POLYPUBLIE
Polytechnique Montréal

POLYTECHNIQUE MONTRÉAL
UNIVERSITÉ D'INGÉNIERIE

Titre: Title:	Comparaison d'algorithmes symboliques et connexionnistes pour corréler l'âge d'enfants en santé avec des paramètres neuromoteurs Sigma-Lognormaux
Auteur: Author:	Zigeng Zhang
Date:	2022
Type:	Mémoire ou thèse / Dissertation or Thesis
Référence: Citation:	Zhang, Z. (2022). Comparaison d'algorithmes symboliques et connexionnistes pour corréler l'âge d'enfants en santé avec des paramètres neuromoteurs Sigma-Lognormaux [Master's thesis, Polytechnique Montréal]. PolyPublie. https://publications.polymtl.ca/10408/

Document en libre accès dans PolyPublie
Open Access document in PolyPublie

URL de PolyPublie: PolyPublie URL:	https://publications.polymtl.ca/10408/
Directeurs de recherche: Advisors:	Réjean Plamondon, & Christian O'Reilly
Programme: Program:	Génie biomédical

Ce fichier a été téléchargé à partir de PolyPublie, le dépôt institutionnel de Polytechnique Montréal
This file has been downloaded from PolyPublie, the institutional repository of Polytechnique Montréal

https://publications.polymtl.ca

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Comparaison d'algorithmes symboliques et connexionnistes pour corréler l'âge d'enfants en santé avec des paramètres neuromoteurs Sigma-Lognormaux

ZIGENG ZHANG
Institut de génie biomédical

Mémoire présenté en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*

Génie biomédical

Juin 2022

© Zigeng Zhang, 2022.

POLYTECHNIQUE MONTRÉAL
affiliée à l'Université de Montréal

Ce mémoire intitulé :

Comparaison d'algorithmes symboliques et connexionnistes pour corréler l'âge d'enfants en santé avec des paramètres neuromoteurs Sigma-Lognormaux

présenté par **Zigeng ZHANG**
en vue de l'obtention du diplôme de *Maîtrise ès sciences appliquées*
a été dûment accepté par le jury d'examen constitué de :

Benjamin DE LEENER, président
Réjean PLAMONDON, membre et directeur de recherche
Christian O'REILLY, membre et codirecteur de recherche
Abolfazl MOHEBBI, membre

DÉDICACE

À tous mes amis du labos,
vous me manquerez...

REMERCIEMENTS

D'abord, je remercie mon directeur de recherche, le Professeur Réjean Plamondon ainsi que le Professeur Christian O'Reilly pour leur support pendant toute ma maîtrise. Ils m'ont appris comment un chercheur indépendant doit prendre son travail au sérieux et surmonter les difficultés.

Je remercie aussi les autres collègues de laboratoire, qui m'ont donné les articles à référencer, et les données à analyser.

Je remercie ma famille qui me fait toujours confiance. Ses membres me supportent et m'encouragent à continuer mes études.

Je remercie les amis que j'ai rencontrés à Polytechnique, à Montréal, en France et sur Internet. Ils m'ont beaucoup soutenu moralement, surtout dans les moments les plus difficiles.

Finalement, je remercie encore toutes les personnes qui m'ont aidé pendant la période de pandémie, qu'elles soient mentionnées ou pas ci-dessus. Les expériences acquises pendant mes études de maîtrise sont un trésor que je n'oublierai jamais.

Je vous remercie beaucoup.

RÉSUMÉ

L'évaluation de la maturité du contrôle moteur est importante pour aider les médecins à diagnostiquer un retard ou une instabilité du développement moteur chez les enfants. Mais il est toujours difficile de concevoir des méthodes d'évaluation qui soient pratiques à mettre en œuvre et qui fournissent en même temps des évaluations précises. L'objectif de cette étude est de développer un algorithme efficace pour prédire la maturité du contrôle moteur basé sur la Théorie Cinématique des mouvements humains rapides. Pour cette étude, nous avons utilisé les enregistrements de 513 enfants (âgés de 5,5 à 13 ans) qui ont effectué des traits écrits sur une tablette électronique. Deux types de tests ont été investigués : le test du trait simple et le test du triangle. Dans un premier temps, les paramètres Sigma-Lognormaux ont été extraits des traces enregistrées et ont été utilisés comme entrée de notre modèle prédictif. Plusieurs modèles différents ont été comparés (régression linéaire, réseau de neurones, méthode des k plus proches voisins, forêt d'arbres décisionnels, amplification du gradient, machine à vecteurs de support). Ces différents modèles ont donné de bons résultats compte tenu de la variabilité intra-enfant des traits manuscrits et de la variabilité inter-enfant de la maturité du contrôle moteur. Le meilleur résultat a été obtenu avec les modèles basés sur les réseaux neuronaux : coefficient de détermination ($\mathbf{R^2}$) : 0,548 ; erreur absolue moyenne (MAE) : 0,937. Les modèles de régression linéaire donnent également de bons résultats ($\mathbf{R^2}$: 0,476 MAE : 1,032). Plus de 90% des prédictions ont une erreur relative inférieure à 20%. Nous avons constaté que l'utilisation des paramètres du test de trait simple n'est pas optimale ; les résultats sont meilleurs lorsqu'on utilise les paramètres du test de tracé triangulaire. En conclusion, notre étude montre que le modèle Sigma-Lognormal propose une nouvelle possibilité pour estimer la maturité du contrôle moteur. Cela nous donne une approche totalement nouvelle en complément des méthodes traditionnelles utilisées actuellement. Cette méthode est rapide et confortable pour l'utilisateur. Puisqu'il suffit aux enfants de faire des traits d'écriture sur une tablette électronique avec un stylo, il devrait être plus pratique pour les médecins d'utiliser ce test dans un contexte clinique pour les pré-pistages avec moins d'intervention professionnelle par rapport aux méthodes traditionnelles avec les questionnaires, c'est un moyen rapide et bon marché .

ABSTRACT

The evaluation of motor control maturity is important to help physicians diagnose delayed or unstable motor development in children. But it has traditionally been difficult to design assessment methods that are practical to implement and provide accurate evaluations at the same time. The aim of this study is to develop an effective algorithm to predict motor control maturity based on Kinematic Theory of rapid human movements. For this study, we used recordings from 513 children (5.5 to 13 years old) who made written strokes on an electronic tablet. Two types of tests were used : a single stroke and a triangle drawing test. First, Sigma-Lognormal parameters were extracted from recorded movements and used as the input to our predictive model. Multiple different models were compared (linear regression, neural networks, K-nearest neighbors regression, random forest, gradient boosting regression, support vector regression). These different models performed well considering the within children variability in handwritten strokes and the between-children variability in motor control maturity. The top score is obtained with the neural network based models : coefficient of determination ($\mathbf{R^2}$) : 0.548 ; mean absolute error (MAE) : 0.937. Linear regression models also performed well ($\mathbf{R^2}$: 0.476 MAE : 1.032). More than 90% of the predictions have a relative error of less than 20%. We found that using simple stroke parameters alone is not optimal ; the results were better when using parameters from the triangle test or a mix of features from both tests. In conclusion, our study shows that the Sigma-Lognormal model proposes a new possibility to estimate the motor control maturity. This gives us a completely new approach to complement the traditional methods currently used. This method is fast and comfortable for the user. Since it only requires for children to make handwriting strokes on an electronic tablet with a pen, it is expected to be more convenient for doctors to use this test in a clinical context for pre-screening with less professional intervention compared to traditional methods with questionnaires, it is a quick and cheap way.

TABLE DES MATIÈRES

DÉDICACE iii

REMERCIEMENTS iv

RÉSUMÉ v

ABSTRACT vi

TABLE DES MATIÈRES vii

LISTE DES TABLEAUX ix

LISTE DES FIGURES x

LISTE DES SIGLES ET ABRÉVIATIONS xi

CHAPITRE 1 INTRODUCTION 1

CHAPITRE 2 REVUE DE LITTÉRATURE 2

2.1 Test du contrôle neuromoteur 2

2.2 Autres applications de la technologie sur le contrôle moteur 3

2.3 La Théorie Cinématique des mouvements humains rapides 6

2.3.1 Les fondements théoriques 6

2.3.2 Le système d'acquisition des données 6

2.3.3 Les modèles et la reconstruction des mouvements 8

2.4 Les modèles et les algorithmes 13

2.4.1 Régression linéaire 13

2.4.2 Machine à vecteurs de support (MVS) 15

2.4.3 k plus proches voisins 17

2.4.4 Forêts aléatoires 18

2.4.5 Amplification du gradient 18

2.4.6 Réseau de neurones artificiels 18

CHAPITRE 3 CONTEXTE DU PROJET, HYPOTHESE ET OBJECTIFS SPECIFIQUES 22

3.1 Contexte du projet 22

3.2 Organisation du travail 22
3.3 Hypothèse de recherche 22
3.4 Objectif 22

CHAPITRE 4 ARTICLE 1 : COMPARING SYMBOLIC AND CONNECTIONIST ALGORITHMS FOR CORRELATING HEALTHY CHILDREN AGE WITH SIGMA-LOGNORMAL NEUROMOTOR PARAMETERS 24
4.1 Abstract 25
4.2 Introduction 25
4.3 Method and experiments 27
4.3.1 Participants 27
4.3.2 Sigma-Lognormal tests 27
4.3.3 Transformation of data 28
4.3.4 Experiments 30
4.3.5 Configuration and models to compare 31
4.4 Results 35
4.5 Discussion and future work 38
4.6 Conclusion 39
4.7 Appendix A 40
4.8 Appendix B 40
4.9 Appendix C 44

CHAPITRE 5 DISCUSSION GÉNÉRALE 49
5.1 L'interprétation des résultats 49
5.2 Les limitations 53
5.3 Recommandations 54

CHAPITRE 6 CONCLUSION 55
6.1 Conclusion 55

RÉFÉRENCES 56

LISTE DES TABLEAUX

Table 4.1 Table for simple stroke 36
Table 4.2 Table for triangular test 37
Table 4.3 Table for simple stroke with only 2 lognormal functions 38
Table 4.4 ablative table of auxiliary loss for the GRU based model 40
Table 4.5 fraction of Percentage absolute error of GRU models 41
Table 4.6 fraction of absolute error of GRU models 41
Table 4.7 fraction of Percentage absolute error of SVR models 42
Table 4.8 fraction of absolute error of SVR models 42
Table 4.9 fraction of Percentage absolute error of Ridge regression models . . . 43
Table 4.10 fraction of absolute error of Ridge regression models 43
Table 4.11 p-values of the T-test (simple test vs triangular test) 47
Table 4.12 p-values of the T-test with/without only 2 lognomral functions in simple tests 48
Table 4.13 p-values of the T-test (MLP vs GRU) 48
Tableau 5.1 Table of difference GRU vs SVR 50
Tableau 5.2 Table of difference GRU vs SVR 51
Tableau 5.3 Table of difference GRU vs SVR 52

LISTE DES FIGURES

Figure 2.1 Assessgame 3
Figure 2.2 InMotion2 et InMotion3 (https ://www.bioniklabs.com/) 5
Figure 2.3 REAplan déployé à la Clinique Valmont (https ://www.axinesis.com/en/our-solutions/reaplan/) 5
Figure 2.4 Les neuf tests du contrôle neuromoteur 7
Figure 2.5 La trajectoire du test de type triangulaire, trajectoire originale et reconstruite (virtual target points VTP en anglais) 8
Figure 2.6 Les compositions des fonctions lognormales d'un mouvement du test triangulaire 9
Figure 2.7 Une machine à vecteurs de support avec la perte de marge douce . . 15
Figure 2.8 le perceptron multicouche 19
Figure 2.9 les unités des réseaux récurrents 20
Figure 4.1 Simple stroke. 29
Figure 4.2 Triangular drawing test. 30
Figure 4.3 Nested-cross-validation 32
Figure 4.4 GRU based models: Here we show the model structure with T trials of 1 participants. The concatenation is done in the feature dimension, all tensor in this has a first dimension T until the last prediction. In the last step, we have a series of predicted results of length T. The average value is calculated based on the number of recorded trials, but not on the maximum number of trials after padding. 33
Figure 4.5 The real age vs. prediction value with quantities of [2.5, 10, 25, 50, 75, 90, 97.5] 44
Figure 4.6 mean trials with the simple test 45
Figure 4.7 full trials with the simple test 45
Figure 4.8 mean trials with the triangular test 46
Figure 4.9 full trials with the triangular test 46
Figure 5.1 Difference distribution GRU vs RIDGE 51
Figure 5.2 Difference distribution RIDGE vs SVR 52

LISTE DES SIGLES ET ABRÉVIATIONS

SNC	Système nerveux central
AVC	Accident vasculaire cérébral
MCO	Moindres carrés ordinaire
SVM	Support-vector Machine
SVR	Support-vector Regression
RBF	Base radiales gaussiennes
MLP	Multi-layer perceptron
FC	Fully-connected
ReLU	Rectified Linear Unit
ANN	Artificial neural network
NN	Neural network
RNN	Recurrent Neural Network
LSTM	Long Short-Term Memory
GRU	Gated Recurrent Unit
AE	Autoencoder
VAE	Variational autoencoder
sEMG	électromyographie
TPE	Tree-structured Parzen Estimator
ICC	Intraclass correlation coefficient

CHAPITRE 1 INTRODUCTION

Le contrôle neuromoteur est une capacité essentielle de l'être humain qui se développe progressivement au cours de l'enfance. Un bon contrôle neuromoteur permet aux enfants d'apprendre et de vivre mieux. Il est donc important de comprendre comment mieux mesurer la maturité du contrôle moteur et toute déviation par rapport au développement typique. Les performances motrices sont le plus souvent mesurées 1) par des professionnels administrant des tests qualifiés du contrôle moteur ou 2) par les parents, les enseignants ou les cliniciens remplissant des questionnaires basés sur l'observation du comportement [1]. Il existe également des tests qui évaluent le contrôle moteur en fonction des performances dans un jeu vidéo, mais ils nécessitent des instruments spécifiques [2], et un niveau d'expérience différent du jeu peut entraîner un biais dans les performances des enfants. Aucune de ces méthodes n'est particulièrement pratique en raison des ressources humaines ou matérielles supplémentaires requises.

L'écriture manuscrite est une tâche typique de contrôle de la motricité fine de l'humain. La Théorie Cinématique des mouvements humains rapides [3–6] et son modèle Sigma-Lognormal [7] ont été utilisés dans plusieurs applications biomédicales pour évaluer le motricité fine associé à l'écriture manuscrite. Par exemple, elle a été utilisée pour analyser les performances graphomotrices d'enfants de maternelle [8], pour évaluer le risque d'accident vasculaire cérébral [9] et pour évaluer la présence d'un trouble de déficit de l'attention avec ou sans hyperactivité (TDAH) chez les enfants [10].

L'enjeu de cette étude est de développer un modèle qui est capable de prédire l'âge du participant à partir des paramètres Sigma-Lognormaux, pour aider les cliniciens à évaluer la maturité du contrôle moteur chez les enfants normaux. De différents moyens d'utiliser ces paramètres sont comparés avec différents modèles.

La suite de ce mémoire est divisée en 5 autres chapitres. Le chapitre 2 présente dans le cadre d'une revue de littérature de certaines notions nécessaires pour comprendre étude. Les chapitres 3 et 4 rapportent les détails du projet, dont le chapitre 4 est un article scientifique récemment accepté à la prestigieuse conférence « International Conférence on Pattern Recognition,(ICPR 2022) qui se tiendra à Montréal en Août prochain. Le chapitre 5 est un complément du chapitre 4. Le chapitre 6 conclue cette étude et propose des recommandations pour la prochaine étude.

CHAPITRE 2 REVUE DE LITTÉRATURE

Cette revue de littérature est composée de trois parties. La première présente certains tests du contrôle neuromoteur. La deuxième présente la Théorie Cinématique des mouvements humains rapides et la troisième présente les modèles concernés dans l'article.

2.1 Test du contrôle neuromoteur

Le contrôle neuromoteur mesure la capacité de contrôler le corps et les membres pour faire un mouvement. C'est une capacité essentielle de l'être humain qui se développe progressivement au cours de l'enfance. Un mouvement est initié par l'envoi de signaux du cerveau. Ensuite, ce signal active le système musculaire pour commencer le mouvement. Ce mouvement renvois aussi des informations proprioceptives au système nerveux central (SNC), ces signaux proprioceptifs nous aident à modifier/corriger/continuer le mouvement actuel. Il est donc important de comprendre comment nous pourrions mieux mesurer au mieux la maturité du contrôle neuromoteur. Pour l'instant, les tests de la performance du contrôle neuromoteur sont basés sur les "capacités motrices", les "aptitudes motrices" ou les “compétences motrices". Les tests de capacités motrices focalisent sur la performance dans des éléments de test individuel qui sont indépendants de la tâche et du contexte. Les capacités motrices représentent la performance physique. Il s'agit de la force, la vitesse, l'endurance, la coordination, la flexibilité, etc. Les tests d'aptitudes motrices focalisent sur la qualité du mouvement. Il s'agit de la précision de la motricité fine, la dextérité, le contrôle des objets (par exemple en lançant une balle), etc. Les tests de compétences motrices évaluent la maîtrise des aptitudes motrices dans des situations spécifiques. Ils se concentrent sur la maîtrise réussie d'aptitudes motrices pour résoudre une situation problématique prédéfinie [11]. Ces tests sont faits souvent 1) par des professionnels qualifiés qui administrent des tests du contrôle neuromoteur ou 2) par les parents, les enseignants ou les cliniciens qui remplissent des questionnaires basés sur l'observation du comportement [1]. De même, pour les méthodes d'évaluation traditionnelles, en particulier les évaluations par questionnaire, l'influence de l'aspect culturel n'est pas négligeable, de sorte que l'adaptation culturelle (par exemple, la traduction) peut accroître la difficulté d'utilisation pour les enfants de différents pays [1]. Cependant, notre approche est moins affectée par les différences culturelles, elle peut donc être appliquée plus facilement dans différents pays. L'utilisation d'un stylo pour tracer des lignes étant moins différente d'un point de vue culturel, il s'agit d'un moyen plus naturel de montrer la capacité de contrôle de la motricité fine de l'enfant.

Il y a aussi des tests qui requièrent moins d'intervention humaine pour éviter d'introduire des biais. Par exemple, Annina Fahr et al [2] proposent une méthode pour évaluer le contrôle neuromoteur sélectif volontaire avec un jeu de vidéo nommé Assessgame (aussi nommé"Catch the Stars").

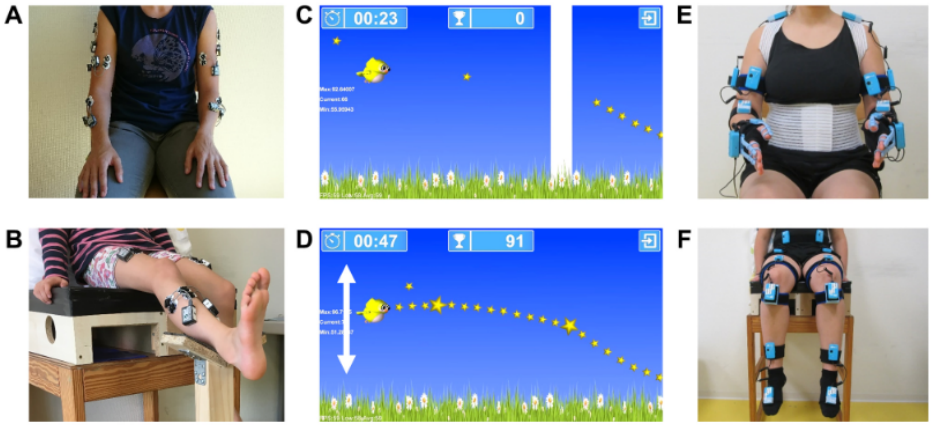

FIGURE 2.1 Configuration des mesures. A et B : Configuration pour la mesure des sEMG C : Capture d'écran du jeu d'évaluation montrant la fin de la phase d'accommodation sur le côté gauche, la ligne de départ verticale et le début de la trajectoire cible composée d'étoiles sur le côté droit. D : Deuxième capture d'écran de l'Assessgame, montrant l'avatar suivant et collectant les étoiles sur la trajectoire cible en étant déplacé de haut en bas par un mouvement articulaire isolé. E et F : Placement et fixation des modules de capteurs accéléromètres à l'aide de sangles flexibles et de gants pour les extrémités supérieures (E) et inférieures (F). Une paire de capteurs était toujours fixée de manière proximale et distale à chaque articulation cible. [2]

La corrélation entre les scores du Assessgame et les âges de 6 ans à 18 ans est de $r > 0.8$. Aucune corrélation avec l'âge n'a été trouvée pour l'évaluation du contrôle neuromoteur sélectif volontaire basée sur l'électromyographie (sEMG) ($r < 0.25$). L'inconvénient du Assessgame est qu'il nécessite un équipement spécial. Les différents niveaux d'expérience des participants peuvent entraîner aussi un biais.

2.2 Autres applications de la technologie sur le contrôle moteur

L'étude sur la détection de TDAH [10] présente que les enfants avec TDAH ont la pire qualité de l'écriture ainsi que le contrôle moteur fin que les enfants normaux. Dans ce cas, le contrôle moteur est étudié, basé sur le mouvement humain, spécialement, l'écriture. Il existe différentes théories et approches pour étudier la production du mouvement humain. Par exemple, les approches basées sur les réseaux de neurones [12], les modèles comportementaux

[13] et les modèles cinématiques [3–6]. Parmi ces modèles, la Théorie Cinématique rapide des mouvements humains est utilisée dans les études classiques sur le contrôle moteur. À part de l'analyse d'écriture, il est possible d'analyser les autres mouvements des participants. La réalité virtuelle est appliquée déjà dans la favorisation du développement des enfants d'âge scolaire ayant un trouble d'apprentissage (https ://aleovr.com/). L'étude [14] nous montre l'application de la technologie de la réalité virtuelle dans l'étude du contrôle de la balance du corps humain. Cela nous inspire que l'on puisse éventuellement aussi utiliser la réalité virtuelle pour enrichir les visions pour les participants. Au lieu d'immobiliser le participant, comme dans le cas de l'Assessgame, nous pouvons également le maintenir en mouvement et obtenir ainsi davantage de données des mouvements.

Le robot est aussi une technologie importante utilisée dans plusieurs domaines biomédicaux comme la réhabilitation et l'analyse du contrôle moteur. L'utilisation du robot peut apporter plusieurs bénéfices : 1. Les machines peuvent prendre en charge la plupart des tâches physiquement exigeantes, laissant aux opérateurs de santé la possibilité de se concentrer plus sur les tâches plus importantes. 2. Les robots sont capables d'aider les patients avec faibles capacités motrices de finir des tâches avec une résistance adaptative. 3. Les robots peuvent proposer des résultats quantitatifs et précis de la performance du patient pendant les tâches motrices. Dans l'étude [15], on a utilisé un robot appelé InMotion2 robot (Interactive Motion Technologies, Inc) pour mesurer la performance du contrôle au niveau de l'épaule et du coude des patients atteints d'un accident vasculaire cérébral chronique. Les participants ont besoin de faire des mouvements point à point du centre vers huit cibles sortantes réparties le long d'un cercle à une distance de 0,14 m avec un temps maximal de 3s. Certains critères sont calculés à partir des mesures basées sur le robot, comme l'évaluation de Fugl-Meyer, le score d'état moteur, la puissance motrice et l'échelle d'Ashworth modifiée, etc. La meilleure performance obtenue avec 8 critères cinématiques est R=0.71 pour l'apprentissage et R=0.72 pour la validation avec 111 sujets. À partir de cette étude, on rajoute deux améliorations : 1. La régulation de force est rajoutée dans le robot. 2. Un Système sensoriel du capteur magnéto-inertiel est rajouté pour mesurer l'accélération en tant que données multimodale [16]. Quelques d'autres atouts d'utilisation du robot dans l'évaluation médicale sont : 1. Les données biomécaniques acquises des participants peuvent être utilisées pour mettre à jour le système pour mieux l'adapter aux patients. 2. Le robot peut donner une action corrective en cas de mouvement incorrect. 3. Les mesures précises de la fonction du patient permettent aux cliniciens de surveiller l'état du patient pendant la thérapie. Une autre partie intéressante dans cette étude, c'est la décomposition du profil de vitesse du robot avec les sous-mouvements. Les signaux de vitesse du robot ont été traités afin d'extraire les sous-mouvements du minimum vital composant le mouvement global du sujet. C'est une

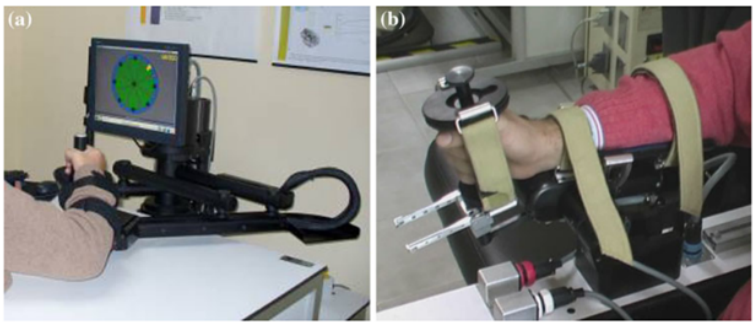

FIGURE 2.2 InMotion2 et InMotion3 (https ://www.bioniklabs.com/)

autre méthode différente que la Théorie Cinématique des mouvements humains rapides pour décomposer le profil de la vitesse. Les paramètres des sous-mouvements sont aussi utilisés dans l'analyse finale. Néanmoins, il existe aussi certaines limites, et les auteurs suggèrent les travaux complémentaires suivants : 1. Proposer plus d'interprétation clinique des paramètres. 2. Faire l'expérience avec plus de participants. 3. Faire des expériences concernant plus de membres du corps. Ces études confirment qu'il est possible d'utiliser des robots et les autres capteurs pour avoir des données multimodales pour une meilleure analyse du mouvement. À part de cette application du robot, REAPlan est aussi un exemple successif [17]. Cette étude fournit un protocole standardisé, valide, fiable pour quantifier les déficiences des membres supérieurs chez les patients victimes d'un AVC (un accident vasculaire cérébral), en utilisant un robot planaire. En plus, ce robot a déjà été déployé dans un centre de réhabilitation suisse.

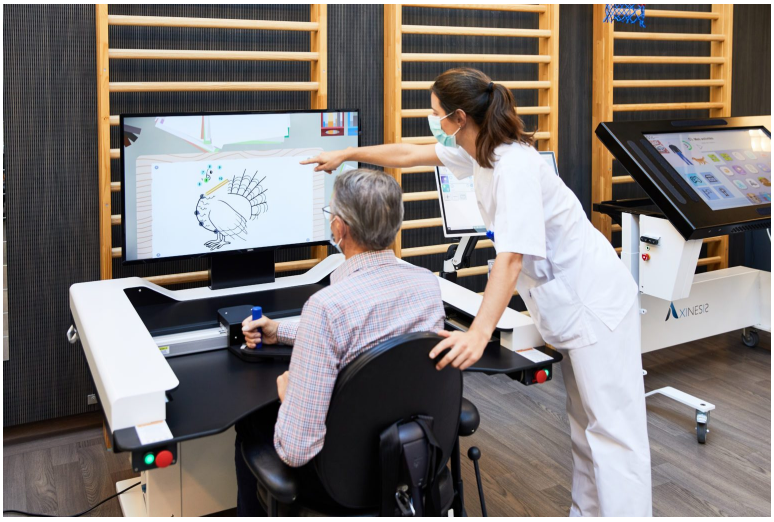

FIGURE 2.3 REAplan déployé à la Clinique Valmont (https ://www.axinesis.com/en/our-solutions/reaplan/)

2.3 La Théorie Cinématique des mouvements humains rapides

L'écriture manuscrite est une tâche typique de contrôle de la motricité fine pour l'humain. La Théorie Cinématique des mouvements humains rapides [3–6] et son modèle Sigma-Lognormal [7] ont été déjà utilisés dans plusieurs applications biomédicales pour évaluer la motricité fine en associant à l'écriture manuscrite. Par exemple, il a été utilisé pour analyser les performances graphomotrices d'enfants de maternelle [8], pour évaluer le risque d'accident vasculaire cérébral [9], et pour évaluer la présence du trouble de déficit de l'attention/hyperactivité (TDAH) des enfants [10]. En comparaison avec les méthodes brièvement présentées dans la partie précédente, l'analyse de l'écriture manuscrite et de la maturité du contrôle neuromoteur avec la Théorie Cinématique sont moins biaisées par l'intervention humaine et plus pratique. Parce qu'elle ne nécessite qu'une tablette à numériser et un stylo (section 2.3.2).

2.3.1 Les fondements théoriques

La Théorie Cinématique modélise le système neuromusculaire comme un effecteur terminal pendant la production d'écriture. Cette théorie a été mise en place d'abord sur les gestes simples dans les années 90 [3, 4], comme un trait droit simple, ensuite elle a été étendue à des mouvements plus complexes comme la signature dans les années suivantes [18]. Elle suppose que le système neuromusculaire humain est composé de multiples sous-systèmes linéaires. Pour faire un trait, le système nerveux central (SNC) envoie une commande initiale qui active ensuite les sous-systèmes graduellement. Le couplage entre ces sous-systèmes est représenté par une relation proportionnelle des temps de délais cumulatifs [19]. En effet, la réponse impulsionnelle de chaque sous-système après avoir été activée peut être considérée comme le produit de la convolution de tous les sous-systèmes précédents. Si le nombre de ces sous-systèmes tend vers l'infini, la théorie prédit par une démonstration mathématique rigoureuse [6, 20] que la vitesse absolue d'un trait tend vers une fonction lognormale par rapport au temps [6]. Cela correspond au profil de vitesse en forme de cloche asymétrique observé dans plusieurs expériences [21].

2.3.2 Le système d'acquisition des données

Le système d'acquisition qui a été utilisé dans cette études et dont nous analysons les données est basé sur une tablette Wacom Cintiq 13HD dont la résolution de son écran est 200 lignes par millimètre et sa fréquence d'échantillonnage est 200 Hz [23]. Le participant est invité à dessiner plusieurs traits avec un stylet en suivant un guide. Les profils de vitesse sont d'abord enregistrés dans le système puis ils sont utilisés pour extraire les paramètres lognormaux avec

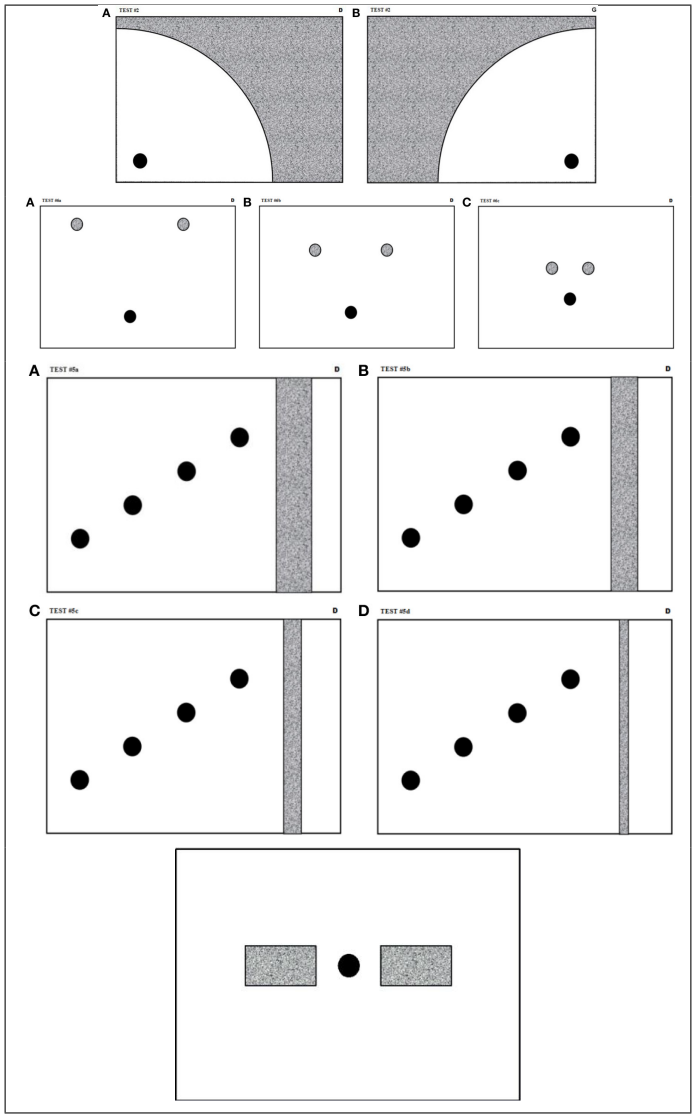

FIGURE 2.4 Les neuf tests du contrôle neuromoteur(les figures sont de l'article [22])

le logiciel ScriptStudio développé par le laboratoire Scribens [7]. Des essais d'apprentissage sont proposés pour que le sujet testé soit familiarisé avec les tests avant de procéder aux tests officiels. Le signal de départ peut être sonore de 500ms/1kHz ou un clignotement de rouge à vert sur l'écran [24]. Des tests différents ont été proposés pour analyser le contrôle neuromoteur [22]. Dans notre recherche, nous avons utilisé principalement le test de trait simple et le test du triangle. Généralement, tous les tests demandent au participant de dessiner une trace entre la zone de départ (cercle noir) et la zone finale (zone grise). Voici une liste de ces tests :

— *Trait simple* : ce test examine la capacité du participant à réagir à un stimulus auditif ou visuel.

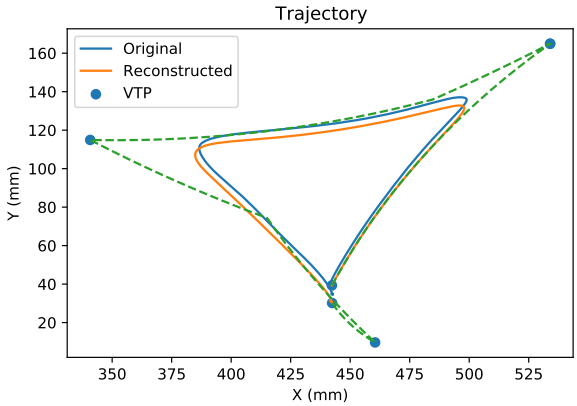

FIGURE 2.5 La trajectoire du test de type triangulaire, la trajectoire originale est marquée en bleu, la trajectoire reconstruite est marquée en orange, les cibles virtuelles (VTP virtual target points en anglais) marquée en bleu sont définies comme les positions ciblées du plan d'écriture, cibles qui ne sont généralement pas atteintes à cause du chevauchement temporelaire de la prochaine fonction lognormale [7].

— *Tracé triangulaire* : Ce test examine la capacité de planifier des mouvements complexes à partir des mouvements simples.
— *Compromis vitesse/précision* : Ce test évalue la capacité de réagir à des contraintes temporelles et spatiales en même temps.
— *Oscilations* :1. oscillations à vitesse maximale : Ce test est effectué pour évaluer si le participant est capable de faire des oscillations à fréquence maximale. Comme seule la vitesse est importante dans ce test, les zones destinées sont grandes. 2. Les oscillations synchronisées évaluent la capacité de suivre à un rythme donné.

Le participant peut choisir sa main préférée pour effectuer les tests dans des conditions confortables. Donc la direction du mouvement dépend de la main utilisée du participant.

2.3.3 Les modèles et la reconstruction des mouvements

Selon la Théorie Cinématique, le modèle Sigma-Lognormal a été développé pour décrire le mouvement d'écriture complexe. Un trait d'écriture peut être décrit par d'un plan d'action constitué de cibles virtuelles connectées par des arc de cercles (virtual target points VTP en anglais). Ces points sont définis comme les positions ciblées du plan d'écriture, cibles qui ne sont généralement pas atteintes à cause du chevauchement temporaire de la prochaine fonction lognormale [7]. La figure 2.5 montre la trajectoire d'un mouvement du test triangulaire avec les cibles virtuelles clairement indiquées par des points verts. Chaque arc est décrit par une fonction lognormale. Le mouvement complet de ce trait est donc représenté par

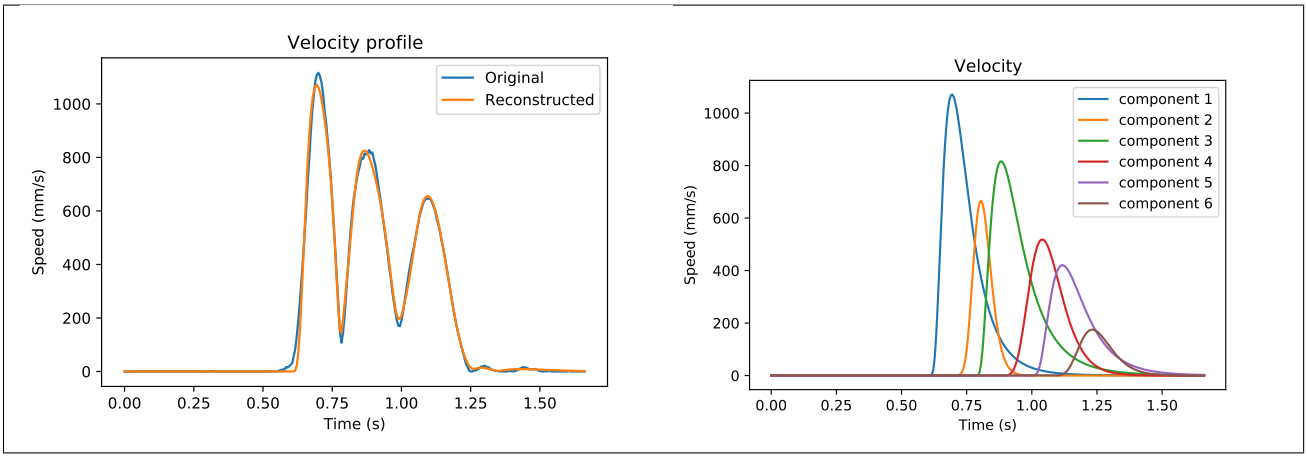

FIGURE 2.6 Les compositions des fonctions lognormales d'un mouvement du test triangulaire

la somme vectorielle de ces fonctions lognormales. Le trait triangulaire peut être considéré comme constitué de trois traits simples. Si un trait simple est représenté par deux fonctions lognormales, un trait triangulaire est donc représenté par six fonctions lognormales. Les figures 2.6 montrent le profil du module de la vitesse et ses six composants lognormaux. Mais le nombre de fonctions lognormales n'est pas nécessairement fixe. Le choix du nombre de fonctions lognormales dépend du contexte du problème étudié et de son application. Dans notre cas, nous voulons étudier la corrélation entre l'âge des enfants et les traits manuscrits, le paramètre SNR (le ratio signal/bruit) est donc un indice important pour différencier leurs performances. En plus, les formes des traits sont relativement simples dans nos tests (triangle, ligne droite, etc.) en comparaison avec l'écriture. Au contraire, si le sujet de recherche est d'étudier la génération d'écritures synthétiques pour enrichir la base de données [25] ou de savoir comment distinguer les signatures produites par l'homme ou par une machine [26, 27], le nombre de fonctions lognormales n'est plus envisagé autant que notre cas, le SNR doit être le plus élevé possible. De plus, les signatures sont généralement complexes et varient fortement d'un individu à l'autre, il est impossible d'indiquer un nombre de fonctions lognormales raisonnable.

Pour chaque fonction lognormale, un ensemble de 6 paramètres est extrait :

$$\{t_0,\ D,\ \mu,\ \sigma,\ \theta_{start},\ \theta_{end}\}_i^{nbLog}$$

— t_0 : Le moment (en secondes) de l'émission de la commande motrice. Dans le contexte d'un test psychomoteur, c'est le moment où le système nerveux commence à réagir après avoir reçu un signal de départ (par exemple, un son ou un stimulus visuel). Ce

temps est nécessaire pour recruter suffisamment de cellules musculaires afin d'initier le mouvement. Pour le modèle Sigma-Lognormal, chaque fonction lognormale a un t_0, décrivant l'instant où chaque commande a été envoyée pour coordonner les sous-trajectoires constituantes.

— D : L'amplitude (en millimètres) de la fonction de vitesse lognormale. Elle correspond à la distance totale de la trajectoire associée à ce mouvement primitif.

— μ : Le temps de délai sur une échelle logarithmique (en $\ln(s)$) pour atteindre la médiane du mouvement. Ce paramètre reflète la vitesse générale de la réaction.

— σ : Le temps de réponse sur une échelle logarithmique en $\ln(s)$ qui est utilisé pour estimer la durée du mouvement.

— $\theta_{start}, \theta_{end}$: Les angles de début et de fin du mouvement en radians.

Le profil mathématique de vitesse s'exprime comme [18] :

$$\begin{aligned} \overrightarrow{v}(t) &= \sum_{i=1}^{nbLog} \overrightarrow{v}_i(t; t_{0i}, \mu_i, \sigma_i^2) \\ &= \sum_{i=1}^{nbLog} D_i \begin{bmatrix} cos(\theta_i(t)) \\ sin(\theta_i(t)) \end{bmatrix} \Lambda_i(t; t_{0i}, \mu_i, \sigma_i^2); N \geq 2 \end{aligned} \tag{2.1}$$

$$\Lambda(t; t_0, \mu_2, \sigma_2^2) = \frac{1}{\sigma_i \sqrt{2\pi}(t - t_{0i})} \exp\left(\frac{-[\ln(t - t_0) - \mu_i]^2}{2\sigma_i^2} \right) \tag{2.2}$$

$$\theta_i(t) = \theta_{start} + \frac{\theta_{end} - \theta_{start}}{2} \left[1 + erf(\frac{\ln(t - t_{0i}) - \mu_i}{\sigma_i \sqrt{2}}) \right] \tag{2.3}$$

Quand le trait dessiné est juste une ligne droite, l'équation 2.1 se réduit à l'équation Delta Lognormale [18] :

$$v(t) = D_1 \Lambda(t; t_0, \mu_1, \sigma_1^2) - D_2 \Lambda(t; t_0, \mu_2, \sigma_2^2) \tag{2.4}$$

Dans cette situation, le terme 1 et le terme 2 représentent respectivement la fonction agoniste et la fonction antagoniste. Ce modèle Delta-Lognormal est utilisé seulement pour les mouvements simples comme une ligne droite. La fonction agoniste représente l'effet des groupes neuromusculaires permettant d'initier le mouvement, et la fonction antagoniste représente les groupes neuromusculaires qui freinent le mouvement. Finalement le mouvement résultant est constitué par la synergie de ces deux groupes neuromusculaires.

Par ailleurs, 3 paramètres lognormaux globaux sont utilisés pour évaluer l'état de reconstruction globale du mouvement : SNR, $nbLog$, and $\frac{SNR}{nbLog}$. Ces paramètres sont définis comme :

— SNR : Le rapport signal/bruit, qui mesure la qualité de la reconstruction du profil de vitesse par rapport au profil de vitesse original du mouvement. Une valeur SNR plus

élevée indique une meilleure reconstruction du profil de vitesse.

$$SNR = 10 \log \left(\frac{\int_{t_s}^{t_e} v_o^2(t) dt}{\int_{t_s}^{t_e} |v_o(t) - v_r(t)|^2 dt} \right) \tag{2.5}$$

v_o représente le profil de vitesse original et v_r représente le profil de vitesse reconstruit. On considère que la reconstruction est bien faite quand le SNR est supérieur à 15dB.

— *nbLog* : Le nombre de fonctions lognormales utilisées pour reconstruire le profil de vitesse. Il indique la régularité du mouvement. Plus cette valeur est faible, plus le mouvement est lisse. Dans le cas idéal, la valeur minimale de *nbLog* est de 2 pour le test du trait simple et de 6 pour le test du triangle, car un mouvement triangulaire est composé de 3 traits simples.

— $SNR/nbLog$: Le rapport entre SNR et *nbLog*. Il reflète la fluidité du mouvement.

En plus des paramètres introduits précédemment, d'autres quantités sont calculées pour aider à interpréter les caractéristiques du profil de vitesse [6]. Les indices i associés à chaque lognormale ont été enlevés ici car il n'influence pas la compréhension.

— *Mode* : M, le temps où la réponse impulsionnelle lognormale atteint la valeur maximale. Plus cette valeur est élevée, plus long est le temps pour l'atteindre.

$$\left. \frac{\mathrm{d}\Lambda(t; t_0, \mu, \sigma^2)}{\mathrm{d}t} \right|_{t=M} = 0 \tag{2.6}$$

$$M = t_0 + \exp\left(\mu - \sigma^2\right) \tag{2.7}$$

En ce moment, la valeur maximale de la réponse est :

$$\Lambda_{\max} = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\mu + 0.5\sigma^2\right) \tag{2.8}$$

— *Médiane* : m, le temps où la moitié du mouvement est parcouru. L'augmentation de cette quantité signifie que l'exécution du mouvement est plus lente. C'est équivalent au moment où l'intégrale de la réponse impulsionnelle atteint la moitié de sa valeur.

$$\int_{t0}^{m} \Lambda(t; t_0, \mu, \sigma^2)\, dt = \frac{1}{2} \tag{2.9}$$

$$m = t_0 + \exp(\mu) \tag{2.10}$$

— *Temps de délai* : C'est le premier moment d'une réponse impulsionnelle lognormale, il peut s'interpréter comme l'évaluation globale du délai de la réponse à la suite d'une commande. L'augmentation de cette quantité indique la réaction plus lente à la com-

mande.

$$\bar{t} = \int_{t0}^{+\infty} t\Lambda(t; t_0, \mu, \sigma^2)\, dt = t_0 + \exp\Big(\mu + 0.5\sigma^2\Big) \tag{2.11}$$

— *Temps de réponse* : Le temps de réponse de la réponse impulsionnelle du système neuromusculaire. s^2 peut également être interprété comme une mesure de l'étalement de cette réponse dans le temps. L'augmentation de cette quantité signifie que le temps pour exécuter le mouvement est plus long.

$$\begin{aligned} S &= \sqrt{\int_{t0}^{+\infty} (t - \bar{t})^2 \Lambda(t; t_0, \mu, \sigma^2)\, dt} \\ &= \sqrt{e^{2\mu+\sigma^2}(e^{\sigma^2} - 1)} \\ &= (\bar{t} - t_0)\sqrt{(e^{\sigma^2} - 1)} \end{aligned} \tag{2.12}$$

— *Asymétrie* : Cette quantité correspond à la forme de la courbe lognormale. Plus grande est l'asymétrie, plus le mode est déplacé vers la gauche, avec un pic de vitesse plus petit et un temps de mouvement global plus long.

$$\begin{aligned} A_c &= \frac{m - M}{m - t_0} \\ &= 1 - \exp\Big(-\sigma^2\Big) \end{aligned} \tag{2.13}$$

Finalement des collègues ont proposé deux autres quantités qui ne viennent pas directement de la théorie, mais elles sont calculées et utilisées dans les articles [28, 29].

— *Temps de réaction* : T_r C'est le temps où le système musculaire initie le mouvement après avoir reçu la commande motrice. (Défini ici comme le moment où la vitesse atteint 10% de la vitesse maximale.)

— *Temps de conduction* : C'est l'écart entre le temps de réaction et t_0. Il s'agit du temps pour propager la commande du cerveau au système musculaire d'exécution.

$$T_c = T_r - t_0 \tag{2.14}$$

Ces paramètres ne sont pas utilisés dans la prédiction finale dans notre recherche. Ceci est dû au fait que nous n'avons pas fait d'étude approfondie sur l'ingénierie des caractéristiques et la sélection des paramètres. En rajoutant directement ces paramètres dans nos entrées comme des paramètres supplémentaires, comparant les modèles sans ces deux paramètres, la performance de prédiction de nos modèles ne s'est pas améliorée mais l'a même rendue pire. Il existe peut-être d'autres paramètres ou des combinaisons de paramètres à concevoir mais

ce sera un sujet à étudier dans le futur. Mais déjà avec les paramètres utilisés actuellement, nous avons des résultats très satisfaisants.

2.4 Les modèles et les algorithmes

Comme nous l'avons dit précédemment, le participant a fait des tests du contrôle neuromoteur différents en fonction du besoin sur la tablette, puis les données du profil de vitesse sont enregistrées et utilisées pour extraire les paramètres lognormaux. C'est ici que notre travail véritable a commencé, à savoir : comment utiliser ces paramètres lognormaux pour prédire la maturité du contrôle neuromoteur. C'est une question de régression, car nous cherchons un modèle qui est capable de prédire l'âge du participant à partir des paramètres lognormaux qui lui sont corrélés. Nous avons comparé plusieurs modèles et méthodes différentes dans ce travail et nous les présentons dans les pages qui suivent.

2.4.1 Régression linéaire

La régression linéaire est le modèle de régression le plus utilisé dans plusieurs domaines. Il suppose que la relation entre les entrées et les sorties est linéaire avec w un vecteur des coefficients de proportionnalité. $\hat{y}_i$ réfère la valeur de prédiction en donnant un vecteur d'entrée x_i pour l'individu i.

$$\hat{y}_i = w * x_i \tag{2.15}$$

Grâce à la simplicité de ce modèle, il est facile d'interpréter et d'analyser les résultats de prédictions. L'influence de chaque paramètre est évidente en observant directement son coefficient pondéré. Ensuite, il faut minimiser la fonction d'erreur afin d'obtenir les coefficients les plus appropriés avec les données d'apprentissage. Différentes fonctions d'erreur peuvent répondre à nos besoins :

— *La méthode des moindres carrés ordinaire (MCO)* : C'est la fonction d'erreur la plus populaire dont la somme des carrés des différences entre les valeurs ciblées (valeurs observées dans la vraie vie) et celles prédites avec les paramètres d'entrée correspondante.

$$L = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 \tag{2.16}$$

Si le nombre de données est limité, il est possible que certains paramètres aient des coefficients inappropriés en respectant la contrainte de la minimisation de la fonction d'erreur. Ce phénomène s'appelle le surapprentissage (overfitting en anglais). Une solution est la régularisation qui consiste à ajouter un terme de pénalité à la fonction

d'erreur 2.16 afin d'empêcher les coefficients d'atteindre de grandes valeurs. Il existe plusieurs options de terme de pénalité.

— *La régression de crête* : Cette méthode de crête s'appelle aussi $L2$ régularisation [30](chapitre 7). Elle rajoute à la somme des carrés des coefficients dans la fonction d'erreur un hyperparamètres λ pour contrôler la pondération relative des pénalités.

$$L = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \|w\|_2^2 \tag{2.17}$$

$$\|w\|_2 = \sqrt{\sum_{i=1}^{M} w_i^2} \tag{2.18}$$

— *La régression Lasso* : Au lieu d'utiliser la somme des carrés des coefficients comme la pénalité, Lasso utilise la somme des valeurs absolues des coefficients [30] (chapitre 7). Quand la pondération *lambda* est suffisamment grande, certains coefficients sont amenés à zéro, ce qui conduit à un le modèle clairsemé en sélectionnant les coefficients les plus contributifs.

$$L = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 + \lambda \|w\|_1 \tag{2.19}$$

$$\|w\|_1 = \sum_{i=1}^{M} |w_i| \tag{2.20}$$

La régularisation $L0$ a le même effet théoriquement, puisque dans ce cas la pénalité est simplement égale au nombre total des coefficients non nuls.

$$\|w\|_0 = \sum_{i=1}^{M} 1 \quad \{w_i \neq 0\} \tag{2.21}$$

Malheureusement, ce terme n'est pas différentiable et la complexité est factorielle du nombre total des coefficients, c'est-à-dire qu'il faut essayer toutes les combinaisons des coefficients (M! possibilités au total si le nombre des coefficients est M). C'est pourquoi cette pénalité est moins utilisée.

— *La régression Huber* : La régression de Huber utilise une fonction d'erreur mixite [31], elle optimise la fonction d'erreur au carrée lorsque l'exemple fait une erreur dont la valeur absolue est inférieure à une valeur C, sinon elle optimise la fonction d'erreur absolue. Cette valeur de C est contrôlée par un hyperparamètre ϵ multiplié par l'écart type σ des erreurs de toutes les données d'apprentissage. L'avantage de cette fonction de perte mixte est que l'optimisation n'est pas très fortement affectée par les valeurs

aberrantes, mais leurs effets ne sont pas non plus complètement ignorés.

$$L_\epsilon(y, \hat{y}) = \begin{cases} (y - \hat{y})^2/2 & |y - \hat{y}| \leq C \\ C \cdot |y - \hat{y}| - C^2/2 & \text{sinon} \end{cases} \tag{2.22}$$

En conclusion, la régression linéaire est une bonne ligne de base. Sa performance est limitée par sa simplicité. Il est possible d'améliorer la capacité d'adaptation des modèles linéaires en créant de nouveaux paramètres d'entrée, mais c'est un travail coûteux.

2.4.2 Machine à vecteurs de support (MVS)

Ce modèle a été développé dans les années 90 (en anglais support-vector machine, SVM). Il repose sur deux critères : 1.la marge maximale et 2. la fonction de noyau. Ce modèle est capable de traiter les problèmes de classification/régression non linéaire [32].

La marge maximale

Dans le problème de classification, SVM essai de trouver une frontière qui sépare les exemples avec un hyperplan. La marge maximale est la distance entre cette frontière et les échantillons les plus proches. L'idée est de maximiser cette distance pour que la classification soit meilleure. Dans le problème de régression, une marge de tolérance (ϵ) est fixée, la fonction d'erreur à minimiser est comptée avec les exemples hors de cette marge de tolérance (figure 2.8).

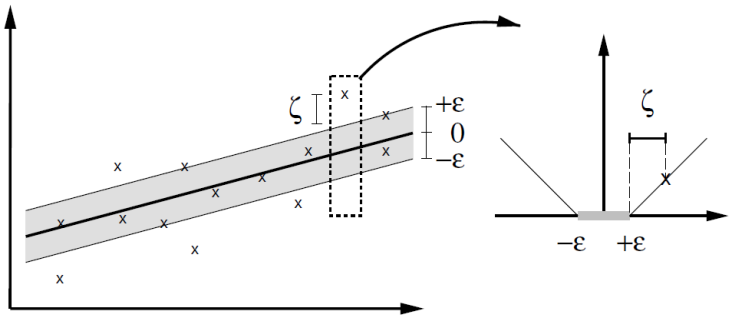

FIGURE 2.7 Une machine à vecteurs de support avec la perte de marge douce, figure de l'article "A Tutorial on Support Vector Regression" [32]

La fonction de noyau

La relation entre les paramètres d'entrées et la valeur ciblée n'est pas souvent une relation simple linéaire. Il est possible qu'il faille faire des calcules dans l'espace de grandes dimensions. Par exemple, si nous avons deux vecteurs $x, y \in R^3$ dans une espace d'entrée de trois dimensions, supposons que nous avons besoin de transposer x et y dans un espace à 9 dimensions. IL faut donc calculer d'abord les transformations avec la complexité totale de $O(N^2)$ si la dimension d'entrée est N :

$$\begin{aligned} \phi(x) = \phi(x_1, x_2, x_3) = (x_1^2, x_1x_2, x_1x_3, x_2x_1, x_2^2, x_2x_3, x_3x_1, x_3x_2, x_3^2)^T \\ \phi(y) = \phi(y_1, y_2, y_3) = (y_1^2, y_1y_2, y_1y_3, y_2y_1, y_2^2, y_2y_3, y_3y_1, y_3y_2, y_3^2)^T \end{aligned} \tag{2.23}$$

dans le cas où nous voulons calculer la multiplication :

$$\phi(x)^T\phi(y) = \sum_{i,j=1}^{3} x_ix_jy_iy_j \tag{2.24}$$

Si la dimension de paramètres d'entrées est grande, la calcule de transformation occuperait beaucoup du temps. Toutefois, si nous utilisons la fonction de noyau, désignée par $k(x, y) = (x^Ty)^2$, au lieu d'effectuer les calculs compliqués dans l'espace à 9 dimensions, nous obtenons le même résultat dans l'espace à 3 dimensions. La complexité de calcul est $O(N)$.

$$\begin{aligned} k(x, y) &= (x^Ty)^2 \\ &= (x_1y_1 + x_2y_2 + x_3y_3)^2 \\ &= \sum_{i,j=1}^{3} x_ix_jy_iy_j \end{aligned} \tag{2.25}$$

Il existe un vaste choix de fonctions de noyau, et nous présentons ici deux noyaux courants.

— *noyau polynomial* : Le degré de ce noyau est contrôlé facilement par l'hyperparamètre d. Quand le d est équal à 1, ce noyau est équivalent au linéaire noyau. Mais quand le degré est très élevé, la valeur $k(x, y)$ fluctue considérablement et le choix de bonnes valeurs pour les trois hyperparamètres est difficile.

$$k(x, y) = (a * x^\intercal y + c)^d \tag{2.26}$$

— *noyau de base radiales gaussiennes (RBF)* : Ce noyau peut s'écrire sous d'une forme de polynôme infinie, c'est pourquoi ce noyau a une très bonne capacité d'adaptation

d'une fonction. Il a seulement un hyperparamètre à choisir.

$$\begin{aligned} k(x,y,\gamma) &= e^{(-\gamma\|x-y\|^2)} \\ &= e^{(-\gamma x^2)}e^{(-\gamma y^2)}e^{(\gamma xy)} \\ &= e^{(-\gamma x^2)}e^{(-\gamma y^2)}\sum_{n}^{\infty}\frac{(\gamma xy)^n}{n!} \\ &= \sum_{n}^{\infty}[\sqrt{\frac{\gamma^n}{n!}}e^{(-\gamma x^2)}x^n] * [\sqrt{\frac{\gamma^n}{n!}}e^{(-\gamma y^2)}y^n] \end{aligned} \tag{2.27}$$

La fonction de noyau peut s'écrire comme :

$$k(x,y) = \phi(x)^{\intercal}\phi(y) \tag{2.28}$$

D'où, la fonction d'erreur à minimiser est :

$$\begin{aligned} L &= \frac{1}{2}\|w\|_2^2 + \sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \\ \hat{y}_i &= \sum_{j=1}^{M} a_j k(x_j, x_i) = \sum_{j=1}^{M} a_j \phi(x_j)\phi(x_i) = w^{\intercal}\phi(x_i) \end{aligned} \tag{2.29}$$

En conclusion, la fonction de noyau nous permet d'offrir un moyen plus efficace et moins coûteux de transformer les données en dimensions supérieures. Cela permet de traiter linéairement dans des dimensions supérieures des données qui ne pouvaient pas être traitées linéairement dans l'espace d'origine. Connaître la distribution ou la forme des données peut aider à choisir la fonction de noyau, voire concevoir une fonction de noyau sur mesure sans faire des centaines d'essais, mais les informations a priori sur la distribution des données sont souvent inconnues. En plus, quand la taille de l'échantillon est très grande, le calcul et le stockage limitent l'utilisation de ce type de modèle.

2.4.3 k plus proches voisins

C'est une méthode non paramétrique [33], qui d'abord chercher les k plus proches voisins de l'entrée dans l'espace des caractéristiques en fonction de la distance euclidienne (L2) ou la distance de Manhattan (L1), ensuite la valeur de prédiction est simplement la valeur moyenne des valeurs des k plus proches voisins. La taille de k affecte la performance du modèle. Une valeur trop faible entraînera un modèle sous-adapté, mais une valeur trop grande de k peut conduire au surapprentissage du modèle. Cette méthode est simple à implémenter et pas besoin du temps d'apprentissage. Mais le calcul est long pour les données nombreuses avec

des dimensions élevées. Ce modèle est vraiment sensible aux points aberrants, une petite valeur aberrante peut conduire à une erreur de prédiction.

2.4.4 Forêts aléatoires

Cet algorithme fait partie de technique de *bagging* qui est une méthode ensembliste [34]. L'idée de la méthode *bagging* est d'agréger des multiples sous-modèles comme un modèle. Chaque modèle "faible" a été entraîné avec les données échantillonnées à partir des données d'apprentissage complètes. Par conséquent, chaque sous-modèle est différent, l'agrégation de ces modèles donne une prédiction moyenne dont la variance est réduite. Si chaque sous-modèle a une erreur de variance $Var(x_i)$, l'erreur de variance totale de la sortie finale est donc $\frac{1}{Var(x_i)}$. Les forêts aléatoires utilisent souvent l'arbre de régression, néanmoins la méthode *bagging* ne se limite pas à l'utilisation de l'arbre de régression, d'autres modèles sont aussi possibles (régression linéaire, réseau de neurones etc).

2.4.5 Amplification du gradient

C'est aussi une méthode ensembliste. Contrairement à la méthode *bagging*, *boosting* met les sous-modèles en cascade [35]. Chaque sous-modèle (souvent l'arbre de régression) est un modèle "faible", l'ensemble de ces sous-modèles devient un modèle "fort".

$$\hat{y} = \sum_{i=1}^{M} \gamma_i h_i(x) + constant \tag{2.30}$$

Dans la phase d'apprentissage, on entraîne un sous-modèle dans une itération. Si $f_m(x)$ représente le métamodèle et $h_m(x)$ représente un sous-modèle.

$$f_m(x) = \begin{cases} f_{m-1}(x) + \rho_m * h_m(x) & m > 1 \\ f_0(x) = h_0(x) & m = 0 \end{cases} \tag{2.31}$$

2.4.6 Réseau de neurones artificiels

Le réseau de neurones artificiels (artificial neural network ANN en anglais) est inspiré du système nerveux biologique. Il y a trois caractères typiques pour cette méthode : 1. Il possède un ensemble de paramètres ajustables (paramètres numériques ajustés par l'algorithme d'apprentissage). 2. Il possède des connexions non linéaires. 3.Des critères d'apprentissages sont définis avec lesquels les paramètres sont ajustés par la rétropropagation du gradient [36]. Grâce à ces caractères, un réseau de neurones est capable de s'adapter à n'importe quelle fonction

complexe avec suffisamment de données. Nous pouvons facilement concevoir l'architecture et les critères du réseau pour l'adapter à notre problème. En raison du développement des technologies de calcul parallèle et d'énormes quantités de données, le réseau de neurones devient plus en plus populaire dans plusieurs domaines. Dans la suite, nous présentons quelques structures populaires.

Perceptron multicouche

Un perceptron multicouche (multi-layer perceptron MLP en anglais) est un réseau de neurones basique [30] (chapitre 6). Il se compose des couches entièrement connectées (fully-connected FC en anglais) avec la fonction d'activation non linéaire comme Relu [37].

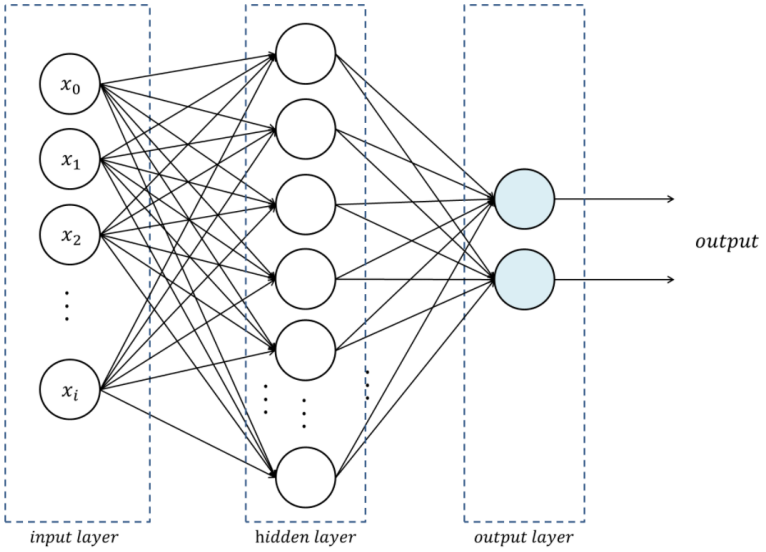

FIGURE 2.8 le perceptron multicouche de https ://www.cc.gatech.edu/ san37/post/ dlhc-fnn/

Cette structure est simple, elle est utilisée souvent comme une ligne de base pour comparer des réseaux de neurones plus avancés.

Réseau de neurones récurrents

Un réseau de neurones récurrents (Recurrent Neural Network RNN en anglais) est un type de ANN composé par des unités connectées sous la forme séquentielle.

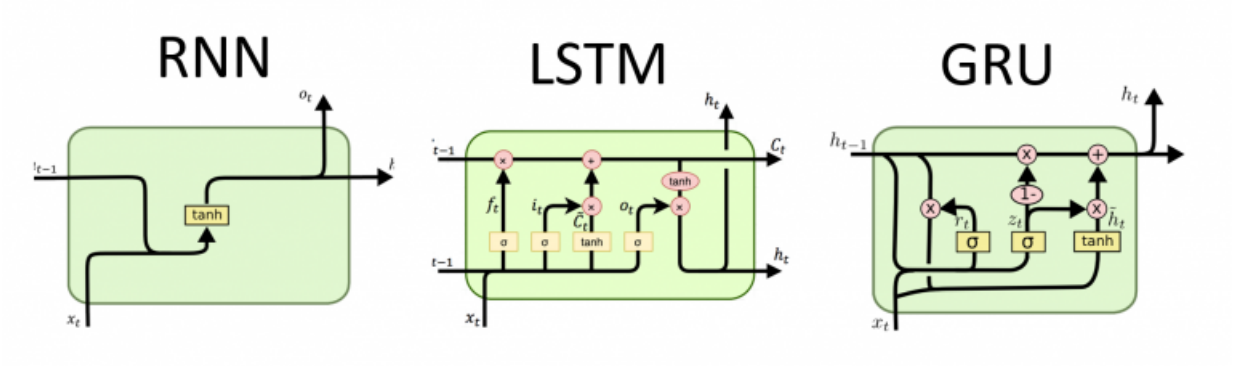

FIGURE 2.9 les unités des réseaux récurrents http ://dprogrammer.org/rnn-lstm-gru

Le nombre des unités dépend donc de la longueur du vecteur d'entrée. Certaines informations, appelées aussi la mémoire, sont cumulées dans l'état interne. À chaque étape temporelle, l'état interne est utilisé pour calculer la sortie et le nouvel état de cette étape avec les entrées actuelles. Mais cette architecture, elle aussi référé comme standard ou « vanille », a deux problèmes critiques, l'évanescence et l'explosion du gradient. L'évanescence fait référence au fait que les paramètres des couches supérieures (les étapes récentes) changent de manière significative alors que les paramètres des couches inférieures (les étapes anciennes) ne changent pas beaucoup ou pas du tout. Les informations des entrées anciennes ne contribuent donc d'aucune valeur dans ce cas. L'explosion du gradient fait référence à la croissance gigantesque des poids des paramètres, ceux-ci deviennent rapidement l'infini.

Pour atténuer ces problèmes, deux structures améliorées ont été inventées.

— *Réseau récurrent à mémoire court et long terme* appelé aussi Long Short-Term Memory (LSTM) en anglais [38]. En comparant avec le RNN vanille ou standard, le LSTM rajoute trois portes dans l'unité : 1.la porte d'entrée 2.la porte de sortie et 3.la porte d'oubli. L'idée est que non seulement un état interne se propage entre les unités, mais qu'il y a un nouveau paramètre C qui joue un rôle de mémoire à long terme. Cette mémoire C_t est la copie de l'ancienne C_{t-1} de l'unité précédente. Les informations les plus utiles des unités anciennes sont enregistrées dans ce C, et l'évanescence de gradient est améliorée.

— *Réseau récurrent à portes* Gated Recurrent Unit (GRU) en anglais [39]. Cette structure peut proposer une performance similaire que LSTM mais avec moins de paramètres. Il n'a qu'un seul état d'interne qui est associé avec la porte de réinitialisation et de mise à jour. D'abord, la porte de réinitialisation contrôle le calcul de l'état réinitialisé $\hat{h}_t$ avec h_{t-1} et l'entrée x_t. La porte de mise à jour s'occupe d'intégrer $\hat{h}_t$ dans h_{t-1} comme le nouveau l'état interne h_t en donnant une pondération Z entre 0 et 1.

Autoencodeur

Un auto-encodeur (AE) comporte généralement un encodeur et un décodeur [40]. L'encodeur est capable de transformer les entrées en une représentation de dimensions réduites dans l'espace latent(l'encodage). Le décodeur sert à reconstruire les entrées (une image, une séquence, etc.) à partir de cette représentation réduite tout en s'assurant qu'elle soit robuste au bruit de l'entrée. Une fois l'apprentissage complété, nous pouvons utiliser l'encodeur et le décodeur tout seul pour la réduction de dimensions ou la génération d'une nouvelle image. Une variation de cette architecture est l'auto-encodeur variationnel (Variational autoencoder VAE en anglais). La différence avec l'AE est que l'on fait des hypothèses concernant la distribution de l'espace latent. Au lieu d'avoir une représentation concrète, les représentations d'un VAE sont des distributions et lors du décodage, il faut échantillonner un exemple à partir de cette représentation distributionnelle. La distribution utilisée le plus souvent est la distribution gaussienne multivariée. Par ailleurs, les représentations distributionnelles du VAE sont aussi utilisées pour améliorer la performance de régression [41].

CHAPITRE 3 CONTEXTE DU PROJET, HYPOTHESE ET OBJECTIFS SPECIFIQUES

3.1 Contexte du projet

La revue de littérature a montré que les méthodes traditionnelles actuelles pour prédire la maturité du contrôle neuromoteur ne sont pas pratiques à mettre en œuvre et sont rarement précises en même temps. La Théorie Cinématique est capable d'extraire les paramètres corrélés fortement avec les performances du contrôle moteur via le modèle Sigma-Lognormal et il est théoriquement possible de prédire la maturité du contrôle moteur en fonction de l'âge du participant en faisant la régression avec ces paramètres, car la performance du contrôle moteur augmente progressivement au cours de l'enfance. Par ailleurs, le développement rapide du domaine de l'apprentissage machine nous donne accès à des modèles performants pour faire cette régression. La mise en place et la validation du modèle complet de prédiction sont aussi des parties les plus importantes de cette étude. Nous voulions étudier s'il est possible de prédire la maturité du contrôle moteur ou l'âge neurontypique d'enfant afin de faire les pré-pistages rapides.

3.2 Organisation du travail

L'acquisition des données a été faite par le laboratoire Scribens. L'extraction des paramètres Sigma-Lognormaux a été faite par le professeur Christian O'Reilly. La conception et la réalisation des expériences avec ces différents modèles ainsi que la rédaction de l'article sont été faites par Zigeng Zhang. L'article a été révisé par Christian O'Reilly et Réjean Plamondon.

3.3 Hypothèse de recherche

Les hypothèses que nous faisons sont : Les paramètres Sigma-Lognormaux sont corrélés avec la performance du contrôle moteur et cette performance est aussi corrélée bien avec l'âge.

3.4 Objectif

1. L'objectif principal de cette recherche est de développer un algorithme de prédiction de maturité du contrôle neuromoteur en utilisant les paramètres extraits par le modèle Sigma-Lognormal basé sur la Théorie Cinématique des mouvements humains rapides.

2. Nous voulons aussi comparer les différents modèles (symbolique et connexionniste) comme la régression linéaire et le réseau de neurones au niveau de la performance et de la simplicité d'utilisation. Dans cette étude, nous avons comparé les modèles : Ordinary least squares regression (OLS), Ridge regression (RR), Huber regression (HR), Support vector regression (SVR), XGBoost (XGB), Random forest (RF), K-nearest neighbors regression (KNN) et utiliser deux ensembles de données différents (test de tracés triangulaires et test de traits simples).

CHAPITRE 4 ARTICLE 1 : COMPARING SYMBOLIC AND CONNECTIONIST ALGORITHMS FOR CORRELATING HEALTHY CHILDREN AGE WITH SIGMA-LOGNORMAL NEUROMOTOR PARAMETERS

Ce chapitre se présente sous la forme d'un article écrit par Zigeng Zhang sous la direction de Christian O'Rielly et Réjean Plamondon. Cet article a été récemment accepté dans sa forme définitive pour une présentation à l'International Conference on Pattern Recognition 2022. Ce chapitre montre des concepts et des comparaisons des architectures et des résultats de prédictions de différents modèles. Il a pporte aussi plus d'informations sur les données utilisées dans cette recherche. La méthode est développée par Zigeng Zhang, sous la direction de Christian O'Reilly et Réjean Plamondon. La collecte des données a été préalablement effectuées dans le cadre d'une subvention de l'Institut TransMedTech obtenue par Réjean Plamondon. La rédaction de la version initiale de l'article a été faite par Zigeng Zhang, et ensuite révisée par Christian O'Reilly et Réjean Plamondon.

Dans la section 4.1, nous avons le résumé de l'article, il s'agit d'un survol du contenu de l'ensemble de l'article. La section 4.2 est l'introduction qui présente le contexte et la motivation de notre recherche. La section 4.3 concerne les détails de la méthode, il s'agit de l'utilisation des ensembles de données, la conception des expériences, les configurations de chaque modèle à comparer, etc. La section 4.4 montre tous les résultats ainsi que les analyses. La section 4.5 contient une discussion approfondie des questions du paragraphe précédente et des travaux connexes futurs. La section 4.6 résume l'ensemble de l'article.

Article publié dans International Conference on Pattern Recognition 2022, 24 janvier 2022.

Authors : Zigeng Zhang, Christian O'Reilly, Réjean Plamondon

Affiliations :

1. Laboratoire Scribens, Département de Génie Électique, École Polytechnique Montréal, Montréal, QC, Canada

2. Artificial Intelligence Institute of South Carolina, Department of Computer Science and Engineering, University of South Carolina, Columbia, SC, USA

Keywords : Sigma-Lognormal model, Kinematic Theory of rapid human movement, Handwriting, Regression, Fine motor control, Machine learning

4.1 Abstract

It is important to accurately evaluate the motor control maturity to help physicians diagnose delayed or abnormal motor development in children. Traditionally, it has been challenging to design assessment methods that are practical and accurate at the same time. This study aims to develop an effective algorithm to predict motor control maturity based on the Kinematic Theory of rapid human movements. We used handwritten pen strokes made on an electronic tablet by 513 children (5.5 to 13 years of age). We considered two types of movements : a single stroke and a triangle drawing test. For the analysis, Sigma-Lognormal parameters were extracted from recordings and used in predictive models. We compared multiple models, including linear regression, deep learning, K-nearest neighbors regression, random forest, gradient boosting regression, and support vector regression. These models performed well considering the within-children variability in handwritten strokes and the between-children variability in motor control maturity. The best score was obtained using the neural network model : coefficient of determination ($\mathbf{R^2}$) : 0.548 ; mean absolute error (MAE) : 0.937. Linear regression models also gives good results (R2 : 0.476 MAE : 1.032). Over 90% of the predictions have a relative error of less than 20%. We found simple stroke parameters alone to be sub-optimal ; the results were better when using parameters from the triangle test. In conclusion, our study demonstrates that the Sigma-Lognormal model offers new possibilities for estimating the motor control maturity. Our method is fast and comfortable for the children, as it only requires performing handwriting strokes on an electronic tablet. This simple and user-friendly test is expected to be more convenient for doctors in a clinical context for pre-screening with less professional intervention.

4.2 Introduction

Motor control is a fundamental skill that develops progressively during childhood. Well-developed motor control allows children to learn and live better. Therefore, it is important to understand how we can best measure the motor control maturity and any deviation from typical motor development. Motor performances are most often measured by 1) qualified professionals administering motor ability tests or 2) by the parent, teachers, or clinicians filling questionnaires based on behavioral observations [1]. Some tests also evaluate motor control according to performances in a video game but this requires specific instruments [2]. Further, different levels of experience with the game may cause bias in children's performances. None of these methods are particularly convenient due to the additional human or material resources required. Also, for traditional assessment methods, especially assessments with

questionnaires, the influence of the cultural aspect is not negligible, so the cultural adaptation (e.g., translation) may increase the difficulty of use for children from different countries [1]. However, our approach is less affected by cultural differences, so it can be applied more easily in different countries. Because there is less cultural difference in using a pen to draw some lines, this is a more natural way to show the child's fine motor control ability.

Handwriting is a typical fine motor control task in educated people. The Kinematic Theory of rapid human movements [3–6] and its Sigma-Lognormal model [7] have been used in several biomedical applications to assess fine motor control associated with handwriting. For example, it has been used for analyzing graphomotor performances in kindergarten children [8], for assessing brain stroke risk [9], and for assessing the presence of Attention-Deficit/Hyperactivity Disorder (ADHD) in children [10]. In this study, we propose to further build on these studies and use a tablet-based system [23] to estimate the motor control maturity in children exploiting the Kinematic Theory. But to our knowledge, no one in the field of motor control has explored the effectiveness of using Sigma-Lognormal parameters to study the motor control maturity in children. The Sigma-Lognormal model decomposes the velocity profile of the movement in several lognormal functions, each capturing the kinematics associated with a neuromuscular command. For each lognormal, a set of 6 parameters is extracted :

$$\{t_0,\ D,\ \mu,\ \sigma,\ \theta_{start},\ \theta_{end}\}_i^{nbLog}$$

- t_0 : The time (in seconds) of the emission of the motor command. In the context of a psychomotor test, this is the time when the nervous system starts to react after receiving a start signal (e.g., a sound or a visual stimulus).
- D : The amplitude (in millimeters) of the lognormal stroke. It corresponds to the total distance of the trajectory associated with this movement primitive.
- μ : The logarithmic time delay (in $\ln(s)$) to reach the logarithm of the median of the motion. This parameter reflects the overall speed of reaction.
- σ : The logarithmic response time in $\ln(s)$. It characterizes the duration of the motion.
- $\theta_{start}, \theta_{end}$: The start and end angles of motion in radians.

  Three parameters are also used to evaluate the reconstruction of the movement : SNR, $nbLog$, and $\frac{SNR}{nbLog}$. These parameters are defined as follows :
- SNR : The signal-to-noise ratio measuring the quality of reconstruction of the velocity profile compared to the recorded velocity. A higher SNR value indicates a better reconstruction of the velocity profile.
- $nbLog$: The number of lognormal functions used to reconstruct the velocity profile. It indexes the smoothness of the motion. The lower this value, the smoother the motion.

In the ideal case, the minimum value of *nbLog* is two for the simple stroke test and six for the triangle test (i.e., the triangular movement is composed of three single strokes).

— $SNR/nbLog$: The ratio between SNR and *nbLog*. It reflects the fluidity of the movement.

4.3 Method and experiments

4.3.1 Participants

The aim is to develop a model that correlates the Sigma-Lognormal parameters with the motor control maturity of neurotypical children. Parents were asked to fill out medical questionnaires about existing neurological, psychological, and motor disorders. These disorders include ADHD, anxiety disorder, oppositional defiant disorder, autism spectrum disorder, Asperger's syndrome, epilepsy, Tourette syndrome, Down syndrome, tic disorders, impulsivity-control disorder, intellectual disorder, muscular hypotonia, or coordination disorders. Participants whose parents reported such disorders were excluded from our analyses. Also, trials reconstructed with an SNR lower than 15dB were rejected. A total of 513 normal children (457 right-handed/56 left-handed ; 281 girls/232 boys) aged from 6 to 13 years from three different Montréal schools participated in the tests. Since previous studies have demonstrated that there were no significant differences between subjects working with their dominant hands [42] and that more effects were observed according to age groups rather than sex in fine motor control tests [43], we have focused on age effects in the present study.

4.3.2 Sigma-Lognormal tests

First, we recorded the movement performed by the children on a tablet-based system [23]. Then, we extracted the Sigma-Lognormal parameters using ScriptStudio, a proprietary program developed by our team to study human movement using the Kinematic Theory. In this study, we studied data from the triangular drawing test and the single stroke test. For each test, participants performed 30 trials. Invalid trials were automatically removed (e.g., unfinished stroke, very slow movements). The direction of the movement was inverted depending on participant handedness so that the same group of muscles gets solicited regardless of handedness [23].

Simple stroke test

For the simple stroke test, the participants are asked to draw a straight line between a starting region and an ending region (Figure 4.1). The starting signal is a tone that lasts 500ms with a frequency of 1kHz. According to the Kinematic Theory, an optimal movement performed for this task has two lognormal functions, one for an agonist component and a second for an antagonist component. In practice, the number of lognormals used to model each movement is automatically determined by the extraction system as a trade-off between maximizing the SNR and minimizing the $nbLog$. Movements with more than six or less than two lognormals were rejected. Data from 511 children were finally used for this test (two children did not finished enough trials in this test).

Triangular drawing test

For the triangular drawing test, the participant had to draw a triangle crossing three round targets (Figure 4.2). The starting signal is a light on the screen that changes from red to green. The number of extracted lognormals is set to six because these triangular movements have been modeled using a prototype-based approach [44] using a template made of a combination of three simple strokes. Data from 509 children are used for this test (two more children did not finished enough trials in this test comparing the previous test).

4.3.3 Transformation of data

As some participants choose the clockwise orientation of drawing the strokes, their angles are complementary of π with those strokes made of counterclockwise orientation. We thus make the one-hot encoding according to the orientation of drawing the strokes. [1,0] represents the clockwise orientation. [0,1] represents the counterclockwise direction. However, the angles of the clockwise orientation are highly different from the angle with opposite direction for drawing, to facilitate the training process of the models, the clockwise angles were transferred to get to the closer range of the counterclockwise angles since the orientation information is already coded :

$$\theta_{new} = atan2(\frac{\sin(\theta_{old})}{-\cos(\theta_{old})}) \tag{4.1}$$

These θ_{new} were further transformed as we found, due to the cyclic nature of angular parameters, that using the sines and cosines instead of the angles increase the performance of our prediction models. Further, apart from using the sines and cosines of θ_{new}, we also

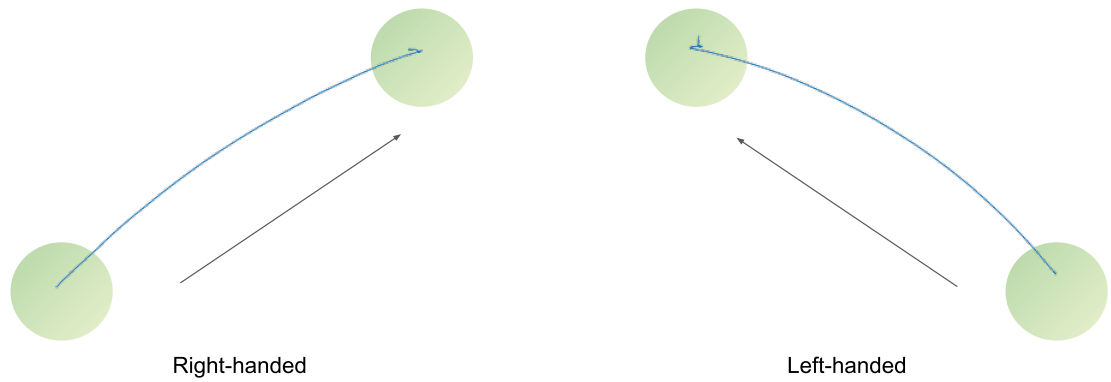

Figure 4.1 Simple stroke.

calculated :

$$\overline{\theta} = arctan2\left(\frac{sin(\theta_{start_new}) + sin(\theta_{end_new})}{cos(\theta_{start_new}) + cos(\theta_{end_new})}\right) \tag{4.2}$$

$$\Delta(\theta) = \theta_{end_new} - \theta_{start_new} \tag{4.3}$$

Hence, the parameters used in our algorithms are : three non-sequential parameters (SNR, $nbLog$, $\frac{SNR}{nbLog}$) and two for one-hot encoding of stroke orientation) and 11 sequential parameters per lognormal function :

$$\{t_0,\ D,\ \mu,\ \sigma,\ sin(\theta_{start_new}),\ sin(\theta_{end_new}),$$
$$cos(\theta_{start_new}),\ cos(\theta_{end_new}),\ sin(\overline{\theta}),\ cos(\overline{\theta}),\ \Delta(\theta)\}_i^{nbLog}$$

The size of our input for one subject is, therefore, equal to $T * L * 11$ for sequential features and $T * 5$ for non-sequential features, with T being the number of valid movements and L being the number of lognormals. In particular, $nbLog$ and $\frac{SNR}{nbLog}$ are not used with triangular movements, because the number of lognormals is fixed at six for this test, so $\frac{SNR}{nbLog}$ is collinear with SNR and $nbLog$ and $\frac{SNR}{nbLog}$ contribute no additional information as compared to the SNR.

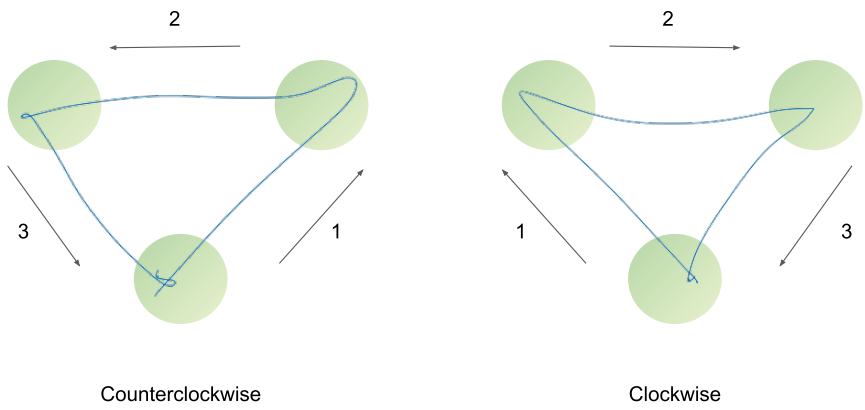

Figure 4.2 Triangular drawing test.

4.3.4 Experiments

The large variability of the movements produced by children is the first difficulty of this study. Even though they could practice a few trials before beginning a formal test, the movements they produced are naturally different from trial to trial. In addition, each child may initially show some residual learning effect at the beginning, then stabilize and show fatigue toward the end of the test. It is difficult to determine the motor control maturity of the participants using a single trial. Their average performance overall valid trials could be expected to index their motor control maturity. Therefore, we distinguished two ways to use these data. The first one is to train models using every movement individually, then make an average of the predictions per participant. Using this approach is equivalent to ignoring the within-subject variance in performance. Some subjects may produce some movements that are very different – either better or worst – compared to their average performances. This individual variance may interfere with the training of our models. The second is to calculate the mean movement parameters for each participant and then train the model. It reduces the effect of variance, so the input is more correlated to their typical performance, but we also lose some information through the averaging. The last one is using all a participant's movements together, without computing averages. In this case, to use a classic method like linear regression, we need to vectorize our tensor input to a long vector of $N*(T\cdot 71)$ (6 lognormals with 11 parameters plus 5 non-sequential parameters, for the triangular drawing task) which makes models harder to train. In addition, for the simple stroke test, the number of lognormals is variable, so the

obtained tensor is padded using masks to standardize the number of features across trials and these masks interfere with the training process. On the contrary, the deep learning models can handle this with flexible and custom structures and benefit from the full information of all the trials available for a participant. These methods are compared in the following. For each test, 30 movements are recorded per participant, but some invalid trials get discarded. For participants that do not have 30 valid trials, we pad using masks so that the length of the tensors is fixed to 30. When the average strokes are used, no padding is necessary, since we just average lognormal parameters across corresponding lognormal components for each participant (i.e., in this case, T=1 after the averaging). The second difficulty is the variability in how motor control maturity translates into a range of produced movements. In accordance with the *Lognormality Principle* [45], we assume that motor control maturity correlates with ages in neurotypical children. This assumption is correct for most children, but there are still some participants whose motor development is delayed or in advance. In addition, the performance can depend on temporary internal (e.g., fatigue, motivation) or external (e.g., distracting sounds) conditions. Hence, our labels (i.e., the age of children) are accurate but the interference of external factors to the test makes the model training more difficult. The last difficulty is the lack of data. Testing the motor control skills of a large number of children is an expensive process, which limits the amount of data that can be collected. With a limited sample size, model validation becomes a challenge [46]. To cope with this problem, we use the nested-cross-validation [47] which helps to get an estimation of the performance of each type of model with a less optimistic bias, especially for the small dataset. All hyperparameters are optimized in the inner cross-validation cycle whereas classification performance is tested in the outer cycle (Figure 4.3). We used five folds for the outer cycle and four folds for the inner cycle. Hence, the dataset is split 60%-20%-20% for train-test-validation sets. Then, the final test score is the average score of all models trained in the outer loop.

4.3.5 Configuration and models to compare

The models and hyperparameters optimized with nested-cross-validation are as follows :

Recurrent neural network (RNN)

RNN has shown its efficiency to process sequential data in several domains, including human handwriting [26]. We use improved RNN structures called Gated Recurrent Unit (GRU) [39] to construct our RNN-based model.(Figure 4.4) The non-sequential parameters are first passed through a fully connected (FC) layer and then the output is concatenated with the output of the GRU. Human motor control performance is naturally variable and should not

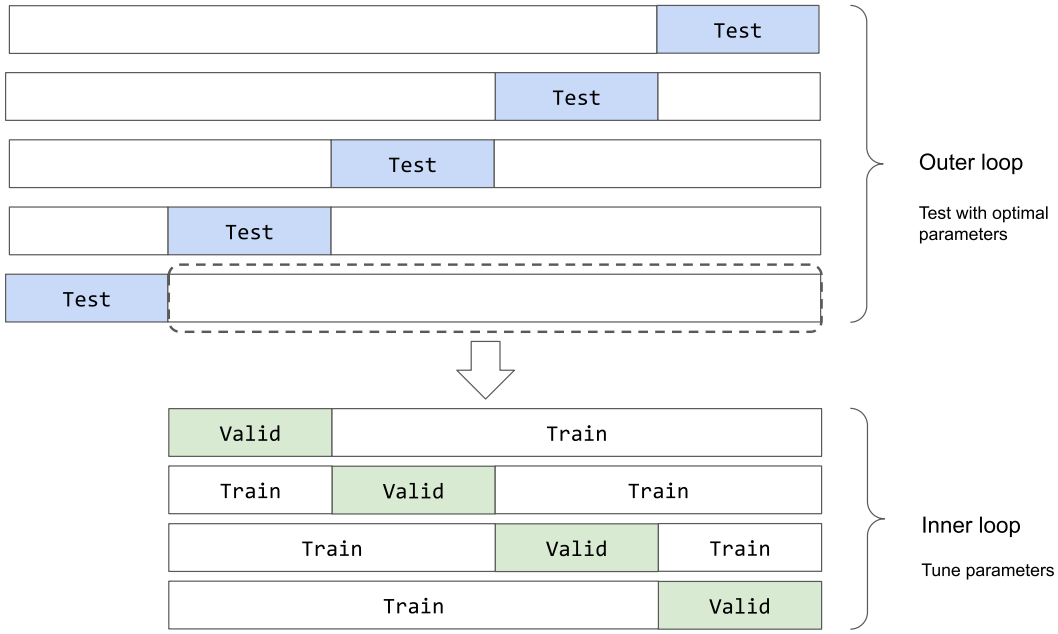

Figure 4.3 Nested-cross-validation

be modeled as a fixed value. We assume that the real motor control performance can be represented as a latent variable following a Gaussian distribution in the latent space of our neural networks. Like the Variational Auto-Encoder (VAE) [40], we used two FC layers to obtain these two tensors (as their output) which represent the means μ and variance σ^2 of our current latent distribution. During the training stage, the latent representation tensor is sampled from this latent distribution $\mathcal{N}_k(\mu, \sigma^2)$, which is equivalent to adding Gaussian noise with a variance σ^2 on our original tensor. This new noisy tensor is used to finish the regression tasks. We can consider this noise as a regularized term with an adaptive intensity during training steps. Since the downstream task is the regression, the more noise is added, the harder the prediction task. To minimize the regression loss, the neural nets tend to set the variance to 0 to get a deterministic prediction. Here, we introduce an auxiliary loss which is the difference between our latent distribution and the normal distribution $\mathcal{N}_k(0, 1)$ measured by the Kullback–Leibler divergence. Hence, our goal is to keep this noise such that our neural net is forced to find a robust latent variable. This auxiliary loss can give slightly better results. The coefficient of auxiliary loss is set to 0.01. Further, we performed an ablation experiment to better illustrate the properties of the auxiliary loss (see appendix A). Finally, using the sampled latent vector, we get T prediction from T trials. The final prediction value of motor control maturity is the average of these T values. We optimized our neural networks with the ADAMW [48] algorithm (an improved version of ADAM [49] that correct the application

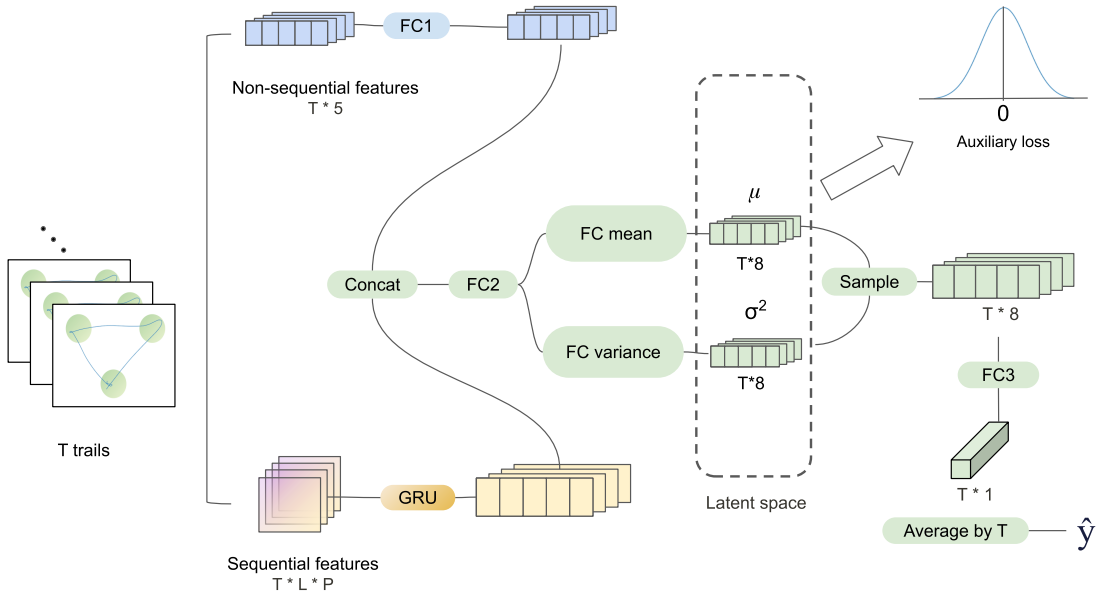

Figure 4.4 GRU based models: Here we show the model structure with T trials of 1 participants. The concatenation is done in the feature dimension, all tensor in this has a first dimension T until the last prediction. In the last step, we have a series of predicted results of length T. The average value is calculated based on the number of recorded trials, but not on the maximum number of trials after padding.

of weight decay in Pytorch) with a batch size of 64 and 150 epochs. The optimal learning rate is searched within the [1e-5, 0.005] range. To avoid overfitting, the dropout [50] and the weight decay [51] are used and their values are searched in the [0, 0.5] and the [0, 0.1] ranges. Gradient clipping (set to 0.05) is used to avoid the gradient explosion and make the backpropagation more stable. We used the hyperbolic tangent as the activation function. The GRU has two layers with 64 hidden neurons. Five neurons were used for the FC for non-sequential features and the latent size is 8. We used the mean absolute error (MAE) as a basic loss function for the regression.

Multilayer perceptron (MLP)

The MLP is a classical multilayer feedforward neural network. For its implementation, we used four intermediate layers, with 64 neurons each, and a Rectified Linear Unit (ReLU) for the activation function. The learning rate, the level of gradient clipping, the dropout, and the weight decay are as previously described for the GRU-based model. This model is also optimized with the ADAMW algorithm and a batch size of 64 and 150 epochs. We choose the MAE as the loss function. The model uses T trials together for each subject like our GRU

model, but MLP does not have a recurrent structure like GRU. The sequential parameters of each lognormal function must be concatenated together. The input size becomes $N * T * (L \cdot P + 5)$.

Ordinary least squares regression (OLS)

This algorithm is a classical regression method that minimizes the square of the fitting error.

Ridge regression (RR)

This model is the combination of OLS with regularization of $L2$ norms coefficients. The weight of the regularization is searched within the $[0, 5]$ range.

Huber regression (HR)

Huber regression [31] optimizes the square loss when the example makes an error whose absolute value is smaller than a C value, otherwise, it optimizes the absolute loss. This C value is controlled by a hyperparameter ϵ multiplied by the standard deviation σ of the errors of all the training examples. The advantage of this mixed loss function is that the optimization is not strongly affected by outliers, but does not completely ignore their effect. ϵ is searched with the $[1, 2.5]$ range.

$$L_\epsilon(y, \hat{y}) = \begin{cases} (y - \hat{y})^2/2 & |y - \hat{y}| \leq C \\ C \cdot |y - \hat{y}| - C^2/2 & \text{otherwise} \end{cases} \tag{4.4}$$

Support vector regression (SVR) [52]

This method is tested with the Radial basis function (RBF) kernel. Its γ hyperparameter is set within the $[1e-5, 0.1]$ range and the L2 norms regularization coefficient $C \in [0.1, 5]$.

$$K(x_i, x_j, \gamma) = \exp\Big(-\gamma \|x_i - x_j\|^2\Big) \tag{4.5}$$

XGBoost (XGB)

XGB [53] is a framework of the gradient boosting algorithms.

Random forest (RF)

RF is a linear method that trains a group of decision trees using subsets of the data. It aggregates all trees by averaging the prediction value across trees [34]. The optimal values for the number of trees, max depth, and max_feature are searched within $[10, 150]$, $[1, 15]$, and $[0.1, 1]$, respectively.

K-nearest neighbors regression (KNN)

This method calculates the prediction value as the average of the K-nearest data points using the Euclidean distance (L2) [54]. The number of nearest neighbors is searched within the $[1, 25]$ range.

We tested and compared the performance of each of these models on two datasets (simple strokes or triangular tests), using the two approaches previously described (i.e., mean trial, full trials).

4.4 Results

Aside from assessing the performance of the regression model using the coefficient of determination (R^2), the mean absolute error (MAE) and the root mean squared error (RMSE) were also computed to compare mean errors. The RMSE is a natural choice for normally distributed residuals as it tends toward the value of the standard deviation for large samples [55]. Moreover, the mean absolute percentage error (MAPE) helps compare the errors relative to the age of the participants. Apart from the R^2, the other metrics are minimized as well.

$$\text{MAPE} = \frac{100}{n} \sum_{i=1}^{n} \frac{|y_i - \hat{y}_i|}{y_i} \tag{4.6}$$

As it can be seen in Table I and II, results are significantly different for the two tests, with the models for the triangular test performing better than the simple stroke models (differences in R^2 around 0.15-0.2, see appendix C). Our first hypothesis was that using a larger number of lognormals is not ideal for modeling the simple stroke test, even if the reconstruction error is minimal. To verify this, we fitted the models again using only the trials fitted with exactly two lognormal functions. The result is shown in Table 4.3. As compared to Table I, only the linear regression models obtain a slight increase in performance (see appendix C), not convincingly supporting our first hypothesis. Therefore, these lower performances for the single stroke test may rather be due to the simplicity of this test. Since even the youngest children master this movement, this test may show a ceiling effect compared to the

Table 4.1 Table for simple stroke

Data	Model	RMSE	MAE	MAPE	R2
Mean trial	OLS	1.516	1.220	0.133	0.283
	HR	1.514	1.218	0.133	0.285
	RR	1.484	1.206	0.132	0.313
	KNN	1.521	1.285	0.139	0.279
	RF	1.460	1.215	0.133	0.335
	XGB	1.478	1.230	0.135	0.318
	SVR	**1.450**	**1.191**	**0.130**	**0.343**
	MLP	1.487	1.214	0.132	0.310
	GRU	1.501	1.222	0.133	0.297
Full trials	OLS	1.553	1.334	0.147	0.249
	HR	1.583	1.367	0.150	0.220
	RR	1.585	1.369	0.151	0.218
	KNN	1.616	1.392	0.156	0.186
	RF	1.541	1.324	0.146	0.261
	XGB	1.518	1.299	0.143	0.283
	SVR	1.497	1.270	0.140	0.302
	MLP	1.485	1.199	0.133	0.309
	GRU	**1.425**	**1.125**	**0.123**	**0.365**

harder triangular test which may better differentiate age-related gain in performance. We used three nested cross-validations, with five sub-cross-validation each, for a total of 15 mutually independent tests. The box plots in appendix C show the performances (R^2) of all models in 4 different situations (two tests with two different data usage). One-way ANOVA (analysis of variance) on R^2 showed only marginal statistical effect for the mean trial and the simple tests (F=1.86 ; p=0.07). However, with full trials for the same tests, the neural networks performed significantly better than other models (t⩽-2.98 ; p⩽6.77e-3). We observed a similar situation for the triangular tests, with all models having similar performances with mean trials (except KNN : F=1.23 ; p=0.29). However, with full trials, the neural networks outperform others (t⩽-3.71 ; p⩽1.20e-3). With the triangular tests, the linear model and the SVR trained with the mean trial have similar performance to GRU models. The small difference in performances can be due to the Sigma-Lognormal parameters already being predictive enough for even linear regression (Ridge) to works well. Further, the lack of data probably limits the performance of neural networks which would be expected to perform better with more data. These results further validate our claim that neural networks can handle complex data structures better than the traditional models, although traditional models perform relatively well when using

Table 4.2 Table for triangular test

Data	Model	RMSE	MAE	MAPE	R2
Mean trial	OLS	1.290	1.032	0.113	0.476
	HR	1.280	1.027	0.112	0.484
	RR	1.239	0.986	0.108	0.517
	KNN	1.365	1.136	0.126	0.416
	RF	1.302	1.060	0.116	0.468
	XGB	1.268	1.029	0.113	0.493
	SVR	**1.210**	**0.971**	0.107	**0.539**
	MLP	1.278	1.010	0.110	0.486
	GRU	1.252	0.978	**0.106**	0.505
Full trials	OLS	1.385	1.166	0.128	0.400
	HR	1.382	1.162	0.128	0.403
	RR	1.386	1.167	0.128	0.399
	KNN	1.453	1.237	0.140	0.341
	RF	1.393	1.178	0.130	0.393
	XGB	1.364	1.149	0.127	0.418
	SVR	1.319	1.094	0.120	0.456
	MLP	1.244	0.991	0.108	0.515
	GRU	**1.196**	**0.937**	**0.102**	**0.548**

features that are sufficiently predictive. This reinforces suggest that the sigma-lognormal model can provide highly efficient features to characterize motor control.

We also observe that SVR and GRU perform best for both types of data usage. During three times nested-cross-validations, for the GRU model, 33.46% of prediction returns a relative error of less than 5%, 77.60% of prediction make an error of less than 15% and 92.53% of prediction makes an error of less than 25%. 89.19% of predictions give an absolute error of fewer than two years. The SVR has similar performance (see appendix B, C).

We calculated the ICC1 (Intraclass Correlation Coefficient 1) based on the errors of these three models. This measure equals the fraction of the variance between groups of same subject outcomes over the total variance. The larger this value, the smaller the prediction variance of these different models. ICC1 for all models had 95% confidence intervals within 0.92-0.95 indicating that the variance of predictions between these models is relatively small.

Table 4.3 Table for simple stroke with only 2 lognormal functions

Data	Model	RMSE	MAE	MAPE	R2
Mean trial	OLS	1.433	1.170	0.128	0.359
	HR	1.434	1.170	0.128	0.359
	RR	1.435	1.171	0.128	0.357
	KNN	1.520	1.274	0.141	0.280
	RF	1.465	1.220	0.134	0.330
	XGB	1.468	1.216	0.133	0.326
	SVR	1.458	1.182	0.129	0.336
Full trials	GRU	1.438	1.153	0.127	0.354

4.5 Discussion and future work

Thanks to the Sigma-Lognormal model, it is possible to estimate the evolution of the motor control maturity with efficiency and accuracy. Once the movements are modeled by the Sigma-Lognormal model, even a simple linear regression can give a decent result. As a daily activity, handwriting is easy to acquire and analyze. Health monitoring using these tests can be used for everyone able to hold a pen. We compared several different algorithms in this study and found GRU and SVR to perform best. Neural networks have huge freedom in customizing their data structure to fit most needs. The downside of this freedom is that they need more data and computing resources. By contradistinction, symbolic algorithms have simpler structures. They are easier to train and their results can be explained in the context of the Kinematic Theory. That is not the case with black-box deep learning models. If the model configuration matches our problem and data, symbolic algorithms can perform very well, even better than neural networks in some cases. One limitation of this study is that all models used the same features, and the effect of feature selection has not been investigated. In practice, each model may perform better with a different selection of features [56].

For future work, we plan to study the kinematics of additional tests. Further, we will consider analyzing the original time series describing the movement kinematics. However, these original data have much larger dimensions than the Sigma-Lognormal parameters. Therefore, symbolic algorithms like SVR will probably not be suitable anymore, and large neural networks will be preferred. Work on self-supervised learning [57] can also be inspiring on how to deal with such long sequential data. These algorithms can train the models to predict the masked parts of sequential points without the label information. This approach can help optimize the model for supervised training steps and thus have a better final performance. This

pre-training technique is popular in the natural language processing domain, but it can also work with time series like in handwriting. In addition, various flavors of VAE models were shown to be able to disentangle the input using a latent space and can generate synthetic samples using this latent space [58–61]. The Kinematic Theory and its Sigma-Lognormal are doing the same things : decompose the original movement into lognormal components and reconstruct that movement from their parameters. Thus, we can consider the Sigma-Lognormal model as a symbolic model and the VAEs as connectionist models and provide an interesting comparison. The two types of models can take advantage of each other or work together to reach the highest performance. This approach could also help us better understand human body motions involved in handwriting.

4.6 Conclusion

In this paper, we have shown a new approach to predicting the motor control maturity of children from neuromuscular tests using the Sigma-Lognormal model. From our experiment, it seems that the simple stroke test might not be sufficient to distinguish the motor control maturity of children of different ages. However, with a more complex test like the triangular test, the Sigma-Lognormal model gives parameters predicting the motor control maturity with a simple linear regression. Neural networks perform best in this task, thanks to their flexible structure, but symbolic models also show good capabilities. Subsequent studies could improve the evaluation of these tests by comparing them against alternative tests and by evaluating their test-retest reliability. In the future, this approach could be used to detect neurodevelopmental issues, identifying children who are potentially ahead or behind in their development. This would allow a teacher further help students that are behind or give more challenging tasks to students that are ahead.

Acknowledgments

The data collection was supported by a grant from the TransMedTech Institute to Réjean Plamondon.

4.7 Appendix A

To compare the effect of our auxiliary loss, we also train GRU based models without the auxiliary loss and no sampling process for prediction. The comparison is shown in the Table 4.4. We observed that the auxiliary loss can slightly help the optimization of our GRU models.

Table 4.4 ablative table of auxiliary loss for the GRU based model

Aux Loss	Data Processing	RMSE	MAE	MAPE	R2
✓	Tri Mean	1.252	0.978	0.106	0.505
	Tri Full	1.196	0.937	0.102	0.548
	Simple Mean	1.501	1.222	0.133	0.297
	Simple Full	1.423	1.125	0.123	0.365
✗	Tri Mean	1.266	0.998	0.108	0.493
	Tri Full	1.221	0.962	0.105	0.531
	Simple Mean	1.515	1.233	0.135	0.281
	Simple Full	1.455	1.172	0.129	0.338

4.8 Appendix B

B.1 Error fraction

The following tables show the error fraction of each model during 3 times nested-cross-validations. We calculate the absolute error fraction and the percentage absolute error fraction. The error fraction for GRU models :

Table 4.5 fraction of Percentage absolute error of GRU models

Percentage abs error(%)	**Fraction(%)**	**Cumulative fraction(%)**
5.0	33.46	33.46
10	26.46	59.92
15	17.68	77.60
20	14.93	92.53
25	3.47	96.00
30	3.14	99.14
40	0.46	99.60
$\geq$45.5	0.39	100

Table 4.6 fraction of absolute error of GRU models

Absolute error(years)	**Fraction(%)**	**Cumulative fraction(%)**
1	62.54	62.54
2	26.65	89.19
3	9.23	98.42
4	1.38	99.80
$\geq$4	0.20	100

The error fraction for SVR models :

Table 4.7 fraction of Percentage absolute error of SVR models

Percentage abs error(%)	**Fraction(%)**	**Cumulative fraction(%)**
5.0	30.39	30.39
10	27.05	57.44
15	18.80	76.24
20	15.39	91.63
25	4.58	96.21
30	2.69	98.90
40	0.72	99.62
$\geq$45.5	0.39	100

Table 4.8 fraction of absolute error of SVR models

Absolute error(years)	**Fraction(%)**	**Cumulative fraction(%)**
1	58.61	58.61
2	31.37	89.98
3	8.58	98.56
4	1.38	99.94
$\geq$4	0.07	100

The error fraction for Ridge regression models :

Table 4.9 fraction of Percentage absolute error of Ridge regression models

Percentage abs error(%)	**Fraction**	**Cumulative fraction**
5.0	28.55	28.55
10	29.14	57.69
15	18.40	76.09
20	15.98	92.07
25	3.86	95.93
30	2.88	98.81
40	0.85	99.66
$\geq$45.5	0.33	100

Table 4.10 fraction of absolute error of Ridge regression models

Absolute error(years)	**Fraction**	**Cumulative fraction**
1	58.87	58.87
2	29.86	88.73
3	9.36	98.09
4	1.64	99.73
$\geq$4	0.26	100

B.2 Real age vs predicted age

Finally, the Fig. 4.5 shows the scatter plot of the real age vs.the age predicted by GRU models with the full trials triangular data.

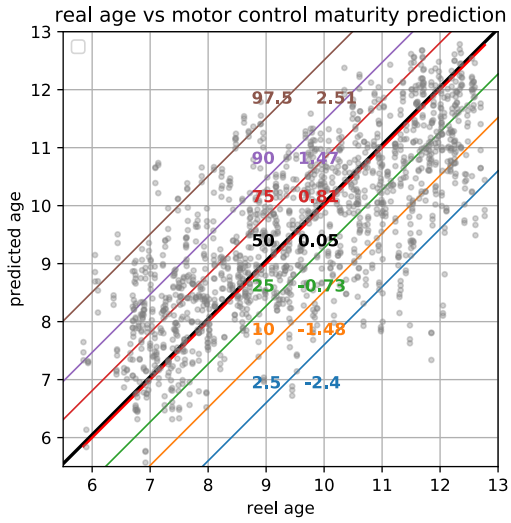

Figure 4.5 The real age vs. prediction value with quantities of [2.5, 10, 25, 50, 75, 90, 97.5]

4.9 Appendix C

C.1 Comparison of all models

The figures 4.6- 4.9 show the R^2 performance of each model with different data sets and different ways of using the data. Each box plot shows the distribution of performance of every test for each model. As we mentioned in the main text, the triangular test gives better results than the simple test in both ways of using data for all models. The Table 4.11 show the p-values of T-tests by comparing all metrics of the same model in both datasets, the improvement of performance is significant by changing the dataset.

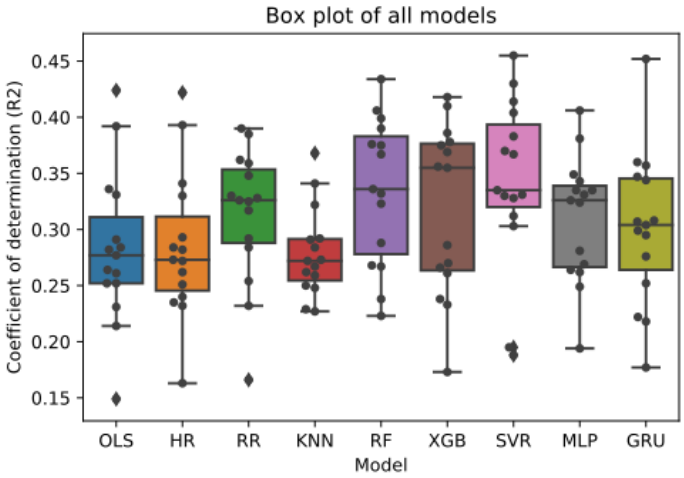

Figure 4.6 mean trials with the simple test

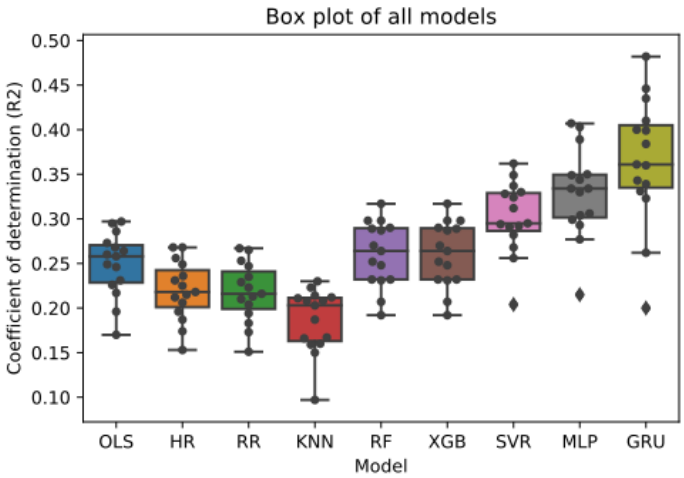

Figure 4.7 full trials with the simple test

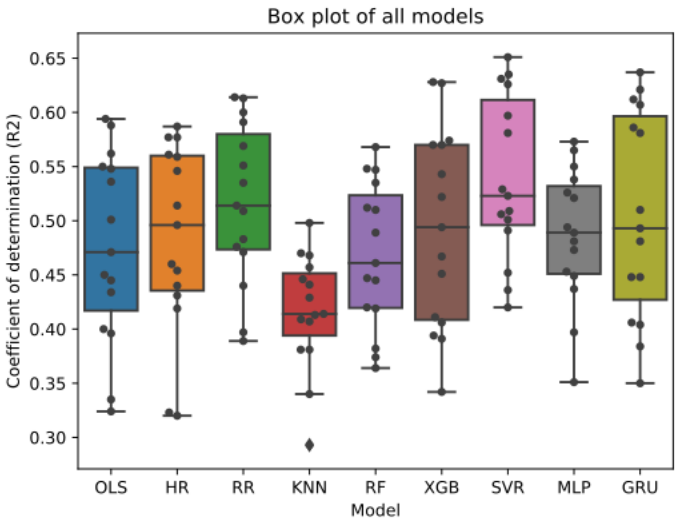

Figure 4.8 mean trials with the triangular test

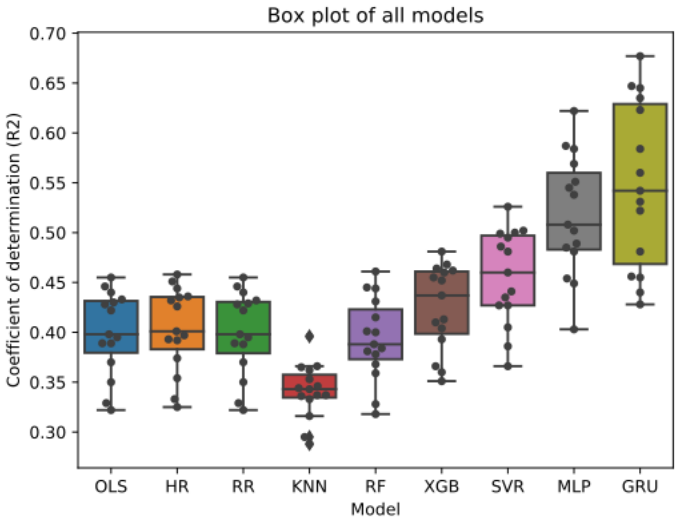

Figure 4.9 full trials with the triangular test

Table 4.11 p-values of the T-test (simple test vs triangular test)

Data	Model	RMSE	R2	MAE	MAPE
Mean trial	OLS	7.31e-07	3.52e-07	1.410e-07	6.140e-07
	HR	2.899e-07	1.732e-07	7.814e-08	3.631e-07
	RR	1.79e-08	7.49e-09	3.38e-09	2.049e-08
	KNN	2.986e-06	1.576e-08	1.682e-06	2.742e-04
	RF	3.228e-05	7.307e-06	1.672e-05	7.848e-05
	XGB	7.141e-06	4.763e-06	2.779e-06	1.088e-05
	SVR	1.344e-07	1.115e-07	5.397e-08	6.766e-07
	MLP	1.630e-08	7.396e-09	2.550e-09	5.521e-08
	GRU	1.371e-07	5.486e-07	1.098e-09	8.211e-09
Full trials	OLS	7.318e-07	3.527e-07	1.410e-07	6.140e-07
	HR	2.899e-07	1.732e-07	7.814e-08	3.631e-07
	RR	1.799e-08	7.496e-09	3.386e-09	2.049e-08
	KNN	2.986e-06	1.576e-08	1.682e-06	2.742e-04
	RF	3.228e-05	7.307e-06	1.672e-05	7.848e-05
	XGB	7.141e-06	4.763e-06	2.779e-06	1.088e-05
	SVR	1.344-07	1.115-07	5.397-08	6.766-07
	MLP	1.630e-08	7.396e-09	2.550e-09	5.521e-08
	GRU	1.371e-07	5.486e-07	1.098e-09	8.211e-09

C.2 Comparison for the only 2 lognormal functions

We also compared each model in the simple test data with only 2 lognormal functions and the original simple test data. There is no performance improvement for all models except the linear model. The Table 4.12 show the p-values of T-tests for those models performance in two different datasets (with only 2 lognormal functions or use all lognormal functions).

Table 4.12 p-values of the T-test with/without only 2 lognomral functions in simple tests

Data	**Model**	**RMSE**	**R2**	**MAE**	**MAPE**
Mean trial	OLS	0.010	0.001	0.041	0.0957
	HR	0.008	0.001	0.040	0.0911
	RR	0.085	0.022	0.155	0.249
	KNN	0.970	0.940	0.699	0.620
	RF	0.886	0.842	0.867	0.826
	XGB	0.811	0.776	0.709	0.714
	SVR	0.824	0.791	0.775	0.767
Full trials	GRU	0.657	0.680	0.267	0.284

C.3 Comparison MLP vs GRU

We compare two structures of neural networks. The GRU outperformed MLP significantly for some metrics (MAE/MAPE) in both datasets if we use the full trials. The Table 4.13 shows the p-values of T-tests for compare those 2 models.

Table 4.13 p-values of the T-test (MLP vs GRU)

Data	**Model**	**RMSE**	**R2**	**MAE**	**MAPE**
Simple mean trials	MLP VS GRU	0.786	0.703	0.913	0.895
Simple full trials	MLP VS GRU	0.102	0.122	0.006	0.003
Triangular mean trials	MLP VS GRU	0.408	0.549	0.192	0.233
Triangular full trials	MLP VS GRU	0.174	0.262	0.038	0.034

CHAPITRE 5 DISCUSSION GÉNÉRALE

Ce projet de recherche visait à implémenter un algorithme innovant pour la prédiction de la maturité du contrôle neuromoteur des enfants et qui soit robuste. C'est pourquoi la Théorie Cinématique des mouvements humains rapides a été choisie dans le premier temps, ses paramètres Sigma-Lognormaux étant des représentations robustes du contrôle moteur et permettant des analyses systématiques de la performance des participants. Dans cette recherche, il existe encore des points intéressants qui ne sont pas beaucoup développés dans le chapitre précédent : 1. L'interprétation des résultats. 2.Les limitations de cette étude. Nous discutons ici de chacun des points.

5.1 L'interprétation des résultats

D'après les résultats présentés au chapitre précédent, la régression par vecteurs de support (SVR) et le réseau de neurones de GRU obtiennent les meilleures performances. Le SVR est comparable au réseau de neurones de GRU dans notre situation. D'ailleurs, Domingos Pedro [62] nous donne un autre aspect de comparaison de ces deux modèles. Il a démontré que tous les réseaux de neurones optimisés par le gradient descendant peuvent être considérés approximativement comme une machine de noyau avec le noyau de chemin (path kernel en anglais). Supposons que l'on optimise un modèle avec la descente de gradient, le chemin d'optimisation du modèle noté comme c(t). Le noyau de chemin mesure la similarité entre deux points pendant l'optimisation du modèle. Si l'on considère que le taux d'apprentissage est infiniment petit, le noyau de chemin calcule l'intégrale du produit scalaire des gradients du modèle aux ces deux points sur le chemin pris par les paramètres pendant la descente de gradient. D'où le réseau de neurones stocke tous les exemples pendant la phase d'apprentissage. Dans la phase de prédiction, le réseau de neurones compare l'exemple de requête et les exemples d'apprentissage et produit une somme pondérée en correspondant simultanément l'exemple de requête avec chaque exemple stocké.

Par ailleurs, nous avons montré que les distributions d'erreurs du SVR, la régression linéaire et le modèle de réseau de neurones de GRU sont similaires. Mais nous voulons savoir si le réseau de neurones de GRU est aussi robuste que le SVR et la régression linéaire. Par exemple, si la régression linéaire produit une prédiction de maturité du contrôle moteur qui diffère significativement du vrai âge du participant, nous pouvons vérifier quels paramètres sont responsables. Contrairement, le réseau de neurones ne peut pas nous donner ces informations à cause de l'ultra-haute complexité des boîtes noires. Dans la section *Results* de l'article,

nous avons montré l'ICC(Intraclass Correlation Coefficient) de ces trois modèles. Nous avons produit ces 3 modèles avec les mêmes données d'apprentissage et fait les prédictions avec même données de test, puis nous mettons les prédictions de chaque personne pour chaque modèle différent dans le même groupe. Cette quantité décrit à quel point les unités d'un même groupe se ressemblent, dans notre situation, elle représente à quel point les prédictions faites par les différents modèles se ressemblent. Les résultats montrent que les sorties de ces trois modèles sont similaires, il n'y a pas de grand nombre de prédictions contradictoires entre les modèles. Ces figures montrent les différences de prédictions de trois modèles deux par deux pour les mêmes exemples.

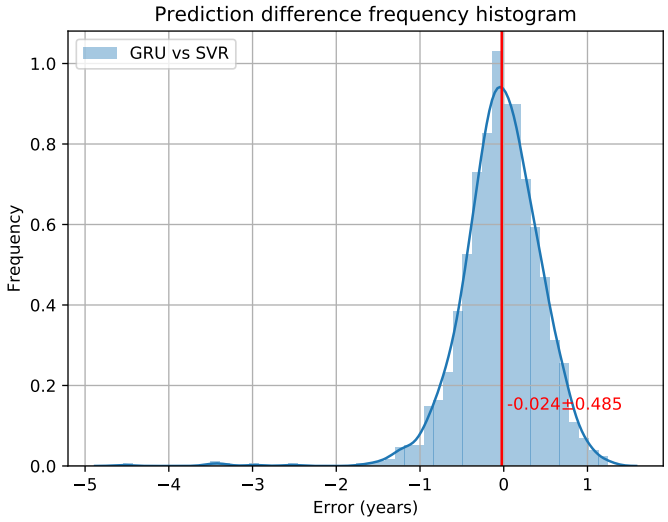

Tableau 5.1 Table of difference GRU vs SVR

Absolute diff(years)	**count**	**count%**	**count% count**
1	1481	96.99	96.99
2	40	2.26	96.61
3	2	0.13	99.74
4	3	0.20	99.94
$\geq$4	1	0.07	100

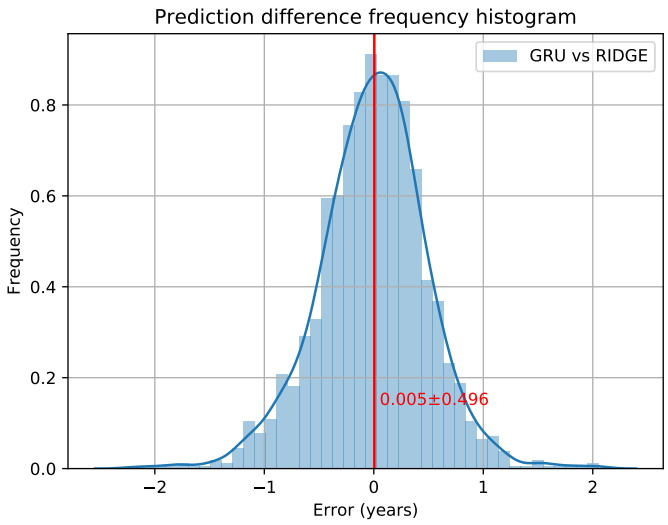

FIGURE 5.1 Difference distribution GRU vs RIDGE

TABLEAU 5.2 Table of difference GRU vs SVR

Absolute diff(years)	**count**	**count%**	**count% count**
1	1452	95.09	95.09
2	71	4.65	99.74
$\geq$2	4	0.26	100

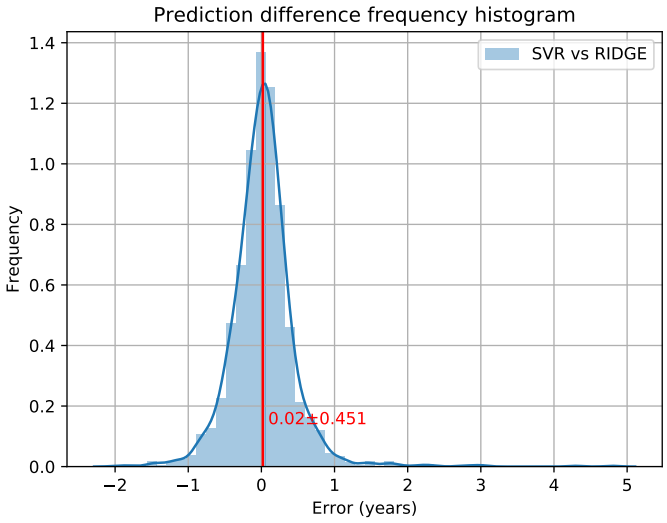

FIGURE 5.2 Difference distribution RIDGE vs SVR

TABLEAU 5.3 Table of difference GRU vs SVR

Absolute diff(years)	**count**	**count%**	**count% count**
1	1483	97.12	97.12
2	36	2.36	99.46
3	5	0.33	99.81
4	1	0.07	99.88
$\geq$4	2	0.13	100

Les résultats ont montré que les trois modèles donnent des résultats similaires, 95% de prédictions ont une différence de moins d'un an.

Même si le réseau de neurones de GRU a obtenu les meilleurs scores dans nos tests, l'écart avec les autres modèles n'est pas si considérable tout particulièrement la régression linéaire. D'une part, c'est possible que la quantité de données limite vraiment le réseau de neurones de GRU, d'autre part, cela signifie peut-être que le modèle Sigma-Lognormal permet une représentation relativement prédictive pour décrire la performance du contrôle neuromoteur. Pour démonter cela, il faudra plus de données. La question de savoir si le réseau de neurones de GRU peut surpasser de beaucoup la régression linéaire demeure ouverte.

5.2 Les limitations

Dans le domaine médical, le manque de données est un phénomène courant. Étant donné que la collecte de grandes quantités de données par nos tests de capacités de contrôle moteur est un processus très coûteux, nous avons pu analyser seulement un peu plus de 700 enfants de 6 ans à 13 ans qui ont participé dans le test. Après l'élimination des cas aberrants, c'est-à-dire en excluant les participants dont les parents avaient rapporté les troubles neurologiques, psychologiques et moteurs existants sur leurs enfants, il ne restait plus que 600 participants à analyser. Vu que ces rapports faits par les parents n'avaient pas été validés, il risquait d'y avoir des sujets anormaux dans notre analyse finale. Par ailleurs, d'une part, on utilisait l'âge comme la cible à prédire pour refléter la maturité du contrôle moteur, mais nous n'avons pas d'informations précises sur la relation entre l'âge et la maturité du contrôle moteur. Tout ce que l'on sait, c'est que le principe de lognormalité prédit que la théorie cinématique, et ses fonctions lognormales décrivent l'humain en parfait contrôle de ses mouvements. Comme corollaire, les jeunes enfants en apprentissage migrent peu à peu vers ce modèle et les personnes âgées s'en éloignent plus ou moins rapidement selon leur état de santé, la lognormalité d'un individu [19, 21].

D'autre part, la performance du contrôle moteur est influencée par l'état général d'un individu, état qui peut varier dans le temps. Par conséquent, cette variance augmente la difficulté et l'incertitude d'interpréter le résultat. Néanmoins, il est possible d'obtenir une évaluation de la maturité du contrôle neuromoteur plus précise en comparant nos résultats avec les autres types de tests comme le diagnostic du médecin, mais cette approche n'est pas économiquement acceptable à grande échelle actuellement. Ce que l'on peut faire c'est de collecter des données auprès de plus de participants dans le futur. La variance d'erreur va s'atténuer surtout pour les modèles de réseau de neurones.

Parce que l'on utilise la validation croisée emboîtée, les hyperparamètres sont cherchés dans les intervalles donnés dans la boucle interne, puis le modèle est testé dans la boucle externe avec ces meilleurs hyperparamètres. La recherche des hyperparamètres optimaux dans la boucle interne est un processus extrême coûteux en temps de calcul. Comme la descente de gradient n'applique pas à l'optimisation des hyperparamètres, il faut penser à l'autre méthode d'optimisation. La recherche de la grille est une solution populaire, mais si le nombre des hyperparamètres à optimiser est élevé ou si les intervalles des hyperparamètres sont vastes, l'espace de recherche sera beaucoup plus grand et donc pas efficace. La recherche aléatoire est une solution plus efficace que la recherche par grille [63]. Mais ces deux approches ne sont pas les plus rapides pour trouver les hyperparamètres. C'est pourquoi nous avons utilisé TPE (Tree-structured Parzen Estimator) pour chercher les hyperparamètres [64]. C'est un

algorithme basé sur l'optimisation bayésienne et le modèle de mélange gaussien. Il s'agit de construire un modèle de probabilité et l'utiliser pour sélectionner les hyperparamètres les plus prometteurs à évaluer. Néanmoins, quand le modèle a beaucoup d'hyperparamètres à optimiser, cet algorithme a une performance comparable à celle d'une recherche aléatoire. Dans notre configuration, l'algorithme de TPE est itéré 25 fois dans chaque boucle. Il est possible que le nombre d'itérations ne soit pas suffisant pour certains modèles qui ont beaucoup d'hyperparamètres comme les modèles du réseau de neurones. Mais à cause de la limite temporelle, 25 répétitions nous sont apparues comme un bon compromis. Certains hyperparamètres du réseau de neurones comme le nombre de couches, le nombre de neurones, changent directement la structure du modèle et cela augmente aussi la difficulté d'optimiser les hyperparamètres. Par exemple, dans notre modèle de GRU, une fonction d'erreur auxiliaire est introduite. Le coefficient de cette fonction d'erreur auxiliaire est fixé directement à 0.01, car ce coefficient est vraiment critique, un petit changement peut influencer totalement le modèle voire l'échec de convergence de l'apprentissage. Par ailleurs, il existe d'autres moyens d'équilibrer les multiples fonctions d'erreur en utilisant l'optimisation de Pareto [65] au lieu de donner un coefficient fixe. Finalement, lorsque nous aurons assez de données, la validation croisée emboîtée ne sera plus nécessaire. Il suffira d'appliquer la validation non-croisée ("hold-out method" en anglais), c'est-à-dire, nous divisons les données par trois parties, "apprentissage", "validation", "test". L'ajustement des hyperparamètres est fait sur les données de validation et la performance est mesurée avec les données de test.

5.3 Recommandations

Taichi Sumi et al ont proposé [61] un modèle appelé Cross-VAE (l'auto-encodeur variationnel) est capable de convertir une image de caractères manuscrits en sa trajectoire temporelle originale et vice versa. Il est composé par deux VAE, dont les espaces latents sont partagés entre-deux. Chaque VAE s'occupe un type de données : soit l'image de caractères manuscrits, soit la trajectoire temporelle originale. Une fonction d'erreur qui mesure la divergence entre les deux espaces latents est introduite pour encourager le modèle à partager les deux modalités. La Théorie Cinématique avec le modèle Sigma-Lognormal est capable d'encoder les informations de la trajectoire temporelle originale dans l'espace latent comme des paramètres Sigma-Lognormal, mais les données des traits manuscrits ne sont pas encore utilisées. Le modèle Cross-VAE nous laisse croire que l'utilisation de ces deux modalités puisse augmenter encore la performance.

CHAPITRE 6 CONCLUSION

6.1 Conclusion

L'objectif de ce mémoire était de présenter une méthode pour corréler l'âge des enfants en bonne santé avec les paramètres Sigma-Lognormaux pour aider à évaluer le développement du contrôle de leur motricité fine. Cette méthode d'évaluation devait être rapide, pas coûteuse, utilisable en clinique. Le protocole nécessite des traits manuscrits sur une tablette à numériser et en utilisant la Théorie Cinmématique des mouvements humains rapides, les paramètres Sigma-Lognormaux sont extraits. Ces paramètres reflètent la performance du contrôle de la motricité fine des participants et ils ont une corrélation significative avec l'âge du participant, en accord avec le principe de lognormalité [13,14]. Nous avons comparé des modèles différents pour cette régression. Les modèles par réseau de neurones de GRU et la régression par vecteur de support donnent les meilleurs résultats et leurs performances sont similaires. En plus, les expériences montrent que le test des traits simples est moins concluant que les tests de tracés triangulaires. Les modèles par réseau de neurones ont démontré leurs capacités de traiter les données de structures temporelles complexes, tandis que les modèles symbolistes, par exemple l'arbre de décisions, avaient plus de difficultés à gérer les structures de données complexes.

En conclusion, l'application de ces travaux devraient permettre un jour d'aider les cliniciens à faire des pré-dépistages rapides avant de l'évaluation formelle du contrôle moteur.

RÉFÉRENCES

[1] A. Cancer, R. Minoliti, M. Crepaldi, and A. Antonietti, "Identifying Developmental Motor Difficulties : A Review of Tests to Assess Motor Coordination in Children," *Journal of Functional Morphology and Kinesiology*, vol. 5, no. 1, p. 16, Feb. 2020.

[2] A. Fahr, J. W. Keller, J. Balzer, J. Lieber, and H. J. van Hedel, "Quantifying age-related differences in selective voluntary motor control in children and adolescents with three assessments," *Human Movement Science*, vol. 77, p. 102790, Jun. 2021.

[3] R. Plamondon, "A Kinematic Theory of rapid human movements : Part I. Movement representation and generation," *Biological Cybernetics*, vol. 72, no. 4, pp. 295–307, Mar. 1995.

[4] R. Plamondon, "A Kinematic Theory of rapid human movements. Part II. Movement time and control," *Biological cybernetics*, vol. 72, pp. 309–20, Feb. 1995.

[5] R. Plamondon, "A Kinematic Theory of rapid human movements : Part III. Kinetic outcomes," *Biological Cybernetics*, vol. 78, no. 2, pp. 133–145, Feb. 1998.

[6] R. Plamondon, C. Feng, and A. Woch, "A Kinematic Theory of rapid human movement. Part IV : a formal mathematical proof and new insights," *Biological Cybernetics*, vol. 89, no. 2, pp. 126–138, Aug. 2003.

[7] C. O'Reilly and R. Plamondon, "Development of a Sigma–Lognormal representation for on-line signatures," *Pattern Recognition*, vol. 42, no. 12, pp. 3324–3337, Dec. 2009.

[8] T. Duval, C. Rémi, R. Plamondon, J. Vaillant, and C. O'Reilly, "Combining sigma-lognormal modeling and classical features for analyzing graphomotor performances in kindergarten children," *Human Movement Science*, vol. 43, pp. 183–200, Oct. 2015.

[9] R. Plamondon, C. O'Reilly, and C. Ouellet-Plamondon, "Strokes against stroke—strokes for strides," *Pattern Recognition*, vol. 47, no. 3, pp. 929–944, Mar. 2014.

[10] P. Laniel, N. Faci, R. Plamondon, M. H. Beauchamp, and B. Gauthier, "Kinematic analysis of fast pen strokes in children with ADHD," *Applied Neuropsychology : Child*, vol. 9, no. 2, pp. 125–140, Apr. 2020.

[11] C. Scheuer, C. Herrmann, and A. Bund, "Motor tests for primary school aged children : A systematic review," *Journal of Sports Sciences*, vol. 37, no. 10, pp. 1097–1112, May 2019.

[12] D. Bullock and S. Grossberg, "Neural dynamics of planned arm movements : Emergent invariants and speed-accuracy properties during trajectory formation."

Psychological Review, vol. 95, no. 1, pp. 49–90, 1988. [Online]. Available : http://doi.apa.org/getdoi.cfm?doi=10.1037/0033-295X.95.1.49

[13] R. A. Schmidt and T. D. Lee, *Motor control and learning : A behavioral emphasis, 3rd ed.*, ser. Motor control and learning : A behavioral emphasis, 3rd ed. Champaign, IL, US : Human Kinetics, 1999, pages : xvi, 493.

[14] A. Mohebbi, P. Amiri, and R. E. Kearney, "Contributions of Vision in Human Postural Control : A Virtual Reality-based Study," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. Montreal, QC, Canada : IEEE, Jul. 2020, pp. 3347–3350. [Online]. Available : https://ieeexplore.ieee.org/document/9175605/

[15] C. Bosecker, L. Dipietro, B. Volpe, and H. Igo Krebs, "Kinematic Robot-Based Evaluation Scales and Clinical Counterparts to Measure Upper Limb Motor Performance in Patients With Chronic Stroke," *Neurorehabilitation and Neural Repair*, vol. 24, no. 1, pp. 62–69, Jan. 2010. [Online]. Available : http://journals.sagepub.com/doi/10.1177/1545968309343214

[16] L. Zollo, L. Rossini, M. Bravi, G. Magrone, S. Sterzi, and E. Guglielmelli, "Quantitative evaluation of upper-limb motor control in robot-aided rehabilitation," *Medical & Biological Engineering & Computing*, vol. 49, no. 10, pp. 1131–1144, Oct. 2011. [Online]. Available : http://link.springer.com/10.1007/s11517-011-0808-1

[17] M. Gilliaux, T. Lejeune, C. Detrembleur, J. Sapin, B. Dehez, C. Selves, and G. Stoquart, "Using the robotic device REAplan as a valid, reliable, and sensitive tool to quantify upper limb impairments in stroke patients," *Journal of Rehabilitation Medicine*, vol. 46, no. 2, pp. 117–125, 2014. [Online]. Available : http://www.medicaljournals.se/jrm/content/?doi=10.2340/16501977-1245

[18] R. Plamondon and M. Djioua, "A multi-level representation paradigm for handwriting stroke generation," *Human Movement Science*, vol. 25, no. 4-5, pp. 586–607, Oct. 2006.

[19] R. Plamondon, C. O'Reilly, C. Rémi, and T. Duval, "The lognormal handwriter : learning, performing, and declining," *Frontiers in Psychology*, vol. 4, 2013.

[20] R. Plamondon, "THE CONVERGENCE OF A NEUROMUSCULAR IPULSE RESPONSE TOWARDS A LOGNORMAL, FROM THEORY TO PRACTICE," p. 45.

[21] R. Plamondon, "The Lognormality Principle : A Personalized Survey," in *Series in Machine Perception and Artificial Intelligence*. WORLD SCIENTIFIC, Dec. 2020, vol. 88, pp. 1–39.

[22] C. O'Reilly, R. Plamondon, and L.-H. Lebrun, "Linking brain stroke risk factors to human movement features for the development of preventive tools," *Frontiers in Aging Neuroscience*, vol. 6, Jul. 2014.

[23] N. Faci, S. P. Boyogueno Bidias, R. Plamondon, and N. Bergeron, "An Interactive Tablet-based System to Run Neuromuscular Tests," in *The Lognormality Principle and its Applications in e-Security, e-Learning and e-Health*, ser. Series in Machine Perception and Artificial Intelligence. WORLD SCIENTIFIC, Jul. 2020, vol. Volume 88, no. Volume 88, pp. 269–288.

[24] N. Faci, S. P. B. Bidias, R. Plamondon, and N. Bergeron, "A New Experimental Set-up To Run Neuromuscular Tests," p. 5.

[25] U. Bhattacharya, R. Plamondon, S. Dutta Chowdhury, P. Goyal, and S. K. Parui, "A sigma-lognormal model-based approach to generating large synthetic online handwriting sample databases," *International Journal on Document Analysis and Recognition (IJDAR)*, vol. 20, no. 3, pp. 155–171, Sep. 2017.

[26] L. A. Leiva, M. Diaz, M. A. Ferrer, and R. Plamondon, "Human or Machine ? It Is Not What You Write, But How You Write It," in *2020 25th International Conference on Pattern Recognition (ICPR)*. Milan, Italy : IEEE, Jan. 2021, pp. 2612–2619.

[27] C. Ramaiah, R. Plamondonm, and V. Govindaraju, "A Sigma-Lognormal Model for Handwritten Text CAPTCHA Generation," in *2014 22nd International Conference on Pattern Recognition*. Stockholm, Sweden : IEEE, Aug. 2014, pp. 250–255.

[28] A. Laurent, R. Plamondon, and M. Begon, "Central and Peripheral Shoulder Fatigue Pre-screening Using the Sigma–Lognormal Model : A Proof of Concept," *Frontiers in Human Neuroscience*, vol. 14, p. 171, May 2020.

[29] A. Laurent, R. Plamondon, and M. Begon, "Reliability of the kinematic theory parameters during handwriting tasks on a vertical setup," *Biomedical Signal Processing and Control*, vol. 71, p. 103157, Jan. 2022.

[30] G. Ian, B. Yoshua, and C. Aaron, *Deep Learning*, ser. Regularization for Deep Learning. MIT Press, 2016, no. 7. [Online]. Available : http://www.deeplearningbook.org

[31] P. J. Huber, "Robust regression : asymptotics, conjectures and Monte Carlo," *The annals of statistics*, pp. 799–821, 1973, iSBN : 0090-5364 Publisher : JSTOR.

[32] A. J. Smola and B. Schölkopf, "A tutorial on support vector regression," *Statistics and Computing*, vol. 14, no. 3, pp. 199–222, Aug. 2004.

[33] K. Taunk, S. De, S. Verma, and A. Swetapadma, "A Brief Review of Nearest Neighbor Algorithm for Learning and Classification," in *2019 International Conference on*

Intelligent Computing and Control Systems (ICCS). Madurai, India : IEEE, May 2019, pp. 1255–1260. [Online]. Available : https://ieeexplore.ieee.org/document/9065747/

[34] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, Oct. 2001.

[35] A. Natekin and A. Knoll, "Gradient boosting machines, a tutorial," *Frontiers in Neurorobotics*, vol. 7, 2013. [Online]. Available : http://journal.frontiersin.org/article/10.3389/fnbot.2013.00021/abstract

[36] E. R. David, E. H. Geoffrey, and J. W. Ronald, "Learning representations by back-propagating errors," vol. 323, pp. 533–536, 1986.

[37] V. Nair and G. E. Hinton, "Rectified Linear Units Improve Restricted Boltzmann Machines," p. 8.

[38] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, Nov. 1997. [Online]. Available : https://direct.mit.edu/neco/article/9/8/1735-1780/6109

[39] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Gated Feedback Recurrent Neural Networks," *arXiv :1502.02367 [cs, stat]*, Jun. 2015, arXiv : 1502.02367.

[40] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," *arXiv :1312.6114 [cs, stat]*, May 2014, arXiv : 1312.6114.

[41] Q. Zhao, E. Adeli, N. Honnorat, T. Leng, and K. M. Pohl, "Variational AutoEncoder For Regression : Application to Brain Aging Analysis," *arXiv :1904.05948 [cs, stat]*, Jul. 2019, arXiv : 1904.05948. [Online]. Available : http://arxiv.org/abs/1904.05948

[42] Z. Pan, S. Talwar, R. Plamondon, and A. W. Van Gemmert, "Characteristics of bi-directional unimanual and bimanual drawing movements : The application of the Delta-Lognormal models and Sigma-Lognormal model," *Pattern Recognition Letters*, vol. 121, pp. 97–103, Apr. 2019. [Online]. Available : https://linkinghub.elsevier.com/retrieve/pii/S0167865518301740

[43] L. Liutsko, R. Muiños, J. M. Tous Ral, and M. J. Contreras, "Fine Motor Precision Tasks : Sex Differences in Performance with and without Visual Guidance across Different Age Groups," *Behavioral Sciences*, vol. 10, no. 1, p. 36, Jan. 2020. [Online]. Available : https://www.mdpi.com/2076-328X/10/1/36

[44] C. O'Reilly and R. Plamondon, "Prototype-Based Methodology for the Statistical Analysis of Local Features in Stereotypical Handwriting Tasks," in *2010 20th International Conference on Pattern Recognition.* Istanbul, Turkey : IEEE, Aug. 2010, pp. 1864–1867.

[45] "The Lognormality Principle and its Applications in e-Security, e-Learning and e-Health," ser. Series in Machine Perception and Artificial Intelligence.

WORLD SCIENTIFIC, Jul. 2020, vol. Volume 88, p. 425. [Online]. Available : https://doi.org/10.1142/12006

[46] M. Hosseini, M. Powell, J. Collins, C. Callahan-Flintoft, W. Jones, H. Bowman, and B. Wyble, "I tried a bunch of things : The dangers of unexpected overfitting in classification of brain data," *Neuroscience & Biobehavioral Reviews*, vol. 119, pp. 456–467, Dec. 2020.

[47] M. Stone, "Cross-Validatory Choice and Assessment of Statistical Predictions," *Journal of the Royal Statistical Society : Series B (Methodological)*, vol. 36, no. 2, pp. 111–133, Jan. 1974.

[48] I. Loshchilov and F. Hutter, "DECOUPLED WEIGHT DECAY REGULARIZATION," p. 18, 2019.

[49] D. P. Kingma and J. Ba, "Adam : A Method for Stochastic Optimization," *arXiv :1412.6980 [cs]*, Jan. 2017, arXiv : 1412.6980.

[50] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout : A Simple Way to Prevent Neural Networks from Overfitting," p. 30.

[51] S. J. Hanson and L. Y. Pratt, "Comparing Biases for Minimal Network Construction with Back-Propagation," p. 9.

[52] H. Drucker, C. J. C. Burges, L. Kaufman, A. Smola, and V. Vapnik, "Support Vector Regression Machines," in *Advances in Neural Information Processing Systems*, M. C. Mozer, M. Jordan, and T. Petsche, Eds., vol. 9. MIT Press, 1997.

[53] T. Chen and C. Guestrin, "XGBoost : A Scalable Tree Boosting System," *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 785–794, Aug. 2016, arXiv : 1603.02754.

[54] T. M. Cover and P. E. Hart, "Nearest neighbor pattern classification," *IEEE Trans. Inf. Theory*, vol. 13, pp. 21–27, 1967.

[55] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE) ? – Arguments against avoiding RMSE in the literature," *Geoscientific Model Development*, vol. 7, no. 3, pp. 1247–1250, Jun. 2014.

[56] J. Heaton, "An Empirical Analysis of Feature Engineering for Predictive Modeling," *SoutheastCon 2016*, pp. 1–6, Mar. 2016, arXiv : 1701.07852.

[57] G. Zerveas, S. Jayaraman, D. Patel, A. Bhamidipaty, and C. Eickhoff, "A Transformer-based Framework for Multivariate Time Series Representation Learning," in *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*. Virtual Event Singapore : ACM, Aug. 2021, pp. 2114–2124.

[58] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "-VAE : LEARNING BASIC VISUAL CONCEPTS WITH A CONSTRAINED VARIATIONAL FRAMEWORK," p. 22, 2017.

[59] T. Q. Chen, X. Li, R. Grosse, and D. Duvenaud, "ISOLATING SOURCES OF DISENTANGLEMENT IN VARIATIONAL AUTOENCODERS," p. 18, 2018.

[60] K. Sohn, H. Lee, and X. Yan, "Learning Structured Output Representation using Deep Conditional Generative Models," p. 9.

[61] T. Sumi, B. K. Iwana, H. Hayashi, and S. Uchida, "Modality Conversion of Handwritten Patterns by Cross Variational Autoencoders," *arXiv :1906.06142 [cs]*, Jun. 2019, arXiv : 1906.06142.

[62] P. Domingos, "Every Model Learned by Gradient Descent Is Approximately a Kernel Machine," *arXiv :2012.00152 [cs, stat]*, Nov. 2020, arXiv : 2012.00152. [Online]. Available : http://arxiv.org/abs/2012.00152

[63] J. Bergstra and Y. Bengio, "Random Search for Hyper-Parameter Optimization," p. 25.

[64] J. S. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for Hyper-Parameter Optimization," p. 9.

[65] P. Ma, T. Du, and W. Matusik, "Efficient Continuous Pareto Exploration in Multi-Task Learning," *arXiv :2006.16434 [cs, stat]*, Aug. 2020, arXiv : 2006.16434. [Online]. Available : http://arxiv.org/abs/2006.16434