

Table de matières

Table de matières	1
Liste d'abréviations	3
Liste des Figures.....	4
Liste des Tableaux.....	5
1 Rezumatul lucrării	6
1.1 Introduction	6
1.2 Context privind boala Parkinson	7
1.3 Relația dintre BP și voce	8
1.4 Extragerea caracteristicilor din semnalul de vorbire	8
1.5 Concluzia implementării și rezultatele	9
2 Planification du projet	11
3 État de l'art	12
4 Fondements théoriques	17
4.1 Introduction	Error! Bookmark not defined.
4.2 L'anatomie et la physiologie du mécanisme de la parole	17
4.2.1 Le système nerveux.....	17
4.2.2 Le système respiratoire	19
4.2.3 Le système laryngé	19
4.2.4 Le système articulaire	20
4.3 Le signal de parole.....	21
4.4 Traitement du signal de parole	24
4.4.1 L'autocorrélation et la corrélation croisée	24
4.4.2 L'analyse fréquentielle	24
4.4.3 L'estimation de la fréquence fondamentale	25
4.4.3.1 PRAAT	25
4.4.3.2 L'algorithme robuste pour le suivi de pitch (RAPT).....	26
4.5 Les caractéristiques extraites du signal de parole.....	27
4.5.1 Les variantes de jitter	28
4.5.2 Les variantes de Shimmer	29
4.5.3 Le rapport harmoniques/bruit	30
4.5.4 Les formants.....	30
4.5.5 L'analyse de fluctuation détendue (DFA).....	31

4.5.6 L'entropie de la période du pitch.....	32
4.6 L'apprentissage automatique	32
4.6.1 L'apprentissage automatique supervisé	32
4.6.1.1 Machine à vecteurs de support (SVM)	33
4.6.1.2 Forêt aléatoire (RF).....	34
4.6.1.3 Arbre de décision (DT)	34
4.6.2 L'apprentissage automatique non supervisé	34
4.6.2.1 L'analyse en composantes principales	35
4.7 Analyse de performance des modèles de classification.....	35
4.7.1 Matrice de confusion.....	35
5 L'implémentation	37
5.1 Description des ensembles de données.....	37
5.1.1 La base de données PVA-dryrun	37
5.1.2 La base de données GITA.....	38
5.1.3 Échelles existantes pour évaluer la MP :	40
5.2 L'analyse et le prétraitement des données	40
5.3 Extraction des caractéristiques	45
5.3.1 Le prétraitement et la comparaison de deux ensembles de données	48
5.4 La réduction de la dimensionnalité.....	50
5.5 Construction des modèles d'apprentissage automatique.....	51
5.5.1 Implémentation de SVM :	52
5.5.2 Implémentation de Forêt aléatoire (RF).....	53
5.5.3 Implémentation d'arbre de décision (DT).....	53
6 Résultats expérimentaux.....	55
6.1 Résultats de la corrélation.....	55
6.2 Résultats de réduction de la dimensionnalité	56
6.1 Comparaison des algorithmes d'apprentissage supervisé	58
6.1.1 Results de SVM	58
6.1.2 Random forest classifier	59
6.1.3 Resultats de Decision Tree.....	60
7 Conclusions	62
8 Bibliographie.....	63

Liste d'abréviations

MP _ Maladie de Parkinson
LSVT _ Lee-Silverman Voice Treatment
SPL _ Sound Pressure Level
IDC _ Intensity During Closure
MSD _ Mean Syllable Duration
HNR _ Harmonics-to-Noise Ratio
APQ _ Amplitude Perturbation quotient
NHR _ Noise-to-Harmonics Ratio
RPDE _ Recurrence Period Density Entropy
DFA _ Detrended Fluctuation Analysis
PPE _ Pitch Period Entropy
KNN _ K-Nearest Neighbors
MLP _ Multilayer Perceptron
RBF _ Radial Basis Function
SVM _ Support Vector Machines
MFCCs _ Mel-Frequency Cepstral Coefficients
DCT _ Discrete Cosine Transform
LR _ Logistic Regression
DT _ Decision Tree
RF _ Random Forest
NB _ Naïve Bayes
DNN _ Deep Neural Networks
PCA _ Principal Component Analysis
KPCA _ Kernel Principal Component Analysis
NN _ Neural network
CART _ Classification and Regression Tree
GP _ Globus Pallidus
VA-VL _ Ventral Anterior-Ventral Lateral
GPI _ Globus Pallidus interne
GABA _ Gamma-Aminobutyric Acid
ACh _ Acétylcholine
FFT _ Fast Fourier Transform
PDA _ Algorithmes de Détection de Pitch
RAPT _ Robust Algorithm for Pitch Tracking
SPTK _ Speech Signal Processing Toolkit
RAP _ relative Average Perturbation
PPQ5 _ 5-point Period Perturbation Quotient
DDP _ Difference of Differences of Periods
LPC _ Codage Prédictif Linéaire.

Liste des Figures

Figure 1. Procesul de detectare a bolii Parkinson prin voce.....	7
Figure 2. Présentation temporelle de l'état de l'art.....	16
Figure 3. Zones corticales impliquées dans la planification et l'exécution motrice [41].....	17
Figure 4. Section transversale anatomie, ganglions de la base [42].	18
Figure 5. Voie directe et indirecte des ganglions de la base.....	18
Figure 6. Le larynx et les cordes vocales [44].	20
Figure 7. Les systèmes principaux impliqués dans la production de la parole [45].	21
Figure 8. Schéma fonctionnel de système de génération et de production d'un signal vocal.	22
Figure 9. L'enveloppe spectrale d'un segment de parole avec les 3 formants [48].	23
Figure 10. Segment d'un signal vocal contient une partie voisée et une partie non voisée [47].	23
Figure 11. Les changements de F0 par rapport au sexe et l'âges entre 20 et 90 ans [46].	23
Figure 12. La représentation de la forme d'onde de l'enregistrement AVPEPUDEAC0013_a.wav.....	26
Figure 13. L'amplitude de segment du signal sélectionné avec le calcul de F0 min et F0 max.....	26
Figure 14. L'autocorrélation temporelle du segment de signal analysé.	26
Figure 15. L'amplitude de l'enregistrement AVPEPUDEAC0006_a.wav dans le domaine temporel.	27
Figure 16. L'estimation de F0 de l'enregistrement AVPEPUDEAC0006_a.wav par RAPT	27
Figure 17. Diagramme du LPC pour l'extraction des formants.	31
Figure 18. Principe de l'algorithme SVM.....	33
Figure 19. Matrice de confusion pour une classification binaire.....	36

Liste des Tableaux

Tableau 1. Résumé de l'analyse bibliographique des travaux réalisés en appliquant des algorithmes d'apprentissage automatique pour la classification et la détection de la MP.	16
Tableau 2. Aperçu général des données PVA des personnes atteintes de la MP.....	37
Tableau 3. Les données cliniques des participants de la base de données GITA [69].....	39
Tableau 4. Réduction de dimensionnalités avec la méthode de PCA.....	56

1 Rezumatul lucrării

În această lucrare, vom urmări câteva dintre etapele procesului de detectare a bolii Parkinson cu ajutorul vocii. În primul rând, vom analiza semnalul vocal pentru a extrage caracteristici din înregistrările vocale. Apoi vom trece la reducerea dimensionalității, care constă în minimizarea ratei de date pentru a utiliza cele mai relevante caracteristici pentru detectarea bolii Parkinson. În cele din urmă, vom aplica unii algoritmi de învățare automată pe bazele de date studiate pentru clasificare. Pentru a face acest lucru, am folosit două seturi de date: PVA_dryrun și GITA.

Primul set de date conține doar înregistrări audio ale persoanelor cu boala Parkinson, în timp ce al doilea set de date conține înregistrări audio ale persoanelor cu boala Parkinson și ale persoanelor sănătoase, care vor fi considerate drept controale.

Această activitate va fi structurată în patru etape principale. În prima parte, semnalul de vorbire din înregistrări va fi analizat pentru goluri și saturații cu ajutorul unei aplicații pe care am dezvoltat-o în Python. A doua parte prezintă contextul teoretic al extragerii caracteristicilor și aplicarea analizei componentelor principale la reducerea caracteristicilor.

Ultimele două părți prezintă câteva abordări de clasificare în context, în strânsă legătură cu rezultatele de performanță.

1.1 Introduction

Boala Parkinson (BP) este o boală neurodegenerativă care afectează sistemul nervos într-o manieră cronică și progresivă. Aceasta se caracterizează prin pierderea (sau degenerarea) celulelor dopaminergice din organism. Partea din creier în care are loc această pierdere de celule se numește substantia nigra, care este situată în mezencefal. Lipsa de dopamină duce la afectarea controlului mișcărilor.

90% dintre pacienții cu boala Parkinson se confruntă cu probleme de vorbire și voce, aceste probleme fiind primele care apar și afectând respirația, articularea, fonația și prozodia.

Vorbirea unui pacient cu BP este caracterizată de o intensitate scăzută, fluență și ritm verbal scăzut, prozodie slabă, adică accent, intonație și intonație reduse și probleme de articulare a sunetelor, ceea ce reduce semnificativ inteligibilitatea acestora.

În prezent, nu există încă metode concrete de diagnosticare a bolii Parkinson. Diagnosticul pus de medici este subiectiv și se bazează pe examinarea semnelor și simptomelor pacientului. Din nefericire, aceste simptome devin evidente doar într-un stadiu avansat al bolii.

Depistarea bolii Parkinson a atras un mare interes științific, iar nenumărate studii de cercetare au identificat o relație strânsă între BP și vocea pacienților, chiar și în stadiile incipiente ale bolii.

În acest context, au fost dezvoltate instrumente automate de detectare a bolii prin intermediul vocii, obținându-se rate ridicate de precizie în ceea ce privește recunoașterea bolii. Pe de altă parte, aceste metode, pe lângă faptul că sunt obiective, sunt și neinvazive și ne permit să știm dacă un pacient are BP mult mai devreme decât metodele tradiționale.

Pentru a utiliza aceste instrumente de recunoaștere automată a bolilor, vocea pacientului trebuie mai întâi înregistrată și apoi caracterizată. Odată ce vocea este reprezentată prin caracteristicile sale, este necesar un proces de selecție a caracteristicilor pentru a reduce costul de calcul. După acest proces, vocea reprezentată de un subset de caracteristici trebuie clasificată pentru a determina dacă o persoană este sănătoasă sau bolnavă. Figura 1 prezintă etapele întregului proces de detectare a BP bazat pe voce.

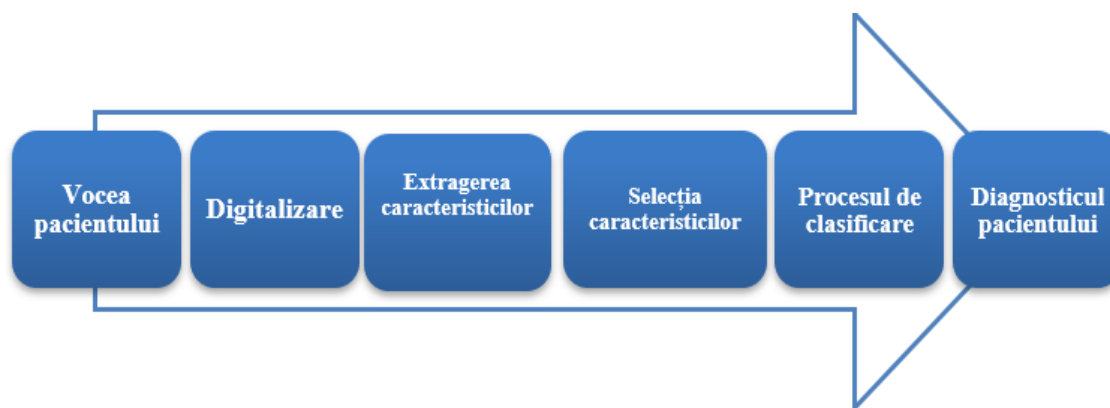


Figure 1. Procesul de detectare a bolii Parkinson prin voce.

Pentru a utiliza aceste instrumente de recunoaștere automată a bolilor, vocea pacientului trebuie mai întâi înregistrată și apoi caracterizată. Odată ce vocea este reprezentată prin caracteristicile sale, este necesar un proces de selecție a caracteristicilor pentru a reduce costul de calcul. După acest proces, vocea reprezentată de un subset de caracteristici trebuie clasificată pentru a determina dacă o persoană este sănătoasă sau bolnavă. Figura 1 prezintă etapele întregului proces de detectare a BP bazat pe voce.

În ultimii ani, datorită progreselor tehnologice, comunitatea științifică a dezvoltat un număr mare de lucrări în domeniul prelucrării digitale a vorbirii, în special în ceea ce privește studiul și detectarea automată a vorbirii patologice.

1.2 Context privind boala Parkinson

Boala Parkinson (BP) este a doua cea mai frecventă boală după boala Alzheimer. Aceasta provoacă pierderea parțială sau totală a reflexelor motorii, a vorbirii, a comportamentului, a procesării mentale și a altor funcții vitale, iar istoria a stabilit confirmarea bolii încă din anul 5000 î.Hr.

PD este o boală neurodegenerativă care afectează sistemul nervos într-o manieră cronică și progresivă, caracterizată prin pierderea (sau degenerarea) celulelor dopaminergice din organism. La pacienții cu boala Parkinson, 80% sau mai mult din aceste celule producătoare de dopamină sunt degenerate, moarte sau deteriorate, provocând descărcări necontrolate și făcându-i pe pacienți incapabili să își gestioneze mișcările.

Studiile concluzionează că vârsta este cel mai important factor de risc pentru apariția bolii, care crește dramatic după vârsta de 50 de ani, aproximativ 1% dintre persoane fiind afectate de boală.

Intervențiile chirurgicale și medicamentele pot încetini progresia bolii și pot atenua unele dintre simptome, dar nu există un tratament pentru această boală. Prin urmare, diagnosticarea precoce este esențială pentru a îmbunătăți calitatea vieții pacientului și pentru a-i prelungi viața.

În prezent, pentru a determina dacă o persoană suferă de această boală, se efectuează examinări fizice cu ajutorul unei varietăți de teste empirice, ceea ce duce la un diagnostic subiectiv care se obține după ce boala a început să cauzeze probleme pacientului și este mai dificil de încetinit progresia acesteia. Pentru a diagnostica boala într-un stadiu incipient, este necesar să se dezvolte instrumente obiective și fiabile pentru a elimina diagnosticele greșite actuale.

Observațiile făcute în timpul testelor fizice sunt comparate cu o măsură special concepută pentru a monitoriza evoluția bolii, de obicei Scala Unificată de Evaluare a Bolii Parkinson (UPDRS), care este utilizată pe scară largă pentru a evalua severitatea bolii Parkinson.

1.3 Relația dintre BP și voce

Studiile raportează că afectarea vocală are o prevalență de 70-90% după debutul bolii. În plus, poate fi unul dintre primii indicatori ai BP, iar 29% dintre pacienți consideră că este unul dintre obstacolele lor majore.

Există dovezi de degradare a performanței vocale odată cu progresia BP, cu disfonie (dificultăți de respirație, răgușeală sau crepitații ale vocii), hipofonie (volum redus al vocii) și disartrie (dificultăți de articulare a sunetelor sau silabelor).

Indicatorii disfonici ai BP fac ca măsurătorile de vorbire să fie o parte importantă a diagnosticului, deoarece au fost propuse ca un instrument fiabil pentru detectarea și monitorizarea BP.

Una dintre cele mai utilizate metode de detectare a bolii Parkinson prin voce este înregistrarea semnalelor vocale, deoarece este ușoară și neinvazivă și este considerată un parametru remarcabil pentru detectarea și urmărirea progresiei simptomelor. În cadrul acestei metode, pentru a evalua amploarea simptomelor vocale se folosesc fonații vocale susținute, pacientul fiind rugat să mențină frecvența de fonație stabilă cât mai mult timp posibil.

Pentru a caracteriza în mod obiectiv simptomele disfoniei, semnalele vocale înregistrate sunt analizate cu ajutorul unor algoritmi de procesare a vorbirii.

1.4 Extragerea caracteristicilor din semnalul de vorbire

Există mulți parametri de caracteristici în semnalul de vorbire, iar diferiți vectori de caracteristici reprezintă semnificații distincte. Alegerea acestor parametri este de mare importanță pentru detectarea cu succes a vorbirii patologice.

Tulburările de vorbire au fost evaluate cu ajutorul unor măsuri clasice și liniare. Abordările clasice de analiză a semnalelor au furnizat adesea rezultate rezonabil de precise și utile, iar modelele liniare au pus în lumină mecanisme fiziologice fundamentale. În general, termenul liniar se referă la o metodă în care ieșirea este proporțională cu o combinație liniară a intrărilor, o proprietate cunoscută sub numele de principiul superpoziției.

Deși metodele liniare sunt încă aplicabile pe scară largă și extrem de utile, abordările neliniare au fost, de asemenea, aplicate semnalelor vocale [53], unde explorările instrumentelor de procesare neliniare au arătat rezultate promițătoare [24].

Cele mai populare măsuri aplicate la procesarea semnalelor de vorbire pentru detectarea bolii Parkinson conform literaturii de specialitate:

Variantele de jitter: este o măsură afectată în principal de lipsa de control a vibrației corzilor vocale. Această măsură urmărește să cuantifice perturbațiile ciclice ale frecvenței fundamentale, reprezentate de mici abateri de la periodicitatea exactă. Se poate calcula folosind fie conturul F_0 , fie perioada de pas invers proporțională $T_0 = 1/F_0$. Măsurile legate de Jitter care măsoară, de asemenea, perturbația sunt Jitter absolut, jitter local (%), jitter RAP (Perturbare medie relativă), jitter PPQ5 (coeficientul de perturbație a perioadei în 5 puncte) și DDP Jitter (în limba engleză: Difference of Differences of Periods), care indică diferența absolută medie a diferențelor dintre ciclurile jitter.

Variantele Shimmer: aceasta este o măsură a scăderii rezistenței glotei și a afectării corzilor vocale. În general, este o măsură a instabilității amplitudinii. Shimmer și variantele sale se obțin prin măsurarea valorii maxime a amplitudinii semnalului în fiecare ciclu vocal. Variantele de strălucire includ:

Shimmer (db) este logaritmul absolut mediu în baza 10 al diferenței dintre amplitudinile unor perioade consecutive.

Sclipirea locală este diferența medie absolută dintre amplitudinile perioadelor consecutive, împărțită la amplitudinea medie.

Sclipirea APQ3 (Coeficientul de Perturbare a Amplitudinii în 3 puncte) reprezintă diferența absolută medie dintre amplitudinea unei perioade și media amplitudinilor vecinilor săi, împărțită la amplitudinea medie.

Sclipirea APQ5 reprezintă raportul dintre amplitudinea perturbațiilor din cinci perioade. Cu alte cuvinte, diferența absolută medie dintre amplitudinea unei perioade și amplitudinile medii ale acestora și ale celor mai apropiate patru vecine ale sale, împărțită la amplitudinea medie.

Strălucirea APQ11 este diferența absolută medie dintre amplitudinea unei perioade și media amplitudinilor acestora și a celor mai apropiate zece vecine ale sale, împărțită la amplitudinea medie.

HNR (raportul armonic/zgomot) este menit să cuantifice zgomotul dintr-un semnal de vorbire, care este de obicei produs de închiderea incompletă a corzilor vocale.

Oferă o indicație a periodicității generale a semnalului prin cuantificarea raportului dintre componentele periodice (armonice) și cele aperiodice (zgomot).

Formantele: un formant este maximul spectral larg care rezultă din rezonanța acustică a tractului vocal uman. Frecvențele formantelor sunt printre cele mai frecvent raportate măsuri acustice ale vorbirii și sunt utilizate într-o varietate de aplicații, inclusiv recunoașterea automată a vorbirii, studii de producție și percepție a vorbirii și evaluări clinice ale tulburărilor de vorbire.

DFA (detrended fluctuation analysis) este o măsură care urmărește să caracterizeze gradul de zgomot din semnalul de vorbire prin cuantificarea autosimilarității stocastice a zgomotului produs de fluxul de aer turbulent din tractul vocal.

Entropia frecvenței fundamentale (Pitch Period Entropy PPE) măsoară afectarea controlului stabilității tonului în timpul fonațiilor susținute, care este un simptom comun în PD. În plus, este important de menționat că această măsură folosește o scară logaritmică a înălțimii tonului și este robustă la factorii de confuzie omniprezenți, cum ar fi vibrato (o ușoară oscilație în frecvența sunetului fundamental), care este prezentă atât în vocile sănătoase, cât și în cele patologice [60].

1.5 Concluzia implementării și rezultatele

Pentru a contribui la detectarea bolii Parkinson prin voce, acest proiect lucrează cu înregistrări vocale din două tipuri de baze de date, una binară cu 50 de voci de subiecți sănătoși și 50 de persoane cu PD, iar cealaltă doar cu înregistrări ale persoanelor care suferă de PD.

Înregistrările vocale sunt de tip .Wav; un format de fișier audio digital utilizat pentru a stoca date în formă de undă cu o frecvență de eșantionare de 44,1 kHz și o rată de 16 biți.

Înainte de extragerea caracteristicilor, s-a propus eliminarea tăcerilor și a saturațiilor. În acest scop, am dezvoltat un script care monetizează înregistrările audio în segmente de o secundă, apoi implementează o metodă foarte simplă de eliminare a tăcerii care calculează energia ca sumă a pătratelor eșantioanelor, apoi calculează un prag la 50% din valoarea mediană a energiei și, în final, reține segmentele a căror energie este mai mare decât acest prag.

O altă aplicație a fost dezvoltată pentru a vizualiza starea semnalului de vorbire; scriptul nostru utilizează Parselmouth [63]; o bibliotecă Python pentru software-ul Praat pentru a analiza, sintetiza și manipula vorbirea. Această aplicație ne permite să extindem graficul vizual al amplitudinii, energiei, înălțimii, raportului armonic/zgomot și spectrograma cu primii patru formanți.

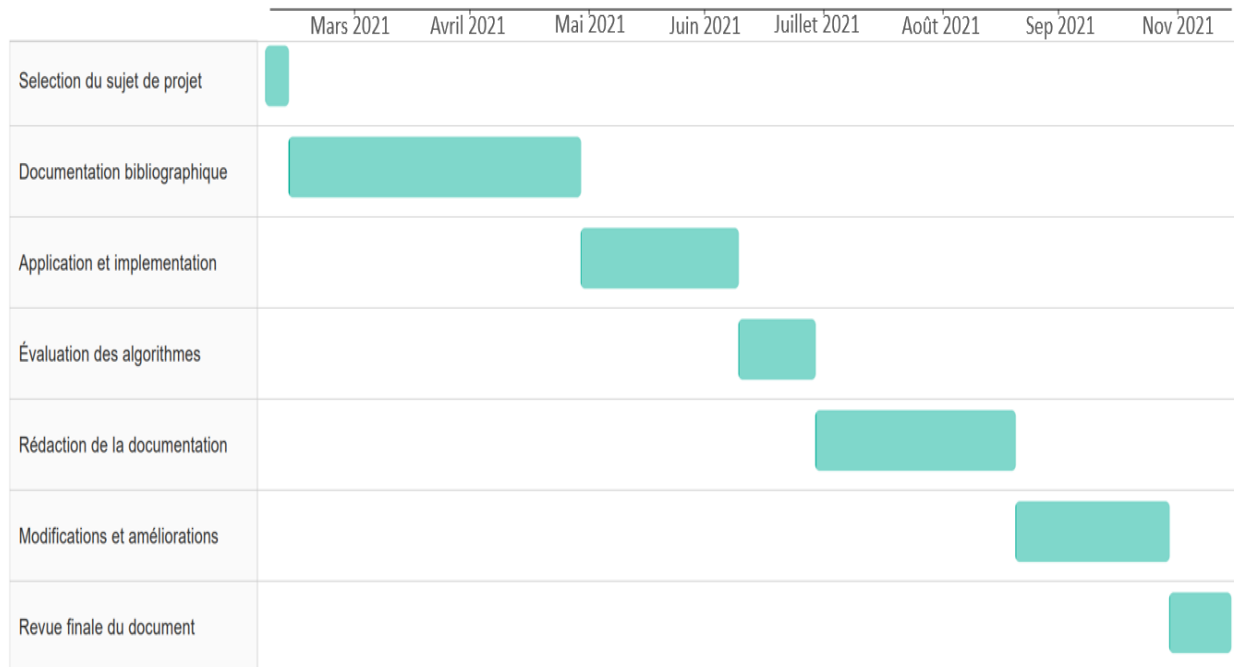
Odată ce înregistrările audio din cele două baze de date au fost procesate, am trecut la următoarea etapă, care este extragerea caracteristicilor. În acest scop, am folosit Surfboard [66], biblioteca python open-source care se bazează pe pachete de analiză audio și care pune accentul pe capacitatea de a gestiona seturi de date mari, necesare pentru abordările moderne de învățare automată. Acesta a fost construit pe baza unui număr de pachete, cum ar fi LibROSA [67] și pysptk

[68], cu o varietate de fișiere care conțin funcții pentru calcularea caracteristicilor necesare pentru studiul nostru. Am fost interesați de variantele de jitter și shimmer, raportul armonici/zgomot, formante (F1, F2, F3 și F4), analiza fluctuației relaxate și entropia perioadei de intonație.

După construirea bazelor noastre de date și verificarea faptului că toate măsurătorile sunt clasificate corect în tipul de date corect, metoda `.corr()` este utilizată pentru a măsura intensitatea corelației dintre caracteristicile din fiecare bază de date. Din interpretarea rezultatelor corelației, observăm că seturile de date au avut multe dimensiuni care sunt foarte corelate, cum ar fi variantele de jitters și shimmers; sunt corelate pozitiv, însă raportul dintre armonică și zgomot este corelat negativ cu acestea. Acest lucru ar putea face din datele noastre un model bun pentru analiza componentelor principale (PCA).

Analiza componentelor principale este o metodă foarte utilă, bazată pe matematică și statistică, care constă în eliminarea caracteristicilor care oferă mai puține informații și luarea în considerare doar a celor care conțin informațiile principale cu cea mai mare capacitate de identificare a vocii patologice. Conform primei aplicații a PCA, am reușit să explicăm peste 90% din variația din setul nostru de date cu primele 15 componente principale. Ulterior, am aplicat algoritmi de învățare automată pe seturile de date reduse și am observat că algoritmul Random Forest este cel mai bun clasificator, cu o precizie de 90%.

2 Planification du projet



3 État de l'art

La parole des personnes atteintes de la maladie de Parkinson (MP) a été étudiée depuis longtemps, mais les plus anciennes descriptions possibles des symptômes de cette maladie remontent à 2000 ans avant Jésus-Christ dans l'Ancien Testament de la Bible [1] et dans la médecine traditionnelle indienne.

Au cours de IIe siècle, le célèbre anatomiste et physiologiste expérimental de la Rome antique Galien dans son livre "De Tremore, Palpitatione, Convulsione et Rigore" [2] a distingué pour la première fois deux sortes de tremblements différents, l'un au repos et l'autre intentionnel. De plus, l'artiste et le chercheur scientifique italien Léonard de Vinci ait décrit une forme détaillée de la paralysie par tremblement comme une combinaison de difficultés à effectuer des mouvements volontaires et de tremblements [3]. Cependant, la première description formelle et scientifique de la MP est l'œuvre marquante de James Parkinson en 1817 rapportée dans "An essay on the shaking palsy", qui offrait un aperçu sur la maladie dans son contexte médical. Cette maladie que James Parkinson appelait "paralysie agitante" sera plus tard renommée en son honneur par le neurologue français Jean-Martin Charcot en 1876.

Des progrès dans les caractéristiques anatomiques et pathologiques de la maladie ont été réalisés plus tard, en 1895 lorsque Édouard Brissaud a formulé l'hypothèse que la substance noire (noyau sous-cortical du cerveau) est le siège pathologique majeur de la MP [4].

Au cours des années 1950, Arvid Carlsson, dans ses études pionnières, a détecté des niveaux particulièrement élevés de la dopamine dans les zones du cerveau qui contrôlaient la marche et d'autres mouvements volontaires. D'après ses expérimentations sur des animaux, il a montré que la dopamine est un neurotransmetteur important dans le cerveau, et que l'épuisement de celle-ci altère la capacité de mouvement. Lorsqu'il a traité ses animaux appauvris en dopamine avec l'acide aminé L-dopa, les symptômes disparaissaient et les animaux s'améliorent. Cela a conduit à l'utilisation de la L-dopa comme traitement et devenue le médicament le plus important pour la MP [70], et grâce à cette découverte Carlsson a reçu le prix Nobel plus tard en 2000.

Durant les années 1955 à 1958, la communauté scientifique n'a pas pu fournir une explication suffisante sur la relation entre les troubles de la parole et la MP, puisque la plupart des rapports ont été fondés sur des observations cliniques limitées telles que celles de Merritt [5], Cooper et al [6] et Hoberman [7]. Ces chercheurs ont noté que la réduction du volume sonore de la voix était la caractéristique la plus fréquente chez les personnes souffrant de la MP, ainsi que le terme "monotone" était souvent utilisé par Morley [8] et Grewel [9] pour décrire la parole des patients atteints de la MP. De même, les recherches présentées par Canter, dans les années 1963 et 1965, étaient concentrées sur la description des caractéristiques perceptives de la parole parkinsonienne, notamment la monotonie, les consonnes imprécises et les silences inappropriés [10] [11].

La relation entre la parole et la MP a été étudiée de manière systématique pendant les années 1970 par Darley et al [12][13] ainsi que Logemann [14]. Ces chercheurs ont remarqué une voix haletante, rauque, plus des problèmes d'articulation, de souffle et de prononciation chez les patients parkinsoniens. De plus, une recherche menée par Perez et al [15], illustre que les patients atteints de la MP présentent une incidence élevée d'anomalies de la voix et du larynx (organe de l'appareil respiratoire situé dans la gorge). La majorité de ces symptômes ont été associés à la dysarthrie hypokinétique. Le terme dysarthrie vient du nouveau latin, dys- « dysfonctionnel » et arthr- « articulation vocale ». Il est généralement utilisé pour englober les symptômes observés dans la parole parkinsonienne.

D'un point de vue clinique, la parole dysarthrique se caractérise par plusieurs anomalies au niveau de la phonation, l'articulation et la prosodie. Les problèmes de phonation sont liés à la courbure et à la fermeture incomplète des cordes vocales [15], et les problèmes d'articulation se manifestent par une réduction de l'amplitude et de la vitesse des mouvements articulatoires des

lèvres, de la mâchoire et de la langue [16]. En outre, les troubles de la prosodie (l'intonation) se manifestent par une voix monotone, des changements dans le rythme de la parole [17], plus des difficultés à exprimer des émotions [18].

L'histoire de la MP est étroitement liée aux interventions thérapeutiques dont la rééducation orthophonique est l'une des thérapies indispensables pour les patients afin de retarder l'impact de la MP sur la parole. En 1985 Dr. Ramig a fondé le programme Lee-Silverman Voice Treatment (LSVT) en l'honneur du Mrs. Lee Silverman, son premier patient décédé avant que la recherche ne soit officiellement publiée et reconnue comme une découverte médicale [19]. LSVT est un protocole d'autothérapies appliqué dans le monde entier pour améliorer la parole des patients parkinsoniens, il a démontré son efficacité après 6 mois de traitement orthophonique constant [20]. Cependant, il n'inclut pas l'évaluation objective des signaux vocaux et il n'a pas été conçu dans le but d'analyser des aspects spécifiques de la parole qui sont liés au processus de communication tels que l'articulation, et la prosodie. En 1992, plusieurs déficits articulatoires dont souffrent les personnes atteintes de la MP ont été documentés par Ackerman et al. [71]. Les auteurs ont examiné les enregistrements vocaux de 12 locuteurs atteints de la MP et 12 témoins sains. Les enregistrements ont été échantillonnés à 20 kHz et segmentés manuellement. L'ensemble des caractéristiques extraites des enregistrements vocaux comprend d'abord le niveau de pression acoustique (SPL : sound pressure level) des consonnes d'arrêt telles que (/b/, /g/, /p/, /k/). Ensuite la durée moyenne de la syllabe (MSD : mean syllable duration) définie comme l'intervalle de temps entre les sommets syllabiques dans le contour SPL estimé sur différents mots consonnes-voyelles. Puis l'intensité pendant la fermeture (IDC : intensity during closure) et enfin le quotient de transition (TQ). L'IDC mesure la qualité acoustique de la production des consonnes stop et le TQ est le quotient entre la portion de mouvement d'une syllabe et la durée totale de la syllabe. Les auteurs ont rapporté une réduction de la précision articulatoire des consonnes produites par les patients atteints de la MP.

Les modèles de phonation dans la parole des patients atteints de la MP sont étudiés par Jiménez et al [21] en 1997. Les auteurs ont évalué 22 patients atteints de la MP non traités par des médicaments dopaminergiques, et 28 sujets sains. Il est demandé aux participants de répéter trois fois la voyelle soutenue /a/ et de lire une phrase. Des mesures phonatoires sont prises en compte comme la fréquence fondamentale (F_0) et ses variations (le shimmer et le jitter), ainsi que Harmonics-to-Noise Ratio (HNR) qui indique l'amplitude du bruit par rapport aux composantes tonales. Finalement, les auteurs ont observé que le jitter et le shimmer sont plus élevés dans la parole des patients par rapport aux sujets sains, cependant le HNR et la F_0 étaient plus faibles chez les patients. Ces observations ont été confirmées plus tard par les auteurs [22] en considérant un ensemble de 41 patients atteints de la MP qui étaient sous traitement dopaminergique.

Des études plus récentes ont montré que plus de 90 % des patients atteints de la MP souffrent de problèmes d'élocution [23][24] et que le premier symptôme qu'on peut observer chez eux est la dégradation de la voix [25]. En outre, une étude pilote menée en 2004 a révélé que les troubles de la parole pouvaient être détectés cinq ans avant le diagnostic de la MP [26] à partir de changements observés au niveau de la variabilité de la fréquence fondamentale (F_0). Ces dernières années, la maladie de Parkinson a fait l'objet de nombreuses recherches, car les coûts des soins de santé liés à cette pathologie ne cessent d'augmenter en raison de la longévité croissante de la population. De plus, cette maladie est de nature progressive, il est devenu nécessaire à la détecter à un stade précoce où les médicaments sont plus efficaces et les traitements tels que LSVT sont également plus faciles à réaliser dans les premiers stades de la maladie et peuvent contribuer à ralentir sa progression.

En ce qui concerne les recherches antérieures, il a été constaté que l'intelligence artificielle et les techniques d'apprentissage automatique ont un bon potentiel pour la classification. Il a été également constaté que le système de classification aide à améliorer la précision et la fiabilité du diagnostic et aussi à minimiser les erreurs ainsi qu'à rendre le système plus efficace. Du coup, plusieurs tentatives remarquables ont été développées par différents chercheurs au fil du temps pour la détection et la prédiction de la MP par l'analyse des caractéristiques de la voix en utilisant

des algorithmes d'apprentissage automatique, à cet égard, nous avons présenté ci-dessous une étude bibliographique des travaux réalisés (tableau.1) pour donner une idée générale de la quantité de travail effectué dans ce domaine.

En 2009, Little et al. [24] ont examiné un ensemble de 23 patients atteints de la MP et 8 sujets sains âgés de 46 à 85 ans. Chaque locuteur a produit la voyelle soutenue /a/ environ six fois. Les auteurs ont calculé différentes caractéristiques de la dysphonie. Ils ont utilisé Praat [27] pour calculer l'ensemble des mesures qui comprennent : le jitter, le shimmer, l'APQ (Amplitude Perturbation quotient) qui mesure la moyenne des variations d'amplitude, puis le Harmonics to Noise Ratio (HNR) et le Noise-to-Harmonics Ratio (NHR) qui quantifient le rapport entre l'information réelle du signal et le bruit. En plus de ces mesures, les auteurs ont estimé d'autres quatre caractéristiques qui sont dérivées de l'analyse non linéaire de la parole. Ces mesures non linéaires comprennent : l'entropie de densité de période de récurrence (RPDE) [28], l'analyse des fluctuations détendues (DFA) [28], la dimension de corrélation (D2) [29] et l'entropie de la période du pitch (PPE) qui est une nouvelle mesure de la dysphonie introduite dans [24]. Les auteurs ont distingué les locuteurs MP des sujets sains en utilisant la machine à vecteur de support (SVM) avec un noyau gaussien. Ils ont rapporté une performance globale de classification correcte de 91,4%.

Également en 2009, Sakar et Kursun [30] ont considéré le même ensemble de données utilisé dans [24]. Les auteurs ont calculé un total de 22 caractéristiques, y compris celles considérées par Little et al., plus deux mesures non linéaires supplémentaires de la variation de la fréquence fondamentale. L'objectif de cette étude est de sélectionner un ensemble minimal de caractéristiques ayant une pertinence conjointe maximale pour le MP-score binaire indiquant si l'échantillon appartient ou non à une personne atteinte de la MP. Dans cette étude, le SVM a été entraîné en suivant une stratégie de validation croisée "leave-one-individual-out" pour évaluer la performance du modèle sur l'ensemble de données. L'idée de cette stratégie est de laisser à côté les échantillons d'un seul individu pour les utiliser à la validation comme un ensemble de test. Finalement, les auteurs affirment que les mesures vocales ont un potentiel évident pour l'évaluation de la MP et ils ont rapporté une précision de 81,5%.

Une année plus tard, Bhattacharya et Bhatia [31] ont utilisé l'algorithme SVM pour séparer les sujets MP des sujets sains. Ils se sont servis de l'ensemble de données et les caractéristiques considérées par Little et al. [24]. Les auteurs ont fait recours à l'outil Weka pour l'exploration de données. Un prétraitement des données a été effectué avant d'appliquer l'algorithme de classification et une division aléatoire a été faite à plusieurs reprises. La meilleure précision de 65,21% a été rapportée en utilisant le SVM avec un noyau linéaire.

En 2011, les auteurs de [32] ont extrait 13 caractéristiques différentes de chaque signal vocal de l'ensemble de données utilisé dans [24]. Ces caractéristiques comprennent : le mean, le max et le min de la fréquence fondamentale, quatre variations de jitter, cinq variations de shimmer et le HNR. Dans leur travail, la dimension des attributs a été réduite et des caractéristiques optimisées ont été obtenues à l'aide de l'algorithme génétique, ainsi que la méthode de k plus proches voisins (k-NN) a été utilisée pour la classification. Un pourcentage de précision de 98,2% pour 9 caractéristiques optimisées a été obtenu, ce qui est un résultat notable par rapport aux autres études.

Pour mener à bien leur classification, Uma Rani et Holi [33] ont utilisé un perceptron multicouche (MLP) et un noyau de fonction de base radiale (RBF). Le RBF est une fonction populaire utilisée dans divers algorithmes d'apprentissage par noyau en particulier dans la classification par SVM, cependant que le MLP est une catégorie de réseau neuronal artificiel. Dans cette recherche [33], Les auteurs ont étudié des données constituées de 136 phonations de voyelles soutenues /a/, dont 83 ont été enregistrées par des personnes souffrant de la MP et 53 par des individus sains. Sachant que 112 phonations ont été utilisées pour l'entraînement et 24 pour le test, un total de 16 caractéristiques a été extrait de données vocales. Les auteurs ont constaté que le RBF a donné de meilleurs résultats par rapport au perceptron multicouche avec un pourcentage de 90,12 %.

Dans un autre travail important, Sakar et al [34] ont proposé une méthode pour l'élaboration d'un modèles prédictifs de télédiagnostic de la MP. L'objectif de cette étude est de concevoir un

système de collecte, de stockage et d'analyse des données assisté par ordinateur pour simplifier le processus de diagnostic et de traitement de la MP dans le département de neurologie de la faculté de médecine de Cerrahpasa à Istanbul. Dans le cadre de leur étude, les données ont été recueillies auprès de 40 personnes (20 sujets sains et 20 patients atteints de la MP). Chaque sujet a dû enregistrer 26 échantillons vocaux comprenant des mots, des phrases courtes, des chiffres et des voyelles soutenues. Le logiciel d'analyse acoustique Praat [27] était utilisé pour l'extraction des caractéristiques et pour la classification, ils ont appliqué le SVM et la méthode des k plus proches voisins. Les valeurs de 1, 3, 5 et 7 ont été choisies pour k successivement, ainsi que pour le SVM ils ont spécifié le noyau linéaire et RBF. Finalement les auteurs ont obtenu des précisions de 82,50% pour le k-NN et 85% pour le SVM.

Poursuivant sur l'axe des études réalisées antérieurement, Mohammad Shahbakhi et al [35] ont proposé une méthode en utilisant le même ensemble de données de Max Little [24] pour lequel 22 caractéristiques linéaires et non linéaires ont été extraites, y compris 14 basées sur : la fréquence fondamentale, le jitter, le shimmer et le HNR. Les auteurs affirment que ces facteurs sont les plus importants dans le signal de la parole. Ils ont adopté l'algorithme génétique pour sélectionner les caractéristiques les plus pertinentes qui sont ensuite introduites dans le SVM pour la classification. Finalement pour 4, 7 et 9 caractéristiques optimisées, des précisions de 94,5 %, 93,66 % et 94,22 % ont été rapportées respectivement.

Afin de développer l'évaluation des troubles de la parole pour détecter les patients atteints de la MP, Achraf Benba et al [36] ont étudié les coefficients cepstraux à l'échelle de Mel (MFCCs) qui sont généralement les mesures standard pour la reconnaissance vocale et elles ont été testées dans la modélisation de différents types de pathologies telles que la dysphonie [72], et la dysarthrie [73]. L'estimation des MFCCs consiste à prendre la transformée discrète en cosinus (DCT) de l'énergie logarithmique dont les fréquences sont également espacées sur l'échelle de Mel. Dans cette étude [36], différents types de noyaux SVM ont été appliqués après avoir extrait 20 MFCCs pour chaque sujet de la base de données, qui représente les enregistrements audios de 34 voyelles soutenues, dont 17 ont été recueillies auprès de personnes atteintes de la MP. Enfin, la meilleure précision de classification obtenue était de 91,18% en utilisant les 12 premiers coefficients des MFCCs par SVM à noyaux linéaires.

Une année plus tard, les mêmes auteurs ont présenté un autre travail [37], dans lequel ils ont cherché à séparer les sujets atteints de Parkinson de ceux qui souffrent des autres maladies neurologiques. Ils ont recueilli des échantillons de 50 personnes, dont 30 étaient des patients atteints de la MP et 20 des autres maladies neurologiques. À partir de chaque échantillon vocal, des coefficients cepstraux ont été extraits et cinq classificateurs supervisés ont été appliqués avec différents types de noyaux. Les 11 premiers coefficients cepstraux de prédiction linéaire perceptive ont montré une précision de 90% avec le noyau linéaire SVM.

La recherche [38] de Anand et al a été menée dans le but d'appliquer six différents algorithmes d'apprentissage automatique parmi lesquels on trouve logistic regression (LR), decision tree (DT), random forest (RF), k-NN, naïve bayes (NB), SVM, et l'algorithme réseaux neuronaux profonds (DNN) de deep learning. En outre, l'analyse en composantes principales (PCA) et l'analyse en composantes principales par noyau (KPCA) sont utilisées pour extraire les meilleures caractéristiques possibles et réduire la dimensionnalité de l'ensemble de données [28] qui contient les mesures vocales biomédicales de 31 personnes, dont 21 souffrants de la maladie de Parkinson. De tous les algorithmes utilisés, le k-NN était capable de donner la plus grande précision comparable de 95,52%.

Dans un autre travail important, Salama et al [39] ont proposé une méthode basée sur l'approche d'évaluation des caractéristiques multiples qui a sélectionné 11 caractéristiques principales. Les algorithmes de classification tels que SVM, NB, RF, DT et le réseau neuronal (NN), ont été mis en œuvre avec une méthode de cross-validation. Pour l'analyse des résultats, l'ensemble de données original a été utilisé en premier lieu, puis les caractéristiques filtrées. Il a été observé que la plus grande précision de 87,75% a été produite par le NN sur l'ensemble de données original, mais sur les caractéristiques filtrées, le RF a produit une précision de 99,49%,

cependant le NB, le RN, le DT et le SVM ont obtenu une précision de 89,34%, 96,95%, 96,95% et 95,43% successivement.

Une étude récente de Zehra Karapinar [40] a présenté un cadre pour le diagnostic précoce de la MP basé sur l'apprentissage automatique. La méthode de diagnostic proposée consiste en des processus de sélection de caractéristiques et de classification. L'auteur utilisait le Réseau Neuronal (RN), CART (Classification and Regression Tree) et le SVM. Avant l'utilisation de la méthode de sélection de caractéristiques, une précision de 85,23%, 79,98% et 80,25% a été obtenue pour CART, SVM et NN respectivement. Après l'utilisation de la méthode de sélection de caractéristiques, la précision a été améliorée à 90,76% pour CART, 91,54% pour le NN et 93,84% pour le SVM. Il a été constaté que le modèle était plus performant après la sélection des caractéristiques.

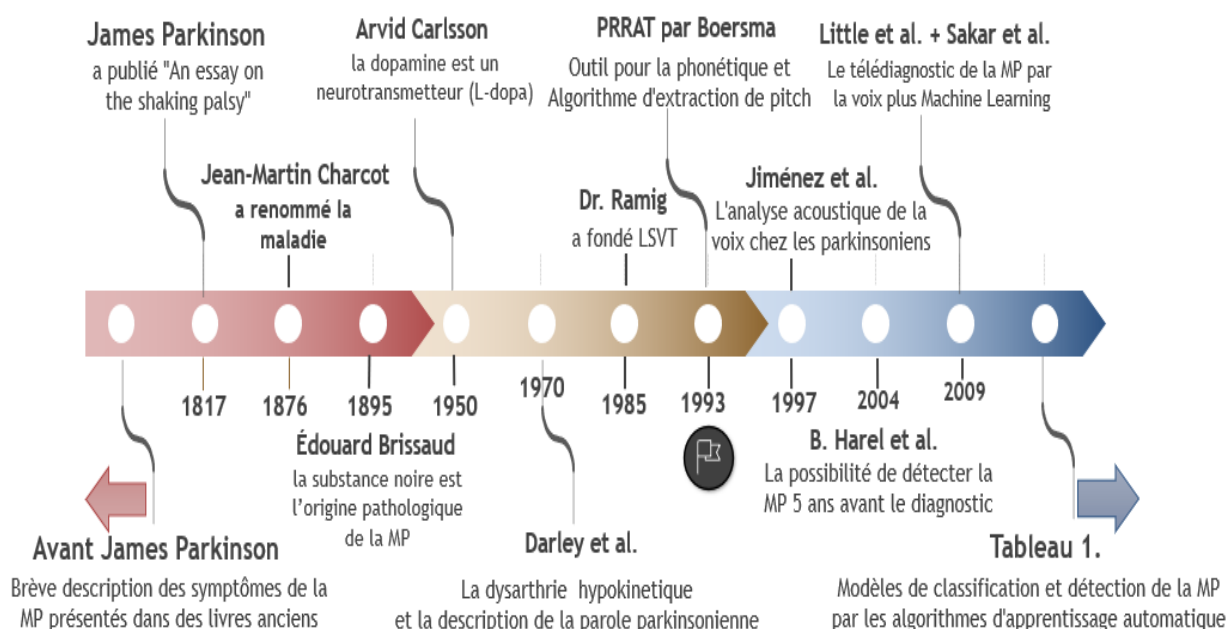


Figure 2. Présentation temporelle de l'état de l'art.

Tableau 1. Résumé de l'analyse bibliographique des travaux réalisés en appliquant des algorithmes d'apprentissage automatique pour la classification et la détection de la MP.

Référence	Année	Auteur	Meilleure Technique appliquée	Précision %
[24]	2009	Little et al.	SVM	91,4
[30]	2009	Sakar et Kursun	SVM	81.5
[31]	2010	Bhattacharya et al.	SVM linéaire	65.21
[32]	2011	Shirvan et Tahami	k-NN	98,2
[33]	2012	Uma Rani et Holi	SVM (RBF)	90,12
[34]	2013	Sakar et al.	SVM linéaire	85
[35]	2014	Shahbakhhi et al.	SVM	94,5
[36]	2015	Achraf Benba et al.	SVM linéaire	91,17
[37]	2016	Achraf Benba et al.	SVM linéaire	90
[38]	2018	Anand et al.	k-NN	95,52
[39]	2019	Salama et al.	Random Forest	99.49
[40]	2020	Zehra Karapinar	SVM	93,84

4 Fondements théoriques

4.1 L'anatomie et la physiologie du mécanisme de la parole

La maladie de Parkinson peut être définie comme une maladie neurodégénérative du cerveau qui provoque des troubles du mouvement. Elle survient lorsque 60 à 80 % des cellules productrices de la dopamine du cerveau sont endommagées et ne produisent plus suffisamment de cette substance chimique. Pour expliquer comment la parole est affectée par la MP, il est important de comprendre les systèmes principaux impliqués dans la production de la parole ainsi que l'anatomie et la physiologie du mécanisme de la parole, et découvrir comment tous ces facteurs peuvent être influencés par la Maladie de Parkinson.

4.1.1 Le système nerveux

Il est nécessaire d'introduire les bases de l'anatomie du cerveau concernant le contrôle moteur et la planification de la parole pour acquérir une compréhension générale de la façon dont le processus de génération de la parole est affecté par la maladie de Parkinson.

Le cortex cérébral est composé de plusieurs parties, parmi lesquelles on trouve le cortex préfrontal, le cortex moteur et le cortex prémoteur, qui sont responsables de la planification et l'exécution des mouvements. Comme l'illustre la figure 3, une partie de l'aire de Broca ; la zone qui est impliquée dans le contrôle de la parole appartient également au cortex moteur et prémoteur.

La plupart des zones situées en dessous de la couche de cortex cérébral sont impliquées dans les activités motrices et peuvent être étudiées en prenant une section coronale du cerveau, précisément les ganglions de la base. La figure 4 illustre une telle section et comprend certaines parties impliquées dans les activités motrices.

Le cortex prémoteur organise les séquences motrices planifiées et transmet l'ordre vers le striatum (formé par le noyau caudé et le putamen). Ce dernier se projette vers le Globus Pallidus (GP), qui contient les neurones de sortie des ganglions de la base ensuite, le GP se projette à son tour vers le noyau ventral antérieur-ventral latéral (VA-VL) dans le thalamus moteur afin de passer l'ordre au cortex moteur, qui envoie finalement les informations aux muscles correspondants pour exécuter le mouvement.

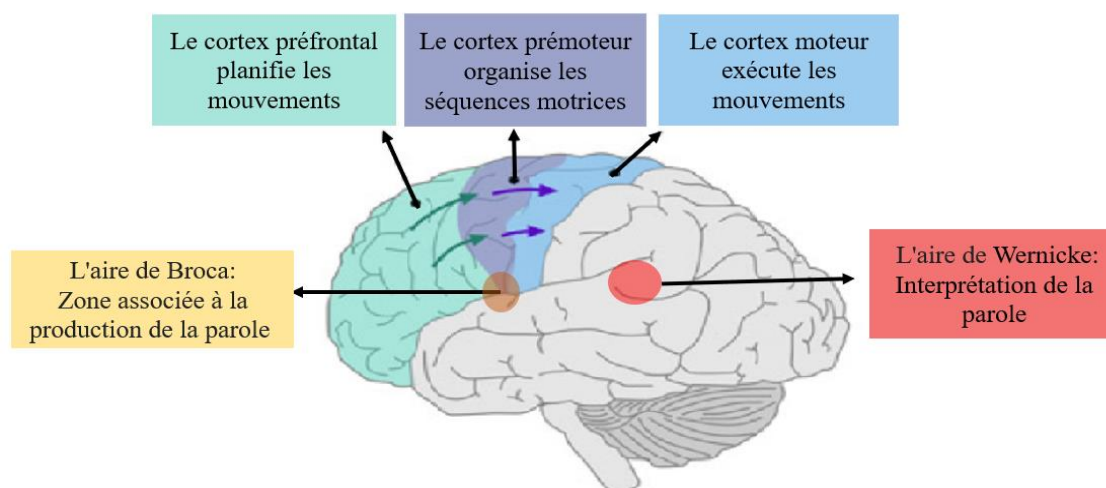


Figure 3. Zones corticales impliquées dans la planification et l'exécution motrice [41].

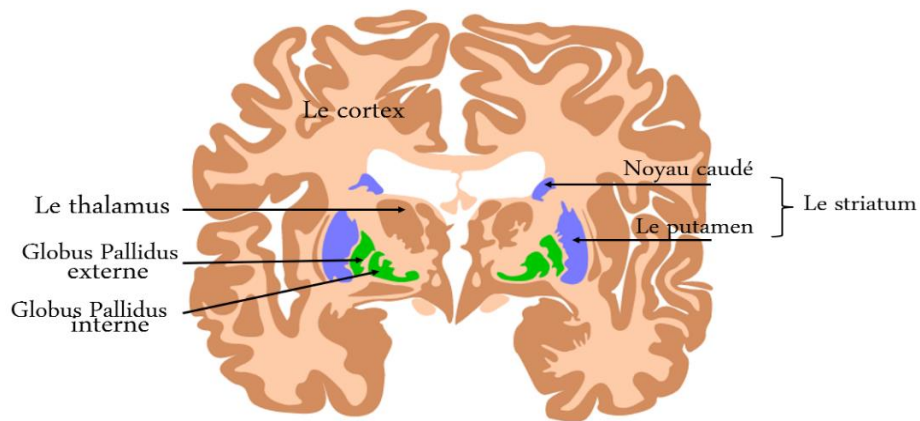


Figure 4. Section transversale anatomie, ganglions de la base [42].

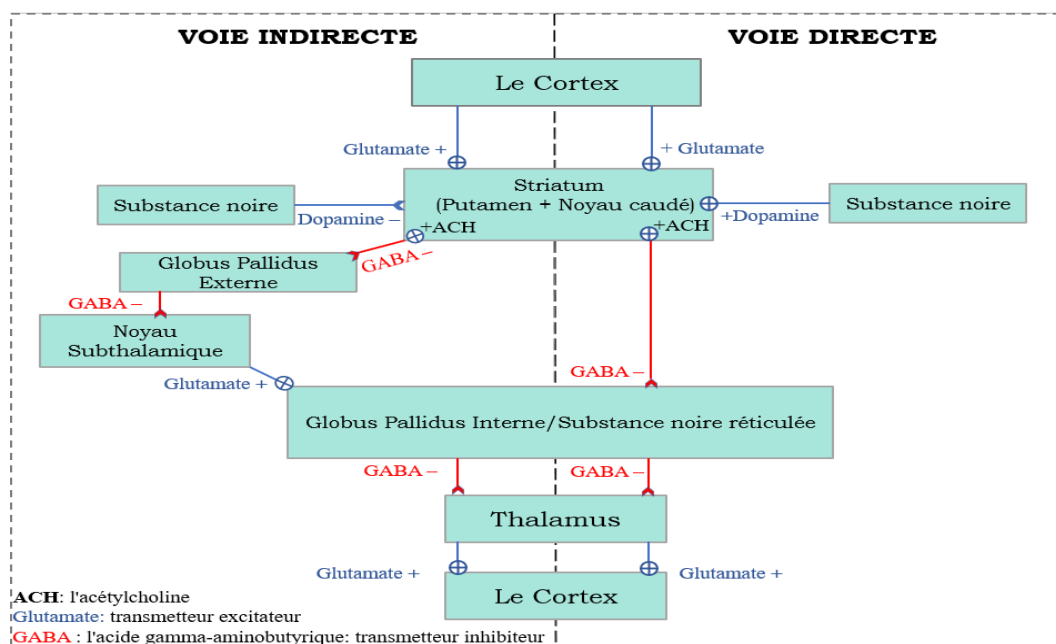


Figure 5. Voie directe et indirecte des ganglions de la base

Comme le montre la figure 5, il existe deux façons d'exciter les neurones du thalamus moteur : la voie directe et la voie indirecte. La différence entre eux réside dans l'inhibition des neurones du noyau subthalamique avant l'excitation du Globus Pallidus interne GPI, qui se produit dans la voie indirecte.

- La voie directe :

les fibres du cortex prémoteur excitent les neurones du striatum par l'intermédiaire des neurotransmetteurs Glutamate. Ces cellules inhibent les neurones du GPI par le biais des neurotransmetteurs de l'acide gamma-aminobutyrique (GABA). Alors plus d'inhibition dans le GPI signifie moins d'inhibition dans le thalamus moteur, ce qui entraîne une augmentation de la stimulation des neurones dans le cortex moteur, en augmentant ainsi l'entrée thalamique dans le cortex moteur, c'est-à-dire que la voie directe augmente l'activité motrice.

- La voie indirecte :

A ce stade, les neurones du striatum qui sont excités par le cortex prémoteur inhibent les cellules du Globus Pallidus externe (GPe) qui, à leur tour, inhibent les neurones du noyau subthalamique. Les cellules subthalamiques sont donc désinhibées et augmentent leur activité, ce

qui entraîne une plus grande excitation des neurones du GPi. Alors les GPi augmentent leur inhibition dans le thalamus moteur, ce qui se traduit par une diminution de l'activité motrice dans le cortex moteur.

- Les neurones dopaminergiques et cholinergiques dans le contrôle moteur :

La dopamine est produite par les cellules de la pars compacta (partie noire) de la substance noire. Les neurones dopaminergiques libèrent la dopamine dans le striatum et leur effet est excitateur sur les cellules impliquées dans la voie directe et inhibiteur sur les cellules de la voie indirecte. D'autre part, les neurones cholinergiques (également appelés acétylcholine - ACh) inhibent les cellules du striatum de la voie directe et excitent les cellules de la voie indirecte.

- Effet de la maladie de Parkinson Généralement

La MP est caractérisée par la perte de neurones dopaminergiques dans la substance noire. Comme la voie directe augmente l'activité motrice et que la dopamine joue un rôle excitateur sur les neurones du striatum, l'activité motrice d'une personne souffrant de la MP va diminuer en raison d'un double effet négatif dans la voie directe, c'est-à-dire la perte de l'excitation induite par la dopamine et la persistante d'inhibition neuronale de l'ACh. Lorsque l'inhibition finalement atteint le thalamus moteur, le cortex moteur soit moins actif. Ce symptôme, connu sous le nom d'hypokinésie qui affecte le contrôle moteur de tous les muscles du corps, y compris ceux impliqués dans la production de la parole. C'est la raison pour laquelle on utilise le terme de dysarthrie hypokinétique pour englober la plupart des symptômes observés dans la parole des personnes atteintes de la MP. Une analyse similaire peut être faite sur la voie indirecte. Lorsque la dopamine inhibe la voie indirecte, on observe une augmentation de l'activité de la voie indirecte. En même temps, les neurones ACh ont un effet excitateur, ce qui réduit l'activité motrice.

4.1.2 Le système respiratoire

Les deux principaux composants du système respiratoire sont les poumons et le diaphragme. Les poumons fournissent l'air nécessaire pour la production de la parole, tandis que les contractions du diaphragme déterminent la puissance des sons vocaux. De plus, les muscles intercostaux externes et internes sont impliqués aussi dans la respiration (figure 7) [43]. Les muscles intercostaux externes sont des muscles inspiratoires qui soulèvent les côtes de la cage thoracique, et les muscles intercostaux internes sont des muscles expiratoires qui abaissent les côtes.

Durant la parole, les muscles inspiratoires remplissent une fonction extrêmement importante appelée ; l'action de contrôle Celle-ci se produit lorsque les muscles inspiratoires modèrent la quantité d'air libérée par les poumons pendant la parole. Cette action est importante car elle permet aux personnes d'avoir un flux d'air constant dans leur conduit vocal, ce qui permet de contrôler la pression sous-glottique sous les cordes vocales [43]. Donc l'action de contrôle joue un rôle clé dans le maintien de l'intensité vocale et de la fréquence à laquelle les cordes vocales vibrent. Si une personne présente un déficit dans le contrôle de l'action, elle ne sera pas en mesure de contrôler un flux d'air régulier dans le conduit vocal.

4.1.3 Le système laryngé

Généralement la structure la plus importante du système laryngé est le larynx, qui est soutenu par l'os hyoïde et les cartilages de la région, comme le montre la figure 6. Il existe trois fonctions biologiques du larynx ; la première fonction est d'empêcher l'air de s'échapper des poumons. Ceci est le résultat de la fixation abdominale (contraction des muscles abdominaux). La deuxième fonction est d'empêcher les substances indésirables de pénétrer dans la trachée, la glotte et les poumons. Ceci est réalisé lorsque l'épiglotte se ferme sur l'ouverture du larynx pendant la déglutition et la troisième fonction du larynx est la phonation, c'est-à-dire ; la production de la voix. Normalement les muscles du larynx fonctionnent de manière involontaire, ce qui signifie que

nous avons un peu de contrôle sur eux directement. Ce contrôle se fait par un processus impliquant la détection et le suivi de la vibration des cordes vocales par le biais du son et de la sensation générée.

Les cordes vocales sont la partie la plus importante du mécanisme de la parole. Ils sont localisés au sein du conduit formé par le larynx, lorsqu'ils phonent (s'ouvrent et se ferment), la parole est produite. La phonation est affectée par un ensemble de muscles du système laryngé, notamment le muscle vocalis, les muscles extrinsèques et les muscles supplémentaires. Les mouvements de ces muscles affectent la nature des ondes sonores produites lorsqu'une personne parle.

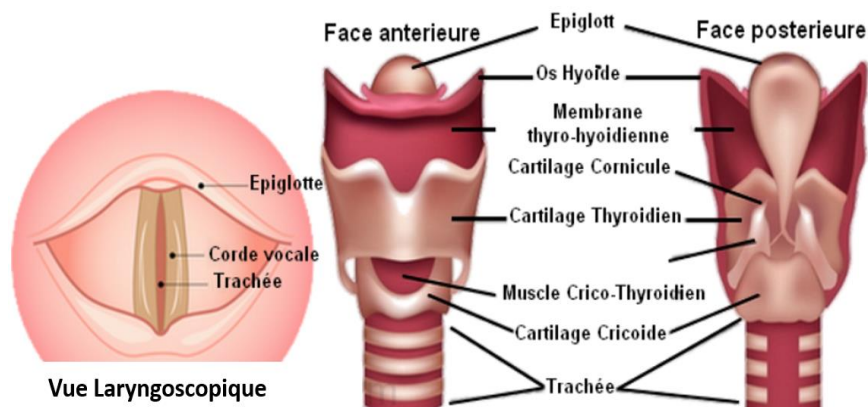


Figure 6. Le larynx et les cordes vocales [44].

4.1.4 Le système articulatoire

Le système articulatoire est situé au-dessus du système laryngé. Il est responsable des nuances des sons de la parole lorsqu'une personne parle. Les humains sont capables de produire des sons complexes en fonction des dimensions et des caractéristiques anatomiques de la région bucco-nasale, qui comprennent les cavités, les muscles, les os et les dents. L'articulation désigne les mouvements coordonnés de ces éléments. L'ensemble des cavités du système articulatoire constitue le conduit vocal, appelé aussi tractus vocal qui est composé de trois cavités ; la cavité nasale, qui est l'espace situé entre le haut du crâne et la mâchoire supérieure ; la cavité orale, qui est l'espace entre la mâchoire supérieure et la mâchoire inférieure ; et la cavité pharyngée, qui est l'espace situé à l'arrière de la gorge. En plus de ces cavités, les structures du système articulatoire, connues sous le nom d'articulateurs, sont également essentielles pour la production de la parole. Ces structures sont divisées en deux groupes : les articulateurs fixes et les articulateurs mobiles. Comme leur nom l'indique, les articulateurs fixes sont des structures anatomiques qui sont fixées en place, notamment des structures en os telles que les dents. Les ondes sonores interagissent avec ces structures, elles ne peuvent pas être modifiées de manière à affecter les qualités physiques des ondes sonores qui sont finalement produites. En revanche, les articulateurs mobiles sont flexibles et peuvent donc être modifiés de manière à influencer les qualités physiques des ondes sonores. Parmi les articulateurs mobiles on trouve les lèvres, la langue, le pharynx, le palais mou et la mandibule. En fin de compte, les interactions entre ces structures et les articulateurs fixes lorsque l'air circule dans les cavités du conduit vocal, déterminent les qualités distinctes des ondes sonores qui se manifestent dans la parole humaine. De plus, pendant la production de la parole, une personne est capable de manipuler l'air en différents sons en utilisant différents articulateurs [43]. Cependant, une personne peut également modifier la résonance d'un son en ouvrant ou en fermant sa cavité nasale, ce qui affecte la nasalité d'un son.

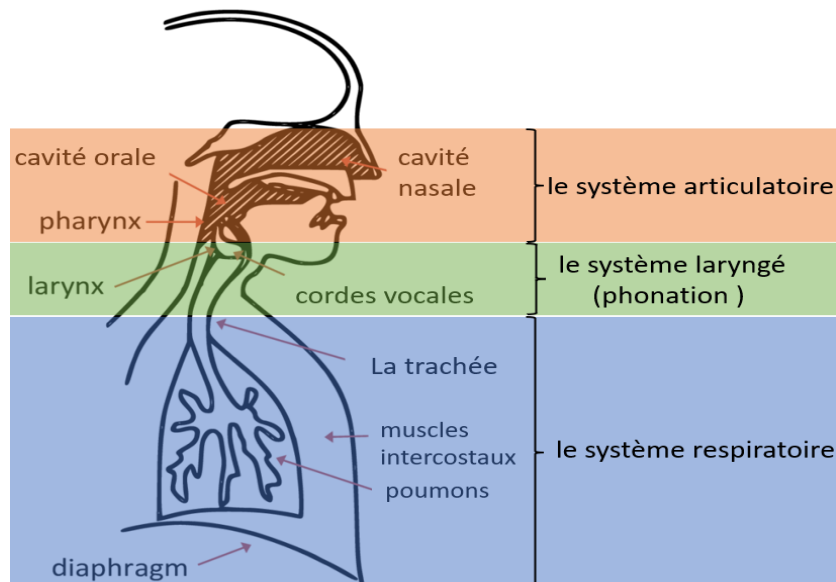


Figure 7. Les systèmes principaux impliqués dans la production de la parole [45].

Le processus d'utilisation des systèmes respiratoire, laryngé et articuloire ainsi que toutes leurs parties pour produire la parole est appelé théorie source-filtre. Selon cette théorie, le larynx est la source de la parole et le conduit vocal supra-laryngé est le filtre [43]. Le larynx et les cordes vocales sont mis en phonation (vibration) lorsqu'une pression suffisante se développe sous les cordes vocales et qu'elles sont écartées, ce qui entraîne la vibration de l'air qui les entoure. Cet air est ensuite déplacé dans le conduit vocal supra-laryngé (tout ce qui se trouve au-dessus du larynx) et modelé par tous les articulateurs, tels que la langue, la mandibule, le palais mou et les lèvres, pour devenir une parole intelligible [43]. Les patients atteints de la MP rencontrent des difficultés à régler leurs cordes vocales en phonation. Ils ont également du mal à déplacer leurs articulateurs pour produire une parole claire, et les cordes vocales ne seraient pas en mesure de se fermer correctement. Cette inadaptation de l'adduction des cordes vocales entraîne une diminution de la qualité de la voix.

4.2 Le signal de parole

Le processus de génération d'un signal de parole commence généralement à partir du cerveau au niveau de deux zones, l'aire de Broca (en jaune) et l'aire de Wernicke (en rouge) de la figure 3 au-dessus. Les deux zones sont interconnectées et travaillent en parallèle pour générer des signaux sonores précis. En fait, la région de Broca reçoit des informations sous forme de gestes auditifs et visuels par l'intermédiaire d'autres organes sensoriels pour l'interprétation et le traitement de la parole, ce qui l'aide à décider de ce qu'il faut dire ou du son à produire. En conséquence, la région de Wernicke génère des signaux de contrôle pour déplacer l'appareil du tractus vocal (nasale, orale et pharyngée) ainsi d'autres organes de production de la parole tels que les poumons, les cordes vocales, la glotte, et les articulateurs fixes et mobiles. Alors après l'exécution de commandes neuromusculaires, le son est produit en fonction de la forme des différentes cavités. L'ensemble du processus est illustré dans la figure 8 qui résume les différentes parties du système de production de la parole et comment elles sont liées pour produire un signal vocal.

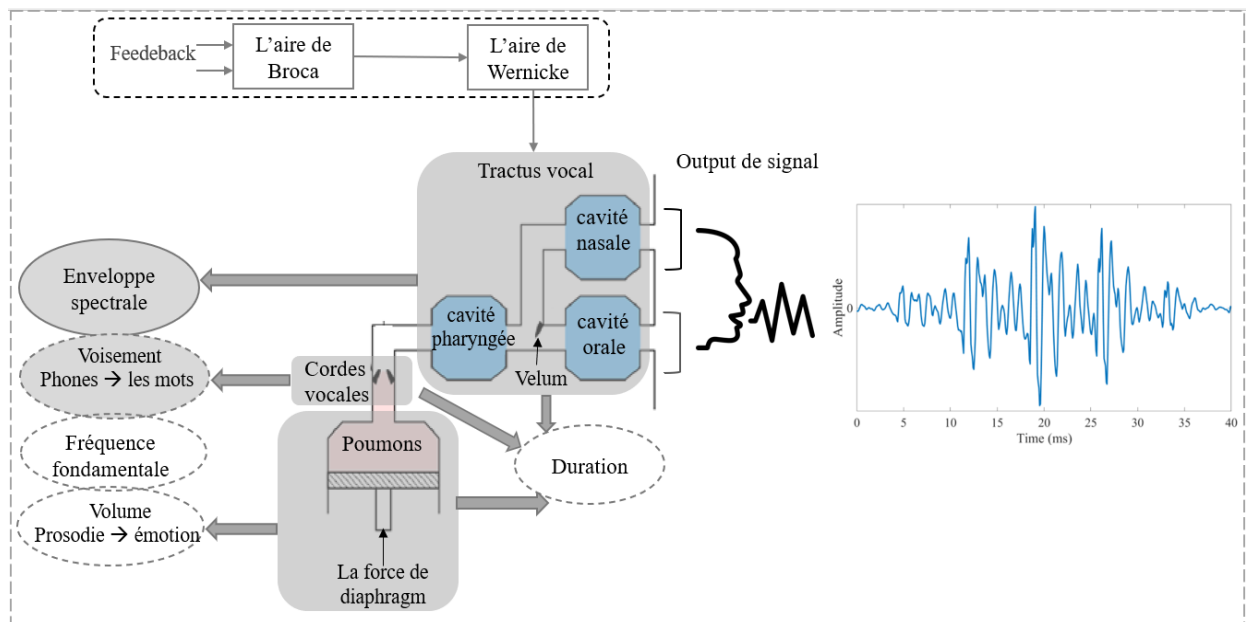


Figure 8. Schéma fonctionnel de système de génération et de production d'un signal vocal.

De manière générale, le signal de parole est le résultat de modification de la pression de l'air par le système articuloire accompagné d'un transfert d'énergie entre la source et la destination. C'est une séquence de sons caractérisée par un contenu fréquentiel variant dans le temps. En outre, un signal vocal est un signal continu, non stationnaire, mais il peut être considéré comme stationnaire sur de courts segments de temps (environ 20 ms) où les paramètres principaux varient en petites quantités et peuvent être déterminés par différentes méthodes.

L'enveloppe spectrale d'un signal est une courbe dans le plan fréquence/amplitude, dérivée d'un spectre de Fourier. Elle décrit un point dans le temps et elle est étroitement ajustée au spectre d'amplitude reliant les pics. Une certaine régularité de la courbe est requise ; cela signifie que l'enveloppe spectrale ne doit pas trop osciller, mais elle doit donner une idée générale de la répartition de l'énergie du signal sur la fréquence. La figure 9 montre une enveloppe spectrale avec des maximaux aux fréquences de résonance du conduit vocal appelées formants, qui sont les caractéristiques des différents phonèmes (les plus petites unités présentées dans la parole).

Lors de la production de la parole, les signaux peuvent être voisés ou non voisés selon la vibration ou non des cordes vocales (figure 10). Le voisement est une caractéristique binaire importante pour distinguer les différents phonèmes. Dans les sons voisés, les cordes vocales vibrent à une fréquence appelée fréquence fondamentale (également appelée F_0 ou pitch). Cette fréquence dépend de la tension appliquée aux cordes vocales et du flux d'air produit par les poumons, elle peut être modulée pour donner à la phrase une certaine intonation, constituant ainsi l'un des paramètres prosodiques les plus importants pour la détermination des émotions. Les caractéristiques du signal vocal sont fortement influencées par l'âge et le sexe. Comme la figure 11 le montre, nous voyons que les hommes et les femmes présentent une caractéristique différente dans leur voix en devenant plus âgés. Les voix féminines ont une valeur F_0 plus élevée, mais elle diminue avec l'âge. En revanche, les hommes ont une valeur F_0 plus élevée à leur jeunesse, puis elle diminue avec l'âge et augmente ensuite après 45 ans. Il est raisonnable de supposer que la combinaison de la perte de contrôle neuronal et musculaire avec l'âge explique un bon nombre des modifications générales de la parole observées chez les personnes âgées.

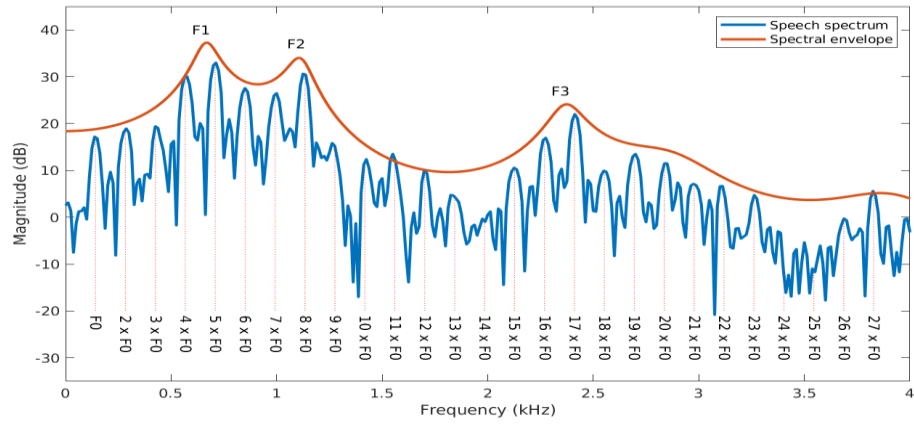


Figure 9. L'enveloppe spectrale d'un segment de parole avec les 3 formants [48].

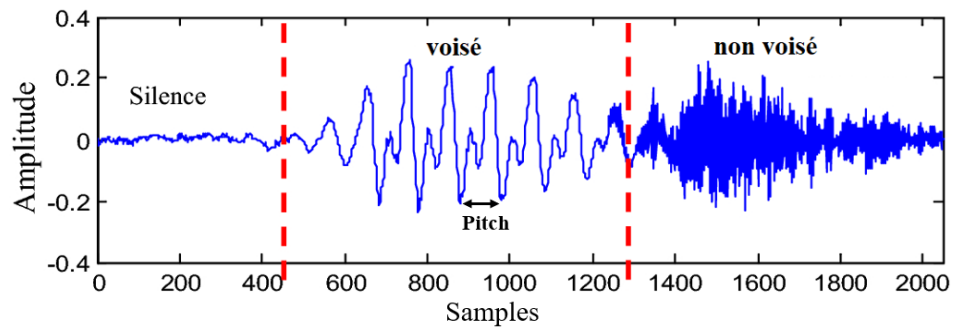


Figure 10. Segment d'un signal vocal contient une partie voisée et une partie non voisée [47].

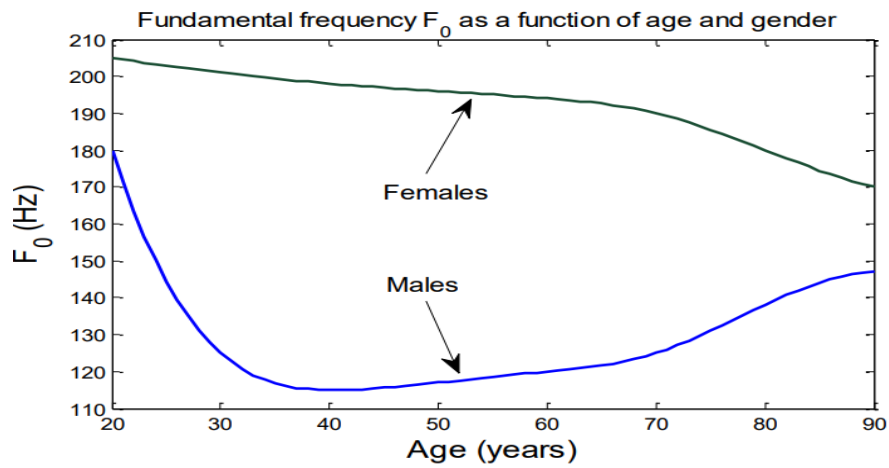


Figure 11. Les changements de F_0 par rapport au sexe et l'âges entre 20 et 90 ans [46].

4.3 Traitement du signal de parole

Ce chapitre présente certaines approches utilisées pour traiter les signaux vocaux afin d'en extraire des informations cliniquement utiles. Ces approches proviennent de la communauté de recherche en matière de traitement des signaux de parole qui ont été développés à diverses fins, notamment l'identification du locuteur, le codage de la parole et la détection de la pathologie.

4.3.1 L'autocorrélation et la corrélation croisée

Plusieurs méthodes classiques de traitement du signal sont largement basées sur des techniques telles que l'autocorrélation à court terme, qui pour un signal (x) à un décalage (lag) donné, est définie comme montré dans l'équation 1, où $(.)^*$ représente le complexe conjugué.

$$R_{xx}(l) = \sum_{n=-\infty}^{\infty} x_n \cdot x_{n-l}^* = x_l \otimes x_{-l}^* \quad (1)$$

L'autocorrélation est un outil permettant de trouver des modèles répétitifs dans un signal. Elle exprime la similarité entre les échantillons en fonction de la différence de leurs indices. L'autocorrélation admet un maximum global à $l = 0$, et la division de $R_{xx}(l)$ par $R_{xx}(0)$ la normalise dans un intervalle de -1 à 1. En conséquence, nous supposons toujours que $-1 \leq R_{xx}(l) \leq 1$.

La corrélation croisée est un concept similaire à celui de l'autocorrélation. Elle s'agit d'une mesure de la similarité entre deux signaux en fonction des retards de l'un des deux :

$$R_{xy}(l) = \sum_{n=-\infty}^{\infty} x_n \cdot y_{n-l}^* = x_l \otimes y_{-l}^* \quad (2)$$

Comme pour l'autocorrélation, nous travaillons avec une estimation normalisée dans un intervalle de -1 à 1, c'est-à-dire que $-1 \leq R_{xy}(l) \leq 1$.

4.3.2 L'analyse fréquentielle

L'analyse fréquentielle est un outil important largement utilisé dans le traitement du signal. Cette méthode représente le signal d'entrée du domaine temporel au domaine fréquentiel à l'aide d'une combinaison linéaire de signaux exponentiels complexes. Les fréquences existantes dans le signal du domaine temporel constituent le spectre.

La représentation d'un signal en temps discret dans le domaine fréquentiel est obtenue à l'aide de la transformée de Fourier en temps discret $F(x)$, qui est définie comme suit :

$$F(x) = X(\omega) = \sum_{n=-\infty}^{\infty} x_n \cdot \exp(-j\omega n) \quad (3)$$

La fonction $F(x)$ est périodique avec une période de 2π , et la fréquence du signal est limitée entre 0 et 2π ; $0 \leq \omega \leq 2\pi$ en raison des effets d'échantillonnage. En pratique, le spectre $X(\omega)$ est évalué à N points de fréquence désignés par $\omega_k = 2k\pi/N$ où $1 \leq k \leq N$. Le signal dans le domaine temporel est exprimé en fonction de k :

$$F(x) = X(k) = \sum_{n=1}^N x_n \cdot \exp\left(-j\frac{2k\pi}{N}n\right) \quad (4)$$

Nous pouvons récupérer le signal original du domaine temporel à partir du signal du domaine fréquentiel en utilisant la transformée de Fourier inverse :

$$F^{-1}(X(k)) = x = \frac{1}{N} \sum_{k=1}^N X(k) \cdot \exp\left(j\frac{2k\pi}{N}n\right) \quad (5)$$

En fait, il existe une méthode rapide pour calculer l'équation (4), en utilisant la transformée de Fourier rapide (FFT) [55]. Il est intéressant de noter qu'il existe une relation algorithmique entre la FFT et l'autocorrélation car elle peut être calculée à partir de la FFT inverse de la densité spectrale de puissance $|X(\omega)|^2$:

$$R_{xx}(l) = \frac{1}{N} \sum_{n=1}^N |X(\omega)|^2 \cdot \exp\left(-j \frac{2k\pi}{N} n\right), \quad 1 \leq k \leq N. \quad (6)$$

4.3.3 L'estimation de la fréquence fondamentale

Dans le domaine du traitement du signal vocal l'estimation correcte de la fréquence fondamentale est essentielle pour caractériser les signaux de parole. Il existe un grand nombre d'algorithmes permettant d'estimer F_0 , certains chercheurs préfèrent les appeler Algorithmes de Détection de Pitch. Les PDA ont été largement utilisés chez la communauté du traitement du signal vocal. Ils fonctionnent soit dans le domaine temporel (la plupart utilisant l'autocorrélation et certains utilisant des approches de la corrélation croisée), soit dans le domaine fréquentiel (spectre de fréquence et approches cepstrales) ou dans certains cas, qui combinent les informations du domaine temporel et du domaine fréquentiel pour obtenir des estimations plus fiables. Parmi les PDA les plus couramment utilisées, nous mentionnons PRAAT et RAPT.

4.3.3.1 PRAAT

Il s'agit d'un algorithme d'estimation de la fréquence fondamentale, initialement proposé par Boersma [73]. Il se présente comme une approche dans le domaine temporel, reposant sur la fonction d'autocorrélation pour le calcul de la F_0 . L'idée principale de PRAAT est la suivante : tout d'abord le fenêtrage est appliqué au signal. Typiquement, une fenêtre temporelle de 40-80 ms est généralement superposée au signal vocal (x) en le divisant en N segments (trames), où $N \geq \frac{\text{La durée du signal}}{\text{La fenêtre temporelle}}$. Ensuite, pour chacun des N segments, la valeur moyenne est soustraite et le signal résultant (s) est multiplié par une fenêtre W appropriée (la fenêtre de Hanning ou gaussienne), ce qui donne N signaux de fenêtre a . En fin l'autocorrélation de chaque segment de signal est calculée comme suit : $r(l) \cong \frac{R_{aa}(l)/\max(R_{aa}(l))}{R_{ww}(l)/\max(R_{ww}(l))}$, où $R_{aa}(l)/\max(R_{aa}(l))$ est l'autocorrélation normalisée du signal de la fenêtre a , et $R_{ww}(l)/\max(R_{ww}(l))$ est l'autocorrélation normalisée de la fenêtre W , sachant que $\max(R_{aa}(l))$ et $\max(R_{ww}(l))$ sont les instants auxquels l'autocorrélation la plus élevée est trouvée. L'indice l se situe entre une valeur maximale et une valeur minimale de la fréquence fondamentale, où :

$$\frac{F_s}{F_{0,max}} \leq l_{max} \leq \frac{F_s}{F_{0,min}}.$$

La bibliothèque Parselmouth [49] peut être utilisé pour accéder à l'algorithme PRAAT qui accompagne cette manipulation, à partir de Python. La figure 13 montre la représentation de la forme d'onde d'un enregistrement audio de 4 secondes d'une voyelle soutenue /a/ de la base de données PC-GITA. Pour la figure 14 nous avons utilisé la bibliothèque parselmouth pour calculer les fréquences maximales et minimales de pitch de la partie de signal sélectionné (entre 2.3 et 2.45 secondes). Nous avons obtenu une valeur maximale de $F_0 = 126.32$ Hz, à l'instant $t = 2.325$ seconds, et une valeur minimale de $F_0 = 117.42$ Hz, à l'instant $t = 2.425$ seconds. Finalement, la figure 15 montre l'autocorrélation temporelle du segment de signal analysé. On observe que ce signal est fortement corrélé. La valeur de l'autocorrélation est très élevée. En raison de la quasi-

stationnarité du signal sur des partitions à court terme, nous pouvons également voir les deux pics dominants secondaires.

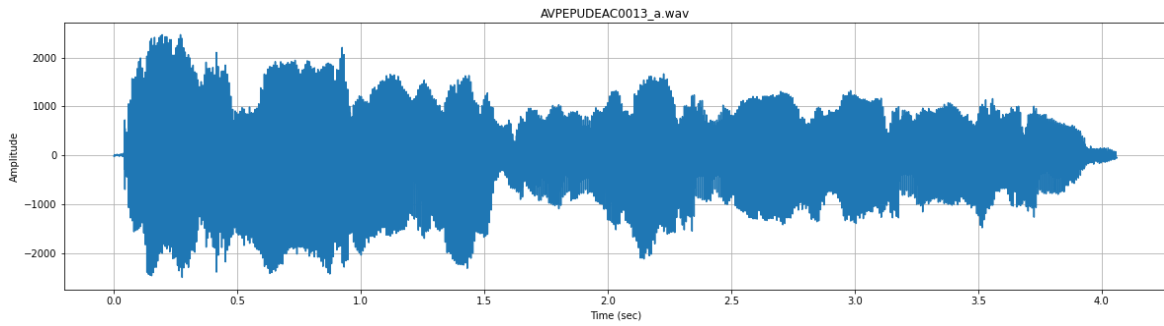


Figure 12. La représentation de la forme d'onde de l'enregistrement AVPEPUDEAC0013_a.wav.

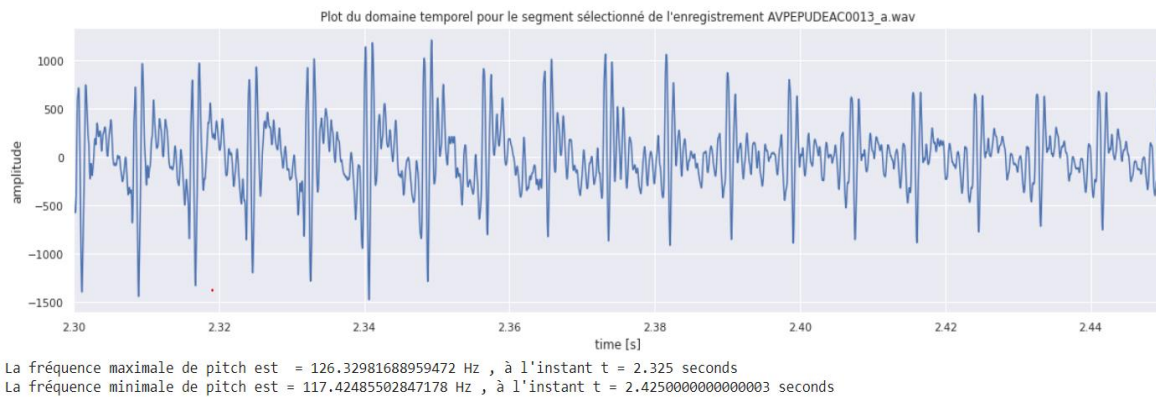


Figure 13. L'amplitude de segment du signal sélectionné avec le calcul de F0 min et F0 max.

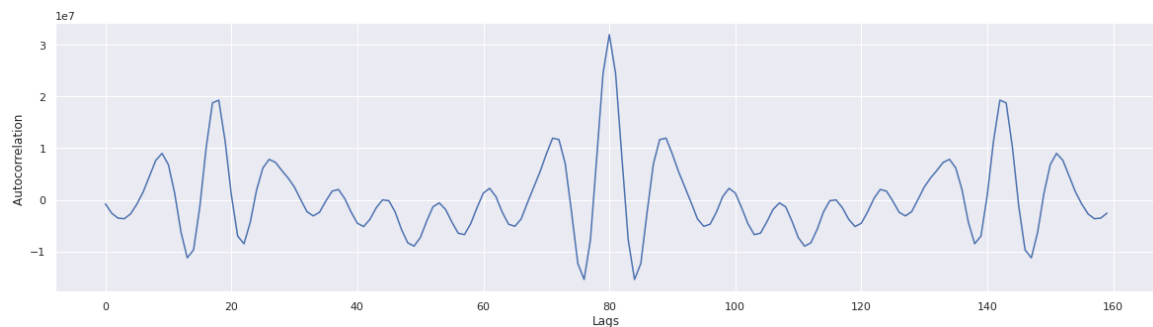


Figure 14. L'autocorrélation temporelle du segment de signal analysé.

4.3.3.2 L'algorithme robuste pour le suivi de pitch (RAPT)

RAPT est l'abréviation de Robust Algorithm for Pitch Tracking (RAPT). C'est un PDA de domaine temporel qui a été initialement proposé par Talkin [50] et s'appuie sur la fonction de corrélation croisée pour estimer la fréquence fondamentale. RAPT travaille dans des fenêtres temporelles segmentées du signal original, exactement comme expliqué dans PRAAT afin de fournir des estimations de F_0 pour chacune de ces trames. En pratique, il compare le signal vocal original avec une version sous-échantillonnée de signal générée qui porte une nouvelle fréquence d'échantillonnage, et tente d'identifier les maxima où la corrélation croisée est proche de 1 (à l'exception du point à zéro décalage). Il calcule la corrélation croisée du signal à faible taux d'échantillonnage pour les décalages qui se situent dans la plage F_0 ($[F_{0,min} F_{0,max}]$) et enregistre les positions des maxima de la trame examinée qui ont une valeur de corrélation croisée supérieure à un certain seuil. Ensuite, la corrélation croisée est calculée sur le signal vocal brut original à proximité des maxima prometteurs identifiés à l'étape précédente. Cette approche en deux étapes est conçue pour réduire la charge de calcul de RAPT. Les pics identifiés à partir de ce

calcul de corrélation croisée à haute résolution (deuxième étape) correspondent aux candidats F0 pour cette trame. Après avoir identifié les temps de décalage où se trouvent les candidats F0 potentiels pour chaque trame, RAPT utilise ensuite un post-traitement pour déterminer les candidats les plus probables. Plus précisément, il utilise la programmation dynamique [51] pour pénaliser certaines quantités, comme le coût de la transition F0 pour des trames successives. Une fois que le point de décalage le plus probable est déterminé, RAPT précise l'estimation de la F0 par interpolation parabolique des trois points de retard autour du pic déterminé. Pour l'implémentation de RAPT en fait appel à l'outil python SPTK (Speech Signal Processing Toolkit) [52] pour l'estimation de la fréquence fondamentale des enregistrements audio comme montré dans la figure 16.

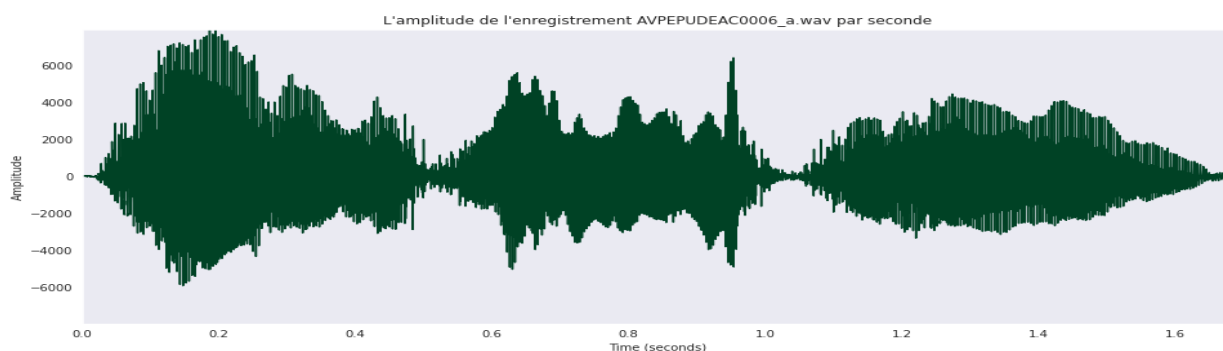


Figure 15. L'amplitude de l'enregistrement AVPEPUDEAC0006_a.wav dans le domaine temporel.

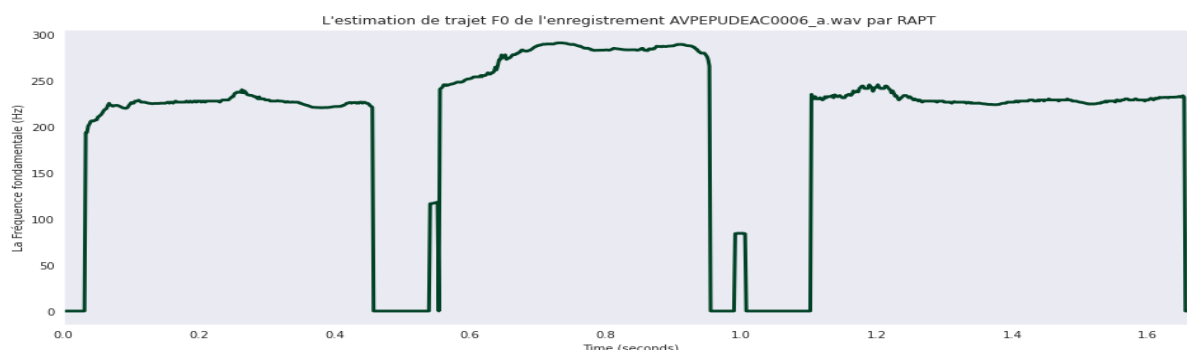


Figure 16. L'estimation de F0 de l'enregistrement AVPEPUDEAC0006_a.wav par RAPT

Dans cette représentation graphique de F0, il est clair de voir où l'audio est voisé et où il est non voisé. Pendant certaines périodes correspondantes à la parole voisée, la fréquence fondamentale se situe entre 200 et 300 Hz et à d'autres moments, pour la parole non voisée, elle se situe à 0 Hz.

4.4 Les caractéristiques extraites du signal de parole

Les paramètres caractéristiques sont nombreux dans le signal de parole et les différents vecteurs caractéristiques représentent des significations distinctes. Du coup, le choix de ces caractéristiques est d'une grande importance pour le succès ou non d'un modèle de détection de la voix pathologique.

La communauté de recherche a évalué les troubles de la parole à l'aide de mesures classiques et linéaires. Bien que ces méthodes soient encore largement applicables et extrêmement utiles, les approches non linéaires ont été également appliquées aux signaux de parole [53] et ils ont donné des résultats prometteurs [24].

Dans la section suivante, nous passons en revue certains mesures les plus populaires pour la détection de la MP, ainsi que leurs implémentations algorithmiques détaillée.

4.4.1 Les variantes de jitter

Le jitter est une mesure principalement affectée par le manque de contrôle de la vibration des cordes vocales. Cette mesure vise à quantifier les perturbations cycle par cycle de la fréquence fondamentale, représentées par de petites déviations de la périodicité exacte. Il peut être calculé en utilisant soit le contour F_0 , soit la période de pitch inversement proportionnelle $T_0 = 1/F_0$.

Les mesures liées au jitter qui permettent également de mesurer la perturbation sont le jitter absolu, le jitter local (%), le jitter RAP (relative Average Perturbation), le jitter PPQ5 (5-point Period Perturbation Quotient), et le Jitter DDP (Difference of Differences of Periods) qui indique la différence absolue moyenne des différences entre les cycles de jitter.

Le jitter absolu est défini comme la différence absolue moyenne des estimations de F_0 entre les cycles successifs [54] :

$$Jitter_{Abs} = \frac{1}{N} \sum_{i=1}^{N-1} |F_{0,i} - F_{0,i+1}| \quad (7)$$

Où F_i est la fréquence du $i^{\text{ème}}$ intervalle (période) et N est le nombre des computations de F_0 .

Le jitter local est la mesure de jitter la plus courante. Il est généralement exprimé en pourcentage et représente la différence absolue divisée par la moyenne F_0 [54] :

$$Jitter_{local} = 100 \cdot \frac{\frac{1}{N} \sum_{i=1}^{N-1} |F_{0,i} - F_{0,i+1}|}{\frac{1}{N} \sum_{i=1}^N F_{0,i}} \quad (8)$$

Le jitter RAP se calcule en prenant la moyenne des différences entre une période T_i et la moyenne des 3 périodes T_{i-1} , T_i et T_{i+1} , divisée par la période moyenne [54] :

$$Jitter_{rap} = \frac{\frac{1}{N} \sum_{i=1}^{N-1} \left| F_{0,i} - \left(\frac{1}{3} \sum_{j=i-1}^{i+1} F_{0,j} \right) \right|}{\frac{1}{N} \sum_{i=1}^N F_{0,i}} \quad (9)$$

Le jitter PPQ5 se calcule de la même façon que le jitter RAP mais en prenant la moyenne de 5 périodes consécutives au lieu de 3 :

$$Jitter_{ppq5} = \frac{\frac{1}{N} \sum_{i=2}^{N-2} \left| F_{0,i} - \left(\frac{1}{5} \sum_{j=i-2}^{i+2} F_{0,j} \right) \right|}{\frac{1}{N} \sum_{i=1}^N F_{0,i}} \quad (10)$$

Finalement, le jitter DDP s'agit de la différence entre deux différences consécutives de deux périodes consécutives, divisée par la période moyenne.

$$Jitter_{ddp} = \frac{\sum_{i=2}^{N-1} |(F_{0,i+1} - F_{0,i}) - (F_{0,i} - F_{0,i+1})| / (N-2)}{\frac{1}{N} \sum_{i=1}^N F_{0,i}} \quad (11)$$

4.4.2 Les variantes de Shimmer

Il s'agit d'une mesure liée à la diminution de la résistance glottique et à la lésion des cordes vocales. Généralement c'est une mesure de l'instabilité de l'amplitude. Le shimmer et ses variantes sont obtenus en mesurant la valeur maximale de l'amplitude du signal dans chaque cycle vocal. Les mesures liées au shimmer qui permettent également de mesurer l'instabilité de l'amplitude d'un signal de parole sont :

Le shimmer (db) s'agit du logarithme absolu moyen en base 10 de la différence entre les amplitudes de périodes consécutives. Il est calculé comme suit [54] :

$$Shimmer_{db} = \frac{1}{M-1} \sum_{i=1}^{M-1} 20 * \log \frac{V(i+1)}{V_i} \quad (12)$$

Où V_i est l'amplitude crête à crête et M le nombre de périodes de fréquence fondamentale. L'amplitude crête à crête est la différence entre l'amplitude maximale et l'amplitude minimale.

Le shimmer local s'agit de la différence absolue moyenne entre les amplitudes de périodes consécutives, divisée par l'amplitude moyenne. Il est calculé comme montré dans l'équation (13) :

$$Shimmer_{local} = \frac{\frac{1}{M-1} \sum_{i=1}^{M-1} |V_i - V(i+1)|}{\frac{1}{M} \sum_{i=1}^M V_i} \quad (13)$$

Le shimmer APQ3 (3-point Amplitude Perturbation Quotient) s'agit de la différence absolue moyenne entre l'amplitude d'une période et la moyenne des amplitudes de ses voisins, divisée par l'amplitude moyenne.

$$Shimmer_{apq3} = \frac{\frac{1}{M-1} \sum_{i=1}^{M-1} \left| V_i - \left(\frac{1}{3} \sum_{j=i-1}^{i+1} V_j \right) \right|}{\frac{1}{M} \sum_{i=1}^M V_i} \quad (14)$$

Le shimmer APQ5 représente le rapport d'amplitude de perturbation de cinq périodes. En d'autres termes, la différence absolue moyenne entre l'amplitude d'une période et les amplitudes moyennes de celle-ci et de ses quatre plus proches voisins, divisée par l'amplitude moyenne.

$$Shimmer_{apq5} = \frac{\frac{1}{M-1} \sum_{i=1}^{M-1} \left| V_i - \left(\frac{1}{5} \sum_{j=i-1}^{i+1} V_j \right) \right|}{\frac{1}{M} \sum_{i=1}^M V_i} \quad (15)$$

Le shimmer APQ11 s'agit de la différence absolue moyenne entre l'amplitude d'une période et la moyenne des amplitudes de celle-ci et de ses dix voisins les plus proches, divisée par l'amplitude moyenne.

$$Shimmer_{apq11} = \frac{\frac{1}{M-1} \sum_{i=1}^{M-1} \left| V_i - \left(\frac{1}{11} \sum_{j=i-1}^{i+1} V_j \right) \right|}{\frac{1}{M} \sum_{i=1}^M V_i} \quad (16)$$

Les équations de deux shimmers APQ5 et APQ11 sont les mêmes que l'équation (14) de shimmer APQ3 mais en prenant respectivement la moyenne de 5 et 11 amplitudes consécutives au lieu de 3.

4.4.3 Le rapport harmoniques/bruit

Le HNR (Harmonic to Noise Ratio) visent à quantifier le bruit dans un signal de parole qui est généralement produit par une fermeture incomplète des cordes vocale. Il fournit une indication de la périodicité globale du signal en quantifiant le rapport entre les composantes périodiques (partie harmonique) et apériodiques (bruit). Une méthode de calcul du HNR proposé par Shaheen et al. [56] dépend de l'identification des marqueurs de limite de pitch appropriés, dont l'énergie acoustique de la composante harmonique H est estimée de la manière décrite par Yumoto [59]:

$$H = N * \int_0^T f_A^2(t) dt \quad (0 \leq t \leq T) \quad (17)$$

Où, N est le nombre de cycles utilisés pour l'analyse ; T égal à la longueur du cycle, et f_A est calculé comme montré ci-dessous :

$$f_A = \sum_{i=1}^N f_i(t)/N \quad (18)$$

Cependant la composante de bruit est définie comme suit :

$$N = \sum_{i=1}^N \int_0^T [f_i(t) - f_A(t)]^2 dt \quad (19)$$

Finalement le HNR est rapporté sur une échelle de décibels, en utilisant l'expression suivant :

$$HNR = 10 * \log_{10}(H/N) \quad (20)$$

4.4.4 Les formants

Les fréquences formants sont parmi les mesures acoustiques de la parole les plus fréquemment rapportées, ils sont utilisés dans une variété d'applications, y compris la reconnaissance automatique de la parole, les études de la production et de la perception de la parole, et les évaluations cliniques des troubles de la parole. Les formants sont des fréquences de résonance du conduit vocal qui apparaissent comme des pics clairs dans le spectre vocal (comme montré dans la figure 9 de la section 4.3), qui marquent les fréquences de grande concentration d'énergie dans les phonèmes vocaux. La structure des formants se compose principalement de six fréquences de formants [58] étiquetées à partir de la fréquence la plus basse, comptées vers le haut. Les fréquences les plus basses contenant une grande énergie mais moins d'intelligibilité que les fréquences les plus hautes. Le premier formant $F1$ apporte la plus grande énergie et reflètent le style d'expression et la structure de la langue. En revanche, les $F2$, $F3$, $F4$ et plus, sont plus importants pour l'intelligibilité de la parole. Il a été observé que $F2$ et $F3$ sont très sensibles aux deux changements de paramètres reflétant le mouvement détaillé de la langue.

Pour l'analyse spectrale du modèle de formants, les méthodes couramment utilisées sont la transformée de Fourier rapide (FFT) et le codage prédictif linéaire (LPC).

L'analyse prédictive linéaire est l'une des techniques les plus puissantes pour extraire les fréquences des formants. L'importance de cette méthode réside dans sa capacité à fournir des estimations précises et dans sa relative rapidité de calcul. Pour l'estimation des formants, la figure 15 montre les étapes nécessaires de l'algorithme suivie pour l'extraction des formants en utilisant le codage prédictif linéaire (LPC). Dans la première étape, le blocage de trame ou cadrage est une technique fondamentale de traitement du signal qui consiste à diviser le signal d'origine en blocs ou trames de courte durée. La deuxième étape consiste à appliquer la fenêtre de Hamming aux trames sélectionnées, puis la procédure de LPC avec l'utilisation de la méthode d'autocorrélation pour calculer les coefficients prédictifs linéaires a été effectuer. Ensuite, le calcul de racines des

polynômes prédictifs pour trouver les emplacements des trames dans les spectres des filtres prédictifs linéaires. Dans cet étape seul les racines avec des angles positifs ont été prêtes en considération, c'est-à-dire $r = \text{imaginaire}(r) > 0,01$, car elles se présentent sous forme de paires conjuguées complexes. L'étape suivante est de convertir les angles en fréquences de façon mathématique en utilisant l'équation (21).

$$F_i = \tan^{-1} \left(\frac{\text{imaginaire}(r)}{\text{réelle}(r)} \right) * f_s / (2 * \pi) \quad (21)$$

Où F_i est le formant $i^{\text{ème}}$ en Hz ; (r) est la racine et f_s est la fréquence d'échantillonnage.

Finalement, les formants F_i sont triés par ordre croissant. La fréquence la plus basse devenant $F1$, la deuxième devenant $F2$ et ainsi de suite jusqu'au le quatrième formant $F4$.

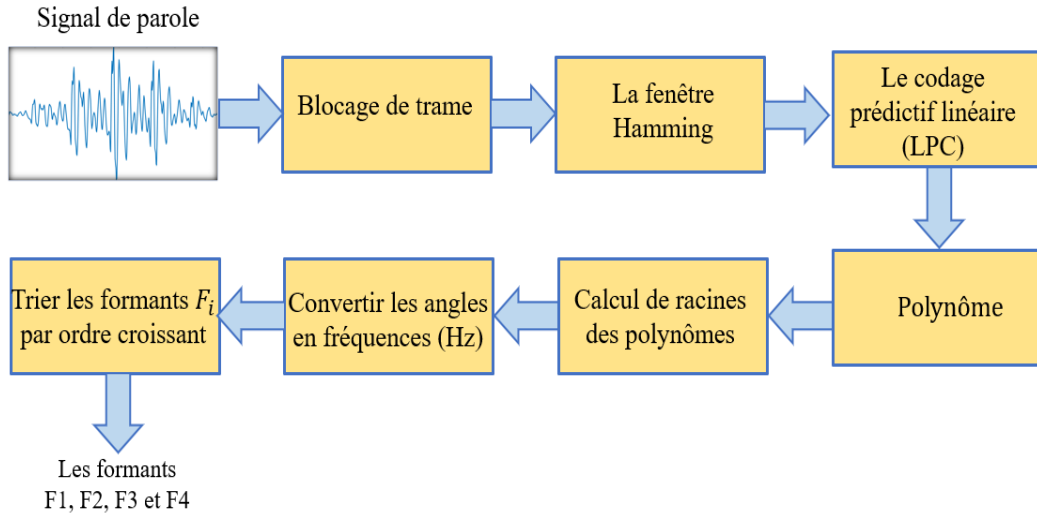


Figure 17. Diagramme du LPC pour l'extraction des formants.

4.4.5 L'analyse de fluctuation détendue (DFA)

La DFA (detrended fluctuation analysis) est une mesure qui vise à caractériser le degré de bruit dans le signal vocal en quantifiant l'autosimilarité stochastique du bruit produit par le flux d'air turbulent dans le conduit vocal. L'une des principales raisons d'utiliser la DFA est qu'elle évite de détecter les structures de corrélation qui sont en fait des artefacts causés par la non-stationnarité potentiellement des séries temporelles. Nous décrivons ici la méthode DFA (équation 23) en utilisant la définition proposée par Little et al [28]. Son concept se concentre sur la composante stochastique du signal de parole et vise à caractériser son exposant d'échelle. Pour ce faire, des lignes droites sont ajustées sur de petits intervalles de temps de longueur L , et la fluctuation moyenne $F(L)$ du signal (équation 22) est mesurée par rapport aux lignes droites dans cet intervalle en utilisant la mesure de la racine carrée moyenne. Ensuite, l'algorithme ajuste une ligne droite à l'ensemble $\{\log L, \log F(L)\}$ pour différentes valeurs de L en utilisant la régression linéaire par la méthode des moindres carrés. Au départ, l'algorithme intègre les échantillons du signal pour induire une autosimilarité dans le signal : $s_n = \sum_{i=1}^n x_i$, où $n = 1 \dots N$, et N est la longueur du signal de parole x . Alors pour chaque intervalle, un meilleur ajustement au sens des moindres carrés est déterminé pour la fenêtre de longueur L : $\argmin_{a,b} E^2 = \sum_{n=1}^L (s_n - a_n - b)^2$, où a est la pente et b est l'ordonnée à l'origine (le point où la ligne croise l'axe). Alors, la fluctuation est:

$$F(L) = \left[\frac{1}{L} \sum_{n=1}^L (s_n - a_n - b)^2 \right]^{1/2} \quad (22)$$

L'exposant γ est déterminé à partir du tracé logarithmique de L par rapport à $F(L)$ en ajustant une ligne droite, et il est normalisé pour se situer dans l'intervalle 0-1, en utilisant la fonction logistique : $f(x) = (1 + \exp(-x))^{-1}$.

$$DFA = \frac{1}{1 + \exp(-\gamma)} \quad (23)$$

4.4.6 L'entropie de la période du pitch

L'entropie de la fréquence fondamentale PPE mesure la déficience du contrôle de la stabilité du pitch lors de la phonations soutenues, qui est un symptôme courant chez les personnes atteintes de la MP. En outre, il est important de mentionner que cette mesure utilise une échelle de pitch logarithmique et qu'elle est robuste aux facteurs de confusion omniprésents tels que le vibrato (une légère oscillation de la fréquence du son fondamental), qui est présent dans les voix saines et pathologiques [60]. L'avantage d'utiliser une échelle de pitch logarithmique, plutôt qu'une échelle de fréquence absolue, est sa pertinence pour évaluer les variations anormales de pitch de la parole [61]. Pour plus de détails sur l'algorithme de PPE (équation 24) veuillez-vous référer à Little et al. [62]. Généralement, la fréquence fondamentale estimée est convertie en demi-ton logarithme (perceptif). Puis le contour perceptuel de F0 est filtré pour aplanir le spectre de la série de demi-tons et éliminer l'effet du demi-ton moyen, ce qui donne lieu à la série qui caractérise l'occurrence des variations de demi-ton r . Ensuite, la densité de probabilité des valeurs de demi-ton $p(r)$ est obtenue et exprimée en utilisant le concept d'entropie ci-dessous, où L_{ppe} est la longueur des points utilisés pour calculer la mesure de l'écart.

$$PPE = \frac{-\sum_i^{L_{ppe}} p(i) \cdot \ln(p(i))}{\ln(L_{ppe})} \quad (24)$$

4.5 L'apprentissage automatique

L'apprentissage automatique est un domaine de l'intelligence artificielle (IA) qui se concentre sur l'utilisation des données et des algorithmes pour imiter la façon dont les humains se renseignent dans le but d'améliorer régulièrement la précision. C'est un élément crucial de la discipline de science des données. Les algorithmes sont formés pour générer des classifications ou des prédictions à l'aide d'approches statistiques, révélant ainsi des informations déterminantes qui favorisent la prise de décision. Il existe de nombreux types d'algorithmes d'apprentissage automatique qui sont sollicités dans des cas d'utilisation très spécifiques, mais deux d'entre eux sont les plus utilisés aujourd'hui ; l'apprentissage automatique supervisé et non supervisé.

4.5.1 L'apprentissage automatique supervisé

L'apprentissage supervisé est défini comme l'utilisation d'ensembles de données étiquetés pour former des algorithmes afin de classer les données ou prédire les résultats. Lorsque les données d'entrée sont introduites dans le modèle, celui-là ajuste ses pondérations via un processus

d'apprentissage par renforcement, garantissant que le modèle est correctement ajusté. L'apprentissage supervisé est basé sur l'entraînement c'est-à-dire qu'au cours de sa phase d'apprentissage, le système est alimenté avec des ensembles de données étiquetées, qui indiquent quelle sortie est liée à chaque valeur d'entrée spécifique. Ensuite, le modèle formé est présenté avec des données de test : il s'agit de données qui ont été étiquetées, mais n'ont pas été révélées à l'algorithme. L'objectif des données de test est de mesurer la précision avec laquelle l'algorithme fonctionnera sur des données non étiquetées. Les algorithmes d'apprentissage supervisé génèrent principalement deux types de résultats : la classification et la régression.

Les algorithmes de classification visent à trier les entrées dans un nombre donné de catégories ou de classes, en fonction des données étiquetées sur lesquelles il a été formé. Ils peuvent être utilisés pour des classifications binaires telles que la reconnaissance de caractéristiques, comme la reconnaissance de lettres et de chiffres manuscrits ou la classification de la pathologie. Cependant que les modèles de régression s'attendent à ce que le modèle produise une relation numérique entre les données d'entrée et de sortie. Ils existe plusieurs algorithmes d'apprentissage supervisé nous citons ici les suivants :

4.5.1.1 Machine à vecteurs de support (SVM)

Les machines à vecteurs de support sont considérées comme une approche de classification, mais peuvent être utilisées dans les deux types de problèmes de classification et de régression. Elles construisent un hyperplan dans un espace multidimensionnel pour séparer différentes classes. Elles génèrent un hyperplan optimal de manière itérative, qui est utilisé pour minimiser une erreur. L'idée principale de SVM est de trouver un hyperplan marginal maximal qui divise l'ensemble de données en classes, comme le montre la figure 18. Les vecteurs de support sont les points de données les plus proches de l'hyperplan. Ces points définiront mieux la ligne de séparation en calculant des marges. Ils sont plus pertinents pour la construction du classifieur.

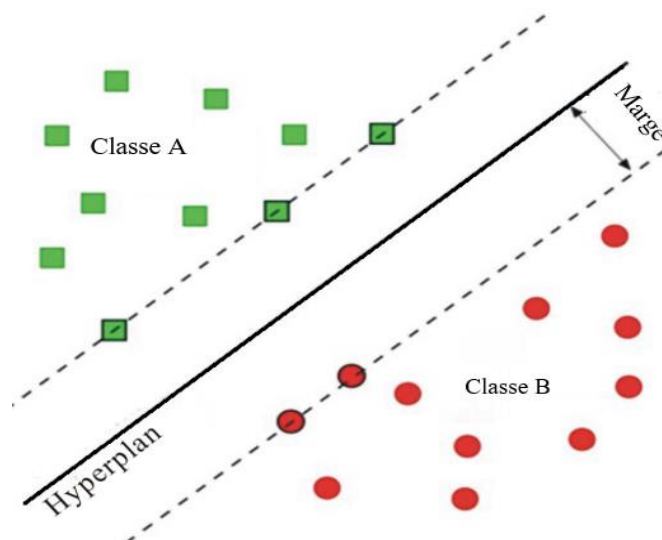


Figure 18. Principe de l'algorithme SVM

L'algorithme de SVM est implémenté en pratique à l'aide d'un noyau qui transforme un espace de données d'entrée dans la forme requise. Il utilise une technique appelée l'astuce du noyau qui est utile dans le problème de séparation non linéaire et aide à créer un classificateur plus précis. Nous mentionnons ici trois types des noyaux ; linéaire, polynomial et RBF.

Le noyau linéaire peut être utilisé comme produit scalaire normal pour deux observations données. Le produit entre deux vecteurs est la somme de la multiplication de chaque paire de valeurs d'entrée. Le noyau polynomial est une forme plus généralisée du noyau linéaire. Il peut distinguer les espaces d'entrée non linéaires où le degré du polynôme. Le noyau de la fonction de base radiale (RBF) est une fonction noyau populaire couramment utilisée dans la classification par

machine à vecteurs de support qui peut représenter un espace d'entrée dans un espace de dimension infinie.

4.5.1.2 Forêt aléatoire (RF)

Random Forest en anglais, c'est un algorithme d'apprentissage automatique supervisé qui est construit à partir d'algorithmes d'arbre de décision. La forêt aléatoire est une technique d'apprentissage automatique utilisée pour résoudre des problèmes de régression et de classification. Il utilise l'apprentissage d'ensemble, qui est une technique qui combine de nombreux classificateurs pour fournir des solutions à des problèmes complexes. Un algorithme de forêt aléatoire se compose de plusieurs arbres de décision. La "forêt" générée par l'algorithme de forêt aléatoire est entraînée par 'bagging' ou 'bootstrap aggregating'. Le bagging est un méta-algorithme d'ensemble qui améliore la précision des algorithmes d'apprentissage automatique.

L'algorithme de forêt aléatoire établit le résultat en fonction des prédictions des arbres de décision. Il prédit en prenant la moyenne de la sortie de divers arbres. L'augmentation du nombre d'arbres accroît la précision du résultat. De plus la forêt aléatoire élimine les limites d'un algorithme d'arbre de décision, il réduit aussi le surajustement des ensembles de données et augmente la précision. Il génère des prédictions sans nécessiter de nombreuses configurations dans les packages comme dans [scikit-learn](#).

4.5.1.3 Arbre de décision (DT)

L'arbre de décision est l'un des algorithmes d'apprentissage automatique qui peuvent être utilisés pour traiter les problèmes de la classification et qui peut être également utilisé pour résoudre les problèmes de régression. Il est également appelé CART (qui signifie arbres de classification et de régression). Chaque DT comprend un nœud racine, des branches et des nœuds feuilles. Les nœuds internes présents dans l'arborescence décrivent les différents cas de test.

Les arbres de décision appliquent une approche descendante à l'ensemble de données alimentés au modèle pendant l'entraînement suit une approche supervisée. L'idée principale est d'utiliser un arbre de décision pour partitionner l'espace de données en régions denses et régions éparses. L'algorithme continue à diviser l'arbre jusqu'à ce que les données soient suffisamment homogènes. À la fin de l'apprentissage, on obtient un arbre de décision qui peut être utilisé pour faire des prédictions de catégorie optimales. Un terme important dans le développement de cet algorithme est l'entropie. Elle peut être considérée comme la mesure de l'incertitude d'un ensemble de données et sa valeur décrit le degré d'aléatoire d'un nœud particulier. Une telle situation se produit lorsque la marge de différence pour un résultat est très faible et que le modèle n'a donc pas confiance à la précision de la prédiction. Plus l'entropie est élevée, plus le caractère aléatoire de l'ensemble de données est important. Lors de la construction d'un arbre de décision, une entropie faible sera préférée.

4.5.2 L'apprentissage automatique non supervisé

L'apprentissage non supervisé fait référence à l'utilisation d'algorithmes d'intelligence artificielle pour identifier des modèles dans des ensembles de données qui ne sont ni classifiés ni étiquetés. Les algorithmes sont ainsi autorisés à classer ou regrouper les points de données contenus dans les ensembles de données sans avoir aucune orientation externe dans l'exécution de cette tâche. En d'autres termes, il permet au système d'identifier lui-même des modèles dans les ensembles de données. L'algorithme d'apprentissage non supervisé peut être classé en deux types de problèmes ; le clustering et l'association.

Le clustering est une méthode de regroupement des objets en clusters, de sorte que les objets présentant plus de similarités se retrouvent dans un groupe et que les objets de l'autre groupe

présentent moins de similarités. L'analyse en cluster trouve les points communs entre les objets de données et les catégorise en fonction de la présence ou de l'absence de ces points communs. Cependant qu'une règle d'association est une méthode d'apprentissage non supervisée qui est utilisée pour trouver les relations entre les variables dans la grande base de données. Il détermine l'ensemble des éléments qui apparaissent conjointement dans l'ensemble de données.

4.5.2.1 L'analyse en composantes principales

L'analyse en composantes principales est un algorithme d'apprentissage non supervisé utilisé pour la réduction de la dimensionnalité d'un ensemble de données. Il s'agit d'un processus statistique qui convertit les observations de caractéristiques corrélées en un ensemble de caractéristiques linéairement non corrélées à l'aide d'une transformation orthogonale. Ces nouvelles fonctionnalités transformées sont appelées les composants principaux. C'est l'un des outils populaires utilisés pour l'analyse des données et la modélisation prédictive. Il s'agit d'une technique permettant de tirer des modèles solides à partir d'un ensemble de données en réduisant les variances. Généralement PCA fonctionne en considérant la variance de chaque attribut car l'attribut élevé montre la bonne répartition entre les classes, et donc réduit la dimensionnalité.

L'algorithme de PCA est basé sur certaines étapes nécessaires. Tout d'abord, il est important de diviser l'ensemble de données d'entrée en deux parties, X et Y, où X est l'ensemble d'apprentissage et Y est l'ensemble de validation. Puis, on représente les données dans une structure où chaque ligne correspond aux éléments de données et les colonnes correspondent aux caractéristiques. On arrive ensuite à la normalisation. Cette étape consiste à normaliser l'ensemble de données pour garantir que chaque caractéristique a une moyenne = 0 et une variance = 1. A ce stade, une matrice carrée est construite pour exprimer la corrélation entre deux caractéristiques ou plus dans un ensemble de données multidimensionnel. Après, on doit trouver les vecteurs propres et les valeurs propres qui sont des scalaires par lesquels nous multiplions le vecteur propre de la matrice de covariance. Puis, on trie les vecteurs propres par ordre décroissant et sélectionnez le nombre de composantes principales. En fin, le nouvel ensemble de fonctionnalités a eu lieu, nous allons donc décider ici ce qu'il faut conserver et ce qu'il faut supprimer. Cela signifie que nous conservons les fonctionnalités pertinentes ou importantes dans le nouvel ensemble de données et les fonctionnalités sans importance seront supprimées.

4.6 Analyse de performance des modèles de classification

La tâche la plus importante dans la construction d'un modèle d'apprentissage automatique est d'évaluer ses performances. A ce propos, il existe différentes métriques qui nous permettent d'être en mesure d'améliorer le pouvoir prédictif global de notre modèle avant de le déployer en production sur des données non examinées.

4.6.1 Matrice de confusion

La matrice de confusion est une mesure de performance pour les problèmes de classification d'apprentissage automatique où la sortie peut être de deux classes ou plus. Il s'agit d'un tableau avec des combinaisons de valeurs prévues et réelles, souvent utilisé pour décrire les performances d'un modèle de classification sur un ensemble de données de test pour lesquelles les vraies valeurs sont connues. Notre classificateur veut prédire si une personne a une maladie de Parkinson en fonction des caractéristiques introduits dans le modèle. Alors, il s'agit d'une tâche de classification binaire, de sorte que la personne soit atteinte de la maladie ou non. Le côté gauche de la matrice de prédiction affiche la classe prédite par le classificateur et la ligne supérieure de la matrice stocke les étiquettes de la classe réelles de l'exemple.

		Classe réelle	
		Positif	Négatif
Classe prédite	Positif	Vrai Positif	Faux Positif
	Négatif	Faux Négatif	Vrai Négatif

Figure 19. Matrice de confusion pour une classification binaire

Comme montré dans la figure 19 le nombre de prédictions positives correctes (vrai positif) est situé dans le coin supérieur gauche. En revanche, si le classificateur appelé est positif, mais que l'exemple était en fait négatif, il s'agit d'un faux positif et il se trouve dans le coin supérieur droit.

Le coin inférieur gauche stocke le nombre d'exemples classés comme négatifs mais qui sont en fait positifs, et enfin le coin inférieur droit stocke le nombre de vrais faux exemples ou de vrais négatifs. Peu n'importe le nombre de classes, le principal est toujours le même, le côté gauche correspond aux valeurs prédites et le haut aux valeurs réelles, en vérifiant simplement où ils se croisent pour voir le nombre d'exemples prédits pour une classe donnée par rapport au nombre réel d'exemples pour cette classe. Nous devons également noter que les instances de prédictions correctes descendront en diagonale de haut en bas et de gauche à droite.

Les matrices de confusion comportent deux types d'erreurs : le type 1 et le type 2. Le faux positif est une erreur de type 1 où l'observation est prédite positive mais elle est en fait négative. Cependant que le faux négatif est une erreur de type 2 où le modèle prédit incorrectement la classe négative.

À partir de notre matrice de confusion, nous pouvons calculer plusieurs mesures différentes pour évaluer la validité de notre modèle :

- Le taux de précision (Accuracy) = $\frac{VP+VN}{VP+VN+FP+FN}$
- La mauvaise classification = $\frac{FP+FN}{VP+VN+FP+FN}$
- La précision (vrais positifs/positifs prédits) = $\frac{VP}{VP+FP}$
- Rappel (sensibilité) = $\frac{VP}{VP+FN}$
- Spécificité (vrais négatifs/tous les négatifs réels) = $\frac{VN}{VN+FP}$
- Score F1 = $2 \cdot \frac{\text{Précision} \cdot \text{Rappel}}{\text{Précision} + \text{Rappel}}$
- AUC-ROC — la Caractéristique de l'Opérateur du Récepteur (ROC) est une courbe de probabilité qui trace le TVP (Taux de Vrais Positifs) contre le TFP (Taux de Faux Positifs) à différentes valeurs de seuil et sépare le « signal » du « bruit ».

5 L'implémentation

5.1 Description des ensembles de données

L'interprétation vocale est généralement réalisée à l'aide des phonations de voyelles soutenues, car elles évitent toute confusion, et les informations obtenues sont suffisantes pour évaluer l'étendue des symptômes vocaux. Notre étude se concentre sur des bases de données de la voyelle soutenue /a/, puisqu'elle repose généralement sur l'idée qu'un sujet sain peut produire une phonation stationnaire, alors que pour une personne souffrant d'une déficience vocale, cela peut être impossible. De plus la voyelle soutenue /a/ offre une configuration de conduit vocal dans laquelle la bouche est ouverte au maximum par rapport aux autres voyelles, ce qui minimise l'impulsion d'air reflétée par les cordes vocales. Les deux ensembles des données que nous avons utilisé sont PVA et GITA. Ces deux ensembles de données contiennent des enregistrements audios de type .WAV, un format de fichier audio numérique utilisé pour stocker des données de forme d'onde généralement avec un taux d'échantillonnage de 44,1 kHz et un débit binaire de 16 bits.

5.1.1 La base de données PVA-dryrun

L'ensemble de données PVA est conçu dans le cadre du projet Parkinson's Voice Analysis qui vise à évaluer la gravité et les fluctuations des symptômes de la maladie de Parkinson à partir des enregistrements vocaux. Le tableau 2 contient les informations de l'âge et le sexe des patients, ainsi la collection de 779 enregistrements vocaux avec leurs PDRS score (certains individus ont participé plusieurs fois).

Tableau 2. Aperçu général des données PVA des personnes atteintes de la MP.

Covariables	Valeurs
Individus uniques	620
Enregistrements vocaux avec PDRS	779 (Certains individus ont participé plusieurs fois).
Âge	Min. est 32 ; Médiane est 64 ; Max. est 92
Sexe	377 Féminin; 363 Masculin ; 7 NAN

Sur le site web de Synapse [\[62\]](#) ils ont partagé les fichiers où se trouvent les enregistrements de 620 personnes atteintes de la MP. Le fichier pva_wav-test.zip contient 390 enregistrements, cependant le fichier pva_wav-train.zip contient 389 enregistrements, plus les données des candidats tels que l'âge, le sexe et le score obtenu à partir du questionnaire PDRS (Annexe A) qui a été rempli par les participants. Ce formulaire est une version de l'échelle unifiée d'évaluation de la maladie de Parkinson (UPDRS), il omet la section d'observation clinique, ce qui laisse 17 questions sur les 42 de l'UPDRS. La réponse à chaque question se fait sur une échelle de 0 à 4 ; « 0 » représente une absence du handicap et « 4 » représente un handicap grave. Le questionnaire PDRS comprend aussi l'échelle de Hoehn et Yahr qui décrit également la progression des symptômes de la MP. Les distributions de l'âge, le sexe, le score PDRS, l'échelle H&Y ainsi le nombre d'années depuis le premier symptôme, de notre base de données sont illustrées dans la figure 20.

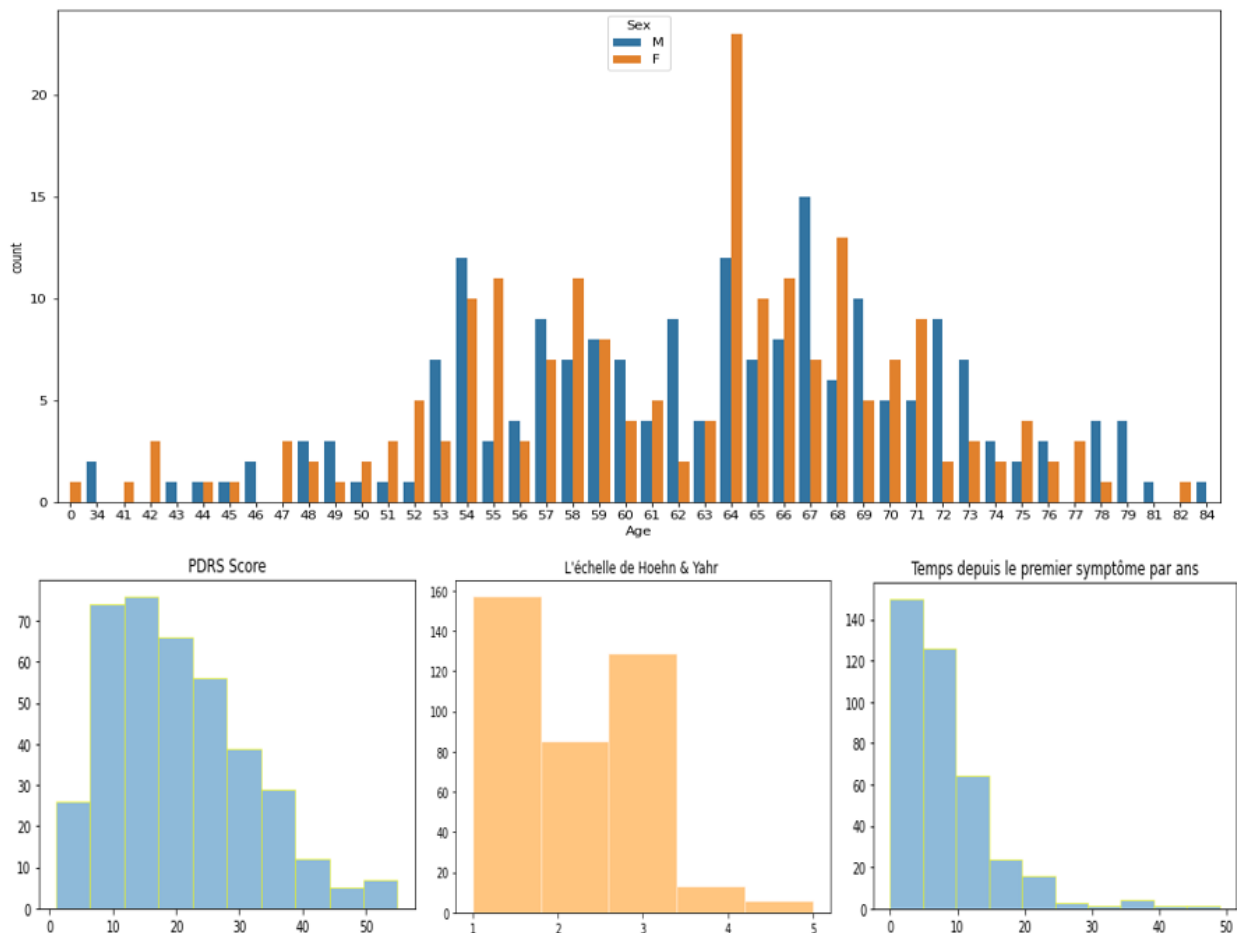


Figure 20. Distribution de l'âge par sexe, les valeurs PDRS, l'échelle de Hoehn & Yahr et le temps après le premier symptôme de la MP (en années).

5.1.2 La base de données GITA.

Le nom GITA provient du groupe de recherche (Grupo de Investigación en Telecomunicaciones Aplicadas) de l'université de Antioquia [69]. La base de données comprend des enregistrements vocaux de 50 personnes atteintes de la MP et 50 sujets sains ; 25 hommes et 25 femmes dans chaque groupe. L'âge des hommes atteints de la MP varie de 33 à 81 ans (moyenne $61,6 \pm 11,2$) et l'âge des femmes atteintes de la MP varie de 49 à 75 ans (moyenne $60,7 \pm 7,3$). Pour les sujets sains, l'âge des hommes varie de 31 à 86 ans (moyenne de $60,3 \pm 11,6$) et celui des femmes de 49 à 76 ans (moyenne de $61,4 \pm 7,0$). Par conséquent, la base de données est bien équilibrée en termes d'âge et de sexe. Les enregistrements ont été réalisés dans des conditions de bruit contrôlées, dans la cabine insonorisée de la Clínica Noel, à Medellín, en Colombie. Tous les patients ont été diagnostiqués par un expert neurologue et ont été étiquetés selon le UPDRS et les échelles de H&Y. Aucune des personnes saines ne présentait des symptômes associés à la MP ou à toute autre maladie neurologique. Les détails de l'âge, des valeurs UPDRS et H&Y, ainsi que le temps écoulé après le premier diagnostic de la MP sont inclus dans le tableau 3. Les distributions de l'âge, du UPDRS, de l'échelle Hoehn & Yahr ainsi que le temps après le diagnostic de la maladie, sont illustrées dans la figure 21.

Tableau 3. Les données cliniques des participants de la base de données GITA [69].

M-MP				MS	F-MP				FS
L'âge	UPDRS	H&Y	t	L'âge	L'âge	UPDRS	H&Y	t	L'âge
81	5	2	12	86	75	52	3	3	75
77	92	5	15	76	73	38	2	4	73
75	13	1	1	71	72	19	2	2.5	76
75	75	3	16	68	70	23	2	12	68
74	40	2.5	12	68	69	19	2	12	65
69	40	3	5	67	66	28	2	4	65
68	14	1	1	67	66	28	2	4	64
68	67	4	20	67	65	54	3	8	63
68	65	3	8	67	64	40	2	3	63
67	28	2	4	65	62	42	3	12	63
65	32	2	12	64	61	21	1	4	63
65	53	2	19	63	60	29	2	7	62
64	28	2	3	63	59	40	2	14	62
64	45	2	3	62	59	71	3	17	61
60	44	3	10	60	58	57	2	1	61
59	6	1	8	59	57	41	3	37	61
57	20	2	0.4	56	57	61	3	17	60
56	30	2	14	55	55	30	2	12	58
54	15	3	4	55	55	43	3	12	57
50	53	2	7	54	55	30	2	12	57
50	19	2	17	51	55	29	2	43	55
48	9	3	12	50	54	30	2	7	55
47	33	2	2	42	51	38	3	41	50
45	21	1	7	42	51	23	2	10	50
33	51	2	9	31	49	53	2	16	49

M-MP : Mâle avec la MP ; MS : Mâle Sain ; F-MP : Femelle avec la MP ; FS : Femelle saine ;
t : temps après le premier diagnostic de la MP (en années).

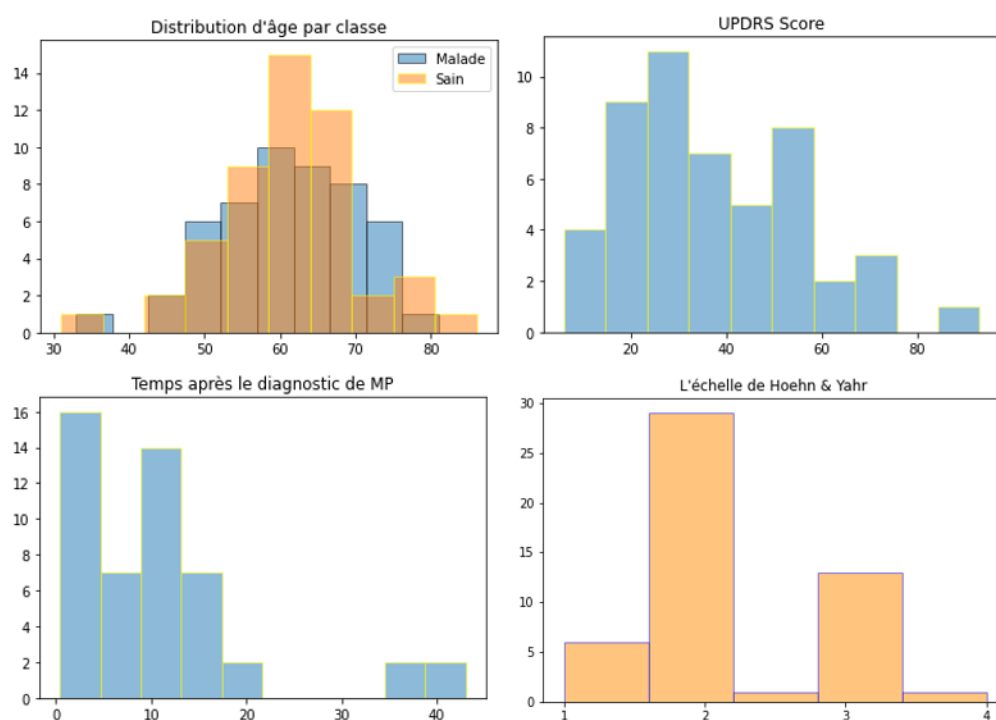


Figure 21. Distribution de l'âge, des valeurs UPDRS, du temps après le diagnostic de la MP (en années), ainsi l'échelle de Hoehn & Yahr de la base de données GITA.

5.1.3 Échelles existantes pour évaluer la MP :

- Échelle de Hoehn & Yahr (H&Y)

Cette échelle a été conçue à l'origine pour fournir une estimation générale de la fonction clinique dans la MP, en combinant les déficits fonctionnels et les signes objectifs. L'échelle comprend un ensemble de cinq stades où 1 est associé à une incapacité fonctionnelle minimale ou nulle et 5 est attribué aux patients qui sont confinés au lit dans l'état grave de la maladie. Il existe deux variantes de l'échelle, l'originale avec des valeurs entières pour les stades de 1 à 5, et la modifiée qui va également de 1 à 5 mais avec des incréments de 0,5. Les stades supérieurs étiquetés avec cette échelle sont en corrélation avec les études de neuro-imagerie de la perte dopaminergique et présentent également des corrélations élevées avec d'autres échelles standardisées du langage.

Bien qu'elle soit largement utilisée et acceptée, l'échelle H&Y présente plusieurs faiblesses, telles que la non-inclusion des problèmes non-moteurs, sa non-linéarité, le mélange de déficiences et d'incapacités, et son poids élevé en faveur de l'instabilité posturale, excluant les déficiences ou les incapacités provenant d'autres symptômes moteurs. Cette échelle ne prend pas en compte l'évaluation explicite des troubles de la parole chez les patients, cependant, il existe plusieurs études en analyse de la parole qui utilisent cette échelle comme base de référence pour effectuer des tâches de régression, de classification multi-classes ou d'analyse statistique.

- Échelle unifiée d'évaluation de la MP (UPDRS) :

Il existe différentes versions de cette échelle, la plus récente appelée MDS-UPDRS (Movement Disorders Society sponsored revision of the Parkinson's Disease Rating Scale). Cette version évalue les aspects moteurs et non-moteurs de la MP. Elle comporte quatre parties, la partie I évalue les expériences non motrices de la vie quotidienne (13 éléments), la partie II comprend les expériences motrices de la vie quotidienne (13 éléments), la partie III comprend l'examen moteur (33 éléments), et la partie IV sert à évaluer les complications motrices (6 éléments). Les notes de chaque élément vont de 0 (normal) à 4 (sévère) et le score total de chaque partie est obtenu à partir de la somme des éléments correspondants. L'impact de la MP sur la parole n'est pris en compte que dans un seul élément ; cependant, comme la sous-échelle motrice (partie III) reflète les problèmes moteurs, elle est généralement utilisée comme vérité de base pour étiqueter les patients dans différentes expériences de reconnaissance des formes. Dans ce cas, les enregistrements vocaux de chaque patient sont étiquetés avec des scores allant de 0 à 132 (33 éléments \times 4 = 132). L'échelle MDS-UPDRS est spécifiquement conçue pour évaluer les patients atteints de la maladie de Parkinson et sa fiabilité a été validée avec des résultats satisfaisants par un essai test-retest réalisé auprès de 435 patients de cinq pays. Bien que l'échelle présente plusieurs points forts, tels que sa large utilisation, sa couverture des symptômes moteurs et non moteurs et ses propriétés clinimétriques (fiabilité et validité), l'évaluation de la parole ne représente qu'un seul élément et en général, le neurologue n'est pas formé à l'évaluation/au dépistage des différents troubles de la parole dont souffrent les patients. La version précédente de l'échelle est simplement appelée UPDRS. Elle évalue les incapacités et les déficiences liées à la maladie de Parkinson. Elle est composée de 42 éléments, dont 4 sur la mentalité, le comportement et l'humeur, 13 sur les activités de la vie quotidienne, 27 sur les activités motrices et 11 sur les complications du traitement.

5.2 L'analyse et le prétraitement des données

Avant de procéder à l'extraction des caractéristiques, nous devons premièrement analyser la qualité du signal des enregistrements vocaux pour éliminer les silences et la saturation. À cette fin, l'application Audio-Signal-Analyser.py a été développée. Notre script fait appel à Parselmouth [\[63\]](#) pour l'analyse et la visualisation de l'état du signal. C'est une bibliothèque Python pour le logiciel

Praat qui accède directement au code C de Praat, ce qui signifie que les algorithmes et leurs résultats sont exactement les mêmes que dans Praat.

Pour l'interface graphique de notre application on fait appel au module PyQt5 ; l'une des bibliothèques GUI multiplateformes les plus puissantes et les plus populaires qui implique d'autres modules comme QtCore pour travailler avec des fichiers et des répertoires, QtGui pour l'intégration du système de fenêtrage, et QtWidgets qui est la classe de base de tous les objets d'interface utilisateur, et pratiquement tout ce que vous voyez dans l'application est un widget. Cela inclut les boîtes de dialogue, les boutons, les barres, et ainsi de suite. En outre, QApplication contient la boucle d'événements principale, où tous les événements du système de fenêtres et d'autres sources sont traités et distribués. Il gère également l'initialisation et la finalisation de l'application et assure la gestion des sessions. De plus nous avons utilisé l'objet QLabel pour afficher du texte désiré à l'aide des balises HTML, et setGeometry() pour définir la taille et la position de la fenêtre. Pour la sélection de fichiers audio on fait appel au QFileDialog qui permet à l'utilisateur de naviguer dans le système de fichiers et de sélectionner un fichier .wav ou .mp3 pour que Parselmouth puisse analyser le signal audio de l'enregistrement désiré, et finalement entrer la boucle principale de l'application par la méthode app.exec_() .

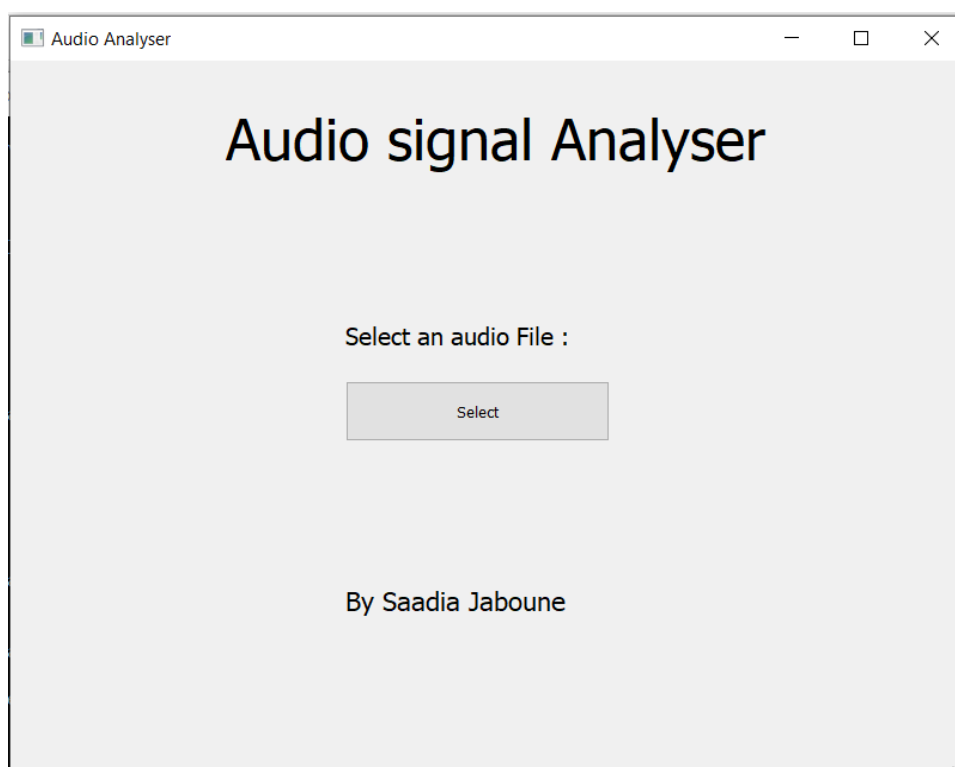


Figure 22. L'interface graphique de l'application Audio-Signal-Analyser.py

Après la conception graphique, et l'importation des bibliothèques nécessaires nous ouvrons et lisons le fichier audio à l'aide de `parselmouth.Sound()`. Ensuite, on appelle aux algorithmes de Praat ci-dessous pour le calcul et l'analyse de :

- Pitch: `to_pitch().selected_array['frequency']`
- L'énergie : `(intensity.xs(), intensity.values.T)`
- Le HNR: `to_harmonicity()`
- Le spectrogramme: `to_spectrogram()`
- Les formants: `to_formant_burg(max_number_of_formants= 4, maximum_formant)`

```

from PyQt5 import QtCore, QtGui, QtWidgets
from PyQt5.QtWidgets import QMainWindow, QApplication
import sys
import parselmouth
from parselmouth.praat import call
import matplotlib.pyplot as plt
import numpy as np

class Ui_MainWindow(object):
    def setupUi(self, MainWindow):
        MainWindow.setObjectName("MainWindow")
        MainWindow.resize(800, 600)
        self.centralwidget = QtWidgets.QWidget(MainWindow)
        self.centralwidget.setObjectName("centralwidget")
        self.pushButton = QtWidgets.QPushButton(self.centralwidget)
        self.pushButton.setGeometry(QtCore.QRect(280, 270, 221, 51))
        self.pushButton.setObjectName("pushButton")
        self.pushButton.clicked.connect(self.audio_select)
        self.label = QtWidgets.QLabel(self.centralwidget)
        self.label.setGeometry(QtCore.QRect(280, 220, 500, 21))
        self.label.setObjectName("label")
        self.label_2 = QtWidgets.QLabel(self.centralwidget)
        self.label_2.setGeometry(QtCore.QRect(180, 10, 461, 111))
        self.label_2.setObjectName("label_2")
        self.label_3 = QtWidgets.QLabel(self.centralwidget)
        self.label_3.setGeometry(QtCore.QRect(280, 400, 461, 111))
        self.label_3.setObjectName("label_3")
        self.label_3.setText("<html><head/><body><p><span style=\" font-size:13pt;\">By Saadia Jaboune</span></p></body></html>")
        MainWindow.setCentralWidget(self.centralwidget)
        self.menubar = QtWidgets.QMenuBar(MainWindow)
        self.menubar.setGeometry(QtCore.QRect(0, 0, 800, 21))
        self.menubar.setObjectName("menubar")
        MainWindow.setMenuBar(self.menubar)
        self.statusbar = QtWidgets.QStatusBar(MainWindow)
        self.statusbar.setObjectName("statusbar")
        MainWindow.setStatusBar(self.statusbar)
        self.retranslateUi(MainWindow)
        QtCore.QMetaObject.connectSlotsByName(MainWindow)
    def retranslateUi(self, MainWindow):
        _translate = QtCore.QCoreApplication.translate
        MainWindow.setWindowTitle(_translate("MainWindow", "Audio Analyser"))
        self.pushButton.setToolTip(_translate("MainWindow",
"<html><head/><body><p><span style=\" font-size:16pt;\">Select</span></p></body></html>"))
        self.pushButton.setText(_translate("MainWindow", "Select"))
        self.label.setText(_translate("MainWindow", "<html><head/><body><p><span style=\" font-size:12pt;\">Select an audio File :</span></p></body></html>"))
        self.label_2.setText(_translate("MainWindow", "<html><head/><body><p><span style=\" font-size:29pt;\">Audio signal Analyser</span></p></body></html>"))
    def audio_select(self):
        global path
        path = None
        file_select = QtWidgets.QFileDialog()
        filters = "Audio files (*.wav *.mp3)"
        path = QtWidgets.QFileDialog.getOpenFileName(file_select, "Select an audio",
".", filters)

```

Après le choix d'un enregistrement audio, l'application affiche le tracé de : l'amplitude, l'énergie, le pitch, HNR et de spectrogramme avec les quatre premiers formants comme le montre la figure 23.

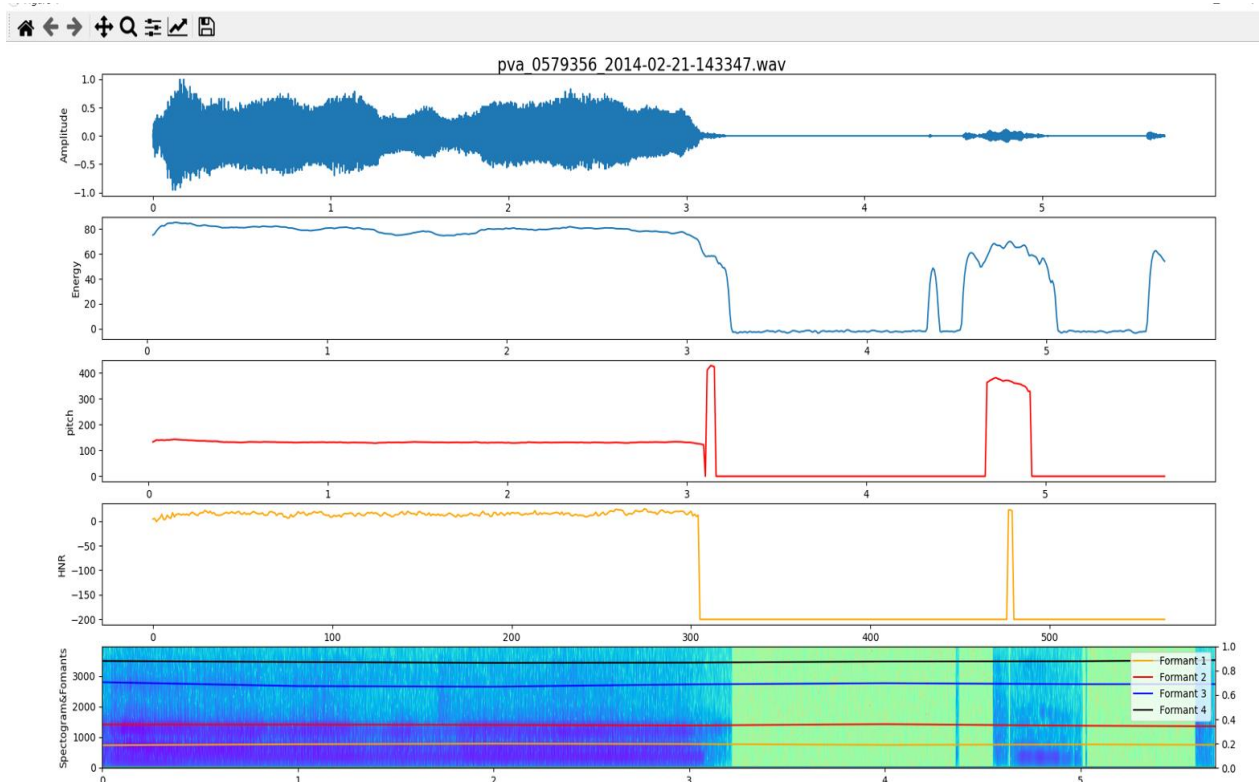


Figure 23. Présentation graphique d'un enregistrement de la base de données PVA avant la suppression du silence.

À partir de l'interprétation du graphique ci-dessus, vous pouvez constater que la voix est forte pendant 3 secondes, cependant elle représente un silence d'environ 2 secondes, ce qui ne permet pas une extraction fiable des caractéristiques du signal. À cette fin, nous avons développé un script pour normaliser le signal et éliminer les régions qui représentent un silence. Notre application utilise le module Pydub [65] qui nous permet de lire, diviser, fusionner et éditer les fichiers audio .wav. En commençant par charger nos données audios dans un tableau Numpy ; le moyen le plus adopté pour traiter les séquences et les vecteurs numériques en utilisant `scipy.io.wavfile`. En procédant par la suite à la normalisation pour effectuer des calculs sur les valeurs du signal audio et les rend indépendantes de la résolution de l'échantillon. Nous avons normalisé le signal audio dans l'intervalle (-1,1), en le divisant simplement par 2^{15} , parce que nous savons que la résolution d'échantillon est de 16 bits par échantillon.

```
import numpy as np
from scipy.io import wavfile
from pydub import AudioSegment
import os
path = "./pva_wav_test/wav/"
for audio in os.listdir(path):
    fs, signal = wavfile.read(path+audio) # read WAV file
    audioname = audio.split(".")[0]

    signal = signal / (2**15) # Normalization
    signal_len = len(signal)
```

En suivant le scénario du script, nous avons divisé les enregistrements audios en segments d'une seconde, cela se fait en se référant simplement aux indices respectifs dans le tableau Numpy.

Évidemment, les indices doivent être en échantillons audio, donc les secondes doivent être multipliées par la fréquence d'échantillonnage. Par la suite on implémente une méthode de suppression de silence très simple qui calcule l'énergie comme la somme des carrés des échantillons, puis il calcule un seuil à 50 % de la valeur d'énergie médiane, et enfin il conserve les segments dont l'énergie est supérieure à ce seuil.

```
segment_size_t = 1 # segment size in seconds
segment_size = segment_size_t * fs # segment size in samples

# Break signal into list of segments in a single-line Python code
segments = np.array([signal[x:x
+ segment_size] for x in np.arange(0, signal_len, segment_size)], dtype=object)

# Remove pauses using an energy threshold = 50% of the median energy:
energies = [(s**2).sum() / len(s) for s in segments]
# (attention: integer overflow would occur without normalization here!)
thres = 0.5 * np.median(energies)
index_of_segments_to_keep = (np.where(energies > thres)[0])

# get segments that have energies higher than a the threshold:
segments2 = segments[index_of_segments_to_keep]

# concatenate segments to signal:
new_signal = np.concatenate(segments2)

# Generate final audio file without silence :
wavfile.write(f"./cutted/{audioname}_processed.wav", fs, new_signal.astype(np.float32))
```

Après l'application du script `cutted.py`, nous avons visualisé à nouveau le signal de l'enregistrement `pva_0579356_2014-02-21-143347.wav`, et comme vous pouvez le voir, nous avons obtenu la partie du signal supérieure à 50%, c'est-à-dire que les régions avec silence sont éliminées.

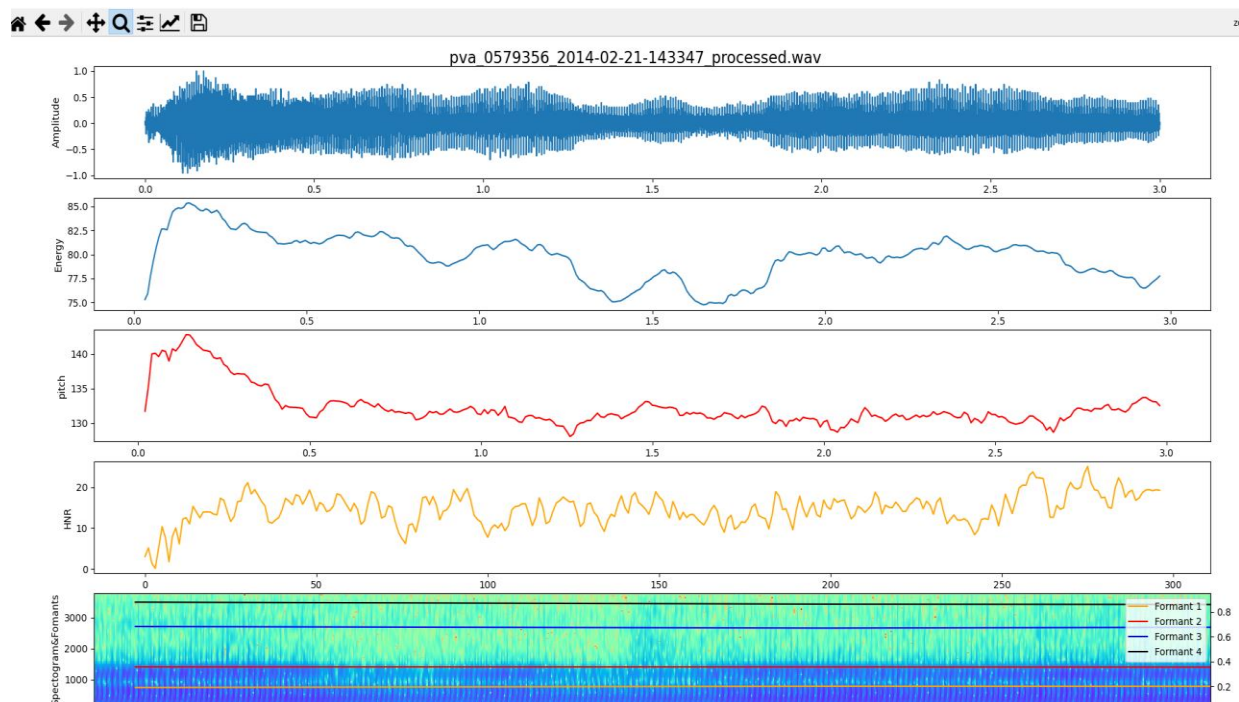
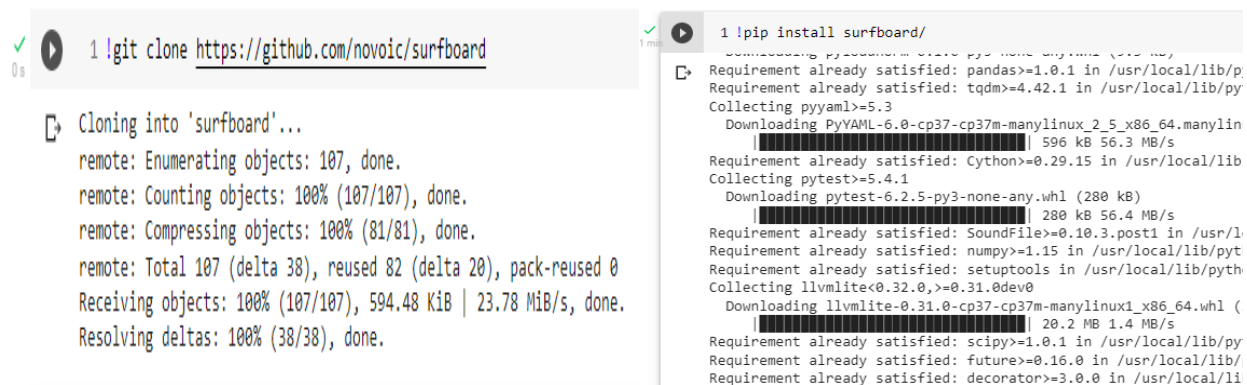


Figure 24. Présentation graphique d'un enregistrement de la base de données PVA après la suppression du silence.

5.3 Extraction des caractéristiques

Une fois que les enregistrements audios de deux bases de données sont traités, on peut procéder à l'étape suivante, qui s'agit de l'extraction des caractéristiques. À ce propos, nous avons utilisé Surfboard [66], la bibliothèque open-source de Python qui est basée sur des paquets d'analyse audio et met l'accent sur la capacité à traiter de grands ensembles de données nécessaires pour les approches modernes d'apprentissage automatique. Elle a été construite sur la base d'un certain nombre de paquets, tels que LibROSA [67] et pysptk [68], et utilise l'algorithme de détection du pitch RAPT présenté dans la section 4.2.3.2, avec une variété de fichiers contenant des fonctions pour le calcul des caractéristiques nécessaires. Pour ce faire, nous avons utilisé l'environnement Colab qui fonction entièrement dans le cloud et prend en charge les bibliothèques d'apprentissage automatique.

Avant de procéder à l'extraction des caractéristiques, nous devons d'abord cloner le dépôt github de Surfboard [66], puis installer le paquet comme suit : ! pip install surfboard/

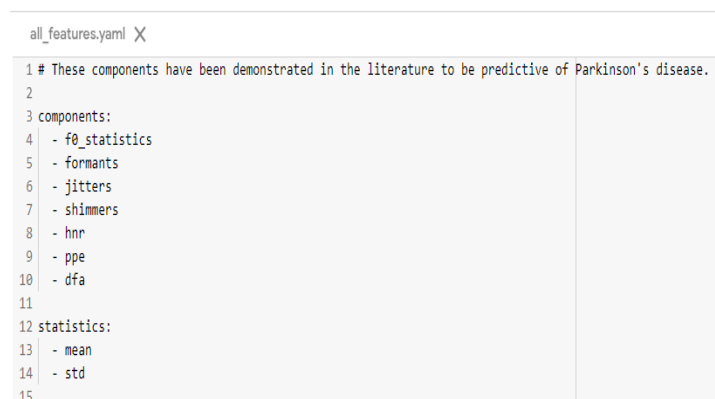


```
1 !git clone https://github.com/novoic/surfboard
Cloning into 'surfboard'...
remote: Enumerating objects: 107, done.
remote: Counting objects: 100% (107/107), done.
remote: Compressing objects: 100% (81/81), done.
remote: Total 107 (delta 38), reused 82 (delta 20), pack-reused 0
Receiving objects: 100% (107/107), 594.48 KiB | 23.78 MiB/s, done.
Resolving deltas: 100% (38/38), done.

1 !pip install surfboard/
Requirement already satisfied: pandas>=1.0.1 in /usr/local/lib/p
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/py
Collecting pyyaml>=5.3
  Downloading PyYAML-6.0-cp37-cp37m-manylinux_2_5_x86_64.manylin
    596 kB 56.3 MB/s
Requirement already satisfied: Cython>=0.29.15 in /usr/local/lib
Collecting pytest>=5.4.1
  Downloading pytest-6.2.5-py3-none-any.whl (280 kB)
    280 kB 56.4 MB/s
Requirement already satisfied: SoundFile>=0.10.3.post1 in /usr/l
Requirement already satisfied: numpy>=1.15 in /usr/local/lib/pyt
Requirement already satisfied: setuptools in /usr/local/lib/pyth
Collecting llvmlite<0.32.0,>=0.31.0dev0
  Downloading llvmlite-0.31.0-cp37m-manylinux1_x86_64.whl (
    20.2 MB 1.4 MB/s
Requirement already satisfied: scipy>=1.0.1 in /usr/local/lib/py
Requirement already satisfied: future>=0.16.0 in /usr/local/lib/
Requirement already satisfied: decorator>=3.0.0 in /usr/local/li
```

Figure 25. Le clonage et l'installation du package surfboard.

Ensuite on doit importer les bases de données à l'environnement colab. Puis, à l'aide du fichier all_features.yaml intégré dans surfboard, on peut déterminer et personnaliser les caractéristiques souhaitées à calculer et supprimer celles non désirées.



```
all_features.yaml X
1 # These components have been demonstrated in the literature to be predictive of Parkinson's disease.
2
3 components:
4   - f0_statistics
5   - formants
6   - jitters
7   - shimmers
8   - hnr
9   - ppe
10  - dfa
11
12 statistics:
13   - mean
14   - std
15
```

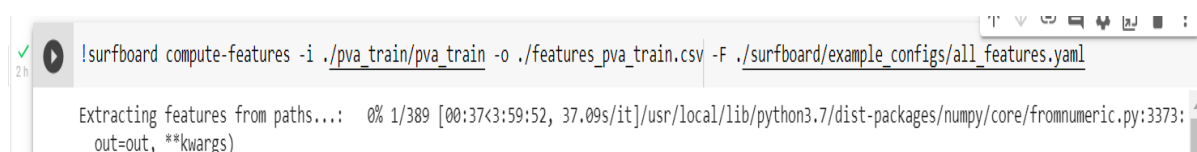
Figure 26. Le fichier all_features.yaml après la personnalisation..

Le fichier Class Waveform contient la classe centrale du paquet Surfboard, et toutes les méthodes correspondantes pour calculer les composantes que nous avons déterminé dans le fichier all_features.yaml. Les paramètres de ces composantes sont cités brièvement dans le tableau 4. À noter que les méthodes de calcul sont les mêmes comme indiqué dans la section 4.4.

Figure 27. Les composantes choisies de la Class Waveform avec leurs paramètres [66].

Composante	Paramètres	Résultat renvoyé
f0_statistics	(hop_length_seconds=0.01, method='rapt') - hop_length_seconds (float) - Argument de taille de saut dans pysptk. - method (str) - la méthode 'rapt' pour le calcul de f0.	"mean" et "std" de f0
formants	(estimate_formants_lpc(waveform, sample_rate, num_formants=5) sample_rate : taux d'échantillonnage - 5 comme base, car le formant f0 sera éliminer et nous voulons f1...f4.	{'f1' 'f2' 'f3' 'f4'}
jitters	(p_floor=0.0001,p_ceil=0.02,max_p_factor=1.3) - p_floor (float) - Période minimale acceptable. - p_ceil (float) - Période maximale acceptable. - max_p_factor (float) - Valeur à utiliser pour le principe du facteur de période.	"localJitter" "localabsoluteJitter" "rapJitter" "ppq5Jitter" "ddpJitter"
shimmers	(max_a_factor=1.6,p_floor=0.0001,p_ceil=0.02, max_p_factor=1.3) - max_a_factor (float) - Valeur à utiliser pour le principe du facteur d'amplitude - p_floor (float) - Période minimale acceptable. - p_ceil (float) - Période maximale acceptable. - max_p_factor (float) - Valeur à utiliser pour le principe du facteur de période.	"localShimmer" "localdbShimmer" "apq3Shimmer" "apq5Shimmer" "apq11Shimmer"
hnr	(min_pitch=75.0,silence_threshold=0.1, periods_per_window=4.5) - waveform (np.array, [T, 1]) : signal de forme d'onde - sample_rate (int > 0) : taux d'échantillonnage - min_pitch (float > 0) : pitch minimum - silence_threshold (1 >= float >= 0) : doit être dans [0, 1]. En dessous de cette amplitude, ne tient pas compte les trames. - periods_per_window (float > 0) : 4.5 est le mieux pour la parole.	hnr (float)
ppe	La fonction ppe() : 1. Transformation des Hertz en demi-tons 2. Filtre de blanchiment linéaire 3. Distribution de probabilité discrète 4. Entropie de cette distribution	ppe (float)
dfa	(window_lengths=[64, 128, 256, 512, 1024, 2048, 4096]) window_lengths (list of int > 0) - Liste de L à utiliser dans le calcul des DFA.	Retourne la valeur alpha de dfa (float)

Avec la ligne de commande ci-dessous un fichier sera extrait contenant l'ensemble de caractéristiques calculées pour chaque fichier .wav de notre dossier.



```
!surfboard compute-features -i ./pva_train/pva_train -o ./features_pva_train.csv -F ./surfboard/example_configs/all_features.yaml
Extracting features from paths...: 0% 1/389 [00:37<3:59:52, 37.09s/it]/usr/local/lib/python3.7/dist-packages/numpy/core/fromnumeric.py:3373:
out=out, **kwargs)
```

La figure 28 est l'aperçu du fichier extrait. Il contient le nom et les valeurs des caractéristiques calculées de chaque enregistrement audio, cependant nous aurons besoin du pdrs_score pour les

sections suivantes, ainsi que ne nous savons pas l'âge et le sexe du propriétaire de chaque enregistrement audio, pour cela nous avons développé ce script ci-dessous qui va importer ces informations depuis un fichier partagé sur le site web de Synapse [\[62\]](#) en se basant sur le nom de chaque enregistrement.

```
import os
import pandas as pd
path = "/pva_train.csv/"
csv = "plmpva_train.csv"
data = pd.read_csv(csv)
df = pd.DataFrame(data)
audio_data = df['call_timestamp']
gender = []
fname = []
age = []
pdr_score = []

for i in os.listdir(path):
    d = i.split('_')
    split = d[2].replace('.wav', '').split('-')
    year = split[0]
    month = split[1]
    day = split[2]
    time = split[3]
    hour = time[0:2]
    min = time[2:4]
    sec = time[4:6]
    audio_name = f"{year}/{month}/{day} {hour}:{min}:{sec} UTC +0000"
    sex = df[df['call_timestamp'] == audio_name].sex.item()
    current_age = df[df['call_timestamp'] == audio_name].current_age.item()
    pdrs = df[df['call_timestamp'] == audio_name].pdrs_score.item()
    gender.append(sex)
    fname.append(i)
    age.append(current_age)
    pdr_score.append(pdrs)

#
new_path = 'pva_train.csv'
csv2 = pd.read_csv(new_path)
df2 = pd.DataFrame(csv2)
df2['sex'] = gender
df2['pdrs_score'] = pdr_score
df2['current_age'] = age
df2.to_csv("pva_train.csv")
```

fnames	f0_mean	f0_std	f1	f2	f3	f4	locallitter	localabsolu	raplitter	ppq5litter	ddplitter	localShimm	localdbShi	apq3Shim	apq5Shim	apq11Shim	hnr	ppe	dfa
pva_0007588_2014-02-03-160636	216.2311	2.6939	1049.1	1321	2845	3716.3	0.002786	1.29E-05	0.000806	0.001773	0.002419	0.0228789	0.211144	0.012178	0.019822	0.03794079	18.37694	-98.533	0.586757
pva_0018574_2014-03-07-000615	217.6604	3.4947	860.89	1422	2727	3064.1	0.002495	1.15E-05	0.000561	0.001191	0.001682	0.0125379	0.10987	0.007437	0.011748	0.02036393	19.12514	-141.268	0.589563
pva_0064804_2014-02-27-025929	198.4582	4.7036	1364.3	1638	2891	3230.1	0.005734	2.89E-05	0.001414	0.003076	0.004242	0.0597628	0.522911	0.028446	0.045539	0.09748604	12.92202	-33.1249	0.744754
pva_0066678_2014-02-14-163147	174.3074	2.3317	1255.8	2211	2885	3471.5	0.003366	1.93E-05	0.000671	0.001501	0.002012	0.0184616	0.162686	0.009217	0.014288	0.02774216	18.46133	-191.296	0.663496
pva_0085126_2014-02-19-120107	162.2994	4.5065	1147.2	2140	2712	3379.4	0.003985	2.46E-05	0.00105	0.002286	0.00315	0.0267971	0.220309	0.014779	0.02305	0.0433649	13.02691	-31.5744	0.609382
pva_0118640_2014-02-18-001831	107.3766	2.5649	1187	2234	3144	3778.5	0.003276	3.05E-05	0.000606	0.001335	0.001817	0.019064	0.315307	0.009805	0.014483	0.02519343	9.084779	-7.64362	0.652352
pva_0230401_2014-02-11-192910	246.6759	1.6236	1316.5	2154	2869	3262.9	0.001561	6.33E-06	0.000407	0.000802	0.001222	0.0180206	0.161369	0.008646	0.014001	0.02894894	21.97861	-44.4448	0.576404
pva_0235147_2014-02-22-195622	131.718	2.7923	1096.1	2137	2985	3235.5	0.002079	1.58E-05	0.000508	0.001148	0.001525	0.0207394	0.364841	0.008676	0.012463	0.02132085	12.05237	-25.9844	0.578193
pva_0239229_2014-02-20-124944	188.9979	8.6804	1057.8	1804	2360	2848.5	0.00328	1.74E-05	0.000742	0.001595	0.002225	0.0183217	0.175126	0.008616	0.013269	0.02732628	18.39656	-37.4298	0.613544
pva_0264330_2014-02-14-212656	220.7692	2.9038	967.02	1879	2531	3023.3	0.003094	1.40E-05	0.000693	0.001458	0.002079	0.0190803	0.16517	0.010962	0.016462	0.03033894	17.20739	-43.2193	0.718397
pva_0318302_2014-02-20-154352	159.7746	1.7991	1209.3	2370	3122	3452.2	0.001891	1.18E-05	0.000476	0.000986	0.001429	0.0069498	0.062939	0.003905	0.00601	0.01099942	18.51604	-105.67	0.585354
pva_0330573_2014-02-20-232818	215.032	0.4949	1478.7	2224	2827	2898.8	0.001481	6.89E-06	0.000594	0.001051	0.001781	0.0188968	0.182198	0.01069	0.016943	0.02821062	18.57106	-44.9185	0.562938
pva_0356723_2014-02-17-220801	227.8684	25.595	970.21	1669	2878	3359.7	0.002495	1.12E-05	0.000724	0.001436	0.002172	0.0407515	0.365287	0.017339	0.026519	0.06033754	17.95264	-47.6137	0.600493
pva_0387945_2014-03-01-063643	131.6128	1.7582	592.07	1169	1548	2552.7	0.001546	1.18E-05	0.000334	0.000686	0.001001	0.0143868	0.12454	0.008422	0.012789	0.02249346	19.59269	-446.739	0.634176
pva_0391610_2014-01-28-035356	140.1141	3.4341	1173.4	2455	3291	3649.7	0.001809	1.29E-05	0.000479	0.000931	0.001438	0.0587845	0.53045	0.030774	0.043112	0.04194963	14.78925	-120.144	0.604462
pva_0402368_2014-02-20-042408	227.1033	6.0567	1306.1	2280	2822	3530.2	0.002535	1.12E-05	0.00066	0.001454	0.001979	0.0133772	0.14779	0.007662	0.0121	0.02067763	17.24305	-23.0282	0.565514
pva_0407476_2014-01-30-194753	134.4649	1.9353	1163.6	2784	2915	3173.3	0.002406	1.79E-05	0.000569	0.001239	0.001707	0.0116871	0.099584	0.006492	0.009601	0.01744176	21.41696	-169.239	0.683668
pva_0413599_2014-02-14-232725	186.9848	2.9105	701.95	1093	1719	2897.5	0.001915	1.02E-05	0.000597	0.001136	0.001792	0.0135996	0.115353	0.007636	0.012113	0.01906186	15.72775	-329.337	0.744616

Figure 28. Aperçu du fichier extrait contenant les caractéristiques générées.

5.3.1 Le prétraitement et la comparaison de deux ensembles de données

Le prétraitement est un aspect important de la gestion des données. Il consiste à examiner les données pour supprimer ou mettre à jour les informations incomplètes, incorrectes ou non pertinentes. Dans cette partie, nous allons examiner les similitudes et les différences entre ces deux ensembles de données à l'aide de Pandas, l'un des paquets de python qui facilite l'importation et l'analyse des données. En commençant tout d'abord par l'affichage de la forme total de données, comme indiqué ci-dessous :

```
# L'extraction des données à l'aide de pandas
import pandas as pd
# Lire les fichiers CSV
GITA = pd.read_csv('/content/GITA.csv')
PVA_train = pd.read_csv('/content/PVA_train.csv')
PVA_test = pd.read_csv('/content/PVA_test.csv')
# Affichage
print("Nombre de lignes et de colonnes dans le cadre de données:")
print("La forme de L'ensembles de données GITA: {}".format(GITA.shape))
print("La forme de L'ensembles de données PVA-
train: {}".format(PVA_train.shape))
```

```
Nombre de lignes et de colonnes dans le cadre de données:
La forme de L'ensembles de données GITA: (100, 25)
La forme de L'ensembles de données PVA_test: (361, 25)
La forme de L'ensembles de données PVA-train: (346, 25)
```

Figure 29. Résultat de l'affichage de la forme total des ensembles de données

Nous remarquons immédiatement que PVA_train contient 346 lignes et PVA_test constitué de 361 lignes de données cela est dû à l'élimination des enregistrements audios de mauvaise qualité durant la phase d'analyse des signaux vocaux. En outre la base de données GITA contient 100 lignes, et le nombre des attributs égale à 25 pour tous les ensembles de données.

Il est important aussi de vérifier l'existence ou non des valeurs manquantes ainsi le type des données :

GITA.info()	PVA_test.info()	PVA_train.info()
<pre> les informations sur la base de données GITA <class 'pandas.core.frame.DataFrame'> RangeIndex: 100 entries, 0 to 99 Data columns (total 25 columns): # Column Non-Null Count Dtype --- --- 0 fnames 100 non-null object 1 sex 100 non-null int64 2 age 100 non-null int64 3 UPDRS 100 non-null int64 4 hoehn_yahr 100 non-null float64 5 time_after_diagnosis 100 non-null float64 6 f0_mean 100 non-null float64 7 f0_std 100 non-null float64 8 f1 100 non-null float64 9 f2 100 non-null float64 10 f3 100 non-null float64 11 f4 100 non-null float64 12 localJitter 100 non-null float64 13 localabsoluteJitter 100 non-null float64 14 rapJitter 100 non-null float64 15 ppq5Jitter 100 non-null float64 16 ddpJitter 100 non-null float64 17 localShimmer 100 non-null float64 18 localdbShimmer 100 non-null float64 19 apq3Shimmer 100 non-null float64 20 apq5Shimmer 100 non-null float64 21 apq11Shimmer 100 non-null float64 22 hnr 100 non-null float64 23 ppe 100 non-null float64 24 dfa 100 non-null float64 dtypes: float64(21), int64(3), object(1) </pre>	<pre> les informations sur la base de données PVA_test <class 'pandas.core.frame.DataFrame'> RangeIndex: 361 entries, 0 to 360 Data columns (total 25 columns): # Column Non-Null Count Dtype --- --- 0 fnames 361 non-null object 1 sex 361 non-null int64 2 current_age 361 non-null int64 3 pdrs_score 361 non-null int64 4 hoehn_yahr 361 non-null int64 5 years_since_first_symptom 361 non-null float64 6 f0_mean 361 non-null float64 7 f0_std 361 non-null float64 8 f1 361 non-null float64 9 f2 361 non-null float64 10 f3 361 non-null float64 11 f4 361 non-null float64 12 localJitter 361 non-null float64 13 localabsoluteJitter 361 non-null float64 14 rapJitter 361 non-null float64 15 ppq5Jitter 361 non-null float64 16 ddpJitter 361 non-null float64 17 localShimmer 361 non-null float64 18 localdbShimmer 361 non-null float64 19 apq3Shimmer 361 non-null float64 20 apq5Shimmer 361 non-null float64 21 apq11Shimmer 361 non-null float64 22 hnr 361 non-null float64 23 ppe 361 non-null float64 24 dfa 361 non-null float64 dtypes: float64(19), int64(5), object(1) </pre>	<pre> les informations sur la base de données PVA_train <class 'pandas.core.frame.DataFrame'> RangeIndex: 346 entries, 0 to 345 Data columns (total 25 columns): # Column Non-Null Count Dtype --- --- 0 fnames 346 non-null object 1 sex 346 non-null float64 2 current_age 346 non-null int64 3 pdrs_score 346 non-null int64 4 hoehn_yahr 346 non-null int64 5 years_since_first_symptom 346 non-null float64 6 f0_mean 346 non-null float64 7 f0_std 346 non-null float64 8 f1 346 non-null float64 9 f2 346 non-null float64 10 f3 346 non-null float64 11 f4 346 non-null float64 12 localJitter 346 non-null float64 13 localabsoluteJitter 346 non-null float64 14 rapJitter 346 non-null float64 15 ppq5Jitter 346 non-null float64 16 ddpJitter 346 non-null float64 17 localShimmer 346 non-null float64 18 localdbShimmer 346 non-null float64 19 apq3Shimmer 346 non-null float64 20 apq5Shimmer 346 non-null float64 21 apq11Shimmer 346 non-null float64 22 hnr 346 non-null float64 23 ppe 346 non-null float64 24 dfa 346 non-null float64 dtypes: float64(20), int64(4), object(1) </pre>

Par la suite, en vérifie s'il y a des valeurs manquantes dans les trois bases de données :

GITA.isnull().sum()	PVA_test.isnull().sum()	PVA_train.isnull().sum()	
fnames	0	fnames	0
sex	0	sex	0
age	0	current_age	0
UPDRS	0	pdrs_score	0
hoehn_yahr	0	hoehn_yahr	0
time_after_diagnosis	0	years_since_first_symptom	0
f0_mean	0	f0_mean	0
f0_std	0	f0_std	0
f1	0	f1	0
f2	0	f2	0
f3	0	f3	0
f4	0	f4	0
localJitter	0	localJitter	0
localabsoluteJitter	0	localabsoluteJitter	0
rapJitter	0	rapJitter	0
ppq5Jitter	0	ppq5Jitter	0
ddpJitter	0	ddpJitter	0
localShimmer	0	localShimmer	0
localdbShimmer	0	localdbShimmer	0
apq3Shimmer	0	apq3Shimmer	0
apq5Shimmer	0	apq5Shimmer	0
apq11Shimmer	0	apq11Shimmer	0
hnr	0	hnr	0
ppe	0	ppe	0
dfa	0	dfa	0
dtype: int64	dtype: int64	dtype: int64	

Après avoir vérifié que toutes les colonnes sont correctement classées dans le bon type de donnée, on fait appel à la méthode corr() pour mesurer la force de corrélation entre les caractéristiques de chaque base de données. Le code ci-dessous est implémenté afin de fournir une visualisation complète des ensembles de données et évaluer la dépendance entre ces caractéristiques.

```

import seaborn as sns
import matplotlib.pyplot as plt
# create dataframe from file
fig = plt.figure(figsize=(40, 15))
sns.heatmap(GITA.corr(), annot = True, vmin=1, vmax=1, center= 0,
cmap= 'coolwarm')

fig = plt.figure(figsize=(40, 15))
sns.heatmap(PVA_train.corr(), annot = True, vmin=-
1, vmax=1, center= 0, cmap= 'coolwarm')

fig = plt.figure(figsize=(40, 15))
sns.heatmap(PVA_test.corr(), annot = True, vmin=-
1, vmax=1, center= 0, cmap= 'coolwarm')

```

5.4 La réduction de la dimensionnalité

Il est important de réduire l'ensemble de données en éliminant les caractéristiques qui fournissent moins d'informations, et de ne considérer que celles qui contiennent les informations principales ayant la plus grande capacité pour l'identification de la pathologie. L'analyse en composantes principales est une méthode très utile basée sur les mathématiques et les statistiques, qui permet de réduire la dimensionnalité en évaluant l'ensemble de données sous différents angles.. Pour l'implémentation de PCA ; python dispose du module `sklearn.decomposition` qui comprend les algorithmes de décomposition de matrices, incluant PCA.

Dans le script ci-dessous nous avons essayé de citer brièvement en quelques commentaires les étapes nécessaires pour l'implémentation de PCA en suivant les démarches mentionnées au niveau de la section 4.6.2.1. A la fin de l'implémentation on peut savoir combien de composantes il nous faut pour récupérer 99 % de la variance :

```
import pandas as pd
from sklearn import preprocessing
from sklearn.decomposition import PCA

# Lire les fichier csv
GITA = pd.read_csv('/content/GITA.csv')

# Stocker les données avec un "nom de fichier"
datasets = [("GITA", "Grupo de Investigación en Telecomunicaciones Aplicadas", GITA)]
GITA_filename, GITA_name, GITA_data = datasets[0]

# Choisissez les colonnes
GITA_columns = ['SEX', 'AGE', 'f0_mean', 'f0_std', 'f1', 'f2', 'f3', 'f4',
'localJitter', 'localabsoluteJitter', 'rapJitter', 'ppq5Jitter', 'ddpJitter', 'localShimmer',
'localdbShimmer', 'apq3Shimmer', 'apq5Shimmer', 'apq11Shimmer', 'hnr', 'ppe', 'dfa']

# Définir la Dimension intrinsèque des données pour trouver la valeur idéale à laquelle les données doivent être réduites :
def intrinsic_dimension(explained_variance_ratio_, recovered_variance=.99):
    dim = 0
    partial_var = 0
    for val in explained_variance_ratio_:
        # Update dim and va
        dim += 1
        partial_var += val
        if partial_var >= recovered_variance:
            break
    return dim
print("\n_____ \n")
print("Analyse PCA de la base de données GITA")
print(f"Ancienne forme des données: {GITA_data.shape}")

# Mettre à l'échelle/centrer nos données, en utilisant seulement les colonnes
scaled_GITA_data = pd.DataFrame(preprocessing.scale(GITA_data[GITA_columns]),
columns = GITA_columns)

# Ajustement de PCA
pca = PCA()
pca.fit(scaled_GITA_data)

# Trouver combien de composantes il nous faut pour récupérer 99 % de la variance
n_components_GITA = intrinsic_dimension(pca.explained_variance_ratio_)
print(f"Les composantes représentant 99% de la variance: {n_components_GITA}")

# Effectuer la transformée PCA en utilisant n_composantes nécessaires pour récupérer 99% de la variance
pca = PCA(n_components=n_components_GITA)
transformed_GITA_data = pca.fit_transform(scaled_GITA_data)
print(f"Nouvelle forme de données: {transformed_GITA_data.shape}")
```

```

Analyse PCA de la base de données GITA
Ancienne forme des données: (100, 25)
Les composantes représentant 99% de la variance: 15
Nouvelle forme de données: (100, 15)

```

```

L'analyse PCA de la base de données PVA_test
Ancienne forme des données: (361, 25)
Les composantes représentant 99% de la variance: 18
Nouvelle forme de données: (361, 18)

```

```

L'analyse PCA de la base de données PVA-train
Ancienne forme des données: (346, 25)
Les composantes représentant 99% de la variance: 17
Nouvelle forme de données: (346, 17)

```

Figure 30. Rapport de la forme des données après l'implémentation de PCA

5.5 Construction des modèles d'apprentissage automatique

- Visualisation des données :

Pour l'implémentation des algorithmes d'apprentissages automatiques SVM, DT et RF, nous avons sélectionné les données de 200 patients de l'ensemble de données PVA et les données de 100 personnes de la base de données GITA. Comme le montre la figure 31, 83% des données sont à des patients de la MP (250 personnes sur 300) et 17% des données sont à des sujets sains (50 personnes sur 300), nous avons donc en totalité les données de 300 candidats. Cependant pour les attributs, nous avons basé sur les résultats obtenus après l'application de PCA.

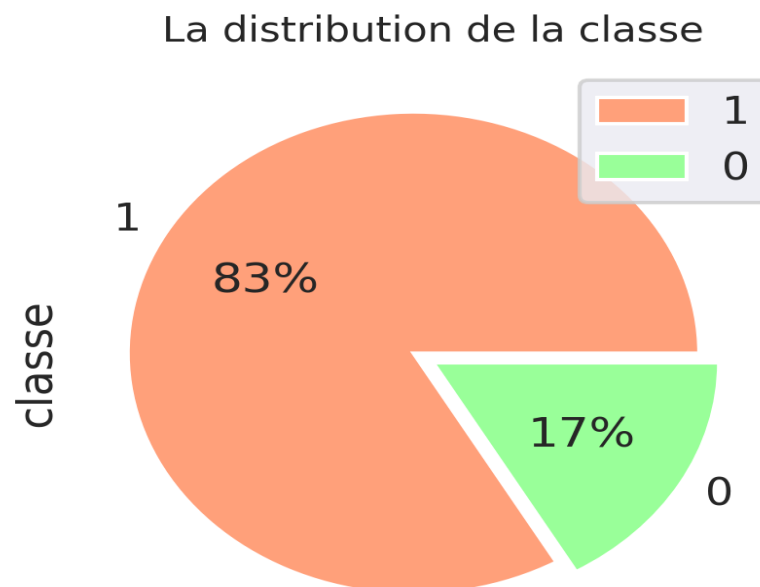


Figure 31. Visualisation de l'ensemble de données.

- Normalisation des données en utilisant l'objet scikit-learn StandardScaler.

```
from sklearn.preprocessing import MinMaxScaler, StandardScaler
```

```
scaler = StandardScaler()
feature = scaler.fit_transform(X)
```

- Fractionnement des données :

On appelle la bibliothèque `sklearn.model_selection` pour diviser les données en deux parties, la partie de formation avec un pourcentage de 70% et la partie de test avec un pourcentage de 30%.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(feature, y,
                                                    test_size=0.3, random_state=42)
```

"X_train" représente les données de formation et "X_test" les données de test. "y_train" représente le résultat (le statut d'une personne atteinte ou non de la maladie de Parkinson) des données de formation, tandis que "y_test" représente le résultat (le statut d'une personne atteinte ou non de la maladie de Parkinson) des données de test.

5.5.1 Implémentation de SVM :

Pour former un classificateur de machines à vecteurs de support, on appelle le modèle `SVC()` à partir de `sklearn` et on ajuste le modèle aux données de formation, mais avant tous nous avons besoin de régler les hyperparamètres de l'estimateur.

Les hyperparamètres sont des paramètres qui ne sont pas directement appris dans les estimateurs. Dans `scikit-learn`, ils sont passés en arguments au constructeur des classes d'estimateur. La recherche de grille est couramment utilisée comme approche de réglage d'hyperparamètres qui construira et évaluera méthodiquement un modèle pour chaque combinaison de paramètres d'algorithme spécifiés dans une grille. A ce propos, `GridSearchCV` nous aide à combiner un estimateur avec un préambule de recherche de grille pour régler les hyperparamètres.

```
def svm_grid_search(X, y):
    #créer un dictionnaire de toutes les valeurs que nous voulons tester
    param_grid = {'C': [0.1, 1, 10, 100], 'gamma': [1, 0.1, 0.01, 0.001, 0.4,
0.2, 0.8], 'kernel': ['rbf', 'poly', 'sigmoid']}
    cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)
    svm = SVC()
    # utiliser gridsearch pour tester toutes les valeurs
    svm_gscv = RandomizedSearchCV(estimator = svm,
    param_distributions = param_grid,
    scoring = 'accuracy', cv = cv, n_jobs = -1)
    # adapter le modèle aux données
    svm_gscv.fit(X, y)
    return svm_gscv.best_params_

svm_grid_search(X_train, y_train)
```

Trouver le meilleur estimateur en utilisant la recherche par grille. Puis utiliser ce modèle de grille pour créer des prédictions à l'aide de l'ensemble de test, puis créer des rapports de classification et des matrices de confusion

```

from sklearn import svm
svm = SVC(gamma=0.4, C=100, kernel='rbf', probability=True)
svm.fit(X_train, y_train)
y_pred_svm = svm.predict(X_test)
print(svm.score(X_train, y_train))
print(svm.score(X_test, y_test))
print(metrics.classification_report(y_test, y_pred_svm))

```

5.5.2 Implémentation de Forêt aléatoire (RF)

```

def rf_grid_search(X, y):
    #create a dictionary of all values we want to test
    param_grid = {
        'n_estimators': [5,10,20,40,50,60,70,80,100],
        'max_features': ['auto', 'sqrt', 'log2'],
        'max_depth' : [4,5,6,7,8],
        'criterion' :['gini', 'entropy']
    }
    cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)

    # Random Forest model
    rf = RandomForestClassifier()

    #use gridsearch to test all values
    rf_gscv = GridSearchCV(rf, param_grid, cv=cv, n_jobs=-
1, scoring='accuracy')
    #fit model to data
    rf_gscv.fit(X, y)
    return rf_gscv.best_params_
rfcl = RandomForestClassifier(n_estimators=10, max_features='auto', max_de
pth=6, criterion='gini')
rfcl = rfcl.fit(X_train, y_train)

y_pred_rf = rfcl.predict(X_test)

print(rfcl.score(X_train,y_train))
print(rfcl.score(X_test,y_test))

print(metrics.classification_report(y_test, y_pred_rf))

```

5.5.3 Implémentation d'arbre de décision (DT)

```

#Decision Tree Classifier
def dtree_grid_search(X, y):
    #create a dictionary of all values we want to test
    param_grid = { 'criterion':['gini','entropy'],'max_depth': np.arange(2
, 15)}
    cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)

```

```

# decision tree model
dtree = DecisionTreeClassifier()

#use gridsearch to test all values
dtree_gscv = GridSearchCV(dtree, param_grid, cv=cv, n_jobs=-
1, scoring='accuracy')
#fit model to data
dtree_gscv.fit(X, y)

return dtree_gscv.best_params_
dtree_grid_search(X_train, y_train)
dTree = DecisionTreeClassifier(criterion = 'entropy', max_depth = 2)
dTree.fit(X_train, y_train)

print(dTree.score(X_train,y_train))
print(dTree.score(X_test,y_test))

y_pred_dtree = dTree.predict(X_test)

print(metrics.classification_report(y_test, y_pred_dtree))

```

6 Résultats expérimentaux

6.1 Résultats de la corrélation

A partir de l'interprétation des résultats de la corrélation des ensembles de données PVA et GITA, on remarque que les variantes les plus corrélées sont celles de jitter et de shimmer, qui ont dépassé un pourcentage de 50%. Cependant que l'harmonique to noise ratio est corrélé avec eux négativement avec un pourcentage varie entre 40% et 50%. Cela pourrait faire de nos données un bon modèle pour l'analyse en composantes principales.

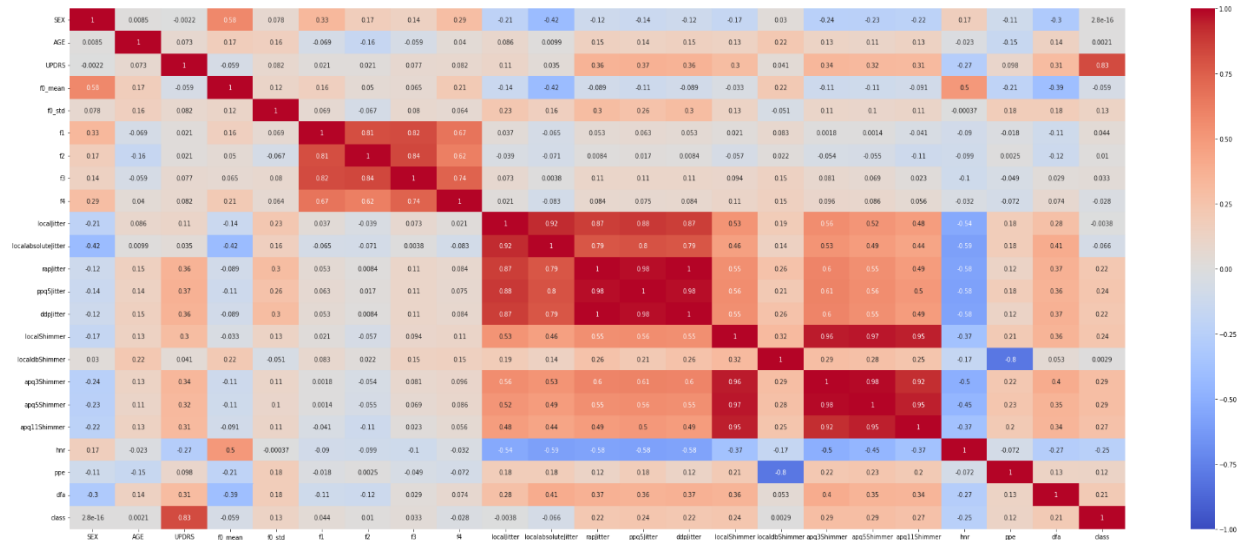


Figure 32. La corrélation des caractéristiques de la base de données GITA

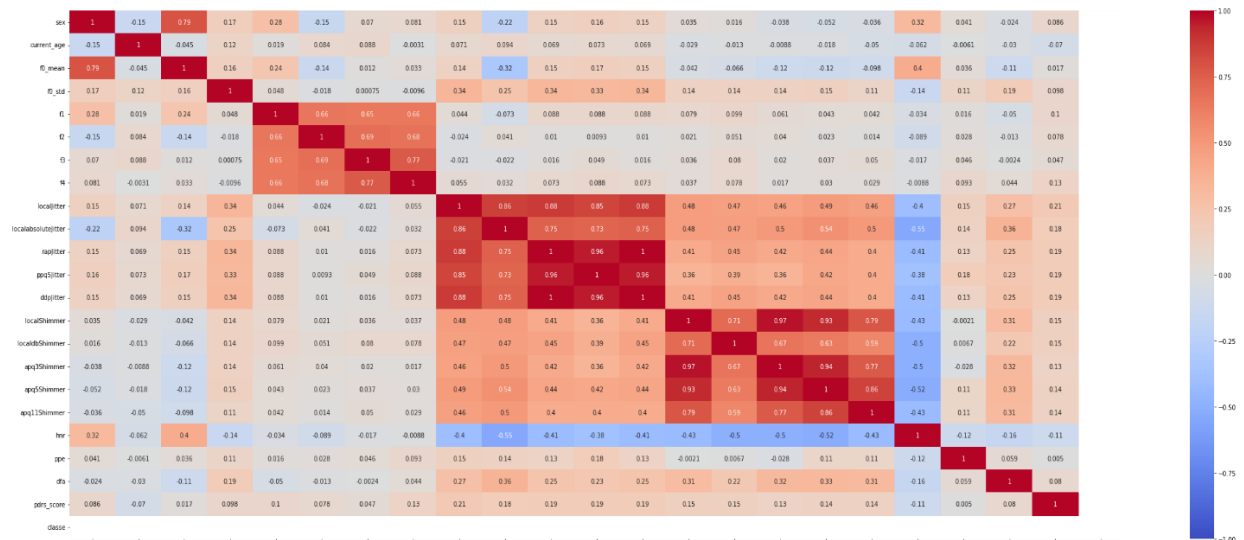


Figure 33. La corrélation des caractéristiques de la base de données PVA_train.

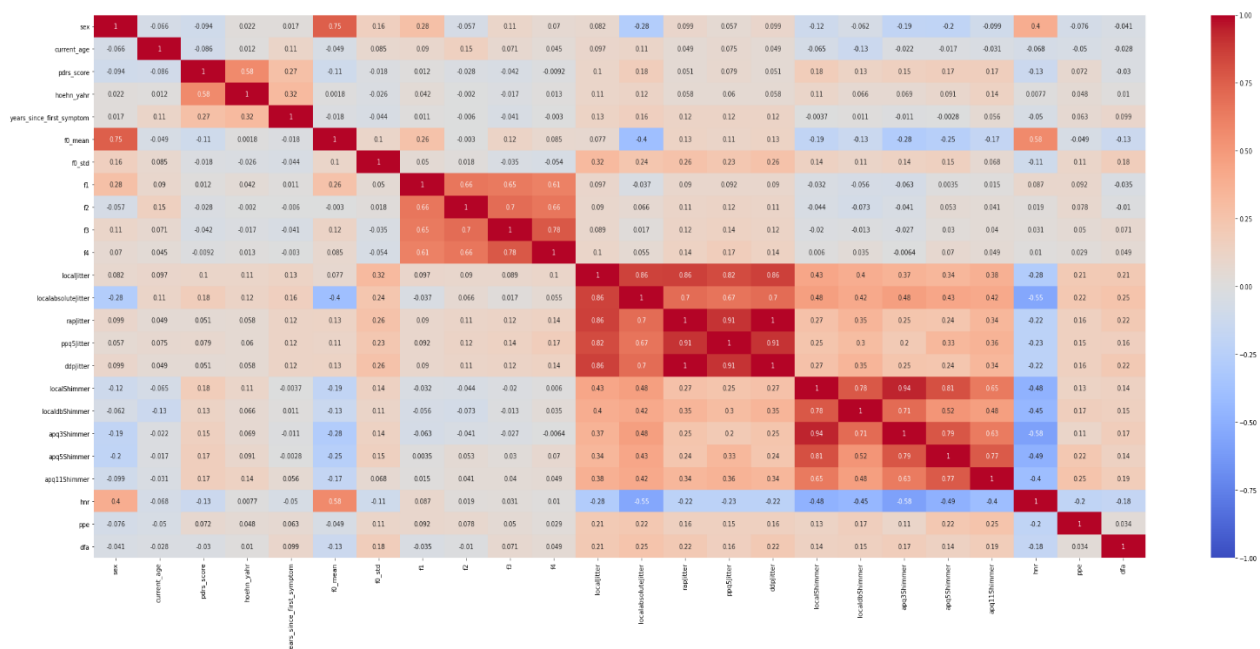


Figure 34. La corrélation des caractéristiques de la base de données PVA_test.

6.2 Résultats de réduction de la dimensionnalité

Chaque ensemble de données comporte de nombreuses caractéristiques et d'après les matrices de covariance, nous constatons que certaines caractéristiques sont fortement corrélées. Nous avons donc essayé de réduire la dimensionnalité en utilisant l'analyse en composantes principales (PCA).

Tableau 4. Réduction de dimensionnalités avec la méthode de PCA

Base de données	Nombre de caractéristiques	Composantes réduites
PVA_train	25	18
PVA_test	25	17
GITA	25	15

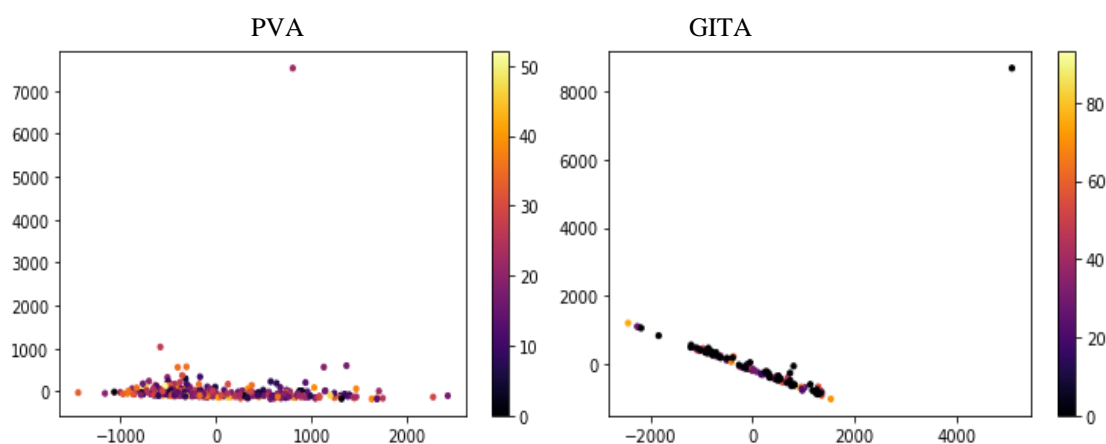


Figure 35. Visualisations pour les deux premières composantes de l'ensemble de données PVA et GITA.

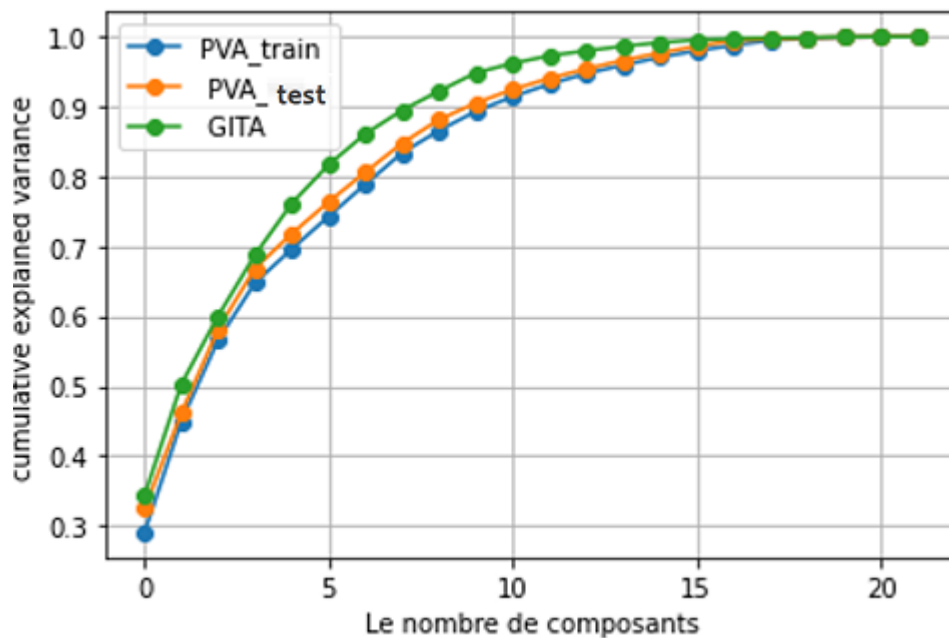


Figure 36. Cumulative explained par nombre de composants

Cela signifie que nous pouvons expliquer plus de 90 % de la variance de notre ensemble de données avec les 15 premières composantes principales.

Nous pouvons maintenant commencer à traiter les chargements de PCA. Il s'agit simplement des coefficients de la combinaison linéaire des variables originales à partir desquelles les composantes principales sont construites. Nous pouvons utiliser les chargements pour trouver des corrélations entre les variables réelles et les composantes principales. S'il y a une forte corrélation entre la composante principale et la variable d'origine, cela signifie que cette caractéristique est importante

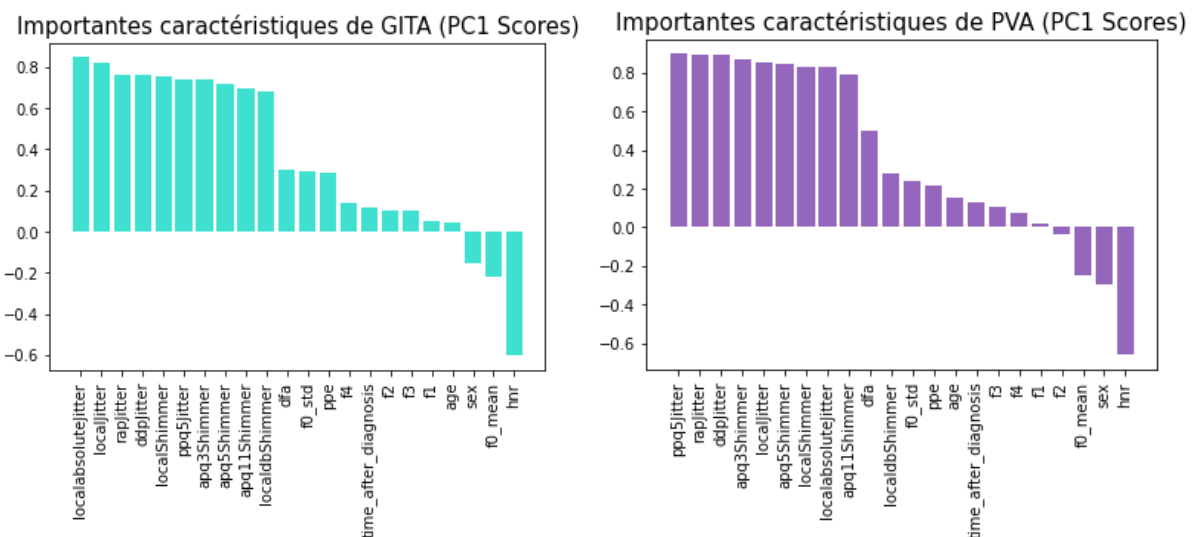


Figure 37. Importantes caractéristiques pertinentes pour la base de données PVA et GITA.

D'après la figure 37, nous constatons que les caractéristiques les plus importantes sont les suivantes : localabsolutjitter, localjitter, rapjitter, ddpjitter, localshimmer, ppq3shimmer, ppq5shimmer, ppq11shimmer et finalement HNR.

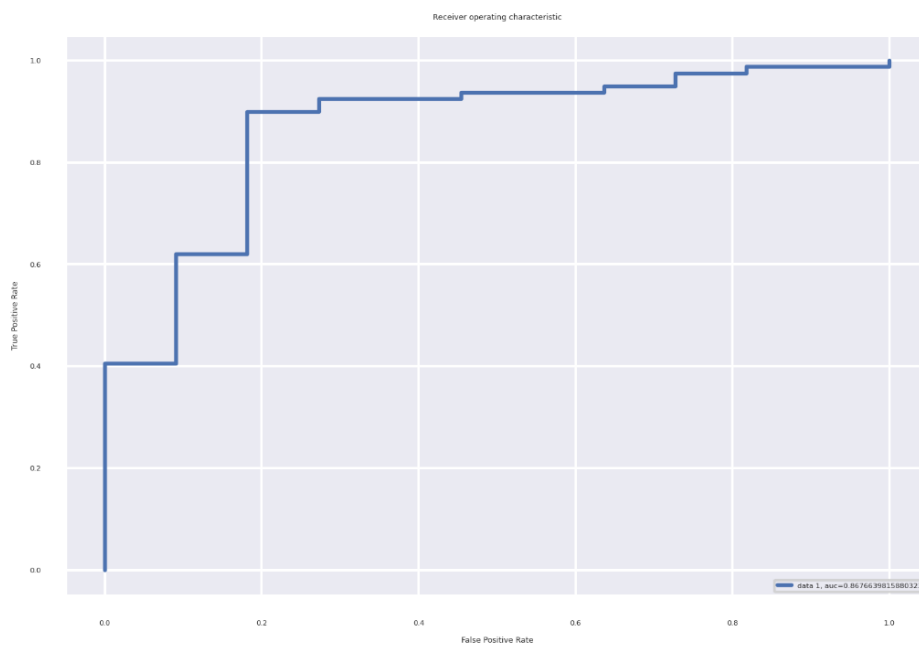
6.1 Comparaison des algorithmes d'apprentissage supervisé

6.1.1 Resultats de SVM

```
1.0
0.8777777777777778
precision    recall  f1-score   support

     0       0.50      0.55      0.52        11
     1       0.94      0.92      0.93        79

 accuracy    0.88        90
 macro avg   0.72      0.73      0.73        90
weighted avg   0.88      0.88      0.88        90
```



On peut analyser divers paramètres d'évaluation de la performance de classification du modèle SVM.

La justesse obtenue dans la phase d'apprentissage du modèle est 100%, mais dans la phase de test celle-ci est tombée à seulement 87%. Afin d'évaluer de manière pertinente l'efficacité d'un modèle, on peut aussi prendre en considération la précision, le recall ou f1-score.

La matrice de confusion résulte en utilisant l'algorithme SVM, nous donne une vue d'ensemble concernant sur la manière dans laquelle les prédictions ont été effectuées. Plus précisément, cette matrice nous donne une représentation visuelle détaillée des erreurs commises par le classificateur et aussi des types d'erreurs qui sont commises, ce qui signifie qu'on peut observer exactement les cas où le tout modèle de classification est « dérouté » lorsqu'il fait des prédictions.

Nous disposons d'un total de 90 échantillons de test, dont 79 correspondent à des personnes souffrant de la maladie de Parkinson et les 11 autres à des personnes saines.

En examinant la matrice de confusion, nous pouvons constater que le SVM a réussi à identifier correctement 73 données de test comme étant celles de personnes souffrant de la maladie de Parkinson. D'autre part, à partir de 11 données de test de personnes en bonne santé, le SVM s'est trompé sur presque la moitié des prédictions. Dans ce cas, il semble probable que le modèle puisse mieux prédire la classe "1" que la classe "0".

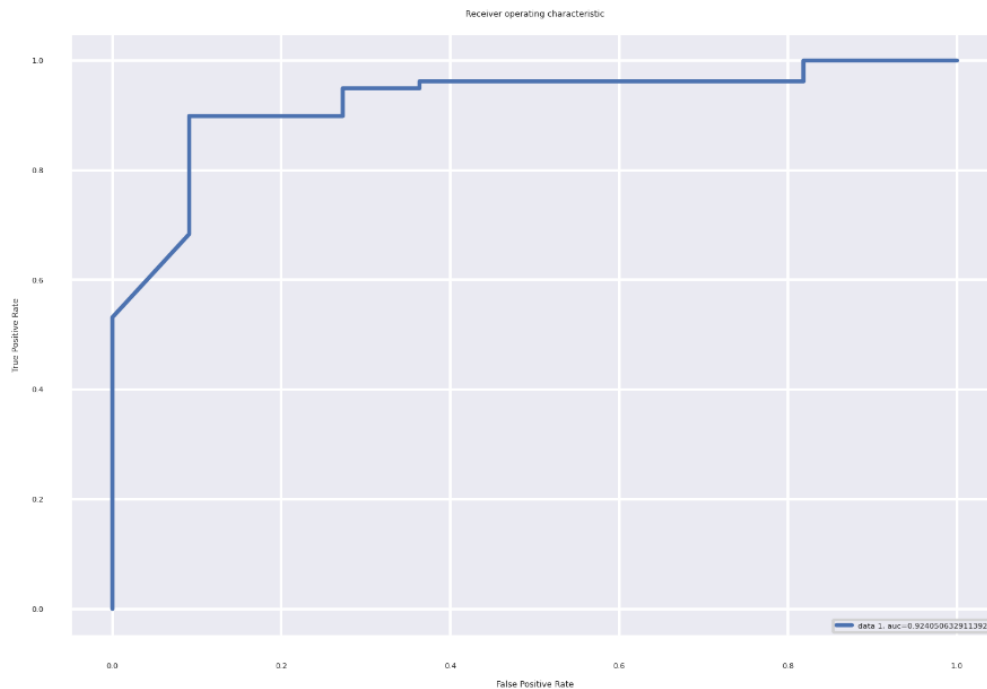
6.1.2 Resultat de Random forest

```
0.9883040935672515
0.9
          precision    recall  f1-score   support

     0       0.57       0.73       0.64         11
     1       0.96       0.92       0.94         79

 accuracy          0.90         90
 macro avg       0.77       0.83       0.79         90
weighted avg       0.91       0.90       0.91         90
```





L'algorithme Random Forest est plus approprié que le SVM pour notre tâche de Classification. En termes de justesse, on a réussi d'obtenir 98% dans la phase d'apprentissage et 90% dans la phase de test.

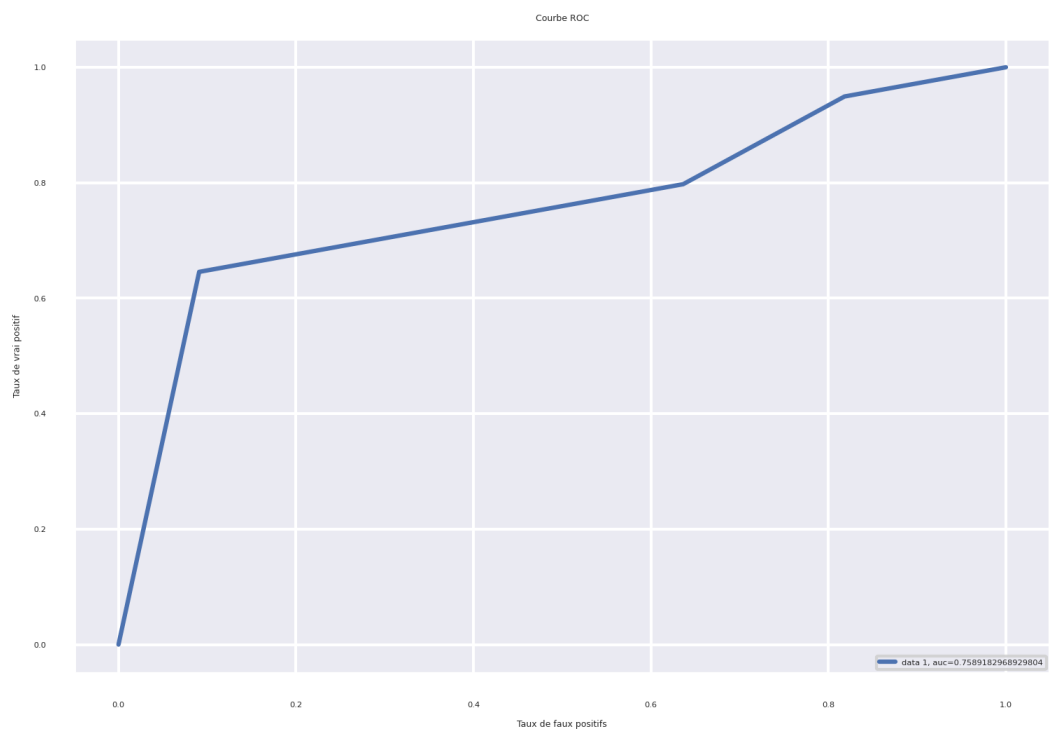
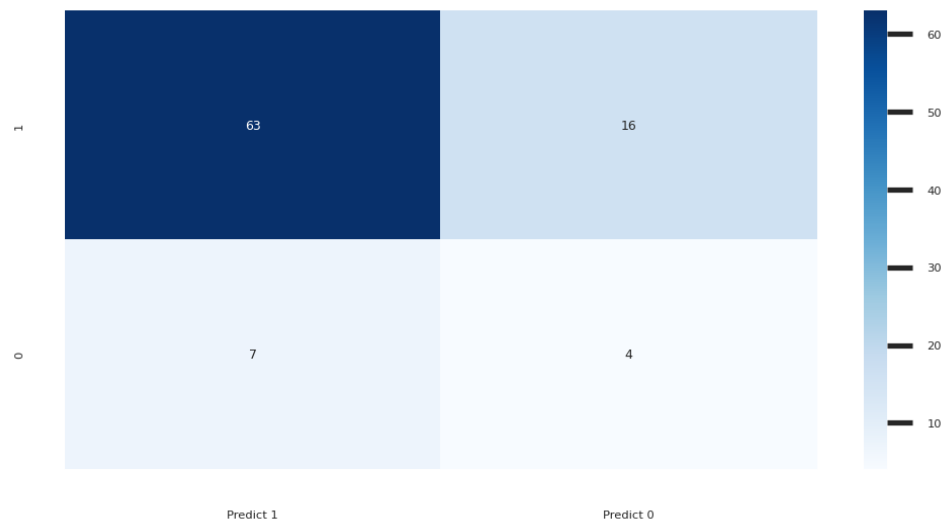
En analysant la figure qui représente la matrice de confusion résultant de l'utilisation de l'algorithme Random Forest, on voit qu'à partir de 79 données de test correspondant à des personnes souffrant de la maladie de Parkinson, 73 données de test ont été correctement classifiées (même nombre que le SVM). D'autre part, quand on regarde les données de test de personnes en bonne santé, on peut voir qu'à partir de 11 échantillons, le Random Forest a bien classifié 8 d'entre eux. En conclusion, ce modèle a réussi de mieux apprendre à reconnaître la classe "0".

6.1.3 Resultats de Decision Tree

```
0.8362573099415205
0.7444444444444445
      precision    recall  f1-score   support

     0       0.20      0.36      0.26         11
     1       0.90      0.80      0.85         79

 accuracy          0.74         90
 macro avg         0.55      0.58      0.55         90
 weighted avg      0.81      0.74      0.77         90
```



7 Conclusions

Afin d'apporter une contribution à la détection de la maladie de Parkinson par la voix, ce projet travaille avec des enregistrements vocaux de deux types de bases de données une binaire avec 50 voix de sujet sains et 50 MP et l'autre avec seulement les enregistrements de personnes souffrant de la MP. Les enregistrements vocaux sont de type .Wav ; un format de fichier audio numérique utilisé pour stocker des données de forme d'onde avec un taux d'échantillonnage de 44,1 kHz et un débit binaire de 16 bits. Avant l'extraction des caractéristiques, il a été proposé de supprimer les silences et les saturations. Pour cela nous avons développé un script qui devise les enregistrements audios en segments d'une seconde, puis il implémente une méthode de suppression de silence très simple qui calcule l'énergie comme la somme des carrés des échantillons, ensuite il calcule un seuil à 50 % de la valeur d'énergie médiane, enfin il conserve les segments dont l'énergie est supérieure à ce seuil. Une autre application a été développée pour la visualisation de l'état du signal vocal ; notre script fait appel à Parselmouth [63] ; une bibliothèque Python pour le logiciel Praat permettant d'analyser, de synthétiser et de manipuler la parole. Cette application nous permet d'extraire le tracé visuel de l'amplitude, de l'énergie, du pitch, du harmonic to the noise ratio et de spectrogramme avec les quatre premiers formants. Une fois que les enregistrements audios de deux bases de données sont traités, nous avons procédé à l'étape suivante, qui s'agit de l'extraction des caractéristiques. À ce propos, nous avons utilisé Surfboard [66], la bibliothèque open-source de python qui est basée sur des paquets d'analyse audio et met l'accent sur la capacité à traiter de grands ensembles de données nécessaires pour les approches modernes d'apprentissage automatique. Elle a été construite sur la base d'un certain nombre de paquets, tels que LibROSA [67] et pysptk [68], avec une variété de fichiers contenant des fonctions pour le calcul des caractéristiques nécessaires pour notre étude. Nous étions intéressés aux variantes de jitters et shimmers, le rapport harmoniques/bruit, les formants (F1, F2, F3 et F4), l'analyse de fluctuation détendue et l'entropie de la période du pitch. Après la construction de nos bases de données et avoir vérifié que toutes les mesures sont correctement classées dans le bon type de donnée, on fait appel à la méthode .corr() pour mesurer la force de corrélation entre les caractéristiques de chaque base de données. À partir de l'interprétation des résultats de la corrélation, on remarque que les ensembles de données avaient de nombreuses dimensions qui sont fortement corrélées, comme les variantes de jitters et de shimmers ; sont corrélées positivement, cependant l'harmonique to noise ratio est corrélé avec eux négativement. Cela pourrait faire de nos données un bon modèle pour l'analyse en composantes principales (PCA) ; une méthode très utile basée sur les mathématiques et les statistiques en éliminant les caractéristiques qui fournissent moins d'informations, et de ne considérer que celles qui contiennent les informations principales ayant la plus grande capacité pour l'identification de la voix pathologique. D'après la première application de PCA nous avons pu d'expliquer plus de 90 % de la variance de notre ensemble de données avec les 15 premières composantes principales. Par la suite nous avons appliqué les algorithmes d'apprentissage automatiques sur les ensembles de données réduite et nous avons remarqué que Random forest est le meilleur classificateur avec une précision de 90%.

8 Bibliographie

- [1] Garcia Ruiz PJ (1996) *Neurologia biblica*. Rev Neurol 24:580–581
- [2] Sider D, McVaugh M. Galen on tremor, palpitation, spasm, and rigor. Trans Stud Coll Physicians Phila. 1979; 1:183-210.
- [3] Calne DB, Dubini A, Stern G. Did Leonardo describe Parkinson's disease? New Engl J Med. 1989; 320:594.
- [4] Brissaud É. Leçons sur les maladies nerveuses (Deuxième série, Hôpital Saint-Antoine) recueillies et publiées par Henry Meige. Paris: Masson; 1899.
- [5] Merritt, H. H., A Textbook of Neurology. Philadelphia: Lea and Febiger, 1955.
- [6] Buck, J. F., and Cooper, I. S., Speech problems in Parkinsonian patients undergoing anterior choroidal artery occlusion or chemopallidectomy. J. Amer. geriat. Soc., 4, 1956, 1285-1290.
- [7] Hoberman, S. G., Speech techniques in aphasia and Parkinsonism. J. Mich. St. med. Soc., 57, 1958, 1720-1723.
- [8] Morly, D. E., The rehabilitation of adults with dysarthric speech. Speech Hear. Dis., 20, 1955, 5864.
- [9] Grewel, F, Dysarthria in post-encephalitic Parkinsonism. Acta psychiat, neurol. Stand., Vol. 32, Fase. 4, 1957, 440-449
- [10] G. J. Canter. "Speech characteristics of patients with Parkinson's disease: I. Intensity, pitch, and duration." Journal of Speech and Hearing Disorders, Vol. 28, No. 3, pp. 221–229, 1963.
- [11] G. J. Canter. "Speech characteristics of patients with Parkinson's disease: II. Physiological support for speech." Journal of Speech and Hearing Disorders, Vol. 30, No. 1, pp. 44–49, 1965.
- [12] F.L. Darley, A. E. Aronson and J. R. Brown, Differential diagnostic patterns of dysarthria. J Speech Hear Res, Vol. 12, pp. 246-269, 1969.
- [13] F.L. Darley, A. E. Aronson and J. R. Brown, Motor Speech Disorders, Philadelphia: WB Saunders.
- [14] J.A. Logemann, H.B. Fisher, B. Boshes and E.R. Blonsky, —Frequency and cooccurrence of vocal tract dysfunctions in the speech of a large sample of Parkinson patients, Journal of Speech and Hearing Disorders, Vol. 43, pp. 47-57, 1978.
- [15] K. S. Perez, L. O. Ramig, M. E. Smith, and C. Dromey, "The Parkinson larynx: tremor and videostroboscopic findings," Journal of Voice, vol. 10(4), pp. 354- 361, 1996.
- [16] S. Skodda, W. Visser, and U. Schlegel. "Vowel articulation in Parkinson's disease." Journal of Voice, Vol. 25, No. 4, pp. 467–472, 2011. Erratum in Journal of Voice. 2012 Mar;25(2):267-8.
- [17] J. Ruzs, R. Cmejla, H. Ruzickova, and E. Ruzicka. "Quantitative acoustic measurements for characterization of speech and voice disorders in early untreated Parkinson's disease." Journal of the Acoustical Society of America, Vol. 129, No. 1, pp. 350–367, 2011.
- [18] J. Möbes, G. Joppich, F. Stiebritz, R. Dengler, and C. Schröder. "Emotional speech in Parkinson's disease." Movement Disorders, Vol. 23, No. 6, pp. 824– 829, 2008.
- [19] L. O. Ramig, C. Mead, Y. H. R. Scherer, K. Larson, and D. Kohler. "Voice therapy and Parkinson's disease: a longitudinal study of efficacy." In: Proceedings of the Clinical Dysarthria Conference, San Diego, USA, 1988.
- [20] S. Sapir, J. Spielman, L. O. Ramig, B. Story, and C. Fox. "Effects of intensive voice treatment (the Lee Silverman Voice Treatment [LSVT]) on vowel articulation in dysarthric individuals with idiopathic Parkinson disease: Acoustic and perceptual findings." Journal of Speech, Language, and Hearing Research, Vol. 50, pp. 899–912, 2007.
- [21] F. J. Jiménez-Jiménez, J. Gamboa, A. Nieto, J. Guerrero, M. Ortí-Pareja, J. Molinas, E. García-Albea, and I. Cobeta. "Acoustic voice analysis in untreated with Parkinson's disease." Parkinsonism & Related Disorders, Vol. 3, No. 2, pp. 111–116, 1997.
- [22] J. Gamboa, F. J. Jiménez-Jiménez, A. Nieto, J. Montojo, M. Ortí-Pareja, J. A. Molina, E. García-Albea, and I. Cobeta. "Acoustic voice analysis in patients with Parkinson's disease treated with dopaminergic drugs." Journal of Voice, Vol. 11, No. 3, pp. 314–320, 1997.

- [23] D. J. Gelb, E. Oliver, and S. Gilman, "Diagnostic criteria for Parkinson disease," *Arch Neurol.*, vol. 56, pp. 33-39, 1999.
- [24] A. Tsanas, M. A. Little, P. E. McSharry, and L. O Ramig, "Accurate telemonitoring of Parkinson's disease progression by non-invasive speech tests," *IEEE transactions on Biomedical Engineering*, vol. 57(4), pp. 884-893, 2009.
- [25] P. Bugalho, and M. Viana-Baptista, "REM sleep behavior disorder and motor dysfunction in Parkinson's disease - A longitudinal study," *Park Relat Disord*, vol. 19(12), pp.1084-1087, 2013.
- [26] B. Harel, M. Cannizzaro and P.J. Snyder, —Variability in fundamental frequency during speech in prodromal and incipient Parkinson's disease: A longitudinal case study,|| *Brain and Cognition*, vol. 56, pp. 24–29, 2004.
- [27] P. Boersma and D. Weenink. "PRAAT, a system for doing phonetics by computer." *Glott International*, Vol. 5, No. 9/10, pp. 341–345, 2001.
- [28] M. Little, P. McSharry, S. Roberts, D. Costello, and I. Moroz. "Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection." *Biomedical Engineering Online*, Vol. 6, p. 23, 2007
- [29] J. R. Orozco-Arroyave, J. F. Vargas-Bonilla, J. D. Arias-Londoño, S. Murillo-Rendón, G. Castellanos-Domínguez, and J. F. Garcés. "Nonlinear dynamics for hypernasality detection in spanish vowels and words." *Cognitive Computation*, Vol. 5, No. 4, pp. 448–457, 2013.
- [30] Sakar, C. O., & Kursun, O. (2009). Telediagnosis of Parkinson's Disease Using Measurements of Dysphonia. *Journal of Medical Systems*, 34(4), 591–599.
- [31] Bhattacharya, I., & Bhatia, M. P. S. (2010). SVM classification to distinguish Parkinson disease patients. *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India -*
- [32] Shirvan, R. A., & Tahami, E. (2011). Voice analysis for detecting Parkinson's disease using genetic algorithm and KNN classification method. 2011 18th Iranian Conference of Biomedical Engineering (ICBME).
- [33] Uma Rani, K., & Holli, M. S. (2012). Analysis of speech characteristics of neurological diseases and their classification. 2012 Third International Conference on Computing, Communication and Networking Technologies (ICCCNT'12).
- [34] Sakar, B. E., Isenkul, M. E., Sakar, C. O., Sertbas, A., Gurgen, F., Delil, S., ... Kursun, O. (2013). Collection and Analysis of a Parkinson Speech Dataset With Multiple Types of Sound Recordings. *IEEE Journal of Biomedical and Health Informatics*, 17(4), 828–834.
- [35] M. Shahbakhi, D.T. Far, E. Tahami, "Speech analysis for diagnosis of Parkinson's Disease using Genetic Algorithm and Support Vector Machine," *J Biomed Sci Eng*, vol. 7, pp. 147-156, 2014.
- [36] A. Benba, A. Jilbab, A. Hammouch, and S. Sandabad, "Voiceprints analysis using MFCC and SVM for detecting patients with Parkinson's disease," *International conference on electrical and information technologies (ICEIT)*, pp. 300-304, March 2015.
- [37] A. Benba, A. Jilbab, A. Hammouch, "Discriminating between patients with parkinson's and neurological diseases using cepstral analysis," *IEEE Trans Neural Syst Rehabil Eng*, vol. 24(10), pp. 1100-1108, 2016.
- [38] Anand, A., Haque, M. A., Alex, J. S. R., and Venkatesan, N. (2018). "Evaluation of machine learning and deep learning algorithms combined with dimensionality reduction techniques for classification of Parkinson's disease," in 2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT) (Louisville, KY), 342–347.
- [39] S. Mostafa, A. Mustapha, M. A. Mohammed, R.I. Hamed, N. Arunkumar, M. K. A. Ghani, M. M. Jaber, and S. H. Al-Dulaimi, "Examining multiple feature evaluation and classification methods for improving the diagnosis of Parkinson's disease," *Cogn Syst Res.*, vol. 54, pp. 90-99, 2019.
- [40] Karapinar Senturk, Z. (2020). Early Diagnosis of Parkinson's Disease Using Machine Learning Algorithms. *Medical Hypotheses*, 109603.
- [41] Cette figure est une version adaptée de l'image partagée en : <http://www.frcneurodon.org/wp-content/uploads/2015/11/Mouvement-2.jpg>
- [42] Cette figure est une version adaptée de l'image partagée en : <https://commons.wikimedia.org/wiki/File:Basal-ganglia-coronal-sections-large.png>

- [43] Seikel, J., King, D., & Drumright, D. (2010). *Anatomy & Physiology for Speech, Language, and Hearing* (4th ed., pp. 1-763). Clifton Park, NY: Delmar Cengage Learning.
- [44] Cette figure est une version adaptée de l'image partagée en : <http://2.bp.blogspot.com/-UOMvXvLyw3c/VV4gCcCTVRI/AAAAAAAAADnI/f1WrMn7HFeY/s1600/Larynx.png>
- [45] Cette figure est une version adaptée de l'image partagée en : <http://www.barcode.ro/tutorials/biometrics/img/speech-production.jpg>
- [46] I.R. Titze, *Principles of Voice Production*. National Center for Voice and Speech, Iowa City, US, 2nd printing, 2000
- [47] Cette figure est une version adaptée de l'image partagée en : <https://link.springer.com/article/10.1007/s00034-020-01373-2/figures/2>
- [48] Cette figure est une version adaptée de l'image partagée en : <https://wiki.aalto.fi/display/ITSP/Speech+production+and+acoustic+properties>
- [49] Yannick Jadoul Revision 312a1947, Copyright 2017-2021: <https://parselmouth.readthedocs.io/en/stable/>
- [50] D. Talkin, A robust algorithm for pitch tracking, in *Speech coding and synthesis* (Eds. W.B. Kleijn and K.K. Paliwal), Elsevier, 1995
- [51] H. Ney, —Dynamic programming algorithm for optimal estimation of speech parameter contours, *IEEE Transactions on Systems, Man, Cybernetics*, Vol. 13, pp. 208- 214, 1983
- [52] Speech Signal Processing Toolkit (SPTK): <http://sp-tk.sourceforge.net/>
- [53] I.R. Titze, —Nonlinear source-filter coupling in phonation: Theory, *Journal of Acoustical Society of America*, Vol. 123 (5), pp. 2733-2749, 2008
- [54] M.A. Little, P.E. McSharry, L.O. Ramig, —Nonlinear speech analysis algorithms mapped to a standard metric achieve clinically useful quantification of average Parkinson's disease symptom severity, *Journal of the Royal Society Interface*, Vol. 8, pp. 842-855, 2011.
- [55] Manolakis, D. G., & Proakis, J. G. (1996). *Digital Signal Processing. Principles, Algorithms, And Applications*. <https://vdocument.in/sign-sysprokais-manolakis.html>
- [56] Awan, S. N., & Frenkel, M. L. (1994). Improvements in estimating the harmonics-to-noise ratio of the voice. *Journal of Voice*, 8(3), 255–262. doi:10.1016/s0892-1997(05)80297-8. [https://sci-hub.se/10.1016/s0892-1997\(05\)80297-8](https://sci-hub.se/10.1016/s0892-1997(05)80297-8)
- [57] Deshpande, S., Chikkerur S., Govindaraju, V., 2005. "Accent classification in speech," *Proceedings of the 4th IEEE Workshop on Automatic Identification Advanced Technologies*, pp. 139-143.
- [58] Yumoto E. The quantitative evaluation of hoarseness: a new harmonic to noise ratio method. *Arch Otolaryngol* 1983; 109:48-52.
- [59] A. Tsanas, Max A. Little, "Accurate telemonitoring of Parkinson's disease progression by non-invasive speech test", *IEEE Transactions on Biomedical Engineering*, vol. 57, no. 4, April 2010, [Online]. Available: <https://ieeexplore.ieee.org/document/5339170>
- [60] A. Zambrano, «Exploración de Nuevos Métodos para la Detección de la Enfermedad de Parkinson mediante la voz,» Trabajo de fin de Máster. Univ. Politécnica de Madrid. España, 2017
- [61] <https://www.synapse.org/#!Synapse:syn2321745/wiki/62077>
- [62] M.A. Little, P. E. McSharry, E. J. Hunter, J. Spielman, L. O. Ramig, —Suitability of dysphonia measurements for telemonitoring of Parkinson's disease, *IEEE Transactions, Biomedical Engineering*, Vol. 56 (4), pp. 1015-1022, 2009
- [63] <https://github.com/YannickJadoul/Parselmouth>
- [64] <https://github.com/jiaaro/pydub>
- [65] <https://surfboard.readthedocs.io/en/latest/> and <https://github.com/novoic/surfboard>
- [66] <https://librosa.org/doc/latest/index.html>
- [67] <https://pysptk.readthedocs.io/en/latest/>
- [68] Orozco-Arroyave, J.R., Arias-Londoño, J.D., Vargas-Bonilla, J.F., González-Rátiva, M.C., and Nöth E.: New Speech Corpus Database for the Analysis of People Suffering From Parkinson's Disease. In *Proc. Of the International Conference on Language Resources and Evaluation (Irec)*, pages 342-347, Reykjavik, Iceland. 2014.

- [69] G.C. Cotzias, —L-dopa for Parkinsonism, *N Engl J Med.* 1968 Mar 14; Vol. 278, pp. 630. doi: 10.1056/nejm196803142781127.
- [70] H. Ackermann and W. Ziegler. “Articulatory deficits in Parkinsonian dysarthria: an acoustic analysis”. *Journal of Neurology, Neurosurgery, and Psychiatry*, Vol. 54, No. 12, pp. 1093–1088, 1991.
- [71] J. I. Godino-Llorente, P. Gómez-Vilda, and M. Blanco-Velasco. “Dimensionality reduction of a pathological voice quality assessment system based on Gaussian mixture models and short-term cepstral parameters”. *IEEE transactions on biomedical engineering*, Vol. 53, No. 10, pp. 1943–1953, 2006.
- [72] T. Bocklet, S. Steidl, E. Nöth, and S. Skodda. “Automatic evaluation of Parkinson’s speech - acoustic, prosodic and voice related cues”. In: *Proceedings of the 14th Annual Conference of the International Speech Communication Association (INTERSPEECH)*, pp. 1149–1153, Lyon, France, 2013.
- [73] P. Boersma, “Accurate short-term analysis of the fundamental frequency and the harmonics to-noise ratio of a sampled sound” *Institute of Phonetic Sciences, University of Amsterdam, Proceedings 17* (1993), 97-110.

Annexe A : PLM-PDRS-Questionnaire.pdf