



Rapport Data Analysis III - Arduino

Groupe :

Ibrahim ALKARDO

Yasmine ELIDRISSI

Guillaume LE DEZ

Jiek RUAN

Tanya TIBOUCHE

1- Introduction

L'objectif de cette expérience est d'explorer et d'analyser les signaux électrocardiographiques (**ECG**) afin de caractériser les différents types de rythmes cardiaques et de détecter les anomalies associées à des pathologies telles que les arythmies. Cette analyse repose sur l'étude des segments ECG extraits de patients réels, dans le but de mieux comprendre les caractéristiques temporelles et statistiques de ces signaux.

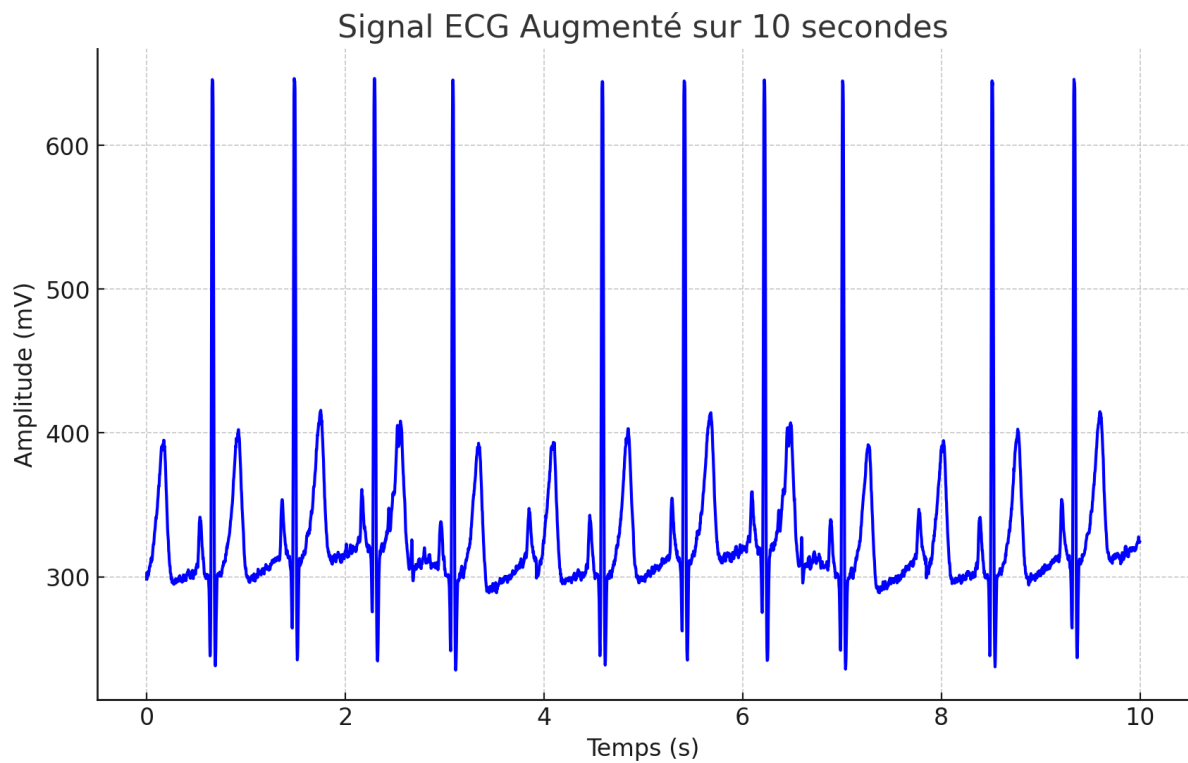
L'**électrocardiogramme (ECG)** est un examen médical simple qui mesure l'activité électrique du cœur à l'aide de capteurs placés sur la peau. Il enregistre les battements cardiaques sous forme d'un graphique composé de courbes et d'ondes.

En analysant la forme et la durée de ces ondes, il est possible de détecter des irrégularités du rythme cardiaque, comme des battements trop rapides, trop lents ou irréguliers. L'ECG est donc un outil essentiel pour surveiller la santé du cœur et identifier d'éventuelles pathologies.

2- Méthodologie

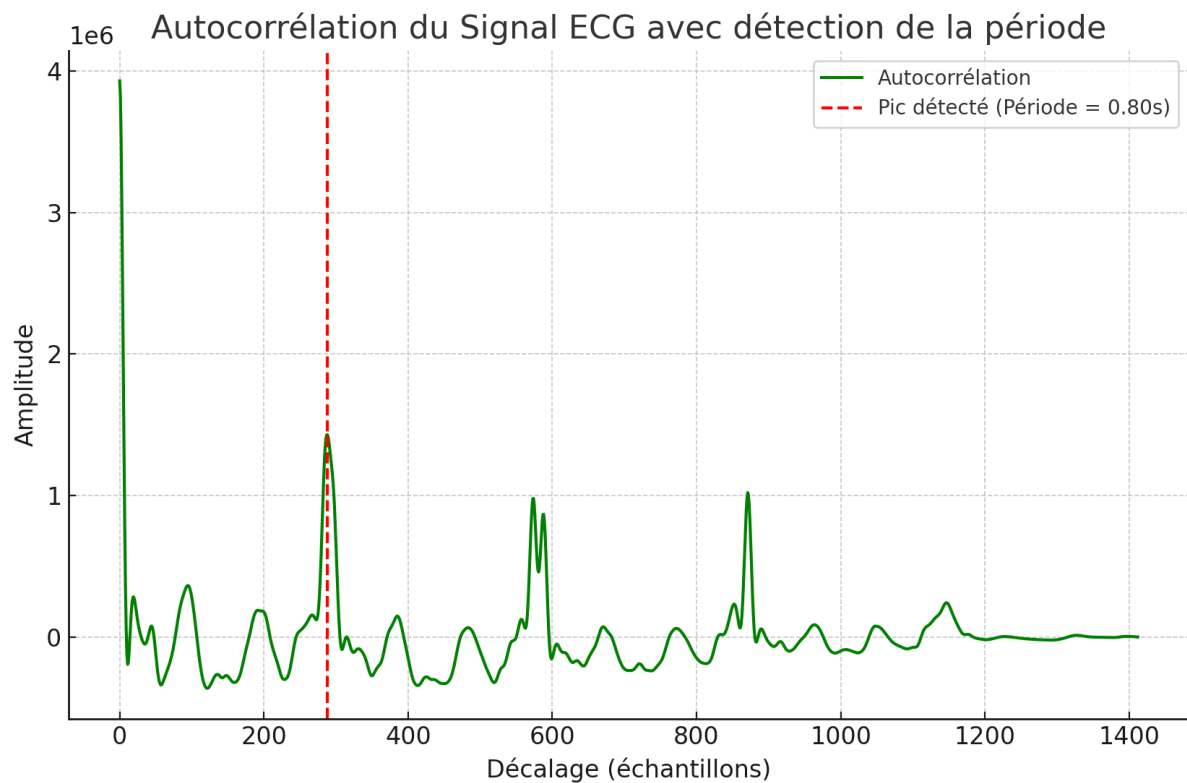
a- Acquisition du signal ECG

Nous avons enregistré un signal de 10 secondes avec une fréquence d'échantillonnage de 360Hz pour rester conforme à la source de nos Datasets (mit-bih). Ce qui donné le graphique du signal brut ci-dessous :



b- Autocorrélation et estimation de la fréquence cardiaque

On identifie le premier pic significatif après le 0 (le signal est parfaitement corrélé avec lui-même à $t=0$). C'est pour pouvoir déterminer la période du cycle cardiaque.



On voit que le premier pic significatif est au 288ème échantillon.

$$Période = \frac{\text{décalage}}{\text{Fréquence d'échantillonnage}} = \frac{288}{360}$$

Ça nous donne une période d'environ 0.8 - 0.9 secondes, et de là on peut calculer la fréquence cardiaque en BPM :

$$Fréquence = \frac{60}{0.8} = 75$$

Ce qui nous donne une estimation de 70-75 BPM (pour notre ECG, on était à 69 BPM), qui est cohérent avec un rythme cardiaque normal.

c- Interpolation du signal

On a opté pour l'implémentation de l'*interpolation cubique* en python comme méthode pour avoir un signal "lissé".

L'interpolation cubique est plus adéquate dans ce contexte que l'interpolation linéaire car elle permet de préserver au mieux la forme du signal initial, vu que les points qu'on relie sont des courbes et non pas des lignes droites.

Pour ça, on a fait du suréchantillonnage pour avoir plus de points.

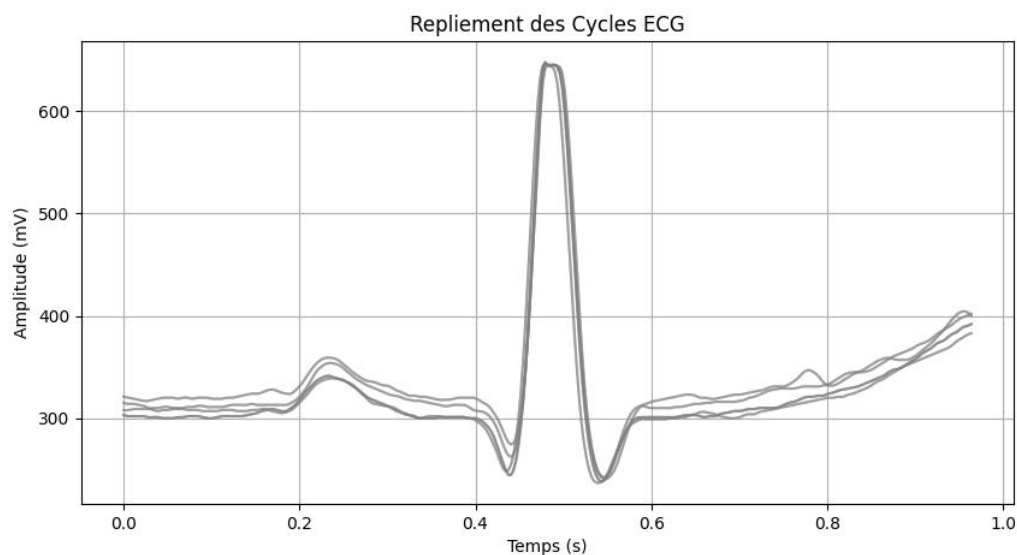
d- Repliement et Moyennage du Signal sur une Période Unique

On a commencé par isoler une période du signal ECG, en détectant le premier pic significatif du signal brut.

On a ensuite fait aligner les pics en superposant plusieurs cycles. Avec ça, on a réussi à avoir un signal lissé.

On a opté pour un repliement avec le pic centré. Etant donné que l'ECG provient d'une personne non atteinte de maladies cardiovasculaires, il suit un certain pattern.

En alignant les cycles autour du pic, on s'assure que chaque battement est correctement superposé.



On a quand même de légères variations liées aux bruits et aux fluctuations naturelles des battements.

e- Sparse du signal

On l'utilise pour simuler un sous-échantillonnage, cela nous servira dans la partie entraînement de modèle pour tester la robustesse de notre modèle face à des données incomplètes.

On a utilisé un facteur de 2 (qui rajoute un 0 entre chaque point).

