of machine learning algorithms, particularly in sensitive areas like social welfare and compliance. |

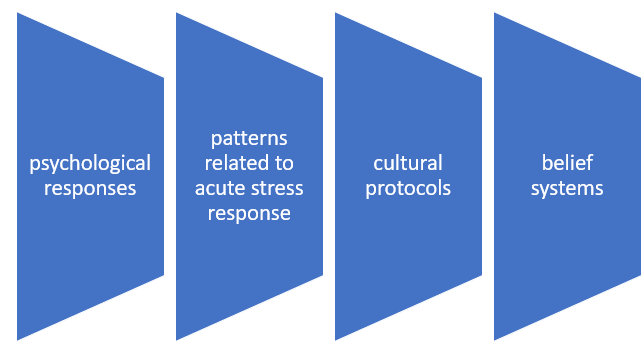
1. Research and list other incidents where the use of AI and ML have caused social and ethical concerns.

|  |
| --- |
| * Bias hiring algorithms * Google photos misclassification * Facial recognition technology * Credit scoring algorithms * Social media algorithms * AI in healthcare and diagnosis |

### Explore by implementation how patterns in human behaviour influence ML and AI software development

**Including:** **psychological responses, patterns related to acute stress response, cultural protocols, belief systems**

Patterns in human psychological responses play a significant role in the development of machine learning (ML) and artificial intelligence (AI) software. Understanding how people think, behave and respond to technology can help developers create systems that are more intuitive, effective and user-friendly. Students consider the effects of not addressing these issues during development. They consider these factors in the development of their projects and provide examples that can be explore during implementation.



**Activity 52**: Jigsaw

Students form teams of 2 to 5 (depending on the class size).

Students in each team are ‘numbered off’ 1 to 4 (representing the 4 ‘human behaviours’)

Each group investigates, with research, one of the human behaviours that influence ML and AI software development.

Each group provides an example or scenario of their human behaviour.

* Group 1 research psychological responses
* Group 2 research patterns related to acute stress response
* Group 3 research cultural protocols
* Group 4 research belief systems

Teachers make explicit that these human behaviours are closely related with concepts that intersect and overlap with each other.

Groups return to their teams to share their findings in a class presentation.

**Group 1:** students watch [How Netflix's Recommendation System Works [NRE] (10:34)](https://youtu.be/1k_vvAFamEo?si=PmEnPPGLVX8OIuvr)

1. How do psychological responses influence ML and AI software development (refer to UX, personalisation, trust and predicting behaviour)

|  |
| --- |
| Developers should design for user experience (UX) and consider how users will interact with AI systems. By understanding cognitive biases, attention spans, and user preferences, they can design interfaces that facilitate easy navigation and comprehension. ML algorithms can be trained to recognise patterns in user behaviour and preferences, allowing for personalised experiences. Understanding psychological responses can help in curating content. Designing AI systems that users trust is critical. Psychological factors like transparency, explainability, and perceived reliability can influence how users accept and adopt AI technologies. AI can be used to predict human behaviour based on historical data. Understanding patterns in psychological responses allows developers to create models that can anticipate user actions more accurately. |

1. Provide an example.

|  |
| --- |
| Recommendation system like streaming service Netflix.  Viewers can often exhibit the ‘paradox of choice’, where too many options can lead to confusion and decision fatigue. This causes users to feel overwhelmed, potentially leading to dissatisfaction or abandoning the service. To combat this, streaming services use recommendation algorithms that analyse users' viewing histories, preferences, and behaviour patterns. By understanding psychological responses, developers design these systems to simplify choice by presenting a curated selection of shows or movies based on previous viewing habits, reducing the cognitive load on users. They enhance engagement by using collaborative filtering to suggest content that similar users enjoyed, tapping into social proof, which can influence users' perceptions of what to watch. They create a feedback loop by continuously refining recommendations based on user interactions (likes, dislikes, watch time), ensuring that the system adapts to changing preferences over time. |

**Group 2**: students watch [Artificial intelligence: A gamechanger in the emergency room (3:22).](https://www.dw.com/en/artificial-intelligence-a-gamechanger-in-the-emergency-room/video-67452562)

1. How do patterns related to acute stress responses influence the development of machine learning (ML) and artificial intelligence (AI) software? (Include UI, feedback, adaptive systems and communication).

|  |
| --- |
| In high-stress situations (for example, emergency response, medical settings) interfaces should be designed to minimise cognitive load. Clear, concise instructions and easily accessible information are crucial to prevent overwhelming users. Stress responses often require quick decision-making. AI systems can be designed to provide real-time feedback and suggestions, enhancing situational awareness and helping users make informed decisions rapidly. Understanding stress responses allows developers to create adaptive systems that change their behaviour based on the user’s emotional state. For instance, if a user is detected to be in a high-stress situation, the system could simplify complex tasks or offer more guidance. In stressful scenarios, how AI systems communicate with users is critical. Systems can be programmed to use calming language and provide reassurance, which can help mitigate stress. |

1. Provide an example.

|  |
| --- |
| In an emergency response application,an AI-powered emergency response app could be used by first responders during a crisis. First responders may experience acute stress during emergencies, leading to heightened anxiety, narrowed attention and the need for quick decision-making.  The app features a streamlined interface with large buttons and clear icons for critical functions (for example, calling for backup, accessing patient information). The app uses ML algorithms to analyse incoming data (for example, location, severity of the incident) and prioritises alerts for the most critical situations, allowing responders to focus on urgent matters. If the app detects signs of stress (for example, rapid voice input or hesitation in touch inputs), it could switch to a more guided mode, providing step-by-step instructions for tasks (for example, triaging patients) to help alleviate cognitive overload.  The app uses natural language processing to communicate with responders in a calm and reassuring tone, providing updates and reminders in a supportive manner. |

**Group 3**: students watch [Shaping AI through Cultural Diversity (6:08)](https://youtu.be/AiK0iYZuNS0?si=r2JMj4Ko2sbEQLXB).

**Homework**: students read [Delivering Indigenous Data Sovereignty](https://aiatsis.gov.au/publication/116530)

**Extension:** [students watch AI Ethics Course: Indigenous Data Sovereignty (39:21)](https://youtu.be/g8qeZihLf1Q?si=zS_50YC6LHJ4i9lk)

**Other readings**: [AI technology is showing cultural biases, here's why and what can be done](https://www.murdoch.edu.au/news/articles/ai-technology-is-showing-cultural-biases-here-s-why-and-what-can-be-done)

1. How do cultural protocolsinfluence ML and AI software development? Consider data representation, UI and UX, the use of NLP, ethical issues and trust.

|  |
| --- |
| Cultural differences can affect how data is collected, represented and interpreted. Datasets need to be representative of the cultural diversity of the user base to avoid biases in the model. Cultural preferences influence design elements such as colours, symbols, language and layout. Understanding these preferences helps create interfaces that resonate with users from different backgrounds. Language use varies across cultures, including slang, idioms and dialects. NLP systems must be trained on culturally relevant data to understand and generate appropriate language for different user groups. Certain actions or features may be acceptable in one culture but offensive in another. Developers must be aware of these nuances to avoid cultural insensitivity in their AI applications. Different cultures have varying levels of trust in technology. Understanding these differences can guide how AI systems interact with users, ensuring that communications and functionalities align with users’ cultural expectations. |

1. Provide an example.

|  |
| --- |
| Cultural norms dictate how people interact, including formal versus informal communication styles, greetings and respect for hierarchy. A virtual assistant needs to adapt its communication style based on cultural context. For example, in cultures that value formality, the assistant uses titles and formal language, while in more casual cultures, it adopts a friendly and informal tone. The assistant needs to be trained on diverse datasets that include various dialects and regional languages, allowing it to understand and respond appropriately to users from different backgrounds. The assistant would recognise local holidays or cultural events and offers culturally relevant suggestions or greetings. For instance, it might remind users about Diwali or Lunar New Year in relevant contexts. The app's design incorporates culturally relevant colours, symbols and imagery. For example, it avoids using colours that may have negative connotations in certain cultures (for example, white is often associated with mourning in some cultures). |

**Group 4**: students watch and discuss [A.C. Grayling – How Belief Systems Work (8:12)](https://youtu.be/myLT3vpgxsA?si=AvCYuMz4BW4m1WEE)

1. How should belief systems influence ML and AI software development? Consider aligning values, mitigating against bias, user trust and acceptance, cultural sensitivity and decision making.

|  |
| --- |
| AI systems need to align with the ethical values and beliefs of the user base. For instance, an AI used in healthcare must prioritise patient welfare, informed consent and privacy, reflecting societal beliefs about health and ethics. Developers must recognise that belief systems can introduce biases into training data. Understanding these biases is essential for creating fair and equitable algorithms that do not perpetuate existing inequalities or stereotypes. Users are more likely to trust AI systems that reflect their beliefs and values. Developers should consider how belief systems influence user perceptions of transparency, accountability and the moral implications of AI decisions. Different belief systems can inform cultural practices and norms. AI applications must be sensitive to these differences to avoid offence and to promote acceptance.  Belief systems can affect how users make decisions. AI systems designed to assist in decision-making should consider the values and priorities that drive users’ choices. |

1. Provide an example.

|  |
| --- |
| Students consider how belief systems about diversity, inclusivity and fairness in the workplace can significantly impact how hiring practices are perceived and implemented. They reflect upon these in proposing the design of an AI system that assists in the hiring process for a company. The AI should be trained on diverse datasets that reflect various demographics to avoid bias against certain groups. Developers must ensure that the model does not favour candidates based on gender, race or socioeconomic status, aligning with the belief in equality and fairness.  The AI can analyse job descriptions to ensure they are inclusive and free of biased language that may discourage certain candidates from applying. For example, using gender-neutral language and focusing on skills rather than experience that might be unfairly attributed to specific demographic groups. The AI system should provide transparent reasoning for its recommendations, allowing hiring managers to understand how decisions were made. This aligns with the belief that individuals should have insight into the processes that affect their careers.  The AI could be programmed to consider cultural differences in communication styles and qualifications, ensuring that it does not overlook qualified candidates due to cultural biases. |

**Activity 53**: students create a summary of the presentations.

|  |
| --- |
| By integrating insights from psychology into the development of ML and AI software, developers can create more effective, user-friendly systems that enhance user satisfaction and engagement. Understanding human psychological responses helps in designing algorithms that not only meet technical requirements but also resonate with users on a cognitive and emotional level. By acknowledging patterns related to acute stress responses, developers can create ML and AI systems that are not only functional but also empathetic and supportive in high-stress situations. This understanding helps ensure that technology enhances decision-making and user experience, ultimately leading to better outcomes in critical scenarios. Incorporating patterns related to cultural protocols into ML and AI software development is essential for creating inclusive and effective technologies. By understanding cultural differences and respecting diverse user needs, developers can build systems that enhance user experience, foster trust and ensure that technology serves a global audience effectively. By understanding and integrating patterns related to human belief systems, developers can create ML and AI systems that are ethical, fair and aligned with the values of their users. This approach enhances user trust, promotes acceptance and ensures that technology serves society in a way that reflects its diverse beliefs and values. |

### Investigate the effect of human and dataset source bias in the development of ML and AI solutions

**Teacher note**: the development of machine learning (ML) and artificial intelligence (AI) solutions is significantly influenced by the biases that can arise from both human decision-making and the datasets used to train models. Understanding these biases is critical for creating fair, effective and reliable AI systems.

**Activity 54**: teacher-led discussion on [Shedding light on AI bias with real world examples](https://www.ibm.com/think/topics/shedding-light-on-ai-bias-with-real-world-examples).

Students watch [Machine Learning: Bias in, Bias out | Experience AI (3:23)](https://youtu.be/MANoa78BQAQ?si=J2hczjnp8NHNdr_O).

**Homework**: students watch [Human Bias in Machine Learning (43:35)](https://youtu.be/oTOUzpzKYDI?si=10KbrLKV9lWpXvVj).

1. In the space below provide a definition of human bias.

|  |
| --- |
| Human bias refers to the subjective judgements and decisions made by individuals during the development of ML and AI systems.  These biases can affect various stages of the AI lifecycle, including data collection, labelling, model selection and evaluation. |

1. In the space below provide examples of types of human bias.

|  |
| --- |
| **Selection bias**: when developers choose specific datasets that reflect their own preferences or experiences, leading to models that do not generalise well to diverse populations. For example, a facial recognition system developed primarily using images of lighter-skinned individuals may perform poorly on darker-skinned individuals.  **Confirmation bias**: developers may unintentionally prioritise results that confirm their existing beliefs or hypotheses, ignoring contradictory data that could lead to more balanced conclusions.  **Cognitive bias**: the subjective judgements of developers can lead to biases in feature selection, model design and evaluation metrics. For instance, a team may overvalue certain features based on personal experiences rather than objectively assessing their relevance. |

1. In the space below, discuss the effects of human bias.

|  |
| --- |
| Models trained with biased data or influenced by human bias may not perform well across diverse scenarios, leading to inaccurate predictions. Biased models can perpetuate or amplify societal inequalities, leading to unfair treatment of certain groups or individuals (for example, biased hiring algorithms, discriminatory loan approval systems). |

1. In the space below provide a definition of dataset source bias.

|  |
| --- |
| Dataset source bias occurs when the data used to train machine learning models is unrepresentative of the target population or the real-world scenarios in which the models will be applied. This bias can arise from the methods of data collection, the specific populations sampled and the inherent characteristics of the data sources. |

1. In the space below provide examples of types of dataset source bias.

|  |
| --- |
| * **Sampling bias**:if a dataset is collected from a specific demographic group, the resulting model may not generalise to others. For example, health data collected primarily from urban populations may not be applicable to rural populations. * **Labelling bias**: bias can be introduced during the labelling process if annotators have their own biases or if the labelling criteria are inconsistent. For instance, a sentiment analysis model trained on product reviews labelled by individuals with different cultural backgrounds may misinterpret sentiments. * **Historical bias**: datasets that reflect historical inequalities (for example, criminal justice data showing higher arrest rates in certain communities) can lead to models that reinforce those biases. AI systems trained on such data may perpetuate existing prejudices. |

1. In the space below discuss the effects of dataset source bias.

|  |
| --- |
| Models trained on biased datasets can lead to decisions that disproportionately disadvantage specific groups, exacerbating existing societal issues. If users perceive that AI systems are biased or unfair, it can erode trust in technology and its applications, impacting adoption and usage. |

**Activity 55**: teacher-led discussion on what can be done to address human and dataset source bias.

**Activity 56**: students list and describe mitigation strategies developers should use.

|  |
| --- |
| * Ensure that datasets are collected from a wide range of sources and populations to capture diverse perspectives and experiences including actively seeking out underrepresented groups to include in the dataset. * Conducting regular audits to assess bias in datasets and models including testing models on different demographic groups and measuring performance disparities. * Implementing fairness metrics to evaluate whether models treat all groups equitably. * Promoting transparency in data collection and model development processes can help stakeholders understand potential biases and the rationale behind model decisions. * Using explainable AI techniques to provide insights into how models make decisions can help identify and address biases. * Providing training for developers and data scientists on recognising and mitigating bias in AI systems can lead to more objective decision-making. * Encouraging diverse teams in AI development to bring different perspectives can reduce the likelihood of groupthink. * Continuously monitor models in production to identify and address biases as they arise and surveying user feedback can be invaluable in identifying issues that might not have been found in development. |

**Activity 57**: students complete the table below to describe how human and data source bias could occur in their assessment task.

Table 20 – data-sourced bias

|  |  |  |
| --- | --- | --- |
| Type of bias | Description | Consequences |
| Selection bias | If the training data is not representative of the entire student population, the model may produce biased predictions. | Leads to inaccurate predictions for underrepresented groups, reinforcing existing inequalities. |
| Measurement bias | Inaccuracies or inconsistencies in the data (for example, grading errors, subjective grading) can introduce measurement bias. | The model may learn from flawed data, resulting in predictions that reflect inaccuracies. |
| Feature bias | Bias can occur if the model relies heavily on certain features (for example, prior grades, socioeconomic status) that do not capture a student's full abilities. | Disadvantages students with barriers to performance or who have high potential not reflected in data. |

Table 21 – human bias

|  |  |  |
| --- | --- | --- |
| Type of bias | Description | Consequences |
| Confirmation bias | Developers may select features or data that confirm their beliefs about certain students or groups, reinforcing stereotypes. | Leads to biased predictions that do not accurately reflect a student's true potential. |
| Implicit bias | Educators' unconscious biases can influence assessments and grade reporting, affecting the data used to train the model. | The model may perpetuate these biases, leading to unfair predictions for marginalised students. |
| Feedback loop | If predictions influence future assessments or opportunities, this creates a feedback loop that reinforces existing biases. | Results in a cycle of disadvantage for certain groups of students, making it difficult to overcome biases. |

## Appendix

### Assessment task sample solutions

Contents

Ensure that your report contains a Table of contents.

To insert a Table of contents in Microsoft Word:

1. Navigate to **References** > **Table of Contents** > **Custom Table of Contents**.
2. Choose the number of heading levels you would like to display in the Table of contents.
3. Select **OK**.

To update the Table of contents:

1. Right click on the table and select **Update Field**.
2. Select **Update entire table**. Your table numbers should then update to reflect your changes.

### Outline

This document provides a guide for teachers in what a low, medium and high-grade student might produce for this assessment task.

The solutions are broken down into how they should progress and include explanations.

It is important to remember that these are only samples and there are a number of ways in which a solution to this assessment could be produced.

Students are to produce all system and data modelling with reference to those tools used and described in the [Course specifications.](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF)

### Setting up the work environment

For this project, students will need as a minimum:

* NumPy – Python module (documentation: <https://numpy.org/doc/stable/>)

**Note**: it is possible for the solutions to work without using this library however the [Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) document uses this in their example so it is included here for completeness.

* scikit-learn – Python module (documentation: <https://scikit-learn.org/stable/getting_started.html>)
* pip [Install](https://scikit-learn.org/stable/install.html#installation-instructions) is a package manager for Python packages, or modules.

Students may also want to setup the following for their general UI:

* [TKinter](https://www.pythontutorial.net/tkinter/)
* [PyGame](https://www.pygame.org/docs/)
* [Flask](https://flask.palletsprojects.com/en/stable/)
* [Matplotlib](https://matplotlib.org/stable/users/index.html) – for creating graphs.

It is anticipated that students will have experimented with the sample code on page 28 of the [NESA Course specifications](https://library.curriculum.nsw.edu.au/341419dc-8ec2-0289-7225-6db7f2d751ef/adfaf192-c7f1-4774-9317-6a51e26a6262/software-engineering-11-12-2022-higher-school-certificate-course-specifications.PDF) (or similar code) prior to starting on the assessment so they are familiar with the scikit-learn module.

### Low-grade solution

This solution will be text based only and will produce an estimate by simply looking at the students' marks in other tasks.

As such it is a rough estimate and does not consider the relative difficulty differences between tasks.

The solution makes use of functions but does not make use of classes.

A student may implement the functionality as one ‘big blob of commands’ without the use of functions.

A full script with comments is provided in the task resources folder called **solution\_low.py**.

Use the csv file **MarksOnly.csv** with this solution.

**Note**: classes could be guided through this solution to identify shortcomings and discuss ways of improving.

### Code explanation

#### General setup

|  |  |
| --- | --- |
| Explanation | Code |
| * Setup the top section, import relevant modules and provide a basic description of the script. | # This script will import a markbook and allow you to estimate a mark  # Written by Bruce Banner - 5/11/2024  # Version 0.4  import numpy  import csv  from sklearn.linear\_model import linear\_regression |
| * Create a main function with a general structure for how the program will run. * Code the name of the file to import data from. * Think through the general steps that will be involved in completing their project. | def main () :  data = import\_data (data\_file)  student, task = get\_which\_task ()  estimate = process\_estimate (data, student, task)  show\_result (student, task, estimate)  main () |
| * Create stub functions for each of those identified in the main function. | def import\_data (data\_file) :    return True    def get\_which\_task () :    return True, True      def process\_estimate (data, student, task) :    return True  def show\_result (student, task, estimate) :  return True |

#### Input

|  |  |
| --- | --- |
| Explanation | Code |
| * A simple CSV file has been created with only the marks in it. * Use the CSV module to read the data in. * The file name has been hard-coded which is not ideal but keeps things simple. * The returned array is printed as a debugging output statement to verify that it works as intended. * The data is imported as strings and needs to be converted to integers. * The imported data has not been validated. | def import\_data (data\_file) :  marks = []  marks\_file = open(data\_file)  marks\_raw = csv.reader(marks\_file)  for student in marks\_raw :  student\_marks\_int = []  for mark in student :  student\_marks\_int.append(int(mark))  marks.append(student\_marks\_int)    return marks  def main () :  data\_file = 'marksSimple.csv'  marks = import\_data (data\_file)  print (marks) # Debugging output statement |
| * The data has been input. * There is no information presented to the user as to which students and tasks are available to choose from. * There is no indication of missing marks provided. * The input data is not validated to make sure it is for a student and task that actually exist. | def get\_which\_task () :  student = int(input("Which student to estimate the mark for : "))  task = int(input("Which task to estimate the mark for : "))  return student, task |

#### Processing

|  |  |
| --- | --- |
| Explanation | Code |
| * The processing is basic. * The training data has been put together without any regard for potentially missing task data. It will only work assuming that no task data is missing. * The general approach matches that which is provided in the Course specifications document. * **Note**: data has not been structured using NumPy. * It still works assuming all the data is fine. | def process\_estimate (data, student, task) :  tasks\_base = []  marks\_base = []  counter = 1  for task\_number in data[0] :  if counter != task :  task\_title = [counter]  tasks\_base.append(task\_title)  marks\_base.append(data[student-1][counter - 1])  counter = counter + 1  model = linear\_regression()  model.fit(tasks\_base, marks\_base)  mark\_prediction = model.predict([[task]])  return mark\_prediction[0] |

#### Output

|  |  |
| --- | --- |
| Explanation | Code |
| * The information is printed in a basic format. | def show\_result (student, task, estimate) :  print(f"Student {student} has an estimated mark of {estimate} for task {task}") |

**Teacher note**: this solution works but gives quite a rough estimate. It also has to have several things ‘just right’ for it to work.

### Medium-grade solution

This solution will be text-based only but is set up in a manner that it can easily be expanded into a graphical or partially graphical implementation.

It is class-based, which helps to structure the code in a manner that allows for easy expansion.

This implementation does basic validation on data.

It provides an estimate by looking at the marks for other students for the task and their ranks.

It then estimates what their rank should be for the task based upon their rank in other tasks.

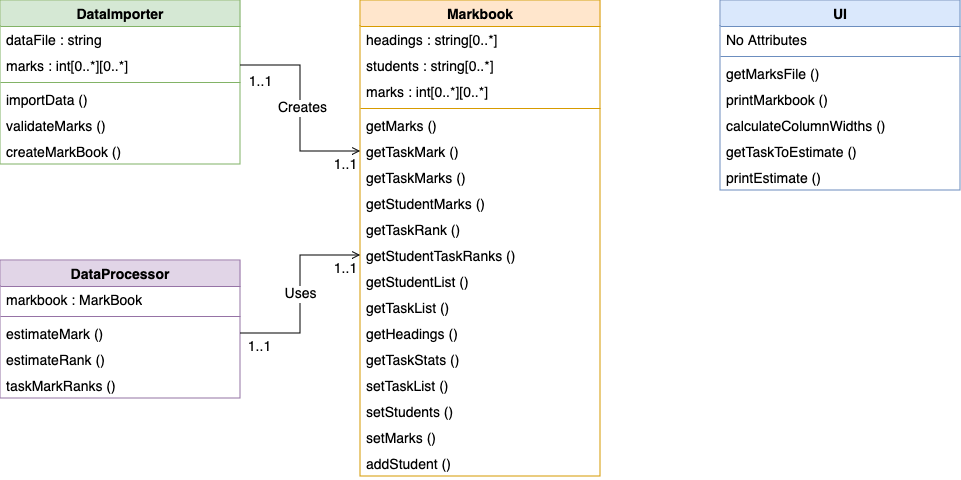
A full script with comments is provided in the task resources folder called **solution\_medium.py.**

Use the CSV file **MarksWithHeadings.csv** with this solution.

#### IPO table

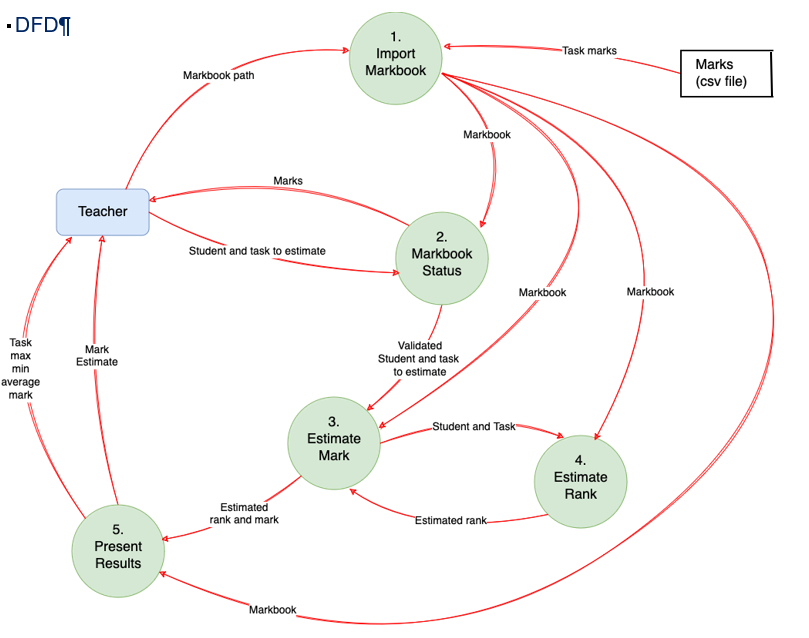
|  |  |  |  |
| --- | --- | --- | --- |
| Outcome | Input | Process | Output |
| Estimate rank | Marks for each task  Student to estimate  Task to estimate | Work out the students rank in each of the other tasks in the markbook.  Create a linear regression model with Task number on the x-axis and Rank on the y-axis.  Interpolate to work out a suggested rank for that student for the given task. | Estimated rank |
| Estimate mark | Student to estimate  Task to estimate  Estimated rank  Marks for that task | Adjust the ranks for students in the given task, leaving a gap at the estimated rank position.  Create a linear regression model with Rank on the x-axis and Mark on the y-axis.  Interpolate to work out a suggested mark for that student for the given task. | Estimated mark |

### Class diagram



**Note**: this diagram is created to the NESA Course specifications document outline. Students may wish to add further detail such as indication of public/private methods and method parameters. Student may also consider creating classes for a student and task.

#### Data Flow Diagram



#### Data dictionary

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Variable | Data type | Format for display | Size in bytes | Size for display | Description | Example | Validation |
| student | int |  | 2 |  | Used as an index into the student’s array. | 7 |  |
| task | int |  | 2 |  | Used as an index into the tasks array. | 2 |  |
| rank\_estimate | int |  | 2 |  | An estimate of the rank for a student for a particular task. | 4 | Cannot be larger than the number of students in the markbook. |
| mark\_estimate | int |  | 2 |  | An estimate of the mark for a student for a particular task. | 67 | Cannot be greater than 100 or less than 0. |
| students | array(string) |  |  |  | A listing of students in the markbook. |  |  |
| tasks | array(string) |  |  |  | A listing of the tasks represented in the markbook. |  |  |
| marks | array(array(int)) |  |  |  | A two-dimensional array of marks. The marks are grouped by student as opposed to task. |  |  |
| mark\_rank | int |  | 2 |  | The rank that a particular mark got in a task. | 5 | Cannot be larger than the number of students in the markbook. |
| student\_data |  |  |  |  |  |  |  |
| Markbook | record |  |  |  |  |  |  |
| headings | array(string) |  |  |  |  |  |  |
| marks | array(int) |  |  |  |  |  |  |
| students | array(string) |  |  |  |  |  |  |

**Note**: not all variables have been included. Only globals, parameters, return values and object attributes. Students should complete this data dictionary.

### Code explanation

**Note:** not all the code will be detailed here, only the main elements of processing.

#### main function

|  |  |
| --- | --- |
| Explanation | Code |
| * Although we are using OOP in this solution, the flow of processing still runs in a fairly linear fashion. * As such we will coordinate that processing through a function to keep it a little simpler and cleaner. | # Manage the overall processing  def main () :  ui = UI()  # Set up Markbook  data\_file = ui.get\_marks\_file()  data\_importer = data\_importer(data\_file)  data\_importer. import\_data()  markbook = data\_importer.create\_mark\_book()  # Show details of markbook currently  ui.print\_markbook(markbook)  # Input the task to estimate  student, task = ui.get\_task\_to\_estimate(markbook)  #student = 7  #task = 2  # Process the estimate  data\_processor = data\_processor(markbook)  rank\_estimate = data\_processor.estimate\_rank(student, task)  mark\_estimate = data\_processor.estimate\_mark(student, task)  # Report on results  ui.print\_estimate(student, task, mark\_estimate, rank\_estimate, markbook)    main() |

#### data\_importer : import\_data ()

|  |  |
| --- | --- |
| Explanation | Code |
| * This function extracts the data out of the CSV file and puts it into a two-dimensional list converting data into relevant data types. * It should ideally use the validate\_marks method to make sure that the imported data is valid, but this has not been included here. | # import the data from a csv file.  # the first row should be headings, the first column should be student names  def import\_data (self) :  marks = []  marks\_file = open(self.data\_file)  marks\_rawInit = csv.reader(marks\_file)  marks\_raw = list(marks\_rawInit)  marks.append(marks\_raw.pop(0))  # iterate over each row in the file  for student in marks\_raw :  student\_marks\_int = []  student\_marks\_int.append(student.pop(0))  # iterate over each task converting the mark from a string to an integer  for mark in student :  if mark == '' :  student\_marks\_int.append(-1)  else :  student\_marks\_int.append(int(mark))  marks.append(student\_marks\_int)  self.marks = marks  return True |

### data\_processor : estimate\_mark ()

|  |  |
| --- | --- |
| Explanation | Code |
| * This function starts by creating a linear regression for ranks and marks for the given task that the mark is to be estimated for. * The x-axis is the rank for each grade. * The y-axis is the mark that the student at that rank got. * Next we need to call self.estimate\_rank as we need to get an estimate of the rank that the student should have for this assessment task. * We now finish by predicting what that rank would get for this task. | # estimate a mark based upon an estimated rank for the student for the given task  def estimate\_mark (self, student, task) :  # Create base data with the task to estimate omitted  # x axis  ranks\_base = [] # this needs to be an array of arrays with a single item which is the task  # y axis  marks\_base = [] # this is an array of ranks in other tasks  mark\_ranks = self.task\_mark\_ranks(task)    counter = 0  for entry in mark\_ranks :  if counter != student and entry[0] != -1 : # So that we omit the student that we are estimating  rank = [entry[1]]  ranks\_base.append(rank)  marks\_base.append(entry[0])  counter = counter + 1    #print(ranks\_base)  #print(marks\_base)    # Create the model  model = linear\_regression()  model.fit(ranks\_base, marks\_base)    # get the estimated rank for the student for the task  rankfor\_student\_estimate = self.estimate\_rank(student, task)    estimated\_mark = model.predict([[rankfor\_student\_estimate]])    return round(estimated\_mark[0]) |

#### data\_processor : estimate\_rank ()

|  |  |
| --- | --- |
| Explanation | Code |
| * This function looks at the rank that the student has achieved in other tasks. * It creates a linear regression to help us estimate their rank for this task. * This solution is unrefined. A student's rank in a task could vary depending on how it fits in with their strengths, however it is the best we can achieve given the data we have. * There are also only limited data points from which to predict. * The x-axis is the task for each grade. * The y-axis is the rank the student achieved for that task. | # estimates the rank for the student by looking at their ranks for other assessment tasks  def estimate\_rank (self, student, task) :  # Create base data with the task to estimate omitted  # x axis  tasks\_base = [] # this needs to be an array of arrays with a single item which is the task  # y axis  ranks\_base = [] # this is an array of ranks in other tasks  student\_ranks = self.markbook.get\_student\_task\_ranks(student)  counter = 0  for rank in student\_ranks :  if counter != task : # So that we omit the task that we are estimating  task\_title = [counter]  tasks\_base.append(task\_title)  ranks\_base.append(rank)  counter = counter + 1  # Create the model  model = linear\_regression()  model.fit(tasks\_base, ranks\_base)  estimated\_rank = model.predict([[task]])  return round(estimated\_rank[0]) |

#### Markbook

|  |  |
| --- | --- |
| Explanation | Code |
| This class and object includes a listing of the attributes and methods that present in the Markbook class.  These are getter and setter functions that make interacting with the data in the markbook easier and more convenient.  The intent is that the getter methods will cover all the different formats in which data is required from other areas of processing so that those areas become simpler and more concise to read. | # holds data on the students and tasks  # returns data in different ways  class Markbook :  self.headings = []  self.marks = []  self.students = []  # returns a 2dimentional array of marks [student][mark]  def get\_marks (self)  # returns the particular mark a student got for a task  # student is student number not name, zero indexed  def get\_task\_mark (self, student, task)  # return all tasks for a particular task  def get\_task\_marks (self, task)    # return marks for each task for a particular student  def get\_student\_marks (self, student)  # returns the rank a student got for a particular task  # student is student number not name  def get\_task\_rank (self, student, task)  # returns an array of the rank for each student for a particular task  def get\_student\_task\_ranks (self, student)  # returns a list of the students in the markbook  def get\_student\_list (self)  # returns a list of the names of the tasks in the markbook  def get\_task\_list (self)  # returns a list of all the headings in the markbook (Students heading + tasks)  def get\_headings (self)  # returns the min, max and average mark for a particular task  def get\_task\_stats (self, task)  # set the array of headings  def set\_task\_list (self, headings)  # set the students in the markbook  def set\_students (self, students)  # set the marks for the tasks  def set\_marks (self, marks)  # add a student and their marks to the student and marks lists  def add\_student (self, student\_data) : |

### Higher-grade solution

A solution will not be elaborated upon here.

Instead, we will discuss possible paths that a higher-grade student could explore.

It is expected that a student at this level should not need much in the way of support from a coding point of view but may benefit from some ideas on what could be pursued in terms of functionality.

#### Interface

At this level it would be expected that the student has a graphical UI for their product (though it could still be an elegant command line interface as well). The interface could be via a PWA or using a library such as TKinter.

The marks would be presented in a clean tabular format with the ability to view distributions of marks via a graph.

The estimate could be shown on the graph as well to give visual feedback on where the mark sits with respect to other marks.

#### Error checking/Validation

Error checking should be performed on the input file. Marks that are above 100% should not be allowed for instance.

It would also be expected that a certain degree of checking is done on the estimated mark to ensure that it is within reasonable ranges.

The marks available in the markbook should also be validated in terms of if it is possible to create a reasonable estimate. For example, not too many marks missing, enough students/tasks in the markbook, marks are not varying too wildly.

#### Processing

It is expected that the estimate would be based upon the distribution of marks from other students for that particular task.

It would then consider the marks that the student got in other assessment tasks and their rank in other assessment tasks to compute a trajectory for the student which will help in the estimate. Students can also make use of z-scores to analyse the volatility of the marks and give a confidence factor in their estimated mark.

### Using polynomial regression

Students can attempt to transform the existing low-code py solution from using linear regression to include polynomial regression. They will require these key changes:

1. Import Polynomial Features

* from sklearn.preprocessing import PolynomialFeatures which allows you to create polynomial features from the input data.

1. Create Polynomial Features:

* In the process\_estimate function, after preparing tasks\_base, you create polynomial features using:

poly = PolynomialFeatures(degree=2)

tasks\_base\_poly = poly.fit\_transform(tasks\_base)

This converts the input features into a format suitable for polynomial regression.

1. Fit the model:

* The model is trained on the polynomial features:

model.fit(tasks\_base\_poly, marks\_base)

1. Make predictions:

* When predicting, you must also transform the input task into polynomial features:

task\_poly = poly.transform([[task]])

mark\_prediction = model.predict(task\_poly)

Sample code is available in the **Software Automation** files folder of the Software Engineering channel in the TAS Statewide Staffroom.

### Using logical regression

Students can attempt to transform the low-code py solution from using linear regression to using logistical regression: They will require these key changes:

1. Change the Linear Regression model to Logistic Regression.

The model is now LogisticRegression() instead of LinearRegression().

1. Adjust the target variable (marks) to be binary.

The target variable marksBase has been replaced with pass\_fail\_base, which contains binary values (1 for pass, 0 for fail) based on whether the marks are equal to or greater than 50.

1. Modify the prediction logic accordingly.

The process\_estimate function now predicts whether the student will pass or fail for the specified task based on the logistic regression model.

1. Output function

The show\_result function has been updated to display whether the student is predicted to ‘pass’ or ‘fail’.

Assuming that a mark of 50 or above indicates a ‘pass’ (1) and below 50 indicates a ‘fail’ (0).

**Extension:** students attempt to modify the code so that it directly maps to existing training data when possible and predicts the value using linear regression when no exact values are known.

They combine previous techniques.

First check if the input value exists in the training data.

If it does return the corresponding output.

If it does not, use the linear regression model to predict the output.

1. Direct mapping with linear regression

* In the process\_estimate function, create a dictionary called existing\_marks that maps each task number to the corresponding mark for the specified student.
* Check if the requested task exists in existing\_marks. If it does, return the exact mark from the training data.

1. Prediction for unknown tasks

* If the task does not exist in the existing\_marks dictionary, we use the linear regression model to predict the mark based on the training data.

1. Output formatting

* The output format in show\_result is updated to display the estimated mark to 2 decimal places for clarity.

## References

This resource contains NSW Curriculum and syllabus content. The NSW Curriculum is developed by the NSW Education Standards Authority. This content is prepared by NESA for and on behalf of the Crown in right of the State of New South Wales. The material is protected by Crown copyright.

Please refer to the NESA Copyright Disclaimer for more information <https://educationstandards.nsw.edu.au/wps/portal/nesa/mini-footer/copyright>.

NESA holds the only official and up-to-date versions of the NSW Curriculum and syllabus documents. Please visit the NSW Education Standards Authority (NESA) website <https://educationstandards.nsw.edu.au> and the NSW Curriculum website <https://curriculum.nsw.edu.au>.

[Software Engineering 11–12 Syllabus](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview) © NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2022.

[Software Engineering course specifications](https://curriculum.nsw.edu.au/learning-areas/tas/software-engineering-11-12-2022/overview#software-engineering-course-specifications-software_engineering_11_12_2022) © NSW Education Standards Authority (NESA) for and on behalf of the Crown in right of the State of New South Wales, 2022.

[Computing Technology 7–10 Syllabus](https://curriculum.nsw.edu.au/learning-areas/tas/computing-technology-7-10-2022/overview) © NSW Education Standards Authority (NESA) for and on behalf of the Crown in the State of New South Wales, 2022.

3Blue1Brown (6 October 2017) ['But what is a neural network? | Deep learning chapter 1' [video]](https://youtu.be/aircAruvnKk?si=XPkBocjIDk92suWt), *3Blue1Brown*, YouTube, accessed 3 February 2025.

Aditya (11 October 2019) ‘[How to build a Neural Network from scratch](https://www.freecodecamp.org/news/building-a-neural-network-from-scratch/)’, *freeCodeCamp*, accessed 3 February 2025.

airSlate (17 March 2021) ['What is Business Process Automation? Pros, Cons, Myths & Tips' [video]](https://youtu.be/MM20QXEGdAc?si=xbo_adRkMYcoBM7j), *airSlate*, YouTube, accessed 3 February 2025.

Atlassian (2024) [*DevOps Best Practices*](https://www.atlassian.com/devops/what-is-devops/devops-best-practices), Atlassian website, accessed 3 February 2025.

——(2024)[*Considerations for your DevOps toolchain*](https://www.atlassian.com/devops/devops-tools/choose-devops-tools), Atlassianwebsite, accessed 3 February 2025.

——(2024)[*What is DevOps Culture?*](https://www.atlassian.com/devops/what-is-devops/devops-culture), Atlassianwebsite, accessed 3 February 2025.

Automation Anywhere (9 March 2021) ['What is RPA (Robotic Process Automation)? | Automation Anywhere' [video]](https://youtu.be/61K7JeJixnU?si=tqJ5M_G-KckMbG-c), *Automation Anywhere*, YouTube, accessed 3 February 2025.

Australian Computer Society (ACS) (2024) [*Information Age*](https://ia.acs.org.au/) [website], accessed 24 October 2024.

Australian Institute of Aboriginal and Torres Strait Islander Studies (AIATSIS) (2019) [*Delivering Indigenous Data Sovereignty*](https://aiatsis.gov.au/publication/116530), AIATSIS website, accessed 3 February 2025.

Baldwin C (10 November 2022) ['Introduction to Robotic Process Automation (RPA)' [video]](https://youtu.be/VKsHgNMsZqo?si=HGKmw54tgJFXKDRZ), *Cody Baldwin*, YouTube, accessed 3 February 2025.

Boesch G (5 October 2024) ‘[The Fundamental Guide to Faster R-CNN [2025]](https://viso.ai/deep-learning/faster-r-cnn-2/)’, *viso.ai*, accessed 3 February 2025.

Brunton S (31 December 2023) ['Supervised & Unsupervised Machine Learning' [video]](https://youtu.be/wvODQqb3D_8?si=0ceNPSLTQubrXQXe), *Steve Brunton*, YouTube, accessed 3 February 2025.

——(30 December 2023) ['A Machine Learning Primer: How to Build an ML Model' [video]](https://youtu.be/Vx2DpMgplEM?si=yAhFZaS1dtu1Wmhp), *Steve Brunton*, YouTube, accessed 3 February 2025.

——(5 January 2024) ['A Neural Network Primer' [video]](https://youtu.be/_56bfCu02ZE?si=7-LQilPzhoOW97vx), *Steve Brunton*, YouTube, accessed 3 February 2025.

Chugh V (13 February 2025) ‘[10 Top Machine Learning Algorithms & Their Use-Cases](https://www.datacamp.com/blog/top-machine-learning-use-cases-and-algorithms)’, *datacamp*, accessed 19 August 2024.

Closer To Truth (30 July 2022) ['A.C. Grayling – How Belief Systems Work' [video]](https://youtu.be/myLT3vpgxsA?si=B6ShIv7lzKXGZfNL), *Closer To Truth*, YouTube, accessed 3 February 2025.

CrashCourse (17 August 2019) ['Supervised Learning: Crash Course AI #2' [video]](https://youtu.be/4qVRBYAdLAo?si=Ivq3BVR1fdLyn8mf), *CrashCourse*, YouTube, accessed 3 February 2025.

——(24 August 2019) ['Neural Networks and Deep Learning: Crash Course AI #3' [video]](https://youtu.be/oV3ZY6tJiA0?si=X6W0-mgN65I6_kO5), *CrashCourse*, YouTube, accessed 3 February 2025.

——(21 September 2019) ['Unsupervised Learning: Crash Course AI #6' [video]](https://youtu.be/JnnaDNNb380?si=_1W8Bgl-SGljZO0C), *CrashCourse*, YouTube, accessed 3 February 2025.

——(12 October 2019) ['Reinforcement Learning: Crash Course AI #9' [video]](https://youtu.be/nIgIv4IfJ6s?si=P0hSTkdYk7t8fEr6), *CrashCourse*, YouTube, accessed 3 February 2025.

Desmos Studio, PBC (2025) [*Graphing calculator*](https://www.desmos.com/calculator), Desmos website, accessed 3 February 2025.

Deutsche Welle (23 November 2023) ['Artificial intelligence: A gamechanger in the emergency room' [video]](https://www.dw.com/en/artificial-intelligence-a-gamechanger-in-the-emergency-room/video-67452562), *Deutsche Welle,* accessed 3 February 2025.

Elastic Platform Team (22 April 2024) ‘[Exploring 11 popular machine learning algorithms](https://www.elastic.co/blog/popular-ml-algorithms)’, *elastic*, accessed 3 February 2025.

Experiments with Google (8 November 2019) ['Teachable Machine Tutorial 1: Gather' [video]](https://youtu.be/DFBbSTvtpy4?si=P5Vz3ns0i3QXEqR4), Experiments with Google, YouTube, accessed 3 February 2025.

——(8 November 2019) ['Teachable Machine Tutorial 2: Train' [video]](https://youtu.be/CO67EQ0ZWgA?si=3V-vuvFEMsdhLrAg), Experiments with Google, YouTube, accessed 3 February 2025.

——(8 November 2019) ['Teachable Machine Tutorial 3: Export' [video]](https://youtu.be/n-zeeRLBgd0?si=9kuv8egbT8HZpuL4), Experiments with Google, YouTube, accessed 3 February 2025.

Fireship (10 September 2021) ['Machine Learning Explained in 100 Seconds' [video]](https://youtu.be/PeMlggyqz0Y?si=6a_DbXFbvG6PkOA-), *Fireship*, YouTube, accessed 3 February 2025.

Fulkerson M (13 January 2024) ['Linear Regression (with Desmos)' [video]](https://youtu.be/_ldsNWOHHBk?si=YUvgyzSVFVF5aR3E), *Fulkerson M*, YouTube, accessed 3 February 2025.

Geeksforgeeks (16 January 2025) ‘[Random Forest Algorithm in Machine Learning](https://www.geeksforgeeks.org/random-forest-algorithm-in-machine-learning/)’, *Geeksforgeeks,* accessed 3 February 2025

——(15 January 2025) ‘[K means Clustering – Introduction](https://www.geeksforgeeks.org/k-means-clustering-introduction/)’, *Geeksforgeeks*, accessed 3 February 2025.

Google Cloud (2024) [*Artificial intelligence (AI) vs. machine learning (ML)*](https://cloud.google.com/learn/artificial-intelligence-vs-machine-learning), Google Cloud website, accessed 19 August 2024.

——(2024)[*What is MLOps?*](https://cloud.google.com/discover/what-is-mlops), Google Cloud website, accessed 19 August 2024.

Google (2024) [*Experiments with Google*](https://experiments.withgoogle.com/) [website], accessed 19 August 2024.

——(n.d.) [*Teachable Machine*](https://teachablemachine.withgoogle.com/) [website], accessed 3 February 2025.

Google Developers (2025) [Collaborative filtering](https://developers.google.com/machine-learning/recommendation/collaborative/basics), Machine Learning website, accessed 3 February 2025.

——(2025) [*Neural networks*](https://developers.google.com/machine-learning/crash-course/neural-networks), Machine Learning website, accessed 3 February 2025.

——(2025) [*Linear regression*](https://developers.google.com/machine-learning/crash-course/linear-regression), Machine Learning website, accessed 3 February 2025.

——(2025) [*Logistic Regression*](https://developers.google.com/machine-learning/crash-course/logistic-regression), Machine Learning website, accessed 3 February 2025.

Google for Developers (22 August 2024) ['Machine Learning Crash Course: Logistic Regression' [video]](https://youtu.be/72AHKztZN44?si=mOPEknssd_rVgxmS), Google for Developers*,* YouTube, accessed 3 February 2025.

Google Labs (2024) [*The home for AI experiments at Google*](https://labs.google/) [website], accessed 3 February 2025.

Google Open Online Education (25 February 2022) ['What is ML?' [video]](https://youtu.be/RdC7KZ6OFPA?si=rBgoIwLO_MgRCvTh), *Google Open Online Education*, YouTube, accessed 3 February 2025.

IBM (2025) [*What are convolutional neural networks?*](https://www.ibm.com/think/topics/convolutional-neural-networks), IBM website, accessed 3 February 2025.

——(2025) [*What is the k-nearest neighbors (KNN) algorithm?*](https://www.ibm.com/think/topics/knn), IBM website, accessed 3 February 2025.

——(2023) [*Shedding light on AI bias with real world examples*](https://www.ibm.com/think/topics/shedding-light-on-ai-bias-with-real-world-examples), IBM website, accessed 3 February 2025.

——(10 April 2023) ['AI vs Machine Learning' [video]](https://youtu.be/4RixMPF4xis?si=I4afUcPfrRjDusoP), *IBM Technology*, YouTube, accessed 3 February 2025.

——(20 September 2022) ['Machine Learning Basics: Supervised v Unsupervised' [video]](https://youtu.be/3fsy2oheRdg?si=ZIrvASF2mKRf8AcF), *IBM Technology*, YouTube, accessed 3 February 2025.

——(10 April 2024) ['Understanding Neural Networks and AI' [video]](https://youtu.be/NMZ0Tgc2jFQ?si=COrZ8WNTw2y1Osk0), *IBM Technology*, YouTube, accessed 3 February 2025.

——(24 May 2022) ['Neural Networks Explained in 5 minutes' [video]](https://youtu.be/jmmW0F0biz0?si=G6A8uHXFnCwNaNNg), *IBM Technology*, YouTube, accessed 3 February 2025.

——(20 July 2024) ['Machine Learning and Logistic Regression' [video]](https://youtu.be/AX-ZEC-71DI?si=Bcp213k56_3N68Wv), *IBM Technology*, YouTube, accessed 3 February 2025.

——(2 September 2024) ['What is the K-Nearest Neighbor (KNN) Algorithm?' [video]](https://youtu.be/b6uHw7QW_n4?si=Agl9gFuCo534OGEH), *IBM Technology*, YouTube, accessed 3 February 2025.

Intuitive Machine Learning (8 July 2020) ['Decision Tree: Important things to know' [video]](https://youtu.be/JcI5E2Ng6r4?si=2opUizUIkVeFGBXr), *Intuitive Machine Learning*, YouTube, accessed 3 February 2025.

Javapoint (2024) [*Difference between Artificial intelligence and Machine learning*](https://www.javatpoint.com/difference-between-artificial-intelligence-and-machine-learning), pointTech website, accessed 3 February 2025.

Jones B (2024) ‘[Practical Application of NESA Software Engineering MLOps](https://github.com/TempeHS/Practical-Application-of-NESA-Software-Engineering-MLOps)’, *TempeHS (Tempe High School*), GitHub website, accessed 18 November 2024.

——(2024) ‘[Code samples of data wrangling](https://github.com/TempeHS/Practical-Application-of-NESA-Software-Engineering-MLOps/blob/main/2.Model_Development/2.1.Data_Wrangling/2.1.1.data_preview.ipynb)’, *TempeHS (Tempe High School)*, GitHub website, accessed 24 October 2024.

——(2024) ‘[Linear regression](https://github.com/TempeHS/NESA_Course_Specifications_Linear_Regression)’, *TempeHS (Tempe High School*), GitHub website, accessed 18 November 2024.

Klinger N (11 October 2024) ‘[ImageNet Dataset: Evolution & Applications](https://viso.ai/deep-learning/imagenet/)’, *viso.ai*, accessed 3 February 2025.

Kubes Petr (2025) [*Neural network playground*](https://nnplayground.com/) [website], accessed 3 February 2025.

Lets Walk Around (28 October 2024) ['How Netflix's Recommendation System Works [NRE]' [video]](https://youtu.be/1k_vvAFamEo?si=-sT8X1S4JKTDLcIM), *Let’s Walk Around*, YouTube, accessed 3 February 2025.

LinkedIn (2025) [*How can you apply neural networks to education?*](https://www.linkedin.com/advice/1/how-can-you-apply-neural-networks-education), LinkedIn website, accessed 3 February 2025.

Lucid Software Inc (2024) [*Decision tree diagram maker*](https://www.lucidchart.com/pages/examples/decision-tree-maker?a=0), Lucidchart website, accessed 19 August 2024.

——(2024)[*What is a Decision Tree Diagram*](https://www.lucidchart.com/pages/how-to-make-a-decision-tree-diagram), Lucidchart website, accessed 3 April 2024.

——(2024) [*Lucidchart*](https://www.lucidchart.com/pages/?) [website], accessed 3 April 2024.

Matplotlib (2025) [*Matplotlib*](https://matplotlib.org/) [website], accessed 3 February 2025.

Microsoft (2024) [*Extensions for Visual Studio Code*](https://marketplace.visualstudio.com/vscode), Microsoft Visual Studio Marketplace website, accessed 7 August 2024.

Murdoch University (7 May 2024) ‘[AI technology is showing cultural biases, here’s why and what can be done](https://www.murdoch.edu.au/news/articles/ai-technology-is-showing-cultural-biases-here-s-why-and-what-can-be-done)’, *Murdoch University* News, accessed 3 February 2025.

NSW Education Standards Authority (NESA) (2024) ‘[Glossary of key words](https://www.nsw.gov.au/education-and-training/nesa/hsc/student-guide/glossary)’, *Student guide*, NSW Government website, accessed 18 November 2024.

NSW Education Standards Authority (NESA) (2023) [*Higher School Certificate Course Specifications: Software Engineering* [PDF 1.3 MB]](https://curriculum.nsw.edu.au/file/b746704f-f613-4bce-9466-19835b2623fc/software-engineering-11-12-2022-higher-school-certificate-course-specifications.pdf), NESA website, accessed 21 March 2025.

NumPy Developers (2008–2024) [*NumPy documentation*](https://numpy.org/doc/stable/), NumPy website, accessed 5 August 2024.

Oracle (2024) [*VirtualBox*](https://www.virtualbox.org/) [website], accessed 24 October 2024.

OxfordSparks (11 January 2017) ['What is Machine Learning?' [video]](https://youtu.be/f_uwKZIAeM0?si=9VNTuombR9Z7KyRT), *Oxford Sparks*, YouTube, accessed 3 February 2025.

Pal S (22 September 2023) ‘[A Practical Examination of 4 Pre-trained Models for Accuracy](https://geekpython.in/practical-examination-of-4-deep-learning-models)’, geekpython, accessed 3 February 2025.

Panini D (28 February 2024) ['Why You Need to Automate Processes in Business & Life' [video]](https://youtu.be/TparimIBWtU?si=nPhuW_LP-DHcevzh), *Demetri Panici*, YouTube, accessed 3 February 2025.

Polycode (1 April 2018) ['Create a Simple Neural Network in Python from Scratch' [video]](https://youtu.be/kft1AJ9WVDk?si=og3RY3K7TzRyPPIJ), *Polycode*, YouTube, accessed 3 February 2025.

Pregowska A and Osial M (11 August 2021) ‘[What Is An Artificial Neural Network And Why Do We Need It?](https://kids.frontiersin.org/articles/10.3389/frym.2021.560631)’, *Frontiers*, accessed 24 October 2024.

Rackspace Technology (13 December 2013) ['What is DevOps? - In Simple English' [video]](https://youtu.be/_I94-tJlovg?si=QqIFi0EXShfplk56), *Rackspace Technology*, YouTube, accessed 3 February 2025.

Raspberry Pi Foundation (2025) [Semantic Waves](https://static.raspberrypi.org/files/curriculum/quickreads/6-Pedagogy_Summary_Semantic_Waves_V3_2023.pdf) .pdf accessed 3 February 2025.

——(6 September 2024) ['Machine Learning: Bias in, Bias out | Experience AI' [video]](https://youtu.be/MANoa78BQAQ?si=IShz08fLnK5oWeee), *Raspberry Pi Foundation,* YouTube, accessed 3 February 2025.

RealPython (2025) [Building a Neural Network & Making Predictions (Overview)](https://realpython.com/videos/build-neural-network-ai-overview/), RealPython website, accessed 3 February 2025.

Sadler D (16 September 2024) ‘[‘Failure of government’: Public servants breached code over Robodebt](https://ia.acs.org.au/article/2024/-failure-of-government---public-servants-breached-code-over-robo.html)’, *Australian Computer Society*, accessed 3 February 2025.

scikit-learn (2025) [*Getting Started*](https://scikit-learn.org/stable/getting_started.html), scikit-learn website, accessed 3 February 2025.

Sheficial Intelligence Hub (12 November 2024) ['Machine Learning Essentials: From Linear Regression to Neural Networks | Companion Podcast' [video]](https://youtu.be/UIJ7r74v4AM?si=eNxFRX-AseBGTZwR), *Sheficial Intelligence Hub*, YouTube, accessed 3 February 2025.

Simplilearn (21 May 2020) ['RPA In 5 Minutes | What Is RPA - Robotic Process Automation? | RPA Explained | Simplilearn' [video]](https://youtu.be/9URSbTOE4YI?si=Iqikni2Ww9Rb5Jki), *Simplilearn*, YouTube, accessed 3 February 2025.

Simpson K (30 September 2024) ['Polynomial Regression for Machine Learning' [video]](https://youtu.be/Q8UfQ_aoCeQ?si=T3nyGmC1wcY8RmHQ), *Kody Simpson*, YouTube, accessed 13 March 2025.

State of New South Wales (Department of Education) (n.d.) [*Digital Learning Selector*](https://app.education.nsw.gov.au/digital-learning-selector/), NSW Department of Education website, accessed 5 August 2024.

——(2024) ‘[Connecting learning](https://education.nsw.gov.au/teaching-and-learning/curriculum/explicit-teaching/explicit-teaching-strategies/connecting-learning)’, *Explicit teaching strategies*, NSW Department of Education website, accessed 5 August 2024.

——(n.d.) [*Assessment schedule – software engineering, Year 12*](https://education.nsw.gov.au/teaching-and-learning/curriculum/tas/tas-curriculum-resources-7-12/tas-11-12-curriculum-resources/assessment-schedule-software-engineering-year-12), NSW Department of Education website, accessed 5 August 2024.

Stugard N (20 November 2024) ['How to choose the Right Machine Learning Algorithm' [video]](https://youtu.be/uh6iYQEHyyI?si=ebp4MbDOTjUd8xOQ), *Nick Stugard,* YouTube, accessed 3 February 2025.

The GovLab (3 March 2021) ['AI Ethics Course: Indigenous Data Sovereignty by Maui Hudson' [video]](https://youtu.be/g8qeZihLf1Q?si=4HVilywxJd3-pRnk), *The GovLab*, YouTube, accessed 3 February 2025.

TechWorld with Nana (19 January 2022) ['What is DevOps? REALLY understand it | DevOps vs SRE' [video]](https://youtu.be/0yWAtQ6wYNM?si=NcXoTHHgd6lm0LDM), *TechWorld with Nana*, YouTube, accessed 3 February 2025.

Tonkin C (29 June 2021) ‘[Robodebt was an AI ethics disaster](https://ia.acs.org.au/article/2021/robodebt-was-an-ai-ethics-disaster.html)’, *Australian Computer Society*, accessed 3 February 2025.

——(10 July 2023) ‘[Crude and cruel': shameful Robodebt findings laid bare](https://ia.acs.org.au/article/2023/crude-and-cruel-shameful-robodebt-findings-laid-bare.html)’, *Australian Computer Society*, accessed 3 February 2025.

Tran M (16 November 2022) ‘[Understanding U-Net](https://medium.com/towards-data-science/understanding-u-net-61276b10f360)’, *Medium*, accessed 3 February 2025.

TwinEd Productions (20 December 2020) ['Deep Neural Network (DNN) | Deep Learning' [video]](https://youtu.be/ZBnHoVMeeJE?si=mJ8xevx-6Bh9zfqX), *TwinEd Productions*, YouTube, accessed 3 February 2025.

Turp M (5 June 2021) ['How Do Decision Trees Work (Simple Explanation) – Learning and Training Process' [video]](https://youtu.be/xDWZzD4TPO0?si=xD0KVKC9yjcI81UR), *Misra Turp*, YouTube, accessed 3 February 2025.

UNESCO (10 March 2022) ['Shaping AI through Cultural Diversity' [video]](https://www.youtube.com/watch?v=AiK0iYZuNS0), *UNESCO*, YouTube, accessed 3 February 2025.

UNSW Teaching (2025) [*The Flipped Classroom*](https://www.teaching.unsw.edu.au/flipped-classroom), UNSW website, accessed 3 February 2025.

UVA School of Data Science (30 November 2021) ['Human Bias in Machine Learning' [video]](https://www.youtube.com/watch?v=oTOUzpzKYDI), *UVA School of Data Science*, YouTube, accessed 3 February 2025.

Van Otten N (31 January 2024) ‘[How To Implement Speech Recognition [3 Ways & 7 Machine Learning Models]](https://spotintelligence.com/2024/01/31/speech-recognition/)’, *Spot Intelligence*, accessed 3 February 2025.

Visually Explained (28 November 2021) ['Linear Regression in 2 minutes' [video]](https://youtu.be/CtsRRUddV2s?si=W3u4Y0dXc15mKJK9), *Visually Explained*, YouTube, accessed 3 February 2025.

## Legislation

*Privacy and Personal Information Protection Act 1998* (No. 133)

**© State of New South Wales (Department of Education), 2025**

The copyright material published in this resource is subject to the Copyright Act 1968 (Cth) and is owned by the NSW Department of Education or, where indicated, by a party other than the NSW Department of Education (third-party material).

Copyright material available in this resource and owned by the NSW Department of Education is licensed under a [Creative Commons Attribution 4.0 International (CC BY 4.0) license](https://creativecommons.org/licenses/by/4.0/).

[](https://creativecommons.org/licenses/by/4.0/)

This license allows you to share and adapt the material for any purpose, even commercially.

Attribution should be given to © State of New South Wales (Department of Education), 2025.

Material in this resource not available under a Creative Commons license:

* the NSW Department of Education logo, other logos and trademark-protected material
* material owned by a third party that has been reproduced with permission. You will need to obtain permission from the third party to reuse its material.

**Links to third-party material and websites**

Please note that the provided (reading/viewing material/list/links/texts) are a suggestion only and implies no endorsement, by the New South Wales Department of Education, of any author, publisher, or book title. School principals and teachers are best placed to assess the suitability of resources that would complement the curriculum and reflect the needs and interests of their students.

If you use the links provided in this document to access a third-party's website, you acknowledge that the terms of use, including licence terms set out on the third-party's website apply to the use which may be made of the materials on that third-party website or where permitted by the Copyright Act 1968 (Cth). The department accepts no responsibility for content on third-party websites.