**Project Assignment – Computer Engineering Track**

ECG Classification for Arrhythmia Detection

Responsible Lecturer: Rajendra Bishnoi (R.K.Bishnoi@tudelft.nl)

## Context

Cardiovascular diseases (CVDs) are a group of disorders involving the heart and the blood vessels. Most CVD patients are accompanied by cardiac arrhythmia and it is one of the most common problems in cardiology. Cardiac arrhythmias are slowed, accelerated or irregular heartbeats caused due to improper functioning of the heart. They are usually identified by using an electrocardiogram or what is more commonly known as an electrocardiogram (ECG). ECG is a physiological signal that records the electrical activity of the heart. It is widely used by health professionals in diagnosing abnormal heart rhythms and investigating problems with the electrical conduction of the heart. Besides, early detection of cardiac arrhythmia can help prevent the emergence of any catastrophic heart diseases. In this Final Project Assignment, you will develop an ML model to detect various types of abnormal heartbeats (arrhythmias) from heart activity recorded in the form of an electrocardiogram (ECG).

## Purpose

By developing this assignment, you will put into practice all the learning concepts introduced so far in Lectures 1 to 4. You will develop strategies for detection of abnormal heartbeats (arrhythmias) and solve practical questions that arise when developing an ML model. This assignment covers this course learning objectives (LOs): LO3, LO4 and LO5.

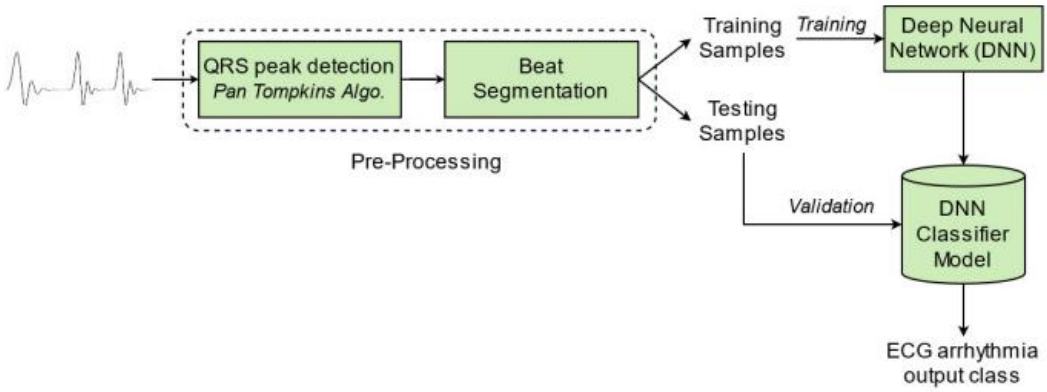
## Resources

You will receive feedback from a Lecturer on Week 7.

## Instructions

### Activities

A fully automated system for ECG arrhythmia classification can be divided into several steps from signal acquisition to identifying the abnormality, as shown in Figure. After capturing the ECG using sensors, the signal is pre-¬processed. The entire recorded dataset is then divided into two parts: ¬ training set and testing set. Usually a split percentage of 80%¬ to 20% of dataset is used for training set and testing set, respectively. Next, the neural network is trained and validated for the predicted output class.




Following two tasks are planned for this Final Project Assignment:

1. Data pre-processing: In this task, you will first convert ECG data into .csv format and then extract heartbeats from the ECG signals.

2. Classification: In this task, you will do the classification of the pre-processed ECG data into AAMI classes. Following are the required steps to perform this task:
   - Split the data into a training set, a validation set and a test set.
   - Train the network using training set.
   - Tune hyperparameters using validation set.

- Report the network performance (accuracy, precision, recall, F1-score, confusion matrix etc.) using the test set.

**Other instructions**

- You will work in pairs.
- Decisions need to be made together, but Tasks can be done individually.
- We recommend splitting the tasks. Any member must be capable of arguing any decision made.
- At least one of the models must be a *deep neural network (Lecture 5)***.**
- One report per pair. The report must follow the proposed structure with a maximum number of pages of 10.
- Deadline: Week 8.

**Deliverables**

1. Final Project Report (see instructions below)
2. Project Assignment Python code

*Report Structure*

- Members, emails, student numbers.
- Summary (less than 200 words)
- Detailed ML pipeline (include workflow figure).
- Task 1: selected options, argumentation for the selection, model(s) developed, results, validation, comparisons.
- Task 2: argumentations for the model(s) developed, validations, results, comparisons.
- Task 3: argumentations for the model(s) developed, validations, results, comparisons.
- Conclusions (less than 200 words)

**Assessment Criteria**

You will be evaluated based on a predefined rubric. Check the course Brightspace page to get access to the rubric.

The Project Final Report can be considered *inadmissible*, which will render a FAIL grade for the group, if

- English is not understandable (e.g., full of typos).
- Deep neural networks were not used (as one of the tested models).
- Figures are not legible.
- The report does not follow the proposed structure.

If the report is considered *admissible*:

- English will *not* render extra points.
- Quality of the Python code will *not* render extra points.

**Submission Instructions**

Please submit your Final Project Report in a PDF format and your Python code in Brightspace before the deadline.