
Aplicação de Redes Neurais Recorrentes para a Predição do Aprendizado Motor

Lucas Cléopas Costa da Silva



UNIVERSIDADE FEDERAL DO MARANHÃO
CENTRO DE CIÊNCIAS EXATAS E TECNOLOGIA
COORDENAÇÃO DO CURSO DE ENGENHARIA DA COMPUTAÇÃO

São Luís
2022

Lucas Cléopas Costa da Silva

**Aplicação de Redes Neurais Recorrentes para a
Predição do Aprendizado Motor**

Monografia apresentada à Coordenação do Curso de Engenharia da Computação da Universidade Federal do Maranhão como parte dos requisitos para a obtenção do título de Bacharel em Engenharia da Computação.

Área de concentração: Engenharia da Computação

Orientador: Prof. Dr. Paulo Rogério de Almeida Ribeiro

São Luís
2022

Agradecimentos

Sou profundamente grato a toda a minha família por tudo o que fizeram por mim até aqui. Minha mãe, Edilene Costa da Silva, por nunca deixar de acreditar, apoiar e festejar cada conquista minha. Meu pai, José Arnaldo da Silva, por nunca me deixar cair em estagnação. Meu irmão, Thiago Sisaque, por sempre me acompanhar de perto e me apoiar quando precisei. Minha companheira, Edilene Monteiro, por estar comigo nos bons e maus momentos. Amo todos vocês!

Ao meu orientador Prof. Dr. Paulo Rogério de Almeida Ribeiro, pela paciência comigo durante todos esses anos. Você foi a inspiração que me trouxe para a Computação em 2017. Obrigado por compartilhar suas experiências e conhecimentos comigo. Eu te admiro muito!

A todos os amigos que fiz durante minha caminhada acadêmica dentro e fora da Universidade Federal do Maranhão (UFMA). Todos vocês fizeram parte disso. Especialmente, Alexandre Wesley, Aridson Nunes, Billyfranklim Pereira, Gleison Rocha, Guilherme Cabral, Jean Charles, Marcelo Marques, Regyvanderson Moura, Victor Valentim e Wanderson Monteiro.

A todos os professores do curso de Engenharia da Computação, que me deram os conhecimentos necessários para eu desenvolver tudo o que estava na minha cabeça. Vocês foram os melhores professores que tive!

À UFMA e ao grupo de pesquisa do LACMOR, que foram meu segundo lar durante esses anos. À FAPEMA, que me deu apoio financeiro durante este trabalho. Ao Moisés Rocha dos Santos, que sanou muitas dúvidas que tive e contribuiu na remoção de dados discrepantes.

Por fim, agradeço a todos que, direta ou indiretamente, contribuíram na minha formação pessoal, acadêmica e profissional. Vocês me ajudaram a ser quem eu sempre quis ser: um cientista/engenheiro!

“Sempre fui sonhador! É isso que me mantém vivo.”
(A Vida É Desafio – Racionais MC’s, Afro-X)

Resumo

Tarefas do dia a dia – por exemplo, andar, escrever, digitar, etc. – envolvem habilidades motoras. No entanto, pacientes com problemas motores apresentam dificuldades para executar essas tarefas rotineiras. Tendo em vista esse cenário, estudos com ênfase em aprendizado motor para aquisição de habilidades motoras e reabilitação dessas habilidades mostram-se necessários para melhorar a qualidade de vida dos pacientes. O objetivo principal deste trabalho visou aplicar Rede Neural Recorrente para estimar quando um indivíduo irá atingir o nível de desempenho desejado dado uma tarefa de aprendizado motor. Em razão disso, 20 voluntários realizaram a tarefa de decalque com a mão não dominante (mão esquerda) para o estudo e predição do aprendizado motor. Essa tarefa foi realizada durante três dias consecutivos (cada dia é uma sessão). Cada sessão continha dez blocos com dez palavras (tentativas) a serem traçadas. Após a coleta, analisou-se o comportamento desses dados por meio de gráficos e análises estatísticas. Utilizou-se a técnica de Agrupamento Hierárquico e intervalo interquartil (IQR) para detecção e eliminação de valores discrepantes (*outliers*) que geram ruído aos dados. Por fim, treinou-se um modelo baseado em Rede Neural Recorrente para prever o erro médio da terceira sessão. Os resultados foram satisfatórios, uma vez que a forma do erro foi predita e a taxa de erro relativa foi de 78,12%.

Palavras-chave: Aprendizado motor. Aprendizado de máquina. Agrupamento hierárquico. Rede Neural Recorrente.

Abstract

Everyday tasks – for example, walking, writing, typing, etc. – involve motor skills. However, patients with motor diseases have difficulty performing these routine tasks. In view of this scenario, studies with an emphasis on motor learning for the acquisition of motor skills and rehabilitation of these skills are necessary to improve the quality of life of patients. The main purpose of this work was to apply Recurrent Neural Network to estimate when an individual will reach the desired performance level given a motor learning task. With that in mind, 20 volunteers performed the tracing task with the non-dominant hand (left hand) for the study and prediction of motor learning. This task was performed for three consecutive days (each day is a session). Each session contained ten blocks with ten words (trials) to be traced. After data collection, the behavior of these data was analyzed using graphs and statistical analysis. The technique of Hierarchical Clustering and interquartile range (IQR) was used to detect and remove outliers that generate noise in the data. Finally, a model based on Recurrent Neural Network was trained to predict the average error of the third session. The results were satisfactory, since the error form was predicted and the relative error rate was 78.12%.

Keywords: Machine learning. Motor learning. Clustering analysis. Recurrent Neural Network.

Lista de ilustrações

Figura 1 – <i>Boxplot</i> , ou diagrama de caixa.	24
Figura 2 – Representação gráfica de uma rede perceptron.	25
Figura 3 – Representação gráfica de uma rede Perceptron Multi-camadas. Essa rede contém 4 camadas, sendo: 1 camada de entrada com 3 unidades, 2 camadas ocultas com 3 unidades cada e 1 camada de saída com 2 unidades.	26
Figura 4 – Representação gráfica de uma RNN desenrolada (LE, 2019).	27
Figura 5 – Representação de uma trial no <i>software</i> da tarefa de decalque. Como indicado pela seta vermelha 1, o participante utiliza a mesa digitalizadora com a mão esquerda para tentar desenhar a palavra que aparece na tela. A palavra desenhada e a palavra modelo, indicados pela seta 2 verde, são borradas com um <i>kernel</i> gaussiano para a extração da métrica de cada trial (seta 3 amarela).	30
Figura 6 – Erro médio de todos os blocos de cada sessão. (a) Grupo com os outliers id11 e id12. Todos os dados entre as sessões são normais ($p = 0.08$, $p = 0.25$, $p = 0.49$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (ANOVA) apenas entre as sessões 1 e 2 (Tukey HSD). (b) Grupo sem id11. Os dados entre as sessões não são normais ($p = 0.00$, $p = 0.03$, $p = 0.02$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (Kruskal-Wallis) apenas entre as sessões 1 e 2 (Wilcoxon). (c) Grupo sem id12. Todos os dados entre as sessões são normais ($p = 0.17$, $p = 0.37$, $p = 0.08$ para a sessão 1, 2 e 3, respectivamente); não há diferença estatística (ANOVA) entre as sessões. (d) O grupo sem id11 e id12. Apenas os dados da sessão 2 são normais ($p = 0.00$, $p = 0.15$, $p = 0.00$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (Kruskal-Wallis) apenas entre as sessões 1 e 3 (Wilcoxon).	36

Figura 7 – Erro médio de cada bloco. (a) Grupo com outliers id11 e id12. Há diferença estatística (teste de Wilcox) entre os blocos 1 e 10 apenas na sessão 3. (b) Grupo sem id11. Existe uma diferença estatística (Teste t) entre os blocos 1 e 10 na sessão 1, e existe diferença estatística (teste de Wilcox) entre os blocos 1 e 10 na sessão 3. (c) Grupo sem id12. Há diferença estatística (teste de Wilcox) entre os blocos 1 e 10 nas sessões 2 e 3. (d) Grupo sem id11 e id12. Existe uma diferença estatística (Teste t) entre os blocos 1 e 10 na sessão 1, e existe diferença estatística (teste de Wilcox) entre os blocos 1 e 10 na sessão 3.	37
Figura 8 – Agrupamento Hierárquico Aglomerativo com dois grupos: C1 e C2. As linhas azuis representam os membros do grupo C1 e a linha vermelha representa o membro do grupo C2.	38
Figura 9 – Grupo C2 do dendrograma na Figura 8 (outlier id11). (a) Erro médio de todos os blocos de cada sessão; os dados são normais ($p = 0.13$, $p = 0.86$, $p = 0.36$ para a sessão 1, 2 e 3, respectivamente) e há diferença estatística (ANOVA) apenas entre as sessões 1 e 2 (Tukey HSD). (b) Erro médio de cada bloco; há diferença estatística (Teste t) entre os blocos 1 e 10 apenas na sessão 1.	38
Figura 10 – Agrupamento Hierárquico Aglomerativo com três grupos: C1, C2 e C3. As linhas azuis representam os membros do grupo C1, a linha vermelha representa o membro do grupo C2 e a linha verde representa o membro do grupo C3.	39
Figura 11 – Grupo C2 do dendrograma na Figura 10 (outlier id12). (a) Erro médio de todos os blocos de cada sessão; apenas os dados da sessão 2 não são normais ($p = 0.44$, $p = 0.03$, $p = 0.98$ para a sessão 1, 2 e 3, respectivamente) e há diferença estatística (Kruskal-Wallis) entre as sessões 1 e 2, e, entre as sessões 1 e 3 (Wilcoxon). (b) Erro médio de cada bloco; não há diferença estatística (Teste t e Wilcoxon) entre os blocos 1 e 10 de cada sessão.	39
Figura 12 – Três melhores predições feitas pelo modelo da predição do erro da terceira sessão. A linha em azul representa o erro real do indivíduo ao longo dos blocos. A linha laranja é o valor predito do erro do participante na terceira sessão, ou seja, os últimos 9 blocos. Estas são as três melhores predições feitas pelo modelo.	41

Lista de tabelas

Tabela 1 – Descrição das camadas contidas no modelo <i>Long Short-Term Memory</i> (LSTM) criado. Total de 524.457 parâmetros treináveis.	32
Tabela 2 – Dados de voluntários convocados para a pesquisa. Esses dados estão ordenados em ordem decrescente pelo valor do QL.	35
Tabela 3 – Resultado do modelo LSTM nas métricas de regressão.	40

Lista de siglas

ANOVA *Análise de Variância*

AD *Árvores de Decisão*

BDI *Beck Depression Inventory*

IQR *Intervalo Interquartil*

KNN *K-Vizinhos Mais Próximos*

LI *Limite Inferior*

LS *Limite Superior*

LSTM *Long Short-Term Memory*

MAE *Mean Absolute Error*

MAPE *Mean Absolute Percentage Error*

MLP *Multi-layer Perceptron*

QL Quociente de Lateralidade

RNA Rede Neural Artificial

RNN Rede Neural Recorrente

RMSE *Root Mean Squared Error*

SVM Máquinas de Vetores de Suporte

UFMA Universidade Federal do Maranhão

Sumário

1	INTRODUÇÃO	19
1.1	Justificativa	20
1.2	Objetivos	20
1.3	Estrutura do Trabalho	21
2	FUNDAMENTAÇÃO TEÓRICA	23
2.1	Aprendizado Motor	23
2.2	Análise de Dados	23
2.2.1	Intervalo Interquartil	23
2.2.2	Agrupamento Hierárquico	24
2.3	Redes Neurais Artificiais	24
2.3.1	Rede Perceptron	25
2.3.2	Rede Perceptron Multi-camadas	25
2.3.3	Redes Neurais Recorrentes	26
2.4	Trabalhos Relacionados	27
3	METODOLOGIA	29
3.1	Participantes e Design do Experimento	29
3.2	Remoção de <i>Outliers</i>	30
3.3	Análise Estatística	31
3.4	Pré-processamento dos Dados	31
3.5	Definição do Modelo LSTM	31
3.6	Avaliação do Modelo	32
3.6.1	Erro Absoluto Médio	32
3.6.2	Raiz do Erro Quadrático Médio	33
3.6.3	Erro Percentual Médio Absoluto	33
4	RESULTADOS	35

4.1	Análise dos Dados	35
4.2	Previsão do Erro da Terceira Sessão	39
5	CONCLUSÃO	43
	REFERÊNCIAS	45

Introdução

As habilidades motoras são de extrema importância para realização de tarefas básicas e diárias das pessoas, por exemplo, andar, beber água, escovar os dentes entre outras. No entanto, enquanto essas tarefas são triviais para as pessoas saudáveis, uma parte da população, isto é, pacientes com problemas motores, não as pode realizar.

Pessoas com algum tipo de deficiência motora representam uma quantidade considerável da população brasileira. Conforme a *Cartilha do Censo publicada em 2012 sobre Pessoas com Deficiência*¹, 7% da população brasileira possui algum tipo de deficiência motora, variando entre lesões traumáticas, como derrame ou paralisia cerebral, e doenças congênitas. Esses tipos de traumas deixam sequelas e tornam a vida dos pacientes mais complicada. Tendo em vista esse cenário, estudos com ênfase em aprendizado motor para aquisição de habilidades motoras e reabilitação dessas habilidades mostram-se necessários para melhorar a qualidade de vida dos pacientes que sofrem essas sequelas. Diversos trabalhos de pesquisa (KITAGO; KRAKAUER, 2013; PRICHARD et al., 2014; NASEER; HONG, 2015; ASTRAND; WARDAK; HAMED, 2014; SHMUELOF; KRAKAUER, 2014) estudam o processo de aquisição de habilidade motora, assim como a reabilitação motora.

As atividades motoras são a principal forma de comunicação com o mundo (PRICHARD et al., 2014), por exemplo, falar, movimentar os olhos, gesticular, etc. Desta forma, caso uma pessoa saudável seja acometida com alguma doença que afete essas atividades - por exemplo, uma doença neurológica - essa deve adquirir uma nova habilidade motora que busque novas formas de ativar a musculatura para realizar a tarefa com o melhor desempenho possível (KITAGO; KRAKAUER, 2013).

Pesquisas nessa área podem melhorar a vida de pacientes. Entretanto, embora a literatura sobre o tema seja numerosa, ainda há uma lacuna sobre a predição da aquisição da habilidade motora. A predição da quantidade de sessões ou blocos de treinamento motor que um indivíduo necessita realizar para adquirir um determinado desempenho em

¹ <http://www.pessoacomdeficiencia.gov.br/app/sites/default/files/publicacoes/cartilha-censo-2010-pessoas-comdeficiencia-reduzido.pdf>

uma tarefa motora é um exemplo de estudo a ser explorado.

O problema de estimar um conjunto de valores de um conjunto de dados pode ser visto como um problema de Aprendizado de Máquina (*Machine Learning*). Existem diversas técnicas dessa área que podem ser utilizadas para esse problema, por exemplo, Rede Neural Artificial (RNA), Algoritmos de Regressão entre outras.

De acordo com (SAMUEL, 1959), o Aprendizado de Máquina é definido como o campo de estudo que dá aos computadores a capacidade de aprender sem serem explicitamente programados. Uma quantidade massiva de dados vem sendo gerada dia após dia e necessita-se de aplicações que processam e geram informações para auxiliar humanos em tomadas de decisões (MAHESH, 2020). Esses algoritmos são treinados a partir de dados coletados e processados conforme o objetivo da tarefa. Dessa forma, é possível fazer predições, como: classificação de imagens, previsão de preços, reconhecimento de padrões, sistemas de recomendação, etc.

1.1 Justificativa

O sistema motor humano tem a capacidade de aprender com a prática e a experiência, sendo que o tempo necessário para esse aprendizado depende da complexidade da tarefa motora (KITAGO; KRAKAUER, 2013).

Uma das técnicas que tem sido largamente utilizada para intensificar e melhorar o aprendizado motor é a estimulação cerebral elétrica não-invasiva, por exemplo, o trabalho de Prichard et al. (2014) investiga o efeito da estimulação elétrica no cérebro para o aprendizado motor. Esse utiliza a *Tracing Task*, isto é uma tarefa de decalque de palavras.

No entanto, pode-se investigar também – por exemplo, com algoritmos de aprendizado de máquina – uma possível predição de quando esse aprendizado motor ocorrerá. Algoritmos de predição vêm sendo aplicados com sucesso em diversas áreas, tais como: saúde, educação, economia, ciência, tecnologia e logística (JORDAN; MITCHELL, 2015).

Assim sendo, este trabalho visou estimar o tempo de treinamento necessário para atingir um determinado nível de precisão na execução de uma atividade motora. Desta forma, pode-se almejar um aprendizado motor personalizado, assim como futuramente aplicar esse modelo para reabilitação personalizada de pacientes. Ou seja, um paciente poderá informar inicialmente o nível de desempenho que deseja obter na tarefa e ter uma estimativa de quando obterá esse objetivo.

1.2 Objetivos

O objetivo principal deste trabalho visa aplicar o algoritmo de Aprendizado de Máquina Rede Neural Recorrente (RNN) para estimar quando um participante irá atingir o nível de desempenho desejado numa tarefa de aprendizado motor.

Como objetivos específicos têm-se:

- ❑ Observar as propriedades do aprendizado motor em participantes saudáveis;
- ❑ Detecção automática de dados discrepantes no conjunto de participantes;
- ❑ Estimar quando um participante irá obter um determinado erro na tarefa de aprendizado motor;
- ❑ Fomentar pesquisas interdisciplinares, nomeadamente área de Engenharia com a área da Saúde.

1.3 Estrutura do Trabalho

Este trabalho está organizado da seguinte forma: no Capítulo 2, descreve-se a literatura atual a respeito do tema proposto; no Capítulo 3, mostra-se a metodologia que será utilizada para realização dos experimentos; no Capítulo 4, apresentam-se os resultados e no Capítulo 5, a conclusão deste trabalho.

Fundamentação Teórica

2.1 Aprendizado Motor

O aprendizado motor é um termo que abrange uma diversidade de fenômenos, abordagens e disciplinas. No campo da psicologia e neurociência, aprendizado motor é definido como qualquer melhoria de desempenho dependente da experiência (KRAKAUER et al., 2019). O sistema motor consegue aperfeiçoa-se com a repetição e experiência em uma determinada tarefa ao longo das execuções.

Krakauer et al. (2019) dividem o aprendizado motor em duas partes: i) aquisição de habilidades - o processo onde o indivíduo adquire a habilidade necessária para a execução de uma tarefa específica com exatidão e precisão; e ii) manutenção de habilidades - a capacidade do indivíduo manter os níveis de desempenho das habilidades adquiridas em situações de mudança.

A forma com uma tarefa motora é organizada durante uma sessão de treinamento, interfere na qualidade e na quantidade de informações que são recebidas, processadas e geradas pelo aprendiz (LAGE et al., 2015). Isto é, um indivíduo pode adquirir um aperfeiçoamento motor desejado após minutos de prática ou meses, dependendo da complexidade da tarefa e/ou da deficiência motora.

O aperfeiçoamento, ou aprimoramento, motor é uma propriedade do aprendizado motor onde a variabilidade do erro diminui ao longo das repetições de uma tarefa. Para (LAGE et al., 2015) quanto maior o número de repetições em uma determinada atividade motora, maior será a precisão com que ela será executada.

2.2 Análise de Dados

2.2.1 Intervalo Interquartil

Um *boxplot* (Figura 1) resume um conjunto de dados em cinco estatísticas descritivas: mínimo, primeiro quartil (Q_1), segundo quartil ou mediana (Q_2), terceiro quartil (Q_3) e

máximo.



Figura 1 – *Boxplot*, ou diagrama de caixa.

O Intervalo Interquartil (IQR) é a diferença entre o terceiro quartil e o primeiro quartil, ou seja, $IQR = Q_3 - Q_1$. O Limite Inferior (LI) e o Limite Superior (LS) do *boxplot* são mostrados nas Equações 1 e 2, respectivamente. As observações abaixo de LI e acima de LS são potenciais valores discrepantes (*outliers*).

$$LI = Q_1 - 1.5 \times IQR \quad (1)$$

$$LS = Q_3 + 1.5 \times IQR \quad (2)$$

2.2.2 Agrupamento Hierárquico

A análise de agrupamento é uma tarefa importante para uma busca exploratória de padrões em conjuntos de dados (ESTIVILL-CASTRO, 2002). Essa tarefa visa extrair conclusões de uma população heterogênea, dividindo-a em alguns subgrupos mais homogêneos com base em uma métrica de similaridade.

O procedimento geral para análise de agrupamento de conjunto de dados segue duas etapas: calcular a matriz de dissimilaridade ou distância entre as observações e aplicar uma técnica de agrupamento. Esta técnica de agrupamento forma grupos com as observações gerando padrões úteis (ZAJACOVA et al., 2015).

Os *clusters* do processo de agrupamento hierárquico são visualizados na forma de dendrogramas. Os dendrogramas são uma estrutura semelhante a uma árvore. A altura de corte da árvore determina o número de cachos.

2.3 Redes Neurais Artificiais

Uma RNA é um algoritmo projetado para assemelhar-se à forma que o cérebro humano analisa e toma uma decisão (SILVA; SPATTI; FLAUZINO, 2010). O cérebro humano

tem bilhões de neurônios e cada neurônio é composto de um corpo celular responsável por computar informações, transportar para outros neurônios e fornecer uma saída final. Assim, as redes neurais podem ser classificadas em diferentes tipos, usados para diferentes propósitos.

2.3.1 Rede Perceptron

O modelo mais simples de uma RNA é a rede Perceptron. Essa rede possui um único neurônio, chamado de nó ou unidade, onde recebe as entradas x_n , multiplica cada entrada ao seu peso correspondente w_n , soma todos os resultados e passa essa soma por uma função $f(x)$, chamada função de ativação, para produzir uma saída y (Figura 2).

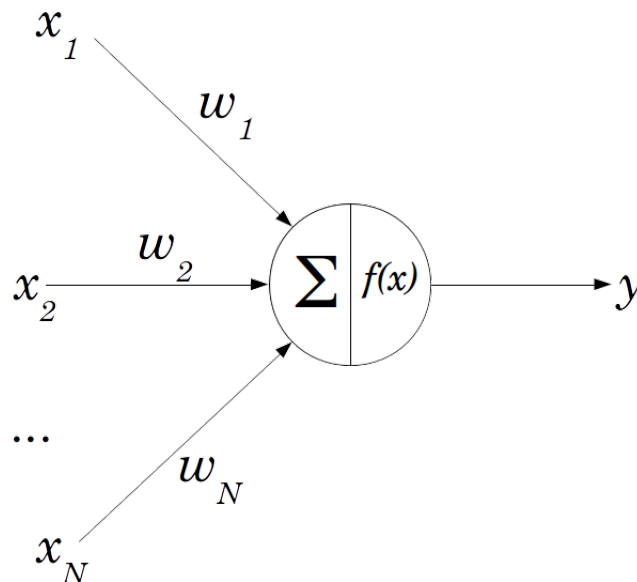


Figura 2 – Representação gráfica de uma rede perceptron.

A função de ativação de uma unidade tem como objetivo transformar a soma dos resultados das multiplicações em uma saída interpretável. Geralmente, uma função de ativação utilizada em uma rede perceptron é a função sigmoide (Equação 3). Essa função mapeia qualquer entrada numérica real para um valor que é 0 ou 1. Dessa forma, a rede é capaz de resolver problemas lineares, como classificações binárias.

$$f(x) = \frac{1}{1 + e^{-x}} \quad (3)$$

2.3.2 Rede Perceptron Multi-camadas

A Perceptron Multi-camadas, ou *Multi-layer Perceptron* (MLP), é uma RNA composta por camadas de entrada e saída e uma, ou mais, camadas ocultas com muitos neurônios

empilhados juntos (Figura 3). Onde cada unidade conecta-se a outra e tem um peso e um limiar associado. Se a saída de qualquer unidade individual estiver acima do valor limite especificado, essa unidade será ativada e enviará dados para a próxima camada da rede. Caso contrário, nenhum dado passará para a próxima camada da rede.

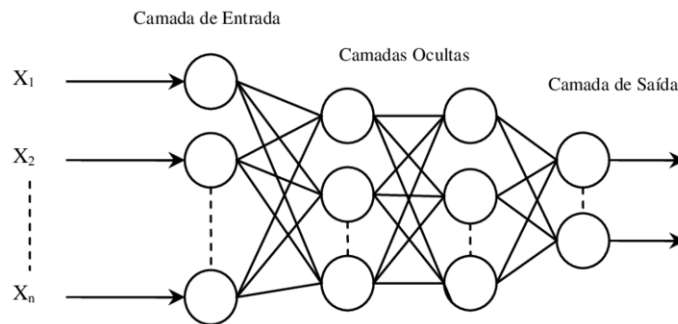


Figura 3 – Representação gráfica de uma rede Perceptron Multi-camadas. Essa rede contém 4 camadas, sendo: 1 camada de entrada com 3 unidades, 2 camadas ocultas com 3 unidades cada e 1 camada de saída com 2 unidades.

O treinamento da MLP utiliza o algoritmo chamado *backpropagation*, retropropagação (RUMELHART; HINTON, 1986). Esse algoritmo é um método que faz o ajuste dos pesos de uma rede neural com base na taxa de erro obtida na época anterior. O ajuste adequado dos pesos permite reduzir as taxas de erro e tornar o modelo preciso, aumentando a sua generalização e precisão.

Dessa forma, uma determinada quantidade de épocas é definida para o treinamento de uma rede neural, bem como outros hiperparâmetros, tais como função de perda, otimizador, função de ativação, número de camadas ocultas, número de neurônios, etc. A escolha de cada hiperparâmetro para o treinamento de uma rede neural deve ser feita com cuidado e pode variar conforme o problema e número de amostras no conjunto de dados. A escolha da quantidade de camadas ocultas em uma RNA e a quantidade de neurônios contidos em cada camada não seguem nenhuma regra universal (BERGSTRÄ; BENGIO, 2012).

2.3.3 Redes Neurais Recorrentes

A RNN é identificada por seus laços de *feedback* (retorno do resultado). Esses algoritmos de aprendizado são aproveitados, principalmente, ao usar dados de séries temporais para fazer previsões sobre resultados futuros, como previsões do mercado de ações ou previsão de vendas com base em informações no passado.

A cada passo de tempo t , a rede se expande para obter a saída no passo de tempo $t + 1$ (Figura 4). Cada rede é semelhante à rede neural *perceptron*. Assim, a rede calcula os valores das unidades ocultas e a saída após os intervalos de tempo. Cada camada

recorrente tem dois conjuntos de pesos: um para a entrada e outro para a unidade oculta. A última camada calcula a saída final para o passo de tempo t .

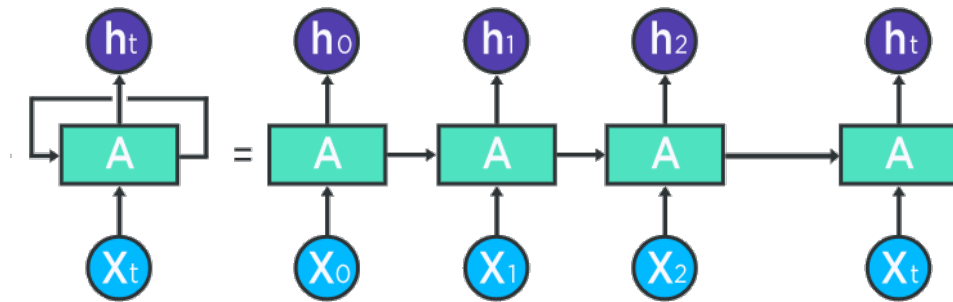


Figura 4 – Representação gráfica de uma RNN desenrolada (LE, 2019).

A rede RNN pode apresentar alguns problemas de dependências de longo prazo quando treinada em séries muito longas de dados. Essas dependências podem aumentar ou diminuir drasticamente os valores no cálculo do treinamento da rede, levando a rede a "esquecer" o que foi aprendido anteriormente.

Para superar esses problemas, Hochreiter e Schmidhuber (1997) desenvolveram uma RNN capaz de manipular o fluxo de informações que entram e saem da rede a fim de evitar os problemas de esquecimento. A RNN LSTM é um tipo de rede neural recorrente utilizada na predição de dados com características temporais, onde dados no passado podem ou não influenciar em dados futuros. Podendo ser aplicada em geração de texto, tradução de texto, previsão de séries temporais, entre outras.

Basicamente, um modelo unitário LSTM é composto por uma célula, uma porta de entrada, uma porta de saída e uma porta de esquecimento. Essa célula consegue recordar valores em intervalos de tempo arbitrários. As três portas regulam o fluxo de informações que entram e saem da célula.

2.4 Trabalhos Relacionados

Santos et al. (2019) introduziram um software para medir o desempenho de indivíduos em um experimento realizando uma tarefa de decalque. A tarefa foi dividida em sessões e blocos de treinamento. Foram aplicados os seguintes algoritmos de Aprendizado de Máquina nos dados coletados do experimento: K-Vizinhos Mais Próximos (KNN), Árvores de Decisão (AD), Máquinas de Vetores de Suporte (SVM) e a rede neural MLP. Os resultados demonstraram que houve aprendizado motor, bem como foi possível aplicar modelos de classificação para prever em qual bloco de treinamento o sujeito está.

Souza et al. (2019) utilizaram o mesmo experimento, dados e algoritmos de Aprendizado de Máquina em Santos et al. (2019) para estimar o tempo de treinamento necessário

para atingir determinado desempenho motor. Os resultados dos algoritmos foram comparados por meio de três métricas: precisão, *F1-Score* (medida F) e coeficiente Cohen Kappa. Onde MLP, SVM e AD, tiveram resultados semelhantes, porém melhores que KNN. Enquanto MLP teve desempenho melhor que todos na métrica *F1-Score*.

Metodologia

Este trabalho utilizou dados pré-existentes (SANTOS et al., 2019; SOUZA et al., 2019) e novos dados para o estudo. Este capítulo descreve os métodos utilizados para a aquisição e análise desses novos dados, bem como a investigação das propriedades e predição do aprendizado motor.

3.1 Participantes e Design do Experimento

Inicialmente, 20 voluntários saudáveis e destros (compostos por alunos e professores da Universidade Federal do Maranhão (UFMA)) foram selecionados para participar do experimento. Cada participante assinou um termo de permissão para autorizar o uso de seus dados para fins de pesquisa, sendo mantido o sigilo dos participantes. Além disso, antes do início do experimento, todos os participantes responderam a dois questionários: i) ao *Edinburgh Handedness Inventory* para mensurar a habilidade do participante em escrever com a mão direita usando o Quociente de Lateralidade (QL) e ii) a uma versão traduzida para o português do *Beck Depression Inventory* (BDI), para identificar sintomas de depressão que podem afetar o resultado do experimento. O estudo não considera os dados dos participantes que apresentam QL igual ou superior a 60,5 e uma pontuação no BDI maior que 12.

Cada um dos participantes realizou o experimento *Tracing Task* (tarefa de decalque), que consiste em sobrescrever (decalcar), com a mão não dominante (esquerda), cada uma das palavras que aparece na tela do computador utilizando uma mesa digitalizadora, como indicado pela seta vermelha (seta 1) na Figura 5. As palavras escolhidas são do português do Brasil, contendo pelo menos 3 a 5 letras, coletadas no banco de dados gratuito pelo Invoke IT¹. Além disso, as palavras impressas na tela pelo *software* estão na fonte cursiva League Script².

¹ <https://invokeit.wordpress.com/frequency-word-lists>

² <https://www.theleagueofmoveabletype.com>



Figura 5 – Representação de uma trial no *software* da tarefa de decalque. Como indicado pela seta vermelha 1, o participante utiliza a mesa digitalizadora com a mão esquerda para tentar desenhar a palavra que aparece na tela. A palavra desenhada e a palavra modelo, indicados pela seta 2 verde, são borradas com um *kernel* gaussiano para a extração da métrica de cada trial (seta 3 amarela).

O experimento é realizado durante três dias consecutivos, em que cada dia é uma sessão, cada sessão é composta por dez blocos e cada bloco contém dez tentativas (*trials*). Em cada tentativa, o tempo necessário para rastrear uma letra foi fixado em 10s; conseqüentemente, o tempo para concluir uma tentativa é $10s \times$ tamanho da palavra. A métrica extraída em cada trial é o erro percentual da diferença entre a palavra modelo (preto) e a imagem decalcada (vermelho), como está indicado pela seta verde (seta 2) na Figura 5. Entretanto, uma margem de erro é levada em consideração, ou seja, ambas as imagens (modelo e decalcada) são borradas com um *kernel* gaussiano de tamanho 50×50 píxeis e 12 píxeis de desvio padrão. Essas imagens são sobrepostas e obtém-se seu erro subtraindo, píxel a píxel, a imagem decalcada da modelo, conforme indicado pela seta amarela (seta 3) na Figura 5.

3.2 Remoção de *Outliers*

Os métodos IQR e Agrupamento Hierárquico foram utilizados para a remoção de dados discrepantes (*outliers*). Cada observação do conjunto de dados corresponde a um participante e as características são seu desempenho na tarefa motora, que é o erro absoluto médio por bloco. Utilizou-se a distância euclidiana para fazer a matriz de dissimilaridade, e, para a fusão de grupos, utilizou-se a abordagem de ligação de Ward (JR, 1963). A implementação do *software* de análise de agrupamento é fornecida pelo pacote *cluster* (MAECHLER et al., 2013) do *R Project for Statistical Software*.

3.3 Análise Estatística

A Análise de Variância (ANOVA) é usada para determinar se existem diferenças estatisticamente significativas entre as médias de três ou mais grupos independentes, ou seja, grupos que não estejam relacionados. Então utilizou-se o método ANOVA para verificar se há diferença estatística entre as sessões, indicando se houve ou não aprendizado motor.

Antes de aplicar o ANOVA, verificou-se a homogeneidade e a normalidade dos dados aplicando os testes de Bartlett e Shapiro-Wilk, respectivamente. Para os conjuntos de dados que não foram normais, foi aplicado o teste não-paramétrico Kruskal-Wallis em vez do ANOVA. Para a análise *post hoc* foi aplicado Tukey HSD e Pairwise Wilcoxon para o ANOVA e o Kruskal-Wallis, respectivamente.

Também aplicou-se o Teste t (STUDENT, 1908) pareado para verificar se há ou não diferença entre o primeiro e último bloco de cada sessão, indicando se houve aprendizado dentro da sessão. Para os dados não normais, aplicou-se o teste de Wilcoxon.

3.4 Pré-processamento dos Dados

A partir dos dados analisados, tem-se o valor do erro de cada tentativa ao longo das sessões. Primeiro, o valor do erro foi escalado entre 0 e 1 para evitar discrepância no intervalo dos valores, utilizou-se o método *MinMaxScaler* da biblioteca Scikit-learn (PEDREGOSA et al., 2011). Depois agrupou-se esses valores por bloco utilizando a função de média e utilizou-se o desvio padrão como outro atributo nos dados. Dessa forma, tem-se o erro médio e seu desvio padrão de cada um dos blocos (10 blocos). Será utilizado esses dois atributos para prever o erro dos blocos da última sessão com base nos blocos anteriores (primeira e segunda sessão).

O primeiro bloco de cada sessão geralmente é ruidoso por ser o primeiro bloco do dia. Assim sendo, excluiu-se o primeiro bloco de cada sessão de todos os voluntários, pelo motivo do erro do primeiro bloco ter muito ruído em relação aos outros, podendo confundir a rede.

3.5 Definição do Modelo LSTM

Utilizaram-se as bibliotecas TensorFlow (ABADI et al., 2015) e Keras (CHOLLET et al., 2015) para a criação de uma rede LSTM com cinco camadas (Tabela 1): i) uma camada de entrada com 256 unidades e formato (18, 2), isto é, 18 valores correspondentes aos blocos da sessão 1 e 2, e, 2 atributos correspondentes ao erro médio e desvio padrão; ii) três camadas ocultas com 128, 64 e 32 unidades, respectivamente; iii) e uma camada de saída (densa) com 9 unidades para a predição de 9 passos à frente.

Tabela 1 – Descrição das camadas contidas no modelo LSTM criado. Total de 524.457 parâmetros treináveis.

Camada (Tipo)	Formato de Saída	Parâmetros
LSTM	(18, 256)	265.216
LSTM 1	(18, 128)	197.120
LSTM 2	(18, 64)	49.408
LSTM 3	32	12.416
Densa	9	297

Ademais, todas as camadas têm a função tangente hiperbólica (Equação 4) como função de ativação. Foram definidas 20 épocas para o treinamento do modelo. Escolheu-se o otimizador Adam (KINGMA; BA, 2014), pois os resultados desse otimizador são geralmente melhores do que todos os outros algoritmos de otimização, o tempo de computação é mais rápido e requer menos parâmetros para ajuste, a taxa de aprendizado foi de 0.01. Por fim, optou-se pela função de perda Huber (Equação 5), a qual é menos sensível a *outliers* que outras funções, como a função erro absoluto médio, já que trata o erro como quadrado apenas dentro de um intervalo.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} = \frac{1 - e^{-2x}}{1 + e^{-2x}} \quad (4)$$

$$L_\delta = \begin{cases} \frac{1}{2} (y - \hat{y})^2 & \text{se } |y - \hat{y}| \leq \delta, \\ \delta (|y - \hat{y}| - \delta/2) & \text{senão.} \end{cases} \quad (5)$$

3.6 Avaliação do Modelo

Utilizou-se validação cruzada na proporção 90/10 para a avaliação do modelo LSTM criado, ou seja, 90% dos dados foram selecionados para o conjunto de treinamento e 10% para o conjunto de teste. Repetiu-se esse particionamento 10 vezes variando os dados do conjunto de teste.

Por fim, utilizaram-se as seguintes métricas para modelos de regressão:

3.6.1 Erro Absoluto Médio

Mean Absolute Error (MAE), ou erro absoluto médio, representa a média da diferença absoluta entre os valores reais e previstos no conjunto de dados. Mede a média dos resíduos no conjunto de dados.

$$MAE(y, \hat{y}) = \frac{1}{N} \sum_{i=1}^N |y_i - \hat{y}_i| \quad (6)$$

, onde \hat{y} é o valor predito de y .

3.6.2 Raiz do Erro Quadrático Médio

Root Mean Squared Error (RMSE), ou raiz do erro quadrático médio, representa a diferença quadrática média entre os valores reais e previsto no conjunto de dados. Nessa métrica o erro retorna à unidade de medida do modelo aplicando a raiz quadrada na diferença quadrática média. Dessa forma, essa métrica mede a variância dos resíduos.

$$RMSE(y, \hat{y}) = \sqrt{\frac{1}{N} \sum_{i=1}^N (y_i - \hat{y}_i)^2} \quad (7)$$

, onde \hat{y} é o valor predito de y .

3.6.3 Erro Percentual Médio Absoluto

Mean Absolute Percentage Error (MAPE), ou erro percentual médio absoluto, é uma métrica sensível a erros relativos, ou seja, o resultado dela não é alterado por uma escala global da variável alvo.

$$MAPE(y, \hat{y}) = \frac{1}{N} \sum_{i=0}^N \frac{|y_i - \hat{y}_i|}{\max(\epsilon, |y_i|)} \quad (8)$$

, onde \hat{y} é o valor predito de y e ϵ é um número arbitrariamente pequeno, mas estritamente positivo, para evitar resultados indefinidos quando y for zero.

Resultados

Utilizou-se os dados das publicações do grupo de pesquisa (SANTOS et al., 2019; SOUZA et al., 2019), assim como novos dados que tinham sido coletados antes da pandemia de COVID-19, mas que ainda não tinham sido analisados. Ao final de todas as sessões programadas do experimento, todos os 20 indivíduos concluíram o experimento e tiveram seus dados analisados. Após aplicar o BDI e o QL, oito voluntários foram excluídos. A Tabela 2 mostra uma visão geral de todos os dados (previamente analisados e não analisados). A coluna à direita apresenta os voluntários excluídos pelo resultado dos questionários ($QL < 60,5$ ou $BDI > 12$).

Tabela 2 – Dados de voluntários convocados para a pesquisa. Esses dados estão ordenados em ordem decrescente pelo valor do QL.

Voluntários	QL	BDI	Voluntários	QL	BDI	Voluntários	QL	BDI
Dados utilizados			<i>Outliers</i>			Dados excluídos		
id01	100	6	id11	78,94	6	id13	86,67	14
id02	100	1	id12	88,66	9	id14	53,54	13
id03	87,5	11				id15	47,37	10
id04	85,71	9				id16	44,44	34
id05	80	9				id17	44,4	9
id06	75	8				id18	36,84	8
id07	68,42	12				id19	20	9
id08	62,5	6				id20	17,64	9
id09	62,5	12						
id10	62,5	4						

4.1 Análise dos Dados

Dois *outliers* foram detectados (id11 e id12), entre todos os dados já coletados, utilizando o método baseado no IQR. Esses dois participantes estão com dados muito diferen-

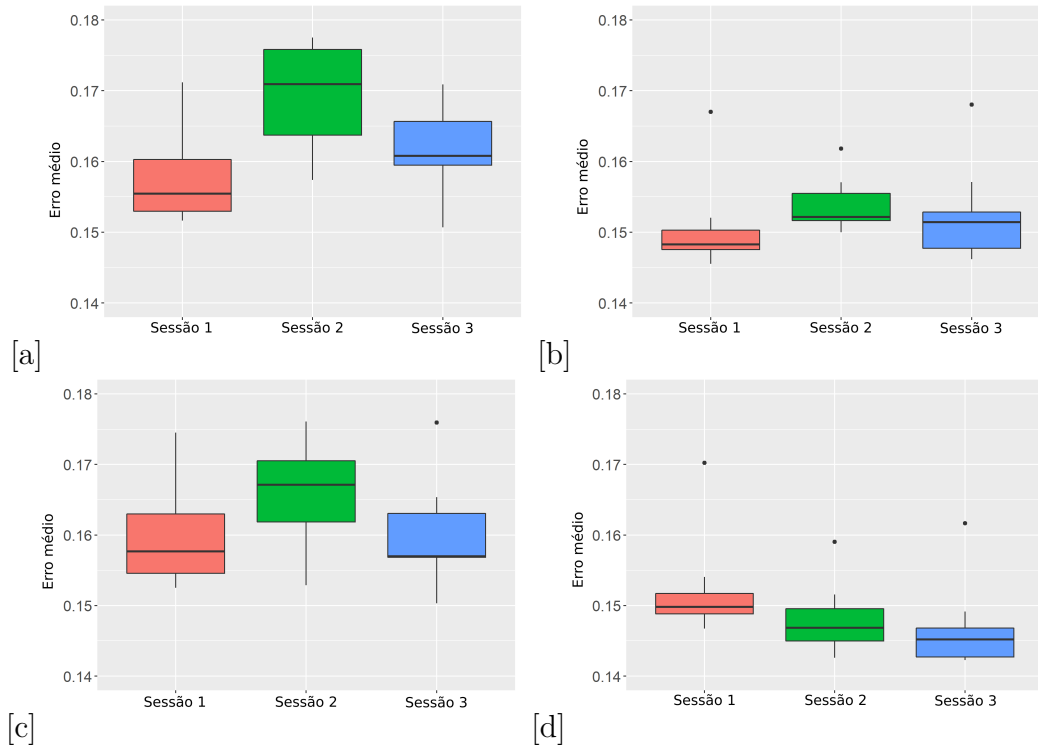


Figura 6 – Erro médio de todos os blocos de cada sessão. (a) Grupo com os outliers id11 e id12. Todos os dados entre as sessões são normais ($p = 0.08$, $p = 0.25$, $p = 0.49$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (ANOVA) apenas entre as sessões 1 e 2 (Tukey HSD). (b) Grupo sem id11. Os dados entre as sessões não são normais ($p = 0.00$, $p = 0.03$, $p = 0.02$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (Kruskal-Wallis) apenas entre as sessões 1 e 2 (Wilcoxon). (c) Grupo sem id12. Todos os dados entre as sessões são normais ($p = 0.17$, $p = 0.37$, $p = 0.08$ para a sessão 1, 2 e 3, respectivamente); não há diferença estatística (ANOVA) entre as sessões. (d) O grupo sem id11 e id12. Apenas os dados da sessão 2 são normais ($p = 0.00$, $p = 0.15$, $p = 0.00$ para a sessão 1, 2 e 3, respectivamente); há diferença estatística (Kruskal-Wallis) apenas entre as sessões 1 e 3 (Wilcoxon).

tes dos demais. As análises estatísticas foram feitas (Figura 6 e Figura 7) com e sem esses *outliers* e viu-se o efeito negativo que eles têm no comportamento geral da base de dados, pois, além de seus dados não corroborarem com a literatura sobre aprendizado motor, têm valores muito diferentes dos demais e mudam o comportamento motor de todo o conjunto de voluntários.

A partir das análises feitas no conjunto de dados, vê-se na Figura 6d e 7d que a média e a variabilidade do erro nos blocos diminuem ao longo das sessões, concordando com a propriedade do aprimoramento motor, bem como indicam o efeito do aprendizado offline. Ou seja, os resultados mostram que os participantes consolidaram o que foi aprendido em uma sessão após um período de 24 horas. Similarmente, nota-se na Figura 7d que o erro dos blocos diminuem durante a sessão, ou seja, indicando um aprendizado online.

O resultado do agrupamento hierárquico realizado na base de dados encontra-se na

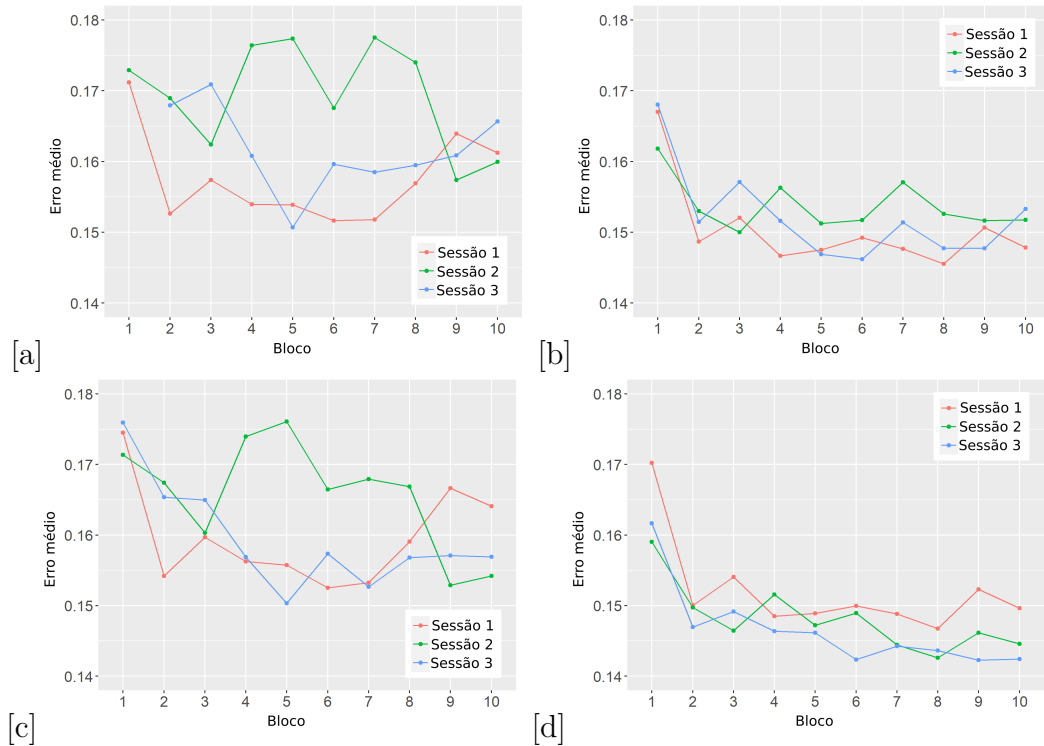


Figura 7 – Erro médio de cada bloco. (a) Grupo com outliers id11 e id12. Há diferença estatística (teste de Wilcoxon) entre os blocos 1 e 10 apenas na sessão 3. (b) Grupo sem id11. Existe uma diferença estatística (Teste t) entre os blocos 1 e 10 na sessão 1, e existe diferença estatística (teste de Wilcoxon) entre os blocos 1 e 10 na sessão 3. (c) Grupo sem id12. Há diferença estatística (teste de Wilcoxon) entre os blocos 1 e 10 nas sessões 2 e 3. (d) Grupo sem id11 e id12. Existe uma diferença estatística (Teste t) entre os blocos 1 e 10 na sessão 1, e existe diferença estatística (teste de Wilcoxon) entre os blocos 1 e 10 na sessão 3.

Figura 8, onde o modelo retornou dois grupos: C1 e C2. O grupo C2 contém apenas um voluntário (id11), que é o valor discrepante encontrado manualmente pelo intervalo interquartil. Analisando esses grupos graficamente (Figura 9a e Figura 9b), pode-se observar que nos dois grupos: visualmente os erros médios das sessões 2 e 3 são maiores que o erro médio da sessão 1; mas só há diferença estatística entre as sessões 1 e 2 de cada grupo; há diferença estatística entre o primeiro (bloco 1) e o último (bloco 10) blocos de cada sessão. O que separa esses grupos é a magnitude do erro médio das sessões, onde o erro médio do grupo C2 é muito maior do que o do grupo C1.

Diminui-se a altura de corte e obteve-se três grupos (C1, C2 e C3) como resultado do agrupamento hierárquico (Figura 10). O grupo C1 é o mesmo grupo sem os outliers vistos anteriormente (Figuras 6d e 7d). Os grupos C2 e C3 contém apenas um voluntário (id12 e id11), sendo estes os outliers encontrados pelo intervalo interquartil. Observa-se pelo dendrograma da Figura 10 que os voluntários id11 e id12 são as observações mais distantes entre as outras, pois estão fora do grupo que contém a maioria dos voluntários, e também estão em dois grupos distintos. Ou seja, o agrupamento hierárquico separou os

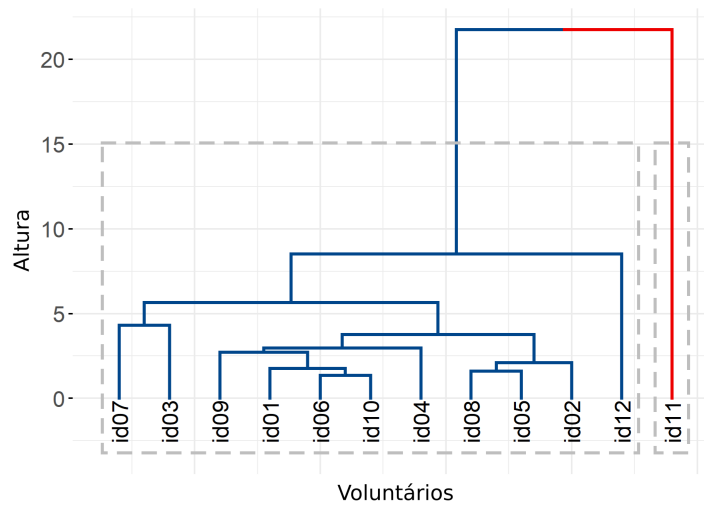


Figura 8 – Agrupamento Hierárquico Aglomerativo com dois grupos: C1 e C2. As linhas azuis representam os membros do grupo C1 e a linha vermelha representa o membro do grupo C2.

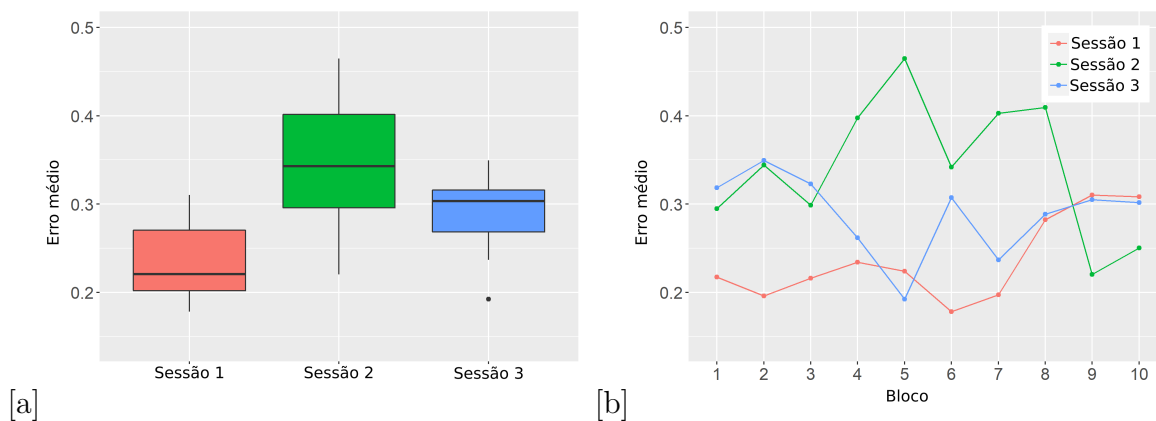


Figura 9 – Grupo C2 do dendrograma na Figura 8 (outlier id11). (a) Erro médio de todos os blocos de cada sessão; os dados são normais ($p = 0.13$, $p = 0.86$, $p = 0.36$ para a sessão 1, 2 e 3, respectivamente) e há diferença estatística (ANOVA) apenas entre as sessões 1 e 2 (Tukey HSD). (b) Erro médio de cada bloco; há diferença estatística (Teste t) entre os blocos 1 e 10 apenas na sessão 1.

dois outliers (previstos manualmente) dos outros dados.

O erro médio da sessão 1 do grupo C2, na Figura 11a, é relativamente menor do que o erro médio das sessões 2 e 3. Há diferença estatística (Kruskal-Wallis) entre as sessões 1 e 2 e sessões 1 e 3 (Wilcoxon), o que contradiz a teoria sobre o aprendizado motor, mas esse é um dado de um outlier. Além disso, na Figura 11b, não há diferença estatística entre os blocos 1 e 10 em nenhuma sessão, indicando que não há aprendizado motor dentro de cada sessão. O grupo C1 da Figura 10 é o mesmo grupo de participantes presente na Figura 6d e 7d.

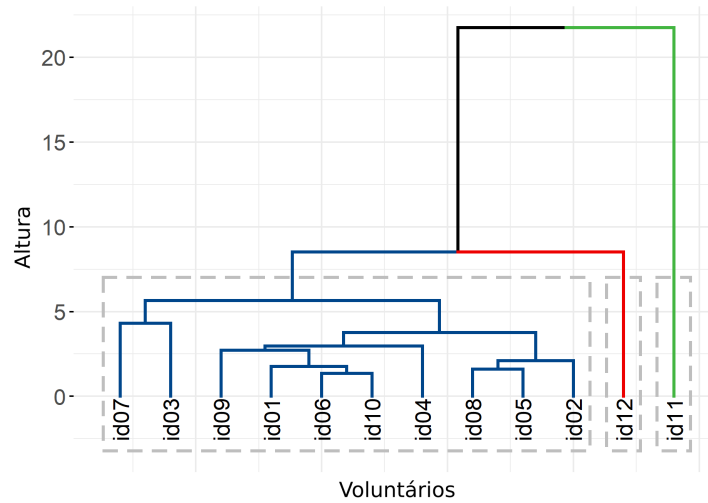


Figura 10 – Agrupamento Hierárquico Aglomerativo com três grupos: C1, C2 e C3. As linhas azuis representam os membros do grupo C1, a linha vermelha representa o membro do grupo C2 e a linha verde representa o membro do grupo C3.

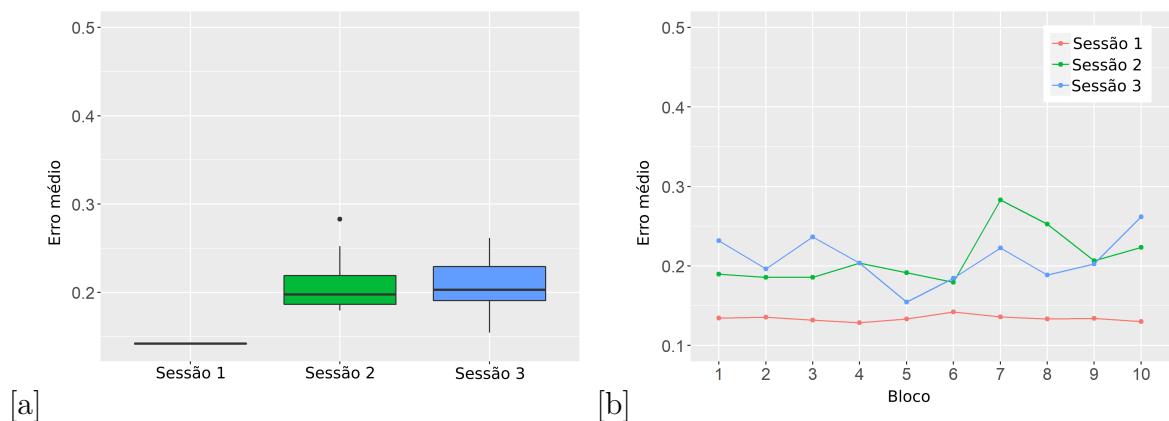


Figura 11 – Grupo C2 do dendrograma na Figura 10 (outlier id12). (a) Erro médio de todos os blocos de cada sessão; apenas os dados da sessão 2 não são normais ($p = 0.44$, $p = 0.03$, $p = 0.98$ para a sessão 1, 2 e 3, respectivamente) e há diferença estatística (Kruskal-Wallis) entre as sessões 1 e 2, e, entre as sessões 1 e 3 (Wilcoxon). (b) Erro médio de cada bloco; não há diferença estatística (Teste t e Wilcoxon) entre os blocos 1 e 10 de cada sessão.

4.2 Previsão do Erro da Terceira Sessão

Após treinamento do modelo LSTM para cada conjunto de treino e teste diferentes, obteve-se a média e desvio padrão de todos os resultados. É possível observar na Tabela 3 que os valores das métricas não variam muito entre os modelos treinados, se aproximando de 0. Ou seja, para as 10 variações do conjunto de treinamento e teste, o modelo conseguiu entregar resultados semelhantes.

Tabela 3 – Resultado do modelo LSTM nas métricas de regressão.

Métrica	Valor Médio	Taxa de Acerto
MAE	0,041 \pm 0,00	95,85 %
RMSE	0,045 \pm 0,00	95,52%
MAPE	0,219 \pm 0,00	78,12 %

Os resíduos (MAE) e a variância (RMSE) do erro entre o valor real e predito se aproximam de 0. Esses resultados mostram que a média das predições se assemelha com a média dos valores reais. É importante destacar que a métrica RMSE é sensível a ruídos por conta da elevação do erro ao quadrado, então o resultado de 95,52% ressalta que os valores preditos não estão significativamente distantes dos valores reais. Em outras palavras, o modelo não previu valores discrepantes.

A Figura 12 mostra que a predição do erro da terceira sessão ao longo dos blocos indica baixa precisão em relação ao valor exato do erro, mesmo que as métricas MAE e RMSE indiquem o oposto. Entretanto, o valor da métrica MAPE, que considera o erro relativo entre as predições e os valores reais, foi 78,12%. Em outras palavras, o desempenho geral do modelo em prever o erro médio dos blocos ao longo da terceira sessão foi de 78,12%.

Ademais, foi possível prever como o erro se comporta ao longo da terceira sessão. O modelo capturou algumas tendências na linha do erro. Por exemplo, na Figura 12b, o valor predito do erro acompanha o valor real do bloco 7 ao bloco 10. É importante destacar que os valores reais da terceira sessão desse conjunto de teste não foram passados ao modelo.

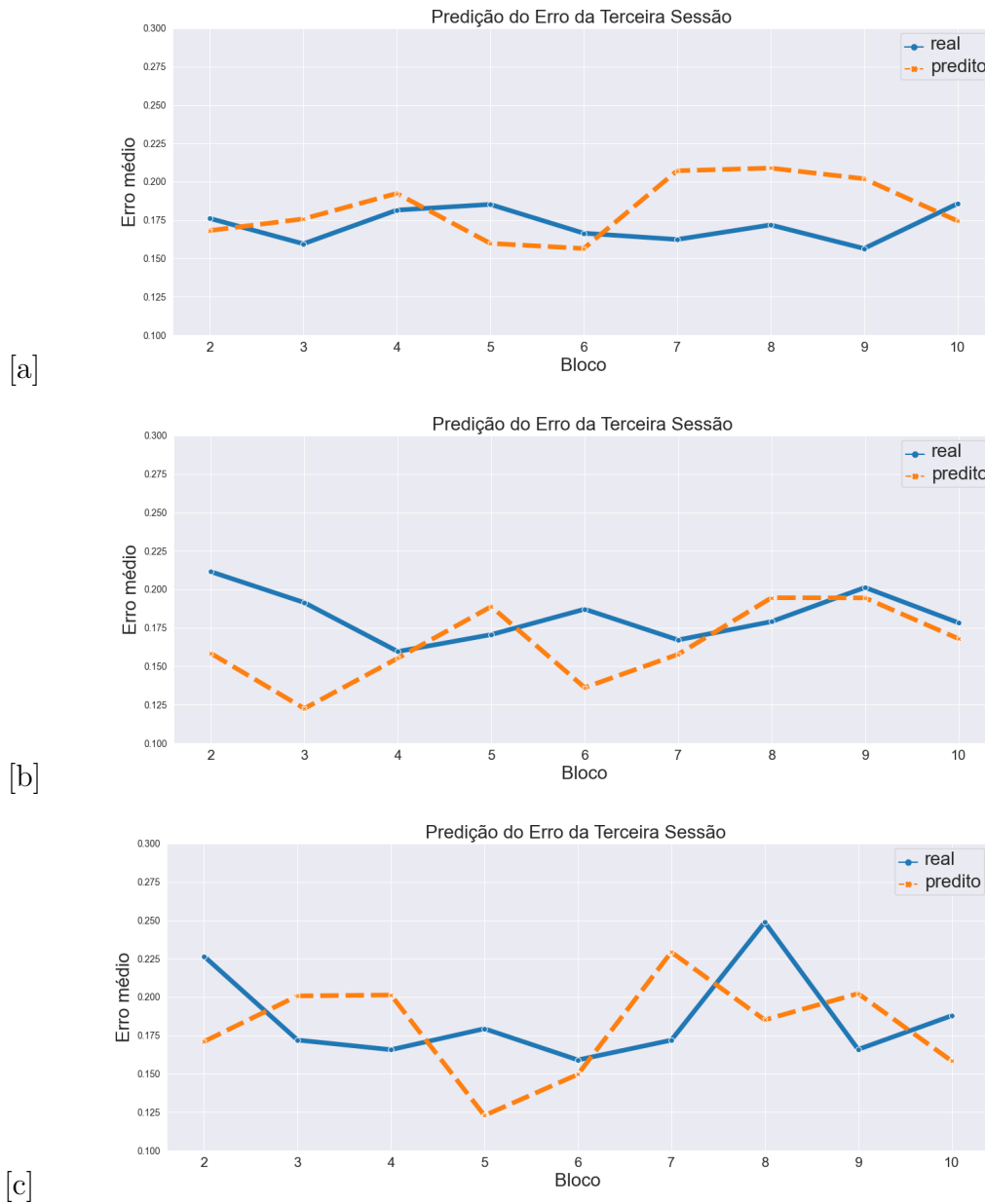


Figura 12 – Três melhores previsões feitas pelo modelo da previsão do erro da terceira sessão. A linha em azul representa o erro real do indivíduo ao longo dos blocos. A linha laranja é o valor predito do erro do participante na terceira sessão, ou seja, os últimos 9 blocos. Estas são as três melhores previsões feitas pelo modelo.

Conclusão

Pela propriedade do aprimoramento motor, nos dados após o agrupamento hierárquico, pode-se dizer que houve melhora no desempenho e na consistência da tarefa de traçado, uma vez que a média e a variabilidade do erro nos blocos diminuem ao longo das sessões. Também foi possível perceber os efeitos do aprendizado offline, quando os participantes que realizaram uma nova sessão, depois de um dia completo após a sessão anterior, obtiverem menor erro.

Foi possível identificar *outliers* automaticamente com o Agrupamento Hierárquico. Essa técnica separou os dados da maioria dos voluntários dos dados discrepantes, evitando que *outliers* interfiram na integridade dos participantes que obtiveram melhora no aprendizado. Essa seleção automática facilita o trabalho de remover ruídos da base de dados, ou seja, ela não necessita de uma extração manual pelo IQR. Dessa maneira, obteve-se uma base de dados mais coesa e pronta para a utilização na etapa de predição do erro da terceira sessão.

Ademais, o resultado da predição do modelo LSTM é satisfatório, uma vez que a forma do erro foi predita e a taxa de erro relativa foi de 78,12%. Com os erros médios preditos de cada bloco na terceira sessão, é possível comparar cada valor predito com o valor que se deseja alcançar na tarefa motora. Entretanto, a base de dados escassa é um fator que influencia negativamente o modelo quanto à generalização para futuros participantes.

Em trabalhos futuros, com o aumento da quantidade de amostras de voluntários, poder-se-á treinar um modelo mais robusto para esta tarefa e utilizar a técnica de Transferência de Aprendizado para adaptar o modelo a um participante em particular, e assim, obter-se a personalização da aquisição de aprendizado motor.

Referências

- ABADI, M. et al. **TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems**. 2015. Software available from tensorflow.org. Disponível em: <<https://www.tensorflow.org/>>.
- ASTRAND, E.; WARDAK, C.; HAMED, S. B. Selective visual attention to drive cognitive brain-machine interfaces: from concepts to neurofeedback and rehabilitation applications. **Frontiers in Systems Neuroscience**, v. 8, 2014. ISSN 1662-5137. Disponível em: <<https://www.frontiersin.org/articles/10.3389/fnsys.2014.00144>>.
- BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. **Journal of machine learning research**, v. 13, n. 2, 2012.
- CHOLLET, F. et al. **Keras**. 2015. <<https://keras.io>>.
- ESTIVILL-CASTRO, V. Why so many clustering algorithms: a position paper. **ACM SIGKDD explorations newsletter**, ACM New York, NY, USA, v. 4, n. 1, p. 65–75, 2002.
- HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. **Neural computation**, v. 9, p. 1735–80, 12 1997.
- JORDAN, M. I.; MITCHELL, T. M. Machine learning: Trends, perspectives, and prospects. **Science**, American Association for the Advancement of Science, v. 349, n. 6245, p. 255–260, 2015.
- JR, J. H. W. Hierarchical grouping to optimize an objective function. **Journal of the American statistical association**, Taylor & Francis Group, v. 58, n. 301, p. 236–244, 1963.
- KINGMA, D. P.; BA, J. **Adam: A Method for Stochastic Optimization**. arXiv, 2014. Disponível em: <<https://arxiv.org/abs/1412.6980>>.
- KITAGO, T.; KRAKAUER, J. W. Chapter 8 - motor learning principles for neurorehabilitation. In: BARNES, M. P.; GOOD, D. C. (Ed.). **Neurological Rehabilitation**. Elsevier, 2013, (Handbook of Clinical Neurology, v. 110). p. 93–103. Disponível em: <<https://www.sciencedirect.com/science/article/pii/B9780444529015000083>>.

- KRAKAUER, J. et al. Motor learning. **Comprehensive Physiology**, v. 9, p. 613–663, 03 2019.
- LAGE, G. M. et al. Repetition and variation in motor practice: a review of neural correlates. **Neuroscience & Biobehavioral Reviews**, Elsevier, v. 57, p. 132–141, 2015.
- LE, J. **Recurrent neural networks: The powerhouse of Language Modeling**. 2019. Disponível em: <<https://builtin.com/data-science/recurrent-neural-networks-powerhouse-language-modeling>>.
- MAECHLER, M. et al. **cluster: Cluster Analysis Basics and Extensions**. [S.l.], 2013. R package version 1.14.4 — For new features, see the 'Changelog' file (in the package source).
- MAHESH, B. Machine learning algorithms-a review. **International Journal of Science and Research (IJSR)**. [Internet], v. 9, p. 381–386, 2020.
- NASEER, N.; HONG, K.-S. fnirs-based brain-computer interfaces: a review. **Frontiers in Human Neuroscience**, v. 9, 2015. ISSN 1662-5161. Disponível em: <<https://www.frontiersin.org/articles/10.3389/fnhum.2015.00003>>.
- PEDREGOSA, F. et al. Scikit-learn: Machine learning in Python. **Journal of Machine Learning Research**, v. 12, p. 2825–2830, 2011.
- PRICHARD, G. et al. Effects of different electrical brain stimulation protocols on subcomponents of motor skill learning. **Brain stimulation**, Elsevier, v. 7, n. 4, p. 532–540, 2014.
- RUMELHART, G.; HINTON, R. Williams, learning representations by back-propagating errors. **Nature**, v. 323, p. 533–536, 1986.
- SAMUEL, A. L. Machine learning. **The Technology Review**, v. 62, n. 1, p. 42–45, 1959.
- SANTOS, M. R. et al. Machine learning to estimate the amount of training to learn a motor skill. In: DUFFY, V. G. (Ed.). **Digital Human Modeling and Applications in Health, Safety, Ergonomics and Risk Management. Human Body and Motion**. Cham: Springer International Publishing, 2019. p. 198–209. ISBN 978-3-030-22216-1.
- SHMUELOF, L.; KRAKAUER, J. W. Recent insights into perceptual and motor skill learning. **Frontiers in Human Neuroscience**, v. 8, 2014. ISSN 1662-5161. Disponível em: <<https://www.frontiersin.org/articles/10.3389/fnhum.2014.00683>>.
- SILVA, I. N. d.; SPATTI, D. H.; FLAUZINO, R. A. **Redes neurais artificiais para engenharia e ciências aplicadas**. [S.l.]: Artliber Editora, 2010.
- SOUZA, E. D. F. de et al. Motor learning and machine learning: Predicting the amount of sessions to learn the tracing task. In: COTA, V. R. et al. (Ed.). **Computational Neuroscience**. Cham: Springer International Publishing, 2019. p. 20–29. ISBN 978-3-030-36636-0.
- STUDENT. The probable error of a mean. **Biometrika**, JSTOR, p. 1–25, 1908.

ZAJACOVA, A. et al. Long-term bmi trajectories and health in older adults: hierarchical clustering of functional curves. **Journal of aging and health**, SAGE Publications Sage CA: Los Angeles, CA, v. 27, n. 8, p. 1443–1461, 2015.