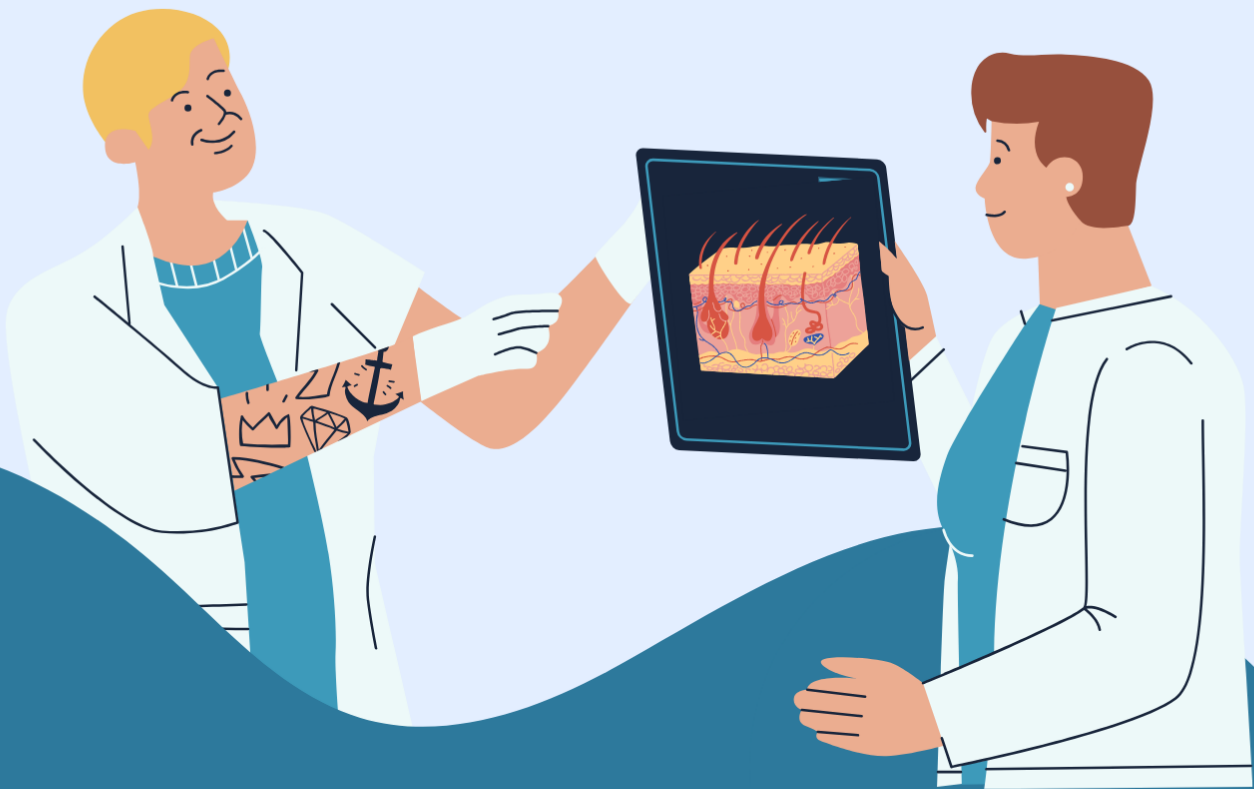


Etude de l'impact du stress et de la fatigue sur la performance

ARCHITECTURE TECHNIQUE



AUTEURS :

AURÉLIE	DESNOYER
LUCIE	KASPERCZYK
INDIRA	LAVOCAT
JACQUES	MEYER
MAEVA	MONTAGNEUX
AYA	TOUNSI

ENSEIGNANTS :

MME. FAIZA BELBACHIR
M. GUILHERME MEDEIROS MACHADO
M. AAKASH SONI

Nous attestons que ce travail est original, qu'il est le fruit d'un travail commun au groupe et qu'il a été rédigé de manière autonome.

TABLE DES MATIÈRES

I.	Présentation du sujet.....	3
II.	Description du système.....	4
1.	Présentation générale.....	4
a.	Architecture fonctionnelle.....	4
b.	Architecture matérielle	5
c.	Architecture logicielle	6
2.	Modèles de données	7
a.	Acquisition des données	8
b.	Traitement des données.....	8
c.	Analyse des données	10
d.	Modèle final.....	14
III.	Gestion des erreurs et tolérance aux pannes	15
IV.	Evolutivité et extensibilité.....	15
V.	Normes et conventions	16
VI.	Conclusion	16
VII.	Annexes	17
VIII.	Sources	17

TABLE DES FIGURES

Figure 1 :	Vue d'ensemble du système.....	3
Figure 2 :	Vue d'ensemble du système.....	4
Figure 3 :	Schéma du modèle final.....	8
Figure 4 :	Graphique de la durée d'enregistrement de l'EDA par étudiant	8
Figure 5 :	Graphique de comparaison de SCR_AUC par groupe de stress.....	10
Figure 6 :	Graphique de comparaison de SCR_AUC par groupe de de notes.....	11
Figure 7 :	Graphique de comparaison de Var_AUC par groupe de fatigue	12
Figure 8 :	Graphique de comparaison de Var_AUC par groupe de de notes	12
Figure 9 :	Graphique du nombre d'hésitations par étudiant ayant une note entre 0 et 10	13
Figure 10 :	Graphique du nombre d'hésitations par étudiant ayant une note entre 10 et 20.....	13
Figure 11 :	Schéma du modèle final	14
Figure 12 :	Courbe d'apprentissage du modèle de MLP pour la prédiction de notes	15

TABLE DES TABLEAUX

Tableau 1 :	Scores de l'augmentation de données	14
Tableau 2 :	Métriques d'évaluation du modèle « Multi Layer Perceptron » (MLP).....	14

I. PRÉSENTATION DU SUJET

L'objectif de ce document est de servir de guide tout au long de ce projet. Il vise à définir un cadre clair pour la conception, la mise en œuvre, la maintenance et l'évolution du système technique. Notre projet de recherche sur l'activité électrodermale et le stress s'est déroulé en plusieurs jalons phares, chacun prenant appui sur des technologies diverses et variées. Ainsi, ci-dessous seront expliqués les principes, les directives techniques, le développement et la mise en œuvre du système.

Comme expliqué dans l'état de l'art [1] et le cahier de laboratoire [2], l'objectif de notre projet est de prédire la note des étudiants pour répondre à notre problématique d'étude de l'impact des états cognitifs de stress et fatigue mentale sur la performance académique. Pour mesurer cela, nous collectons plusieurs types de données :

- Données physiologiques : l'activité électrodermale (EDA) ;
- Données audios : enregistrement vocaux lors de l'examen ;
- Données contextuelles : questionnaire de fatigue, [questionnaire préliminaire](#) d'autoévaluation des connaissances, [questionnaire final](#) de ressenti après expérience, réponses écrites de l'examen.

Afin de récolter ces données, nous avons réalisé des expériences sur des étudiants volontaires. Comme expliqué dans le protocole expérimental [3], avec l'aide de nos référents nous avons mis en place une expérience d'examen blanc de *Machine Learning* avec des étudiants ING 4 en BIG DATA & IA. Ces expériences se sont déroulées à une semaine de leur partiel dans cette matière. A l'issue de ces expériences, nous avons pu collecter des données pour ensuite les traiter et les analyser. Ces trois étapes clés peuvent être représentées par le diagramme suivant qui propose une vue d'ensemble du système.

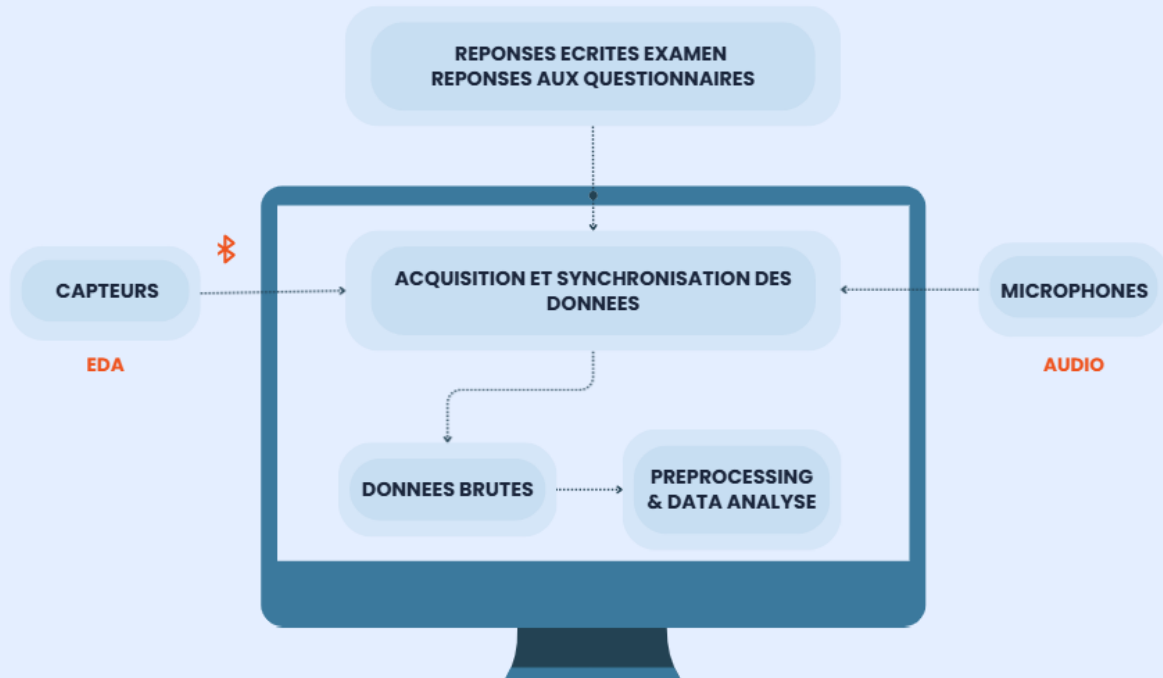


Figure 1 : Vue d'ensemble du système

II. DESCRIPTION DU SYSTÈME

1. PRÉSENTATION GÉNÉRALE

Afin de réaliser notre projet, nous avons mis au point un système qui permet de collecter, traiter et analyser des données.

Les objectifs fonctionnels de notre projet s'orientent autour de la possibilité d'extraire des données, de les analyser et de les traiter avec le matériel et les logiciels choisis. Les objectifs non fonctionnels se concentrent eux sur les aspects de performance du système et de la sécurité des données.

A. ARCHITECTURE FONCTIONNELLE

Comme expliqué précédemment, notre système a pour fonction de collecter, traiter et analyser les données d'EDA, audios et contextuelles. Chacune de ces fonctions sont reliées entre elles. En effet, il est impossible de commencer à traiter les données si elles n'ont pas été préalablement collectées. Ainsi, pour pouvoir interagir correctement entre ces fonctions, nous avons créé un drive qui nous permet de stocker les données à chaque étape. Ce drive est privé car nous collectons des données de santé considérées comme étant des données sensibles. D'autre part, pour les étapes de traitement et d'analyse des données, nous avons créé un serveur privé GitHub afin d'avoir une interaction plus efficace.

I. COLLECTE DES DONNÉES

Dans un premier temps, nous avons collecté les données contextuelles des participants. Lors de leur passage, ils ont reçu un [questionnaire préliminaire](#) et un [questionnaire après expérience](#) en ligne permettant de jauger leur niveau de fatigue et de stress. Ce questionnaire permet également de recueillir leur ressenti sur la qualité de leurs révisions à l'examen. Pendant l'examen écrit, nous avons recueilli leurs réponses écrites sur papiers que nous avons ensuite corrigées. A la fin de l'examen, les participants ont de nouveau complété un questionnaire en ligne pour donner leur ressenti après expérience sur leur niveau de stress, de fatigue, et de révision. pour donner leur ressenti après expérience sur leur niveau de stress, de fatigue, et de révision.

Nous utilisons le logiciel OpenSignals pour recueillir en temps réel les signaux physiologiques d'EDA provenant du capteur. Il reçoit les données du capteur d'EDA Psychobit et les affiche instantanément, fournissant ainsi un retour d'information immédiat sur l'activité électrodermale des participants. En parallèle, nous avons collectées les données audio grâce au microphone de téléphone.








Figure 2 : Vue d'ensemble du système

Toutes les données collectées ont été anonymisées en raison de leur sensibilité et stockées sur un drive privé.

B. ARCHITECTURE MATÉRIELLE

II. DESCRIPTION DU MATÉRIEL

Voici la liste exhaustive du matériel que nous avons utilisé dans notre projet. Ce tableau est issu du Protocole expérimental [3].

MATERIEL	IMAGES
BOITIER ET CAPTEUR EDA PSYCHO BIT ET ELECTRODES POUR MESURER L'ACTIVITE ELECTRODERMALE DES INDIVIDUS	
ORDINATEUR + CHARGEUR POUR AFFICHER LES SUJETS ET RECUPERER LES DONNEES DU CAPTEUR + MICROPHONE	
QUESTIONNAIRES NUMERIQUES POWERPOINT	
LOGICIELS : OPENSIGNAL, VISUAL STUDIO CODE, ANACONDA, ELAN	
LANGAGE DE PROGRAMMATION : PYTHON ET BIBLIOTHÈQUES ASSOCIÉES	

III. SPÉCIFICATION DES SERVEURS

Comme en témoigne la description du matériel, nous avons utilisé GitHub pour réaliser la partie programmation de notre projet. Néanmoins, un projet GitHub a une capacité de stockage maximale de 2Go. Cela étant relativement faible pour nos besoins, nous avons décidé de stocker nos fichiers de données sur un drive Microsoft afin d'être plus flexibles.

IV. JUSTIFICATION DES CHOIX MATÉRIELS

Nous avons opté pour le capteur Psychobit en raison de sa réputation de fournir des mesures précises de l'activité électrodermale. La fiabilité des données est cruciale pour nos recherches ou nos applications cliniques. Un autre facteur déterminant dans notre choix du capteur EDA Psychobit est sa capacité à capter non seulement l'activité électrodermale, mais également

d'autres signaux physiologiques tels que l'ECG (électrocardiogramme). Si nous allions plus loin, nous pourrions analyser ces autres signaux physiologiques. En intégrant plusieurs types de signaux dans notre étude, cela permettrait d'enrichir nos analyses et nos interprétations pour comprendre les interactions complexes entre les processus émotionnels, cognitifs et physiologiques. Ainsi, le capteur Psychobit est un outil polyvalent qui répond à notre besoin.

Nous avons besoin d'un ordinateur pour réaliser les expériences avec les sujets, afin de présenter les tests et récupérer les données du capteur. De plus, l'ordinateur doit être compatible avec nos logiciels, nécessitant idéalement le système d'exploitation Windows pour assurer la compatibilité et le bon fonctionnement de l'ensemble. L'utilisation du Bluetooth permet une connectivité sans fil avec le capteur. De plus, un chargeur assure une alimentation continue, évitant ainsi les interruptions imprévues qui pourraient compromettre le déroulement de l'expérience.

Afin d'enregistrer les réponses et donc la voix des participants, nous avons utilisé le microphone de nos téléphones. Étant donné qu'ils sont faciles et fiables à utiliser, cela nous a permis de récolter l'entièreté des audios avec une qualité adéquate. De plus, nous avons effectué un transfert rapide des fichiers vers le drive.

C. ARCHITECTURE LOGICIELLE

I. DESCRIPTION DES LOGICIELS

- **OpenSignals**

Pour capter et analyser les signaux physiologiques de nos participants, nous avons utilisé OpenSignals en conjonction avec notre capteur d'activité électrodermale (EDA). OpenSignals est un logiciel polyvalent qui permet l'acquisition et la visualisation en temps réel de signaux physiologiques, offrant une gamme de fonctionnalités pour l'analyse avancée des données biométriques. En intégrant notre capteur d'EDA à OpenSignals, nous avons pu surveiller de manière précise les réponses physiologiques de nos participants tout au long de nos expériences.

- **Visual Studio Code**

Visual Studio Code est un éditeur de code développé par Microsoft pour Windows, Linux et Mac. Cet éditeur puissant et polyvalent inclut plusieurs langages de programmation notamment Python, langage utilisé dans notre projet. Il permet également de travailler la gestion de données, indispensable à notre projet.

- **ELAN**

ELAN est un logiciel développé par le Max Planck Institute for Psycholinguistics pour l'annotation et l'analyse de données audio et vidéo. Il permet aux chercheurs de transcrire, d'annoter et de visualiser des médias multimodaux, tels que des vidéos et des enregistrements audios, avec une grande précision.

- **Git et GitHub**

Git est un logiciel de gestion de version qui permet de conserver l'historique des modifications effectuées sur un projet. Cela permet de revenir à une ancienne version du projet en cas de problème. Les fichiers d'un projet sont copiés sur un serveur central, un dépôt. GitHub permet l'hébergement de ces dépôts de façon public ou privée. Le statut public permet à n'importe quel utilisateur d'avoir accès aux différents fichiers du projet et de travailler dessus tandis que le statut privé n'autorise l'accès qu'aux personnes autorisées.

Dans notre cas, nous avons créé un projet ayant le statut privé car nous sommes dans un projet de recherche et nous utilisons des données de santé sensibles. Ainsi, uniquement les personnes de notre projet et nos référents ont accès au contenu des fichiers.

- **Microsoft OneDrive**

Microsoft OneDrive est un espace de stockage en ligne qui permet d'enregistrer de nombreux fichiers, photos et vidéos. Nous avons sélectionné cet espace de stockage pour ses nombreuses fonctionnalités :

- Sauvegarde et protection des fichiers ;
- Accès aux fichiers en tout lieu ;
- Partage en privé des fichiers ;
- Accès aux applications Office.

Par conséquent, nous avons organisé un drive partagé de façon à pouvoir stocker et modifier nos documents et nos données.

V. INTÉGRATION DE BIBLIOTHÈQUES OU FRAMEWORKS

- **Anaconda**

Anaconda est une distribution pour Python utilisée dans la gestion de données et le *Machine Learning*. Elle fournit l'accès à plusieurs outils notamment le langage de programmation Python mais également l'accès à des bibliothèques telles que Pandas, NumPy, etc.

- **Neurokit2**

Comme expliqué précédemment, nous manipulons des données de santé. Ainsi pour pouvoir traiter les données et mieux les analyser nous avons utilisé la bibliothèque Neurokit2 qui est adaptée aux signaux physiologiques et particulièrement l'EDA.

- **NumPy et Pandas**

La bibliothèque NumPy nous permet de faire des calculs numériques et également de gérer des tableaux de données. Pandas est une librairie basée sur NumPy qui permet de manipuler et lire facilement des données structurées, dans notre cas des tableaux.

- **Scikit Learn**

Scikit Learn est une bibliothèque qui permet de réaliser des algorithmes de *Machine Learning*. Nous avons pu notamment utiliser les algorithmes Random Forest, Gradient Boosting, etc. Cette librairie permet également d'évaluer la qualité et la précision des modèles.

- **Wav2Vec2**

Wav2Vec2 est un modèle de Deep Learning développé par Facebook pour la reconnaissance de la parole. Il est particulièrement efficace car il peut obtenir de bons résultats même avec une quantité limitée de données étiquetées. Il est utilisé pour des tâches telles que la transcription de la parole, la reconnaissance des émotions dans la voix, et bien d'autres applications liées à l'audio.

- **Multi Layers Perceptron**

Un perceptron multicouche (MLP) est un type de réseau de neurones artificiels utilisé en apprentissage automatique. Il est constitué de plusieurs couches de neurones : une couche d'entrée, une ou plusieurs couches cachées et une couche de sortie. Chaque neurone est connecté à tous les neurones de la couche suivante. Les poids et biais associés à ces connexions sont ajustés lors de l'apprentissage pour améliorer les performances du modèle. Les MLP sont utilisés pour des tâches variées comme la classification et la régression.

2. MODÈLES DE DONNÉES

A. ACQUISITION DES DONNÉES

Les données sont dans un premier temps, acquises selon le protocole [3]. L'objectif de notre modèle final de données est d'utiliser les données contextuelles, données récoltées grâce aux questionnaires, et les données physiologiques (EDA et audio), afin de prédire la note d'un étudiant à un examen blanc comme illustré ci-dessous :



Figure 3 : Schéma du modèle final

B. TRAITEMENT DES DONNÉES

La collecte des données nous a permis d'effectuer un premier nettoyage et traitement de données. Ce dernier a pour objectif d'extraire du signal les caractéristiques adaptées à notre étude. Ainsi, nous avons réalisé les parties suivantes :

- Extraction des features (caractéristiques) du jeu de données vocales (audio),
- Extraction des features (caractéristiques) du jeu de données électrodermales (EDA),
- Étude statistique de la pertinence des features extraites,

DONNÉES EDA

Nous avons dans un premier temps vérifié que nos fichiers de données brutes soient conformes à certains paramètres :

- Fréquence d'échantillonnage : 1 kHz
- Type de données : numériques
- Durée d'enregistrement du signal : 36 min
- Amplitude du signal : μS (microSiemens)

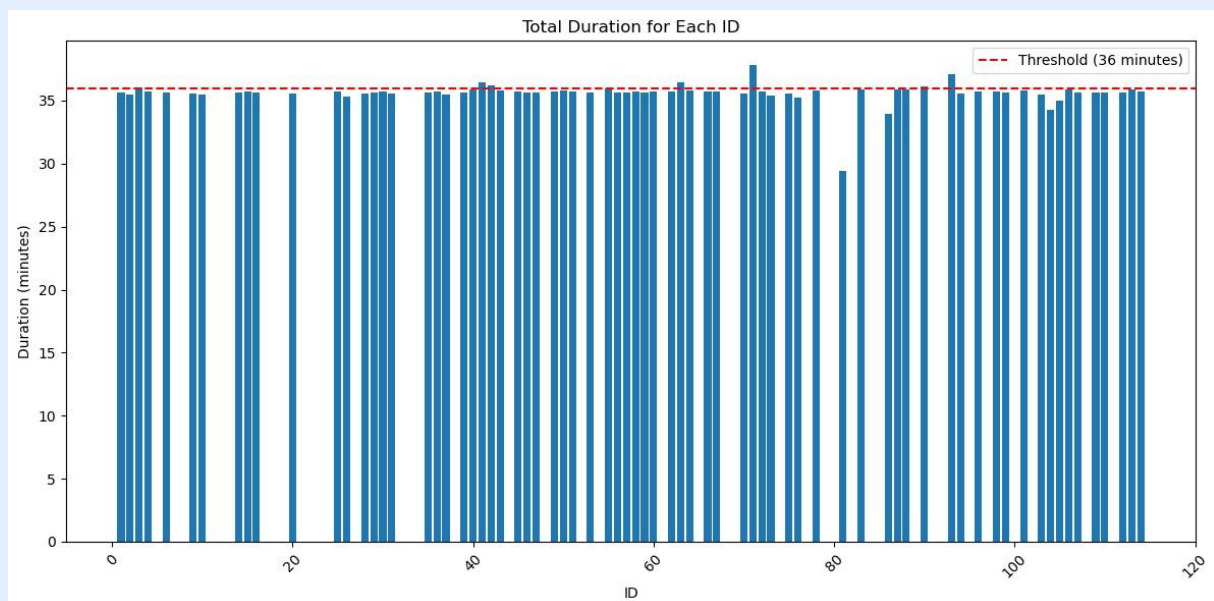


Figure 4 : Graphique de la durée d'enregistrement de l'EDA par étudiant

Par la suite, nous avons procédé au traitement du signal qui a pour objectif de décomposer le signal brut en deux composantes : phasique et tonique [1]. Cela se résume en trois étapes phares :

- Suppression du bruit et lissage du signal :

En appliquant un filtre passe bas de fréquence de coupure 3 Hz et un filtre de Butterworth de 4^{ème} ordre, le bruit du signal est supprimé et le signal est lissé.

- Décomposition du signal nettoyé :

Le signal nettoyé est décomposé en deux composantes phasique et tonique en appliquant un filtre passe haut de fréquence de coupure 0,2 Hz.

- Extraction des pics :

Les pics significatifs, pics dépassant un seuil de 0,0015 μ S, sont extraits.

A l'issue du traitement des données, nous obtenons différents fichiers répertoriant les caractéristiques de l'EDA par individu. Nous avons également schématisé certains résultats obtenus grâce au traitement de données, notamment les différentes composantes du signal d'EDA.

DONNÉES AUDIOS

Le jeu de données audio est particulièrement volumineux, chaque fichier représentant un enregistrement d'écran (vidéo) de 30 minutes par étudiant, incluant à la fois la partie orale et la partie écrite. Cela correspond à une taille de 2 Go par étudiant. Au total, 75 étudiants ont participé à l'expérience, générant ainsi un volume de données considérable de 150 Go à traiter.

Avant de commencer l'analyse de la parole des étudiants, nous avons mené des recherches sur les outils disponibles pour le traitement et l'analyse des caractéristiques audio. Tout d'abord, notre objectif était de déterminer l'état de stress des étudiants à partir de leur voix (stressé ou non-stressé). Pour cela, nous cherchions un modèle pré-entraîné capable de reconnaître les émotions, incluant le stress, afin de réaliser l'annotation des audios des étudiants. Cette annotation serait ensuite utilisée dans notre modèle final.

Cependant nous avons constaté que les technologies dans ce domaine sont encore émergentes, et celles récemment développées sont principalement axées sur des audios en anglais, ce qui ne correspond pas à notre cas.

Face à l'absence d'outils adaptés, nous avons dû développer notre propre solution pour caractériser le stress à partir des données audios des étudiants.

Cette solution se concentrera sur les disfluences vocales présentes dans les enregistrements audios. Cela regroupe les hésitations vocaliques, les marqueurs discursifs, les autocorrections, les répétitions, les tics de langage.

Pour ce faire, nous avons premièrement conçu notre propre algorithme Python pour pré-traiter les données vocales de manière suivante :

- Convertir les fichiers au format mp4 en .wav ;
- Extraire uniquement les segments audios relatifs à l'examen oral (15 premières minutes), en éliminant les parties correspondant à l'examen écrit ;
- Supprimer les périodes où l'étudiant ne parle pas, en fixant un seuil de blanc.

Ce travail préliminaire nous a permis de réduire la taille des fichiers, passant de 2 Go à environ 15 Mo chacun, ce qui représente un gain substantiel en termes d'espace de stockage et de temps d'analyse.

Par la suite, nous avons dû annoter manuellement plusieurs fichiers audios en nous basant sur leurs disfluences vocales. Étant donné que la quantité de données relatives aux autocorrections, répétitions, marqueurs discursifs et tics de langage était limitée, nous avons décidé de nous concentrer uniquement sur les hésitations vocaliques. Pour cela, nous avons utilisé le logiciel ELAN Audio afin de réaliser cette annotation. Ensuite, nous avons entraîné un modèle de Deep Learning, Wav2Vec2, pour reconnaître les hésitations vocaliques sur des données d'entraînement et ainsi pouvoir annoter les données de test.

Ces données audios annotées ont ensuite constitué notre base de données pour les étapes suivantes

C. ANALYSE DES DONNÉES

DONNÉES EDA

Une fois les données traitées, nous avons réalisé une étude statistique sur le lien entre les features de l'EDA, le stress et la fatigue cognitive. Ces features ont préalablement été étudiées dans la littérature scientifique comme expliqué dans notre état de l'art [1].

- **STRESS**

Nous avons étudié plusieurs features d'EDA ayant un lien avec le stress. Pour les explications, nous prendrons le cas de la feature « Skin Conductance Response - Area Under the Curve (SCR AUC) ».

Dans un premier temps, nous avons comparé cette caractéristique par rapport à l'état de stress indiqué par les étudiants dans les questionnaires. Cela nous a permis de constater que les étudiants ayant indiqué être stressés ont une moyenne de SCR AUC plus élevée que les étudiants ayant déclaré être non stressés comme en témoigne le graphique ci-dessous. Cela confirme donc que la caractéristique d'EDA étudiée peut refléter l'état de stress de l'étudiant.

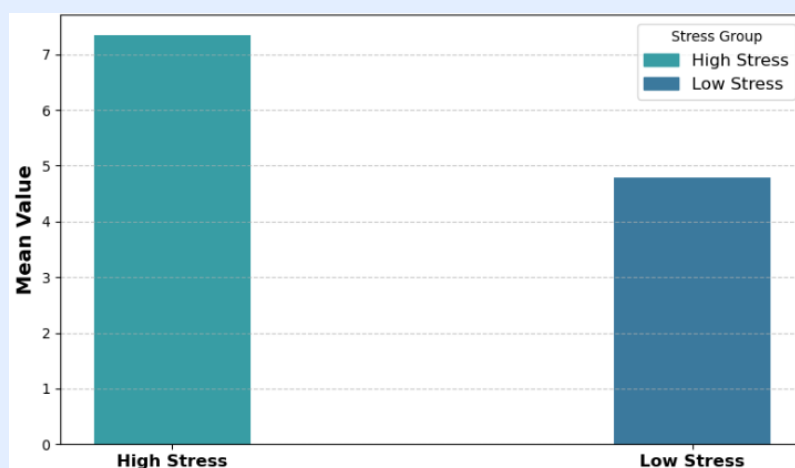


Figure 5 : Graphique de comparaison de SCR_AUC par groupe de stress

Une fois que nous avons étudié les liens entre certaines caractéristiques d'EDA et de stress, nous avons sélectionné les plus adaptées pour notre étude. Pour la suite des explications, nous utilisons la même caractéristique qu'étudiée précédemment (SCR AUC).

Nous avons donc comparé l'état de stress lié à cette caractéristique avec le type d'examen (oral ou écrit cf. protocole expérimental [3]) et par groupe de notes.

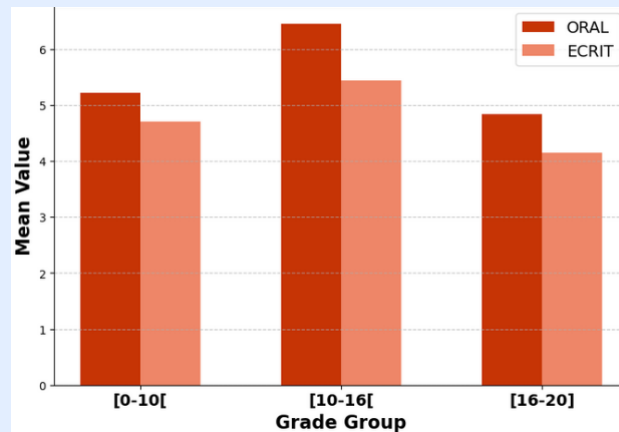


Figure 6: Graphique de comparaison de SCR_AUC par groupe de de notes

Le graphique ci-dessus met en évidence que l'examen oral est plus stressant que l'examen écrit. De plus, on remarque que les étudiants du groupe de notes [10-16[sont globalement les plus stressés tout au long de l'examen. Pour comprendre ces résultats, nous avons fait l'hypothèse que les étudiants du groupe de notes [0-10[, étudiants ayant confirmés être peu, voire pas préparés, sont moins stressés car ils sont conscients de leur manque de connaissance. A l'opposé, les étudiants du groupe de notes [16-20[ayant bien révisé, sont plus sereins au moment de passer l'examen. Les étudiants les plus stressés, appartenant au groupe de note [10-16[, ont confirmé avoir révisé mais ne sont pas totalement confiants dans leurs réponses et leur connaissances, entraînant donc plus de stress.

Par conséquent, le stress impacte certaines caractéristiques d'EDA, et plus globalement, il impacte la performance académique des étudiants.

• FATIGUE

En parallèle, nous avons fait les mêmes analyses avec la fatigue cognitive. Ainsi, nous avons étudié plusieurs caractéristiques de l'EDA ayant un lien avec cet état cognitif. Tout au long des explications suivantes, nous prendrons l'exemple de la « Variance Skin Conductance Response (SCL VAR) », la variance de la composante tonique.

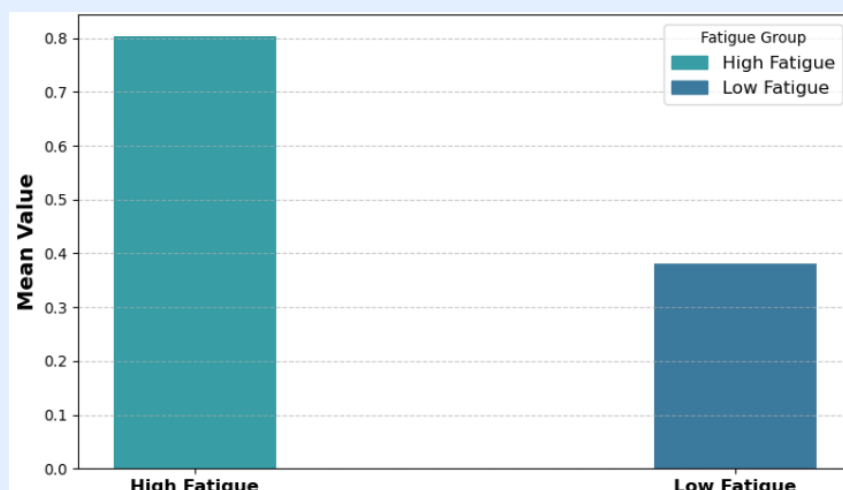


Figure 7: Graphique de comparaison de SCL_VAR par groupe de fatigue

Le graphique ci-dessus illustre la moyenne de Var AUC par rapport à l'état de fatigue déclaré par les étudiants. On remarque que pour les étudiants ayant déclaré être très fatigués, la moyenne de la caractéristique est élevée comparé aux étudiants ayant déclaré être peu fatigués. Cela confirme que la caractéristique étudiée est représentative de l'état de fatigue cognitive.

Par la suite, nous avons étudié la moyenne de SCL VAR par rapport au type d'examen (oral et écrit) ainsi qu'aux groupes de notes comme suit.

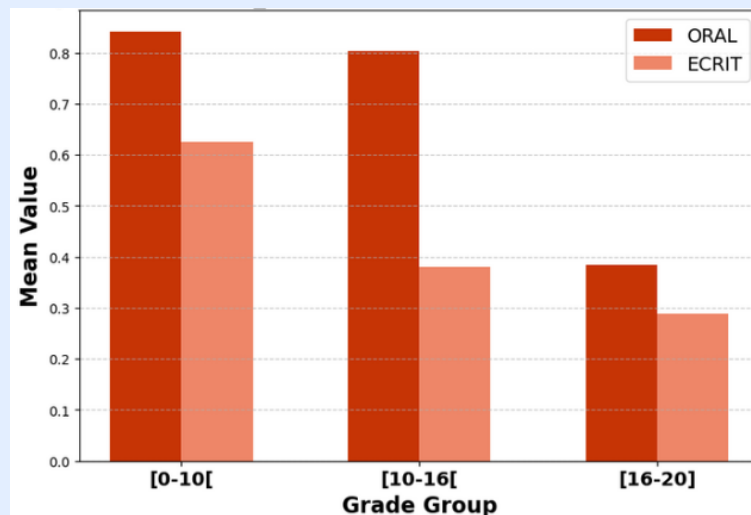


Figure 8: Graphique de comparaison de Var_AUC par groupe de notes

Nous remarquons que l'examen oral est plus fatigant cognitivement que l'examen écrit. De plus, nous remarquons que plus la note est faible, plus l'étudiant est fatigué. Cela peut s'expliquer par le fait que les étudiants ayant moins bien réussi avaient besoin de plus de concentration ce qui peut entraîner une fatigue cognitive plus importante.

Ainsi, la fatigue cognitive impacte certaines caractéristiques d'EDA et impacte plus globalement la performance académique des étudiants.

L'analyse statistique nous a donc permis d'établir un lien solide entre les caractéristiques de l'EDA, le stress, la fatigue cognitive et la performance académique.

DONNÉES AUDIOS

Comme expliqué précédemment, l'étude de données audio en programmation est encore peu développée. Compte tenu de nos recherches et des résultats, nous avons choisi d'utiliser un modèle Wav2Vec2 développé par Facebook, pour étudier ces enregistrements. Ce modèle de Deep Learning est capable d'extraire des représentations riches à partir de données brutes, avec possibilité de fine-tuning (ajustements pour obtenir la meilleure performance) pour s'adapter à des tâches spécifiques comme la détection du stress en français.

Avec la base de données audios précédemment pré-traitée nous avons, grâce à une augmentation de données, obtenu des résultats concluant sur le compte du nombre d'hésitations par audio.

Une fois ces résultats obtenus, nous avons analysé le nombre d'hésitations par rapport à la réussite des étudiants lors de l'examen comme en témoignent les graphiques ci-dessous.

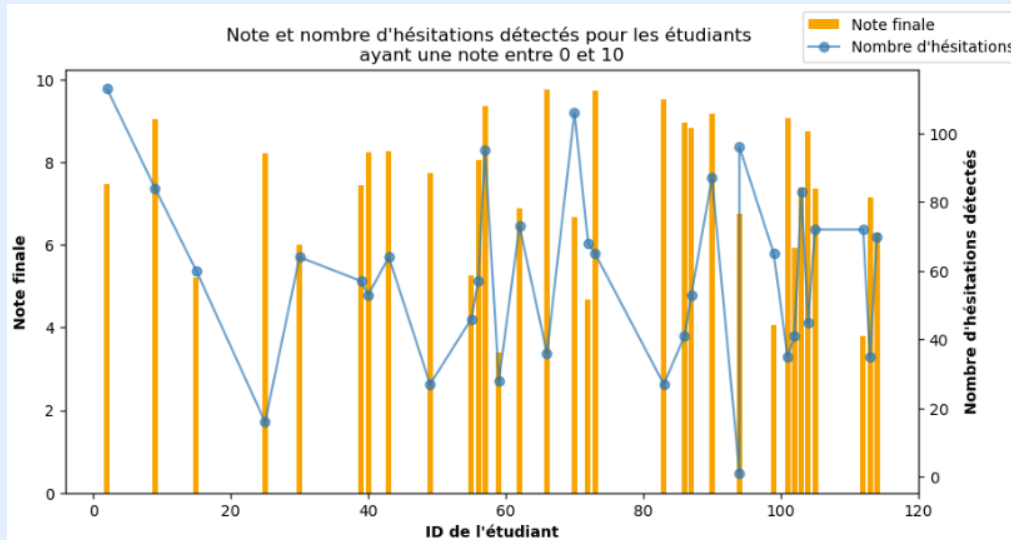


Figure 9: Graphique du nombre d'hésitations par étudiant ayant une note entre 0 et 10

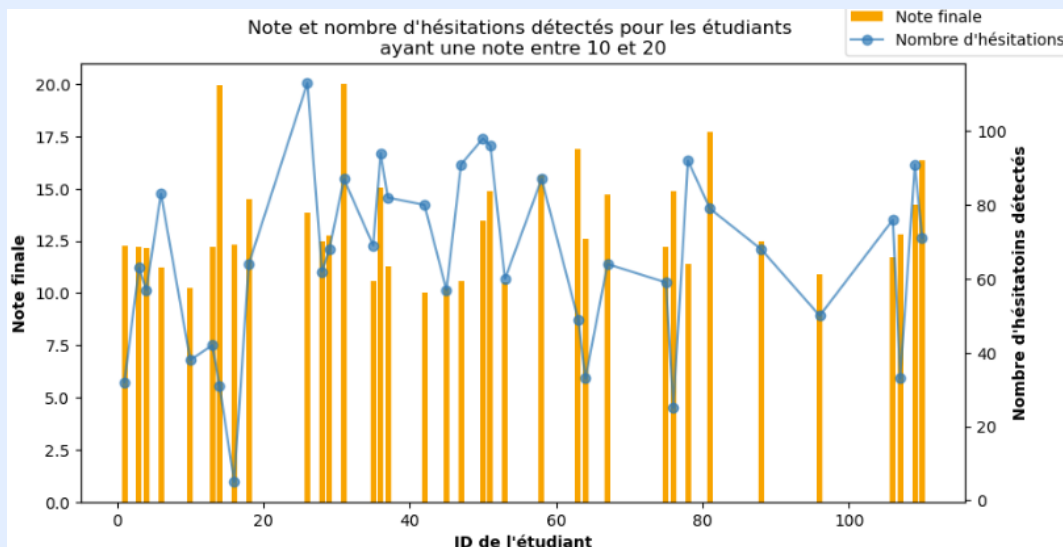


Figure 10: Graphique du nombre d'hésitations par étudiant ayant une note entre 10 et 20

Ces graphiques illustrent bien le fait qu'un étudiant peut avoir une très bonne note tout en ayant un nombre d'hésitations important ou faible. On remarque la même chose pour les étudiants ayant obtenu une note basse. Pour expliquer ces disparités, nous avons formulé plusieurs hypothèses :

- Les étudiants avec des notes faibles n'ont pas répondu à la question, il n'y a donc pas d'hésitation ;
- Les étudiants avec des notes moyennes ou hautes prennent le temps de réfléchir et de formuler leur réponse ce qui peut impliquer un nombre d'hésitation important ;
- Les hésitations peuvent également refléter des tics de langage.

Par conséquent, il n'y a pas de corrélation entre le nombre d'hésitation et la performance académique des étudiants. Grâce à cette analyse, nous avons choisi d'écarter ces données pour notre modèle final.

D. MODÈLE FINAL

La première étape de notre modèle final a été d'augmenter notre jeu de données afin d'avoir un plus gros dataset sur lequel travailler. Ainsi, après cette opération, notre échantillon contient 150 étudiants avec leurs données associées. Afin de mesurer la véracité de notre augmentation de données, nous l'avons analysée comme suit :

Score des distributions des colonnes	81.88%
Score des relations entre les colonnes	85.71%
Score de qualité des données	83.79%

Tableau 1: Scores de l'augmentation de données

Ces scores sont proches de 100% indiquant que l'augmentation des données est optimale. Nous avons donc utilisé ces données dans notre modèle final.

Notre modèle final est un modèle de Machine Learning intitulé « Multi Layers Perceptron (MLP) ». C'est un modèle informatique qui aide à trouver des relations complexes entre des informations d'entrée et de sortie. Il est composé de plusieurs "couches" de neurones artificiels : une pour les entrées, une ou plusieurs couches cachées pour le traitement, et une pour les sorties. Chaque neurone de ces couches est connecté aux neurones de la couche suivante, et ces connexions sont pondérées. Les poids et les biais de ces connexions sont ajustés pendant l'apprentissage pour améliorer les résultats du modèle. Des fonctions spéciales appelées fonctions d'activation permettent au modèle de traiter des relations complexes entre les données.

Les MLP sont très flexibles et peuvent être utilisés pour diverses tâches comme la classification d'images ou la prédiction de valeurs. Dans notre cas, nous l'utilisons pour la prédiction de valeurs.

Pour notre étude, les données d'entrées sont les données contextuelles et les données physiologiques (EDA) pour prédire la note d'un étudiant à un examen blanc.



Figure 11: Schéma du modèle final

Grâce à notre modèle final, nous avons obtenu les résultats suivants :

Erreur quadratique moyenne (MSE)	0.017	Valeur optimale : proche de 0
Ecart quadratique moyen (RMSE)	0.132	Valeur optimale : proche de 0
Coefficient de détermination (R^2)	0.613	Valeur optimale : proche de 1

Tableau 2: Métriques d'évaluation du modèle « Multi Layer Perceptron » (MLP)

La métrique qui nous intéresse particulièrement est R^2 . La valeur optimale de cette se rapproche de 1, signifiant que le modèle explique bien la variance des données. Dans notre cas, $R^2 = 0.614$. Ce résultat est proche de 1 mais il n'est pas le plus optimal. Cela est également visible sur le graphique ci-dessous.

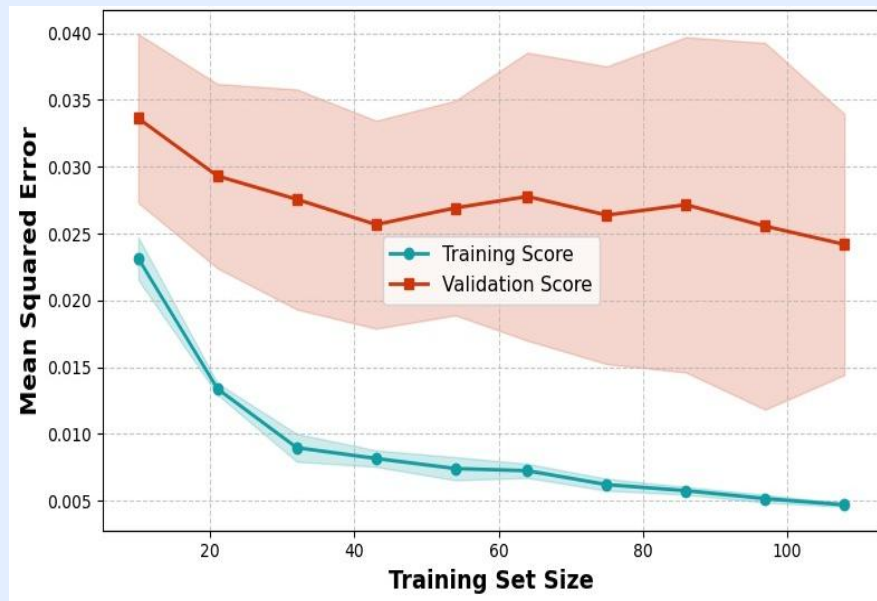


Figure 12: Courbe d'apprentissage du modèle de MLP pour la prédiction de notes

Notre courbe de validation, en rouge suit globalement la même allure que notre courbe d'apprentissage en bleu. Néanmoins, la marge d'erreur reste légèrement importante. Notre modèle est donc suffisamment efficace mais il pourrait être amélioré.

Grâce à notre modèle, **nous sommes capable de déterminer la note d'un étudiant à un examen blanc avec des données physiologiques et contextuelles.**

III. GESTION DES ERREURS ET TOLÉRANCE AUX PANNES

Afin de gérer au mieux les erreurs, nous avons utilisé la stratégie de gestion des erreurs impliquant la détection, la notification et la résolution des erreurs qui se produisent lors de l'exécution du système. Ainsi, avec notre IDE, nous avons la possibilité de déboguer notre code pour la détection et la notification des erreurs. De plus, avec les ressources disponibles en ligne, nous avons pu résoudre les problèmes de compilation.

En cas de problème plus conséquent, nous avons la possibilité d'avoir accès aux anciennes versions de notre projet grâce à Git et GitHub qui nous servent donc de backup en cas de besoin.

IV. EVOLUTIVITÉ ET EXTENSIBILITÉ

Notre système a été conçu avec l'évolutivité et l'extensibilité à l'esprit. Grâce à son architecture modulaire, il peut facilement être étendu pour inclure de nouvelles fonctionnalités ou s'adapter à des besoins changeants. Par exemple, de nouveaux capteurs peuvent être intégrés pour collecter des types de données supplémentaires, ou de nouvelles méthodes d'analyse peuvent être ajoutées pour extraire des informations plus précises à partir des données existantes. De plus, notre système est capable de gérer une augmentation du volume de données sans compromettre ses performances, ce qui le rend idéal pour une utilisation dans des environnements de recherche en constante évolution.

En outre, notre système est suffisamment flexible pour permettre des modifications dans le nombre d'expériences, la population étudiée, et d'autres facteurs. Cela signifie que nous

pouvons élargir notre base de données avec une plus grande diversité de données, ce qui nous permettrait d'aller plus loin dans nos recherches. Par exemple, nous pourrions étudier différents groupes d'âge, des populations avec différentes conditions de santé, ou même effectuer des expériences dans différents environnements. Cette flexibilité nous permet d'adapter notre recherche à nos objectifs spécifiques et d'explorer de nouvelles directions à mesure que nos connaissances et nos ressources augmentent.

V. NORMES ET CONVENTIONS

Nous avons respecté les normes et conventions tout au long du développement de notre système. En particulier, nous avons suivi les bonnes pratiques de codage pour garantir la lisibilité et la maintenabilité de notre code. Nous avons également respecté les normes de sécurité pour protéger les données sensibles que nous manipulons.

En ce qui concerne la protection des données, nous prenons très au sérieux le Règlement général sur la protection des données (RGPD). Avant de commencer notre étude, nous faisons signer un formulaire de consentement à chaque participant, dans lequel nous expliquons clairement comment leurs données seront utilisées et protégées. Nous veillons à ce que nos données ne soient pas stockées sur des drives accessibles à tous, mais qu'elles soient conservées en toute sécurité dans un environnement contrôlé.

En outre, nous avons adopté des conventions pour la documentation de notre code et de notre architecture, afin de faciliter la compréhension de notre système par d'autres développeurs ou chercheurs.

VI. CONCLUSION

En somme, nous avons mis en place un système qui, nous l'espérons, est un outil précieux pour la recherche sur l'activité électrodermale. Sa conception modulaire et sa conformité aux normes et conventions pertinentes lui confèrent une flexibilité et une robustesse appréciables. De plus, sa capacité à s'adapter à une variété de paramètres de recherche offre un potentiel considérable pour l'exploration de nouvelles pistes d'étude.

Nous sommes conscients que notre travail n'est qu'une petite contribution à un champ de recherche vaste et complexe. Cependant, nous espérons que notre système pourra servir de base solide pour des recherches futures et aider d'autres chercheurs à approfondir notre compréhension de la fatigue cognitive, du stress et de la performance académique.

Nous sommes impatients de voir comment notre système sera utilisé et amélioré dans le futur, et nous sommes déterminés à continuer à apprendre et à nous adapter pour répondre aux défis à venir.

VII. ANNEXES

1. Etat de l'art
2. Cahier d'expérimentation
3. Protocole expérimental

VIII. SOURCES

Bhat, S., Gupta, D., Akel, O., Polos, P. G., DeBari, V. A., Akhtar, S., McIntyre, A., Ming, S. X., Upadhyay, H., & Chokroverty, S. (2018). The relationships between improvements in daytime sleepiness, fatigue and depression and psychomotor vigilance task testing with CPAP use in patients with obstructive sleep apnea. *Sleep Medicine*, 49, 81–89.
<https://doi.org/10.1016/j.sleep.2018.06.012>

DataMListic. (2022, November 3). *WAV2VEC2 A Framework for Self-Supervised Learning of Speech Representations - paper explained* [Video]. YouTube.
<https://www.youtube.com/watch?v=fMqYul2TvBE>

Home — PsychoPy®. (n.d.). <https://www.psychopy.org/>

lsl-website. (n.d.). <https://labstreaminglayer.org/#/>

README.md · Lajavaness/wav2vec2-lg-xlsr-fr-speech-emotion-recognition at main. (n.d.).
<https://huggingface.co/Lajavaness/wav2vec2-lg-xlsr-fr-speech-emotion-recognition/blob/main/README.md>

scikit-learn: machine learning in Python — scikit-learn 1.6.1 documentation. (n.d.). <https://scikit-learn.org/stable/>

Tuto ELAN. (2020, December 16). *ELAN : Présentation des bases et Tutoriel* [Video]. YouTube.
<https://www.youtube.com/watch?v=KMmsxqXajHw>

WAV2Vec2. (n.d.). https://huggingface.co/docs/transformers/en/model_doc/wav2vec2