1. **Introduction**

Heart failure (HF) is a health condition that occurs when the heart cannot pump enough blood to meet the needs of the body. The healthcare industry relies on machine learning model to predict heart failure cases. You have been asked by SHU clinic team to explore different programming and analytics techniques to analyseand evaluate heart failure model performance to make informed decisions on patient’s survival. For this purpose, you will make use of programming concepts such as use of custom module, function definitions, file processing and exception handling, use of scientific computing, data analysis, data visualization and machine learning libraries (such as numpy, pandas, matplotlib, and scikitlearn) in the implementation to predict patients’ survival.

# Dataset

The dataset for this assignment can be downloaded from the PCP Blackboard module site or the link below. Please study the dataset in terms of size, data type and variables.

* + Health failure clinical dataset (available on Blackboard & [UCI repository](https://archive.ics.uci.edu/ml/datasets/Heart%2Bfailure%2Bclinical%2Brecords)).

# Assignment Key Tasks

The following tasks are to be performed in this assignment:

1. Exploratory Data Analysis

You are required to write codes to check if there’s any data missing (if yes, apply an appropriate cleaning technique). Is there any other data preprocessing you need to conduct? If yes, write codes for this purpose. In addition, the module should have functions/methods that perform descriptive statistical analysis of the dataset (such as mean, median, standard deviation, variance, minimum, maximum, skewness and kurtosis): choose a range of the variables of your interest, find their frequencies and dependencies through bar plots, grouped bar plots, pie- charts, etc. Draw conclusions.

1. Classification I

**Split the dataset on training and testing sets.** You are required to fit machine learning algorithms namely, Naïve Bayes, Logistic Regression, Support Vector Machine, Random Forest classifier, K-Nearest Neighbour and Multi-Layer Perceptron Neural Networks. Evaluate your models using test dataset and provide the confusion matrix for all models. Report and

compare performance of the models in terms of accuracy, precision, recall and F1-Score. Draw conclusions and provide recommendations.

1. Classification II

Investigate **class imbalance problem** by producing the plot of the target variable class distribution. If there is presence of class imbalance problem, use at least **2** techniques to balance the class distribution (Algorithm or Sampling technique). This means you will have a balanced dataset. Using the **balanced dataset,** you are required to build classification models using machine learning algorithms namely, Naïve Bayes, Logistic Regression, Support Vector Machine, Random Forest classifier, K-Nearest Neighbour and Multi-Layer Perceptron Neural Networks. Evaluate your models using test dataset and provide the confusion matrix for all models. Report and compare performance of the models in terms of accuracy, precision, recall and F1-Score. Compare your result with the result of II above. Draw conclusions and provide recommendations. Please provide justification for chosen methods.

1. Feature Selection

It is advisable to use limited features for prediction so far it produces good performances. To achieve this, you are required to investigate the significance of the features for selection purpose. Using Mann-Whitney test & Chi-Square test, you are to compare the distribution of each features between the two groups of the target class (Survived vs Dead). You can then rank the features in the most significant order (using P = 0.05). Secondly, you are to produce the plot of the feature importance graph (from Random Forest Classifier in III above). Use the plot and the statistical tests conducted to decide on features to be selected.

1. Classification III

Using the features selected in IV above, you are required to build classification models using machine learning algorithms namely, Naïve Bayes, Logistic Regression, Support Vector Machine, Random Forest classifier, K-Nearest Neighbour and Multi-Layer Perceptron Neural Networks. Evaluate your models and provide the confusion matrix for all models. Report and compare performance of the models in terms of accuracy, precision, recall and F1 -Score. Compare your result with the result of II & III above.

1. Conclusion

Draw conclusion on your best performed model and provide justification.

# Requirement

This assignment is an **individual piece of work**, and your submission must be in the form of modules (.py files) or Jupyter Notebook file. Tutors should be able to open and run your modules on a standard campus computer.

1. You are required to submit at least two python files. One of that should implement **a custom module** (.py files)**.** An example can be custom module (saved as EDA.py) which consists of functions defined to perform the descriptive statistics. Seconded by main.ipynb file which consists of EDA module imported (to perform exploratory data analysis), regression and classification solutions.
2. You are required to submit a report (Ms word or PDF file). The report should provide justifications for your analysis of the solution, design decisions and pseudocodes. It should explain the relationships between the modules. A good report should be based on evidence with critical analysis of the implemented system. In addition, your report should include a reflection section of your experience while executing this project. The reflection should detail what went well or not and lesson learnt. What would you do differently if you have another chance to execute this project again?
3. You are required to submit a video recording demonstrating the programming and machine learning concepts you have adopted. **Kindly explain the concepts in detail.**
4. Any evidence of collusion/plagiarism will be penalised if appropriate! If there is some doubt about the authenticity of a particular piece of work, then the person submitting it will be expected to defend such work, including reasons for the programming decisions taken. You must document with references any use of libraries or existing code in your report.
5. Appropriate use of variable names for clearer understanding is desirable
6. Adequate commenting of your codes for easier understanding during grading is also desirable.

# Submission Process

* + Your assignment should be submitted electronically through the module's Blackboard site as a single ZIP file that contains **all your source codes, video demonstration and report**.
  + In addition, a copy of your report should be submitted on Turnitin.
  + If your video is longer than 15 minutes, we will stop watching it during grading at exactly 15th minute.
  + Please check your upload to ensure you have submitted the correct files successfully as any issues will not be considered after the deadline.
  + Kindly provide an explanation in your report on how to execute your application.
  + You must also check your report’s similarity score using Turnitin on the Blackboard before final submission. Please do not submit any report with similarity score higher than 20%. Otherwise, you will be penalized for plagiarism or collusion.
  + Your assignment must be submitted on **…**
  + Note that late submission will attract penalty. The penalty is capping of your mark to 50%.

# Suggested Structure of the Report:

1. Cover page with your name, student number and title
2. Introduction which contains a short description of the context & method.
3. Answers on the stated questions should be **well discussed.** For example, approach, and result/findings.
4. All evaluative results should be presented in a table (screenshots of Python result in Appendix)
5. All plots, figures and graphs must be numbered and clearly labelled.
6. Provide conclusion and recommendations.
7. Please include the reference, especially, to all libraries used in the programming. You are required to use the APA for referencing.

# Assessment Criteria

This assignment will be assessed through the report, testing of implementation and video demonstration of the submitted codes using the data files submitted. The video demo should demonstrate how your solution meets the assessment criteria. In general, the coursework will be assessed against the Learning Outcomes (LOs) using a set of assessment criteria. This set of assessment criteria allows assessing how successful you have met the LOs. In order to ensure consistent use of the relevant criteria, the assessment criteria are summarised in the following assessment matrix and grid. This is an indicator of how the marks will scale across each category of the learning outcomes it covers.