



UNIVERSIDADE FEDERAL DO PARÁ  
INSTITUTO DE CIÊNCIAS BIOLÓGICAS  
PROGRAMA DE PÓS GRADUAÇÃO EM NEUROCIÊNCIAS E BIOLOGIA  
CELULAR

Caio de Oliveira Bastos

**Classificação de Perfis de Produtividade Usando  
Redes Neurais Artificiais a partir de Registros  
Eletroencefalográficos: Uma Aplicação na  
Mineradora Vale S.A., Complexo S11D Eliezer  
Batista**

Belém

2024

Caio de Oliveira Bastos

**Classificação de Perfis de Produtividade Usando Redes  
Neurais Artificiais a partir de Registros  
Eletroencefalográficos: Uma Aplicação na Mineradora Vale  
S.A., Complexo S11D Eliezer Batista**

Dissertação apresentada ao Programa de Pós  
Graduação em Neurociências e Biologia Celu-  
lar do Instituto de Ciências Biológicas como  
requisito parcial para obtenção do grau de  
Mestre em Neurociências e Biologia Celular.

Universidade Federal do Pará

Orientador: Prof. Dr. Bruno Duarte Gomes  
Coorientador: Prof. Dr. Antônio Pereira Júnior

Belém

2024

**Dados Internacionais de Catalogação na Publicação (CIP) de acordo com ISBD**  
**Sistema de Bibliotecas da Universidade Federal do Pará**  
**Gerada automaticamente pelo módulo Ficat, mediante os dados fornecidos pelo(a) autor(a)**

---

Bastos, Caio de Oliveira.

Classificação de perfis de produtividade usando redes neurais artificiais a partir de registros eletroencefalográficos: : uma aplicação na mineradora Vale S.A., complexo S11D Eliezer Batista / Caio de Oliveira Bastos. — 2024.  
62 f. : il. color.

Orientador(a): Prof. Dr. Bruno Duarte Gomes  
Coorientador(a): Prof. Dr. Antônio Pereira Júnior  
Dissertação (Mestrado) - Universidade Federal do Pará,  
Instituto de Ciências Biológicas, Programa de Pós-Graduação em  
Neurociências e Biologia celular, Belém, 2024.

1. Eletroencefalografia. 2. Neurociência Aplicada. 3.  
Inteligência Artificial. 4. Aprendizado de Máquina. 5.  
Indústria da Mineração. I. Título.

CDD 016.6168

---

Caio de Oliveira Bastos

**Classificação de Perfis de Produtividade Usando Redes  
Neurais Artificiais a partir de Registros  
Eletroencefalográficos: Uma Aplicação na Mineradora Vale  
S.A., Complexo S11D Eliezer Batista**

Dissertação apresentada ao Programa de Pós  
Graduação em Neurociências e Biologia Celu-  
lar do Instituto de Ciências Biológicas como  
requisito parcial para obtenção do grau de  
Mestre em Neurociências e Biologia Celular.

Conceito: \_\_\_\_\_

Belém, 30 de Janeiro de 2024.

BANCA EXAMINADORA

---

**Prof. Dr. Bruno Duarte Gomes** - Orientador  
UFPA

---

**Prof. Dr. Antônio Pereira Júnior** - Coorientador  
UFPA

---

**Prof. Dr. Givago da Silva Souza** - Avaliador 1  
UFPA

---

**Prof. Dr. Anderson Manoel Herculano** - Avaliador 2  
UFPA

---

**Prof. Dr. Rommel Thiago Jucá Ramos** - Avaliador 3  
UFPA

# Resumo

Resultados e métodos da neurociência já podem ser aplicados em escala rotineira. De fato, a neurociência aplicada já é usada para, por exemplo, medir e estudar a atividade cerebral sob alta demanda usando eletroencefalografia (EEG). A produtividade em certos ambientes é função direta da atividade cerebral. A atividade intensa de algumas funções cognitivas como atenção sustentada e memória de trabalho influenciam a produtividade diretamente. O ambiente de trabalho em companhias de mineração, onde trabalhadores são muito exigidos física e mentalmente, é um exemplo. O presente trabalho fez parte de um projeto maior chamado “Usando Treinamento Cognitivo para o Desenvolvimento de Operadores de Alto Desempenho” da empresa Vale S. A. e, portanto, todos os operadores que participaram do estudo trabalham para a empresa Vale S. A. e fazem essa simulação de escavação como parte de sua rotina de treinamento. Usamos o EEG para medir produtividade em uma tarefa sem restrições por controle em laboratório. Para isso criamos um algoritmo de aprendizado de máquina para analisar os registros de EEG de operadores de escavadeira de alta capacidade (com  $37 \pm 7$  anos de idade) durante a simulação da escavação, processo usado para treinamento desses operadores, e estimar a produtividade desses operadores durante a simulação, usando somente os dados de EEG coletados em repouso antes e após a operação simulada. Esses dados foram pré-processados usando um filtro passa-banda de 0,5-100 Hz e uma filtragem usando ICA (Independent Component Analysis). Para classificar esses operadores, de acordo com sua produtividade estimada, foi utilizada uma rede neural artificial do tipo *inception* responsável por extrair as características dos dados de EEG e reduzir sua dimensionalidade. Os parâmetros dessa rede, como o número de camadas da rede e o número de neurônios por camada foram otimizados usando a otimização bayeziana. Essas características foram então inseridas em 13 classificadores diferentes e os que tiveram a melhor performance foram escolhidos para compor o algoritmo final e validá-lo. Foi utilizada uma validação *hold-out* para testar a acurácia do algoritmo final, com 20% dos dados disponíveis. Já os 80% restantes foram usados para treinar o algoritmo usando uma validação cruzada. O algoritmo final foi testado em uma classificação com 4 classes e uma classificação binária a partir dos mesmos dados de EEG e teve uma acurácia de classificação bem alta, mesmo usando os dados da validação *hold-out*, chegando a 91,35%, com um classificador *Random Forest*, usando 4 classes e 95,05% na classificação binária, cujo melhor classificador foi um *Extra trees*. Nossos resultados mostram que nosso algoritmo foi bem sucedido e está pronto para ser usado em campo, uma vez que os dados usados nele foram coletados com quase nenhuma alteração nos processos rotineiros dos operadores. Temos, portanto, um protótipo patenteável.

**Palavras-chave:** Eletroencefalografia; Neurociência Aplicada; Inteligência Artificial; Aprendizado de Máquina; Indústria da Mineração.

# Abstract

Results and methods from neuroscience can already be applied on a routine scale. Applied neuroscience is in use, for example, to measure and study brain activity under high demand using electroencephalography (EEG). In certain work environments, human productivity is a direct function of brain activity. The intense recruitment of cognitive functions such as sustained attention and working memory influence productivity directly. The labor demands experienced by workers in mining companies are an example. This work is a part of a bigger project called “Usando Treinamento Cognitivo para o Desenvolvimento de Operadores de Alto Desempenho” from the Vale S. A. mining company and, therefore, all of the workers that participated in this study worked for that company. We used EEG to measure productivity during a task without the constraints typically found in experiments conducted in the laboratory. To this aim, we created a machine learning algorithm to analyze the resting EEG recorded before and after a 4D simulation, where mining workers ( $37 \pm 7$  years old) specialized to operate giant high-capacity shovels. The simulated task consisted of the operator using the shovel in a digging and loading routine. The task was not planned for our research. It was part of the worker’s routine training. That is, we took advantage of the workers’ training to carry out the study. Recordings were pre-processed using a band-pass filter (0.5-100 Hz) followed by filtering using ICA (*Independent Component Analysis*). After each operator finish the simulation, their productivity was measured by VALE S.A technical staff. The main parameter for good productivity was the amount of ore excavated. The operators were divided into groups according to ranges of productivity. The productivity was used as a label for the learning of the algorithm that consisted of an artificial neural network of the type inception. The number of neurons and layers was optimized using Bayesian optimization. The features extracted by the inception were the input to 13 classifiers. The classifier chosen for the final algorithm was the best, that is the one providing the best accuracy in the productivity classification. The training set contained 80% of the data. A hold-out validation was used to test the accuracy of the final algorithm using 20% of the data. The resulting accuracy when the operators were divided into four groups of productivity reached 91.35%. When there were only two groups of productivity the accuracy peaked at 95.05%. Our results showed that even under no laboratory constraints – during the regular training of the operators and using resting EEG – our algorithm succeed and it is ready to be used in future field operations. We have a prototype that is patentable.

**Keywords:** Electroencephalography; Applied Neuroscience; Artificial Intelligence; Machine Learning; Mining Industry.

# Lista de ilustrações

Figura 1.	Modelos Teste-Treino-Validação com validação <i>holdout</i> . . . . .	14
Figura 2.	Modelo de validação cruzada. . . . .	15
Figura 3.	Modelo de validação cruzada com <i>holdout</i> . . . . .	16
Figura 4.	Exemplo de Curva ROC e AUC. A curva ROC é a linha sólida que divide o gráfico e a área o gráfico abaixo dessa linha é chamada de AUC. . . . .	18
Figura 5.	Exemplo de curva de aprendizado. . . . .	19
Figura 6.	Exemplo de gráfico de limiar de decisão. . . . .	20
Figura 7.	Ondas cerebrias com frequências dominantes pertencendo a beta, alpha, theta e delta. . . . .	22
Figura 8.	Filtro passa banda e suas variações dependendo da ordem do filtro. . . . .	24
Figura 9.	Esquemático das áreas de trabalho e seus responsáveis no Programa de Operadores de Alto Desempenho (POAD). . . . .	27
Figura 10.	Protocolo de coleta de dados cognitivos e técnicos . . . . .	31
Figura 11.	Eletrodos presentes no equipamento BrainMaster 24D. . . . .	32
Figura 12.	Fluxo de Modelagem do Machine Learning . . . . .	33
Figura 13.	Matriz de dados de EEG para inserção no classificador cognitivo . . . . .	35
Figura 14.	Mudança de eixos na matriz de dados de EEG . . . . .	35
Figura 15.	Modelo <i>Inception</i> otimizado para extração de padrões dos dados de EEG . . . . .	36
Figura 16.	Média dos registros de EEG, no eletrodo C3, após a filtragem, separados por classe. . . . .	40
Figura 17.	Curva de aprendizado do modelo RF com e sem <i>cross-validation</i> . . . . .	41
Figura 18.	Curvas ROC de performance do modelo RF com <i>cross-validation</i> . . . . .	42
Figura 19.	Limiares de decisão do modelo RF para predição das classes . . . . .	43
Figura 20.	Matriz de confusão do modelo RF nos dados da validação . . . . .	43
Figura 21.	Precisão <i>recall</i> e <i>f1-score</i> do classificador com modelo RF. . . . .	44
Figura 22.	Média dos registros de EEG, no eletrodo C3, após a filtragem, separados por classe. . . . .	46
Figura 23.	Curva de aprendizado do modelo ET com <i>cross-validation</i> . . . . .	47
Figura 24.	Curvas ROC de performance do modelo ET com <i>cross-validation</i> . . . . .	48
Figura 25.	Limiares de decisão do modelo ET para predição das classes . . . . .	49
Figura 26.	Matriz de confusão do modelo ET nos dados da validação . . . . .	49
Figura 27.	Precisão <i>recall</i> e <i>f1-score</i> do classificador com modelo ET. . . . .	50

# Lista de tabelas

Tabela 1.	A matriz de confusão de uma classificação binária. . . . .	17
Tabela 2.	Tabela de classes de produtividade em toneladas por hora. . . . .	30
Tabela 3.	Tabela de classes de produtividade em toneladas por hora. . . . .	31
Tabela 4.	Comparativo de performance de treino entre classificadores treinados usando 4 classes . . . . .	45
Tabela 5.	Métricas de performance dos métodos de validação utilizados, treinados usando 4 classes . . . . .	45
Tabela 6.	Comparativo de performance de treino entre classificadores treinados usando 2 classes . . . . .	51
Tabela 7.	Métricas de performance dos métodos de validação utilizados, treinados usando 2 classes . . . . .	51

# Lista de abreviaturas e siglas

EEG	Eletroencefalografia
IA	Inteligência Artificial
SMR	Ritmo Sensório Motor
TC	Treinamento Cognitivo
PCA	Principal Component Analysis
SMOTE	Synthetic Minority Oversampling Technique
RF	Random Forest Classifier
MEC	Mutual Equidistant-scattering Criterion
EDF	European Data Format
Hz	Hertz
ICA	Independent Component Analysis
LightGBM	Light Gradient Boosting Machine
CSV	Comma-separated values
ROC	Receiver Operating Characteristics
SINDy	Sparse Identification of Nonlinear Dynamics
SIAAFT	Stochastic Iterative Amplitude Adjusted Fourier Transform

# Sumário

<b>1</b>	<b>INTRODUÇÃO</b>	<b>11</b>
<b>1.1</b>	<b>Uma breve história do EEG</b>	<b>21</b>
<b>1.2</b>	<b>Aprendizado de máquina para classificação de EEG</b>	<b>23</b>
<b>2</b>	<b>JUSTIFICATIVA</b>	<b>26</b>
<b>3</b>	<b>OBJETIVOS</b>	<b>29</b>
<b>3.1</b>	<b>Objetivo Geral</b>	<b>29</b>
<b>3.2</b>	<b>Objetivos Específicos</b>	<b>29</b>
<b>4</b>	<b>MATERIAIS E MÉTODOS</b>	<b>30</b>
<b>4.1</b>	<b>Dados utilizados</b>	<b>30</b>
<b>4.2</b>	<b>Previsão de performance usando a rede inception</b>	<b>33</b>
4.2.1	Carregamento e pré processamento dos dados	33
4.2.2	Engenharia de Características e Redução de Dimensionalidade I	34
4.2.3	Engenharia de Características e Redução de Dimensionalidade II	38
4.2.4	Treinamento e Validação do Classificador	38
<b>5</b>	<b>RESULTADOS</b>	<b>40</b>
<b>5.1</b>	<b>Classificação em quatro classes</b>	<b>40</b>
5.1.1	Visualização dos dados de EEG por classe	40
5.1.2	Curva de Aprendizado	41
5.1.3	Curvas ROC	41
5.1.4	Limiar de Decisão do Modelo	42
5.1.5	Matriz de Confusão	42
5.1.6	Relatório de Classificação	44
5.1.7	Validação	44
<b>5.2</b>	<b>Classificação em duas classes</b>	<b>46</b>
5.2.1	Visualização dos dados de EEG por classe	46
5.2.2	Curva de Aprendizado	47
5.2.3	Curvas ROC	47
5.2.4	Limiar de Decisão do Modelo	48
5.2.5	Matriz de Confusão	48
5.2.6	Relatório de Classificação	50
5.2.7	Validação	50
<b>6</b>	<b>DISCUSSÃO</b>	<b>53</b>

<b>7</b>	<b>CONCLUSÕES . . . . .</b>	<b>56</b>
	<b>REFERÊNCIAS . . . . .</b>	<b>59</b>

# 1 Introdução

Métodos e resultados da neurociência não estão mais restritos aos laboratórios acadêmicos e chegaram definitivamente às empresas (POP; DABIJA; IORGA, 2014; CHERUBINO et al., 2019). É nesse contexto científico e do mundo corporativo que se situa o presente projeto que é parte de um projeto maior chamado “Usando Treinamento Cognitivo para o Desenvolvimento de Operadores de Alto Desempenho”, fruto da cooperação entre o Laboratório de Neurofisiologia Eduardo Oswaldo Cruz (LNEOC) e o Instituto Tecnológico Vale - Desenvolvimento Sustentável. O projeto atende a crescente demanda pela cooperação entre universidades e indústria e tem potencial não só para a publicação de artigos, mas também para o desenvolvimento de patentes.

Dentre os dados coletados nesse projeto, os que são utilizados nessa dissertação consistem em registros coletados através do eletroencefalograma (EEG) e registros de produtividade coletados por um simulador de escavação. O EEG é um método que monitora de modo não invasivo as atividades elétricas cerebrais em grande escala através de eletrodos colocados no couro cabeludo (GOMES, 2015). No presente trabalho, usamos a dinâmica cerebral revelada pelo EEG em busca de suas características relacionadas a produtividade e tentamos relacionar essa dinâmica com a produtividade do operador durante seus treinamentos rotineiros.

Os dados neurais tem um forte componente estocástico e por isso, há consenso de que o cérebro é um sistema caótico-determinístico. Isso se reflete na natureza do EEG; uma série temporal usualmente não-linear e não estacionária (RODRIGUEZ-BERMUDEZ; GARCIA-LAENCINA, 2015). Para realizar classificações utilizando dados tão complexos, técnicas de aprendizado de máquina são normalmente preteridas por apresentarem características similares ao sistema cujos dados estão classificando: são métodos não lineares, e capazes de aprender padrões que seguem distribuições variadas de probabilidade. Ou seja, para estudar um sistema complexo, nada melhor do que usar um outro sistema complexo.

Mas esses algoritmos nem sempre tiveram como sua principal característica o aprendizado. A história do aprendizado de máquina é longa, tendo começado nos anos 1950, inicialmente com os pesquisadores tentando encontrar um algoritmo generalista e universal, que fosse capaz de resolver qualquer tipo de problema, o que caracteriza a primeira era do aprendizado de máquina, que pode ser chamada de era do raciocínio. No entanto, conforme a área foi avançando e se tornando mais popular os pesquisadores começaram a perceber que as Inteligências Artificiais (IA) criadas precisavam adquirir conhecimento já existente para serem inteligentes, dando início a uma segunda era do aprendizado de máquina no meio da década de 1970, chamada de era do conhecimento,

em que o foco foi no desenvolvimento de sistemas especialistas a partir da representação simbólica do conhecimento, que foram desenvolvidos e aplicados com sucesso em uma grande variedade de campos (ZHOU, 2021).

Já no início dos anos 1990, com a geração de cada vez mais dados e informação, começaram a surgir vários campos na pesquisa e na indústria em que os algoritmos da era do conhecimento não eram mais suficientes para atender às demandas, uma vez que demanda muito esforço e tempo extrair e resumir conhecimento de forma que os computadores possam entendê-lo. Foi então que alguns cientistas começaram a explorar a ideia de que esses algoritmos poderiam aprender sozinhos, dando início à terceira, e atual, era, a era do aprendizado. Nessa era mais recente algumas técnicas ganharam mais destaque, como as máquina de vetores de suporte, os algoritmos evolutivos e as redes neurais artificiais, técnicas em que os algoritmos aprendem sozinhos, através do aprendizado por conexionismo, em que esses algoritmos vão criando conexões entre os dados mostrados a eles para gerar o aprendizado e, para isso, precisam de uma grande quantidade de dados para serem efetivos (ZHOU, 2021).

No começo do século 21 o conexionismo ganhou ainda mais destaque com o surgimento do aprendizado profundo, ou *deep learning*. O *deep learning* é um grupo de métodos que contêm múltiplas camadas de representação e permitem à máquina ser alimentada com dado bruto e descobrir automaticamente as representações necessárias para classificação ou detecção de padrões (LECUN; BENGIO; HINTON, 2015). Esse grupo de métodos é muito interessante não só por terem uma melhor performance do que métodos de aprendizado com somente uma camada, mas também por possuírem uma grande flexibilidade, podendo ser aplicado em uma grande quantidade de áreas diferentes com alta taxa de sucesso.

Esse grupo de métodos é especialmente eficaz em tarefas de classificação onde, a cada camada de representação, os aspectos mais relevantes das entradas são amplificados e os irrelevantes são suprimidos. Como esses métodos possuem diversas camadas conectadas entre si eles são capazes aprender funções e realizar tarefas de classificação altamente complexas, e por isso são muito utilizados para classificação de dados de EEG (BAKARJI et al., 2022).

Esses métodos que usam o *deep learning* tem, naturalmente, muitos parâmetros que podem ser otimizados para melhor resultado da rede, como o número de camadas e o número de neurônios por camada, que podem tem um impacto alto na performance da rede. Normalmente essa otimização é feita manualmente, com o especialista em IA alterando e testando várias configurações diferentes de parâmetros, mas quando o algoritmo demora várias horas para rodar essa otimização manual começa a ser inviável. Para isso existem métodos automáticos de otimização de parâmetros, dentre eles um dos mais utilizados e que tem maior sucesso é otimização bayeziana (BASHASHATI; WARD; BASHASHATI,

2016).

A otimização bayeziana é uma técnica de otimização global de funções caixa-preta, como as redes neurais artificiais, que visa encontrar a melhor performance global de um grupo de hiper-parâmetros, como o número de camadas e de eletrodos em uma rede neural. Ela funciona criando um *surrogate*, ou seja uma série temporal linear artificial com características estatísticas iguais a série original, e quantificando a incerteza usando uma técnica de regressão chamada processo Gaussiano, que modela a função objetivo, da qual o *surrogate* criado deve se aproximar e uma função de aquisição que define qual o próximo passo a ser tomado para se aproximar da função objetivo. Esse processo é então repetido diversas vezes trocando os hiper-parâmetros e tentando se aproximar cada vez mais dessa função de objetivo, de forma a percorrer um espaço de fase bem grande em busca dos hiper-parâmetros ótimos globais (FRAZIER, 2018).

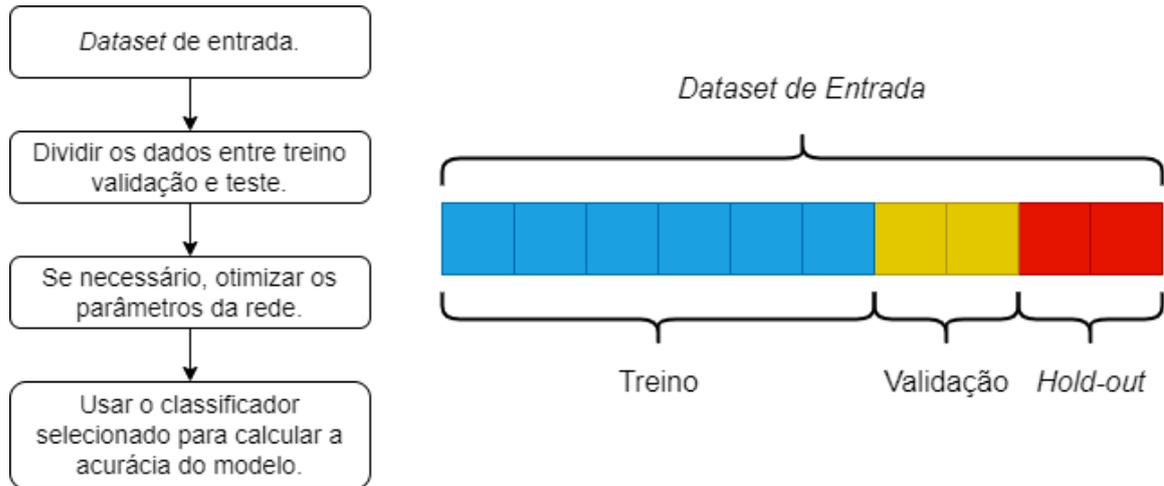
Outra forma muito interessante e efetiva de usar esse grupo de redes neurais artificiais é usá-las como extratores de características, que depois serão adicionadas a outros classificadores, geralmente mais simples, para fazer a classificação. Um dos classificadores mais utilizados para isso é o classificador *random forest*, onde são treinadas várias árvores de decisão, algoritmos muito mais simples do que redes neurais artificiais, em vários pequenos subgrupos dos dados de treinamento e suas previsões são combinadas através de uma contagem de votos entre as melhores dessas árvores (ALPAYDIN, 2020). Outro classificador muito utilizado é o classificador *extra trees* que usa um método muito similar ao *random forest*, com a criação de árvores de decisão que vão classificar os dados conjuntamente através de um sistema de votação, mas adicionando um fator extra de aleatoriedade ao se aleatorizar quais os dados presentes em cada uma dessas árvores, misturando os dados de treinamento das árvores entre si.

Quando se usa esse tipo de algoritmo de aprendizado de máquina para classificação é importante que se obtenha acurácias altas, tipicamente maiores que 90%. Conseguir 100%, por outro lado, não deixa espaço à variações, típicas em situações reais, processo esse chamado de *overfitting*. Isso ocorre quando o algoritmo fica especialista em classificar conjuntos de dados pertencentes a uma dada distribuição de probabilidade. Mas quando é apresentado qualquer exemplo minimamente diferente, sua acurácia cai muito. Isso implica numa falta de capacidade de generalização e torna o algoritmo praticamente inútil para uso prático. No aprendizado de máquina esse fenômeno é um dos problemas mais fundamentais e difíceis de se superar nesse campo (ZHOU, 2021).

Para evitar esse tipo de problema é necessário avaliar a capacidade de generalização desses algoritmos durante e depois do treinamento e isso é feito através do uso de técnicas de validação. Para realizar a validação normalmente são usados conjuntos de teste para estimar a capacidade do algoritmo de classificar novos exemplos e a taxa de erro obtida usando o conjunto de teste é tratada como uma estimativa da taxa de erro que seria obtida

com a generalização, ou seja, ao adicionar dados inteiramente novos à base de dados. Para que isso seja eficaz o conjunto de teste e o conjunto de treinamento do algoritmo devem ser tão mutuamente exclusivos quanto o possível, de forma que as amostras de treino nunca ou quase nunca apareçam no conjunto de teste. Um dos modelos mais clássicos para fazer validação desses algoritmos é o modelo treino-validação-teste com validação *holdout*, que pode ser visto na figura 1.

Figura 1. Modelos Teste-Treino-Validação com validação *holdout*



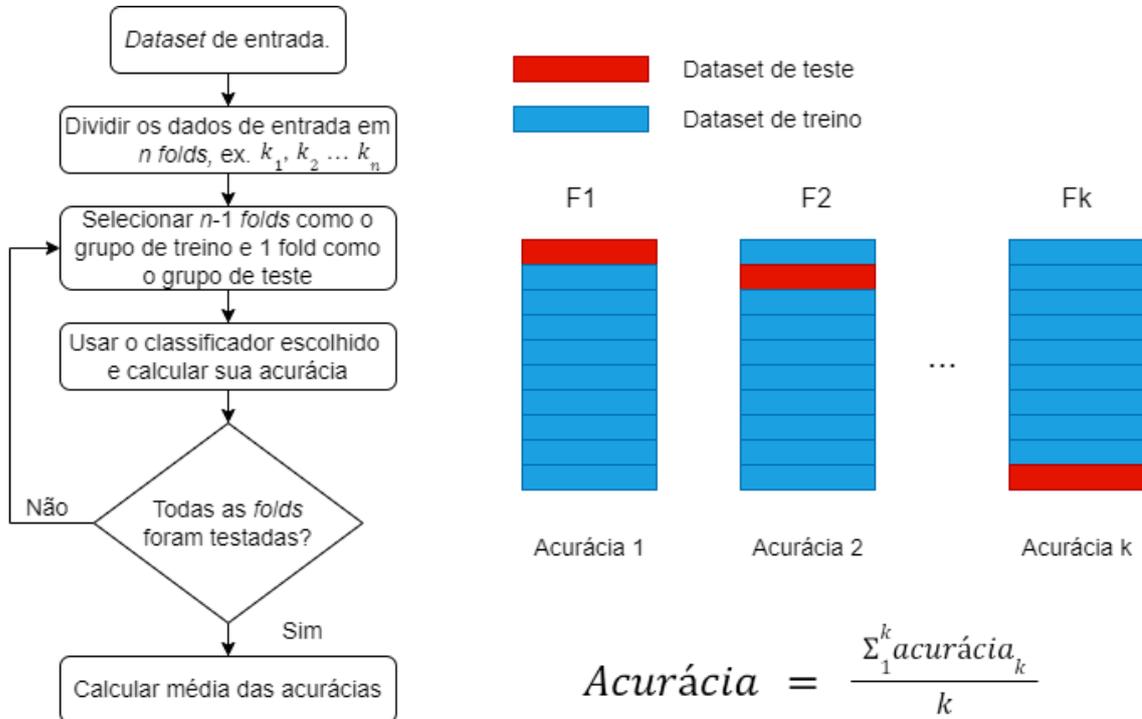
Fonte: Adaptado de Horvat (2020).

O termo validação *holdout* implica somente que os dados da validação somente serão mostrados para o classificador no momento da validação, não sendo utilizados nem para treinamento nem para validação do algoritmo. Dentro do modelo mostrado na figura 1 o dado que entra no algoritmo é separado em 3 blocos diferentes, sendo eles blocos de treino, validação e teste (ou *holdout*, no caso da validação *holdout*). Os dados do bloco de treino são usados para treinar o algoritmo, por isso normalmente ele é o maior. Durante o treinamento os parâmetros do algoritmo precisam ser atualizados para que a performance melhore e, para isso é necessário testá-lo com dados que ainda não foi usado em seu treinamento e que vai simular, o bloco de dados de validação é utilizado, sendo que essa etapa de testar o algoritmo com os dados da validação acontecem várias vezes durante a otimização dos parâmetros dos algoritmos. O ultimo grupo utilizado nesse modelo é o de teste *holdout* cujos dados só são usados no final de toda a execução do algoritmo, quando os parâmetros já foram todos otimizados, para fazer o teste final da performance do algoritmo, com dados que ele nunca viu anteriormente.

Esse método é bastante utilizado e bem eficaz, mas ele vem com um dilema. Quanto mais se aumenta a porcentagem dos dados presentes no grupo de treino melhor o algoritmo vai prever o conjunto de dados usado para treiná-lo, mas a previsão feita por ele vai se tornar menos confiável por conta do menor tamanho dos grupos de validação e teste. Já se o tamanho dos grupos de validação e teste forem muito grandes haverá menos dados

para treinamento do algoritmo, o que implica numa queda de sua performance (ZHOU, 2021). Não existe nenhuma solução perfeita para esse dilema, mas, por conta dele, outros tipos de validação foram criados, como a validação cruzada, apresentada na figura 2.

Figura 2. Modelo de validação cruzada.



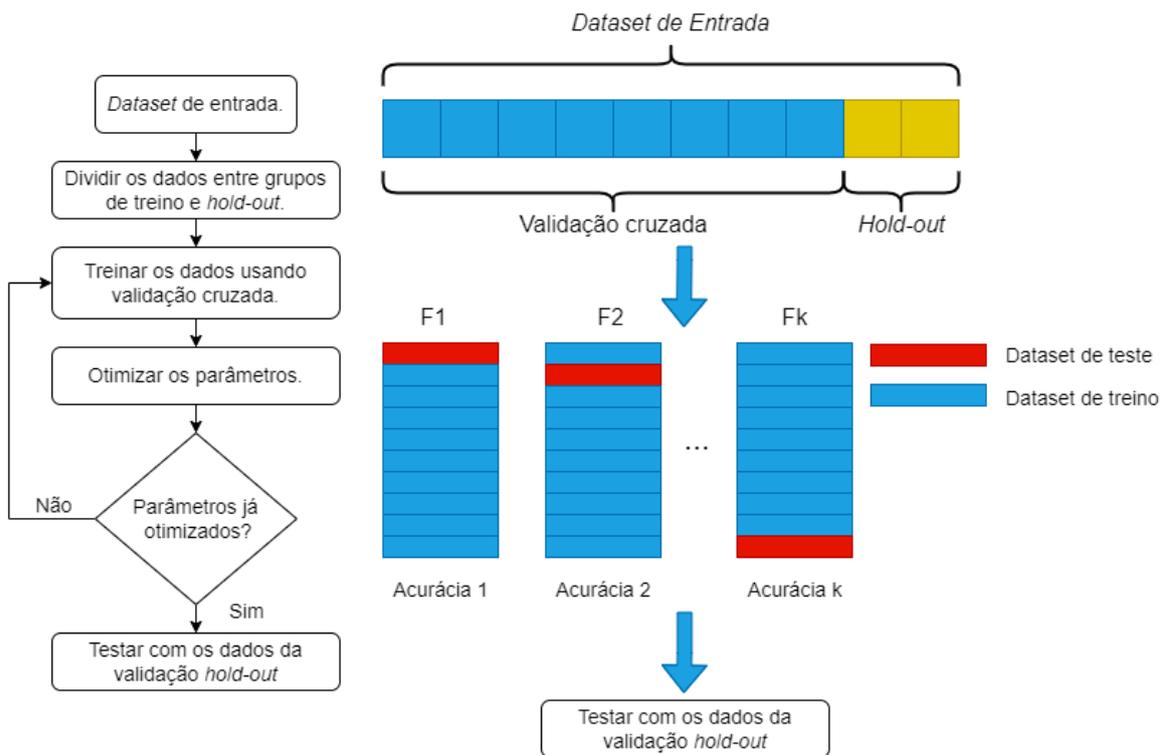
Fonte: Adaptado de Horvat (2020).

No método da validação cruzada o grupo inicial de dados é dividido em um número  $k$  de subgrupos, que juntos totalizam todos os dados de entrada. Esse número  $k$  de grupos depende da situação em que o algoritmo é aplicado e pode variar bastante, sendo normalmente estipulado pela opinião do especialista em aprendizado de máquina que o está aplicando. Com esse método um dos  $k$  subgrupos é usado para testar a capacidade de aprendizado do algoritmo enquanto os outros são usados para treiná-lo. Esse processo é repetido  $k$  vezes e a média das acurácias calculadas nessas  $k$  repetições é considerada como a acurácia final do modelo, como mostrado na figura 2 (HORVAT; HAVAŠ; SRPAK, 2020).

O método da validação cruzada é muito eficaz principalmente para lidar com *datasets* pequenos, que não contem uma quantidade suficiente de dados para fazer uma separação prévia de grupos de treino-teste-validação no começo do processo. Sua principal vantagem em relação ao modelo teste-treino-validação, portanto, é que todos os dados de entrada são utilizados tanto para treino, quanto para teste (BERRAR et al., 2019). A desvantagem é que ele não é tão confiável uma vez que ele não guarda dados para uma validação *holdout*, que é muito eficiente em simular como o algoritmo se comportaria com dados novos, algo essencial em aplicações reais, quando o influxo de novos dados é constate.

Para compensar por essa fraqueza da técnica adicionamos, na aplicação desse método, uma camada de validação *holdout* ao modelo da validação cruzada, separando o dado no início entre um grupo de teste-treino em que foi usada a validação cruzada e um grupo de validação *holdout* que só é mostrado após todo o processo de treinamento e otimização de parâmetros do algoritmo, para obter uma classificação final com dados que nunca antes tinham sido mostrados ao algoritmo. Esse método é chamado na literatura de validação cruzada com *holdout* (TOUGUI; JILBAB; MHAMDI, 2021) e é ilustrado na figura 3.

Figura 3. Modelo de validação cruzada com *holdout*.



Fonte: Produzido pelo autor.

Existem várias métricas na literatura usadas para avaliar a performance dos algoritmos de aprendizado de máquina e sua capacidade de generalização. Cada uma dessas métricas avalia aspectos específicos e o uso de múltiplas delas simultaneamente nos permite avaliar os resultados por ângulos diferentes, uma vez que a qualidade do modelo varia muito a partir dos algoritmos usados e do que os dados utilizados requerem. A acurácia, por exemplo é uma das métricas de performance mais primordiais e utilizadas em problemas de classificação e representa a proporção de amostras corretamente classificadas e foi a métrica principal utilizada ao longo desse trabalho (ZHOU, 2021).

Uma única medida de performance, no entanto, não é suficiente para uma análise completa da situação e, por isso foram usadas outras métricas auxiliares, que são Precisão, *Recall*, *F1* e *AUC*, todas calculadas a partir da matriz de confusão da classificação. Uma matriz de confusão é uma maneira simples e muito utilizada de compilar algumas

informações úteis para a avaliação de performance de algoritmos de previsão. Ela compila 4 valores, sendo eles: Número de verdadeiros positivos; Número de falsos negativos; Número de falsos positivos e Número de verdadeiros negativos, que são usadas para calcular várias outras métricas de performance. Um esquemático de uma matriz de confusão para classificação binária pode ser encontrada na tabela 1.

Tabela 1. A matriz de confusão de uma classificação binária.

Classe Real	Classe prevista	
	Positiva	Negativa
Positiva	Verdadeiros Positivos	Falsos Negativos
Negativa	Falsos Positivos	Verdadeiros Negativos

Fonte: Adaptado de ZHOU (2021).

Um verdadeiro positivo é uma amostra que o modelo corretamente classifica como da classe positiva, enquanto um falso positivo é uma amostra originalmente da classe negativa que o modelo incorretamente classifica como da classe positiva. Já um falso negativo é uma amostra originalmente da classe positiva que o modelo incorretamente classifica como da classe negativa e um verdadeiro negativo é uma amostra que o modelo corretamente classifica como da classe negativa. Essas classificações podem ser feitas tanto em forma do número de ocorrências quanto como porcentagens, que em cada linha da tabela devem somar 100%.

A partir desses valores podemos calcular a precisão e o *recall*. A precisão responde a pergunta: Dentre todas as amostras classificadas como positivas pelo classificador, que porcentagem delas foi corretamente classificada como tal? Já o *recall* responde a pergunta: Dentre as amostras cuja classe real é positiva, que porcentagem delas foi corretamente classificada como tal? As formulas que calculam essas métricas podem ser vistas nas equações 1.1 e 1.2, respectivamente, e ambas, idealmente, devem ter valores o mais próximos de 1 possível (ALPAYDIN, 2020).

$$Precisão = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosPositivos} \quad (1.1)$$

$$Recall = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosNegativos} \quad (1.2)$$

Em algumas aplicações a importância da precisão e do *recall* são diferentes. Por exemplo, precisão é mais importante em sistemas de recomendação, uma vez que é mais desejável que o conteúdo seja de alto interesse do usuário e o atrapalhe o mínimo possível. Já o *recall* é mais importante em sistemas como os de reconhecimento facial de criminosos, já que é imprescindível que o maior número possível deles seja identificado, mesmo que hajam

alguns erros (ZHOU, 2021). Mas em muitos problemas a importância dessas estatísticas é bem próxima e, nesses casos, a métrica F1 é muito útil. Essa métrica compila informações tanto da precisão quanto do *recall*, sendo calculada através da média harmônica entre essas duas métricas, como pode ser visto na equação 1.3.

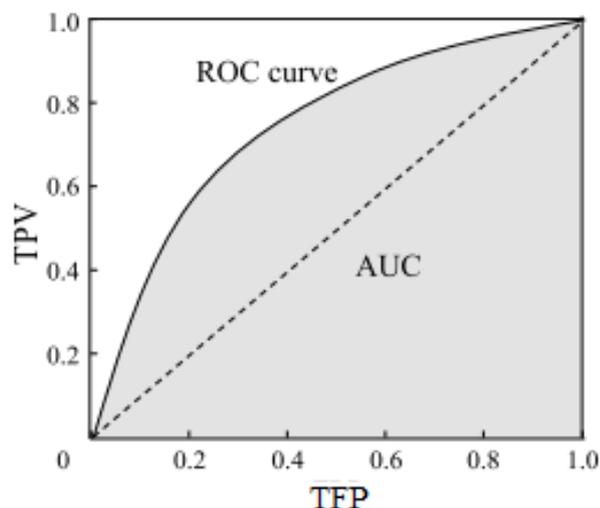
$$F1 = \frac{2 * Precisão * Recall}{Precisão + Recall} \quad (1.3)$$

Além das métricas numéricas vários gráficos podem ser usados para avaliar a performance de algoritmos baseados no aprendizado de máquina, sendo um dos mais importantes a curva ROC (do inglês *Receiver Operating Characteristics*). A curva ROC foi desenvolvida inicialmente para detecção de sinais em radares e já era usada desde a década de 1970, mas sua primeira aplicação em IA foi feita por Spackman em 1989. As curvas ROC são gráficos bidimensionais em que o eixo x representa a taxa de falsos positivos e o eixo y representa a taxa de positivos verdadeiros, que são calculadas através das equações 1.4 e 1.5, respectivamente (FAWCETT, 2006). Um exemplo da curva ROC pode ser visto na figura 4.

$$TaxadeFalsosPositivos = \frac{FalsosPositivos}{FalsosPositivos + VerdadeirosNegativos} \quad (1.4)$$

$$TaxadePositivosVerdadeiros = \frac{VerdadeirosPositivos}{VerdadeirosPositivos + FalsosNegativos} \quad (1.5)$$

Figura 4. Exemplo de Curva ROC e AUC. A curva ROC é a linha sólida que divide o gráfico e a área o gráfico abaixo dessa linha é chamada de AUC.

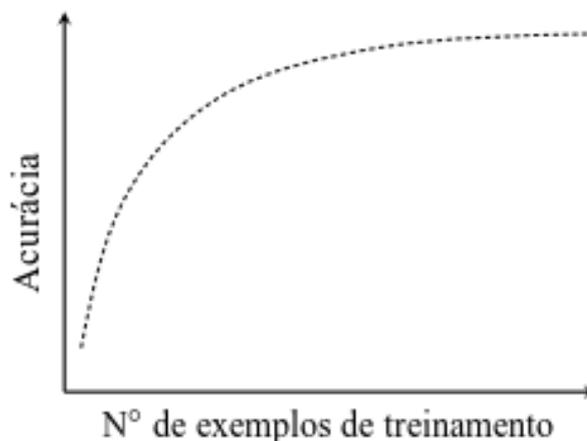


Fonte: Adaptado de Zhou (2021)

Além de ser interessante por mostrar uma análise visual interessante para a avaliação de performance as curvas ROC também são interessantes porque a informação dada por elas pode ser resumida em uma métrica numérica chamada AUC (do inglês *Área Under Curve*) que representa a área do gráfico que está abaixo da curva ROC. Um exemplo de AUC pode ser visualizado na figura 4.

Outra análise visual muito usada ao se avaliar a performance de algoritmos de IA é a análise das curvas de aprendizado. As curvas de aprendizado são curvas que foram originalmente introduzidas na psicologia no final do século 19 e eram muito utilizadas para previsão da produtividade humana, mas no contexto do aprendizado de máquina elas ganharam uma nova atribuição que é a de auxiliar no monitoramento do aprendizado de classificadores durante o seu treinamento. Nessas curvas a performance ou acurácia do classificador é traçada no eixo y enquanto o número de vezes que o grupo de dados de treinamento foram apresentados ao classificador é traçada no eixo x, como mostrado na figura 5 (VIERING; LOOG, 2022).

Figura 5. Exemplo de curva de aprendizado.

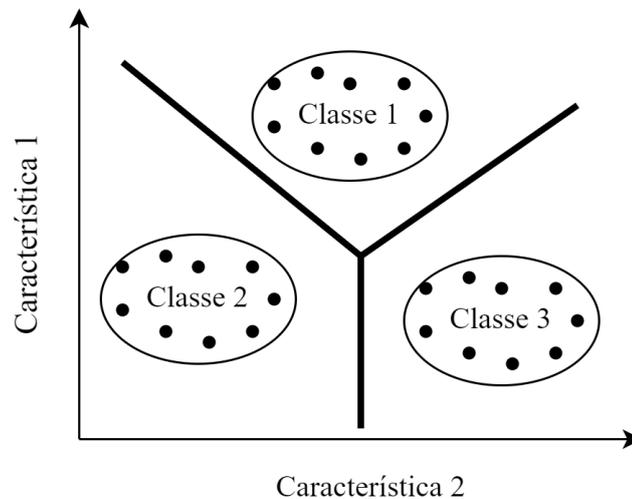


Fonte: Produzido pelo autor.

Outro gráfico muito utilizado para análise de características de algoritmos de IA é o gráfico de limiares de decisão. Nesse gráfico são elencadas duas características usadas na classificação como eixo x e y, são colocados como pontos do gráfico as amostras usadas durante a fase de validação ou de teste e é feita uma divisão espacial, onde o gráfico é dividido em setores que representam as classes previstas pelo algoritmo, a partir da disposição das amostras classificadas pelo algoritmo, como mostrado na figura 6 (LEE; LANDGREBE, 1997).

Essas técnicas, métricas e conceitos de aprendizado de máquina foram, ao longo desse trabalho, como já mencionado anteriormente, aplicadas no contexto de um projeto maior chamado “Usando Treinamento Cognitivo para o Desenvolvimento de Operadores de

Figura 6. Exemplo de gráfico de limiar de decisão.



Fonte: Adaptado de Lee e Landgrebe (1997).

Alto Desempenho”. O contexto de desenvolvimento da pergunta do projeto é o seguinte: A mina Complexo S11D Eliezer Batista da mineradora Vale, situada em Canaã dos Carajás, implantou um moderno sistema de treinamento de operadores de escavadeiras de alta capacidade do tipo CAT 7495 (Caterpillar®). Esse treinamento simulado é realizado em simuladores 4D (Immersive Technologies®) que simulam de maneira altamente fidedigna a realidade da operação na mina. Durante a simulação, diversos parâmetros são medidos e o processamento desses parâmetros fornece o necessário para medir o desempenho do operador.

Esse desempenho é simbolizado por uma medida de produtividade fornecida pelo simulador. Essa produtividade é medida em toneladas escavadas por hora durante o treinamento simulado. Para que a classificação seja possível é preciso discretizar as produtividades, transformando-as de valores contínuos em valores discretos que correspondem a classes. O primeiro passo para isso é a determinação da quantidade de classes nos quais os operadores devem ser atribuídos. Como o número de operadores é limitado, um índice de validade não-paramétrica de agrupamento é usado, o qual é chamado de MEC (Mutual Equidistant-scattering Criterion) (FLEXA et al., 2018). O índice MEC é baseado na dispersão equidistante mútua entre os dados dentro do agrupamento, para o ajuste fino do número de classes. A partir da técnica MEC obtivemos um valor de 4 classes. Então foi utilizado um método *K-Means* (MACQUEEN, 1967) para agrupar os operadores em classes de acordo com seu nível de produtividade, respectivamente: “A”, “B”, “C” e “D”, em qualidade decrescente de produtividade.

Além do treinamento simulado também foi realizado um treinamento cognitivo utilizando *neurofeedback*, ou seja, que utiliza sensores no escalpo do indivíduo conectados à softwares que detectam, amplificam e gravam a atividade cerebral e a informam ao

indivíduo treinado quando a sua atividade cerebral se encontra no alcance desejado (WANG; HSIEH, 2013). Esse treinamento é feito através de um jogo em que o operador deve manter uma nave no centro da tela ao satisfazer as seguintes condições: O aumento do ritmo sensorio motor (12-15Hz); A redução das ondas Teta (4-7Hz) e a redução das ondas HiBeta (20-30Hz). Quando o operador atinge as condições desejadas por um tempo determinado, inicialmente de meio segundo, ele recebe um som que serve como informação de que ele está no caminho certo e como um reforço positivo. Esse treinamento foi feito através do software BrainAvatar e os sinais cerebrais do operador, que definem o movimento da nave, foram obtidas usando o capacete neural da empresa Brain Master.

O objetivo do treinamento cognitivo é condicionar os participantes a controlar a energia do seu ritmo sensorio motor (SMR). Esse treinamento utiliza *neurofeedback* para estimular o aumento de SMR tanto em frequência quanto em duração e foi escolhido porque tem efeito positivo na regulação da atenção (TIMMERS, 2014), que é parte essencial na segurança e eficiência do trabalho com escavação de minérios. Esse ritmo cerebral é específico da parte mais posterior do lobo frontal cérebro e é comumente usado no tratamento de epilepsia para reduzir a atividade cerebral durante as crises epiléticas e no tratamento de TDAH para diminuir a excitação e regular a atenção, mas já foi até usado para melhoria da produtividade de indivíduos saudáveis em contexto de esporte (TIMMERS, 2014; PAUL et al., 2012).

Posto isso, a principal pergunta do projeto é: É possível classificar a produtividade de operadores de escavadeira de alto desempenho da mina complexo S11D Eliézer Batista a partir de suas atividade cerebrais medidas através de registros eletroencefalográficos em estado de repouso?

Para responder essa pergunta foram necessários 4 passos: 1. Pré processar os dados de EEG coletados; 2. Desenvolver uma rede neural artificial capaz de extrair características desses dados de EEG; 3. Treinar um grupo de classificadores concorrentes usando as características extraídas e selecionar o que melhor performa; 4. Avaliar se a classificação usando as características obtidas é adequada para a classificação em níveis de produtividade.

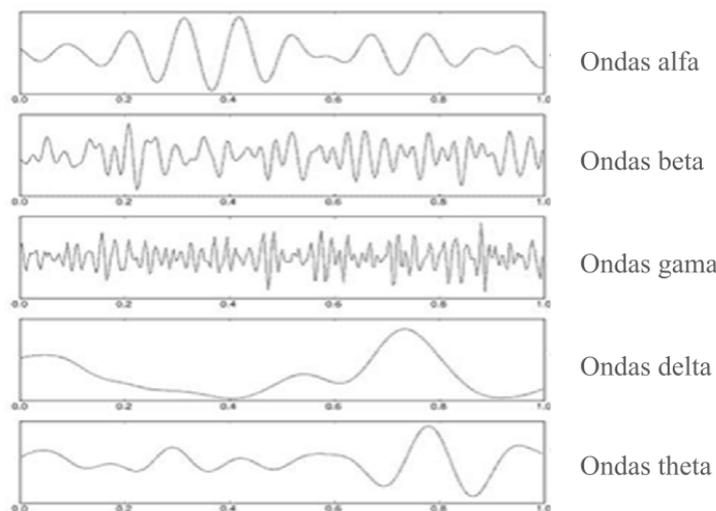
## 1.1 Uma breve história do EEG

Como já mencionado, o eletroencefalograma, ou EEG, é um método não invasivo de usado para monitorar as atividades elétricas cerebrais em grande escala através de eletrodos externos, sendo hoje um dos métodos mais amplamente utilizado em laboratórios clínicos e psicológicos para monitorar não invasivamente a atividade cerebral (SUN et al., 2020). Para realizar esse monitoramento ele se baseia nas variações de voltagem captadas por diferentes eletrodos. Por conta de toda a interferência e atraso gerada pelo crânio, a pele e as demais partes do corpo humano entre os neurônios e os eletrodos, o EEG do

couro cabeludo é uma versão aproximada, espaço-temporal, do potencial de campo local integrado em uma grande área, mas, em contraste com a sua excelente resolução temporal, a resolução espacial é muito limitada (GOMES, 2015).

A história do EEG começa com um pesquisador chamado Richard Caton, que descobriu em 1875 a existência de correntes elétricas no cérebro e coletava o EEG de cérebros expostos de coelhos e macacos. Mas foi em 1924 que um neurologista alemão chamado Hans Berger usou seu equipamento comum de rádio para amplificar a atividade elétrica cerebral medida no couro cabeludo humano, anunciando, em seguida, que a corrente elétrica fraca gerada no cérebro podia ser gravada sem a abertura do crânio e também poderia ser retratada graficamente em uma folha de papel. Naturalmente a atividade que ele observou mudava de acordo com o estado do cérebro no momento da observação, como por exemplo, em caso de sono, anestesia, falta de oxigênio e em certas doenças cerebrais, como a epilepsia. Com o tempo foram percebidos padrões na atividade cerebral humana e essa atividade foi categorizada em quatro grupos básicos de acordo com a sua frequência, como mostrado na figura 7 (TEPLAN et al., 2002).

Figura 7. Ondas cerebrias com frequências dominantes pertencendo a beta, alpha, theta e delta.



Fonte: (TEPLAN et al., 2002).

Cada um desses grupos representa um alcance específico de frequências, sendo elas: gama (>30 Hz) beta (30-12 Hz), alfa (8-12 Hz), theta (4-8 Hz) e delta (0.5-4 Hz). Dentre eles o grupo de frequência mais estudado é o alfa, que normalmente é induzido pelo relaxamento mantendo os olhos fechados e é muito associado ao relaxamento, meditação leve, aprendizado e concentração. Existem ritmos, no entanto, que não estão necessariamente dentro de nenhum desses alcances, como o ritmo sensorio motor (SMR), que está associado a frequências de 12 a 15Hz no cortex motor é importante no contexto do projeto no qual esse trabalho está inserido e é muito utilizado em treinamentos que usam

o *neurofeedback*. Esse padrão cerebral ocorre durante um estado mental alerta, mas em que não há movimento. O treinamento usando o SMR tem um efeito positivo na regulação da atenção e tende a reduzir a atividade motora. Esse treinamento tem sido utilizado para a redução de crises de epilepsia e para a regulação da atenção em casos do transtorno do déficit de atenção e hiperatividade (TIMMERS, 2014).

## 1.2 Aprendizado de máquina para classificação de EEG

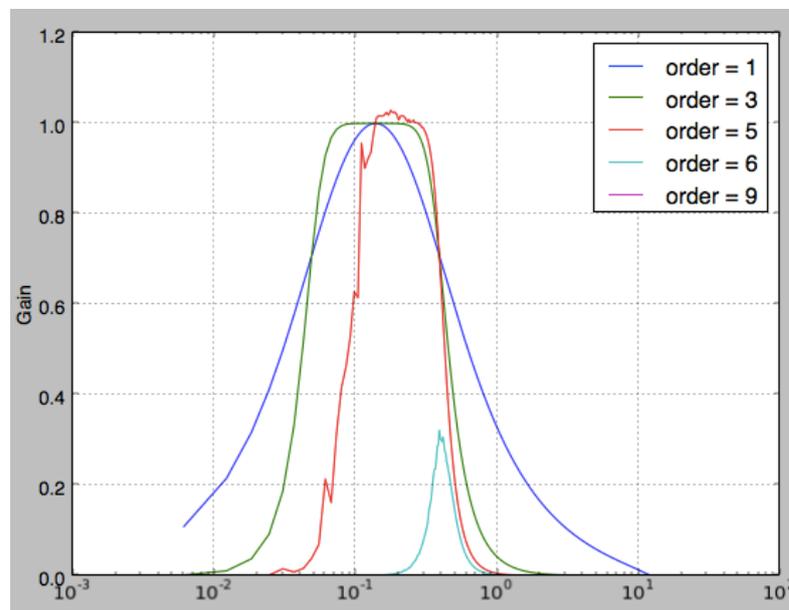
Atualmente o EEG é muito usado em pesquisas envolvendo neurologia, prática clínica, engenharia biomédica, análise do sono e detecção de convulsões, por conta de sua alta resolução temporal, por ser uma técnica não invasiva e pelo seu baixo custo. Muitas pesquisas envolvendo o EEG também envolvem o aprendizado de máquina, especialmente no que se chama *decoding*, que corresponde a conhecer um dado parâmetro do EEG que está bem correlacionado a um comportamento específico. Outra contribuição importante do aprendizado de máquina para o uso de EEG para pesquisa é a classificação automática desses sinais uma vez que ela permite a aplicação prática desse exame em grande escala, de forma que essa aplicação seja menos dependente de profissionais treinados para realizar avaliar esse tipo de exame. Nesse ponto o aprendizado de máquina pode ser muito útil, por possibilitar soluções que podem ser muito acuradas para classificação automática de dados de EEG adaptadas para individualmente para o problema abarcado por essas pesquisas, usando diversas técnicas de aprendizado de máquina, como as redes neurais convolucionais, algoritmos baseados em árvore de decisão *autoencoders* (AGGARWAL; CHUGH, 2022). O processo típico de classificação de EEG inclui a remoção de artefatos (na fase que chamamos de pré processamento de dados), a extração de características e, finalmente, a classificação (CRAIK; HE; CONTRERAS-VIDAL, 2019).

A análise e escolha de como lidar com o pré processamento de dados de EEG para a remoção de artefatos é essencial e indispensável para o sucesso de qualquer classificação automática de EEG, visto que esse tipo de dado é não estacionário e tem uma quantidade muito grande de ruído, uma vez que a aquisição desse tipo de sinal acontece sob muitas fontes de ruído ambiental, fisiológico e de movimento, que criam ruídos com grande amplitude que são chamados de artefatos (ROY et al., 2019). Existem várias técnicas para filtragem e redução de ruído e dependendo do objetivo do estudo e da qualidade dos dados utilizados as formas de lidar com ruído podem variar. Em aplicações que usam EEG e *Deep learning*, quando se tem registros mais limpos feitos em ambiente laboratorial controlado a tendência é que os pesquisadores optem por não usar nenhum tipo de filtragem, para que não haja nenhuma perda de informação (por volta de 22% dos estudos recentes) (CRAIK; HE; CONTRERAS-VIDAL, 2019). Já quando esses registros são feitos em lugares menos controlados e com maior quantidade de ruído a tendência é se adotar técnicas de pré processamento para que o ruído presente nos registros não prejudique muito a classificação

(por volta de 29% dos estudos recentes) (CRAIK; HE; CONTRERAS-VIDAL, 2019).

Um tipo de filtragem muito comum é o uso de filtros *butterworth* passa banda, um tipo de filtro digital que usa que busca uma resposta de frequência o mais próximo de plana possível, sem que as nuances do sinal filtrado sejam perdidas (FEDOTOV, 2016). Quando usado como um passa banda esse filtro atua na minimização da magnitude de frequências de sinais muito abaixo de um certo limiar e acima de outro. Esses limiares são definidos pelo profissional que está aplicando o filtro, bem como o quão perto de plano plano o filtro é, que é definido pela ordem do filtro. Um exemplo de filtro *butterworth* passabanda e de como ele muda dependendo da ordem escolhida pela pessoa que o aplicar podem ser vistos na figura 8.

Figura 8. Filtro passa banda e suas variações dependendo da ordem do filtro.



Fonte: stackoverflow (2014).

Anotações: disponível em: <<https://stackoverflow.com/questions/21862777/bandpass-butterworth-filter-frequencies-in-scipy>>. Acesso em: 08 de março de 2023

Outro tipo de filtragem muito utilizada é a técnica do *Independent Component Analysis* (ICA), uma técnica que atua na maximização do grau de independência estatística entre funções aproximadas de contraste da divergência de Kullback-Leibler, uma medida não-simétrica da diferença entre duas distribuições de probabilidade (SUN; LIU; BEADLE, 2005). O ICA é muito usado para remoção de artefatos estereotipados de movimento ocular, movimento muscular e ruídos de linha (NOORBASHA; SUDHA, 2021).

Nos estudos que usam *deep learning* para classificação, depois da filtragem e pré processamento dos dados é importante definir como os dados serão adicionados à rede. Eles podem, de maneira geral, ser adicionados de três formas: 1. Como séries temporais; 2. Como características calculadas e 3. Como imagens. (AL-SAEGH; DAWWD; ABDUL-

JABBAR, 2021; CRAIK; HE; CONTRERAS-VIDAL, 2019). Na primeira forma, quando os dados são adicionados como séries temporais, eles são diretamente adicionado na rede, sem grandes mudanças na forma de organiza-los. Outra maneira é através do calculo de conjuntos de características dos registros, que serão adicionados à rede, usando técnicas como matrizes de correlação, transformadas de wavlet, média, mediana, moda e entropia (AL-SAEKH; DAWWD; ABDUL-JABBAR, 2021). Além disso, quando usadas redes eficientes para classificação de imagens, os registros podem ser organizados para serem adicionados nas redes na forma de imagens, na forma de matrizes tridimensionais, ou mapas de transformadas, por exemplo (CRAIK; HE; CONTRERAS-VIDAL, 2019).

Existem vários tipos de redes neurais que podem ser usadas para classificação de EEG, mas o mais comum são as redes neurais convolucionais (ROY et al., 2019; AL-SAEKH; DAWWD; ABDUL-JABBAR, 2021; CRAIK; HE; CONTRERAS-VIDAL, 2019). Esse tipo de rede foi desenvolvido para processar dados que vem na forma de múltiplos arrays, como no processamento de imagens, por exemplo (LECUN; BENGIO; HINTON, 2015). Esse tipo de rede normalmente são formadas por combinações de dois tipos principais de camadas, as camadas convolucionais e de *pooling*. As camadas convolucionais organizam suas entradas em mapas de características, a partir de pesos locais e seu objetivo é detectar conjunções de características da camada anterior. O papel das redes de *pooling*, por sua vez, é de fundir características semelhantes em uma só (LECUN; BENGIO; HINTON, 2015).

As redes convolucionais usadas e tem parâmetros que podem ser otimizados, como o número de camadas da rede, o tipo de cada camada e o número de neurônios presentes em cada camada. Muitas técnicas são usadas nessa quesito, como *Stochastic Gradient Descent*, *Adam*, Gradiente conjugado e otimização bayeziana (ROY et al., 2019; AL-SAEKH; DAWWD; ABDUL-JABBAR, 2021).

Existem várias aplicações para a combinação do aprendizado de máquina com os dados de EEG seja, como no caso do nosso trabalho, na classificação de níveis de produtividade, seja na detecção de convulsões em pacientes epiléticos ou no reconhecimento de emoções (CRAIK; HE; CONTRERAS-VIDAL, 2019; AL-SAEKH; DAWWD; ABDUL-JABBAR, 2021).

## 2 Justificativa

Como uma consequência da inserção de princípios e técnicas da neurociência em grandes empresas esse projeto faz parte de um projeto maior criado pela mineradora Vale S. A., que visa melhorar a eficiência do trabalho dos operadores de escavadeiras de alta capacidade, através da criação de treinamentos personalizados a partir da classificação de seus dados de EEG. Esses operadores trabalham usando escavadeiras do modelo CAT 7495 que tem alta capacidade e são controladas por um único operador, com um segundo operador que pode, eventualmente, estar presente se necessário e sempre está a postos. Usando a escavadeira, o operador cava o minério de ferro da lavra e move esse minério para a um repositório chamado moega, que faz o transporte (e seleção) do minério através de correias para que finalmente sejam depositados em um pátio. A partir desse ponto, caminhões podem coletar esse minério de ferro e levá-lo aos trens que fazem o transporte para portos.

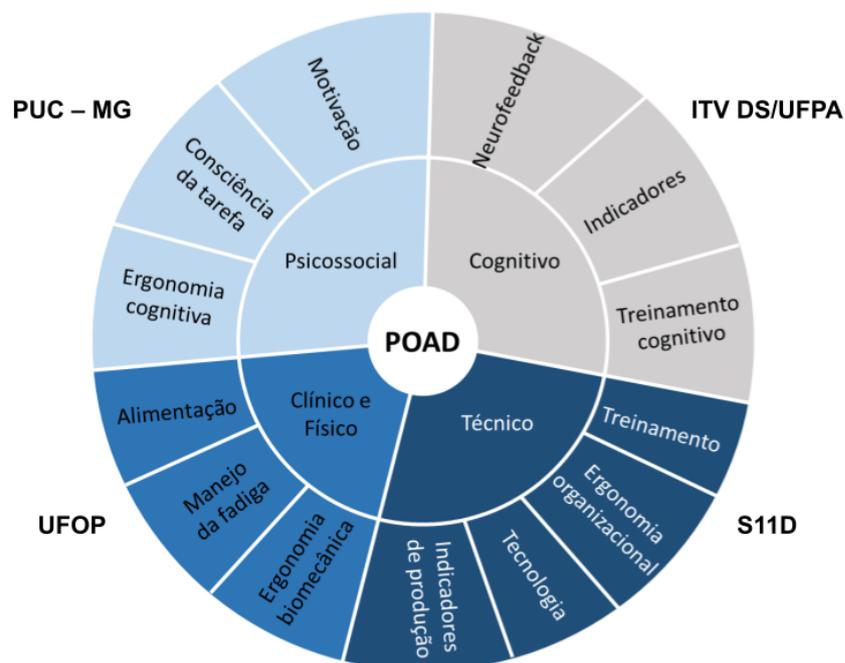
A jornada de trabalho desses operadores é de 8 horas por dia, cinco dias por semana, como é muito comum no mercado de trabalho. No entanto, esse é um tipo de trabalho que exige alto grau de atenção sustentada para que a precisão da operação seja mantida em um nível constante. Para citar alguns dos principais problemas que podem ocorrer caso a precisão não seja mantida tem-se: I. Perda de material devido choques do carregador da escavadeira com a moega, o que pode gerar insegurança a integridade física do operador, bem como prejuízos elevados a empresa. II. Quantidade inadequada de minério sendo carregada na escavadeira, o que obviamente leva a prejuízos. III. Falta de atenção aos procedimentos de segurança, que levam a ameaças a integridade física dos operadores.

O trabalho de um operador de escavadeira de alto desempenho não é como qualquer outro. A saúde física e mental desses operadores precisa ser monitorada e quando identificados problemas de saúde, estes precisam ser corrigidos. Essas pessoas precisam ser bem assessoradas quanto aos cuidados com a saúde. Esses operadores são um pequeno número quando comparadas ao número de pessoas que compõe outros grupos de trabalhadores, mas suas atividades rotineiras são diretamente responsável por parcela muito significativa do produto interno bruto do Brasil e, portanto, direta e indiretamente responsável pela manutenção de muitos empregos.

Posto isso, a Vale criou o Programa de Operadores de Alto Desempenho (POAD). A figura 9 ilustra as várias partes do POAD, cada uma encarregada de ações focais em diferentes aspectos da saúde do operador. Esse programa visa uma abordagem multidisciplinar para instituir uma rotina de cuidados a saúde dos operadores. Ele engloba cuidados com a saúde mental dos operadores através da criação de um Núcleo Psicos-

social, que objetiva detectar e caracterizar possíveis problemas psicológicos que podem ocorrer com os operadores. Esse núcleo esteve sob responsabilidade de pesquisadores da pontifícia universidade católica de Minas Gerais (PUC-MG). O Núcleo Clínico Físico, sob responsabilidade de pesquisadores da Universidade Federal de Ouro Preto, esteve focado principalmente no manejo de fadiga através de seus aspectos ergonômicos e biomecânicos, bem como aconselhamentos quanto a nutrição dos operadores e detecção de indicadores clínicos de dislipidemia e síndrome metabólica não tratada. O Núcleo Técnico, composto de engenheiros e demais técnicos do complexo S11D, responsável pela seleção e treinamento técnico dos operadores. E finalmente, o Núcleo Cognitivo, sob responsabilidade do ITV-DS e Universidade Federal do Pará (UFPA), e do qual fazemos parte. Esse núcleo está responsável por detectar fatores cognitivos usando a eletroencefalografia (EEG), bem como usar o treinamento cognitivo com *neurofeedback* para corrigir eventuais problemas cognitivos como atenção ou memória de trabalho prejudicadas. Nossa tarefa é a de decodificar o sinal cerebral usando o EEG (*Brain decoding*) e agir para colocar essa resposta (quando possível), dentro de parâmetros aceitáveis.

Figura 9. Esquemático das áreas de trabalho e seus responsáveis no Programa de Operadores de Alto Desempenho (POAD).



Fonte: Relatório Técnico do Treinamento Cognitivo - Agosto 2020

Nesse contexto a interdisciplinaridade do projeto é essencial, dado que um problema cognitivo tem razões multifatoriais. Ele pode ocorrer devido a problemas psicológicos de natureza crônica (ansiedade generalizada, depressão). Pode ocorrer devido problemas relacionados a síndrome metabólica (diabetes). Pode ocorrer devido alterações cardiovasculares (pressão arterial alta) e pode ocorrer devido a excesso de trabalho, dentre outras

razões. Dessa forma a melhoria das baixas de produtividade deve ser analisada por diversas áreas e detectada o quanto antes. Nesse sentido a detecção dessas baixas de produtividade usando somente dados de EEG pode ser muito vantajosa, ao acelerar essa detecção e, conseqüentemente, proporcionar às equipes responsáveis um tempo de resposta maior.

Esse projeto, portanto, tem todo o potencial de não se limitar a classificar a produtividade diária desses operadores, mas também contribuir ao longo de seu desenvolvimento futuro, melhorar a qualidade de vida e de trabalho desses profissionais, através da criação de treinamentos e rotinas personalizadas, vindas da coleta e análise dos dados cognitivos desses operadores.

## 3 Objetivos

### 3.1 Objetivo Geral

Classificar a produtividade de operadores de escavadeira de alto desempenho da mina complexo S11D Eliézer Batista a partir de suas atividade cerebrais medidas através de registros eletroencefalográficos em estado de repouso.

### 3.2 Objetivos Específicos

- I. Pré processar os dados de EEG coletados usando filtragens passa-banda e ICA.
- II. Desenvolver uma rede neural artificial, otimizada usando otimização bayeziana, capaz de extrair características dos dados de EEG
- III. Treinar um grupo de classificadores usando as características extraídas e selecionar o que melhor performa entre eles.
- IV. Avaliar se a classificação usando as características obtidas é adequada para a classificação em níveis de produtividade.

## 4 Materiais e Métodos

### 4.1 Dados utilizados

Os dados utilizados foram coletados em parceria com a mineradora Vale, especificamente na mina complexo S11D Eliezer Batista, situada em Canaã dos Carajás. Lá foi implantado um moderno sistema de treinamento de operadores de escavadeiras de alta capacidade do tipo CAT 7495 (Caterpillar®). Esse treinamento faz parte da rotina de trabalho dos operadores e é realizado em simuladores 4D (Immersive Technologies®) que aproximam tanto quanto possível a operação real. Durante a simulação, diversos parâmetros são medidos e o processamento desses parâmetros fornecem o necessário para medir o desempenho do operador. No contexto do nosso trabalho, no entanto, usaremos apenas a quantidade de toneladas escavadas em uma hora no simulador para avaliar a performance dos operadores. Para este trabalho foram coletados os registros de EEG de 16 operadores (com  $37 \pm 7$  anos de idade). Esses operadores podem ser classificados em 4 classes de performance que são de A a D, sendo A a melhor performance e D a pior, podendo ser vistas na tabela 2. Os limites de produtividade que distinguem as classes foram feita pela empresa e entregue à nós. Para adicionar esses dados de classes de performance no classificador, no entanto, precisamos transformá-los em valores numéricos, por isso essas classes de A a D também são identificadas como de 0 a 3 no trabalho.

Tabela 2. Tabela de classes de produtividade em toneladas por hora.

Classificação	Classe	Produtividade Mínima	Produtividade Máxima	N. de Registros
Excelente	A ou 0	10,000 t/h	∞	6
Boa	B ou 1	8,800 t/h	10,000 t/h	32
Mediana	C ou 2	7,500 t/h	8,800 t/h	14
Ruim	D ou 3	∞	7,500 t/h	8

Fonte: Produzido pelo autor.

Por conta da escassez de dados para uso em uma rede neural robusta também criamos uma segunda classificação, dividida em somente duas categorias, sendo elas boa (que contém os registros referentes às classes excelente e boa da tabela 2) ou ruim (que contém os registros referentes às classes mediana e ruim da tabela 2), como pode ser visto na tabela 3. As duas classificações foram utilizadas ao longo do trabalho e seus resultados serão mostrados separadamente no capítulo 5.

Para o produto dessa dissertação foram selecionados 16 operadores com uma média de  $36 \pm 7$  anos de idade que, cujos dados de EEG foram coletados antes e depois de

Tabela 3. Tabela de classes de produtividade em toneladas por hora.

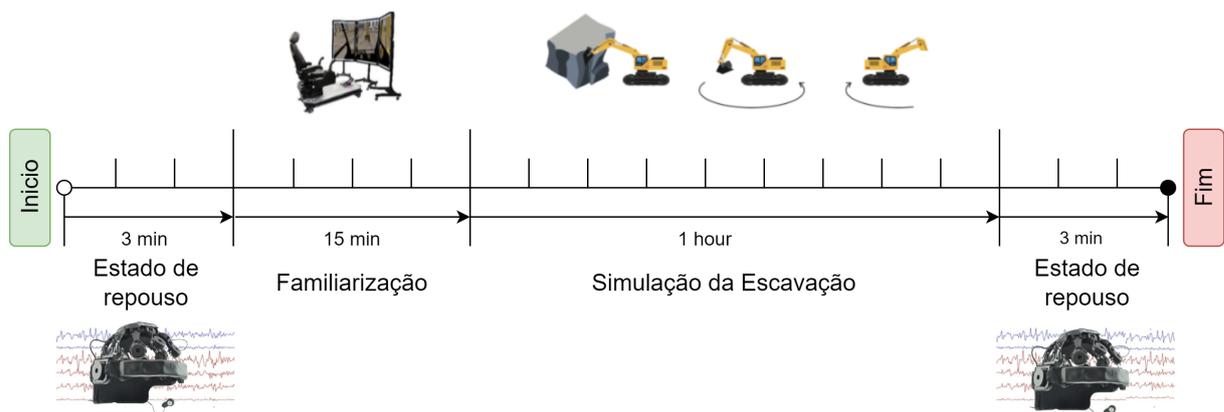
Classificação	Classe	Produtividade Mínima	Produtividade Máxima	N. de Registros
Boa	A ou 0	8,800 t/h	∞ t/h	38
Ruim	B ou 1	∞ t/h	8,800 t/h	22

Fonte: Produzido pelo autor.

seu treinamento usando o simulador 4D. Para coleta do EEG em estado de repouso os operadores foram acomodados confortavelmente na cadeira do simulador, e posicionados em frente ao ambiente de simulação que consiste em uma parede branca. Nenhum desses operadores relatou possuir doenças crônicas ou serem usuários de medicamentos contínuos ou ansiolíticos. A coleta de dados de EEG foi avaliada pelo comitê de ética do Instituto de Ciências da Saúde da Universidade Federal do Pará, através do projeto de nome: Utilizando Treinamento Cognitivo para o Desenvolvimento de Operadores de Alto Desempenho, submetido pelo professor Bruno Duarte Gomes e foi aprovado pelo parecer de número 3.601.269 em setembro de 2019.

O protocolo de coleta de dados consistiu, em um primeiro momento, manter o olhar fixo imediatamente a frente durante 3 minutos. Na sequência, o operador possui 15 minutos para se familiarizar com o ambiente de simulação. Na sequência, são capturados 60 minutos de simulação sem interferências externas, mas que não entraram no escopo desse trabalho. Imediatamente após a sessão são capturados mais 3 minutos de dados em repouso. A Figura 10 ilustra essa sequência de passos do protocolo.

Figura 10. Protocolo de coleta de dados cognitivos e técnicos



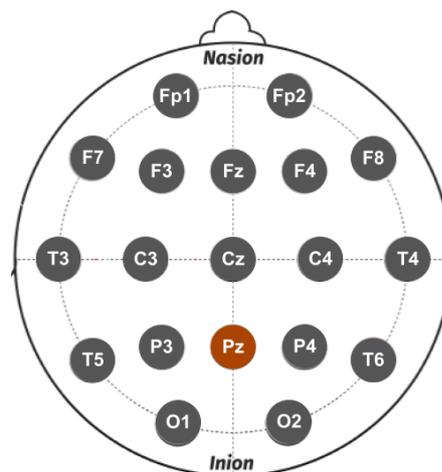
Fonte: Produzido pelo autor.

Principalmente durante os períodos de descanso orienta-se que sejam evitadas deglutições, piscadas excessivas e movimentos voluntários. Também, durante do experimento de escavação foi solicitado que movimentos abruptos fossem evitados, assim como movimentos faciais, maxilares, ou mesmo dos membros inferiores.

É importante frisar que o treinamento ao qual esses operadores foram submetidos é um treinamento de rotina que eles fazem quando precisam treinar. A única diferença é no nosso experimento que eles tem seus registros de EEG medidos durante a execução do treinamento e tem de permanecer por 3 minutos em estado de repouso para coleta de EEG antes e depois do treinamento.

Dos 16 operadores selecionados 2 deles só puderam fazer o registro em um dia e os outros 14 puderam fazer registros em 2 dias diferentes, tendo um espaço de duas semanas entre os registros e podendo ter classes de produtividade diferentes associadas a esses dois registros, uma vez que houveram algumas semanas entre eles. Tivemos um total de 30 seções de coleta de dados e cada uma delas com 2 registros de 3 minutos de EEG, um antes da simulação e um depois da simulação, como indicado pela figura 10. Terminamos então com 60 registros de EEG, contendo 3 minutos de sinal cada. Esses registros foram coletados através do software DSI-streamer (Wearable sensing®), que recebia os dados fornecidos por um capacete neural Freedom 24D (Brain Master Technologies®) que os exporta em formato European Data Format (edf), um formato muito utilizado para troca e armazenamento de séries temporais médicas. Esse aparelho é um capacete sem fio ajustável que possui 21 eletrodos secos, 3 eletrodos secos opcionais removíveis e 1 trigger digital de 8 bits e transmite o EEG para o software de captura com uma taxa de amostragem de 300 Hz. Um esquemático contendo os eletrodos que foram utilizados neste trabalho, seguindo o padrão do capacete neural utilizado para a coleta dos dados de EEG pode ser encontrado na figura 11.

Figura 11. Eletrodos presentes no equipamento BrainMaster 24D.



Fonte: BrainMaster (2018).

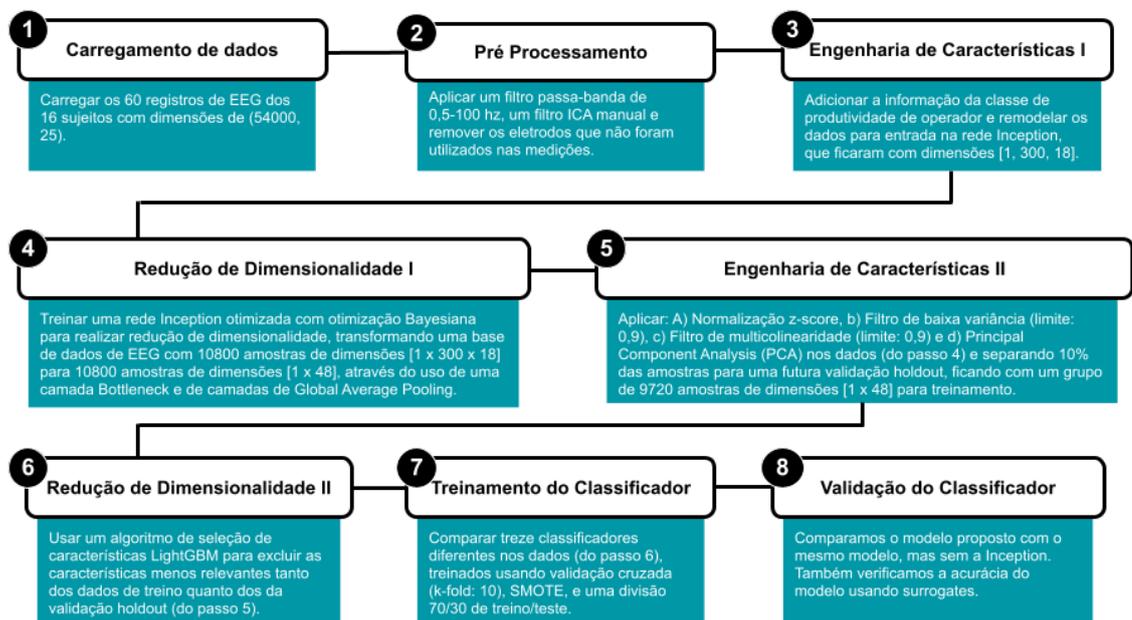
Dos canais de referência testados, os canais A1 e A2, também chamados *linked ears* não apresentaram bons resultados para serem utilizados como referência, visto a quantidade excessiva de ruído. Desta forma, o canal Pz foi considerado como referência, restando 18 canais de sinal EEG útil. Portanto coletamos todos os eletrodos presentes na

figura 11, usando o eletrodo Pz como referência para os demais. Esses dados então foram importados no software Matlab®, na versão 2021A, onde foi feito o pré processamento dos dados usando uma biblioteca chamada EEGLAB (DELORME; MAKEIG, 2004), que é uma ferramenta específica para tratamento e processamento de dados eletroencefalográficos criada em 2004 e constantemente atualizada até hoje.

## 4.2 Previsão de performance usando a rede inception

Para contemplar o objetivo principal desse trabalho, de construir um modelo empírico que classifique a produtividade dos operadores de escavadeiras de alta capacidade a partir de seus EEGs, foi necessário criar um método de várias partes que classifica os dados de EEG desses operadores. O processo completo contém 8 partes que vão desde o carregamento dos dados até a validação final do modelo híbrido treinado, como mostrado na figura 12. O algoritmo criado foi disponibilizado no github e pode ser acessado através do link: <<https://github.com/labneoc/EEG-Inception-Modeling>>.

Figura 12. Fluxo de Modelagem do Machine Learning



Fonte: Produzido pelo autor.

### 4.2.1 Carregamento e pré processamento dos dados

Para construir o modelo de classificação de performance dos operadores via EEG primeiramente foi necessário importar e pré processar esses dados no algoritmo criado. Como já mencionado, os dados de EEGs estavam em formato EDF. Eram um total de

60 registros, sendo 30 deles de antes da simulação e 30 deles de depois da simulação que, ao serem carregados, tinham dimensões de [54000,25], contendo 25 canais de EEG de 3 minutos de sinal que estava sendo capturado com uma frequência de 300Hz.

Em seguida esses dados foram carregados em uma ferramenta chamada EEGLAB, aonde foram pré processados. Dos 25 canais presentes nos arquivos EDFs apenas 18 eram úteis, pois o Pz foi usado como referência e os outros 6 eram eletrodos opcionais do sistema que não foram utilizados na coleta de dados e, portanto, eram compostos apenas por ruído, ou eletrodos normalmente usados para referência, como os eletros A1 e A2, mas que não foram usados, tendo sido, por conta disto, retirados durante o pré processamento. Os 18 eletrodos utilizados são C3, C4, Cz, F3, F4, F7, F8, Fz, Fp1, Fp2, P3, P4, T3, T4, T5 e T6, todos usando Pz como referência, de acordo com o esquemático mostrado na figura 11. Em seguida foi passado um filtro passa-banda *butterworth* de 0,5-100Hz, para remoção de ruído com frequências muito diferentes das quais estávamos interessados, seguido pela filtragem usando *Independent Component Analysis* (ICA), que foi utilizada para remover artefatos estereotipados de movimento ocular, movimento muscular e ruídos de linha (NOORBASHA; SUDHA, 2021). Os dados de EEG terminam esse passo na forma de 10800 amostras de dimensões [54000, 18], que contêm os 3 minutos de registro eletroencefalográfico dos 16 operadores.

#### 4.2.2 Engenharia de Características e Redução de Dimensionalidade I

Depois do pré processamento os dados são extraídos do EEGLAB, novamente em formato EDF, e são convertidos para um arquivo no formato *Comma-separated values* (CSV), como matrizes de dimensões [54000,18] através de um programa criado por nós no MATLAB®. Então os dados são importados para um programa no Python, como exemplificado na figura 13, onde eles foram processados e adicionados ao classificador, juntamente com a informação da classe de produtividade desse operador, definida a partir de sua produtividade durante a uma hora de treinamento que aconteceu entre os registros de 3 minutos de EEG coletados. Essa classificação foi dada pela própria equipe da Vale baseada na quantidade de toneladas de minério extraídas durante a uma hora de treinamento realizada, como especificado na tabela 2.

Para adicionar essas matrizes à nova arquitetura de Machine Learning criada para processamento desses dados foi necessário mudar a forma em que os dados estavam organizados, de forma que as matrizes não estivessem mais nos eixos x e y e sim nos eixos y e z. Essa mudança pode ser vista na figura 14.

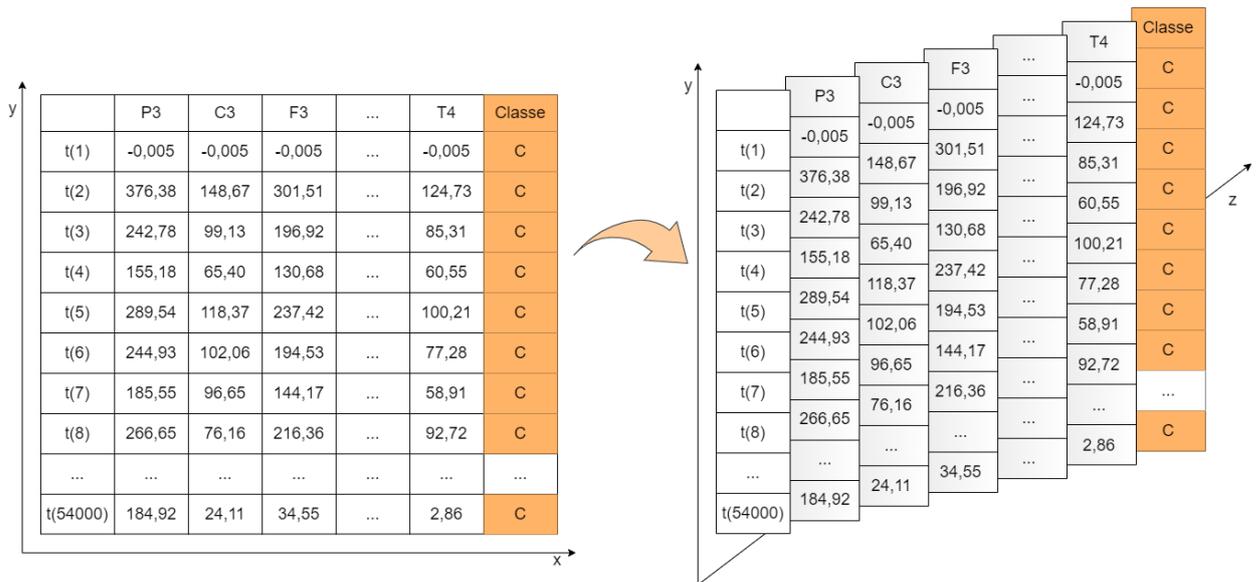
Uma vez que os dados foram preparados para serem incorporados pelo classificador precisamos passar esses dados do Matlab®, ambiente em que os dados foram carregados e processados, para o Python, linguagem em que o classificador foi construído. Para isso compilamos todos os registros 60 de EEG e as classes as quais eles pertenciam em

Figura 13. Matriz de dados de EEG para inserção no classificador cognitivo

	P3	C3	F3	...	T4	Classe
t(1)	-0,005	-0,005	-0,005	...	-0,005	C
t(2)	376,38	148,67	301,51	...	124,73	C
t(3)	242,78	99,13	196,92	...	85,31	C
t(4)	155,18	65,40	130,68	...	60,55	C
t(5)	289,54	118,37	237,42	...	100,21	C
t(6)	244,93	102,06	194,53	...	77,28	C
t(7)	185,55	96,65	144,17	...	58,91	C
t(8)	266,65	76,16	216,36	...	92,72	C
...	...	...	...	...	...	...
t(54000)	184,92	24,11	34,55	...	2,86	C

Fonte: Produzido pelo autor.

Figura 14. Mudança de eixos na matriz de dados de EEG



Fonte: Produzido pelo autor.

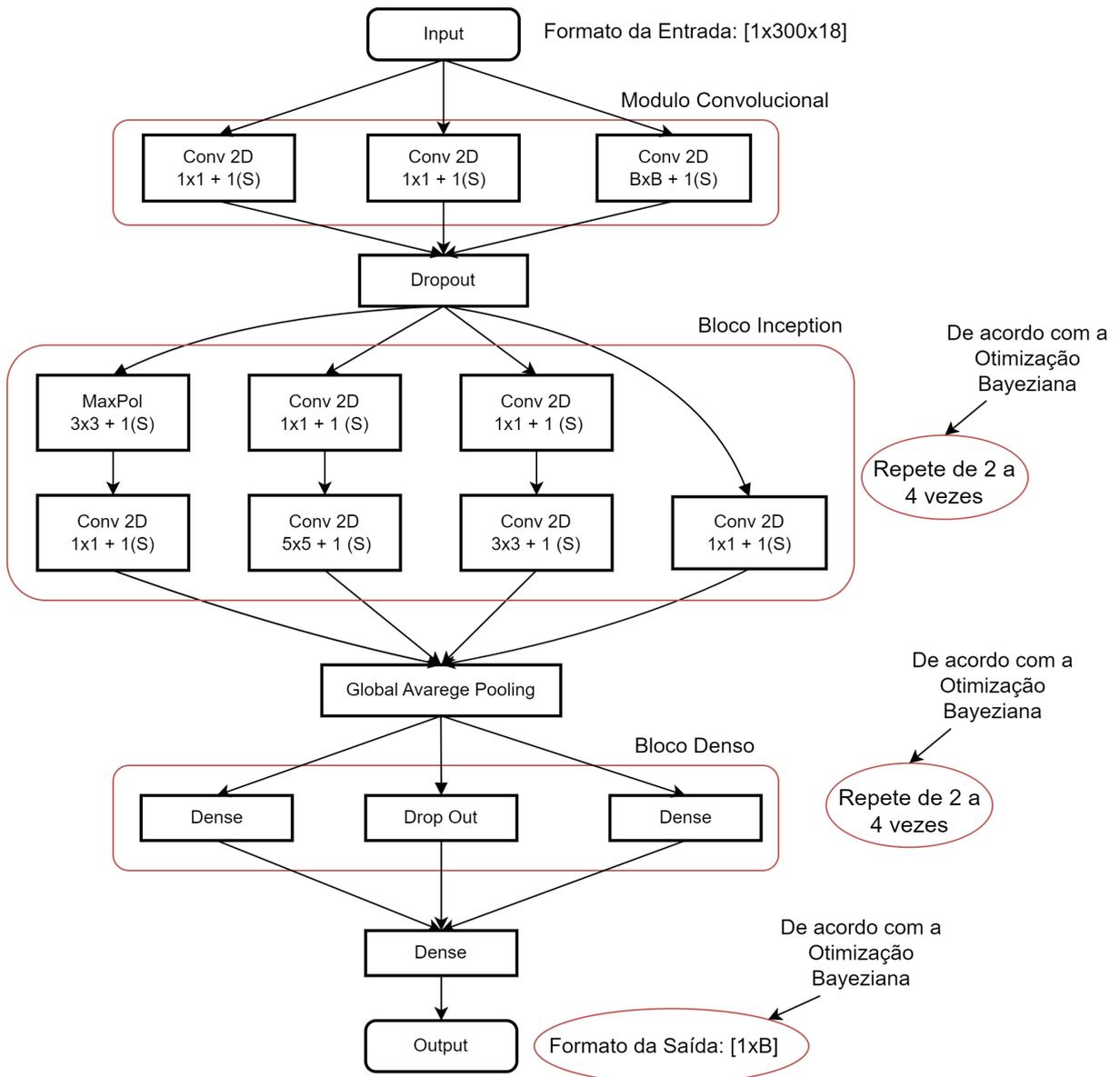
um arquivo no formato CSV que ficou com dimensões [3240000,19]. Em seguida esses registros foram organizados para entrarem no classificador em formato de 10800 matrizes de dimensões [300,19], cada uma contendo um segundo de sinal eletroencefalográfico de 18 eletrodos, com mais uma coluna contendo a classe do operador, que será usada apenas para treinamento e validação da rede.

Uma vez que esse arquivo, foi carregado no classificador chegamos no passo 4 presente na figura 12, no qual treinamos uma rede neural artificial para extrair as caracte-

rísticas mais importantes dos dados criados, para que possam ser interpretados de maneira mais acurada e eficiente pelo classificador. Para isso utilizamos uma arquitetura de rede neural convolucional do tipo *Inception*, inspirada pelo modelo GoogLeNet, proposto em 2015 (SZEGEDY et al., 2015).

A figura 15 mostra a arquitetura otimizada do modelo *Inception* utilizado para extração de padrões existentes nas amostras representadas pela figura 14.

Figura 15. Modelo *Inception* otimizado para extração de padrões dos dados de EEG



Fonte: Produzido pelo autor.

A rede proposta na figura 15 pode ser dividida em três tipos de estruturas diferentes, sendo elas: Camadas, blocos e módulos. As camadas são as menores estruturas da rede, representam suas camadas e são representadas na figura por caixas, com o tipo da camada especificada em seu interior (ex. Convolutacional 2D, MaxPolling ...). Já os blocos são grupos

de camadas que podem ser empilhados em outros blocos idênticos para maior eficiência. Os módulos, por sua vez, são grupos de camadas semelhantes ou de blocos empilhados que têm a mesma função e trabalham uns com os outros para cumprir um objetivo.

Nessa rede, temos 3 módulos principais: Um módulo convolucional, um módulo *inception* e um módulo denso. O módulo convolucional é o primeiro módulo da rede, é composto por três camadas convolucionais e é responsável por iniciar a rede. Uma camada de dropout liga o primeiro módulo ao segundo módulo, denominado módulo *inception*, que é composto por 2 a 4 blocos *inception* empilhados, estruturas esparsas locais de visão computacional, responsáveis por efetuar convoluções 1x1, 3x3 e 5x5, juntamente com um *pooling* paralelo (SZEGEDY et al., 2015). Uma camada de *global average pooling* liga o módulo *inception* ao último módulo, chamado de módulo denso, que é composto por 2 a 4 blocos densos empilhados, cada um composto por 2 camadas densas, totalmente conectadas, e uma camada *drop out*, para resolver o problema do *overfitting*. A saída do módulo denso passa por uma última camada densa e a sua saída é a saída final da rede, uma matriz de tamanho  $[1 \times B]$ , em que B representa o comprimento da matriz e é otimizada através da otimização Bayeziana.

Utilizamos *Gaussian process*, um poderoso algoritmo para predição em amostras com base em informações adquiridas a priori no processo de treinamento. Uma de suas aplicações mais extensivas é a modelagem de funções para um dado conjunto de dados, que ocorre através da atribuição de probabilidades a cada uma destas funções, onde a média desta distribuição de probabilidade representa a mais provável caracterização do conjunto de dados (FRAZIER, 2018). Com isto, podemos estimar com maior precisão a função de aquisição que melhor explorou o espaço amostral de onde foi derivado o conjunto parâmetros que levou à convergência a arquitetura *Inception*. A função de aquisição utilizada foi a *Expected Improvement* (FRAZIER, 2018), em conjunto com o algoritmo de otimização *Limited-memory Broyden-Fletcher-Goldfarb-Shanno (LM-BFGS)* (LIU; NOCEDAL, 1989), o que finaliza nosso módulo de otimização bayeziana, através do qual alcançamos uma taxa de erro na validação de 8.6% (0.086).

Como demonstrado na figura 15, a camada de entrada do modelo *Inception* possui dimensões  $[1 \times 300 \times 18]$ , para se adequar à alta complexidade dos dados ilustrados na figura 14. Desta forma, utilizamos camadas de Max Pooling e uma camada de *GlobalAveragePooling2D* ao longo da rede para reduzir sua dimensionalidade, resultando assim não em uma predição, mas sim um vetor de dimensões  $[1 \times B]$ , em que B é definido pela quantidade de blocos *inception* presentes na rede, quantidade essa definida usando a otimização bayeziana. Esse vetor gerado contém padrões de alta importância preditiva, constituindo um conjunto de dados final de 10800 amostras, pronto para ser inserido em um classificador. A rede *Inception* já é capaz de classificar o dado, mesmo sem a presença de um classificador externo, mas durante os testes que realizamos e que nos fizeram chegar

a essa configuração percebemos que o uso de um classificador mais simples para classificar os vetores com alta densidade de informações, produzidos pela *inception*, tende a ter uma performance melhor, uma vez que a informação já foi processada pela rede e o tamanho do dado já é muito menor. No momento da validação *hold-out* também será testada a rede *inception* sem o uso dos próximos passos propostos pela metodologia exposta nesse trabalho.

### 4.2.3 Engenharia de Características e Redução de Dimensionalidade II

A partir desses vetores de dimensões  $[1 \times B]$  altamente informativos gerados pelo modelo *Inception* foi feita a Engenharia de Características II, onde foram aplicadas algumas técnicas de pré processamento e transformação nos dados para facilitar a convergência do classificador final. Primeiramente normalizamos os preditores utilizando a técnica *z-score*, onde  $z = (x - \mu)/s$ . Em segundo lugar removemos preditores com variância próxima à zero (onde noventa por cento dos valores de uma mesma coluna sejam iguais). No terceiro passo aplicamos uma técnica de remoção de preditores por multicolinearidade. No quarto passo aplicamos o algoritmo de Principal Component Analysis (PCA) para redução de dimensionalidade.

Por último aplicamos o algoritmo Light Gradient Boosting Machine (LightGBM) para calcular a importância de cada coluna da base de dados de características com base na quantidade de informação presente nestes preditores. Tanto os métodos de filtragem de variância e multicolinearidade, quanto a seleção de atributos com o LightGBM servem para mitigar problemas causados pelo erro produzido pela rede *Inception* nas amostras baseadas nos mapas de características dos blocos convolucionais, devido a estocasticidade do modelo. Foi então dividido o total de 10800 amostras em dois grupos, sendo 2160 delas reservadas para serem usados como *holdout* após a validação cruzada e 8640 dessas para treinamento e teste.

### 4.2.4 Treinamento e Validação do Classificador

Utilizamos um classificador baseado em árvore de decisão devido à facilidade de interpretação destes modelos como um todo. Ao contrário de redes neurais artificiais, consideradas soluções "*black-box*", arquiteturas baseadas em árvores de decisão são consideradas "*white-box*", de mais fácil modelagem e entendimento pós-modelagem.

Treinamos um total de treze classificadores que usam diferentes técnicas de *machine learning*, calculando cinco métricas de performance. Da base de dados de 10800 amostras, 20% foram separadas para validação *hold out*, e as outras 80% foram utilizadas para treino e teste usando *cross-validation* sendo  $k\text{-fold} = 10$ .

Além da validação *hold-out* realizada, foi feita uma validação dos métodos propostos

nesse trabalho, de forma a avaliar se os métodos usados em conjunto foram determinantes para o sucesso do modelo. Para realizar essa validação foram comparados três modelos distintos: 1. Classificação usando o método completo proposto nesse trabalho; 2. Classificação usando somente a rede *Inception* e 3. Classificação usando apenas os 13 classificadores que competem entre si. Para avaliar esses modelos foram utilizadas cinco métricas de performance, sendo elas: Acurácia; Precisão; *AUC*; *Recall* e *F1*. Esses modelos de classificação foram, então, treinados, testados e validados com os dados de EEG, usando a mesma validação *hold-out* usada ao longo do trabalho e foram testados tanto para classificação multi-classe quanto para classificação binária.

Outra análise interessante feita durante a fase de validação foi a criação de *surrogates* a partir dos dados de EEG, normalmente utilizadas para testes estatísticos de não-linearidade (VENEMA; AMENT; SIMMER, 2006). Esses dados de *surrogate* são séries temporais lineares que reproduzem várias propriedades estatísticas de uma série temporal usada para criá-las, mas sem suas propriedades não lineares. No caso do nosso trabalho foi utilizado o algoritmo proposto por Venema, Ament e Simmer (2006) chamado de *Stochastic Iterative Amplitude Adjusted Fourier Transform* (SIAAFT), que é uma melhoria do algoritmo proposto por Schreiber e Schmitz (2000). Nesse algoritmo os dados de *surrogates* criados compartilham a mesma distribuição e espectro de potência dos dados usado para gera-los. 100 *surrogates* foram gerados usando esse algoritmo separadamente para cada eletrodo de cada registro, no formato de arrays de dimensões [54000,1], e um desses 100 *surrogates* é selecionado aleatoriamente para compor as matrizes de *surrogates*. Esse processo se repete para todos os eletrodos de cada registro separadamente e para todos os registros, de forma a terminar com 60 matrizes de *surrogates*, de dimensões [54000,18], tendo 1 coluna para cada eletrodo do registro. Após isso essas matrizes de *surrogates* foram utilizadas para treinar um classificador usando toda a metodologia proposta nesse trabalho e sua performance de validação foi avaliada usando as mesmas cinco métricas de performance utilizadas ao longo do trabalho. Isso foi feito para avaliar se as características não-lineares do dado de EEG usado ao longo do trabalho foram determinantes para o sucesso da classificação. Depois de calculados os *surrogates* foram adicionados à rede seguindo a mesma ordem de entrada dos eletrodos usados originalmente para treinamento da rede.

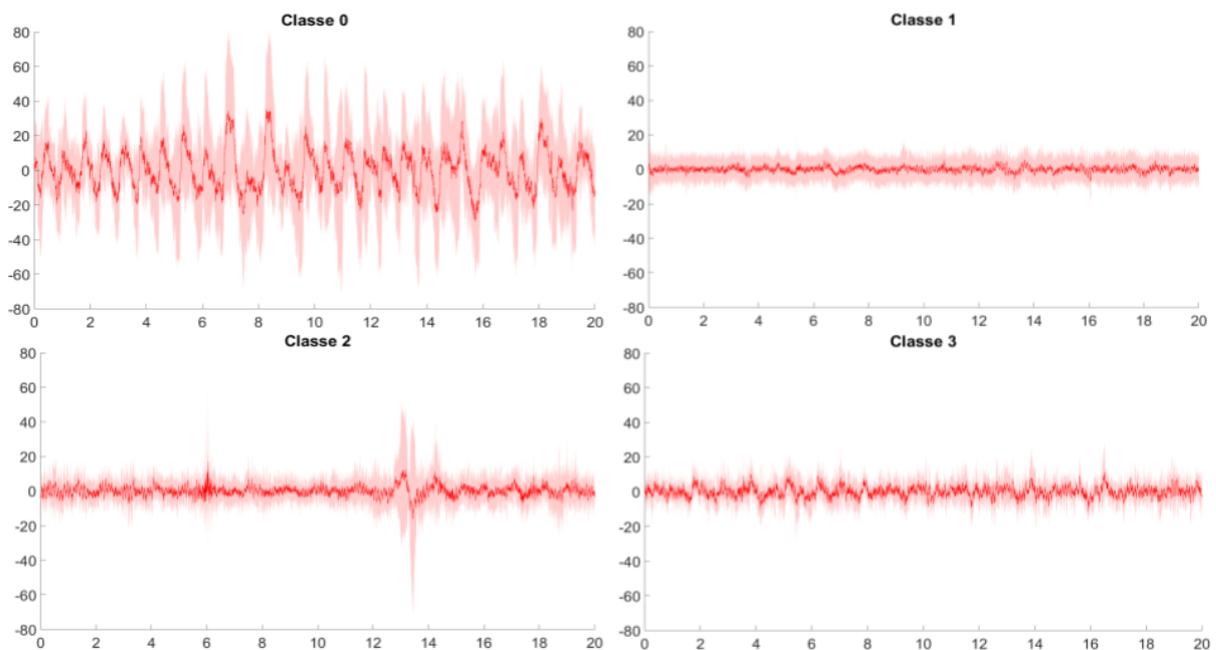
## 5 Resultados

### 5.1 Classificação em quatro classes

#### 5.1.1 Visualização dos dados de EEG por classe

Primeiramente é importante ter uma ideia de como são os EEGs dos operadores pertencentes a cada classe. Nessa linha a figura 16 mostra a média dos EEGs divididos de acordo com a classe de produtividade real dos operadores, usando a classificação em 4 classes, como apresentado na tabela 2.

Figura 16. Média dos registros de EEG, no eletrodo C3, após a filtragem, separados por classe.



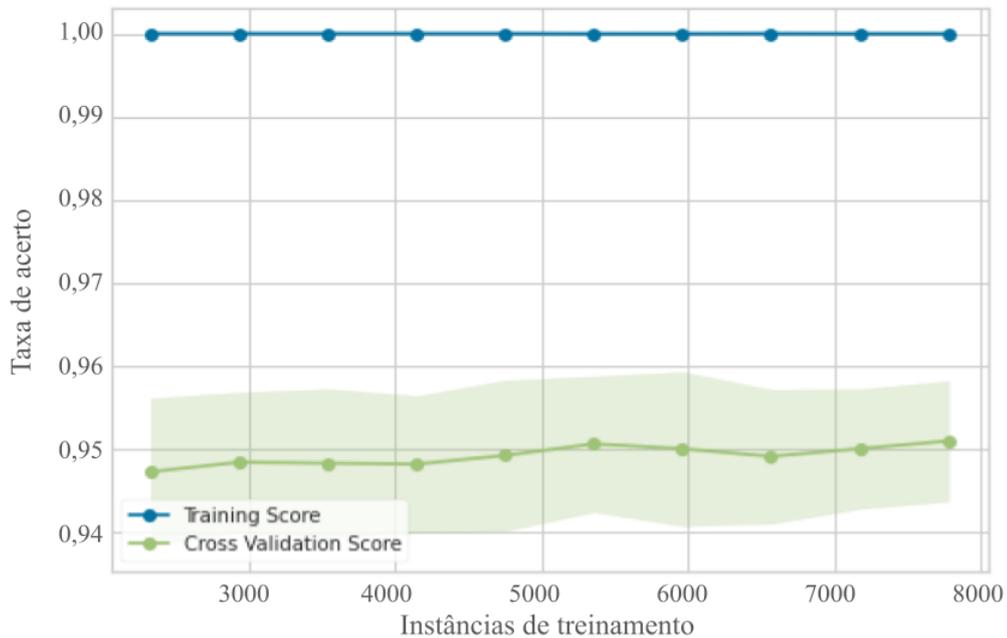
Fonte: Produzido pelo autor.

Nessa figura podemos ver a média dos EEGs dos operadores, em um recorte de 20 segundos do eletrodo C3, representadas pela linha vermelha no centro de cada uma das 4 partes da imagem, divididos por sua classe de produtividade. Na imagem também é possível ver o desvio padrão dessas médias, que é representado através das sombras vermelhas nos gráficos. Nessa imagem é possível observar o padrão oscilatório típico presente no sinal cortical captado através do eletrodo C3, que representa, em sua maioria, a área central do cérebro. É importante ressaltar que essas médias foram calculadas com um número diferente de indivíduos, uma vez que o número de indivíduos não é igual em todas as classes, como mostrado na tabela 2.

### 5.1.2 Curva de Aprendizado

O modelo classificador *Random Forest* (RF) foi o classificador com maior acurácia dentre os testados para a classificação em quatro classes. Esse modelo foi treinado através do procedimento de validação cruzada. Na figura 17 abaixo são exibidas as variações da acurácia durante o treinamento. A linha azul representa o valor da acurácia de treinamento sem usar a validação cruzada e a linha verde representa o valor da acurácia de treinamento, enquanto sua sombra representa seu desvio padrão, usando a validação cruzada.

Figura 17. Curva de aprendizado do modelo RF com e sem *cross-validation*.



Fonte: Produzido pelo autor.

Podemos ver, ao examinar a linha azul, que em menos de 3000 épocas a curva de aprendizado atinge 100% de acurácia e não sai de lá, o que indica a presença de *overfitting* e demonstra a importância da validação cruzada que diminuí muito a chance de *overfitting* no treinamento dos algoritmos. Já na linha verde podemos observar uma acurácia que varia de 94% a 95%, com desvio padrão sempre próximo a 2%, mas com pico de 95,09%, como mostrado na tabela 4.

### 5.1.3 Curvas ROC

A figura 18 exibe um dos métodos mais populares de avaliação de desempenho de modelos de classificação, as curvas ROC, que mostram a performance de classificação do algoritmo por classe do problema. Na figura 18 a seguir, as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

Podemos ver que em todas as classes temos uma pontuação muito boa, possuindo uma quantidade muito maior de positivos verdadeiros do que de falsos positivos, tendo valores de AUC maiores do que 97% em todas as classes.

#### 5.1.4 Limiar de Decisão do Modelo

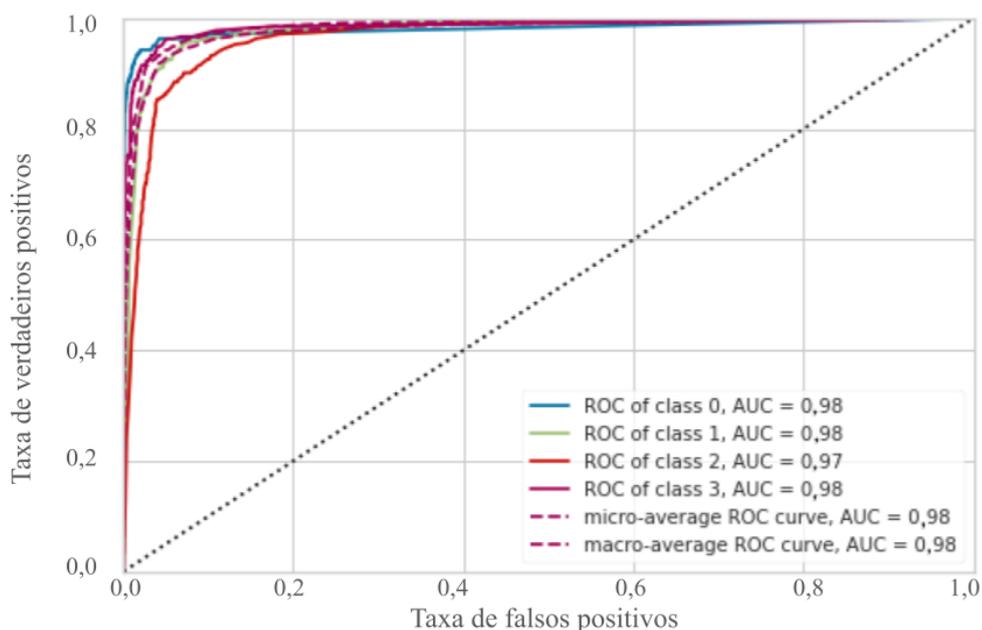
Já na figura 19 são exibidos os limiares de decisão do modelo para predição de cada uma das classes. Podemos observar que o modelo conseguiu classificar as amostras com alta precisão e confiabilidade, errando apenas poucas amostras. Na imagem as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

Podemos ver que o limiar de decisão que ocupa uma maior área na imagem é o referente à classe 1. Isso é esperado, uma vez que essa é a classe com maior quantidade de amostras na base de dados utilizada. É interessante, por outro lado, que a segunda maior área na imagem é referente à classe 0, o que indica que, apesar de ela não ter um grande número de exemplos na base de dados, eles tem um grande variação entre si, pelo menos quando observamos as variáveis representadas pela figura 19.

#### 5.1.5 Matriz de Confusão

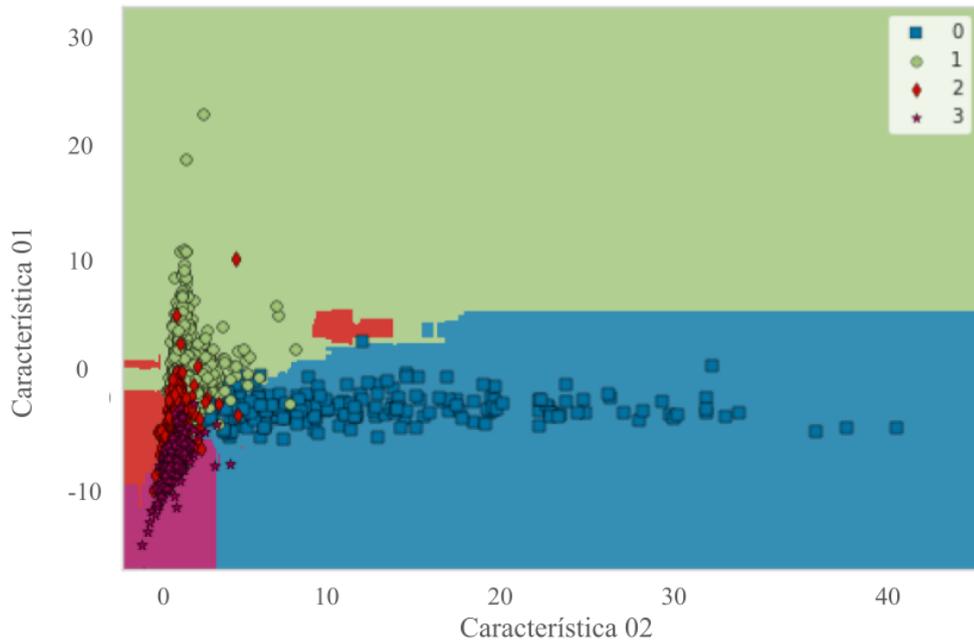
Na figura 20 é exibida a matriz de confusão do modelo RF, calculada com base nos dados de teste. Na imagem as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

Figura 18. Curvas ROC de performance do modelo RF com *cross-validation*



Fonte: Produzido pelo autor.

Figura 19. Limiões de decisão do modelo RF para predião das classes



Fonte: Produzido pelo autor.

Podemos ver através da matriz de confuso que a classe com a maior taxa de acerto  a classe 1, que representa os operadores bons, o que  esperado, visto que essa  a classe com maior quantidade de amostras de treinamento. Por conta disso o algoritmo tende a classificar mais sujeitos com a classe 1 do que deveria. Isso pode ser visto ao analisar a

Figura 20. Matriz de confuso do modelo RF nos dados da validao

	0	1	2	3
0	88,9%	10,7%	0,4%	0%
1	0,7%	95,4%	3,2%	0,2%
2	0,6%	10,3%	84,6%	4,5%
3	1,1%	0,7%	10,7%	87,5%
	0	1	2	3

**Classe verdadeira** (y-axis) vs **Classe predita** (x-axis)

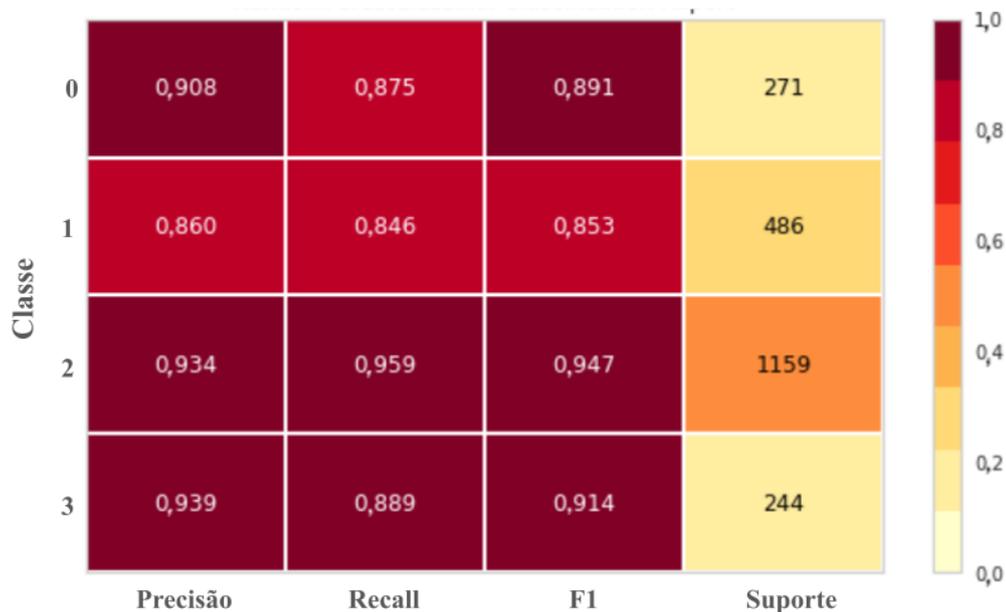
Fonte: Produzido pelo autor.

segunda coluna da matriz de confusão, que mostra uma porcentagem considerável de erros de sujeitos pertencentes às classes 0 e 2 de produtividade, mas que são classificados como pertencentes à classe 1. Observando na matriz as relações entre as classes 2 e 3, é possível ver também uma dificuldade do programa de discernir entre essas duas classes.

### 5.1.6 Relatório de Classificação

Na figura 21 são exibidas as métricas de precisão *recall* e *f1-score*. Estas são métricas utilizadas para avaliar a acurácia de um modelo. Precisão é a relação entre verdadeiros positivos (predição positiva correta) e todos os positivos, enquanto *recall* é a medida de acurácia do modelo em identificar verdadeiros positivos. Na imagem as classes 0, 1, 2 e 3 correspondem a produtividades de nível A, B, C e D.

Figura 21. Precisão *recall* e *f1-score* do classificador com modelo RF.



Fonte: Produzido pelo autor.

Podemos observar que a classe melhor classificada, segundo essas métricas, é a classe 1, o que corrobora com o que foi visto na matriz de confusão. Já a classe que teve pior classificação, segundo essas métricas, foi a classe 2. Isso provavelmente se dá por conta do programa entender que essa classe se parece bastante tanto com a classe 1, quanto com a classe 3, gerando um erro considerável para ambos os lados, como pode ser visto na figura 20.

### 5.1.7 Validação

Como mencionado anteriormente, um total de 20% do conjunto de dados inicial foi separado para validação *holdout*, totalizando 2160 amostras randomicamente selecionadas,

contendo exemplares de todas as classes do problema. A tabela 4 a seguir mostra o comparativo de performances entre os modelos treinados, usando 4 classes de classificação, sendo a acurácia a métrica selecionada como principal meio de avaliação para o problema de classificação.

Tabela 4. Comparativo de performance de treino entre classificadores treinados usando 4 classes

	Acurácia	AUC	Recall	Precisão	F1
RF	0,9509	0,9919	0,9433	0,9503	0,9500
Catboost	0,9488	0,9934	0,9412	0,9491	0,9488
ET	0,9487	0,9922	0,9421	0,9488	0,9486
XGBoost	0,9484	0,9927	0,9411	0,9488	0,9184
Lr	0,9481	0,9916	0,9377	0,9484	0,9481
LightGBM	0,9479	0,9929	0,9406	0,9482	0,9479
GBC	0,9471	0,9929	0,9390	0,9474	0,9471
KNN	0,9454	0,9814	0,9390	0,9453	0,9136
SVM	0,9318	0,0000	0,9163	0,9321	0,9311
DT	0,9226	0,9403	0,9140	0,9229	0,9226
LDA	0,9218	0,9869	0,8953	0,9233	0,9218
NB	0,9003	0,9783	0,9104	0,9065	0,9016
Ridge	0,8774	0,0000	0,8287	0,8788	0,8731

Fonte: Produzido pelo autor.

Podemos notar que a performance dos primeiros oito modelos da lista são extremamente similares e confiáveis, uma vez que tanto as métricas de erro quanto de precisão apontam a qualidade do aprendizado destes algoritmos. O modelo elencado como mais eficiente para a classificação com 4 classes foi o *Random Forest Classifier (RF)*, tendo chegado a uma acurácia de 95,09%, usando os dados de treino e 91,39% usando os dados da validação *holdout*. Após a passagem de todas estas amostras pelo *pipeline* de processamento que termina com o modelo classificando o operador, pudemos calcular uma acurácia total de 91,39%, com precisão de 91,35% e *recall* de 88,92%, como pode ser visto na tabela 5. Estes resultados estabelecem como confiáveis os dados supracitados referentes à performance do modelo na tarefa de treinamento e validação com *cross-validation*.

Tabela 5. Métricas de performance dos métodos de validação utilizados, treinados usando 4 classes

	acurácia	AUC	Recall	Precisão	F1
RF	0,9135	0,9783	0,9135	0,9135	0,9134
Inception	0,8999	0,9784	0,9000	0,8998	0,8997
Surrogates	0,7083	0,8509	0,6727	0,7003	0,6991
RF sem Inception	0,5882	0,7456	0,5882	0,5809	0,5828

Fonte: Produzido pelo autor.

Na tabela 5 também podemos ver uma linha intitulada de *inception*, uma linha intitulada RF sem *inception* e uma linha intitulada Surrogates. A linha intitulada *inception* representa as métricas obtidas usando o classificador da própria *inception* para classificar os dados, sem a adição de um classificador a mais. Podemos ver que a performance usando o todo o método proposto nesse trabalho é superior à performance de classificação da rede *inception* sozinha em quase todas as métricas, sendo a AUC a única em que o classificador original foi um pouco inferior.

Já a linha intitulada RF sem *inception* representa as métricas obtidas usando apenas os 13 classificadores que competem entre si para realizar a classificação, de forma a excluir a *inception* de todo o processo. Podemos ver que, na classificação em 4 classes, esse foi o classificador com pior acurácia, tendo uma acurácia de validação de menos de 60%.

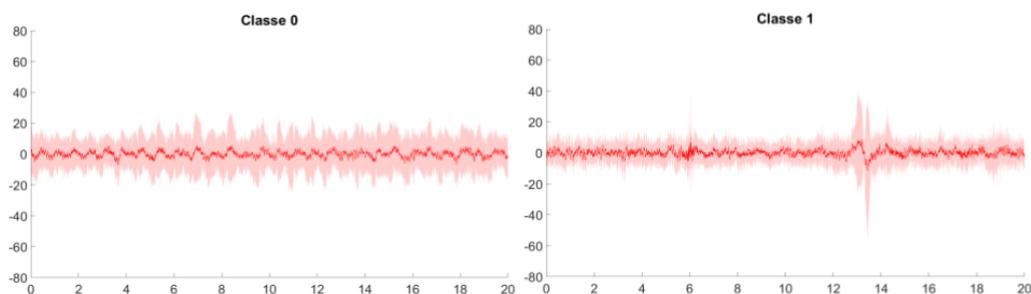
Finalmente a linha intitulada *Surrogates* representa o resultado de usar o mesmo modelo proposto por esse trabalho, mas trocando os dados utilizado para treiná-lo, substituindo os dados de EEG coletados dos operadores por *surrogates* desses mesmos dados. A performance da classificação usando os dados dos *surrogates* foi bem inferior à do classificador usando os dados originais, o que implica que as características não-lineares dos registros de EEG são essenciais para sua classificação.

## 5.2 Classificação em duas classes

### 5.2.1 Visualização dos dados de EEG por classe

Primeiramente é importante ter uma idéia de como são os EEGs dos operadores pertencentes a cada classe. Nessa linha a figura 22 mostra a média dos EEGs divididos de acordo com a classe de produtividade real dos operadores, usando a classificação binária, como apresentado na tabela 3.

Figura 22. Média dos registros de EEG, no eletrodo C3, após a filtragem, separados por classe.



Fonte: Produzido pelo autor.

Nessa figura podemos ver a média dos EEGs dos operadores, em um recorte de 20 segundos do eletrodo C3, representadas pela linha vermelha no centro de cada uma das 4

partes da imagem, divididos por sua classe de produtividade. Na imagem também é possível ver o desvio padrão dessas médias, que é representado através das sombras vermelhas nos gráficos. Nessa imagem é possível observar o padrão oscilatório típico presente no sinal cortical captado através do eletrodo C3, que representa, em sua maioria, a área central do cérebro. É importante ressaltar que essas médias foram calculadas com um número diferente de indivíduos, uma vez que o número de indivíduos não é igual em todas as classes, como mostrado na tabela 3.

### 5.2.2 Curva de Aprendizado

O modelo classificador *Extra Trees* (ET) foi o classificador com maior acurácia dentre os testados para a classificação em duas classes e, como mencionado anteriormente, treinado através do procedimento de validação cruzada. Na figura 23 abaixo são exibidas as variações da acurácia no treinamento e no teste.

Figura 23. Curva de aprendizado do modelo ET com *cross-validation*.



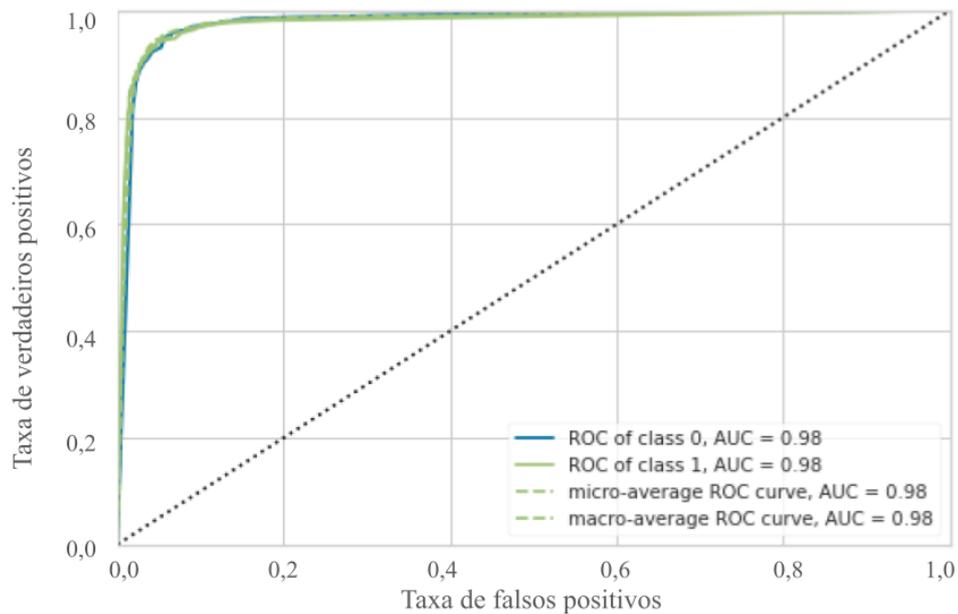
Fonte: Produzido pelo autor.

Podemos ver que a acurácia varia de 97,5% até bem próximo de 98%, com desvio padrão sempre próximo a 2%, mas com pico de 97,97%, como mostrado na tabela 6.

### 5.2.3 Curvas ROC

A figura 24 exhibe as curvas ROC, que mostram a performance de classificação do algoritmo por classe do problema. Na figura 24 a seguir, as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

Figura 24. Curvas ROC de performance do modelo ET com *cross-validation*



Fonte: Produzido pelo autor.

Podemos ver que as duas classes tem pontuações muito boas, possuindo uma quantidade muito maior de positivos verdadeiros do que de falsos positivos e tendo valores de AUC maiores do que 98% nas duas classes.

#### 5.2.4 Limiar de Decisão do Modelo

Na figura 25 são exibidos os limiares de decisão do modelo para predição de cada uma das classes. Podemos observar que o modelo conseguiu classificar as amostras com alta precisão e confiabilidade, errando apenas poucas amostras. Na imagem as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

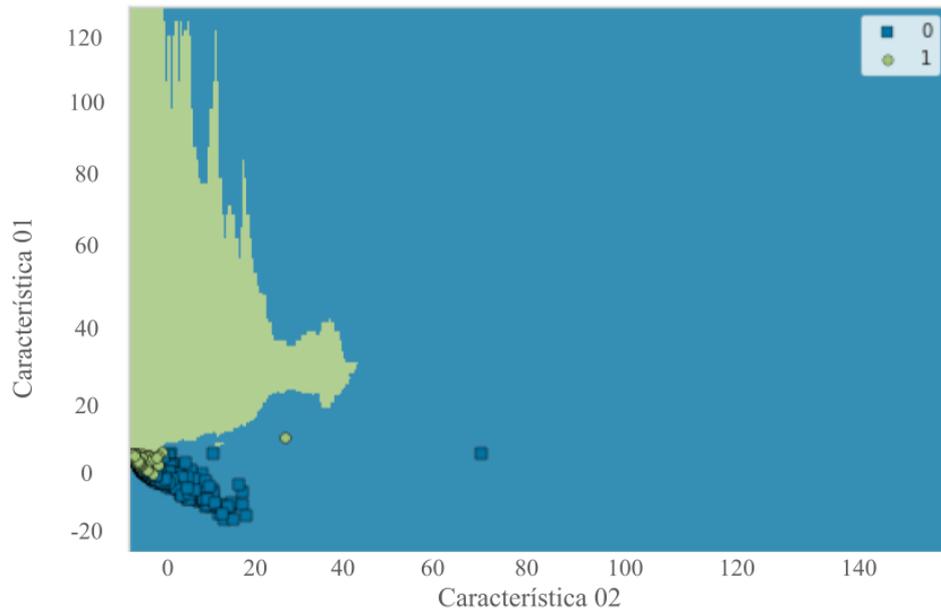
Podemos ver que a classe 0 ocupa uma área muito maior no limiar de decisão do sistema do que a classe 1, o que indica que os dados da classe 0 estão mais espalhados no espaço de fase gerado pelas características extraídas pela rede *Inception* e presentes na figura 25. Isso é esperado, uma vez que existe uma quantidade bem maior de sujeitos na classe 0 do que na classe 1.

#### 5.2.5 Matriz de Confusão

Na figura 26 é exibida a matriz de confusão do modelo ET, calculada com base nos dados de teste. Na imagem as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

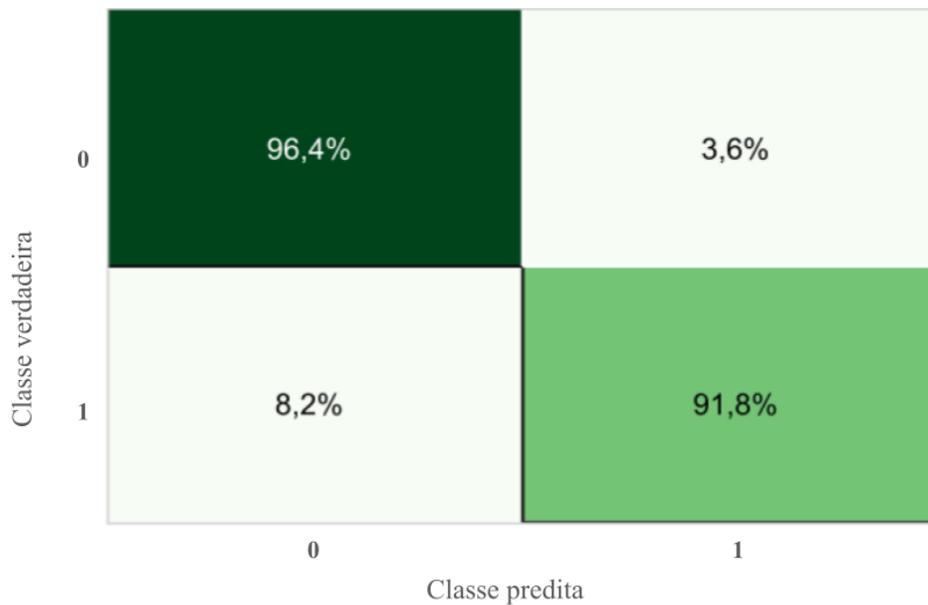
Na matriz de confusão do modelo podemos ver que há uma disparidade de 4,6% entre as duas classes, sendo que a classe 0 teve a maior acurácia, com valor de 96,4%. Já

Figura 25. Limiões de decisão do modelo ET para prediço das classes



Fonte: Produzido pelo autor.

Figura 26. Matriz de confuso do modelo ET nos dados da validaço



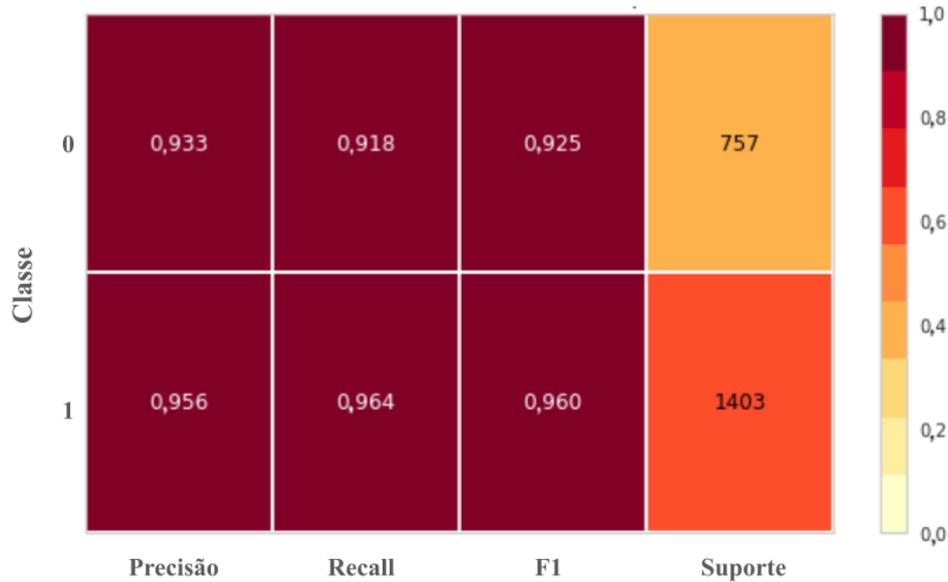
Fonte: Produzido pelo autor.

a classe 1 teve uma acurcia de 91,8%. Essa disparidade provavelmente se d devido  menor quantidade de dados presentes na classe 1 e provavelmente seria diminuda com a adiço dos dados de EEG de mais operadores.

### 5.2.6 Relatório de Classificação

Na figura 27 são exibidas as métricas de precisão *recall* e *f1-score*. Estas são métricas utilizadas para avaliar a acurácia de um modelo. Precisão é a relação entre verdadeiros positivos (predição positiva correta) e todos os positivos, enquanto *recall* é a medida de acurácia do modelo em identificar verdadeiros positivos. Na imagem as classes 0, 1, 2 e 3 corresponder a produtividades de nível A, B, C e D.

Figura 27. Precisão *recall* e *f1-score* do classificador com modelo ET.



Fonte: Produzido pelo autor.

Podemos observar que a classe melhor classificada, segundo essas métricas, é a classe 0, o que corrobora com o que foi visto na matriz de confusão. Apesar disso as duas classes possuem boas métricas de precisão *recall* e *F1*, todas acima de 91% o que indica uma alta confiabilidade do modelo em geral e uma performance de classificação consideravelmente superior à classificação usando 4 classes.

### 5.2.7 Validação

Como mencionado anteriormente, um total de 20% do conjunto de dados inicial foi separado para validação *holdout*, totalizando 2160 amostras randomicamente selecionadas, contendo exemplares de todas as classes do problema. A tabela 6 a seguir mostra o comparativo de performances entre os modelos treinados, usando uma classificação binária.

Podemos notar que os primeiros nove modelos da tabela 6 tem acurácias muito similares e tem bons resultados nas outras métricas sendo, portanto, todos bastante confiáveis. Dentre eles o que produziu a melhor acurácia para classificação com duas classes foi o *Random Forest Classifier (RF)*, tendo chegado a uma acurácia de 97,97%, usando os dados de treino e 95,05% usando os dados da validação *holdout*. Após a passagem de todas

Tabela 6. Comparativo de performance de treino entre classificadores treinados usando 2 classes

	Acurácia	AUC	Recall	Precisão	F1
ET	0,9797	0,9952	0,9710	0,9744	0,9726
SVM	0,9791	0,0000	0,9803	0,9639	0,9720
GBC	0,9791	0,9959	0,9719	0,9717	0,9718
RF	0,9789	0,9953	0,9716	0,9717	0,9716
Catboost	0,9789	0,9968	0,9713	0,9719	0,9716
KNN	0,9779	0,9892	0,9694	0,9711	0,9702
LR	0,9778	0,9962	0,9760	0,9646	0,9702
ADA	0,9772	0,9960	0,9688	0,9698	0,9692
LDA	0,9711	0,9930	0,9713	0,9518	0,9614
DT	0,9685	0,9687	0,9550	0,9600	0,9574
Ridge	0,9659	0,0000	0,9625	0,9465	0,9544
NB	0,9503	0,9850	0,9853	0,8922	0,9364
QDA	0,8549	0,9227	0,9884	0,7296	0,8375

Fonte: Produzido pelo autor.

estas amostras pelo *pipeline* de processamento que termina com o modelo classificando o operador, pudemos calcular uma acurácia total de 95,05%, com precisão de 93,29% e *recall* de 91,81%, como pode ser visto na tabela 7. Estes resultados estabelecem como confiáveis os dados supracitados referentes à performance do modelo na tarefa de treinamento e validação com *cross-validation*.

Tabela 7. Métricas de performance dos métodos de validação utilizados, treinados usando 2 classes

	Acurácia	AUC	Recall	Precisão	F1
ET	0,9505	0,9780	0,9181	0,9329	0,9254
Inception	0,9494	0,9717	0,9161	0,9152	0,9156
ET sem Inception	0,9364	0,9073	0,9806	0,9503	0,9652
Surrogates	0,7796	0,8354	0,6248	0,7113	0,6653

Fonte: Produzido pelo autor.

Na tabela 7 também podemos ver uma linha intitulada de *inception*, uma linha intitulada ET sem inception e uma linha intitulada Surrogates. A linha intitulada *inception* representa as métricas obtidas usando o classificador da própria *inception* para classificar os dados, sem a adição de um classificador a mais. Podemos ver que a performance usando o todo o método proposto nesse trabalho é superior à performance de classificação da rede *inception* sozinha em todas as métricas, mas a diferença de performances é pequena.

Já a linha intitulada RF sem inception representa as métricas obtidas usando apenas os 13 classificadores que competem entre si para realizar a classificação, de forma a excluir a *inception* de todo o processo. Podemos ver que, na classificação em 2 classes,

esse classificador teve boas métricas, mas ainda foi inferior ao classificador original em termos de acurácia e AUC.

Finalmente a linha intitulada *Surrogates* representa o resultado de usar o mesmo modelo proposto por esse trabalho, mas trocando os dados utilizado para treiná-lo, substituindo os dados de EEG coletados dos operadores por *surrogates* desses mesmos dados. A performance da classificação usando os dados dos *surrogates* foi bem inferior à do classificador usando os dados originais, o que implica que as características não-lineares dos registros de EEG são essenciais para sua classificação.

## 6 Discussão

Nesse trabalho nos mostramos que uma alta acurácia foi obtida para a classificação de perfis de produtividade usando dados de EEG de trabalhadores de minas responsáveis por operar escavadeiras de alta capacidade sob alta demanda de precisão e atenção sustentada. Uma classificação de alta acurácia do EEG usando aprendizado de máquina não é um novo resultado dentro da ampla lista de aplicações bem sucedidas desse tipo de técnica. O que é novo é o fato de que um padrão comportamental complexo, como a produtividade desses trabalhadores, pode ser classificado dentro das seguintes condições: I. O EEG foi coletado em repouso, ou seja, sem a realização de tarefas durante o registro; II. A tarefa de onde as classes de produtividade foram extraídas não era uma tarefa de laboratório tradicional, bem regulada e feita com o intento de detectar variações de alguma função cognitiva, mas sim de uma tarefa que fazia parte da rotina de treinamento desses operadores na empresa, que teria acontecido independentemente do nosso experimento ter sido realizado ou não. Ou seja, o treinamento do operador não foi adaptado para se encaixar no nosso estudo, nosso estudo que precisou ser adaptado para se adequar ao protocolo de treinamento do trabalhador, o que foi uma escolha metodológica deliberada da nossa parte.

O fato do EEG usado para a classificação ter sido obtido em estado de repouso torna esse resultado ainda mais extraordinário uma vez que isso implica que a análise do estado cerebral do operador antes tarefa já seria o suficiente para classificar sua produtividade nela, o que pode ser usado no futuro para guiar a rotina desses trabalhadores, servindo como uma base para uma rotina de trabalho individualizada. Ou seja o procedimento de fazer 3 minutos de EEG em repouso pode ser adicionado à rotina dos operadores com custo muito baixo em questão de tempo gasto e investimento financeiro, uma vez que o algoritmo proposto é capaz de estimar, com somente 3 minutos de EEG, como será a produtividade desse operador durante suas próximas horas de trabalho, o que pode guiar a rotina diária dele, trocando horas de trabalho com baixa produtividade, por treinamento ou descanso, por exemplo, que podem gerar uma maior produtividade no longo prazo.

Esse tipo de resultado usando EEG de repouso é inovador, mas não é impensável, dado os vários estudos que estão sendo feitos com esse tipo de registro. O estudo de Lee *et al.* (2021), é um grande exemplo do potencial da classificação usando redes neurais artificiais desse tipo de registro. Nesse estudo Lee *et al.* (2021) criaram um classificador usando redes neurais convolucionais para classificação de pessoas com Doença de Parkinson com mais de 92% de acurácia usando somente o EEG de repouso. Em outro estudo, este feito por Zhang *et al.* (2020) foram usados registros de EEG de repouso para análise e classificação de transtorno depressivo maior, tendo chegado a 93% de acurácia. Existem, inclusive,

revisões sistemáticas sobre o uso de AI e métodos estatísticos para a classificação de EEG em estado de repouso, como a escrita por Tzimourta et. al. (2021). Essas publicações mostram que existe um grande campo para o estudo e a classificação de dados de EEG em estado de repouso e de que é possível inferir muito sobre o estado e condição mentais das pessoas através desse tipo de registro.

O fato de que os dados usados para compor esse estudo não foram tarefas de laboratório tradicionais, altamente controladas e específicas para o estudo, mas foram tarefas que já faziam parte da rotina dos operadores também é uma contribuição muito interessante do trabalho. Essa contribuição é importante pois implica que o método proposto é facilmente adaptável para compor uma rotina de treinamento, podendo impactar diretamente na vida e rotina de trabalho desses operadores. Ao propor métodos para serem aplicados na indústria é essencial eles tenham uma aplicação fácil e rápida na rotina de trabalho dos operadores e que a alterem o mínimo possível, pois isso é determinante no momento dos responsáveis na empresa decidirem se esses métodos serão de fato aplicados ou não. Foi nesse sentido que decidimos propor um experimento que interferisse o mínimo possível na rotina dos trabalhadores.

Nossos resultados se localizam na moderna abordagem de sistemas automáticos de classificação de dados de EEG (CRAIK; HE; CONTRERAS-VIDAL, 2019). Já a nossa contribuição para o sucesso da classificação ocorreu através de duas abordagens principais, sendo elas a extração de características feita através da rede *inception*, otimizada usando a otimização bayesiana e a subsequente classificação das amostras, a partir das características coletadas, usando 13 classificadores diferentes e escolhendo o que performasse melhor na classificação, tendo a acurácia como principal métrica para isso.

A rede *inception* é uma rede muito promissora no campo do *deep learning* por introduzir um novo nível de organização à esse tipo de rede, na forma do "módulo *Inception*", além ter uma profundidade maior de conexões. Esses módulos são grupos de camadas convolucionais paralelas 1x1, 3x3 ou 5x5 que fazem redução de dimensionalidade de seções do dado analisado de forma independente e em seguida são ligadas por uma camada de *MaxPooling*, como mostrado na figura 15.

A principal vantagem desse tipo de rede é o ganho significativo de qualidade em troca de um modesto aumento no custo computacional do uso desse tipo de técnica, comparado com redes neurais mais simples e menos profundas (SZEGEDY et al., 2015). Outra vantagem é que esse tipo de módulo diminui probabilidade de haver *overfitting*. Isso é especialmente útil na análise de sinais de EEG, uma vez que esses sinais são muito complexos, o que exige uma rede profunda e eficaz para analisá-los.

Os classificadores que se saíram melhor dentre os selecionados, por sua vez, são ambas versões do classificador *random forest*, um tipo de classificador que usa várias árvores de decisão treinadas durante a fase de treinamento e gera previsões baseadas na média das

previsões geradas pelas árvores individuais. Esse tipo de classificador é muito mais simples do que os classificadores usados em técnicas de *deep learning*, como a *inception*, mas ainda assim podem ser muito efetivos. Acreditamos que como a informação contida nos EEGs foi processada pela *inception* a complexidade final dos dados à serem classificados caem muito, justamente pelo processamento, simplificação e limpeza dos dados feita pela rede e, por isso, esses classificadores baseados em *random forest* performam melhor. Esse tipo de classificador já foi usado em muitos trabalhos de classificação usando EEG, como o no trabalho publicado por Donos, Dümpelmann, e Schulze-Bonhage (2015), usado para detecção de convulsões e no trabalho de Edla *et. al.* (2018) usado para classificação de estados mentais.

Acreditamos que foi essa união de técnicas que foi diferencial no trabalho e proporcionou uma acurácia tão elevada na tarefa realizada. Isso é corroborado pelo fato de que testamos variantes do algoritmo criado, uma delas usando somente a rede *inception* e outra usando somente os 13 classificadores que competem entre si, usando os mesmos dados para treinamento e elas tiveram menor acurácia do que o método proposto tanto na classificação usando duas classes, quanto na classificação usando 4 classes.

Além disso percebemos a importância das características não lineares presentes nos registros de EEG coletados para a classificação ao testar o método proposto treinado com dados gerados por seus *surrogates*. Os *surrogates* são séries temporais lineares que reproduzem várias propriedades estatísticas de uma série temporal usada para criá-las, mas sem suas propriedades não lineares (VENEMA; AMENT; SIMMER, 2006), ou seja, são versões lineares e muito similares das séries temporais usadas para criá-las. O fato da acurácia usando esses dados ter sido bem mais baixa do que a acurácia usando os registros de EEG implica que as características não lineares desses registros são essenciais para sua classificação. Isso pode influenciar os passos futuros desse trabalho, que serão discutidos mais aprofundadamente no capítulo 7.

Acreditamos também que o trabalho é inovador ao usar uma metodologia de coleta que não influencia em quase nada na rotina do operador de mina, por se aproveitar de momentos presentes em sua rotina para realizar a coleta de dados, sem atrapalhar sua rotina. O fato de estarmos tentando classificar produtividade através do EEG de repouso também é interessante, pois ele não ocorre no momento da tarefa, apenas um pouco antes e um pouco depois. Esses fatos são os que, na nossa opinião, diferenciam nosso trabalho da maioria dos trabalhos que realizam a classificação automática de dados de EEG.

## 7 Conclusões

Avaliamos que o presente estudo cumpriu seus objetivos. Foi feito o pré processamento dos dados coletados usando uma filtragem passa-banda e uma filtragem manual usando ICA. Foi desenvolvido, em seguida, uma rede neural artificial com parâmetros otimizados usando a otimização bayeziana, com o papel de extrair arrays numéricos resultantes de diversas convoluções que foram usados como entrada de um classificador. Foi então treinado um grupo de 13 classificadores clássicos na área do aprendizado de máquina e o melhor desses classificadores para cada grupo de dados foi selecionado. Por fim foi feita uma avaliação da classificação obtida usando como métricas: acurácia, AUC, *recall*, precisão e F1, além das curvas de aprendizado e das curvas ROC.

O modelo proposto alcançou uma acurácia bem alta na classificação de perfis de produtividade a partir do registro de EEG dos operadores em estado de repouso, tanto na classificação multi-classe, tendo atingido uma acurácia de validação de 91,35%, quanto na classificação binária, tendo atingido uma acurácia de validação de 95,05%. Acreditamos que essa alta acurácia tenha sido obtida devido uma abordagem inovadora na arquitetura proposta, ao usar técnicas não muito difundidas na classificação de registros de EEG (como a rede *inception*) e ao acoplar essas técnicas a outras (como os classificadores baseados em árvore de decisão e a otimização bayesiana) que auxiliaram na melhoria da performance do classificador criado. Destacamos em especial o "modulo *inception*" que dá nome a arquitetura *inception*, e a otimização Bayesiana, ambos métodos que proporcionaram um grande aumento de performance na tarefa de classificação.

A partir do software proposto foi enviada uma análise de viabilidade de patente para a VALE, para que seja gerado uma patente com registro de software do algoritmo proposto, caso seja de interesse da empresa. Essa proposta de patente está sendo analisada nesse momento e é bem provável que seja aceita, uma vez que o método proposto é inovador e eficaz. Esse tipo de patente é comum na empresa, inclusive tendo sido feita uma patente de um software, criado anteriormente a este trabalho e usando dados diferentes como base, para o mesmo projeto no qual este trabalho está inserido, chamado de “Usando Treinamento Cognitivo para o Desenvolvimento de Operadores de Alto Desempenho”. Essa outra patente do mesmo projeto tem como identificação de registro: BR102019025998A2 e está disponível em: <<https://worldwide.espacenet.com/patent/search/family/076865429/publication/BR102019025998A2?q=102019025998>>.

Como nossos próximos passos para esse trabalho existem três caminhos naturais a se seguir: I. A exploração dos dados de EEG coletados durante o treinamento feito no simulador 4D; II. A análise dos dados coletados sob a perspectiva dos métodos dinâmicos;

III. Um estudo mais aprofundado sobre quais são as características de EEG que foram fundamentais para o funcionamento do método proposto.

O passo mais natural e simples para a continuação desse estudo é a utilização do mesmo método proposto nesse trabalho, mas ao invés de usar o dado de EEG em repouso usar o dado coletado durante a tarefa, que é o treinamento no simulador 4d. Esse dado não foi explorado ainda porque ele é muito grande e, portanto, demanda mais tempo para ser processado e para que todo o algoritmo rode. Se usássemos todos os 60 minutos de EEG coletados à 300 Hz, dos 60 registros que foram feitos, com todos os 20 eletrodos analisados na pesquisa, teríamos mais de 1 bilhão de pontos de registro eletroencefalográfico para analisar. Para analisar esse dado será necessária a continuação dessa pesquisa no futuro.

Outro passo importante seria a análise dos dados coletados sob a perspectiva dos métodos dinâmicos. Há um consenso hoje de que o cérebro é um sistema caótico-determinístico e que os dados de EEG são séries temporais não-lineares e não estacionárias e o uso de métodos não-lineares tendem a ser muito eficientes nesse tipo de sistema (RODRIGUEZ-BERMUDEZ; GARCIA-LAENCINA, 2015). Além disso identificamos através da validação usando *surrogates* que as dinâmicas não lineares dos registros de EEG foram essências para a classificação dos mesmos. Portanto seria interessante explorar técnicas voltadas para dados não-lineares, como a entropia, a sparse identification of nonlinear dynamics (SINDy) e a reconstrução do espaço de fase atrasado na análise dos dados coletados. No trabalho de Duan, Zhu e Lu (2013), por exemplo, a entropia diferencial é usada como uma característica para classificação de emoções através de registros de EEG. No trabalho de Bruton *et. al.* (2016) que propõe um algoritmo capaz de estimar as equações governantes de dados não lineares. Já no estudo de Altındiş *et. al.* (2021) foi usado o *time delay embedding* para realizar uma reconstrução no espaço de fase de um sinal de EEG que, por sua vez, foi usada para analisar topologicamente dados de EEG para encontrar ruído e artefatos nesses dados. Poderíamos testar abordagens similares às apresentadas nos dois trabalhos citados acima com nossos dados, para avaliar se a acurácia e capacidade de generalização do classificador cresceria e para que possamos entender melhor a dinâmica não linear presente nesses registros de EEG.

Finalmente, a continuação mais robusta que pensamos para esse trabalho, a qual será continuada no projeto de doutorado, é o estudo mais aprofundado das características extraídas pela rede *inception* e do motivo pelo qual a combinação de técnicas específicas que usamos nesse trabalho (redes *inception*, otimização bayeziana e classificadores baseados em árvores de decisão) terem funcionado tão bem. Modelos que usam redes neurais artificiais, especialmente os de *deep learning* são considerados considerados caixas pretas, uma vez que todos os processos que eles fizeram para chegar na sua acurácia final não podem ser explicados de forma intuitiva para seres humanos. Isso se dá pois muito do aprendizado é comprimido na forma de pesos (valores numéricos) entre os nós da rede e esses valores

impactam no funcionamento da rede, mas são muito complexos para as pessoas que as analisam. Para lidar com essas questões alguns pesquisadores estão criando modelos chamados de modelos de inteligência artificial explicáveis. O objetivo desse tipo de modelo é tornar o comportamento de IAs mais inteligível à seres humanos criando explicações, na forma de textos ou imagens, sobre o funcionamento de cada parte da rede (GUNNING et al., 2019). Nesse sentido a aplicação desse tipo de software ao nosso método o tornaria muito mais confiável, por termos explicações mais inteligíveis sobre as minúcias de seu comportamento. Além disso seria interessante uma pesquisa mais profunda e complexa sobre as implicações computacionais e matemáticas da combinação das técnicas que usamos, pesquisando em bibliografia e comparando com outras combinações de técnicas já presentes na literatura.

## Referências

- AGGARWAL, S.; CHUGH, N. Review of machine learning techniques for eeg based brain computer interface. *Archives of Computational Methods in Engineering*, Springer, p. 1–20, 2022. Citado na página 23.
- AL-SAEGH, A.; DAWWD, S. A.; ABDUL-JABBAR, J. M. Deep learning for motor imagery eeg-based classification: A review. *Biomedical Signal Processing and Control*, Elsevier, v. 63, p. 102172, 2021. Citado na página 25.
- ALPAYDIN, E. *Introduction to machine learning*. [S.l.]: MIT press, 2020. Citado 2 vezes nas páginas 13 e 17.
- ALTINDIŞ, F.; YILMAZ, B.; BORISENOK, S.; İÇÖZ, K. Parameter investigation of topological data analysis for eeg signals. *Biomedical Signal Processing and Control*, Elsevier, v. 63, p. 102196, 2021. Citado na página 57.
- BAKARJI, J.; CHAMPION, K.; KUTZ, J. N.; BRUNTON, S. L. Discovering governing equations from partial measurements with deep delay autoencoders. *arXiv preprint arXiv:2201.05136*, 2022. Citado na página 12.
- BASHASHATI, H.; WARD, R. K.; BASHASHATI, A. User-customized brain computer interfaces using bayesian optimization. *Journal of neural engineering*, IOP Publishing, v. 13, n. 2, p. 026001, 2016. Citado na página 13.
- BERRAR, D. et al. *Cross-Validation*. 2019. Citado na página 15.
- BRUNTON, S. L.; PROCTOR, J. L.; KUTZ, J. N. Discovering governing equations from data by sparse identification of nonlinear dynamical systems. *Proceedings of the national academy of sciences*, National Acad Sciences, v. 113, n. 15, p. 3932–3937, 2016. Citado na página 57.
- CHERUBINO, P.; MARTINEZ-LEVY, A. C.; CARATU, M.; CARTOCCI, G.; FLUMERI, G. D.; MODICA, E.; ROSSI, D.; MANCINI, M.; TRETTEL, A. et al. Consumer behaviour through the eyes of neurophysiological measures: State-of-the-art and future trends. *Computational intelligence and neuroscience*, Hindawi, v. 2019, 2019. Citado na página 11.
- CRAIK, A.; HE, Y.; CONTRERAS-VIDAL, J. L. Deep learning for electroencephalogram (eeg) classification tasks: a review. *Journal of neural engineering*, IOP Publishing, v. 16, n. 3, p. 031001, 2019. Citado 4 vezes nas páginas 23, 24, 25 e 54.
- DELORME, A.; MAKEIG, S. Eeglab: an open source toolbox for analysis of single-trial eeg dynamics including independent component analysis. *Journal of neuroscience methods*, Elsevier, v. 134, n. 1, p. 9–21, 2004. Citado na página 33.
- DONOS, C.; DÜMPELMANN, M.; SCHULZE-BONHAGE, A. Early seizure detection algorithm based on intracranial eeg and random forest classification. *International journal of neural systems*, World Scientific, v. 25, n. 05, p. 1550023, 2015. Citado na página 55.

- DUAN, R.-N.; ZHU, J.-Y.; LU, B.-L. Differential entropy feature for eeg-based emotion classification. In: IEEE. *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*. [S.l.], 2013. p. 81–84. Citado na página 57.
- EDLA, D. R.; MANGALOREKAR, K.; DHAVALIKAR, G.; DODIA, S. Classification of eeg data for human mental state analysis using random forest classifier. *Procedia computer science*, Elsevier, v. 132, p. 1523–1532, 2018. Citado na página 55.
- FAWCETT, T. An introduction to roc analysis. *Pattern recognition letters*, Elsevier, v. 27, n. 8, p. 861–874, 2006. Citado na página 18.
- FEDOTOV, A. Selection of parameters of bandpass filtering of the ecg signal for heart rhythm monitoring systems. *Biomedical Engineering*, Springer, v. 50, p. 114–118, 2016. Citado na página 24.
- FLEXA, C.; SANTOS, R.; GOMES, W.; SALES, C. A novel equidistant-scattering-based cluster index. In: IEEE. *2018 7th Brazilian Conference on Intelligent Systems (BRACIS)*. [S.l.], 2018. p. 540–545. Citado na página 20.
- FRAZIER, P. I. A tutorial on bayesian optimization. *arXiv preprint arXiv:1807.02811*, 2018. Citado 2 vezes nas páginas 13 e 37.
- GOMES, M. da M. Bases fisiológicas do eletroencefalograma. *Revista Brasileira de Neurologia*, v. 51, n. 1, 2015. Citado 2 vezes nas páginas 11 e 22.
- GUNNING, D.; STEFIK, M.; CHOI, J.; MILLER, T.; STUMPF, S.; YANG, G.-Z. Xai—explainable artificial intelligence. *Science robotics*, American Association for the Advancement of Science, v. 4, n. 37, p. eaay7120, 2019. Citado na página 58.
- HORVAT, T.; HAVAŠ, L.; SRPAK, D. The impact of selecting a validation method in machine learning on predicting basketball game outcomes. *Symmetry*, MDPI, v. 12, n. 3, p. 431, 2020. Citado na página 15.
- LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015. Citado 2 vezes nas páginas 12 e 25.
- LEE, C.; LANDGREBE, D. A. Decision boundary feature extraction for neural networks. *IEEE Transactions on Neural Networks*, IEEE, v. 8, n. 1, p. 75–83, 1997. Citado na página 19.
- LEE, S.; HUSSEIN, R.; WARD, R.; WANG, Z. J.; MCKEOWN, M. J. A convolutional-recurrent neural network approach to resting-state eeg classification in parkinson’s disease. *Journal of neuroscience methods*, Elsevier, v. 361, p. 109282, 2021. Citado na página 53.
- LIU, D. C.; NOCEDAL, J. On the limited memory bfgs method for large scale optimization. *Mathematical programming*, Springer, v. 45, n. 1-3, p. 503–528, 1989. Citado na página 37.
- MACQUEEN, J. Classification and analysis of multivariate observations. In: *5th Berkeley Symp. Math. Statist. Probability*. [S.l.: s.n.], 1967. p. 281–297. Citado na página 20.
- NOORBASHA, S. K.; SUDHA, G. F. Removal of eeg artifacts and separation of different cerebral activity components from single channel eeg—an efficient approach combining ssa-ica with wavelet thresholding for bci applications. *Biomedical Signal Processing and Control*, Elsevier, v. 63, p. 102168, 2021. Citado 2 vezes nas páginas 24 e 34.

PAUL, M.; GANESAN, S.; SANDHU, J.; SIMON, J. Effect of sensory motor rhythm neurofeedback on psycho-physiological, electro-encephalographic measures and performance of archery players. *Ibnosina Journal of Medicine and Biomedical Sciences*, Thieme Medical and Scientific Publishers Pvt. Ltd., v. 4, n. 02, p. 32–39, 2012. Citado na página 21.

POP, N. A.; DABIJA, D.-C.; IORGA, A. M. Ethical responsibility of neuromarketing companies in harnessing the market research—a global exploratory approach. *Amfiteatru economic*, v. 16, n. 35, p. 26–40, 2014. Citado na página 11.

RODRIGUEZ-BERMUDEZ, G.; GARCIA-LAENCINA, P. J. Analysis of eeg signals using nonlinear dynamics and chaos: a review. *Applied mathematics & information sciences*, Natural Sciences Publishing Corp, v. 9, n. 5, p. 2309, 2015. Citado 2 vezes nas páginas 11 e 57.

ROY, Y.; BANVILLE, H.; ALBUQUERQUE, I.; GRAMFORT, A.; FALK, T. H.; FAUBERT, J. Deep learning-based electroencephalography analysis: a systematic review. *Journal of neural engineering*, IOP Publishing, v. 16, n. 5, p. 051001, 2019. Citado 2 vezes nas páginas 23 e 25.

SCHREIBER, T.; SCHMITZ, A. Surrogate time series. *Physica D: Nonlinear Phenomena*, Elsevier, v. 142, n. 3-4, p. 346–382, 2000. Citado na página 39.

SPACKMAN, K. A. Signal detection theory: Valuable tools for evaluating inductive learning. In: ELSEVIER. *Proceedings of the sixth international workshop on Machine learning*. [S.l.], 1989. p. 160–163. Citado na página 18.

SUN, L.; LIU, Y.; BEADLE, P. J. Independent component analysis of eeg signals. In: IEEE. *Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology, 2005*. [S.l.], 2005. p. 219–222. Citado na página 24.

SUN, Y.; WEI, C.; CUI, V.; XIU, M.; WU, A. Electroencephalography: clinical applications during the perioperative period. *Frontiers in Medicine*, Frontiers Media SA, v. 7, p. 251, 2020. Citado na página 21.

SZEGEDY, C.; LIU, W.; JIA, Y.; SERMANET, P.; REED, S.; ANGUELOV, D.; ERHAN, D.; VANHOUCHE, V.; RABINOVICH, A. Going deeper with convolutions. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. [S.l.: s.n.], 2015. p. 1–9. Citado 3 vezes nas páginas 36, 37 e 54.

TEPLAN, M. et al. Fundamentals of eeg measurement. *Measurement science review*, v. 2, n. 2, p. 1–11, 2002. Citado na página 22.

TIMMERS, D. Chapter six - treating attention deficits and impulse control. In: CANTOR, D. S.; EVANS, J. R. (Ed.). *Clinical Neurotherapy*. Boston: Academic Press, 2014. p. 139 – 169. ISBN 978-0-12-396988-0. Disponível em: <<http://www.sciencedirect.com/science/article/pii/B9780123969880000064>>. Citado 2 vezes nas páginas 21 e 23.

TOUGUI, I.; JILBAB, A.; MHAMDI, J. E. Impact of the choice of cross-validation techniques on the results of machine learning-based diagnostic applications. *Healthcare informatics research*, Korean Society of Medical Informatics, v. 27, n. 3, p. 189–199, 2021. Citado na página 16.

TZIMOURTA, K. D.; CHRISTOU, V.; TZALLAS, A. T.; GIANNAKEAS, N.; ASTRAKAS, L. G.; ANGELIDIS, P.; TSALIKAKIS, D.; TSIPOURAS, M. G. Machine learning algorithms and statistical approaches for alzheimer's disease analysis based on resting-state eeg recordings: A systematic review. *International journal of neural systems*, World Scientific, v. 31, n. 05, p. 2130002, 2021. Citado na página 54.

VENEMA, V.; AMENT, F.; SIMMER, C. A stochastic iterative amplitude adjusted fourier transform algorithm with improved accuracy. *Nonlinear Processes in Geophysics*, Copernicus Publications Göttingen, Germany, v. 13, n. 3, p. 321–328, 2006. Citado 2 vezes nas páginas 39 e 55.

VIERING, T.; LOOG, M. The shape of learning curves: a review. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, IEEE, 2022. Citado na página 19.

WANG, J.-R.; HSIEH, S. Neurofeedback training improves attention and working memory performance. *Clinical Neurophysiology*, v. 124, n. 12, p. 2406 – 2420, 2013. ISSN 1388-2457. Disponível em: <<http://www.sciencedirect.com/science/article/pii/S1388245713006949>>. Citado na página 21.

ZHANG, B.; YAN, G.; YANG, Z.; SU, Y.; WANG, J.; LEI, T. Brain functional networks based on resting-state eeg data for major depressive disorder analysis and classification. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, IEEE, v. 29, p. 215–229, 2020. Citado na página 53.

ZHOU, Z.-H. *Machine learning*. [S.l.]: Springer Nature, 2021. Citado 5 vezes nas páginas 12, 13, 15, 16 e 18.