# Rapport de stage :

Traitement et annotation de données médicales

Du 15/04/2021 au 23/06/2021

Tuteur de stage : Youcef SKLAB

Professeur référent : David HEBERT

Entreprise : Institut de Recherche pour le développement
32 Avenue Henri Varagnat 93600 Bondy

Service : UMI 209 UMMISCO

Réalisé par Ismail SARR

## REMERCIEMENTS :

**À Ahmad FALL, Edi Prifti et Youcef Sklab**

Pour m'avoir accueilli et suivi tout au long de ce stage et m'avoir apporté leurs précieux conseils. Leur suivi m'a permis de traiter le sujet défini dans l'objectif de mon stage.

**À Gaspar Roy, Alex Lence et Fatima Zohra Souafi**

Pour leur présentation chaque semaine sur l'avancement de leurs stages qui m'a apporté des connaissances intéressantes.

**À M. Hébert et M. Santini**

Pour leurs cours de Programmation qui mon été d'une précieuse aide durant mon stage mais également pour mon apport personnel.

**À Mme. El Alaoui et Mme. Deseilligny**

Pour ses conseille pour la rédaction du rapport et la préparation de la soutenance.

**À Mme. Cardoso**

Pour son aide et son accompagnement dans ma recherche de stage

**À l'ensemble des membres de l'UMMISCO et de l'IRD**

Pour leur accueil et leur bienveillance tout au long de mon stage

# Sommaire :

Introduction ..... 5

L'Institut ..... 6

1.1 Présentation de l'IRD ..... 6

1.2 Présentation de l'unité ..... 7

1.3 Contexte et environnement de travail ..... 7

Le Projet ..... 8

Travail réalisé ..... 9

3.1 La base de données ..... 9

3.2 Traitement des fichiers ECG ..... 10

3.3 Fonction de visualisation des données ..... 16

3.4 Traitement des fichiers Excel ..... 19

Conclusion ..... 21

Sitographie ..... 22

Références ..... 22

Annexes ..... 23

# Table des figures :

Figure 1 IRD dans le monde ..... 6
Figure 2: Exemple d'un ECG ..... 11
Figure 3 : Entête récupéré par la classe Holter ..... 12
Figure 4 : Entête de l'ECG 1 du patient 080213101609 GAL_JE ..... 14
Figure 5: Aperçu du signal de l'ECG 1 du patient 080213101609 GAL_JE ..... 15
Figure 6: Aperçu de la première visualisation du signal de l'ECG 1 du patient ..... 16
Figure 7: Visualisation globale de l'ECG 1 du patient CAR-DI ..... 18
Figure 8: Récupération des données avec pandas ..... 19
Figure 9: Aperçu du fichier XML final ..... 20
Figure 10: Schéma récapitulatif de toute ma démarche ..... 21

## Résume :

Dans ce rapport vous trouverez une description détaillée du travail que j'ai pu fournir durant mon stage de fin d'études. J'ai été accueilli au sein d'une une équipe constituée de chercheurs, de doctorants/chercheurs et de stagiaires. Mon responsable de stage m'a confié comme mission la préparation de données médicales afin que l'équipe puisse ensuite faire des analyses statistiques sur celles-ci. L'objectif était de réaliser une fonction qui convertit toute la base de données dans un format permettant de réaliser ces analyses. Ces analyses permettront de prévoir l'apparition de maladie complexe. Au moment où j'écris ce rapport, la conversion des données est en cours grâce au fonction que j'ai pu écrire. Les prochaines étapes du projet seront les analyses statistiques à partir de ces nouvelles données.

In this report you will find a detailed description of the work I was able to provide during my final internship. In summary, I was welcomed into a team made up of researcher, doctoral student/researcher and intern. My mission was to prepare medical data so that the team could then do statistical analyzes on it. The goal was to create a function that converts the whole database into a format that allows these analyzes to be performed. These tests will help predict the onset of complex disease. At the time of writing this report the data conversion is underway thanks to the function I was able to write. The next steps of the project will be the statistical analysis based on these new data.

# Introduction

Dans le cadre du DUT Statistique et Informatique décisionnel j'ai eu l'opportunité d'effectuer un stage de fin d'étude afin de compléter ma formation et d'avoir une première expérience professionnelle. Celle-ci nous permet de travailler et d'apprendre directement avec des professionnels, et donc d'enrichir notre panel de compétences. Cela nous permet également d'apprendre de nouvelles méthodes de travail et d'adapter nos connaissances, parfois très théoriques à la pratique au sein d'un milieu professionnel et dans le cadre de mon stage un cadre scientifique également.

C'est donc dans ce but que dès la rentrée scolaire, j'ai entrepris mes recherches afin de trouver une entreprise pouvant m'accueillir pour une durée entre 8 et 12 semaines sur un sujet en adéquation avec mon DUT. Avec beaucoup de satisfaction, j'ai pu signer une convention de stage à la mi-avril et j'ai eu la chance d'être accueilli par l'Institut de Recherche pour le Développement (IRD) au sein de l'Unité de Modélisation Mathématique et Informatique des Systèmes Complexes (UMMISCO), ou l'on m'a permis de travailler dans le cadre du projet DeepECG4U tout au long de mon stage.

Dans ce rapport vous trouverez un exposé complet de mon stage et de ma contribution au projet, mais également sur ce que le projet m'a apporté au cours de cette première expérience professionnelle. Dans un premier temps, je présenterai l'institut et l'unité dans laquelle j'ai effectué mon stage d'étude. Dans un second temps, j'exposerai le projet et les travaux que j'ai pu réaliser, dans le cadre de cette collaboration avec l'équipe de UMMISCO. Et pour compléter ce rapport, je vous décrirai ma conclusion sur la réalisation de mes travaux.

# I. L’institut :

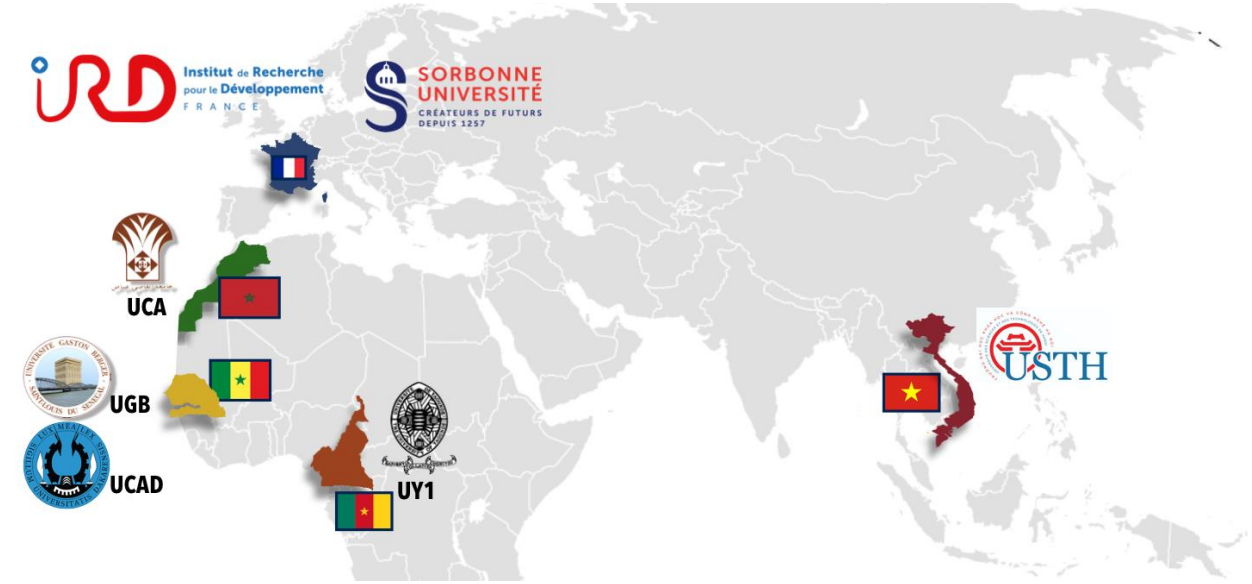



*Figure 1 IRD dans le monde*

Qu'est-ce que l’IRD ?

L’Institut de recherche pour le développement (IRD) est un établissement public français placé sous la double tutelle des ministères de l’Enseignement supérieur, de la Recherche et de l’Innovation et de l’Europe et des Affaires étrangères. Il porte une démarche originale de recherche, d’expertise, de formation et de partage des savoirs au bénéfice des territoires et pays qui font de la science et de l’innovation un des premiers leviers de leur développement.

L’IRD, c’est un acteur français majeur de l’agenda international pour le développement. Son modèle est original : le partenariat scientifique équitable avec les pays en développement, principalement ceux des régions intertropicales et de l’espace méditerranéen. Pour ma part j’ai été accueilli dans la délégation ile de France qui administre et accompagne 22 unités de recherche ainsi que leurs équipes.

La mission première de l’IRD est de produire de la science focalisée sur la zone intertropicale et méditerranéenne et fondée sur un partenariat scientifique équitable avec les communautés d’enseignement supérieur et de recherche des pays et régions concernés. Cette mission correspond à un double objectif :

- Contribuer aux avancées de la connaissance scientifique en matière de développement durable
- Aider à mieux fonder les politiques de développement sur la science.

C’est au sein de l’Unité de Modélisation Mathématique et Informatique des Systèmes Complexes (UMMISCO) que j’ai travaillé durant trois mois. UMMISCO est une Unité Mixte Internationale placée sous la tutelle de 7 organismes universitaires en France, au Maroc, au Sénégal, au Cameroun et au Vietnam. Faisant le constat que beaucoup de questions liées au développement durable nécessitent d’intégrer de multiples savoirs, points de vue et échelles d’observations sur le monde réel, et que les dynamiques globales des systèmes concernés sont le fruit d’interactions complexes entre leurs composantes abiotiques, biologiques, écologiques et sociales, les thématiques scientifiques de UMMISCO sont dédiées à la conception d’approches et d’outils de modélisation mathématique et informatique qui permettent de mieux appréhender et comprendre les dynamiques émergentes des "systèmes complexes", de dessiner leurs scénarios d’évolution les plus probables, et d'alimenter, in fine, la prise de décision en matière de développement durable. L’implantation principale de l'UNESCO en France est à Bondy, dans le centre IRD-France Nord.

Intégration à l’entreprise :

Malgré le contexte sanitaire actuel, j’ai eu la chance de pouvoir me rendre dans les locaux de Bondy une à deux fois par semaine et rencontrer certains membres de l’équipe. Le reste de mon stage s’est déroulé en distanciel *« télétravail »,* avec des réunions tous les vendredis, et parfois en milieu de semaine si nécessaire. Malgré les limites du « télétravail », j’étais en contact quotidiennement avec l’équipe que j’ai sollicitée afin d’obtenir de l’aide et des réponses à mes questions, mais également sur l’état d’avancement de mon travail.

Durant cette période de stage, j’ai également pu assister à la réunion de toute l’unité, au cours de laquelle ont participé des intervenants exerçant leurs fonctions au Vietnam, au Sénégal ou encore au Maroc. Ces réunions étaient très enrichissantes, il y a eu des présentations de plusieurs projets, mais également des thèses soutenues par des doctorants. Pour ma part, j’ai travaillé avec l’équipe du projet DeepECG4U que je détaillerai ultérieurement. Grâce à la réactivité des collègues du projet, j’ai toujours pu obtenir de l’aide et des réponses à mes questions, ce qui m’a permis de surmonter mes blocages, de susciter ma curiosité, d’accroître ma motivation et de travailler efficacement pour faire évoluer mes compétences. Nos réunions hebdomadaires avaient lieu sur ZOOM et la majorité de nos échanges s'effectuent sur la plateforme de communication SLACK ou par mail. En ce qui concerne mon environnement de travail, mon tuteur m’a créé un accès au serveur Gitlab du projet, dans lequel je pouvais à tout moment, poser les questions techniques et exposer mes problèmes au niveau de la programmation. J’avais également accès au serveur Jupyter Lab de l’unité ou j’ai réalisé la majeure partie de mon travail. Jupyter Lab est une application web sur laquelle j’ai travaillé durant mes cours de DUT statistiques informatique décisionnelle. Cette plateforme (Annexe 1) permet de programmer avec différents langages de programmation et de générer des notebooks.

# II. Le Projet :

Durant mon stage j'ai donc pu participer au vaste projet DeepECG4U sous la direction de Monsieur Edi PRIFTI (chargé de recherche et coordinateur du projet), de Monsieur Youcef Sklab (ingénieur de recherche) et de Monsieur Ahmad Fall (doctorant). Ce projet étant assez complexe, je vais tenter de l'expliquer avec des mots simples. Certaines maladies cardiovasculaires comportent des effets secondaires liés à la prise de médicaments qui peuvent provoquer une forme particulière d'arythmie, appelée Torsade de Pointe, qui peut dégénérer entraînant la mort. Une arythmie est une anomalie qui affecte la fréquence cardiaque normale. En présence d'arythmie, le cœur a tendance à battre trop lentement, trop vite ou de façon irrégulière. Sur un électrocardiogramme (ECG), qui est une représentation graphique de l'activité électrique du cœur, on peut apercevoir quelques spécificités.

L'objectif est de développer un outil de prédiction personnalisée automatisée du risque de Torsade de Pointe des patients. Cela permettra d'améliorer la précision de l'évaluation du médecin et réduire le risque d'événements indésirables. Cet outil est aujourd'hui développé en utilisant l'intelligence artificielle (IA), qui apporte aujourd'hui une réelle aide à la pratique médicale.

L'apprentissage du modèle suivra l'ensemble des méthodes du Deep Learning (apprentissage profond). L'ajout de ces règles ne fait l'objet d'aucune intervention humaine. L'apprentissage profond utilise alors différentes couches neuronales qui forment un réseau artificiel. L'apprentissage profond en particulier a apporté un changement radical dans le domaine de la reconnaissance des formes et de l'apprentissage machine lui-même, améliorant la plupart des modèles antérieurs tels que la classification des images et le traitement du langage nature. Plus précisément, en cardiologie, l'apprentissage profond a récemment été utilisé par plusieurs applications, notamment la détection de divers types d'arythmies cardiaques courantes telles que la fibrillation auriculaire, l'infarctus du myocarde, etc. Cependant, son utilisation pour prédire les événements de Torsade de Pointe n'avait pas encore été explorée. C'est dans ce contexte que les chercheurs de l'unité(UMMISCO) ont proposé cette approche originale, avec leur démonstration dans un article en preprint sur researchsquare : *Edi Prifti, Ahmad Fall, Giovanni Davogustto et al. Deep learning analysis of drug-induced ECG changes to inform arrhythmia risk and improve diagnosis of congenital long QT syndrome, 22 February 2021, PREPRINT(Version1)available at Research Square*

*[ https://doi.org/10.21203/rs.3.rs-256040/v1]*

Le but final du projet est de faire progresser ce sujet de recherche et de créer une application translationnelle qui pourra être déployée dans plusieurs services de cardiologie.

# III. Le Travail réalisé :

Comme j'ai pu vous le décrire, le projet est assez vaste et demande des connaissances et des compétences qui dépassent le niveau du DUT STID. Néanmoins, malgré mon faible niveau d'étude je pouvais comprendre ce projet et y participer. Je vais revenir sur le travail que j'ai réalisé. J'ai travaillé sur la préparation des données nécessaires et plus particulièrement le parsing, indispensable pour tester les différents modèles. Ma mission était de changer le format des données afin de les rendre exploitables et de créer des fonctions qui permettent de visualiser l'ensemble des données.

## 3.1 La base de données :

Les données proviennent d'une étude clinique menée par Le Centre d'Investigation Clinique de l'Hôpital de la Pitié-Salpêtrière. L'objectif de base était d'appliquer une approche pangénomique dans la recherche de facteurs génétiques (gènes et variantes) impliqués dans les modifications de l'intervalle QT (symptôme de la Torsade de Pointe) en réponse à une stimulation pharmacologique (Sotalol) et Protocole GENEREPOL physiologique (stimulation, sympathique) dans, la population générale apparemment bien portante.

Critères de sélection :

- Sujets européens et nord-africains âgés de 18 à 60 ans, apparemment sains, indemnes de toute pathologie significative et de tout traitement au long cours.
- Nombre de sujets nécessaire : 1000 participants (volontaires)
- Durée de la recherche : 3 ans
- Durée de participation de chaque volontaire : environ 6 heures
- Méthodologie : Étude transversale de 1000 sujets apparemment sains qui recevront une dose unique de 80 mg de Sotalol et auront une épreuve d'effort sur bicyclette ergonomique et une stimulation auditive.

## 3.2 Traitement des fichiers ECGs :

J'ai débuté mon stage avec l'objectif de construire des parseurs permettant de lire et importer des données tests avec les fichiers comprenant les électrocardiogrammes. Ces fichiers étaient en format binaires donc illisibles avec un simple éditeur de texte (Annexe 2). Ma première tâche consistait à partir de dix fichiers test contenant les électrocardiogrammes (ECG) pour un patient, de trouver le moyen de les convertir vers un format exploitable.

Mon premier objectif était de réussir à extraire les informations du signal. Afin de m'aider, mes encadrants ont mis à ma disposition une description détaillée sur le format d'enregistrement (Annexe 3) du fichier, le format ISHNE. Ce document expliquait comment les données étaient structurées dans le fichier. Autrement dit, la première partie contenait l'entête avec des informations comme le nom du fichier, sa taille ou encore des informations sur le patient. Le reste du fichier contenait l'électrocardiogramme classé par leads. Pour expliquer ce qu'est un lead je dois d'abord expliquer les méthodes qu'utilisent les médecins pour enregistrer la fréquence cardiaque.

Pour réaliser un électrocardiogramme, il convient de relier un appareil enregistreur (en général un électrocardiographe) à plusieurs électrodes. Elles sont au nombre de 10 : 4 sont placées sur les poignets et les chevilles, et 6 au niveau du thorax. L'enregistrement est réalisé en variant successivement les paires d'électrodes enregistrées. Diverses associations de ces électrodes qui correspondent à différents circuits d'enregistrement, sont reliées à un stylet assurant le tracé correspondant à la dérivation (reflet localisé de l'activité du cœur). En tout, il y a 12 dérivations. Ce sont ces dérivations que l'on nomme Leads. Cela permet d'enregistrer plusieurs points de vue différents du même signal et de ne pas perdre la moindre information.

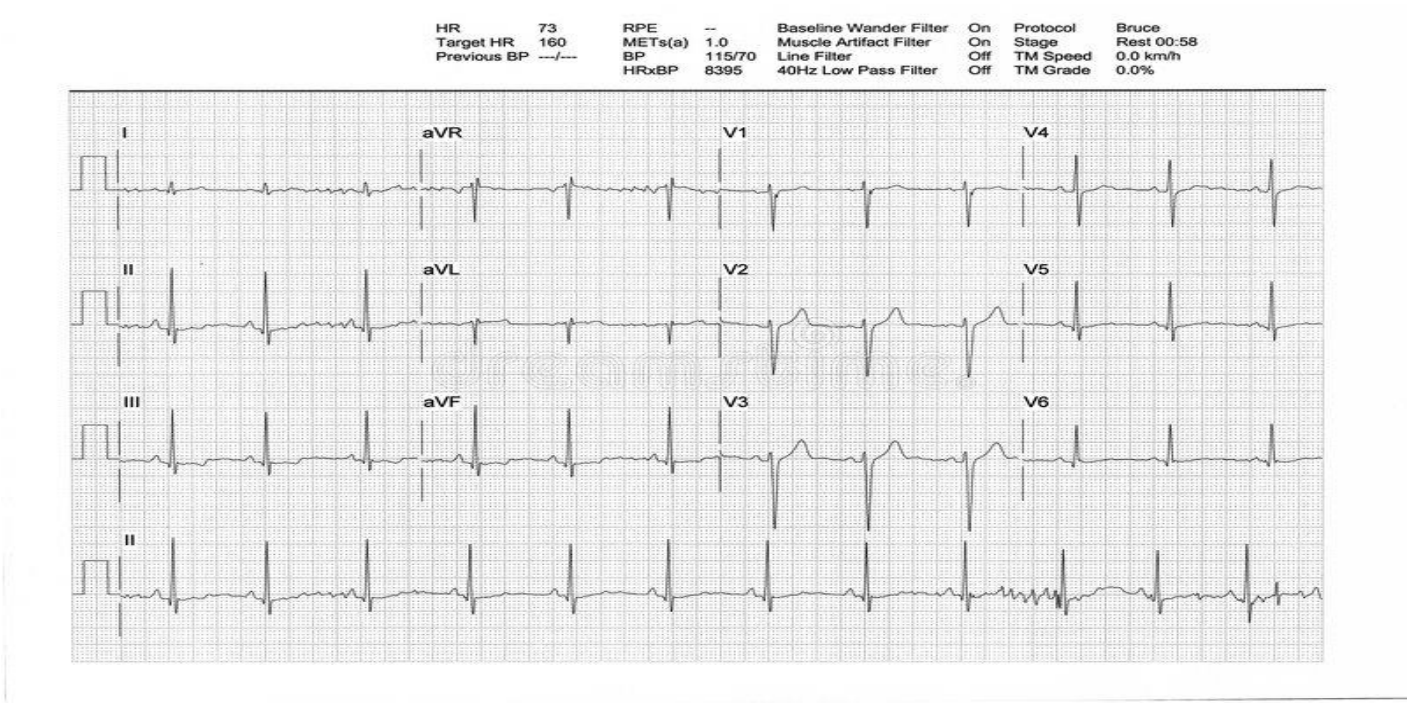



*Figure 2: Exemple d'un ECG*

Une fois le format compris, il fallait lire les fichiers avec l'outil de mon choix et j'ai décidé de me diriger vers le langage de programmation Python. Dans l'exploitation de Python. Mes encadrants m'ont conseillé d'utiliser une classe particulière qui permet de lire ces fichiers. En Python une classe peut se définir simplement comme un regroupement de données sur lesquelles plusieurs attributs sont définis ce qui permet de les manipuler. Pour réussir à utiliser cette classe j'ai dû apprendre le fonctionnement de la programmation orienté objet, avec Python grâce à des vidéos et des articles. En résumé, la classe prend en entrée le fichier binaire et me permet de récupérer les données grâce à divers attributs.

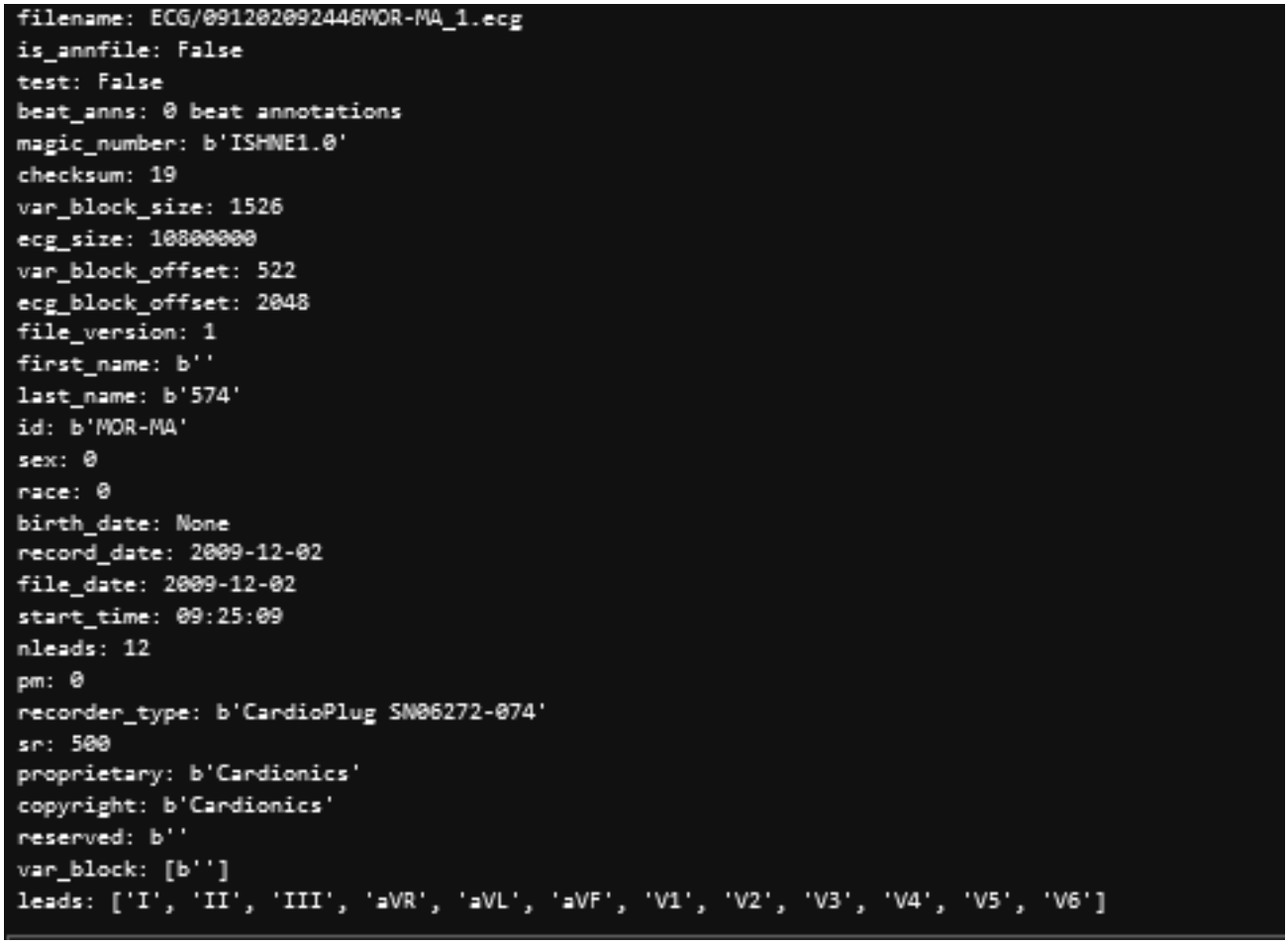



*Figure 3 : En tête récupéré par la classe Holter*

De la même manière que dans l'exemple, cela m'a permis d'afficher une partie du signal. Mais cela n'était pas suffisant, car il fallait directement charger l'information dans une variable que je pouvais intégrer au futur fichier parsé.

J'ai donc dû créer une fonction qui récupère chaque valeur de l'entête dans plusieurs variables et qui les conservent en mémoire. Pour ce qui est du signal, en m'inspirant de la classe j'ai trouvé un moyen de créer une liste Python qui contient en index le nom du Lead en attribuant les valeurs du signal correspondant. Dès lors que j'avais obtenu ces bonnes bases, je détenais tous les outils pour créer un nouveau fichier contenant d'un côté l'en-tête et de l'autre le signal.

J'ai donc commencé par faire mes recherches concernant le futur format des fichiers. Pour commencer, sur les conseils de Monsieur Ahmad FALL (doctorant à l'UMMISCO) et co-encadrant, je me suis orienté vers le format HDF5.

Le format HDF5 (Hierarchical Data Format, V5) est un format de type conteneur de fichier. Ce format permet de créer des groupes dans le même fichier et donc de ranger le signal dans différents datasets. Cependant, il comporte plusieurs inconvénients. Premièrement, il est impossible de créer progressivement les fichiers car le fichier hdf5 sera une sorte de bases de données qui regroupera tous les fichiers, et à ce stade, nous sommes donc dans l'obligation de créer tous les fichiers d'un seul coup ce qui risque d'utiliser beaucoup trop de RAM (mémoire informatique ou sont enregistrées les informations traitées). Et l'autre difficulté résulte de son format qui

est trop restrictif : sous forme de table ce qui risque de nous priver de pas mal de liberté.

Après concertation avec l'équipe nous avons décidé d'utiliser le format de fichier XML (Extensible markup language) car il présente des avantages : Le format est très pratique car il permet de structurer les données grâce à des balises. Ce système va me permettre de créer plusieurs balises en fonction du type d'information que je souhaite traiter. Il s'agit également d'un avantage pour les chercheurs qui pourront facilement récupérer l'information désirée grâce à ces balises. De plus, le fichier présente un autre atout, celui de ne contenir que des chaînes de caractères, qui peuvent être compressées afin de réduire le volume des données. C'est à partir de ce constat, et de ces avantages que nous avons décidé d'utiliser le format XML.

Tout d'abord j'ai travaillé sur un seul fichier ECG tout en me documentant sur ce nouveau format. J'ai réfléchi à comment organiser le futur fichier XML. J'ai divisé les informations contenues dans l'entête en trois balises :

- Header : qui contiendra les informations propres au fichier (taille, nom etc.)
- Patient : qui regroupe les informations personnelles du patient (nom, origine ethnique, etc.)
- Record : où seront entreposées les informations sur les méthodes d'enregistrement du signal (fréquence, machine utilisée etc.)
- Signal : qui sera la balise contenant le signal de l'électrocardiogramme.

Une fois ce modèle réalisé, j'ai écrit directement le fichier XML pour obtenir un aperçu du format final. Après plusieurs corrections de mes encadrants, le modèle de fichier XML fut approuvé, et j'ai pu passer à l'étape suivante, L'automatisation du processus.

J'ai pu très vite me rendre compte que la quantité de données est très importante et qu'il m'était impossible d'écrire manuellement les fichiers XML. En effet, il y a environ 1000 patients avec une dizaine de fichiers, ce qui revient à 221G. J'ai donc entrepris des investigations pour apprendre à écrire un fichier XML sur Python. Après plusieurs jours de documentation et de tests, j'ai décidé d'utiliser la librairie «*xml.dom*». Cette librairie me permet de concevoir un document XML vide puis de rajouter les balises avec le nom souhaité. Ensuite, dans chaque balise, j'ai la possibilité d'ajouter la valeur que je souhaite, dès lors qu'elle respecte la chaîne de caractère. J'ai donc adapté la fonction qui récupère le signal afin qu'elle convertisse toutes les valeurs en chaînes de caractères. Après cette étape, il ne me restait plus qu'à écrire les balises afin d'obtenir le fichier test préalablement écrit. Voici un aperçu :


*Figure 4 : Entête de l'ECG 1 du patient 080213101609 GAL_JE*

Nous pouvons constater que certaines variables ont été rajoutées par rapport à l'entête brut des ECG. En effet, à la demande d'un de mes encadrant le chercheur Edi PRIFTI (porteur du projet) j'ai rajouté certaines informations utiles :

- La durée : « duration » contient la durée totale de l'enregistrement. Pour l'obtenir, j'ai dû diviser la taille totale du fichier et la diviser par la fréquence (nombre de points par secondes)
- L'heure de fin de l'enregistrement : « end_time », pour l'obtenir j'ai dû créer une fonction qui convertit la durée précédente en heures/minutes et qui l'ajoute à la durée du début de l'enregistrement.

Toujours dans le même fichier, nous retrouvons directement, à la suite le signal. Voici comment il est construit :

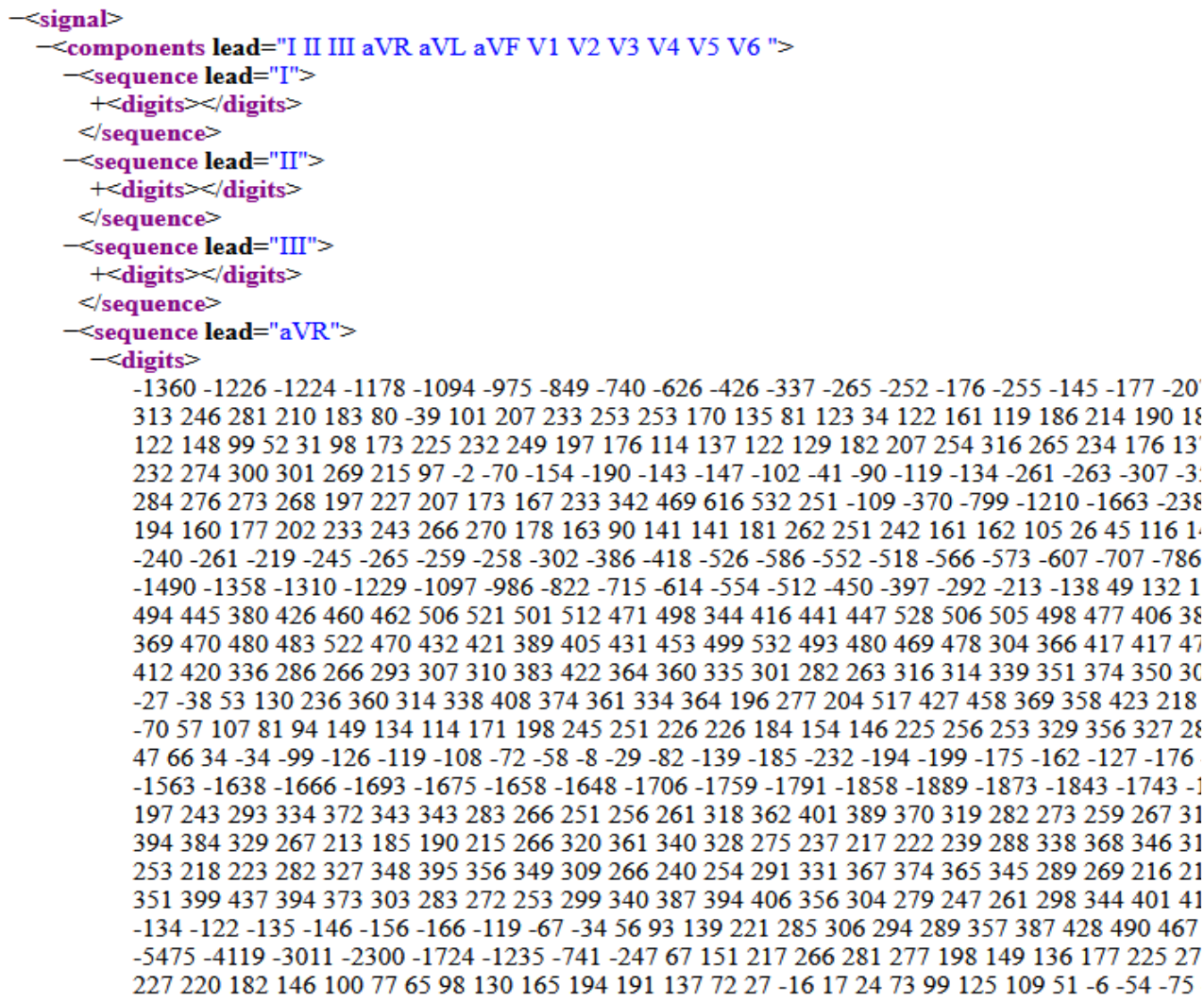



*Figure 5: Aperçu du signal de l'ECG 1 du patient 080213101609 GAL_JE*

Nous avons gardé la même structure que dans le fichier de base. La balise « components » contient le nom de chaque Leads. Nous détenons maintenant, une balise qui contient le nom Lead puis en dessous une autre balise, « digits », dans lequel est stocké le signal associé au Lead correspondant

## 3.3 Fonction de visualisation des ECGs :

Une fois la fonction appliquée au fichier test est validée avec succès par mes encadrants, il fallait passer à l'étape suivante. A la demande des responsables, j'ai dû réaliser une fonction qui permet de visualiser le signal.

Au début de mon stage, j'avais essayé de visualiser les données. J'avais créé une fonction qui réalise cette tâche mais les membres du projet, ils m'ont signalé plusieurs anomalies.

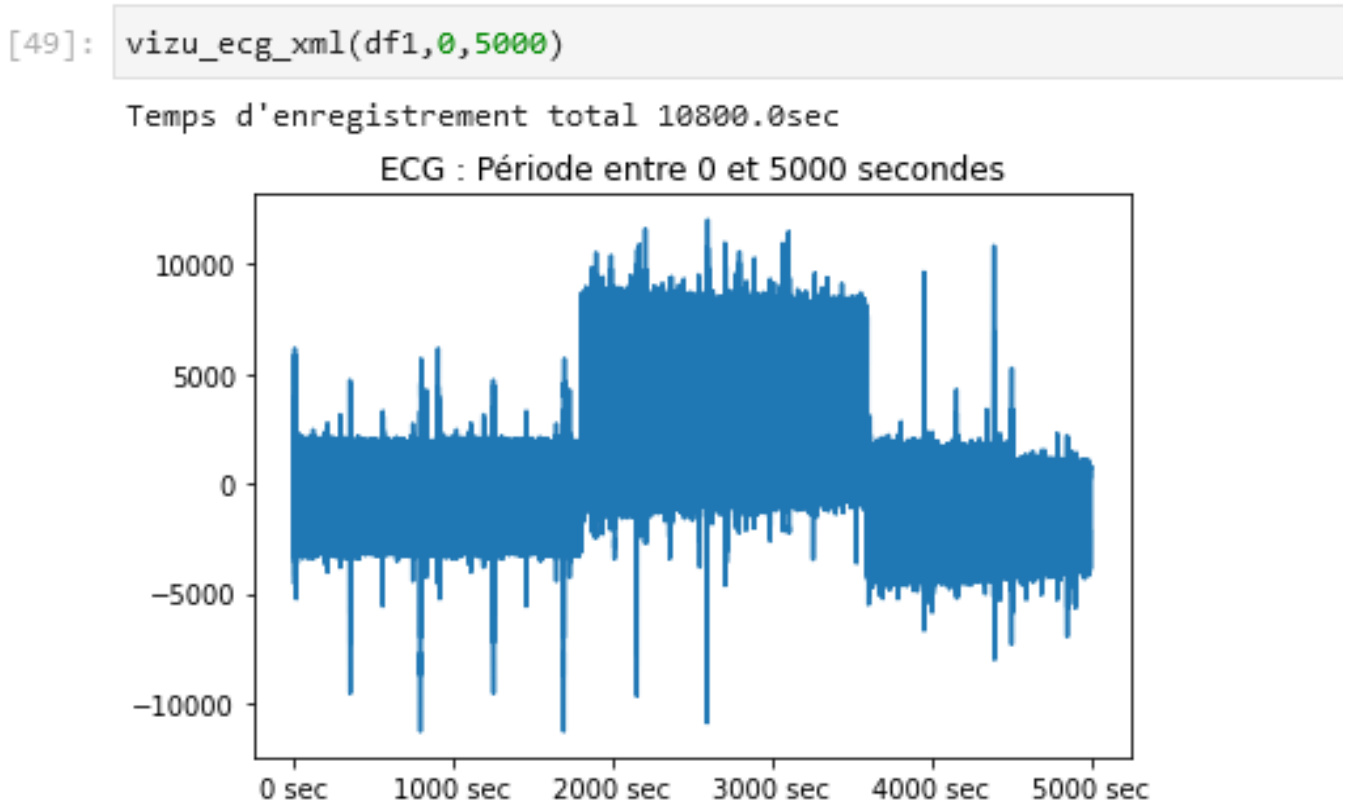



*Figure 6: Aperçu de la première visualisation du signal de l'ECG 1 du patient*

Premièrement, nous avons observé des décalages anormaux au niveau du signal. Ces décalages sont dus au fait que le signal est mal concaténé (voir figure 6). Celui-ci étant divisé en 12 parties, j'ai cru qu'il fallait mettre la deuxième partie du signal à la suite de la première partie. Mais je n'avais pas compris la structure propre aux électrocardiogrammes. Chaque partie correspond aux signaux enregistrés sur une zone du corps par une électrode. Donc toutes les parties ont été enregistrées sur la même période. Il n'est donc pas pertinent de mettre le signal enregistré au niveau du bras droit à la suite du signal enregistré par l'électrode situé sur la poitrine.

Enfin, le troisième problème rencontré provenait des données prises en entrée par la fonction. En effet, la fonction crée une visualisation à partir des fichiers ECG de bases. Le but est d'écrire une fonction qui crée une visualisation à partir du

fichier XML parsé. Après avoir pris en compte tous ces paramètres, j'ai commencé à réécrire une fonction qui répond à tous ces critères.

Dans un premier temps, je me suis renseigné sur la lecture de fichier XML sur Python. Grâce à la librairie « pandas » (1) j'ai pu stocker le signal dans chaque Leads dans différentes colonnes. Cela m'a permis de résoudre la troisième difficulté. Pour résoudre la première difficulté, j'ai dû écrire une fonction qui accepte en entrée tous les fichiers XML du patient et qui crée un grand dictionnaire python avec 12 clés, une pour chaque lead, et qui prend chaque lead de chaque fichier et qui les concatène. Pour illustrer, nous avons le premier Leads du deuxième ECG qui sera à la suite du premier Leads du premier fichier ECG. Le travail est identique à chaque fichier et à chaque Lead sans décalages anormaux lors de la concaténation.

Après avoir étudié sur les données d'entrées, j'ai travaillé sur le cœur de la fonction, la visualisation. J'ai décidé de la rendre plus élaborée. En partant d'un visuel réalisé par mon tuteur sur le logiciel Rstudio, j'ai tenté de réaliser un visuel similaire avec python. Concrètement, il fallait rajouter plusieurs options :

- Le choix du titre par l'utilisateur.
- Le choix d'afficher le visuel en couleur ou en noir et blanc.
- Option pour zoomer sur les zones anormales
- Le choix de la durée à visualiser

Un autre point important se situe dans la conversion des données. En effet les données dans le fichier XML sont sous forme de chaînes de caractères. Il faut donc reconvertir environ 5400000 points par fichiers en nombres entiers pour les visualiser. Ce processus prend beaucoup de temps et je n'ai pas encore trouvé le moyen de le raccourcir.

En appliquant tous ces critères à ma fonction voilà un aperçu du visuel final :

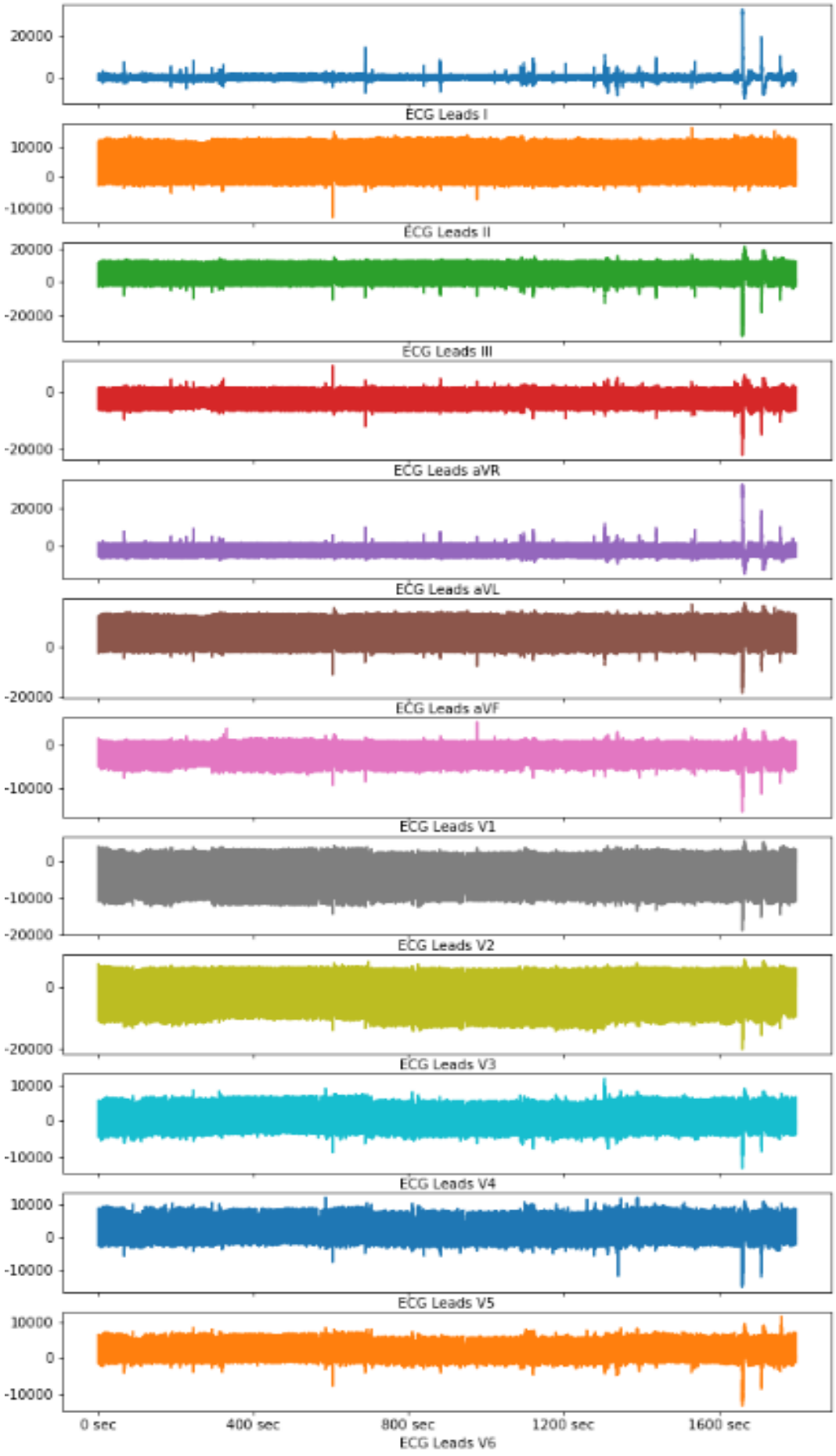



*Figure 7: Visualisation globale de l'ECG 1 du patient CAR-DI*

## 3.4 Parsing des fichiers Excel :

La deuxième partie de mon stage était le parsing des fichiers Excel qui regroupent les informations du patient. Les fichiers Excel contiennent le protocole clinique. Ce document contient des informations personnelles sur le patient, par exemple les différents traitements, ses habitudes tabagiques etc. Mais il contient également les commentaires et les notes prises par le médecin (Annexe 4). Ces données ont été anonymisées au préalable Il est structuré en cinq parties que j'ai conservées pour le futur fichier XML. Sachant que j'avais déjà conçu ce genre de fichier en début de stage, cela m'a permis d'y consacrer moins de temps. La seule difficulté se trouvait dans la récupération des informations contenues dans le fichier Excel.

Au départ j'ai utilisé la librairie « pandas »(1) et son module « read_xml»(2) qui permet de lire un fichier Excel et de le convertir en dataframe. Voici le résultat obtenu :

[12]:

| | Unnamed: 0 | Unnamed: 1 | Unnamed: 2 | Unnamed: 3 | Unnamed: 5 | Unnamed: 6 | Unnamed: 7 | Unnamed: 8 | Unnamed: 9 | Unnamed: 10 | ... | Unnamed: 22 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | Démographie | NaN | NaN | NaN | NaN | Date de naissance | NaN | NaN | NaN | NaN | ... | NaN |
| 1 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 2 | NaN | le volontaire est | NaN | NaN | NaN | NaN | incluable | NaN | NaN | NaN | ... | NaN |
| 3 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 4 | NaN | Sexe | NaN | Homme | NaN | NaN | NaN | NaN | Origine géographique des 2 parents | NaN | ... | NaN |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 211 | NaN | Actions entreprises | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 212 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 213 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 214 | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | NaN |
| 215 | NaN | Signature de l'investigateur | NaN | NaN | NaN | NaN | NaN | NaN | NaN | NaN | ... | _ |

*Figure 8: Récupération des données avec pandas*

Nous pouvons constater, qu'il y a beaucoup de cases vide comportant « Nan », Ces cases sont dues aux cellules vides du fichier Excel. En effet la librairie pandas est très utile lorsqu'il s'agit du tableur Excel, bien organisé alors que ce n'est pas le cas de notre fichier (Annexe 4). J'ai donc commencé à me documenter sur d'autre librairie est j'ai trouvé « xlrd ». Cette librairie permet de récupérer dans des variables python directement les informations présentes dans les cellules du fichier Excel. J'ai donc récupéré toutes les informations du fichier que j'ai ensuite stockées dans des

variables python pour enfin créer un fichier XML, bien organisé grâce à la méthode utilisée précédemment


*Figure 9: Aperçu du fichier XML final*

Ce fichier XML patient est donc relié aux fichiers ECG grâce à un identifiant ce qui permet de naviguer facilement entre les fichiers. La structure est la même que le fichier Excel à l'exception de quelques optimisations.

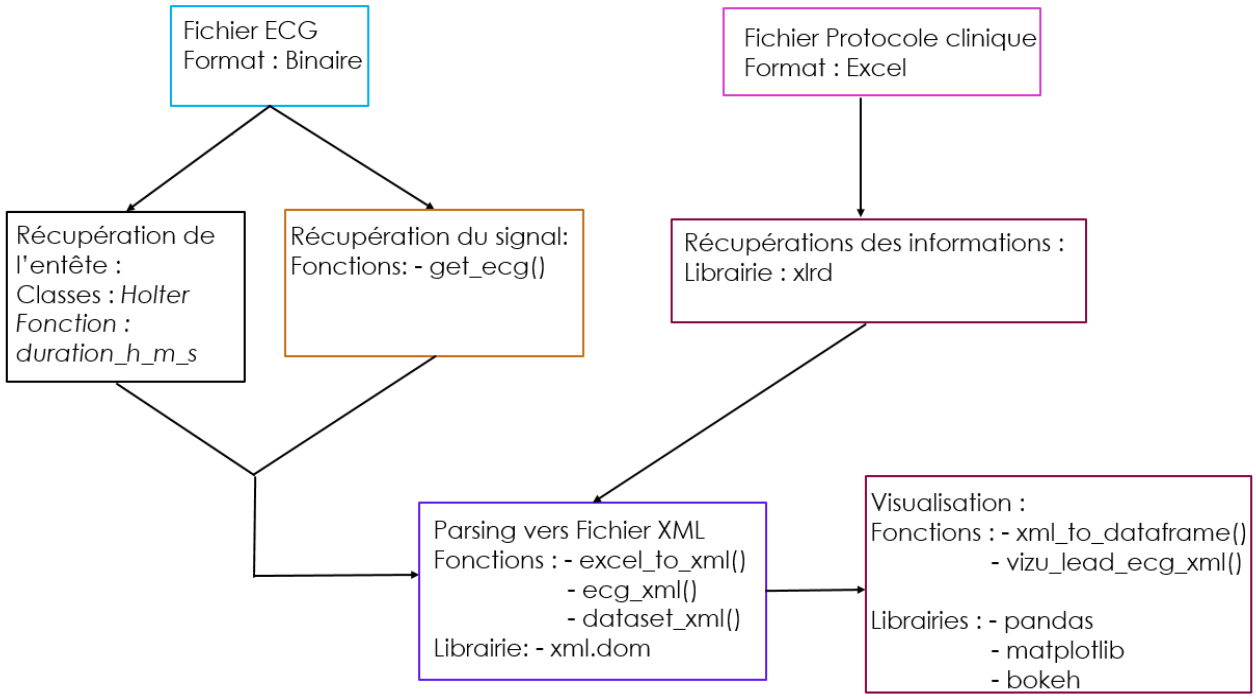



*Figure 10: Schéma récapitulatif de toute ma démarche*

Voilà qui résume le travail que j'ai pu réaliser en trois mois de stage. Néanmoins il reste encore beaucoup à faire :

- Améliorer la fonction de visualisation en ajoutant des légendes à partir du fichier XML qui contient les informations sur le patient (heure de prise du Sotalol, heure de l'effort physique etc.)

- Conversion de toute la base de données. Maintenant que toutes les fonctions de parsing ont été approuvées il faut les appliquer à toute la base de données et créer 24000 fichiers XML. Ce processus sera réalisé sur 30 serveurs à la fois, il durera plusieurs jours et utilisera beaucoup de mémoire.

- Entraînement du modèle. Une fois que les données sont sous un format exploitable, la suite consiste à entraîner le modèle à partir de ces données grâce à la méthode d'apprentissage profond (Deep Learning)

# Conclusion :

Ce stage m'a énormément appris et fait grandir sur le plan professionnel comme personnel.

Professionnellement il m'a permis de réaliser un véritable travail dans une durée limitée. A travers ce travail j'ai pu me rendre compte que mes connaissances en informatique n'étaient pas suffisantes, mais grâce aux bases de programmation acquises au cours du DUT Statistique et informatique décisionnel, je n'ai eu aucun problème à aller chercher des informations et surtout comprendre de nouveaux outils de travail. Ce stage m'a aussi permis de suivre un projet très intéressant avec un véritable enjeu. Ce travail en équipe sur un projet vient directement compléter et enrichir l'expérience du projet en entreprise effectué à l'IUT. Surtout, j'ai eu la chance d'avoir au moins trois personnes à mes côtés pour m'accompagner et m'aider dans mon travail. Toutefois, pour surmonter les blocages que j'ai pu rencontrer, j'ai dû apprendre à poser les bonnes questions et exposer mes difficultés de manière claires et précises.

Sur le plan personnel, J'ai pu découvrir un milieu que je ne connaissais pas et qui m'a beaucoup plus, la Recherche. Même si, à proprement parler je n'ai pas fait de recherche, j'ai pu assister à la présentation de plusieurs exposés par des doctorants mais aussi d'autres projets réalisés par des chercheurs de l'IRD à travers le monde. Les sujets étaient tous très variés et passionnants. Et j'ai trouvé très gratifiant d'apporter ma modeste contribution à un projet qui permettra, je l'espère, à améliorer la prise en charge des maladies complexes comme les événements de Torsade de Pointes.

Cette expérience professionnelle a confirmé mon choix qui était de continuer mes études dans le milieu du Décisionnel et de la Statistique tout en l'appliquant au domaine médical, et peut être un jour faire de la recherche.

# Sitographie :

IRD : https://www.ird.fr/node/8

UMMISCO : https://www.ummisco.fr

Langage : XML : Vidéo : https://www.youtube.com/watch?v=8FuJEwEmb-0

Parsing XML : https://pub.phyks.me/sdz/sdz/dom-parser-du-xml-l-exemple-du-zcode.html

Linux : -https://lea-linux.org/
-clé public/privé (accès au serveur): https://www.digitalocean.com/community/tutorials/how-to-set-up-ssh-keys-on-ubuntu-1804-fr

ECG : https://www.sante-sur-le-net.com/maladies/examens-medicaux/electrocardiogramme-ecg/

GitHub: http://codeur-pro.fr/wp-content/uploads/2018/07/aide_m%C3%A9moire_git.pdf

Librairie en général : https://docs.python.org/fr/

Résolution de problèmes de codages : https://stackoverflow.com/

Format HDF5 : regroupement de données https://deusyss.developpez.com/tutoriels/Python/hdf5/
https://perso.liris.cnrs.fr/martial.tola/presentations/hdf5

Programmation orienté objet :
Web formation : https://www.youtube.com/watch?v=YSKPyAE_w_0
classes et attributs : https://www.youtube.com/watch?v=91dPooHyNIo
https://courspython.com/classes-et-objets.html

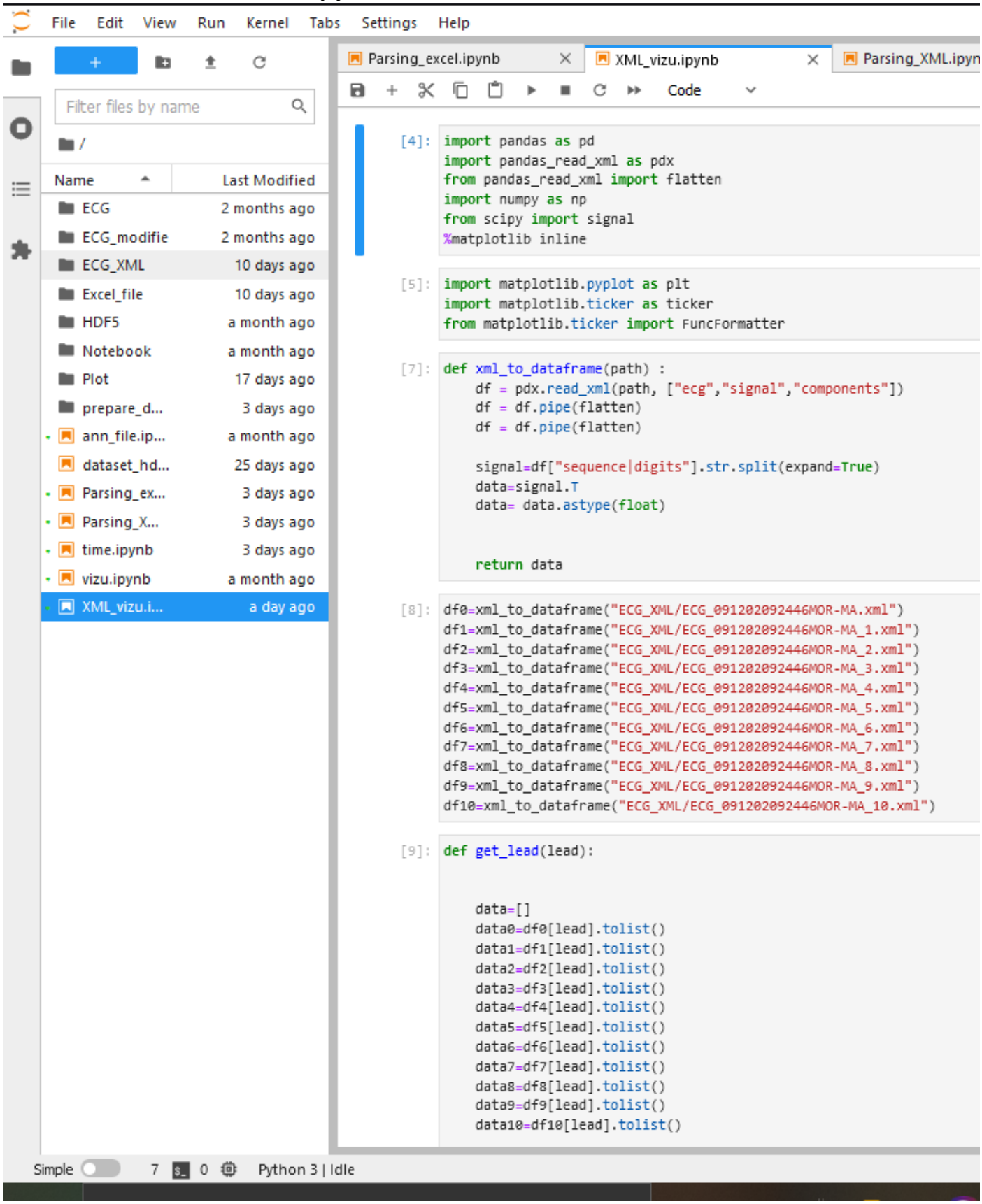
# Référence :

1. Pandas version 1.2.5 : https://pandas.pydata.org

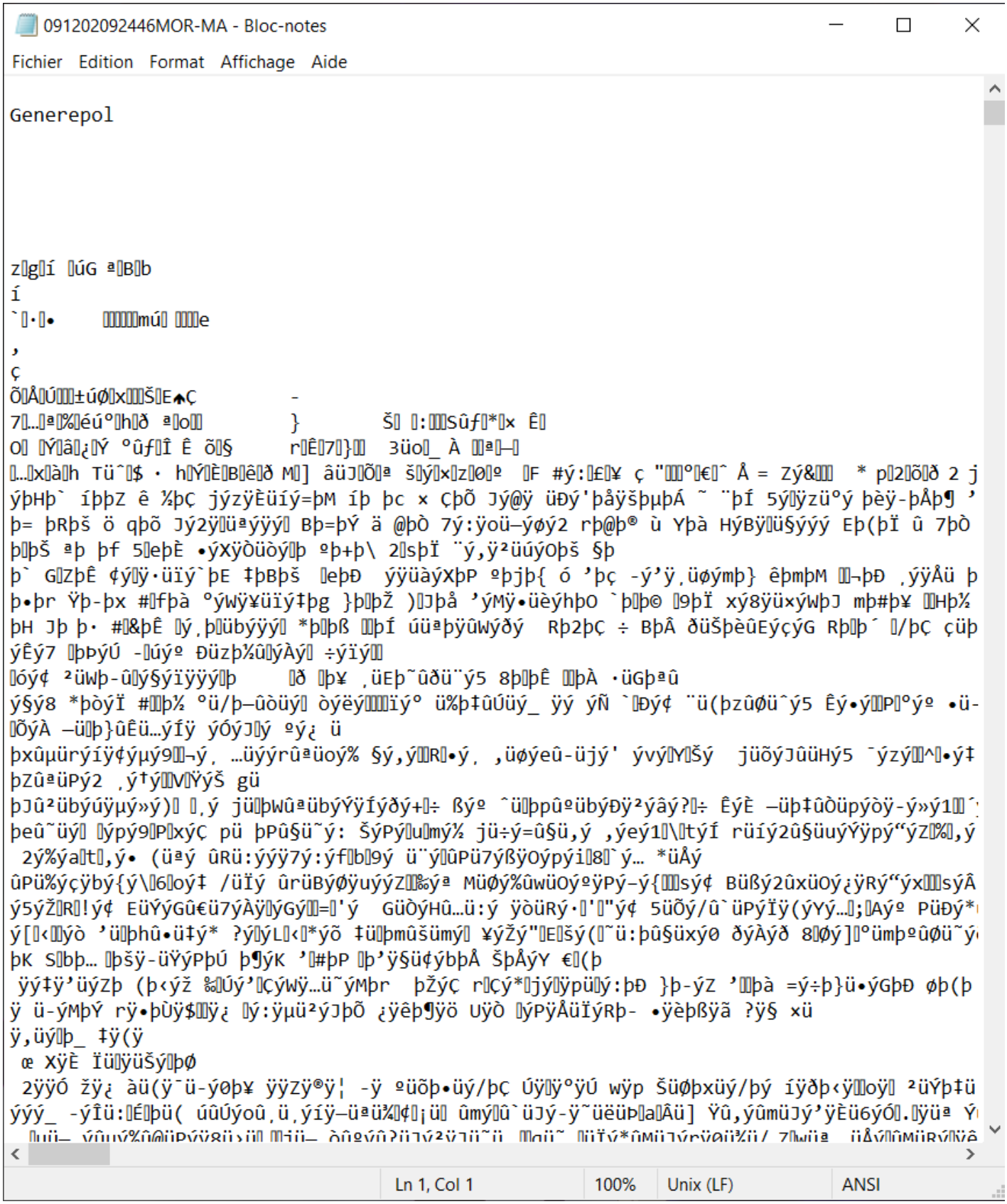
2. read.xml :
https://pandas.pydata.org/pandas-docs/dev/reference/api/pandas.read_xml.html

# Annexes :

## Annexe 1 : Environnement Jupyter Lab :



## Annexe 2 : Aperçu du fichier ECG a parser :



**Annexe 3 : Documentation format ISHNE**

# The ISHNE Holter Standard Output File Format

Fabio Badilini, Ph.D., for the ISHNE Standard Output Format Task Force

*From the Hôpital Lariboisière, Paris, France*

## Introduction

Standard Output Format for Digital Holter Data is a single file structurally organized in a header followed by a (larger) data block containing all stored electrocardiographic (ECG) digital samples. The format of this file is the outcome of several meetings with different manufacturers and it aims to maximize the ratio between simplicity and flexibility.

A freeware viewer of files in Standard Output Format will be made available by the end of Summer 1998 for those who may be interested.

## CONVENTIONS ON DATA TYPES

In this document, the following data type conventions will be adopted:

short int → 2 bytes
long int → 4 bytes
unsigned int → 2 bytes
char → 1 byte

All string are null-terminated (last character is the ASCII value 0).

## MAGIC NUMBER

A "magic number," which will consist of a predefined string of characters, will be inserted at the very beginning of the file. This will allow for a quick verification that the file referenced is indeed in ISHNE format. The magic number will consist of the string of the eight characters ISHNE1.0.

## CHECKSUM

The two bytes following the magic number will be a CRC-CCITT checksum calculated over the complete header (fixed-length and variable-length blocks, see next Section). The listing of the algorithm for calculation of this checksum is provided in the Appendix B.

## HEADER

The header will start at the 11th byte of the file, i.e., immediately following the checksum. It will consist of a fixed-length block (512 bytes) and a variable-length block reserved for freestyle general comments. The fixed-length block will come first and one of its fields will indicate the size and offset of the variable-length block. The variable-length block will consist simply of a stream of ASCII (extended set of 256 characters) characters that any user or manufacturer will use according to his needs.

The main goal of the header is to provide all necessary information on the associated ECG file. Beat annotations are not considered (they may be the target of a future task force). Consequently, issues such as paced versus nonpaced beats, exact localization of noise, or lead-intermittent regions are not involved. Nonetheless, the information on whether or not a pacemaker is present or whether or not a lead is noisy will be given.

## HEADER SIZE

The global size of the header will depend upon the size of the variable-length block (General comment). Sizes of this block will be specified at the beginning of the fixed-length (512 bytes) block.

### ECG Block Size

The total size of the ECG will depend on the number of leads and on the sampling rate (see description in ECG raw data paragraph). Size (in number of samples) and offset (in bytes) of this block

*Address for reprints: Fabio Badilini, Ph.D., Hôpital Lariboisière, Service du Pr. Coumel, 2 rue Ambroise Paré, 75010 Paris, France. Fax: 33-1-49-95-84-39; E-mail: badilini@aol.com*

**Table 3.**

| Description | Data Type | No. of Bytes |
|---|---|---|
| Size (in bytes) of variable length block | long int | 4 |
| Size (in samples) of ECG | long int | 4 |
| Offset of variable length block (in bytes from beginning of file, i.e., 8 + 2 + 512 = 522) | long int | 4 |
| Offset of ECG block (in bytes from beginning of file) | long int | 4 |
| Version of the file | short int | 2 |
| Subject First Name | char[40] | 40 |
| Subject Last Name | char[40] | 40 |
| Subject ID | char[20] | 20 |
| Subject Sex (0: unknown, 1: male, 2: female) | short int | 2 |
| Race (0: unknown, 1 Caucasian, 2 Black, 3 Oriental, 4–9 Reserved) | short int | 2 |
| Date of Birth (European: day, month, year) | 3 short int | 6 |
| Date of recording (European) | 3 short int | 6 |
| Date of creation of Output file (European) | 3 short int | 6 |
| Start time (European: hour [0–23], min, sec) | 3 short int | 6 |
| Number of stored leads | short int | 2 |
| Lead specification (see included Table) | 12 short int | 24 |
| Lead quality (see included Table) | 12 short int | 24 |
| Amplitude resolution in integer no. of nV | 12 short int | 24 |
| Pacemaker code (see text for description) | short int | 2 |
| Type of recorder (either analog or digital) | char[40] | 40 |
| Sampling rate (in hertz) | short int | 2 |
| Proprietary of ECG (if any) | char[80] | 80 |
| Copyright and restriction of diffusion (if any) | char[80] | 80 |
| Reserved | char[88] | 88 |

lead is stored) is fixed in a way "convenient" to all manufacturers. By "convenient" we intend a format allowing the lossless encoding of all present (and possibly future) Holter digital ECGs.

## Format of One ECG Sample

The storage size of one ECG sample has been fixed to two bytes. Data will be stored in the signed format with digital 0 matching 0 mV; most significant bit is "dedicated" to the sign and the range of stored values covers the interval from −32,768 to +32767. Negative values will be stored in a two-complement way. All two-byte samples will be stored in little-endian form (LSB first).

| | | |
|---|---|---|
| | ch1, 1st sample | 2 bytes |
| | ch2, 1st sample | |
| | ch3, 1st sample | |
| | ..... | |
| | chn, 1st sample | |
| | ch1, 2nd sample | |
| | ch2, 2nd sample | |
| | ch3, 2nd sample | |
| | ..... | |
| | chn, 2nd sample | |
| | ch1, 3rd sample | |
| | ..... | |
| | ..... | |

## Storing Order

The ECG block will follow the header block with samples stored multiplexed, as shown in Figure 1, where the number n of leads actually stored (and their identification) will be specified in the header block.

## Use of a Specific Sample Value to Indicate Lead Fault

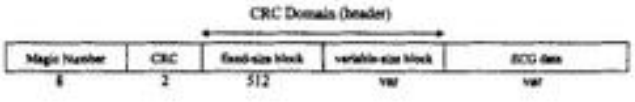
Some members of the group has proposed the use of a reserved sample value (e.g., −32,768 or 0×8,000) to indicate lead fault. This would permit us to overcome the drawbacks of AHA and MIT databases.

## Size of Standard Output File

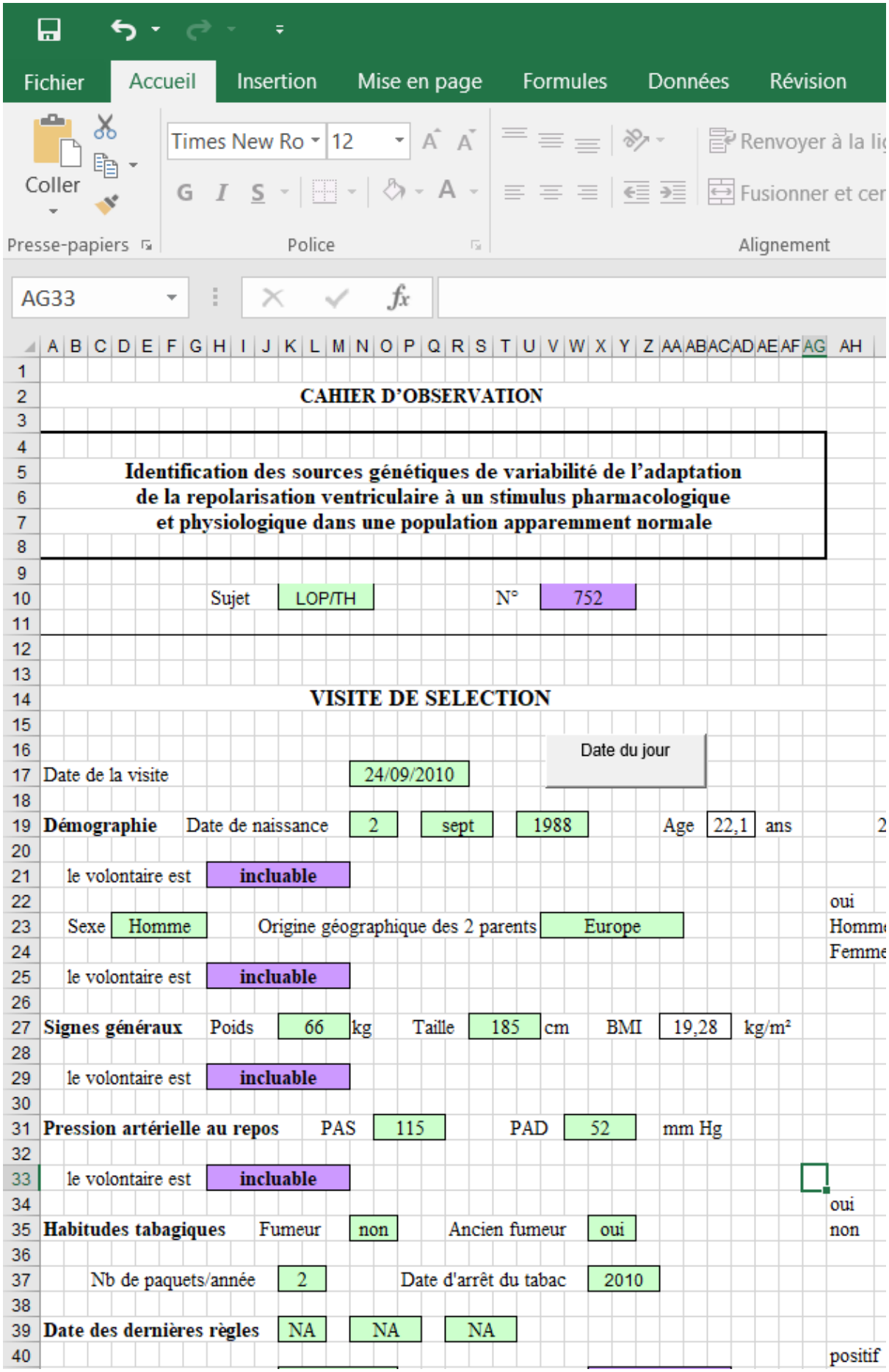
For a fixed time length, this will depend on sampling frequency and number of channels stored. For example (neglecting the header block), a 24-hour ECG with three leads will be ~66 MB at 128 Hz, ~103 MB at 200 Hz, and ~206 MB at 400 Hz. A 24-hour ECG with two leads will be ~44 MB at 128 Hz, ~69 MB at 200 Hz, and ~138 MB at 400 Hz. In some instances, these files may be "redundant," i.e., the same information could have been stored in a smaller space (think of those who store samples in 8 or 10 bits encapsulated). This is somewhat the price paid to be able to store all current formats without any loss of information.

# CONCLUSION

The following schema (Fig. 2) summarizes the sequential structure of the Standard Output File in



## Annexe 4 : Aperçu du fichier Excel



Fichier | Accueil | Insertion | Mise en page | Formules | Données | Révision

Coller | Presse-papiers | Police | Alignement

AG33

**CAHIER D'OBSERVATION**

**Identification des sources génétiques de variabilité de l'adaptation de la repolarisation ventriculaire à un stimulus pharmacologique et physiologique dans une population apparemment normale**

Sujet LOP/TH N° 752

**VISITE DE SELECTION**

Date du jour

Date de la visite 24/09/2010

**Démographie** Date de naissance 2 sept 1988 Age 22,1 ans

le volontaire est **incluable**

Sexe Homme Origine géographique des 2 parents Europe

le volontaire est **incluable**

**Signes généraux** Poids 66 kg Taille 185 cm BMI 19,28 kg/m²

le volontaire est **incluable**

**Pression artérielle au repos** PAS 115 PAD 52 mm Hg

le volontaire est **incluable**

**Habitudes tabagiques** Fumeur non Ancien fumeur oui

Nb de paquets/année 2 Date d'arrêt du tabac 2010

**Date des dernières règles** NA NA NA

oui
Homme
Femme
oui
non
positif

## Annexe 5 : Fonction de visualisation

[16]:
```
def plot_ecg(dico,start, end, title, couleur=True) :
    start1=start*500
    end1=end*500
    lead=["I", "II", "III", "aVR", "aVL", "aVF", "V1", "V2", "V3", "V4", "V5", "V6"]
    long=len(data.keys())

    def time(x, pos):
        return '%1.0f sec' % (x*1/500)
    formatter = FuncFormatter(time)

    color = plt.rcParams["axes.prop_cycle"]()
    fig, ax=plt.subplots(long, sharex=True,figsize=(10,20))


    lowpass=100
    highpass=0.01

    a, b = signal.butter(6,(highpass, lowpass), btype="bandpass", analog=True)
    plt.suptitle(str(title))


    i=0
    while i<long :

        if couleur==True :
            c = next(color)["color"]
        else :
            c="black"

        ax[i].plot(dico[lead[i]][start1:end1],c=c)

        ax[i].set_xlabel("ECG Leads "+ lead[i])
        ax[i].xaxis.set_major_formatter(formatter)
        i+=1
```

## Annexe 6 : Visualisation zoomer sur une zone de l'ECG