University of Westminster School of Computer Science (Subject to external moderation)

5DATA002W.2	**Machine Learning & Data Mining**
Module leader	Mahmoud Aldraimli, IIT ML: Nipuna Senanayake
Component	Coursework
Weighting:	60%
Qualifying mark	30%
Description	Students are expected to critically engage in effectively applying and evaluating novel data mining and machine learning techniques for two specific problem domains Classification and Regression and reflect on the knowledge of how different data mining and machine learning algorithms perform for each problem. Students are expected to methodologically analyse the output of the data mining tasks and machine learning algorithms by drawing technically appropriate and sound decisions and conclusions resulting from the application of data mining and machine learning algorithms to the given problem.
Learning Outcomes Covered in this Assignment:	This assignment contributes towards the following Learning Outcomes (LOs): • **LO1** Suitably prepare a realistic data set for data mining / machine learning and discuss issues affecting the scalability and usefulness of learning models from that set. • **LO3** Evaluate, validate and optimise learned models • **LO4** Effectively communicate models and output analysis in a variety of forms to specialist and non-specialist audiences.
Handed Out:	17th February 2025
Due Date	**1- Coursework Code Reuse:** Three in-class sessions during own timetabled seminar WK5, WK7 and WK11. The Prompt-code blocks pair matching formative submissions for each session are during own timetabled seminars. **2- Coursework Code Peer Review:** Three in-class Interim Python Notebook Code Peer-Review sessions during own timetabled seminar WK6, WK10 and WK12. The Peer Review forms formative submissions for each session are during own timetabled seminar. **3- Final Implementation Python Notebooks:** This is a summative submission of THREE Python Different Notebooks 2nd May 2025 – 1:00PM IST a) Final Python Notebook 1: Data Understanding and Preprocessing b) Final Python Notebook 2: Classification Modelling & Hyperparameters Tuning c) Final Python Notebook 3: Regression DT and Ensemble Learners **4- Analysis Report template:** This is a single summative submission 2nd May 2025 – 1:00 PM IST
Expected deliverables	Submit the following on the Blackboard: **1- Analysis Report Template in (word format).** **2- Final Implementation Python Notebooks (ipynb format):** which consist of a) Final Python Notebook 1: Data Understanding and Preprocessing b) Final Python Notebook 2: Classification Modelling & Hyperparameters Tuning c) Final Python Notebook 3: Regression DT and Ensemble Learners
Method of Submission:	Electronic submission on Blackboard via provided links close to the submission time.
Type of Feedback and Due Date:	Detailed Blackboard rubric feedback will be provided on the Blackboard for summative components. Automated solutions and feedback will be provided on the Blackboard for formative components

Refer to section 4 of the "How you study" guide for undergraduate students for a clarification of how you are assessed, penalties and late submissions, what constitutes plagiarism etc.

Penalty for Late Submission

If you submit your coursework late but within 24 hours or one working day of the specified deadline, 10 marks will be deducted from the final mark as a penalty for late submission, except for work which obtains a mark in the range of 40 – 49%, in which case the mark will be capped at the pass mark (40%). If you submit your coursework more than 24 hours or more than one working day after the specified deadline, you will be given a mark of zero for the work in question unless a claim of Mitigating Circumstances has been submitted and accepted as valid. It is recognised that, on occasion, illness or a personal crisis can mean that you fail to submit a piece of work on time. In such cases, you must inform the Campus Office in writing on a mitigating circumstances form, giving the reason for your late or non-submission. You must provide relevant documentary evidence with the form. This information will be reported to the relevant Assessment Board, which will decide whether the mark of zero shall stand. For more detailed information regarding University Assessment Regulations, please refer to the following website: http://www.westminster.ac.uk/study/current-students/resources/academicregulations

Coursework Description

The Real-world Problem Description

A) The Domain

The deployment of machine learning modelling in this coursework aims to tackle a real-world tool by developing effective early screening machine learning models for breast cancer mortality and survival prediction to help doctors enhance their treatment planning and management.

Cancer is a disease in which cells in the body grow out of control. Breast cancer is a disease in which abnormal breast cells grow out of control and form tumours. If left unchecked, the tumours can spread throughout the body and become fatal. Breast cancer cells begin inside the milk ducts and/or the milk-producing lobules of the breast. In females in the UK, breast cancer is the 2nd most common cause of cancer death, with around 11,400 deaths every year (2017-2019). In males in the UK, breast cancer is not among the 20 most common causes of cancer death, with around 85 deaths every year (2017-2019).

Stages and grades of breast cancer

The tests and scans the patient have to diagnose breast cancer give information about:

- the size of the cancer and whether it has spread (the stage)
- how abnormal the cells look under the microscope (the grade)

Knowing the stage and grade helps doctor plan the patient's treatment. The stage of a cancer tells the patient how big it is and whether it has spread. It helps the doctor decide which treatment the patient need.

There are different systems used in the UK to stage breast cancer. The most common one is the TNM system. TNM stands for Tumour, Node and Metastasis. the patient might also be told about the number staging system. There are 4 main stages in this system, from 1 to 4.

The tests the patient has also give information about the type of breast cancer they have.

The information below is an overview of the TNM staging for all types of cancer.

- **T** describes the size of the tumour (cancer)
- **N** describes whether there are any cancer cells in the nearby lymph nodes
- **M** describes whether the cancer has spread to parts of the body further away from where the cancer started

Stage 1 breast cancer means that the cancer is small and only in the breast tissue or it might be found in lymph nodes close to the breast (see Figure 1). It is an early-stage breast cancer.

The stage of cancer tells the patient how big it is and how far it has spread. It helps the doctor decide the best treatment for the patient. There are different systems used in the UK to stage breast cancer. Stage 1 is part of the number staging system. Doctors may also use the TNM staging system.

Staging for breast cancer is very complex. Many different factors are considered before doctors can confirm the patient's final stage.

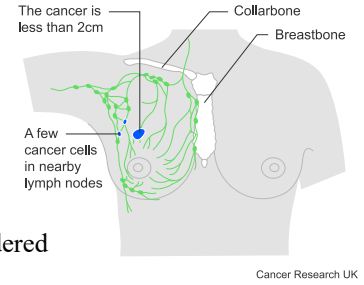
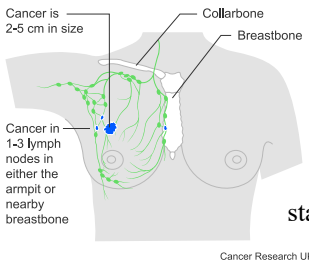
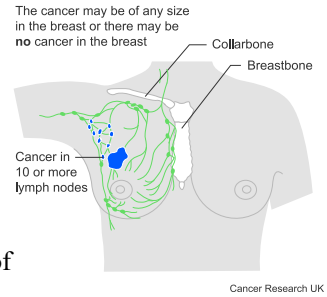


Fig.1 Illustration of stage 1 breast cancer



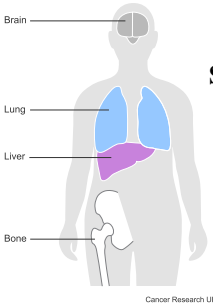
Stage 2 breast cancer means that the cancer is either in the breast or in the nearby lymph nodes or both. It is an early-stage breast cancer.
Stage 2 is part of the number staging system. Doctors may also use the TNM staging system.
Stage 2 can be divided into 2A and 2B. Opposite is a simplified description of stage 2A and 2B breast cancer (see Figure 2).

Fig.2 Illustration of stage 2 breast cancer



Stage 3 means that the cancer has spread from the breast to the lymph nodes close to the breast, to the skin of the breast or to the chest wall. It is also called locally advanced breast cancer. Stage 3 is part of the number staging system. Doctors may also use the TNM staging system.
Stage 3 can be divided into 3A, 3B and 3C. opposite is a simplified description of stage 3A, 3B and 3C breast cancer (see Figure 3).

Fig.3 Illustration of stage 3 breast cancer



Stage 4 breast cancer has spread to another part of the body (see Figure 4). It is also called advanced cancer or secondary breast cancer. The aim of treatment is to control the cancer and any symptoms. Treatment depends on a number of factors.
In stage 4 breast cancer:

- the cancer can be any size
- the lymph nodes may or may not contain cancer cells
- the cancer has spread (metastasised) to other parts of the body such as the bones, lungs, liver or brain.

Fig.4 Illustration of stage 4 breast cancer

Hormonal Treatment
Hormone therapy is a common treatment for secondary breast cancer. It can often shrink and control the cancer wherever it is in the body. It works well if the cancer cells have particular proteins called hormone receptors, estrogen receptor, progesterone receptor.
If one hormone therapy stops working so well, the doctor might suggest you try a different one.

Other Treatment
Doctor will take many different factors into account when deciding which treatment is best for the patient. These include:

- the type of cells the cancer started in
- which part of your body the cancer has spread to
- the treatment you have already had
- your general health
- whether the patient have had the menopause.
- whether the cancer is growing slowly or more quickly
- whether the cancer cells have receptors for particular cancer drugs

If your cancer doesn't have hormone receptors or has spread to the liver or lungs, the doctor might suggest Chemotherapy. Radiotherapy might be recommended if the cancer has spread to the bones or the skin near the breast. Targeted and immunotherapy drugs might be recommended for secondary breast cancer.

C) The Domain Problem
The importance of predicting mortality and short- and long-term survival of patients with cancer may improve their care. Prior predictive models either use data with limited availability or predict the outcome of only 1 type of cancer. In this case, breast cancer.

D) Your Role as A Data Scientist
You are hired as a data scientist to work alongside a team of doctors to
1- Build predictive machine-learning models for breast cancer mortality status.
2- Build predictive machine-learning models to estimate the patient survival period.

The team of doctors provided you with historical records of breast cancer patients and had their mortality status. Also, obtained the number of months they survived.
The doctors rely on your work to answer the following **two research question** on the dataset; the key objective is to create a new, predictive tool powered by a machine learning model to assist doctors in enhancing their treatment planning and cancer care. **The Research Questions are:**

a) Does machine learning have the potential to assist doctors to predict those who would survive breast cancer or not?

b) For patients who would not survive cancer, can machine learning offer a reliable estimate of their survival period?

E) Your Dataset

This dataset of breast cancer patients was obtained from the 2017 November update of the SEER Program of the NCI, which provides information on population-based cancer statistics. The dataset contains the following attributes:

Table.1 Data Dictionary

Attribute	Description
Patient ID	Unique identification for each patient
Month of Birth	A patient's month of birth
Age	A patient's month of birth in years
Sex	A patient's genomic sex
Occupation	The field of a patient's job role
T Stage	The T stage in breast cancer refers to the size of the tumour from T1, T2, T3 and T4
N Stage	Used to indicate if the breast cancer has spread to surrounding lymph nodes (N), with a higher number representing a greater number of lymph nodes impacted, from N1, N2 and N3.
6th Stage	Breast Imaging Reporting and Data System or BI-RADS
Differentiated	How the cancer cells look and are growing compared with normal cells.
Grade	Breast Cancer Grades (Nottingham Grading System)
A Stage	Breast cancer is staged based on how far it has spread. **Regional:** The cancer has spread to nearby lymph nodes or tissues. **Distant:** The cancer has spread to distant parts of the body, such as the lungs, liver, or bones
Tumour Size	Tumor size measured in millimeters
Estrogen Status	Cancer cells have estrogen hormone receptors or not.
Progesterone Status	Cancer cells have progesterone hormone receptors or not.
Regional Node Examined	Count of examined regional lymph nodes for cancer spread
Regional Node Positive	Count of cancer positive regional lymph nodes to contain metastases
Survival Months	Survival months based on date of last contact.
Mortality Status	Any patient that dies after the follow-up cut-off date is recoded to alive as of the cut-off date. If date of last contact > study cutoff date, vital status recoded = alive.

Note: *For general knowledge, further information about the collection of patients' data can be found at* https://ieee-dataport.org/open-access/seer-breast-cancer-data

The survival calculations can be found at
https://seer.cancer.gov/survivaltime/
https://seer.cancer.gov/survivaltime/SurvivalTimeCalculation.pdf

Your Data Mining Framework

As a data scientist, you are a logician, a mathematician, a technician, and an analyst, and you need doctors to understand your analyses. Doctors are usually busy individuals, and they don't have all the time in the world. One essential skill that you must adhere to is to **be concise and straight to the point**. Focus on the answers needed for each task and **provide just enough words for the answer only**. There is no need to provide lengthy descriptions of algorithms and methods unless you are asked to do.

Also, doctors are only interested in assessing your interpretation of the work, analysis and modelling results, so **you MUST NOT paste any Python code** in this report **unless specifically asked to so**. You will receive a separate link to submit your code as a Python notebook file (mandatory). **ipynb extension**. Your data mining tasks will be aligned with the popular CRISP-DM methodology phases but without the deployment phase (see Figure 5).

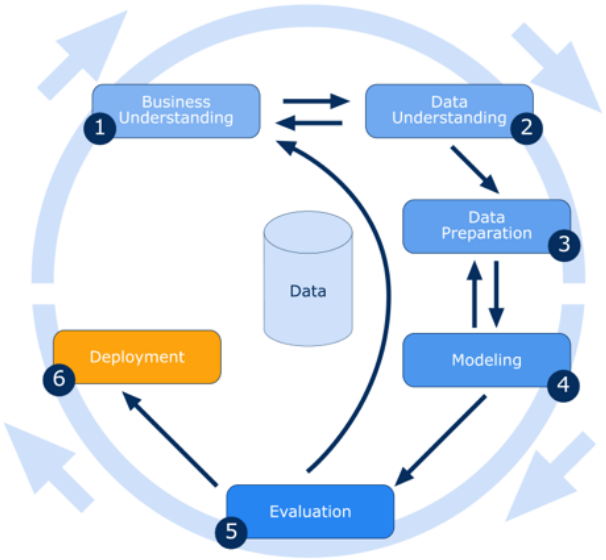


Fig.2 CRISP-DM Phases

⚠ **Important Note:** You must answer each task chronologically and use the questions as headers ⚠

PART (1) Coursework Code Reuse Sessions

A series of prompt-code pair matching to leverage and reuse code to understand and prepare cancer patients' data and to model and evaluate your predictive analysis performance

Timings: THREE 45-minute meetings will be assigned to groups during your dedicated tutorial classroom**.** These THREE code peer-review meetings will take place in Week 4, Week 8 and Week 10.

Code Reuse Session 1
Date & Time: Week 5 (Week Starting 17 Feb 2025) during your timetabled seminar session
Place: Your timetabled seminar lab

Code Reuse Session 2
Date & Time: Week 7 (Week Starting 3 March 2025) during your timetabled seminar session
Place: Your timetabled seminar lab

Code Reuse Session 3
Date & Time: Week 11 (Week Starting 31 March 2025) during your timetabled seminar session **Place:** Your timetabled seminar lab

Coursework Code Reuse Session 1 (Week 5)

Coursework Data Understanding and Preparation
Background:
Like many datasets, your coursework cancer dataset needs preparation before applying machine learning modelling. In this session, you are presented with a number of key Python code blocks, libraries, modules, functions, and methods. These are designed to help you quickly implement your coursework only if you understand their purpose; by putting them in the right sequence when applying them and modifying them, you will get the desired outputs you need to clean and prepare your coursework dataset.

How will this help me with my coursework?
Reusing these leveraged code blocks can help you clean and prepare TWO datasets, one set for classification modelling of cancer mortality status and the other for regression modelling cancer survival months. And understand the characteristics of the variables in each set. The work in this session assists you to submit your final Peer-Reviewed Python Notebook 1 out of 3 Notebooks. The final Python Notebooks submission carries 21% of the coursework mark.

Session Instructions:
1- On the blackboard, you are presented with a list of prompts that can be useful to understand and prepare your coursework data. Your job is to link each data understanding and preparation prompt to the relevant Python code block that performs such a task.
2- This is an individual open-web assessment session; you must be present in the classroom to participate. A remote attempt and/or unauthorised absence disqualifies the student, resulting in no feedback given.
3- You can use the class material published on this module's blackboard and internet search engines. However, the use of AI assistants is prohibited. Communication with others except tutors via any means during the session shall disqualify the student instantly, resulting in no feedback being given. You can only consult your tutors.
4- This session is a booster to your coursework. It is timed to 45 minutes with an automatic submission feature.
5- Full automated feedback will be provided by the end of the week. After receiving the feedback, you should be able to leverage and reuse the code blocks from this session at best to build your **Python Notebook 1** code for coursework data understanding and preparation.

Code Leverage and Reuse Pair-matching Activity:
You are presented with a list of tasks that can be useful to understand and prepare your coursework data. Your job is to link (match) each data understanding and preparation task to the correct Python code block that performs such a task.

Coursework Code Reuse Session 2 (Week 7)

Mortality Status Classification Modelling & Hyperparameters Tuning
Background:
Now that you have prepared your cancer datasets, it is time to model your data with machine learning classification and regression algorithms. In this session, you are presented with a number of Python code blocks, libraries, modules, functions, and methods. These are designed to help you quickly implement your coursework only if you understand their purpose; by putting them in the right sequence when applying them and modifying them, you will get the desired outputs you need to build meaningful and useful predictive Naïve Bayes, Logistic Regression K-Nearest Neighbours classification models for cancer mortality status prediction. You are also required to optimise the hyperparameters of your best model.

How will this help me with my coursework?
When reusing these leveraged code blocks, you are expected to build Logistic Regression, Naïve Bayes and K-Nearest Neighbour models for cancer mortality status prediction and optimise their hyperparameters.
The work in this session assists you to submit your final Peer-Reviewed Python Notebook 2 out of 3 Notebooks. The final Python Notebooks submission carries 21% of the coursework mark.

Session Instructions:
1- On the blackboard, you are presented with a list of prompts that can be useful to model your coursework data to predict mortality status. Your job is to link each prompt to the relevant Python code block that performs such a task.
2- This is an individual open-web assessment session; you must be present in the classroom to participate. A remote attempt and/or unauthorised absence disqualifies the student, resulting in no feedback given.
3- You can use the class material published on this module's blackboard and internet search engines. However, the use of AI assistants is prohibited. Communication with others except tutors via any means during the session shall disqualify the student instantly, resulting in no feedback being given. You can only consult your tutors.

4- This session is a booster to your coursework. It is timed to 45 minutes with an automatic submission feature.
5- Full automated feedback will be provided by the end of the week. After receiving the feedback, you should be able to leverage and reuse the code blocks from this session at best to build your **Python Notebook 2 code** for coursework classification modelling and hyperparameter optimisation.

Code Leverage and Reuse Pair-matching Activity:
You are presented with a list of prompts that can be useful to build and assess THREE classification models and optimise their hyperparameters. Your job is to link (match) each activity prompt to the correct Python code block that performs such a task.

Coursework Reuse Session 3 (Week 11)

Building Regression DT and an Ensemble Learner
Background:
Now that you have prepared your datasets, it is time to model cancer patients' mortality status with classification ensemble learning and create regression models to predict their survival months. In this session, you are presented with a number of Python code blocks, libraries, modules, methods and functions. These are designed to help you build a machine-learning ensemble voting classifier for mortality status predictions and Decision Tree regression models for survival months prediction in your coursework. You need to understand the purpose of the code blocks, use them in the right sequence of application, and know how to modify (reuse) them to get the desired output you need to build meaningful and useful predictive models for your coursework.

How will this help me with my coursework?
When reusing the leveraged code blocks from this session, you are expected to build a probability-based voting ensemble classifier for cancer mortality status prediction by combining ONLY TWO out of the three base learners, Logistic Regression, Naïve Bayes and K-Nearest Neighbours, which are Logistic Regression, Naïve Bayes and K-Nearest Neighbour. Also, reusing the rest of the leveraged code blocks can help you build decision tree regression models to predict cancer survival months. The work in this session assists you to submit your final Peer-Reviewed Python Notebook 3 out of 3 Notebooks. The final Python Notebooks submission carries 21% of the coursework mark.

Session Instructions:
1- On the blackboard, you are presented with a list of prompts that can be useful to model your coursework data to predict mortality status. Your job is to link each prompt to the relevant Python code block that performs such a task.
2- This is an individual open-web assessment session; you must be present in the classroom to participate. A remote attempt and/or unauthorised absence disqualifies the student, resulting in no feedback given.
3- You can use the class material published on this module's blackboard and internet search engines. However, the use of AI assistants is prohibited. Communication with others except tutors via any means during the session shall disqualify the student instantly, resulting in no feedback being given. You can only consult your tutors.
4- This session is a booster to your coursework. It is timed to 45 minutes with an automatic submission feature.
5- Full automated feedback will be provided by the end of the week. After receiving the feedback, you should be able to leverage and reuse the code blocks from this session at best to build your **Python Notebook 3 code** for coursework ensemble learning classification modelling and Decision Tree Regression modelling.

Code Leverage and Reuse Pair-matching Activity:
You are presented with a list of prompts that can be useful for building Ensemble Classifiers and Decision tree regression models. Your job is to link (match) each activity prompt to the correct Python code block that performs such a task.

PART (2) Coursework Code Peer Review Meetings

A series of three code peer-review meetings will be held to test and get peer feedback on your interim Python notebook, which you constructed by reusing code blocks from your Coursework Reuse sessions.

Timings: THREE 45-minute meetings will be assigned to groups during your dedicated tutorial classroom**.** These THREE code peer-review meetings will take place in Week 4, Week 8 and Week 10.

Code Peer-Review Meeting 1
Date & Time: Week 6 (Week Starting 24 Feb 2025) during your timetabled seminar session **Place:** Your timetabled seminar lab

Code Peer-Review Meeting 2
Date & Time: Week 10 (Week Starting 24 March 2025) during your timetabled seminar session
Place: Your timetabled seminar lab

Code Peer-Review Meeting 3
Date & Time: Week 12 (Week Starting 7 April 2025) during your timetabled seminar session **Place:** Your timetabled seminar lab

Code Peer-Review Meeting 1 (Week 6)

Purpose and Preparation: With some help from the leveraged and reused code blocks from the COURSEWORK CODE REUSE pair matching session 1**, before arriving to this session,** you should have done your best to clean and prepare your coursework dataset in your **Python Notebook 1**. Once prepared, you should create two datasets, one for cancer mortality status classification modelling and the other for regression modelling of cancer survival months. It is expected that your interim Python Notebook 1 is a draft; it can have errors, and this is a session to help you correct it.

In the meeting: In collaboration with your group members, you should demo your code to your group; you are required to get them to test your interim Python Notebook 1 code implementation and gather their feedback on any improvement you should make to fix any bugs or to get your code working.

Filling a Blackboard Meeting Form: Each group is expected to fill in a short form. The form contains the following sections:
Team Members Names
Meeting Date/Start Time/ Finish Time/Venue
Attendees and Absentees
List of Tests/Checks performed on each interim Python Notebook (Minimum 3 Tests per notebook)
Results of tests (Pass/Fail) with evidence.
Summary points of improvement and feedback on each notebook.

Final Python Notebook 1 Specifications (for final submission):
1- Your final Python Notebook 1 must contain both the Python code and results for data understanding preparation tasks ONLY.
2- The results in your **final Analysis report** must match the results in your Python Notebook 1
3- Each line of code must have a comment above it explaining what the code line does.
4- Each Code Cell must have a Text Cell above it that mentions the exact code block which was leveraged and reused from the **Code Reuse Session 1 and/or from your Seminar Sessions (excluding any Auto-ML); ensure you note the name of the session correctly.**
5- The coursework is made to allow you to apply the programming concepts learnt in your module, and it is very unlikely you will need external code. If you need external code, you must demo your need for it and understand it and get it approved by your Tutor in the peer review session. Ensure you note the approval in the text cell with a full reference, too.
6- You must have your name in the notebook as an author, and the name of the person who conducted the Code Peer Review.

7- The use of Generative AI/any AI assistant tool is prohibited for this coursework and can result in zero grade given. For university guidance and regulation regarding the use of Guidance for students on using Generative AI please visit:
https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/guidance-for-students-on-using-generative-ai

Your Responsibility: During the Code Peer-Review Meeting 1, you should have gathered as much feedback from your group as possible to improve and fix your code to ensure that you have two clean and prepared datasets and meeting the Python Notebook 1 Specifications before reaching the Code Peer-Review Meeting 2 (Week 8)

Code Peer-Review Meeting 2 (Week 10)

Purpose and Preparation: With some help from the leveraged code blocks from the COURSEWORK CODE REUSE pair matching session 2**, before arriving to this session,** you should have done your best to build, evaluate and optimise the hyperparameters for THREE classifiers (Naïve Bayes, Logistic Regression and K-Nearest Neighbours) to predict cancer mortality status with your prepared dataset in your interim **Python Notebook 2**. It is expected that your interim Python Notebook 2 is a draft; it can have errors, and that is ok, and this is a session to help you correct it.

In the meeting: In collaboration with your group members, you should demo your code to your group; you are required to get them to test your interim Python Notebook 2 code implementation and gather their feedback on any improvement you should make to fix any bugs or to get your code working.

Filling a Blackboard Meeting Form: Each group is expected to fill in a short form. The form contains the following sections:
Team Members Names
Meeting Date/Start Time/ Finish Time/Venue
Attendees and Absentees
List of Tests/Checks performed on each interim Python Notebook (Min 3 Tests per notebook)
Results of tests (Pass/Fail) with evidence.
Summary points of improvement and feedback on each notebook.

Final Python Notebook 2 Specifications (for final submission):
1- Your final Python Notebook 2 must contain both the Python code and results for cancer mortality status classification modelling tasks and their hyperparameter optimisation ONLY.
2- The results in your **Final Analysis Report** must match the results in your Python Notebook 2
3- Each line of code must have a comment above it explaining what the code line does.
4- Each Code Cell must have a Text Cell above it that mentions the exact code block which was leveraged and reused from the **Code Reuse Session 2 and/or from your Seminar Sessions (excluding any Auto-ML); ensure you note the name of the session correctly.**
5- The coursework is made to allow you to apply the programming concepts learnt in your module, and it is very unlikely you will need external code. If you need external code, you must demo your need for it and understand it and get it approved by your Tutor in the peer review session. Ensure you note the approval in the text cell with a full reference, too.
6- You must have your name in the notebook as an author, and the name of the person who conducted the Code Peer Review.
7- the use of Generative AI/any AI assistant tool is prohibited for this coursework and can result in zero grade given. For university guidance and regulation regarding the use of Guidance for students on using Generative AI please visit:
https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/guidance-for-students-on-using-generative-ai

Your Responsibility: During the Code Peer-Review Meeting 2, you should have gathered as much feedback from your group as possible to improve and fix your code to ensure that you have two clean and prepared datasets and meeting the Python Notebook 2 Specifications before reaching the Code Peer-Review Meeting 3 (Week 10)

Code Peer-Review Meeting 3 (Week 12)

Purpose and Preparation: With some help from the leveraged code blocks from the COURSEWORK CODE REUSE pair matching session 3**,** at this stage, **before arriving to this session** you should have done your best to combine TWO out of THREE base learners (Naïve Bayes, Logistic Regression and K-Nearest Neighbours) into a voting ensemble learner and evaluate it in predicting cancer mortality status with your prepared dataset in your interim **Python notebook 3**. Also, in the same notebook, you need to create and evaluate fully-grown and pruned regression decision trees to predict mortality cancer survival months. It is expected that your interim **Python Notebook 3** is a draft; it may have errors, and that is ok, this is a session to help you correct it.

In the meeting: In collaboration with your group members, you should demo your code to your group; you are required to get them to test your interim Python Notebook 3 code implementation and gather their feedback on any improvement you should make to fix any bugs or to get your code working.

Filling a Blackboard Meeting Form: Each group is expected to fill in a short form. The form contains the following sections:
Team Members Names
Meeting Date/Start Time/ Finish Time/Venue
Attendees and Absentees
List of Tests/Checks performed on each interim Python Notebook (Minimum 3 Tests per notebook)
Results of tests (Pass/Fail) with evidence.
Summary points of improvement and feedback on each notebook.

Final Python Notebook 3 Specifications (for final submission):
1- Your final Python Notebook 3 must contain both the Python code and results for cancer mortality status ensemble classifier with its TWO base learners, and mortality month regression modelling with Decision Trees ONLY.
2- The relevant results in your **Final Analysis Report** must match the results in your Python Notebook 3
3- Each line of code must have a comment above it explaining what the code line does.
4- Each Code Cell must have a Text Cell above it that mentions the exact code block which was leveraged and reused from the **Code Reuse Session 3 and/or from your Seminar Sessions (excluding any Auto-ML); ensure you note the name of the session correctly.**
5- The coursework is made to allow you to apply the programming concepts learnt in your module, and it is very unlikely you will need external code. If you need external code, you must demo your need for it and understand it and get it approved by your Tutor in the peer review session. Ensure you note the approval in the text cell with a full reference, too.
6- You must have your name in the notebook as an author, and the name of the person who conducted the Code Peer Review.
7- the use of Generative AI/any AI assistant tool is prohibited for this coursework and can result in zero grade given. For university guidance and regulation regarding the use of Guidance for students on using Generative AI please visit:
https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/guidance-for-students-on-using-generative-ai

Your Responsibility: During the Code Peer-Review Meeting 3, you should have gathered as much feedback from your group as possible to improve and fix your code to ensure that you have two clean and prepared datasets and meeting the Python Notebook 3 Specifications before reaching the final submission deadline (Week 12)

PART (3) Final Python Notebooks of Both Case Studies, Predicting Cancer Patients Mortality Status and Survival Months [21 Marks]

Over the course of this module, you developed **THREE Final Python Notebooks**. Thus, we expect that you would have by now perfected your work throughout this coursework.

Therefore, you must submit the following:

1- A fully functional Final Implementation for Python Notebook 1 in **.ipynb** format that contains all the functional Python code and the final associated results you included in your Analysis Report along with the **dataset used to develop this notebook**. [7 Marks]

2- A fully functional Final Implementation for Python Notebook 2 in **.ipynb** format that contains all the functional Python code and the final associated results you included in your Analysis Report **along with the dataset used to develop this notebook.** [7 Marks]

3- A fully functional Final Implementation for Python Notebook 3 in **.ipynb** format that contains all the functional Python code and the final associated results you included in your Analysis Report along with the dataset used to develop this notebook. [7 Marks]

Your Final Python Notebooks are graded based on meeting the following specifications:

1- Your final Python Notebooks must contain both the Python code and results for the following:
Final Python Notebook 1: Data Understanding, Cleaning and Preparation
Final Python Notebook 2: Mortality Status Classifiers, their performances and Hyperparameters optimisations.
Final Python Notebook 3: Mortality Status ensemble classifier with its base learners' performances and Survival Months regression Decision Trees, with their graphical representation and performances.

2- The relevant results in your Analysis Report must match the results in your Submitted Final Python Notebooks

3- Each line of code must have a comment above it explaining what the code line does.

4- Each Code Cell must have a Text Cell above it that mentions the exact code block which was leveraged and reused from the **Code Reuse Sessions and/or from your Seminar Sessions;** also, ensure you reference the name of the session correctly and reference it in full including any page numbers.

5- The coursework is made to allow you to apply the programming concepts learnt in your module, and it is very unlikely you will need other external code different to your tutorials. If you need other external code, In the text cell above it, you must note the tutor's name who approved its use, the date of approval, the name of your peer reviewer who confirmed your understanding of the code and a full reference in Harvard style, too.

6- Each of your **Final Python Notebook**s must have your name in the notebook as an author, the name of the student (classmate) who conducted the Code Peer Review and the date of the review.

7- Focus on reusing the libraries within the scope of your teaching material, **excluding PyCaret library**. You MUST NOT use AutoML libraries/tools such as PyCaret or others, and the use of Generative AI/any AI assistant tool is prohibited to produce any answers for this coursework and can result in zero grade given as well as the use of tools outside the scope of this coursework. For university guidance and regulation regarding the use of Guidance for students on using Generative AI please visit:
https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/guidance-for-students-on-using-generative-ai

==========

Final Python Notebooks – Critical Notes & Penalties:

1- Submitting any Final Python Notebook in any format other than .ipynb and/or the absence of results (i.e. submitting the code only) in any of the submitted **Final Python Notebooks** will result in a 0% grade given for the notebook and the associated analysis part of the report.

2- Do not share/transfer any of your Python Notebooks to any of your classmates. Do not share/transfer any of your Python Notebooks to peer reviewers. You are only allowed to demo your code to your peer reviewer and get feedback on your code in the peer code review sessions to fix issues in your code or improve it.

3- You can get face-to-face feedback from your tutor on your interim code either during the seminar or via face-to-face appointments or from the module leader via an open-door policy.

4- To verify the authenticity of the work, the owner of a submission may be invited for a video-recorded 20-minute viva. So be prepared to explain your results/answers and code should you have been invited for one. Failure to attend without authorisation on reasonable grounds can result in a zero-mark given to the components in question.

PART (4) Analyses Report of Two Case Studies, Predicting Cancer Patients Mortality Status and Survival Months [Total 79 Marks]

Case Study (A): Predicting Cancer Patients Mortality Status. [Total 43 Marks]

Research Question: Does machine learning have the potential to assist doctors in predicting those who will survive breast cancer or not?

Purpose: The health professionals will give you requirements and data mining tasks in this part to help them answer their first research question. You are expected to perform the tasks as per their requirements, analyse, interpret, criticise, and report your classification of machine learning and data mining decisions, findings, and results to assist doctors in answering this case study research question.

Case Study (A) Analyses Report for Predicting Mortality Status Tasks

Task (1) – Domain Understanding: Classification **[Total 3 Marks]**

The doctors decided that classification modelling is required. Indicate in the table below for each of the listed variables in your data which ones you should RETAIN and can be included in the classification modelling of Breast Cancer Mortality (Alive vs. Dead) and the variables you should DROP (REMOVE). Justify your decision logically and/or by research (include in-text citation)

Variable Name	RETAIN or DROP	Brief justification for retention or dropping
Patient ID		
Month of Birth		
Age		
Sex		
Occupation		
T Stage		
N Stage		
6th Stage		
Differentiated		
Grade		
A Stage		
Tumour Size		
Estrogen Status		
Progesterone Status		
Regional Node Examined		
Regional Node Positive		
Survival Months		
Mortality Status		

Task (2) – Exploring and Understanding Your Dataset **[Total 2 Marks]**

With the aid of your **Final Python Notebook 1**, for your RETAINED input variables and your class "Target" variable, produce basic descriptive stats and variable scale type. Plot the distribution of your target variable. (Paste screenshots of code OUTPUTS ONLY for evidence of these elements).

Task (3) – Data Preparation: Cleaning and Transforming your data **[Total 8 Marks]**

a) With the aid of your **Final Python Notebook 1**, when you first explored your retained variables in the cancer dataset, you may have found some issues. Report any issues you found in your retained dataset variables. Based on the issues you found in your data, suggest a suitable possible method to fix each of these issues and provide your justification for using your suggested fix method. Use the table below to organise your findings and analysis, and add more rows if needed: [4 Marks]

Variable Name	Issue found	Proposed fix	Justification for used fix method
⋮	⋮	⋮	⋮

b) With the aid of Python packages and your **Final Python Notebook 1**, implement your suggested fixes of issues in the previous Task (3.a). Show evidence (before and after) of implementing your suggested fix to the problems you identified for your dataset in Task (3.a). To show your evidence, paste screenshots of your relevant code OUTPUTS ONLY (Do not paste the code). Indicate and annotate the issue and the fix in each of your provided evidence screenshots. [4 Marks]

Task (4) – Classification Modelling of Cancer Patients Mortality Status **[Total 6 Marks]**

a) In your **Final Python Notebook 2**, you built THREE different models to predict cancer mortality status: Logistic Regression (LR), K Nearest Neighbour (KNN) and Naïve Bayes (NB). These algorithms are a mix of parametric and non-parametric algorithms. List down the type of each algorithm (parametric vs non-parametric), name any learnable parameters, and list any strategic hyperparameters for each algorithm which you want to consider tuning. Organise your answer in a table as before. See below: [3 Marks]

Algorithm Name	Algorithm Type	Learnable Parameters	Some Strategic Hyperparameters
NB			
LR			
KNN (N=?)			

b) With the aid of your **Final Python Notebook 2,** use the training–test split approach with your retained applicable input features only and the target output feature to build your predictive classification models. [3 Marks]

i. Screenshot the list of all feature names used for building your classification models and the corresponding data shape function output. (Paste screenshots of the relevant code output only; do not paste the Python code).

ii. In less than 150 words, research and justify (defend) your choice of the training-test split ratio and provide an in-text citation.

iii. Provide as evidence the code block line and code output from your **Final Python Notebook 2** that ensures two conditions: one, all your models were tested on the same test instances (patients) in your dataset; second, the labels ratio of Mortality Status "Alive" to "Dead" is the same in the training and test subsets. State the training-test split function parameters in your code line that are responsible for meeting both conditions. In less than 150 words, research and justify (defend) your decision to implement both conditions in your **Python Notebook 2** notebook with in-text citations where possible.

Task (5) – Evaluating your Cancer Mortality Status Classification Models **[Total 24 Marks]**

Your healthcare professionals provided the following success criteria to guide you when evaluating and selecting your best model: *"When evaluating your cancer patients' mortality status classification mode's performance, which addresses your research question. The best model is expected to have some misclassifications. Thus, the model should aim to better discriminate between "Dead" and "Alive" cancer patients"*

a) With the aid of **Final Python Notebook 2**, paste the test confusion matrix, the classification report and the AUC-ROC curve graphs for each of your models (Logistic Regression LR, Naive Bayes NB and K-Nearest Neighbours KNN) as screenshots from the output of your Python code. [3 marks]

b) Five different classification evaluation metrics are calculated in your **Final Python Notebook 2**. State which evaluation metric/metrics to "USE or "DO NOT USE" to closely interpret the above success criteria. For justification, explain how closely your choice of "USE" or "DO NOT USE" for a metric interprets the given success criteria. With the aid of your **Final Python Notebook 2**, document all the TEST SCORES for each built model in the table below. [7 marks]

Metrics	USE or DO NOT USE	Justification for choosing "USE" or "DO NOT USE" in relation to the success criteria	Model Name	Test Score
Accuracy			*NB*	
			LR	
			KNN (K=?)	
Recall			*NB*	
			LR	
			KNN (K=?)	
Precision			*NB*	
			LR	
			KNN (K=?)	
F-Score			*NB*	
			LR	
			KNN (K=?)	
AUC-ROC			*NB*	
			LR	
			KNN (K=?)	

c) Suggest a single best mortality status classification model based on the 'USED' performance metrics scores you identified in Task (5.b). In less than 100 words, briefly describe how well your best model satisfies the needs of your healthcare professionals. [2 marks]

d) To enhance your selected best model/s performance, tune some of its possible hyperparameters, which you indicated in Task (4.a) for that specific algorithm. With the aid of **Final Python Notebook 2**, Re-train and test the best algorithm again with GridSearchCV [5 marks]

i. With the aid of your **Final Python Notebook 2,** paste into this report the line of code which shows evidence of specifying a parameters grid and applying the GridSearchCV function to rebuild your selected best model. Then, document the estimated best hyperparameters for the optimised model.

ii. With the aid of your **Final Python Notebook 2,** paste into this report the test confusion matrix for your best model before and after hyperparameter tuning into this report. Also, document the new score/s of the "USED" performance metric/s of your choice to interpret the success criteria indicated in Task (5.b) before and after tuning. Comment on whether the tuning of hyperparameters of your best model improved its positive predictive ability in line with the success criteria.

e) Based on your selected best model, criticise your best-performing model, and state any limitations you may have identified and any ethical issues your model may raise if used for predicting breast cancer mortality status. [2 Marks]

f) With the aid of your **Final Python Notebooks 3,** combine only TWO out of the THREE base learners (NB, LR, KNN) that you already built into a probability-based voting ensemble classifier. [5 marks]

i. From your **Final Python Notebooks 3,** paste the Python code block that you used to import, declare your base learners, and fit your ensemble learner.

ii. In this analysis report, paste the test confusion matrices, AUC-ROC Curves and the classification reports for each of the TWO base learners you chose to combine, as well as the test confusion matrix and classification report for the voting Ensemble Learner. Use these screenshots to justify (defend) your choice of the TWO base learners which you used as base learners for your Ensemble learner.

ii. Comment on any improvement in classification performance as a result of building an Ensemble Learner compared to the individual TWO base learners. Decide whether to recommend your ensemble learner for mortality prediction or one of the TWO base learners; justify your recommendation.

==========END OF CASE STUDY (A)==========

Case Study (B): Predicting Cancer Patients Survival Months. [Total 36 Marks]

Research Question: Does machine learning have the potential to assist doctors in predicting survival months for patients who are not going to survive breast cancer?

Purpose: The health professionals will give you requirements and data mining tasks in this part to help them answer their second research question. You are expected to perform the tasks as per their requirements, analyse, interpret, criticise, and report your classification of machine learning and data mining decisions, findings, and results to assist doctors in answering this case study research question.

Case Study (B) Analyses Report for Predicting Survival Months Tasks

Task (1) – Domain Understanding and Designing Your Regression Experiments [Total 2 Marks]

The healthcare professionals decided that regression modelling is required to predict survival months for those who would not survive breast cancer. With the aid of your **Final Python Notebook 1 code outputs,** paste in this analysis report, the Python code output, which shows the dimensions and the list of the features' names of your RETAINED data subset to use for this regression case study.

Task (2) – Modelling: Build Predictive Regression Models [Total 12 Marks]

a) Your healthcare team decided to use a decision tree regression (DT) algorithm to model the survival months. In less than 150 words, explain some added benefits of using a DT regressor to this healthcare prediction problem. [2 marks]

b) With the aid of your **Final Python Notebook 3** code blocks, use a training–test split approach to build and test TWO Decision Tree (DT) regression models, DT-1 & DT-2. [6 marks]

i. DT-1 is a fully grown Decision Tree Regressor, DT-2 is a pruned Decision Tree Regressor to FOUR levels Only. Insert in this analysis report the Python code blocks that you used to import, declare, and fit each DT regressor, DT-1 and DT-2.

ii. Explain clearly, in less than 200 words, from your inserted code block in (i), the type of pruning you used for DT-2. Explain some of the benefits and disadvantages of the pruning method you used in the context of (relation to) your cancer patients' regression modelling.

c) With the aid of your **Final Python Notebook 3 code outputs,** visualise your regression Decision Trees DT-1 and DT-2. Paste in this analysis report a high-resolution graphical representation of DT-1 and DT-2. [4 Marks]

Task (3) – Evaluating your Cancer Survival Months DT Regression Models [Total 22 Marks]

Your healthcare professionals provided the following success criteria to guide you when evaluating your DT-1 and DT-2 models.

"When evaluating both models' performances, which addresses your research question (b), the model is expected to make some errors in estimating the survival months. However, since the survival months calculations are made to try to save the lives of those who may not survive cancer by prioritising their treatment plans, it is important that the selected model signifies even small errors in survival months predictions."

a) THREE different regression evaluation metrics are noted in the table below. State which evaluation metric/metrics to USE or NOT USE to closely interpret and satisfy the above success criteria. Justify (Defend) your choice of USE or DO NOT USE for each metric. With the aid of **Final Python Notebook 3 code outputs**, document each metric's TEST SCORES for each built model in the table below. [8 marks]

Metrics	**USE or DO NOT USE**	**Justification in relation to the success criteria**	**Model Name**	**Test Score**
MSE			DT-1 (Fully Grown DT)	
			DT-2 (Pruned DT)	
MAE			DT-1 (Fully Grown DT)	
			DT-2 (Pruned DT)	
R-Square			DT-1 (Fully Grown DT)	
			DT-2 (Pruned DT)	

b) Suggest a **single best regression model** (DT-1 or DT-2) based on your 'USED' performance metric/s scores, which you defended in Task (3b). Explain how your suggested model fulfils the success criteria. [4 marks]

c) Describe to your healthcare team any concerns you have about your selected performance metric/s that you used to select your best decision tree model, which satisfies the success criteria. [4 marks]

Task (4) – Interpreting Cancer Survival Months Decision Tree Outcomes [Total 6 Marks]

a) Patient B002565 breast cancer was deemed terminal. With the aid of your **Final Python Notebook 3 outputs**, use your high-resolution graphical representation of your selected best DT regression model from Task (3.b) to predict the survival months for breast cancer patient B002565; you must write down which regression Decision Tree you used (DT-1 or DT-2) to estimate the survival months, you must write down the path of rules (decision steps/tests) you used from your selected best DT to explain to patient B002565 how you estimated their predicted survival months. Patient B002565 attributes' values are in the following table: [6 marks]

Variable Name	Value
Patient ID	**B002565**
Month of Birth	**July**
Age	**29 Years old**
Sex	**Female**
Occupation Code	**15**
T Stage	**T3**
N Stage	**N1**
6th Stage	**IIIC**
Differentiated	**Moderately differentiated**
Grade	**2**
A Stage	**Regional**
Tumour Size	**41**
Estrogen Status	**Negative**
Progesterone Status	**Positive**
Regional Node Examined	**5**
Regional Node Positive	**1**

==========END OF CASE STUDY (B)==========

Analyses Report – Critical Notes & Penalties

1- Submitting a final analysis report with answers/results that do not match those in the associated submitted **Final Python Notebooks** can receive a zero grade for the relevant task with discrepancy.

2- The Final Analyses Report is limited to a maximum of 17 pages, including referencing figures and tables. You must stick to the word count as specified per question. The minimum font is Arial size 10 single-spaced. A minimum of 1-inch page margins. Exceeding the 17-page limit and/or the word limits and not complying with the specified minimum font size may result in an automatic 20% penalty deduction of your Final Analyses Report's mark.

3- Use the question numbers as headers; answer the tasks in each case study in the correct chronological order; attempt all tasks. You do not need to copy the full question as a header, but that is up to you. Your answers must map to each question's number and task in the correct order. Otherwise, this may lead to a significant delay in marking your work and the potential of missing out on marks lost between the lines.

4- Do not share/transfer your analysis report answers and/or results with any other student. Students who shared their analysis/answers/results with others can be investigated for collusion by an Academic Misconduct Panel and awarded a zero mark.

5- To verify the authenticity of your work, the owner of a submission may be invited for a video-recorded 20-minute viva. So be prepared to explain your results/answers and code should you have been invited for one. Failure to attend without authorisation on reasonable grounds can result in a zero-mark given to the component in question.

6- The use of Generative AI/any AI assistant tool is prohibited to produce any answers for this coursework and can result in zero grade given. For university guidance and regulation regarding the use of Guidance for students on using Generative AI please visit:
https://www.westminster.ac.uk/current-students/studies/study-skills-and-training/guidance-for-students-on-using-generative-ai

END OF ALL COURSEWORK PARTS