**L'utilisation d'arbres de décision pour classifier la présence ou l’absence de la maladie de Parkinson par la performance dactylographique**

Rodrigo Alejandro Flores-Soto, BSc, MD, CCMF (MU) et Tristan Martin, BSc (Hons.)

**1. Introduction**

La maladie de Parkinson est le deuxième trouble neurologique progressif le plus répandu au monde. Au Canada, on compte présentement plus de 106 000 personnes atteintes et près de 12 000 nouveaux cas annuellement (Public Health Agency of Canada, 2023). Histopathologiquement définie par l’accumulation d’alpha-synucléine au sein de certaines régions du système nerveux central, cette maladie peut atteindre le système moteur, autonome et psychiatrique. Elle est caractérisée par des symptômes moteurs emblématiques : un tremblement initialement asymétrique, une rigidité, une bradykinésie et une instabilité posturale. Bien qu’il n’existe pour l’instant aucun traitement curatif à cette maladie, les symptômes peuvent être adressés avec une combinaison de médicaments ou de thérapies et substantiellement améliorer la qualité de vie du patient atteint d’où l’importance de détecter précocement cette entité (Schrag et al., 2000). Présentement, son diagnostic est essentiellement clinique, basé sur l’histoire du patient et son examen physique (Postuma et al., 2015).

Nous savons que ce trouble du mouvement entraîne des répercussions fonctionnelles sur plusieurs sphères de la vie des patients atteints, particulièrement sur les activités nécessitant une motricité fine des membres supérieures comme l’alimentation ou l’utilisation d’un ordinateur (Cunningham et al., 2009). Dans une réalité où l’accessibilité à un clavier d’ordinateur est presque ubiquitaire, il serait intéressant d’utiliser ce dernier comme outil de dépistage. Nous souhaitons construire un classificateur binaire qui serait capable de discriminer les patients parkinsoniens des sujets sains à partir de leur signature dactylographique.

Un outil de ce genre serait simple et peu onéreux à déployer, il serait accessible à la majorité de la population nord-américaine, incluant les communautés peuplant des régions éloignées où l’accessibilité à des évaluations médicales est limitée. Il pourrait accélérer le diagnostique et la prise en charge subséquente des patients parkinsoniens. Se montrant efficace, cet outil ouvrirait aussi la voie au suivi et monitoring quotidien du patient malade, ce qui permettrait un ajustement personnalisé et en temps réel des traitements pharmacologiques selon l’évolution de la maladie.

**2. Méthodologie**

2.1 Prétraitement des données et ingénierie des caractéristiques

Nous avons utilisé les banques de données PD\_MIT-CS1PD et PD\_MIT-CS2PD pour notre analyse. Ces ensembles contiennent l’enregistrement temporelle des frappes de clavier collectés auprès de 85 sujets atteint ou pas de la maladie de Parkinson, où l’épreuve principale consistait à la transcription de textes en Italien de longueur et de contenu variable. D’ailleurs, l’enregistrement du nombre d’appuis d’une même touche en un lapse de temps, effectué d’abord avec la main dominante et ensuite avec la main non-dominante, consistait en une autre épreuve dont la moyenne a été enregistrée sous le nom *sTap*.

Le prétraitement des données a été réalisé en python v3.11.5 avec la librairie pandas v2.1.1 dans un environnement Windows 10. Les données utilisées provenaient de 2 différentes banques de données : PD\_MIT-CS1PD qui contenait deux enregistrements de données de frappe pour 34 patients et PD\_MIT-CS2PD qui n’en contenait qu’un seul pour 51 patients. Chacune de ses banques contenait aussi un fichier regroupant pour chaque pID (« patient ID ») les informations suivantes :

* *gt* : « ground truth » par rapport au diagnostic de parkinson.
* *updrs108* : Unified Parkinson’s Disease Rating Scale.
* *afTap* : « alternate finger tapping ».
* *sTap* : « single finger tapping ».
* *nqScore :* IndexNeuroQWERTY*.*
* *typingSpeed* : vitesse de frappe.
* *file\_1* et *file\_2* : le nom des fichiers contenant les données brutes de frappe sous formes de 4 colonnes, soit la touche frappée, la durée en seconde qu’elle a été enfoncée, le temps en seconde par rapport au début du test à laquelle elle a été frappée et le temps en seconde par rapport au début du test à laquelle elle a été relâchée.

La première étape du pré-traitement des données a consisté à mettre en commun ces fichiers, respectivement nommés GT\_DataPD\_MIT-CS1PD et GT\_DataPD\_MIT-CS2PD. Un premier DataFrame nommé *addedscores* fut créé à partir de ces derniers. Nous avons ensuite effectué une concaténation des fichiers de données brutes *file\_1* et *file\_2* (lorsque présent) pour ne travailler que sur un seul fichier de données brutes par patient. Après une exploration des données nous avons noté que certains fichiers contenaient des erreurs intrinsèques où le nom du fichier était intégré à la valeur en secondes du début du test ce qui rapportait des données aberrantes. Cette erreur pouvait se retrouver dans la première, deuxième ou troisième ligne en fonction des touches ou combinaisons de touches frappées, mais elle ne se retrouvait jamais dans le deuxième mot. Pour corriger cette erreur nous avons retiré toutes les lignes avant la première frappe de la barre d’espace pour débuter notre analyse des données avec l’entrée « space » avant le deuxième mot. En retravaillant les données brutes nous avons exclu tout les caractères spéciaux ou caractères nécessitant une combinaison de touches pour être produit pour tenter d’éliminer le bruit que ces frappes atypiques pourraient produire. Nous n’avons donc qu’essentiellement gardé la frappe des lettres de « a » à « z », les chiffres de 0 à 9, la barre d’espace et remplacé les autres entrées par des NaNs tout en gardant leurs caractéristiques temporelles.

Grâce aux données brutes, les auteurs de ce jeu de données ont intégré comme caractéristiques la vitesse de frappe, le score sTap, ainsi que l’index neuroQWERTY obtenu par un algorithme de régression d'ensemble validé (Giancardo et al., 2016) (Arroyo-Gallego et al., 2018). Cependant, nous avons éliminé la colonne de données *updrs108*, car ce score de sévérité de la maladie est directement corrélé avec la cible, soit la présence (1) ou l’absence (0) de la maladie de Parkinson. La colonne afTap (Alternating finger tapping result) correspondant à l’évaluation des mouvements alternés rapides lors de l’examen physique, pouvant ainsi témoigner de la bradykinésie vue chez les patients parkinsoniens, a été également supprimé par soucis d’utiliser seulement des données provenant du test sur clavier.

A keyboard with numbers and symbols

Description automatically generated

Figure 1 : Schéma illustrant la notion de position sur un clavier QWERTY

Grâce à l’accès aux données brutes pour chaque patient, nous avons tenté d’améliorer leur ingénierie de fonctionnalités en intégrant le temps entre chaque frappe de touche correspondant à une lettre ou à la barre d'espace sur un clavier QWERTY, tel qu'il est utilisé en Italien, puis normalisé par la distance euclidienne entre les touches en unités arbitraires[[1]](#footnote-1). Chaque touche d'intérêt a été associée à un tuple dans un dictionnaire où les coordonnées correspondaient à la rangée et à la colonne de la touche, respectivement[[2]](#footnote-2) (voir figure 1). L'utilisation des caractères spéciaux et des autres fonctionnalités du clavier ou même de la souris a été ignorée par souci de simplicité. Bien que nous ayons implémenté une notion de distance généralisée (norme *Lp*), le choix de *p = 2*, soit la distance euclidienne (*L2*), était le plus approprié, car c'est la métrique qui minimise la distance entre deux points dans un espace Euclidien 2D tel la surface d’un clavier, ce qui est également la distance la plus naturellement empruntée par les doigts d’un utilisateur. Par simplicité, nous ignorons la distance parcourue par la pointe des doigts de l’usager dans l’axe perpendiculaire au clavier.

2.2 Description, justification et implémentation des modèles

Avec ce jeu de données traité avec l'ingénierie des fonctionnalités, nous avons utilisé *compare\_models* de la librairie pycaret v3.2.0 dans un environnement Python v3.8.5 sur macOS Monterey v12.6.3 pour produire une liste des meilleurs modèles en termes d’exactitude, de précision, de rappel, de l’aire sous la courbe ROC (AUC) et du score F1. Les trois meilleurs modèles étaient tous des méthodes d’ensemble utilisant des arbres de décision, soit *Extra Trees Classifier* (ETC), *Random Forest Classifier* (RFC) et *XGBoost* (XGB)

D’abord, la modélisation d’ensemble consiste à l’entrainement de plusieurs modèles où leurs prédictions individuelles sont ensuite agréées pour produire une prédiction finale. De plus, l’arbre de décision est un type de modèle d’apprentissage supervisé qui divise le jeu de données en sous-ensembles où chaque nœud de l’arbre représente une décision basée sur une caractéristique (ex : vitesse de frappe) du jeu de donnée. Un facteur important qui différentie ces modèles est leur méthode d’ensemble. Spécifiquement, les RFC utilisent comme méthode d’ensemble le *bootstrap aggregating* qui consiste à créer plusieurs jeu d'entraînement en échantillonnant de manière aléatoire et avec remplacement les données du jeu d'entraînement initial (Goodfellow et al., 2016). Ces ensembles variés sont ensuite utilisés pour entraîner plusieurs modèles, et leurs prédictions sont agrégées (Goodfellow et al., 2016). ETC construit aussi plusieurs arbres, mais utilise l’entièreté du jeu de donnée initial pour l’entrainement de chaque model tout en introduisant de l’aléatoire dans ces jeux. En revanche, le XGB exploite la technique de *boosting*, impliquant la création séquentielle de plusieurs *weak learners*, tels que des arbres de décision peu profonds (Goodfellow et al., 2016). Cependant, leur combinaison avec d'autres *weak learners* transforme l'ensemble en un modèle robuste (*strong learner*) (Goodfellow et al., 2016).

Les arbres de décision sont pertinents dans ce contexte de classification binaire pour plusieurs raisons. D’abord, la modélisation d’ensemble inhérent à ces modèles aide a palier la basse taille du jeu d’entrainement pour optimiser la généralisabilité. D’ailleurs, ces modèles sont aptes à apprendre des associations non-linéaires entre les caractéristiques et la cible, ce qui est probablement le cas ici. Ils ont aussi le bonus d’être interprétable, contrairement à des modèles d’apprentissage profond.

Nous avons utilisé une recherche en grille pour explorer une gamme de seuils d’information mutuelle entre chaque caractéristique et la cible nécessaire pour qu’une caractéristique soit inclus au sein du model. Les valeurs de seuil sont choisies dans un espace uniforme (0,05 à 0,2) avec 15 incréments. Pour déterminer le seuil optimal, une approche de validation croisée 15-fold a été appliquée. Pour les caractéristiques sélectionnées au seuil optimal, nous avons effectué une analyse de variance (ANOVA) afin d'évaluer leur signification statistique. Les valeurs F et p de l'ANOVA sont calculées pour chaque pair caractéristique/cible grâce à la fonction *f\_classif* de la libraire *sklearn*. Un seuil de signification de 0,05 est fixé et les caractéristiques dont les valeurs *p* sont inférieures à ce seuil sont considérées comme statistiquement significatives. L’arbre de décision est ensuite entrainé sur les caractéristiques statistiquement significatives au-delà du seuil optimal. Tous nos modèles ont été implémenté grâce à sklearn v1.2.2.

**3. Résultats**

Pour tous nos modèles, les caractéristiques dépassant le seuil optimal d’information mutuelle avec la cible étaient le *nqScore* et le temps entre chaque frappe normalisée par la distance euclidienne. Cependant, le *nqScore* était statistiquement significatif (p = 2.4e-5 < 5e-2), tandis que le temps normalisé ne l’était pas (p = 2e-1 > 5e-2). Malgré nos bonnes intentions d’améliorer l’ingénierie de fonctionnalités faite par les auteurs du jeu de donnée, c’était en vain et le *nqScore*, qui a déjà été validé en clinique (Arroyo-Gallego et al., 2018),était la seule caractéristique avec une puissance prédictive statistiquement significative.

La précision, le rappel et le score F1 selon la classe prédite (Vrai = sujet Parkinsonien, Faux = sujet sain) ainsi que l’exactitude pour chaque modèle sont présentés dans le tableau 1. Mis appart le rappel pour la classe faux, le FRC est le meilleur modèle selon tous les métriques calculées.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Modèle | Classe | Précision | Rappel | Score F1 | Exactitude |
| ETC | Vrai | 0.55 | 0.75 | 0.63 | 0.59 |
| Faux | 0.67 | 0.44 | 0.53 |
| FRC | Vrai | 0.60 | 0.75 | 0.67 | 0.65 |
| Faux | 0.71 | 0.56 | 0.63 |
| XGB | Vrai | 0.55 | 0.44 | 0.53 | 0.59 |
| Faux | 0.67 | 0.75 | 0.63 |

**Tableau 1** : Précision, rappel et score F1 selon la classe prédite (Vrai = sujet Parkinsonien, Faux = sujet sain) avec l’exactitude pour chaque modèle, la meilleure valeur étant surligné en jaune. Les meilleures valeurs pour la précision, le rappel et score F1 sont surlignées en vert et en rouge pour les classes vrai et faux, respectivement.

**4. Discussion**

Il est bien établi que les stigmates moteurs du Parkinsons peuvent être utilisés pour son diagnostic. Cette étude à tenter d’explorer la possibilité d’utiliser des modèles d’apprentissage machine afin de discerner les patients atteints de la maladie de Parkinson à partir de leur signature dactylographique. Bien que les classificateurs obtenus ne soient pas parfaits, les résultats sont encourageants. Les modèles simples que nous avons entraînés semblent effectivement détecter un signal dactylographique différenciant les patients sains de ceux atteints. Lorsque nos résultats sont comparés à d’autre équipes ayant tenté de répondre à la même question, nos modèles simples semblent à première vue moins bien performer. Une première équipe (Lan et al., 2019) rapporte une sensibilité de 70% et une spécificité de 80% alors qu’une autre (Adams, 2017) rapporte une sensibilité 96%, spécificité 97% et une ROC AUC de 0.98. Il est toutefois pertinent de noter que les résultats qui semble les plus impressionnants proviennent d’une étude incluant seulement 103 sujets où l’algorithme final est en fait un méta-classificateur composé de 8 modèles d’apprentissage machine différents agrégés en un seul. Cette architecture méthodologique pourrait nous faire faire suspecter du surapprentissage d’ensemble.

Ce dernier point nous apporte à réfléchir sur nos propres limitations méthodologiques. Il est évident qu’un ensemble de données composé de plus de 85 participants, comme celui que nous avons utilisés, nous permettrait d’obtenir une bien meilleure exactitude dans l’identification du signal dactylographique générée par le patient parkinsonien. Cette exploration ne tenait pas compte de la sévérité du parkinson lorsque le patient était atteint ni des aptitudes dactylographiques préalables, tous deux des facteurs qui auraient pu moduler l’aisance avec laquelle le sujet aurait pu taper sur un clavier d’ordinateur. Il aurait aussi intéressant de leur faire taper un texte standardisé aux participants afin introduire la notion d’erreur de frappe dans notre analyse des données.

Bien qu’il n’existe pas encore de modèle assez robuste pour être utilisé cliniquement, nous pensons que l’exploration de l’analyse des marqueurs biométriques pour diagnostiquer précocement le Parkinson demeure une question d’importance. Avec le développement des montres intelligentes, une des prochaines avenues à sonder pourrait être d’interroger l’accéléromètre de ces dernières pour voir si la différence entre la démarche d’un patient sain ou atteint de Parkinson peut être perceptible.

**Bibliographie :**

1. Arroyo-Gallego T, Ledesma-Carbayo MJ, Butterworth I, Matarazzo M, Montero-Escribano P, Puertas-Martín V, Gray ML, Giancardo L, Sánchez-Ferro Á. Detecting Motor Impairment in Early Parkinson's Disease via Natural Typing Interaction With Keyboards: Validation of the neuroQWERTY Approach in an Uncontrolled At-Home Setting. J Med Internet Res. 2018 Mar 26;20(3):e89. doi: 10.2196/jmir.9462. PMID: 29581092; PMCID: PMC5891671.
2. Cunningham LM, Nugent CD, Finlay DD, Moore G, Craig D. A review of assistive technologies for people with Parkinson's disease. Technol Health Care. 2009;17(3):269-79. doi: 10.3233/THC-2009-0547. PMID: 19641262.
3. Goodfellow, I.J., Bengio, Y., Courville, A. *Deep Learning*. MIT Press, 2016.
4. Giancardo, L., Sánchez-Ferro, A., Arroyo-Gallego, T. *et al.* Computer keyboard interaction as an indicator of early Parkinson’s disease. *Sci Rep* **6**, 34468 (2016). https://doi.org/10.1038/srep34468
5. Postuma RB, Berg D, Stern M, Poewe W, Olanow CW, Oertel W, Obeso J, Marek K, Litvan I, Lang AE, Halliday G, Goetz CG, Gasser T, Dubois B, Chan P, Bloem BR, Adler CH, Deuschl G. MDS clinical diagnostic criteria for Parkinson's disease. Mov Disord. 2015 Oct;30(12):1591-601. doi: 10.1002/mds.26424. PMID: 26474316.
6. Public Health Agency of Canada. Canadian Chronic Disease Surveillance System (CCDSS), Data Tool 2000–2020, 2022 Edition. Ottawa (ON): Public Health Agency of Canada; 2023
7. Schrag A, Jahanshahi M, Quinn N. What contributes to quality of life in patients with Parkinson's disease? J Neurol Neurosurg Psychiatry. 2000 Sep;69(3):308-12. doi: 10.1136/jnnp.69.3.308. PMID: 10945804; PMCID: PMC1737100.
8. Adams WR. High-accuracy detection of early Parkinson's Disease using multiple characteristics of finger movement while typing. PLoS One. 2017 Nov 30;12(11):e0188226. doi: 10.1371/journal.pone.0188226. PMID: 29190695; PMCID: PMC5708704.
9. Lan BL, Yeo JHW. Comparison of computer-key-hold-time and alternating-finger-tapping tests for early-stage Parkinson's disease. PLoS One. 2019 Jun 27;14(6):e0219114. doi: 10.1371/journal.pone.0219114. PMID: 31247037; PMCID: PMC6597101.

1. Voir feature\_eng.ipynb sur github: https://github.com/tristmartin/MMD6060/blob/main/feature\_eng.ipynb [↑](#footnote-ref-1)
2. Inspiré de https://stackoverflow.com/questions/72885324/find-the-distance-between-letters-in-the-keyboard [↑](#footnote-ref-2)