CECÍLIA GARCIA DA SILVEIRA

APLICAÇÃO DE *MACHINE LEARNING* NA CLASSIFICAÇÃO DE PERFIS DE IMAGEM ACÚSTICA DE POÇOS DE PETRÓLEO

Trabalho apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do diploma de Engenharia de Petróleo.

São Paulo
2023

CECÍLIA GARCIA DA SILVEIRA

APLICAÇÃO DE *MACHINE LEARNING* NA CLASSIFICAÇÃO DE PERFIS DE IMAGEM ACÚSTICA DE POÇOS DE PETRÓLEO

Trabalho apresentado à Escola Politécnica da Universidade de São Paulo para obtenção do diploma de Engenharia de Petróleo.

Área de Concentração:
Engenharia de Petróleo, *Machine Learning*

Orientador:
Profa. Dra. Elsa Vásquez Alvarez

Co-orientador:
Geo. Érica Kato Pacheco Ferraz

São Paulo
2023

Catalogação-na-publicação

Silveira, Cecília Garcia da
Aplicação de Machine Learning na Classificação de Perfis de Imagem Acústica de Poços de Petróleo / C. G. Silveira -- São Paulo, 2023.
58 p.

Trabalho de Formatura - Escola Politécnica da Universidade de São Paulo. Departamento de Engenharia de Minas e de Petróleo.

1.Aprendizado de Máquina 2.Classificação de Imagens 3.Engenharia de Petróleo 4.Perfilagem de Poços 5.Perfil Sônico I.Universidade de São Paulo. Escola Politécnica. Departamento de Engenharia de Minas e de Petróleo II.t.

AGRADECIMENTOS

Gostaria de expressar minha profunda gratidão a todos que tornaram possível a conclusão deste trabalho.

Agradeço à minha orientadora, Profa. Dra. Elsa Vásquez Alvarez, e à co-orientadora, Érica Kato Pacheco Ferraz, pela orientação e apoio constante. Aos professores Marcio Augusto Sampaio Pinto e Rodrigo César Teixeira de Gouvêa, que forneceram *insights* valiosos e substanciais para a realização deste trabalho.

À Escola Politécnica, ao Departamento de Engenharia Minas e de Petróleo e à Universidade de São Paulo pelas oportunidades de aprendizado e crescimento. Agradeço à ANP pela concessão de uso dos dados utilizados na realização deste trabalho.

À minha família e amigos, o meu eterno agradecimento pelo amor, compreensão e incentivo.

Este trabalho é resultado de uma rede de apoio incrível, e por isso, expresso minha sincera gratidão a todos que fizeram parte dessa jornada.

"Tenho a impressão de ter sido uma criança brincando à beira-mar, divertindo-me em descobrir uma pedrinha mais lisa ou uma concha mais bonita que as outras, enquanto o imenso oceano da verdade continua misterioso diante de meus olhos."

-- Isaac Newton

RESUMO

A indústria de petróleo é responsável pela geração de diversos tipos de dados cruciais para a administração e tomada de decisões em campos de exploração. Atualmente, a indústria está incorporando o uso de inteligência artificial e aprendizado de máquina como ferramentas para aprimorar a tomada de decisões e a gestão, tirando proveito dos vastos conjuntos de dados já coletados. O objetivo deste Trabalho de Conclusão de Curso (TCC) é fazer a classificação de seções de perfis de imagem acústica para ser utilizado em modelos de *Machine Learning* e também implementar esses algoritmos para classificar e localizar pontos de coleta de amostras em perfis de imagem acústica de poços. Para alcançar esse fim, foi feita a seleção dos dados de entrada e critérios de classificação. Inicialmente, foram empregados algoritmos de aprendizado supervisionado tradicionais, como *Support Vector Machine* (SVM) e *Random Forest* (RF), porém posteriormente foi necessário implementar modelos de Redes Convolucionais para fazer a classificação e localização das amostras no perfil. Os dados do poço utilizados foram disponibilizados pela ANP (SEI nº 48610.233579/2023-38). Ao todo foram utilizadas 2566 imagens para treinamento e validação dos modelos. Os resultados finais do modelo não convergiram para a classificação e localização das amostras, entretanto os dados classificados para treinamento foram organizados em um *dataset* (banco de dados) para serem utilizados em projetos futuros.

Palavras-Chave – Aprendizado de Máquina, Classificação de Imagens, Engenharia de Petróleo, Perfilagem de Poços, Perfil de Imagem Acústico.

ABSTRACT

A petroleum industry is responsible for generating various types of crucial data for exploration field management and decision-making. Currently, the industry is incorporating the use of artificial intelligence and machine learning as tools to enhance decision-making and management, leveraging vast sets of already collected data. The objective of this Undergraduate Thesis is to classify sections of acoustic image logs for use in machine learning models and to implement these algorithms to classify and locate sample collection points in well acoustic image profiles. To achieve this goal, the selection of input data and classification criteria was carried out. Initially, traditional supervised learning algorithms such as Support Vector Machine (SVM) and Random Forest (RF) were employed. However, it was later necessary to implement Convolutional Neural Network models for sample classification and localization in the profile. Well data used were provided by ANP (SEI No. 48610.233579/2023-38). A total of 2566 images were used for model training and validation. The final model results did not converge for sample classification and localization; however, the classified training data was organized into a dataset for use in future projects.

Keywords – Machine Learning, Image Classification, Petroleum Engineering, Well Logging, Acoustic Image Log.

LISTA DE FIGURAS

LISTA DE TABELAS

SUMÁRIO

1 INTRODUÇÃO

Atualmente, mesmo com todos os avanços e investimentos em energias renováveis, o petróleo segue como uma das fontes de energia mais utilizadas no mundo (QIN *et al.*, 2023). A exploração do combustível fóssil acumulado em bacias sedimentares se inicia pela identificação de áreas promissoras, aquisição de dados sísmicos até a perfuração dos primeiros poços exploratórios. A perfilagem destes poços agrega uma série de informações petrofísicas que irão alimentar o modelo geológico do reservatório. Dessa forma, muitas são as informações obtidas direta ou indiretamente de bacias sedimentares. Nesse cenário, a indústria petrolífera enfrenta diversos desafios na organização de banco de dados e no processamento das informações. A análise técnica apropriada dos dados é de fundamental importância no desempenho da indústria de hidrocarbonetos e atualmente ela ainda é feita em grande parte por especialistas, que fazem a classificação e interpretação dos dados. A otimização na classificação desses dados permitiria contribuir para agilizar o processo de análise técnica, deixando os especialistas apenas com a tarefa de interpretação dos dados coletados. Esse cenário pode ser alcançado através da digitalização de sistemas de instrumentação contribuindo assim, para o desenvolvimento de tecnologias de infraestrutura compreensível para os campos de produção e compartilhando conhecimento a fim de otimizar os processos de exploração e produção (SIRCAR *et al.*, 2021).

A utilização de Inteligência Artificial (IA), tem se mostrado uma ferramenta importante para as tomadas de decisões na indústria de óleo e gás (KOROTEEV; TEKIC, 2021), e com o subcampo dela, o *machine learning* (aprendizado de máquina), que deverá apresentar um crescimento ainda maior nos próximos anos, contribuindo com seu valor para o desenvolvimento desta indústria, pois a suas técnicas tem o potencial de aprimorar inúmeras ações críticas realizadas por administradores e engenheiros nesse setor.

Atualmente, as tarefas de *machine learning* (ML) podem ser classificados como tarefas de regressão, classificação, *clustering* (clusterização) e *non-clustering* (não clusterização), existindo diversos algoritmos com diferentes abordagens para suas soluções. Entre eles pode-se citar a regressão linear, árvores de decisão,*support vector machine* (máquinas de

vetores de suporte), redes neurais e *Deep Learning* (aprendizagem profunda), entre outros.

Para a aquisição de dados petrofísicos, perfis de poços de petróleo são essenciais para o desenvolvimento e melhor compreensão dos campos de produção. As ferramentas de perfilagem descidas durante ou após a perfuração dos poços apresentam diferentes propriedades para a aquisição de uma série de dados essenciais para a exploração, como propriedades elétricas, acústicas, mecânicas, radioativas, entre outras, que são registradas no perfil geofísico. Durante a coleta de dados de poços, podem ser retiradas amostras laterais que são utilizadas para identificação de dados diretos da rocha, de grande importância para a calibração de perfis do poço. Uma das dificuldades relacionadas a coleta de amostras laterais é a identificação da profundidade real de coleta em imagens acústica devido a imprecisão da ferramenta quanto à profundidade que a amostra foi coletada. A identificação dessas amostras, por intermédio de técnicas computadorizadas seria de extrema ajuda para os especialistas que fazem sua classificação, podendo reduzir consideravelmente o tempo investido na identificação dessas amostras permitindo que eles utilizem melhor o seu tempo para a interpretação dos dados das amostras ou do perfil estudado.

1.1 Objetivo

1.1.1 Geral

O presente trabalho tem como objetivo geral a identificação de pontos de coleta de amostras laterais no poço 7-BUZ-12-RJS utilizando aprendizado de máquina.

1.1.2 Específico

O presente trabalho de TCC tem como objetivos específicos:

- Montar um banco de dados para o treinamento dos modelos de aprendizado de máquina
- Implementar algoritmos de aprendizado de máquina para a classificação e localização de coleta de amostras em imagens de perfis de poço, sendo possível a identificação de áreas onde houve a coleta de amostras laterais em perfis de imagem acústica de poços;

2 REVISÃO BIBLIOGRÁFICA

2.1 Perfilagem

A perfilagem de poço é a prática de realizar registro detalhado das formações geológicas atravessadas durante a perfuração do poço. Os perfis geofísicos (veja Figura 1) são medidas indiretas, ou seja, são uma interpretação feita com base nas propriedades físicas das rochas, que fornece dados das rochas e fluidos da formação. Já as medidas diretas, que são obtidas através de ensaios feitos em amostras da formação, através da coleta de amostras do poço, podendo ser amostras laterais ou testemunhos (ASSAIFE, 2022).

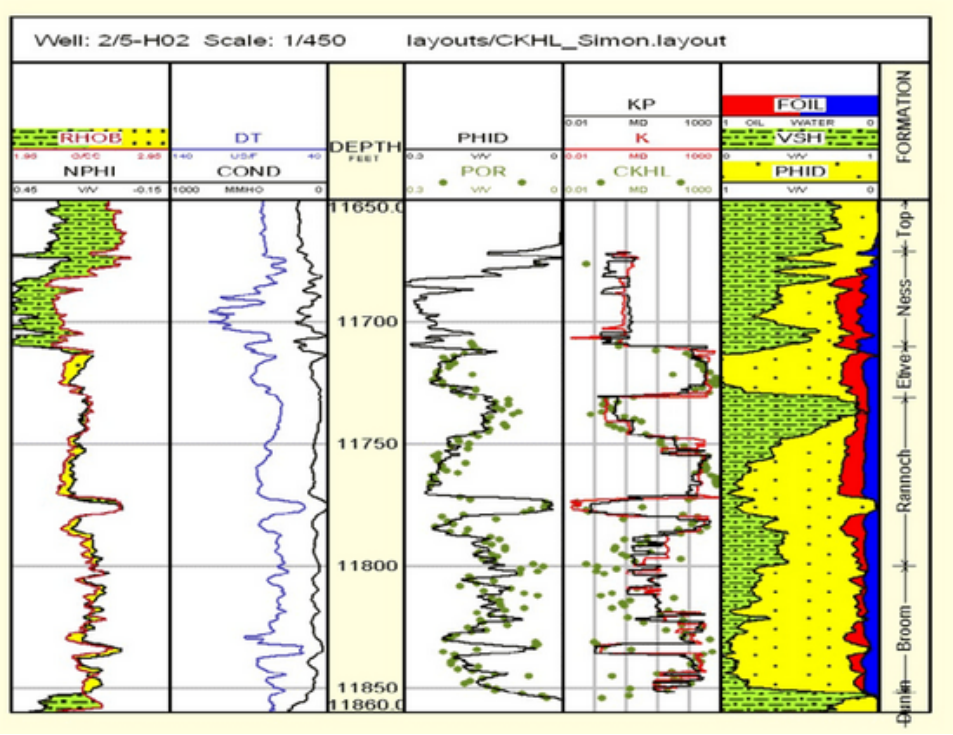


Figura 1: Exemplo de perfis de poços
Fonte: Assaife (2022)

As principais informações que costumam ser adquiridas através de perfilagem são: perfis de raios gama, de resistividade, de densidade, de nêutrons, de ressonância Magnética,

de espectroscopia de raios gama, litogeoquímico, testes de pressão, perfis de imagem resistiva, de imagem acústica, coleta de amostras laterais e amostragem de fluidos. Cada uma dessas ferramentas tem papel importante no melhor desenvolvimento do campo.

2.2 Coleta de Amostras

Amostras laterais (Figura 2) são amostras de rocha cilíndricas de 5" de comprimento, retiradas da parede do poço por uma ferramenta de perfilagem a cabo (Figura 3), composta basicamente por uma pequena broca de diâmetro de 1 ou 1,5" (Figura 4) e tubos bipartidos para o armazenamento das amostras.

Figura 2: Amostras laterais de rocha
Fonte: "SCHLUMBERGER"... (2022)

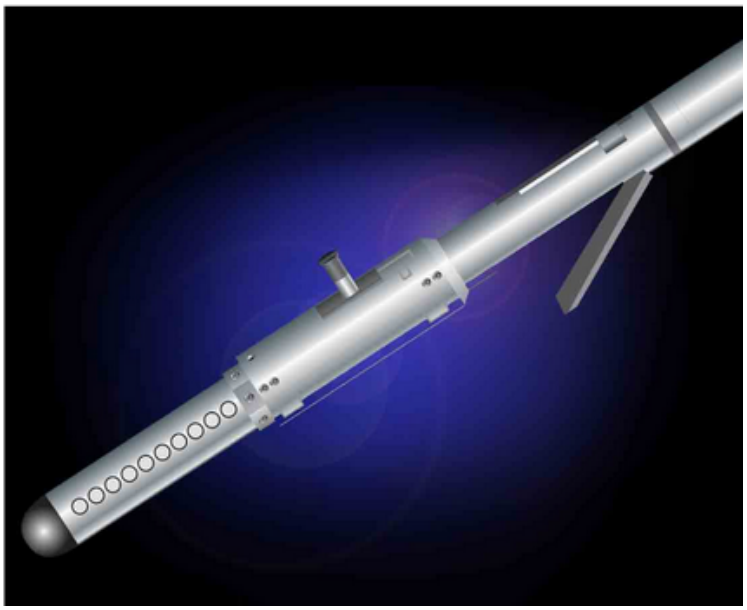


Figura 3: Exemplo de ferramenta de amostragem lateral MSCT da Schlumberger
Fonte: "SCHLUMBERGER"... (2022)

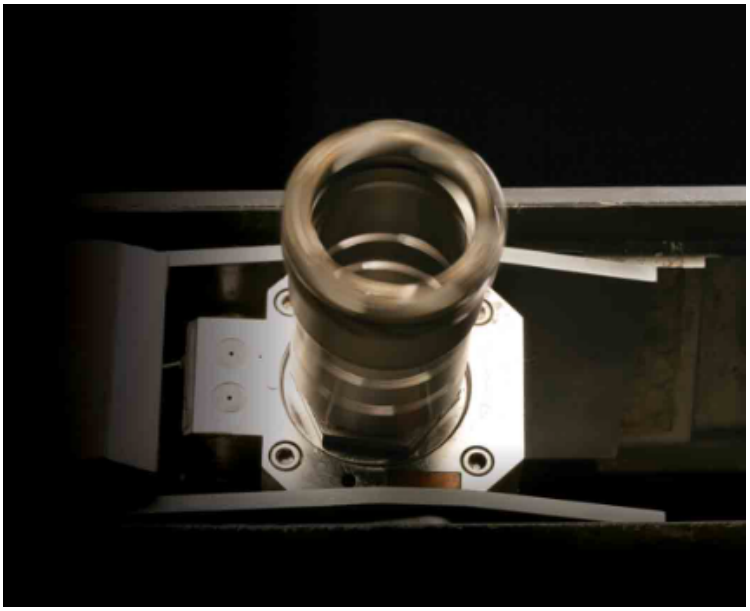


Figura 4: Broca para corte das amostras laterais
Fonte: "SCHLUMBERGER"... (2022)

Meio a tantos dados indiretos obtidos em uma perfilagem, as amostras laterais se destacam como dados diretos da rocha, importantíssimos para a calibração dos demais perfis petrofísicos e melhor compreensão das rochas. Após a aquisição essas amostras passam por limpeza, descrição macro e microscópica e uma série de ensaios de laboratório que fornecerão informações como datação, porosidade, permeabilidade, granulometria, molhabilidade, entre outros.

É inerente à aquisição dessas amostras, que haja imprecisão quanto à profundidade de retirada, que se dá pelo esticamento do cabo e movimentação do guincho que sustenta a ferramenta. Dessa forma, para a melhor correlação dessas imagens com os demais perfis e posicionamento preciso na coluna geológica, utiliza-se a identificação visual das amostras retiradas nos perfis de imagem acústica adquiridos após a coleta. As imagens acústicas refletem bem a parede do poço e por isso, os orifícios deixados após a coleta são de fácil identificação nessas imagens, conforme a Figura 5.

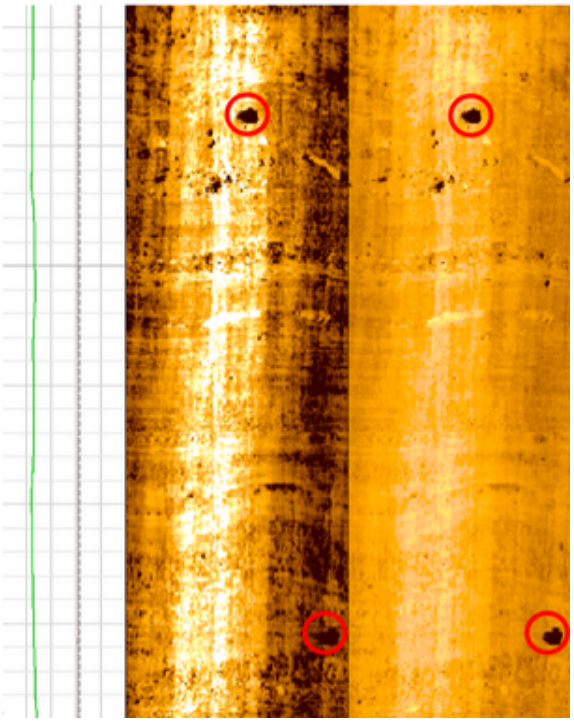


Figura 5: Identificação em imagens acústicas, dos orifícios deixados pela retirada de amostras laterais.
Fonte: Elaboração Própria

Pacheco menciona que atualmente, esse reposicionamento das amostras laterais, é feito amostra a amostra pelo petrofísico. Trata-se de um trabalho moroso, mas de extrema importância, uma vez que as informações detalhadas obtidas das amostras laterais, se posicionadas em uma camada equivocada podem prejudicar o modelo geológico, levando a uma série de erros cumulativos (informação pessoal)[1].

2.3 *Machine Learning* (ML)

Mitchell (2013) define ML como, "Um programa de computador que aprende pela experiência E, em relação a algum tipo de tarefa T e performance P, se sua performance P na tarefa T melhoram com sua experiência E." É a técnica de programar computadores para melhorar sua performance a partir de experiências prévias. O aprendizado da máquina é na realidade a execução de programas que otimizam parâmetros do seu modelo com dados de treinamento ou experiência de execuções passadas. Seus modelos podem ser preditivos, descritivos ou ambos (ALPAYDIN, 2010).

Embora o ML seja associado à inteligência artificial, outras áreas de pesquisa, como

[1]Pacheco, E. K. F. via mensagem eletrônica enviada à destinatária Cecília Garcia, em 20 de julho de 2022.

probabilidade e estatística, teoria da computação e teoria da informação, entre outras, tem contribuição direta para os avanços de métodos de aprendizado de máquina (FACELI *et al.*, 2011). Alpaydin (2010) complementa afirmando que o *Machine Learning* é um subconjunto derivado da Inteligência Artificial e que utiliza teorias estatísticas nos seus modelos matemáticos. Para isso, o que esses algoritmos realmente fazem é deduzir uma hipótese que é capaz predizer resultados para dados que não foram usados no treinamento do algoritmo, ou seja, dados não apresentados no seu treinamento. Essa capacidade do algoritmo de inferir uma hipótese que ainda é aplicável para novos dados de domínio similar é chamada de capacidade de generalização de hipótese (FACELI *et al.*, 2011).

Entretanto, há também hipóteses com baixa capacidade de generalização, que são muito especializadas no conjunto de treinamento, não sendo capaz de predizer bons resultados para dados novos. Esses casos são chamados de *overfitting* (sobreajuste). O inverso do *overfitting* seria o *underfitting* ou subajustamento, no qual o algoritmo desenvolve uma hipótese com menor acurácia. O *underfitting* pode ocorrer devido a dados não representativos, ou o modelo de ML utilizado ser muito simples, com poucos parâmetros, ou cuja parametrização não seja capaz de representar a distribuição de dados, não conseguindo representar os dados utilizados (FACELI *et al.*, 2011).

A Figura 6 apresenta uma representação gráfica de cada uma das possíveis hipóteses de um algoritmo após a fase de treinamento.

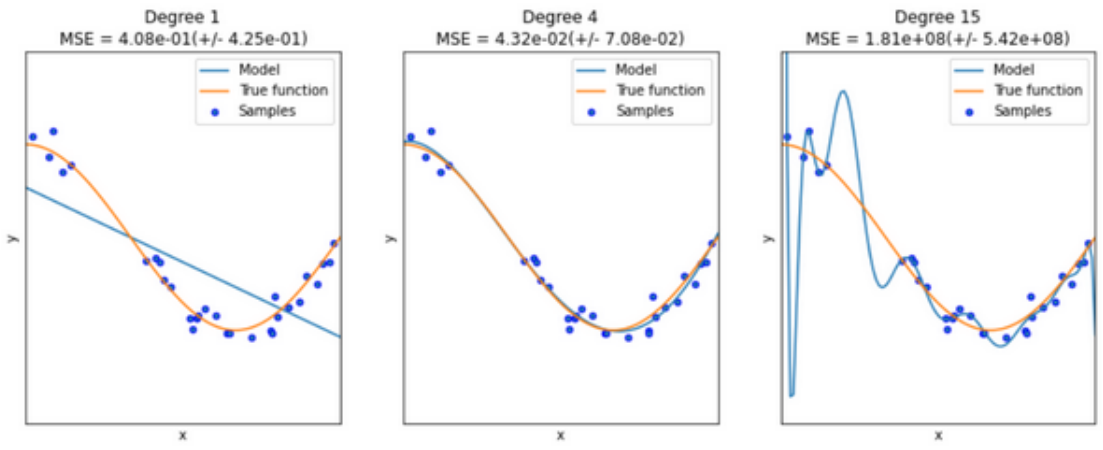


Figura 6: Representação gráfica de casos de *underfitting* e *overfitting*
Fonte: DATASCIENCE... (2022)

Além disso, cada algoritmo de ML apresenta dois vieses, um viés de representação e um viés de busca. Esses vieses são necessários para restringir o número de hipóteses a serem visitadas no espaço de busca, que segundo Faceli *et al.* (2011), sem um viés o algoritmo

de aprendizado de máquina não conseguiria generalizar o conhecimento adquirido no seu treinamento e fazer a sua aplicação com sucesso em novos dados.

Ademais, em muitos casos, algoritmos de ML precisam utilizar dados imperfeitos para fazer suas previsões, como dados ruidosos, inconsistentes, ou com valores ausentes. Por esse motivo muitos algoritmos utilizam técnicas de pré-processamento para evitar ou minimizar o efeito dessas inconsistências (FACELI *et al.*, 2011).

Uma das principais distinções entre os tipos de algoritmos de aprendizado de máquina é a forma que o algoritmo aprende e treina. É durante a fase de treinamento que o algoritmo aprimora a sua hipótese, e onde ele pode se tornar especializado no conjunto de treinamento (*overfitting*) ou é pouco treinado e não tem uma taxa de acerto satisfatória (*underfitting*).

Os métodos de aprendizados mais comuns são o aprendizado supervisionado, e o não supervisionado. A Figura 7 apresenta um esquema com os diferentes tipos de aprendizado, segundo Faceli *et al.* (2011). Há ainda o aprendizado por reforço, que é mais complexo e tem lógica de aplicação diferente (MITCHELL, 2013).

Figura 7: Hierarquia de aprendizado
Fonte: Faceli *et al.* (2011)

2.3.1 Aprendizado supervisionado

O aprendizado supervisionado é utilizado por alguns algoritmos, e consiste na utilização dos resultados do conjunto de dados para ajustar a hipótese conforme ocorre o treinamento. Faceli *et al.* (2011) faz a relação com o termo supervisionado que vem da simulação e da presença de um supervisor externo, que conhece o resultado para o conjunto de dados.

Esse tipo de aprendizado é utilizado para resolver tarefas de previsão onde é encontrada uma hipótese ou função a partir dos dados de treinamento para ser utilizado com conjuntos de novos dados para prever uma resposta. Esses algoritmos caracterizam modelos preditivos. Alguns modelos que utilizam esse método de treinamento são Árvores de Decisão, Algoritmo de Naïve Bayes, Redes Neurais, Regressão Linear e Logística, entre outros (FACELI *et al.*, 2011).

2.3.2 Aprendizado não supervisionado e por reforço

O aprendizado não supervisionada é aquele que aprende com dados de teste sem classificação. Para tarefas de descrição, o objetivo do algoritmo é explorar e identificar formas de representar um conjunto de dados. Por tanto, algoritmos utilizados para esse tipo de tarefa não fazem a comparação entre os dados de saída e do rótulo dos dados de treino, caracterizando o aprendizado não supervisionado. Alguns exemplos desse tipo de tarefa são o agrupamento de um conjunto de dados e a identificação de regras de associação entre grupos de atributos diferentes (FACELI *et al.*, 2011).

Para problemas que caracterizam o aprendizado por reforço, o algoritmo durante o treinamento tem como objetivo reforçar ou recompensar uma ação considerada positiva e de punir uma ação considerada negativa (FACELI *et al.*, 2011).

Um fator que pode dificultar o aprendizado por reforço do algoritmo é quando o sistema tem informações sensoriais não confiáveis ou enviesadas, aumentando a incerteza que deve ser levada em consideração no algoritmo. É possível também que uma tarefa demande a operação de múltiplos agentes que precisam interagir entre si para conseguir completar a tarefa ou objetivo comum, dificultando a implementação desse tipo de algoritmo (ALPAYDIN, 2010).

2.4 Classificação

O objetivo de algoritmos de classificação é a partir de um conjunto de dados de entrada definir um resultado de um conjunto de respostas com valor discreto. Na maioria dos casos nesse conjunto as respostas tem baixa correlação entre si, para que apenas uma seja definida como resultado para aqueles dados de entrada específicos (BISHOP, 2006).

Após a fase de treinamento o algoritmo pode determinar uma regra de descriminação entre as classes. Essa regra é uma função que separa os dados de treinamento para as diferentes classes de classificação, e se essa regra conseguir ser abstraída ela pode fazer a

classificação de novos dados que o algoritmo ainda não teve contato (ALPAYDIN, 2010).

Para modelos probabilísticos, a regra de classificação para algoritmos com apenas duas respostas, ou duas classes, retorna uma representação binária, onde o resultado varia em um intervalo de 0 a 1, para valores próximos ou igual a um (1) representaria uma classe, e para valores próximos ou igual a zero (0) representaria a outra classe. Para modelos onde há mais de duas classes para atribuir ao conjunto de entrada é utilizado vetores com k elementos, onde k seria o número de classes utilizadas para a classificação, e onde todos os valores variariam de 0 a 1, sendo a posição com valor mais próximo de 1 o resultado da classificação (BISHOP, 2006).

Em alguns casos esse tipo de classificação pode representar a probabilidade condicional P(Y|X) de um determinado conjunto de dados ser classificado com uma determinada classe, onde X seria os dados de entrada, e Y seria uma das classes utilizada pelo algoritmo (ALPAYDIN, 2010).

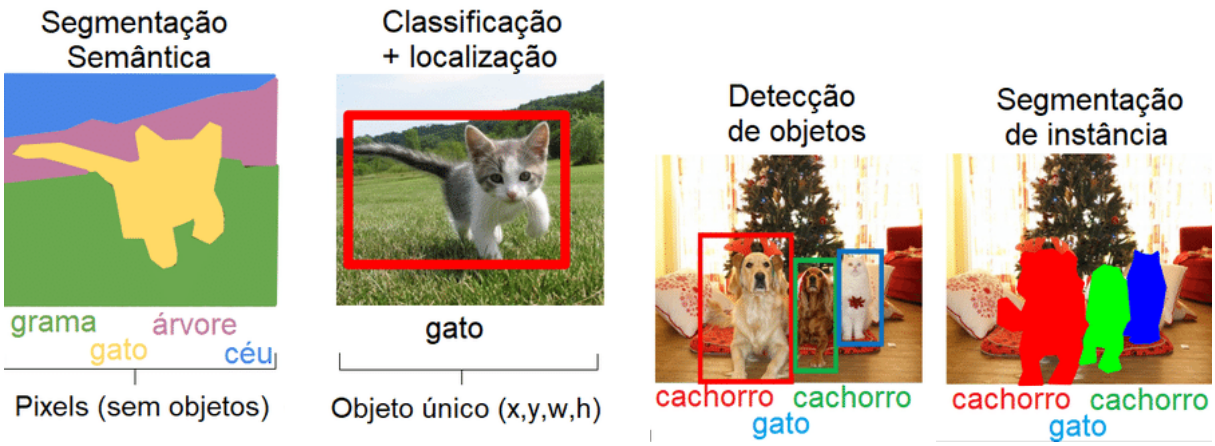
2.4.1 Segmentação de Imagens

A segmentação de imagem é a tarefa de particionar uma imagem em múltiplos segmentos (GÉRON, 2019). Bishop (2006) descreve que o objetivo da segmentação é o particionamento de imagens em regiões onde há homogeneidade visual aparente ou a qual corresponde a um objeto ou parte de um objeto. Os algoritmos de segmentação não convolucionais tratam cada pixel na imagem como pontos independentes caracterizados por dados, que são utilizados para que a classificação do pixel seja feita (BISHOP, 2006).

Géron (2019) divide a segmentação de imagens em dois tipos: *"semantic segmentation"* (segmentação semântica) e *"instance segmentation"* (segmentação de instância). Em segmentação semântica todos os pixels que fazem parte do mesmo tipo de objeto são designados em um mesmo segmento. Em um exemplo, o sistema visual de um carro autônomo identificaria todos os pixels que compõem a imagem de um pedestre como do tipo "pedestre", sem fazer a distinção se há um ou mais pedestres na imagem, sobrepostos ou não, todos os pixels serão classificados como a classe "pedestre". Já em segmentação de instância há a diferenciação entre as instâncias do tipo segmentado. No exemplo anterior haveria a diferenciação entre os pedestre identificados na imagem original (GÉRON, 2019).

A Figura 8 ilustra alguns algoritmos de classificação de imagens, a Figura 8a representam os dados de saída de um algoritmo de segmentação semântica, e a Figura 8d

representa os dados de saída de um algoritmo de segmentação de instância.



(a) (b) (c) (d)

Figura 8: Algoritmos de classificação de imagens e seus valores de saída (a) segmentação semântica; (b) classificação e localização (c) Detecção de objetos; (d)Segmentação de instância.
Fonte: Olivatto (2021)

2.4.2 Algoritmos de Classificação

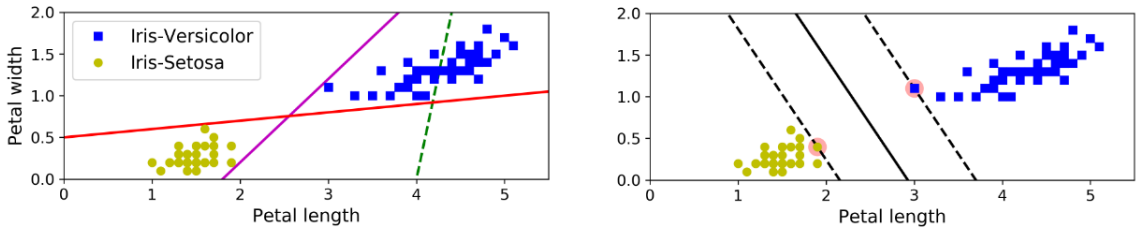
Dentro dos algoritmos de classificação de imagens há modelos clássicos de aprendizado de máquina que podem ser utilizados para fazer essa tarefa, porém são mais simples e menos eficientes para classificar imagens mais complexas. Alguns desses algoritmos são o *Support Vector Machine* (Máquina de Vetores de Suporte) e o *Random Forest* (RF) que serão abordados a seguir.

2.4.2.1 *Support Vector Machine* (SVM)

Uma SVM é um modelo de aprendizado de máquina capaz de performar classificação linear e não linear, regressão e até mesmo detecção de *outliers*. SVMs são excelentes para a classificação de *datasets* complexos, mas que não são muito extensos, devido ao seu custo computacional (GÉRON, 2019).

O modelo SVM é uma representação de pontos no espaço, mapeados para que cada categoria seja divididos por um hiperplano que deve estar o mais distante possível dos conjuntos que ele divide. A classificação de novos pontos é feita baseada no posicionamento do ponto em relação ao hiperplano. A posição do hiperplano é determinada pelos pontos mais próximos de cada classe, e ele é equidistante de ambos. Esses pontos mais próximos são chamados de Vetores de Suporte (GÉRON, 2019). A Figura 9 ilustra graficamente

uma SVM em duas dimensões.



(a) (b)

Figura 9: Representação gráfica de valores de um *dataset* em um plano. (a) representação gráfica de possíveis hiperplanos que podem dividir o *dataset* as duas classes; (b), representação de uma SVM com hiperplano equidistante dos vetores de suporte
Fonte: Géron (2019).

2.4.2.2 *Random Forest* (RF)

As RF são um *ensemble* de várias árvores de decisão, sendo um algoritmo de aprendizado supervisionado (JIANG *et al.*, 2023). O *ensemble* funciona combinando várias árvores de decisão, nas quais cada uma é treinada em um subconjunto aleatório dos dados de treinamento e características selecionadas de forma aleatória. Após a construção de todas as árvores, a RF faz previsões por meio da votação (classificação) ou da média (regressão) das saídas de todas as árvores para aquele conjunto de entrada (KHAN *et al.*, 2021).

As árvores de decisão são um algoritmo menos complexo, que subdivide um problema mais complexo em problemas menores que quando combinados em forma de árvore são a representação da solução desse problema (FACELI *et al.*, 2011). As árvores de decisão são compostas por nós, no topo da árvore são inseridos os valores de entrada em um nó de divisão que contém um teste condicional, e a partir do resultado deste o nó seguinte no mesmo galho é utilizado como nova condição. Esse processo é repetido até os nós folhas, chamados assim por estarem nas extremidades de um galho da árvore de decisão, que contem uma função que representativa da solução do problema para aquele conjunto de entrada (FACELI *et al.*, 2011). A Figura 10 representa uma árvore de decisão simples.

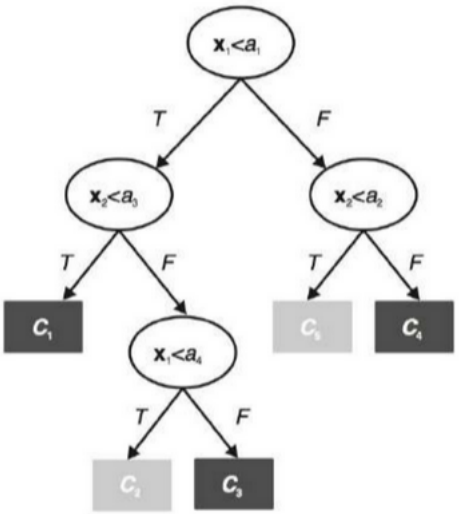


Figura 10: Exemplo de Árvore de Decisão
Fonte: Faceli *et al.* (2011)

A Figura 11 ilustra um algoritmo de RF. Nela são montadas diferentes árvores de decisão utilizando diferentes subconjuntos dos dados de entrada originais. Ao final, o modelo coleta os resultados de todas as árvores e calcula o resultado final através da votação para modelos de classificação, e da média dos resultados para modelos de regressão.

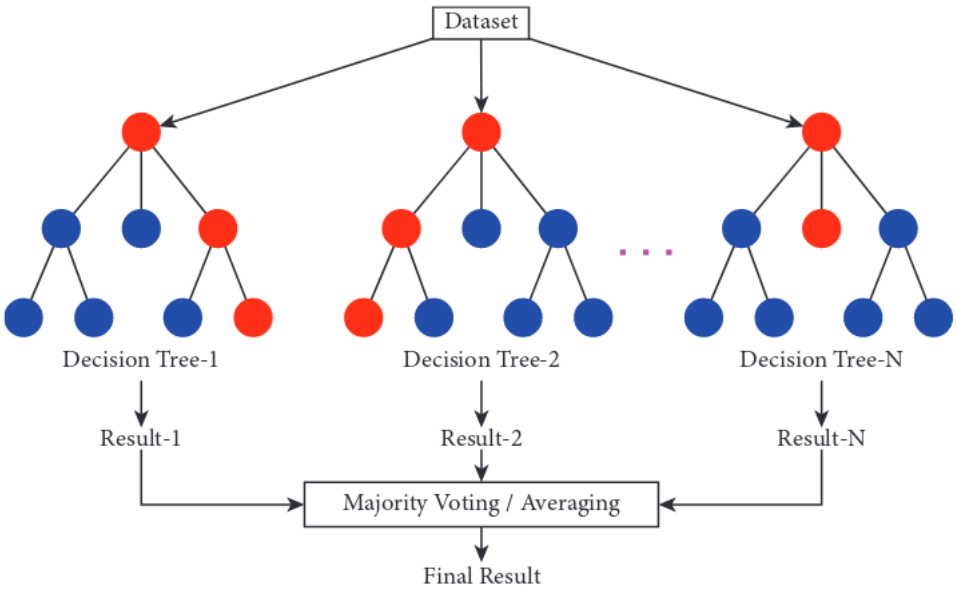


Figura 11: Representação de Random Forest
Fonte: Khan *et al.* (2021)

Uma das limitações das árvores de decisão é que elas são muito sensíveis à variações nos dados de treinamento. Uma pequena variação nesses dados e a árvore resultante pode ser completamente diferente da árvore instanciada inicialmente (GÉRON, 2019). Por conta

disso, a solução de votação feita em RF ameniza o problema de *overfitting* que árvores de decisão sofrem, diminuindo sua dependência em relação aos dados de treinamento (KHAN *et al.*, 2021).

2.5 Redes Neurais

Uma rede neural é um modelo computacional inspirado no funcionamento do cérebro humano, projetado para realizar tarefas complexas de aprendizado e reconhecimento de padrões. Composta por camadas interconectadas de unidades de processamento, uma rede neural é capaz de aprender e generalizar informações a partir de conjuntos de dados.

2.5.1 Inspiração

Mitchell (2013) afirma que o desenvolvimento de redes neurais foi inspirado de certa forma pela observação de sistemas de aprendizado biológicos, e de como eles são compostos por redes complexas de neurônios interconectados. O termo 'Redes Neurais' também teve origem da inspiração de sistemas de processamento de informações biológicos (BISHOP, 2006). Mesmo que a inspiração das *artificial neural networks* (ANN) sejam sistemas neurais biológicos, há diversas complexidades dos sistemas biológicos que não são ou não podem ser modelados pelas ANNs (MITCHELL, 2013).

2.5.2 Neurônios

Segundo Piccinini (2004) o primeiro neurônio foi proposto pelo neurofisiologista Warren McCulloch e o matemático Walter Pitts. O modelo do neurônio artificial tinha como conceito conter uma ou mais entradas de dados binários, e conter uma única saída binária. O neurônio também só seria ativado e retornava uma saída quando uma quantia pré-estabelecida de dados de entrada eram usados (GÉRON, 2019).

Uma das arquiteturas mais simples de redes neurais chamada *Perceptron* foi criada em 1997 por Frank Rosenblatt, e utilizava um neurônio artificial conhecido como TLU (*Threshold Logic Unit*) (Figura 12), que tem números como dados de entrada e saída, ao contrário dos valores binários do modelo de McCulloch e Pitts, e cada valor de entrada é associado a um peso específico (GÉRON, 2019).

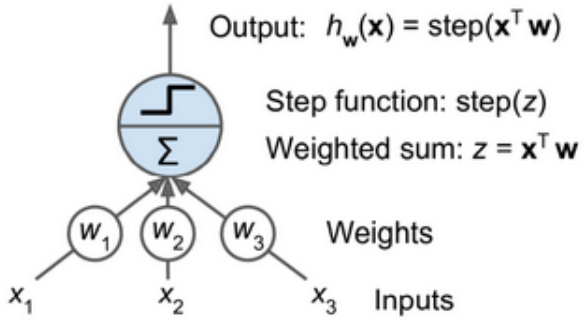


Figura 12: Esquema de um neurônio artificial TLU
Fonte: Géron (2019)

O TLU calcula a soma dos seus valores de entrada com seus respectivos pesos e aplica o resultado em uma função *step* resultando no valor de saída do neurônio. Essa função pode variar de acordo com a necessidade do algoritmo. No treinamento dessas redes eram alterados os valores dos pesos para melhorar os resultados, e determinar uma hipótese que descreve bem o problema (GÉRON, 2019).

2.5.3 *Deep Learning*

Deep Learning (Aprendizagem Profunda) é uma subcategoria de ML onde essas estruturas aprendem e interpretam o conceito dos dados. As redes neurais são um dos tipos de algoritmos utilizados em ML para que seja feito o modelamento de dados. Na indústria de Petróleo e Gás essas estruturas são utilizadas para processar grandes quantidades de dados e para conseguir uma boa performance nestes tipos de processamento. Alguns exemplos de trabalhos que aplicam redes neurais nessa área são o trabalho de Sheikhoushaghi, Gharaei e Nikoofard (2022), que aplica redes neurais para o cálculo do taxa de produção de um campo de óleo, e o trabalho de Al-AbdulJabbar *et al.* (2022), no qual foi desenvolvido um modelo que calcula a velocidade de perfuração da rocha em um campo de formação carbonática, utilizando dados de outros poços para treinamento. Segundo Sircar *et al.* (2021) as *features* (atributos) utilizadas são escolhidas sem nenhuma intervenção humana. Algoritmos de *Deep Learning* são utilizados para executar operações mais complexas, onde os algoritmos de ML não teriam uma performance plausível.

Um exemplo de rede neural extraído de Diersen *et al.* (2011) é apresentado na Figura 13. Nela se observa que a arquitetura das redes neurais é composta por neurônios (nodes) que são conectados por ligamentos (links) entre si. Cada neurônio recebe sinais de entrada, que são tratados pela sua função de ativação e retornam um sinal de saída. As funções de ativação são únicas de cada neurônio e seus links têm um peso próprio

ditando a importância de cada sinal de entrada para a função de ativação. Os sinais de saída dos neurônios são utilizados como entrada para outros neurônios.

Com esses elementos básicos, as redes neurais são dispostas em camadas, sendo apenas uma a *Input Layer* (camada de entrada), apenas uma *Output Layer* (camada de saída) e podendo ter *n Hidden Layers* (camadas intermediárias) que são compostas por neurônios e ligamentos com suas respectivas funções de ativação e pesos. São nas *Hidden Layers* que são dadas as entradas de sinais das camadas anteriores e é feita a atribuição de valores para cada neurônio a partir da sua função de ativação. Esse valor é então enviado como sinal de saída para a próxima camada da rede (DIERSEN *et al.*, 2011). O nome *Hidden* é utilizado pois os dados de saída da camada só podem ser acessados de dentro da rede, sendo considerados como uma "caixa preta" para um observador externo à rede (MITCHELL, 2013).

Na sua arquitetura, também são utilizados ligamentos ou unidades de viés (bias links). Eles atuam para ajustar o sinal de saída do neurônio, ditando um valor mínimo a função de ativação, com o intuito de aumentar o grau de liberdade da rede (DIERSEN *et al.*, 2011).

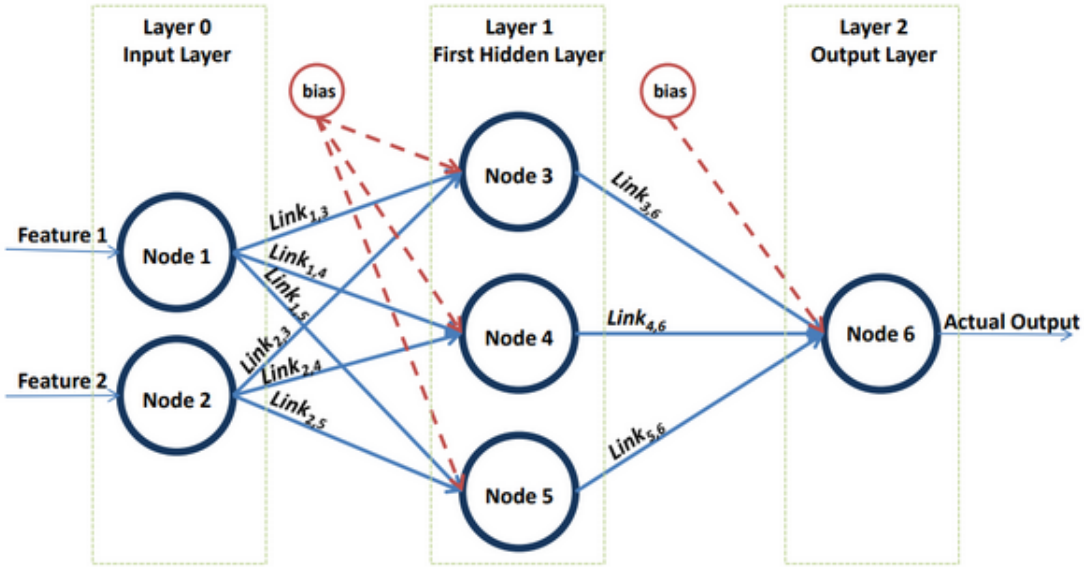


Figura 13: Exemplo de Rede Neural
Fonte: Diersen *et al.* (2011)

Rumelhart, Hinton e Williams (1986) propuseram o algoritmo de treinamento conhecido como *backpropagation* (retropropagação), que emprega o *gradient descent* (método do gradiente descendente) para otimizar um modelo por meio de uma técnica computacional eficiente (GÉRON, 2019). Este algoritmo ainda é amplamente utilizado na atualidade

e envolve duas fases distintas: a fase *forward* (fase para frente) e a fase *backward* (fase para trás) (FACELI *et al.*, 2011).

Na fase *forward*, os valores de entrada e saída de cada camada são transmitidos para a camada subsequente. Ao término da primeira fase, os valores de saída da rede são comparados com os valores esperados para os dados de entrada, e a diferença entre eles representa o erro de *output* (FACELI *et al.*, 2011).

Uma vez que o erro é obtido, a rede identifica quais neurônios de saída contribuíram mais significativamente para esse erro. Essa análise é conduzida de forma analítica, fazendo uso da regra da cadeia. Em seguida, é avaliada a contribuição para o erro na camada anterior à camada atualmente analisada. Esse processo é repetido iterativamente até atingir a camada de entrada. Dessa maneira, a rede é capaz de calcular o gradiente do erro em relação à camada de *output*, ao longo de toda a sua extensão. Por fim, utilizando o gradiente do erro, o algoritmo ajusta os pesos das conexões dos neurônios aplicando o método do gradiente descendente (GÉRON, 2019).

2.5.4 Redes Neurais Convolucionais

As redes convolucionais (CNNs) tiveram sua origem nos estudos sobre o córtex visual do cérebro humano e têm sido aplicadas no reconhecimento de imagens desde a década de 1980. Essas redes apresentam camadas distintas em relação às redes neurais tradicionais, incluindo camadas convolucionais que nomeiam a rede e camadas de *pooling* (agrupamento) (GÉRON, 2019).

Nas camadas convolucionais, os neurônios não se conectam diretamente a cada pixel da imagem de entrada, mas, em vez disso, respondem a uma pequena área retangular específica. Essa arquitetura permite que as camadas iniciais da rede se foquem em detalhes de baixo nível na imagem. À medida que as convoluções ocorrem, esses detalhes se agregam para identificar características mais complexas e de alto nível nas camadas subsequentes (GÉRON, 2019). Essa abordagem encontra respaldo na obra de Bishop (2006), que enfatiza a utilização dessa estratégia na visão computacional, onde propriedades das imagens são extraídas de pequenas sub-regiões, uma vez que pixels próximos em uma imagem tendem a exibir uma correlação mais acentuada do que aqueles situados a maiores distâncias.

A identificação de aspectos é feita por filtros com dimensões pré-definidas que permitem a extração de novas informações da imagem original, transformando-as em um *feature*

map (mapa de aspectos). A Figura 14 esquematiza de maneira ilustrativa a relação dos neurônios nas camadas convolucionais com os dados de entrada. Por sua vez, a Figura 15 exemplifica de forma explícita o impacto de cada filtro na imagem original, sendo que o primeiro realça as características verticais da imagem, enquanto o segundo enfatiza suas características horizontais (GÉRON, 2019).

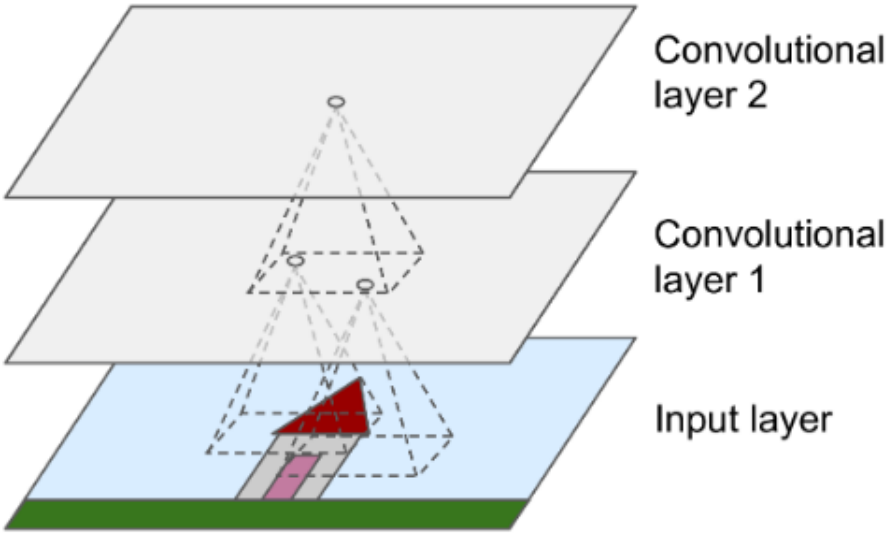


Figura 14: Camadas convolucionais e divisão das sub-áreas atribuídas a um neurônio
Fonte: Géron (2019)

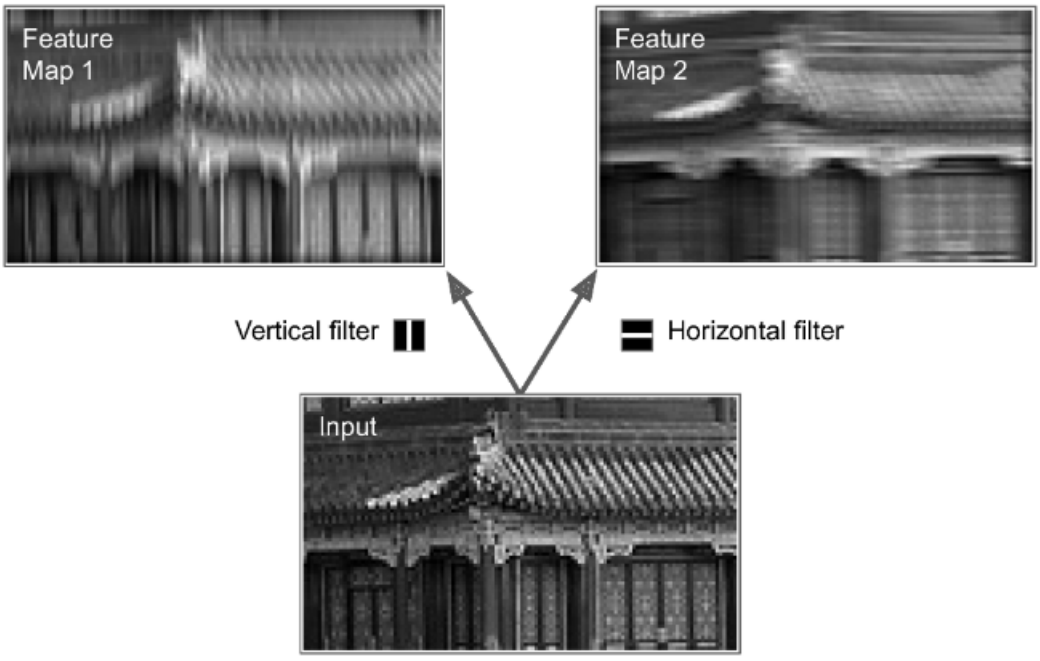
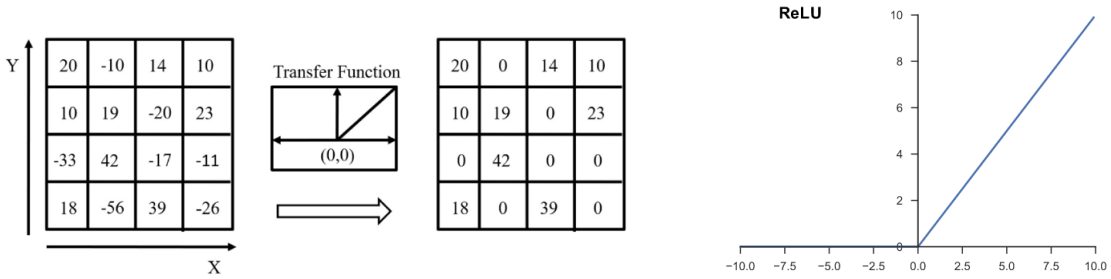


Figura 15: Aplicação de filtros em uma imagem
Fonte: Géron (2019)

Cada camada da rede neural aplica vários filtros distintos à imagem original, extraindo diferentes características da imagem de entrada para a camada convolucional. Além dos filtros, a função de ativação ReLU (Unidade Linear Retificada) é empregada para gerar os *feature map*. Esta função atribui o valor zero a entradas negativas e mantém inalteradas as entradas positivas. A Figura 16 ilustra a transformação dos *feature maps* antes e depois da aplicação da função de ativação ReLU. Com base nisso, uma camada convolucional pode ser representada como a sobreposição de diversos *feature map*, cada um descrevendo um aspecto da imagem. Na Figura 17 é apresentada uma camada convolucional, com diversos *feature maps* sobrepostos, e alguns dos filtros utilizados para cada mapa (GÉRON, 2019).



(a) (b)

Figura 16: Função Unidade Linear Retificada aplicada nas camadas convolucionais:
(a) Alteração do mapa de aspectos pela função de ativação ReLU. Fonte: Parab e Mehendale (2021);
(b) Gráfico da função ReLU. Fonte: Yamashita *et al.* (2018).

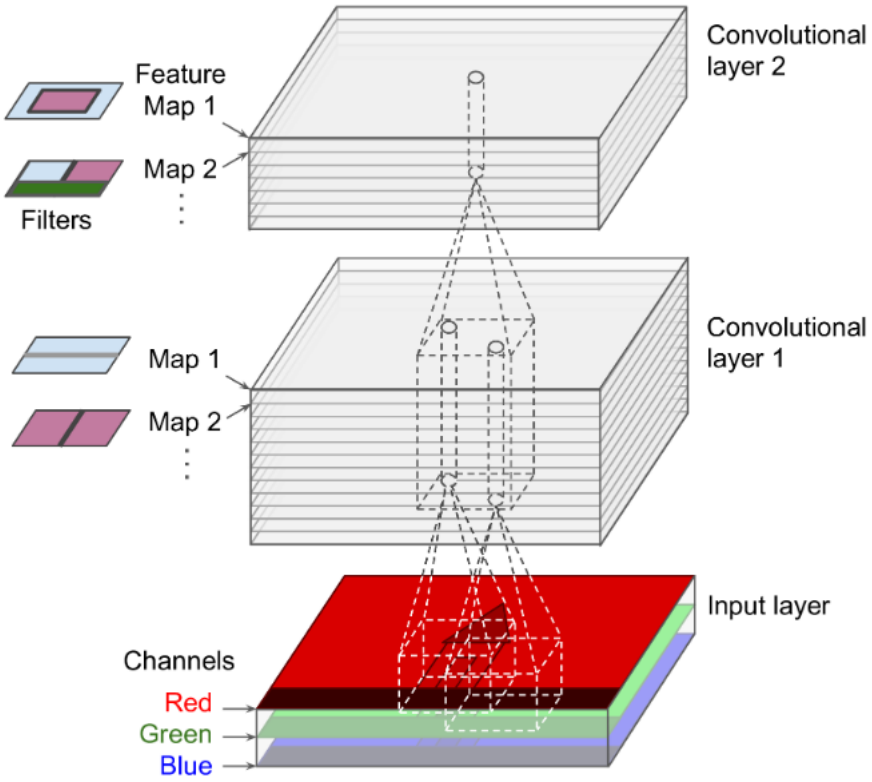


Figura 17: Estrutura de camadas convolucionais e como seus neurônios se relacionam
Fonte: Géron (2019)

Outra camada fundamental nas CNNs é a camada de *pooling*, a qual é empregada para reduzir as dimensões dos *feature maps* enquanto ainda retém as características previamente mapeadas. Nessa camada, os neurônios não possuem pesos; em vez disso, eles realizam uma agregação dos valores de entrada utilizando uma função de agregação, podendo ser a média dos valores de entrada ou, mais comumente, o valor máximo entre os elementos que serão agregados. A Figura 18 demonstra o funcionamento da agregação nesse tipo de camada, com um *kernel* (filtro) de dimensão 2x2. O conjunto de pixels da imagem original é condensado em um único pixel, que armazena o valor máximo entre os quatro pixels originais (GÉRON, 2019).

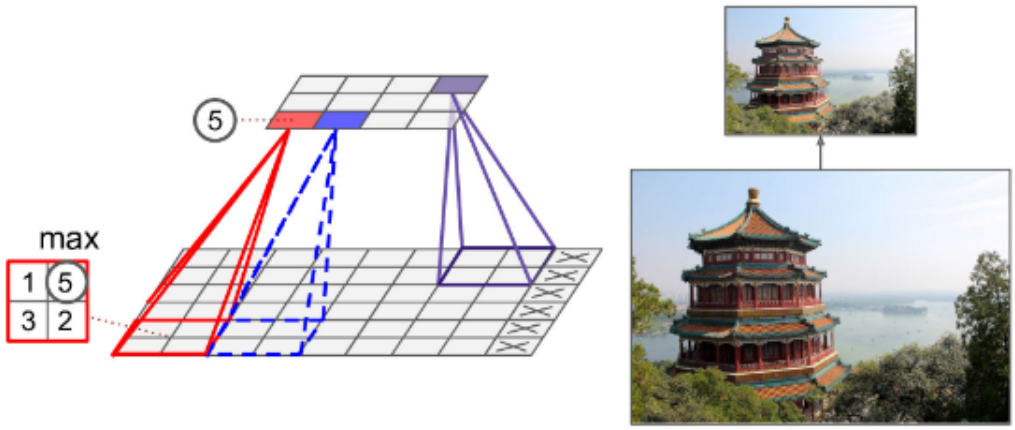


Figura 18: Camada de *pooling* utilizando o máximo valor como parâmetro da função de agregação
Fontes: Géron (2019)

Na arquitetura de uma Rede Convolucional, esse tipo de camada é empregado para reduzir as dimensões dos *feature maps* gerados na camada convolucional, contribuindo para a economia de recursos computacionais, enquanto ainda preserva as características extraídas da imagem original. As arquiteturas mais básicas das CNNs consistem em camadas convolucionais alternadas com camadas de *pooling*. Na etapa final da rede, são adicionadas camadas *fully connected* (camada densa), onde a primeira camada possui o mesmo número de neurônios que o número de pixels dos *feature maps* criados na última camada convolucional ou de *pooling* da rede. As camadas subsequentes atuam como as *hidden layers* e a camada de *output* de uma rede neural. A Figura 19 exemplifica uma arquitetura simples de uma CNN, composta por duas camadas convolucionais, duas de *pooling*, duas *hidden layers* e uma camada de *output* (GÉRON, 2019).

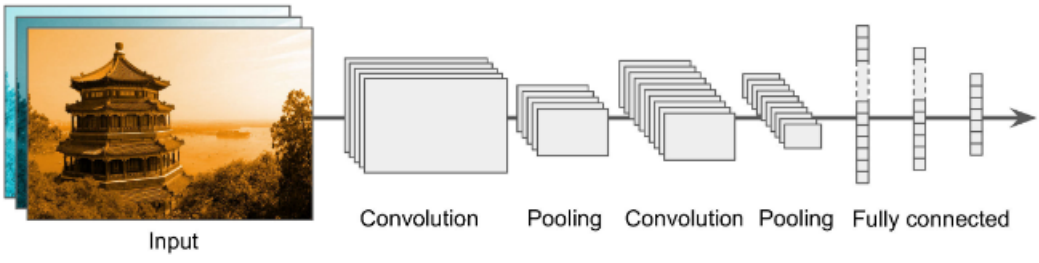


Figura 19: Arquitetura de uma CNN simples, utilizando camadas de convolucional, *pooling* e *fully connected*
Fonte: Géron (2019)

2.6 *Data Augmentation*

Para desenvolver modelos úteis de *Deep Learning*, é imperativo que o erro nos dados de validação continue a diminuir concomitantemente com o erro nos dados de treinamento. A técnica de *data augmentation* (aumento de dados) surge como um método essencial para atingir esse objetivo, além de contribuir para mitigar o fenômeno de *overfitting* do modelo. *Data augmentation*, como o próprio nome sugere, envolve a expansão do *dataset* artificialmente. Isso pode ser realizado por meio da manipulação das imagens no conjunto de dados original ou pela criação de novas instâncias sintéticas dos dados de entrada. Essa abordagem adquiriu uma relevância significativa devido à crescente necessidade de utilizar conjuntos de dados extensos, especialmente em modelos de *deep learning* mais complexos, como as redes convolucionais (CNNs), amplamente empregadas na análise de imagens (SHORTEN; KHOSHGOFTAAR, 2019).

A necessidade de lidar com um grande volume de dados ou a complexidade de organizar um número substancial de dados em algumas áreas, como a medicina, que pode ter um número limitado de dados disponíveis, torna o *data augmentation* uma solução eficaz para o problema de *overfitting*. Outra situação na qual esse método pode ser útil para aprimorar a classificação do modelo é quando o *dataset* apresenta classes desbalanceadas, com um número significativamente maior de instâncias em uma classe em relação às outras (SHORTEN; KHOSHGOFTAAR, 2019).

A manipulação de imagens é uma das técnicas utilizadas em *data augmentation*. Nesse tipo de técnica, são aplicadas transformações na imagem original, tais como inversão do eixo horizontal ou vertical, alteração do espectro de cores, recorte, rotação, translação e outras. O objetivo desse tipo de *data augmentation* é unicamente alterar a imagem original (SHORTEN; KHOSHGOFTAAR, 2019). As Figura 20 e 21 ilustram algumas dessas transformações, incluindo inversão de eixo da imagem, rotação, alteração do espectro de cores e recorte da imagem original.

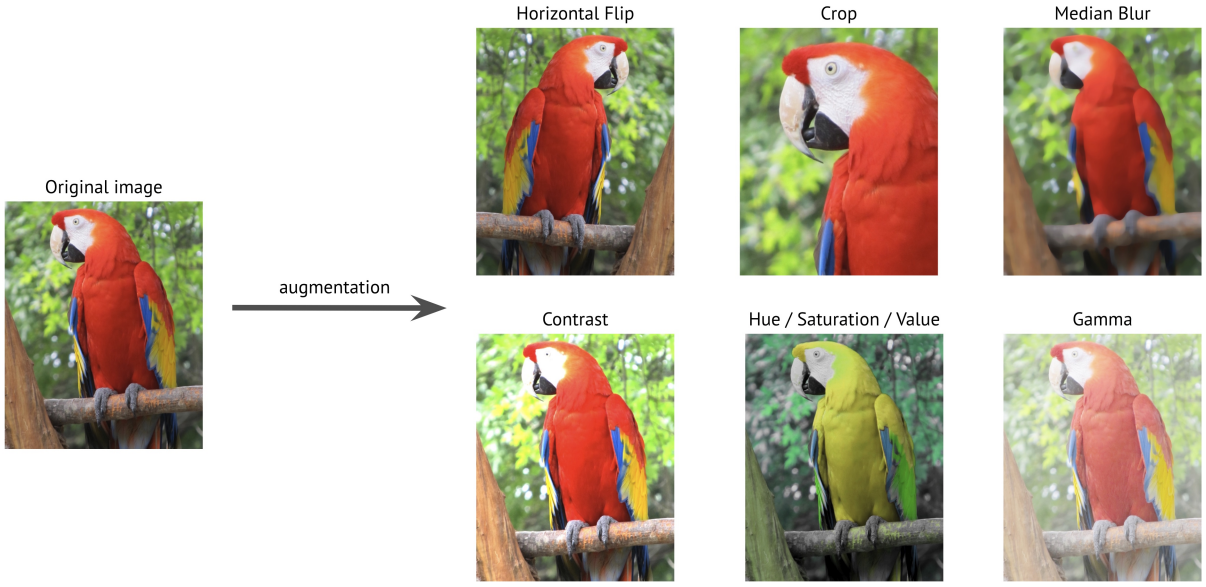


Figura 20: Manipulação da imagem original em *data augmentation*
Fonte: ALBUMENTATIONS... (2023)

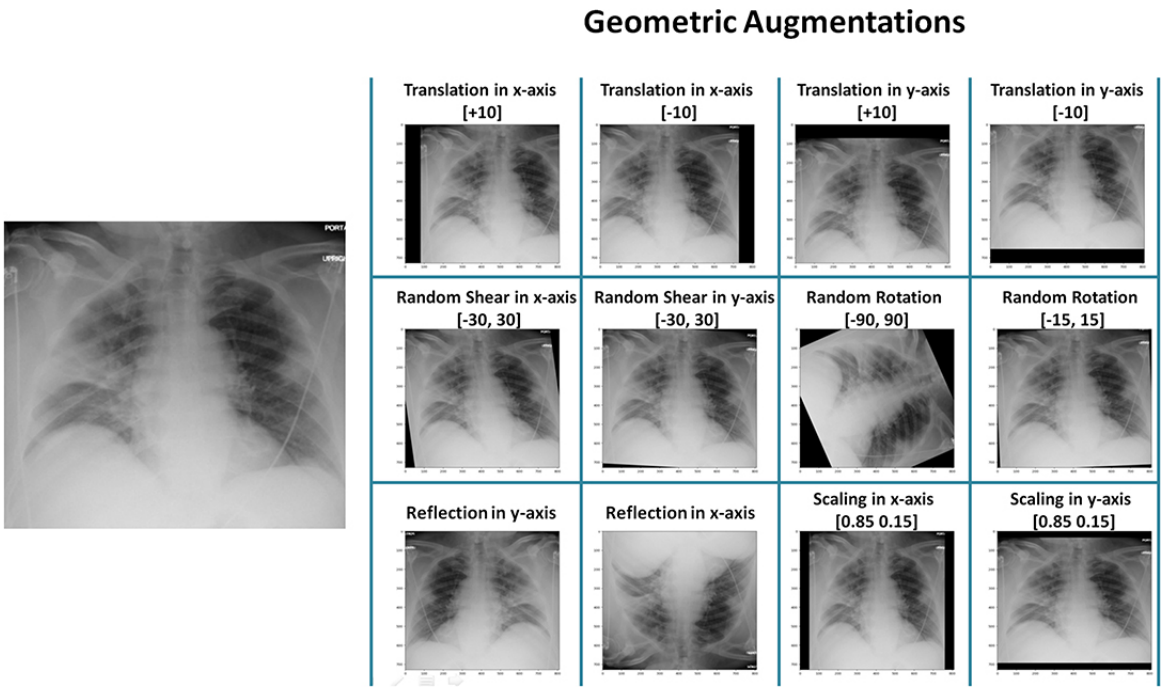


Figura 21: Translação e rotação da imagem original
Fontes: Elgendi *et al.* (2021)

Outras técnicas utilizadas aplicam filtros nas imagens originais ou combinam diferentes classes em uma imagem. Técnicas mais avançadas de *data augmentation* empregam modelos de ML para a síntese de novas imagens. Um dos modelos amplamente utilizados para essa finalidade são as Redes Adversárias Generativas (GANs), devido aos seus excelentes resultados e eficiência computacional (SHORTEN; KHOSHGOFTAAR, 2019). A

Figura 22 apresenta alguns exemplos de imagens sintéticas geradas por uma GAN.

Figura 22: Dados criados sinteticamente utilizando CycleGANs
Fontes: Shorten e Khoshgoftaar (2019)

3 MATERIAIS E MÉTODOS

Para o desenvolvimento do presente trabalho de TCC foi realizado primeiramente o embasamento teórico de aprendizado de máquina e de perfilagem de poços. Nesse sentido, inicialmente, foi feita a reunião de artigos e de livros que abordavam os temas de aprendizado de máquina e de perfilagem de poços. Posteriormente, foram levantadas informações necessárias para a revisão bibliográfica apresentada no Capítulo 2 e para conhecimentos usados no desenvolvimento e no treinamento dos modelos de ML desenvolvidos, conforme fluxograma do trabalho apresentado na Figura 23.

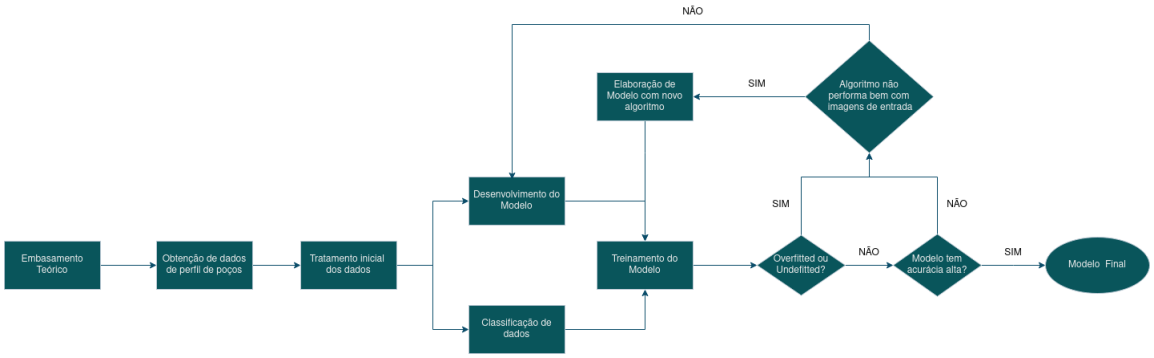


Figura 23: Fluxograma de desenvolvimento do trabalho.
Fonte: Elaboração Própria.

Dados do poço 7-BUZ-12-RJS localizado no campo de Búzios foram solicitados à ANP (processo SEI nº 48610.233579/2023-38) para serem utilizados como informações de entrada dos algoritmos desenvolvidos durante o TCC. O campo de Búzios fica na Bacia de Santos, e tem como operadora a Petrobrás. Os perfis de poço a serem utilizados nesse trabalho foram coletados em novembro de 2016 e já são públicos.

3.1 Mineração de dados

Com os arquivos brutos do poço 7-BUZ-12-RJS, foi realizado o processamento dos dados do perfil de imagem acústica. O processamento desse perfil compreende uma série

de etapas cruciais. Inicialmente, destaca-se a correção de velocidade, na qual o perfil é ajustado em profundidade para corrigir possíveis distorções causadas pela pressão do cabo. Em seguida, ocorre o alinhamento em relação ao norte magnético, seguido pela correção de excentricidade, que visa corrigir descentralizações potenciais da ferramenta durante a aquisição da imagem. A etapa subsequente é a harmonização da imagem, seguida pela normalização dos dados e distribuição de cores na imagem. Esses procedimentos combinados são fundamentais para obter resultados precisos na análise do perfil de imagem acústica do poço.

Após a conclusão desse processamento nos dados do perfil, o próximo passo envolveu a exportação do arquivo resultante para o formato PDF. Essa escolha foi motivada pela facilidade de manipulação do arquivo e pela capacidade de recortar a imagem original em imagens menores, simplificando assim seu manuseio e sua classificação.

O tratamento inicial dos dados foi feito para obter um conjunto de imagens que foram utilizadas como dados de entrada para o modelo. O primeiro passo foi subdividir o perfil de imagem acústica estático em arquivos de imagens menores que foram utilizadas no treinamento do modelo. O arquivo pdf original do perfil continha 6 páginas de 394×5000 mm (largura × altura). Ele foi então convertido para 6 arquivos PNG com dimensões de 3100×39370 px. Os novos arquivos de imagem foram subdivididos em arquivos menores de 678×200 px. Além disso, foi adotado um *step* que permitia que as imagens fossem sobrepostas verticalmente em 50%. Com as imagens de entrada prontas foi feita a sua classificação, em imagens com coleta de amostra e imagens sem coleta de amostra.

3.2 Desenvolvimento dos modelos

3.2.1 Modelos de Segmentação de Imagens

Modelos de aprendizado de máquina supervisionado foram desenvolvidos para a classificação das imagens. Nos primeiros testes, foram utilizados o modelo de SVM, mas foram substituídos por modelos de segmentação de imagens, que poderiam localizar onde houve a coleta da amostra na imagem de entrada. Os dados de entrada precisaram ser adaptados, foram utilizadas as imagens originais do perfil de imagem acústica estático e perfil de tempo de trânsito. Essas imagens também foram modificadas utilizando filtros para obter novos aspectos da imagem original, listados na Tabela 1. Além disso, foi necessário criar máscaras que foram utilizadas como *label* da imagem original, para validar se o pixel da imagem era positivo para a coleta de amostra ou não.

Tabela 1: Filtros utilizados na imagens de entrada do modelo de Segmentação de imagem

Filtro
Gabor
Canny Edge
Roberts
Sobel
Scharr
Prewitt
Farid
Gaussano $\sigma = 3$

Fonte: Elaboração Própria

Os filtros Gabor foram aplicados com os parâmetros do seu *kernel* variados, foram feitas iterações nos domínios listados na Tabela 2 para obter trinta e dois (32) *kernels* diferentes.

Tabela 2: Parâmetros utilizados nos Filtros Gabor

Parâmetro Filtro Gabor	Domínio
θ (theta)	$[0, \frac{\pi}{4}]$
λ (lambda)	$[0, \frac{\pi}{4}, \frac{2\pi}{4}, \frac{3\pi}{4}]$
σ (sigma)	[1.0, 3.0]
γ (gamma)	[0.05, 0.5]
ψ (psi)	0

Fonte: Elaboração Própria

3.2.1.1 Modelo RF

Após alguns testes de classificação, foi feito o *tuning* (otimização ou ajuste de hiperparâmetros) do modelo, utilizando *Grid Search* (busca em grade) para os hiperparâmetros de *n_estimators*, *min_samples_splits*, *min_samples_leaf*, *max_features*, *max_depth* e *bootstrap*. O modelo após o *tuning* utilizou os hiperparâmetros listados na Tabela 3.

Tabela 3: Hiperparâmetros do modelo *Random Forest*

Hiperparâmetro	Valor após *tuning*
n_estimators	600
min_samples_split	5
min_samples_leaf	1
max_features	None
max_depth	True

Fonte: Elaboração Própria

3.2.1.2 Modelo SVM

Com os mesmos dados de entrada foi feito o treinamento e o *tuning* de um modelo de SVM. O *tuning* foi feito utilizando *Grid Search* para os hiperparâmetros *C*, *gamma* e *kernel*. O modelo após o *tuning* utilizou os hiperparâmetros listados na Tabela 4.

Tabela 4: Hiperparâmetros do modelo SVM

Hiperparâmetro	Valor após *tuning*
C	0.1
gamma	1
kernel	rbf

Fonte: Elaboração Própria

3.2.2 Modelo para Classificação e Localização

Após os modelos de segmentação de imagens foi desenvolvido um modelo de rede neural convolucional para ser feita a classificação e localização da imagem. Para isso foi necessário alterar os dados de entrada, e criar *bounding boxes* (caixa delimitadora) que foram utilizadas para fazer a localização das amostras nas imagens. A *bounding box* utilizada é compostas de dois pontos p_1 (x_1, y_1) e p_2 (x_2, y_2) que descrevem o ponto superior esquerdo e inferior direito respectivamente da *bounding box* na imagem.

Para imagens onde não há coleta de amostra, foram adotados p_1 (0, 0) e p_2 (0, 0). Originalmente, o banco de dados continha 2142 imagens de treinamento, com 106 imagens que apresentavam amostras. Foi feita então a augmentação do dados com amostras, aplicando inversão horizontal, vertical e aumentando o contraste de cores das imagens. A augmentação adicionou 424 novas imagens com amostras ao *dataset*, resultando em 2566

imagens, das quais 530 eram imagens do perfil onde houve a coleta de amostra.

A porcentagem usada na separação dos conjuntos de treinamento e de teste foi de 90%/10%, ou seja, 10% do *dataset* foi utilizado como dados de teste, enquanto o restante seria utilizado para treinamento. Essa divisão resultou em 2309 imagens utilizadas para o treinamento do modelo e 257 imagens para teste.

O modelo desenvolvido contém 5 camadas convolucionais, 5 camadas de *pooling* máximo, duas camadas densas e uma camada *flatten* (achatada), e duas cabeças de saída, uma sendo para os dados da *bounding box*, e outra para os dados da classificação da imagem. As duas contêm 4 camadas densas, 2 camadas *dropout* e uma camada de *output*. Os hiperparâmetros utilizados no modelo foram listados na Tabela 5, 6 e 7:

Tabela 5: Hiperparâmetros do modelo CNN

Camada	Número Filtros	Número Neurônios	Dimensão *Kernel*	Função Ativação	*Padding*
conv2d	64	-	7x7	ReLU	*zero padding*
max_pooling2d	-	-	2x2	-	-
conv2d_1	128	-	3x3	ReLU	*zero padding*
max_pooling2d_1	-	-	2x2	-	-
conv2d_2	256	-	3x3	ReLU	*zero padding*
max_pooling2d_2	-	-	2x2	-	-
conv2d_3	512	-	3x3	ReLU	*zero padding*
max_pooling2d_3	-	-	2x2	-	-
conv2d_4	1024	-	3x3	ReLU	*zero padding*
max_pooling2d_4	-	-	2x2	-	-
flatten	-	-	-	-	
dense	-	1024	-	ReLU	-
dense_1	-	512	-	ReLU	-

Fonte: Elaboração Própria

Tabela 6: Hiperparâmetros do modelo CNN para a saída da *bounding box*

Camada	Número Neurônios	Taxa *Dropout*	Função Ativação
dense_6	256	-	ReLU
dense_7	128	-	ReLU
dropout_2	-	20%	-
dense_8	64	-	ReLU
dropout_3	-	20%	-
dense_9	32	-	ReLU
box_output	4	-	Sigmoid

Fonte: Elaboração Própria

Tabela 7: Hiperparâmetros do modelo CNN para a saída da classe

Camada	Número Neurônios	Taxa *Dropout*	Função Ativação
dense_2	256	-	ReLU
dense_3	128	-	ReLU
dropout	-	10%	-
dense_4	64	-	ReLU
dropout_1	-	20%	-
dense_5	32	-	ReLU
class_output	2	-	Softmax

Fonte: Elaboração Própria

Para o treinamento do modelo foi utilizado o otimizador *Stochastic Gradient Descent* (Gradiente Descendente Estocástico) com momento de 0.5 e *learning rate* (taxa de aprendizado) de 10^{-4}. As funções de *loss* aplicadas aos *outputs* foram erro quadrático médio (MSE) e *Cross-Entropy* (Entropia Cruzada) para as cabeças da *bounding box* e da classe respectivamente. O peso das funções de *loss* foram iguais a 1 e as métricas utilizada durante o treinamento do modelo foram a acurácia para o *output* da classe, e o MSE para o *output* da *bounding box*.

O modelo foi programado para treinar por 150 épocas com *batch size* de 50. Além disso, foram aplicadas funções de parada caso o modelo não evoluísse a função de *loss* para os dados de validação.

4 RESULTADOS

4.1 Banco de Dados

Conforme discorrido no item 3.1 o arquivo PDF do perfil acústico de imagem foram tratadas para serem utilizadas no banco de dados dos modelos. Esse processo é ilustrado na Figura 24 e a Figura 25 apresenta um exemplo de dado de entrada usado nos modelos de ML.

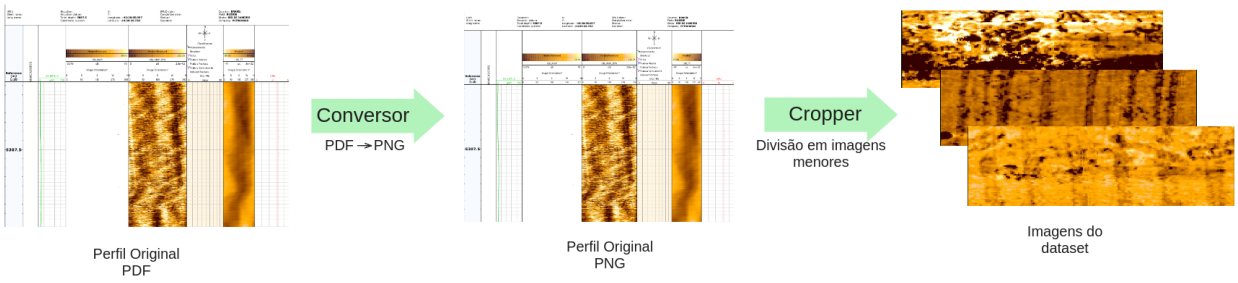


Figura 24: Tratamento feito no arquivo PDF do perfil de imagem acústica do poço.
Fonte: Elaboração Própria.

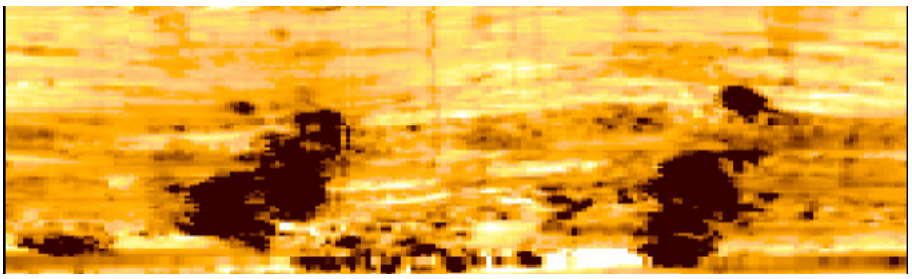
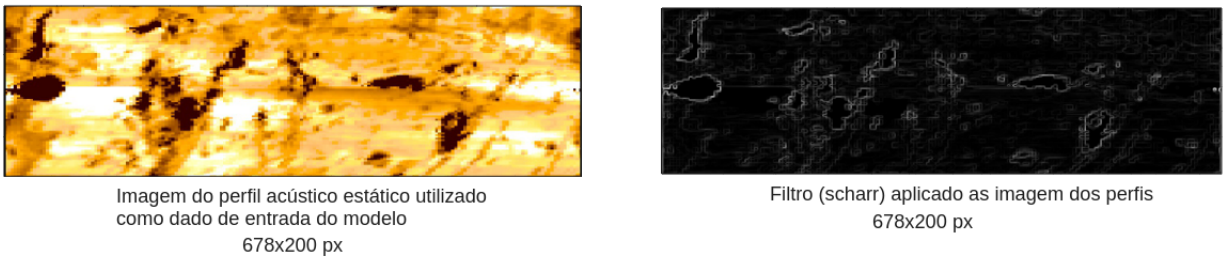


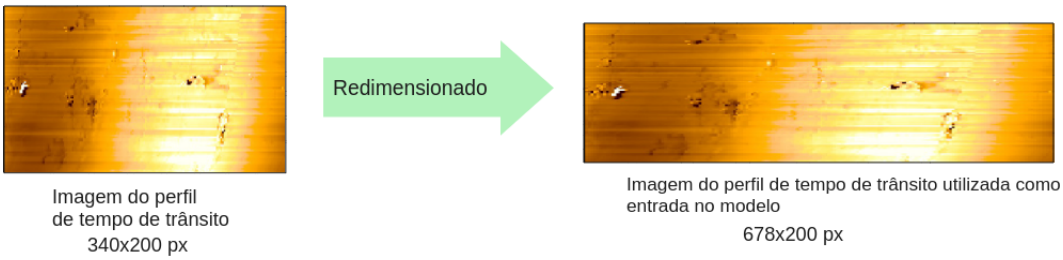
Figura 25: Exemplo de uma imagem do *dataset* usado nos modelos.
Fonte: Elaboração Própria.

Na Figura 26 são apresentados os dados utilizados no modelo de segmentação. A Figura 26a apresenta uma imagem do perfil de imagem acústica estático utilizadas como entrada e um exemplo de filtro aplicado às imagens originais do poço para extração de dados. Para o perfil de tempo de trânsito foi necessário fazer o redimensionamento da

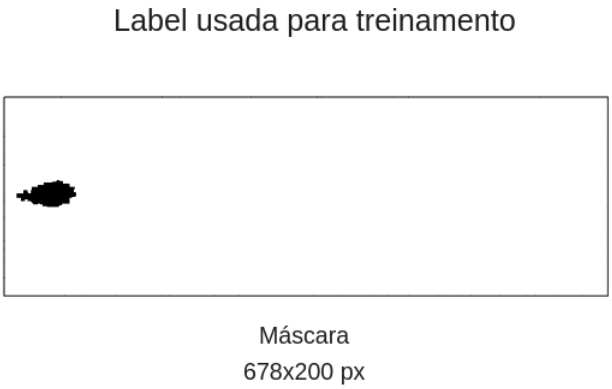
imagem original para ter mesma dimensão que as imagens do perfil acústico estático (vide Figura 26b). A máscara utilizada no modelo para validar os dados de saída do modelo durante o treinamento foram criadas a partir do perfil de imagem acústica estático. Na Figura 26c é apresentado um exemplo de máscara utilizado no trabalho.



(a)



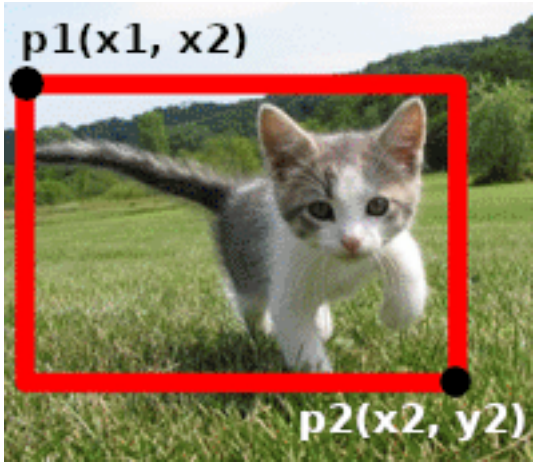
(b)



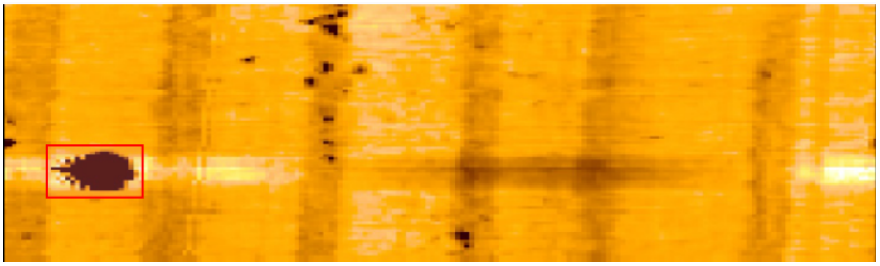
(c)

Figura 26: Dados utilizados no modelo de segmentação. (a) Imagens do perfil de imagem acústica estático utilizadas como entrada e exemplo de filtro aplicado às imagens originais do poço para extração de dados; (b) Imagens do perfil acústico de tempo de trânsito que foram redimensionadas; (c) máscara.
Fonte: Elaboração Própria

Na Figura 27 é apresentado como foi feita a *bounding box* para os dados a serem utilizados neste TCC. A Figura 27b apresenta um exemplo de uma imagem do *dataset* com sua respectiva *bounding box*.



(a)



(b)

Figura 27: *Bounding Box* e sua aplicação nas imagens do banco de dados.
(a) Pontos da *bounding box*.
Adaptado de Olivatto (2021);
(b) Exemplo de uma imagem do banco de dados.
Fonte: Elaboração Própria

4.2 Treinamento dos algoritmos de Segmentação

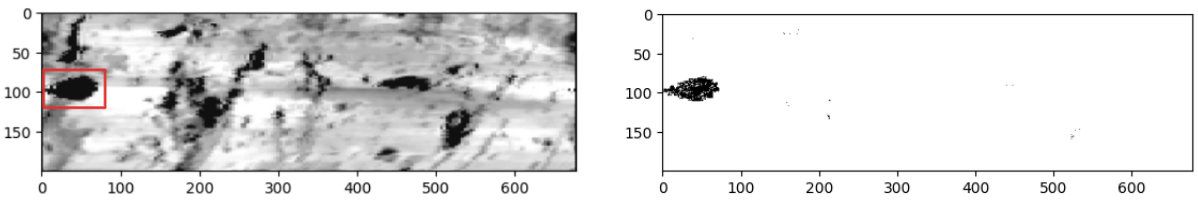
4.2.1 Modelo RF

Os dois modelos utilizados para a segmentação de imagens foram SVM e RF. Os dados de entrada do novo modelo foram organizados em uma matriz, na qual cada coluna continha os dados de uma das imagens de entradas, achatadas em um vetor. As imagens contidas na tabela eram a imagem do perfil de imagem acústica estático, e do perfil de trânsito, e as suas variantes criadas através dos filtros listados na Tabela 1 do item 3.1. Ambas as imagens originais do perfil eram primeiramente convertidas em uma escala de cor cinza.

O valor utilizado para fazer a validação dos resultados do modelo foi uma máscara da imagem original do perfil de imagem acústica estático. A máscara apresenta valor 0 para

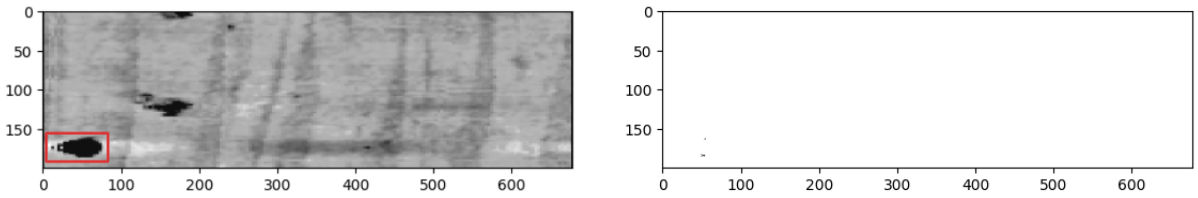
a região onde houve coleta de amostra, e valor 1 para as regiões onde não houve a coleta de amostra.

Nos primeiros testes o modelo não conseguiu segmentar imagens novas, foi então feito o aumento dos dados de treinamento e *tunning* do modelo. Os hiperparâmetros do modelo final de RF foram listados na Tabela 3 do item 3.2. Após o treinamento, o modelo ainda não conseguia fazer a segmentação de novas imagens e estava *overfitted*, pois ele conseguiu classificar muito bem as imagens originais de treinamento, mas não conseguia extrapolar para imagens que não havia entrado em contato antes. Nas Figura 28 e 29 se apresentam as imagens utilizadas como dado de entrada para o modelo e o resultado da previsão do modelo. Na Figura 28a é apresentada a imagem original que foi utilizada como dado de entrada do modelo em conjunto com suas variantes e seção do perfil de tempo de trânsito, e na sub figura 28b é apresentado o *output* do modelo. Na Figura 28 o *input* foi de uma imagem utilizada no treinamento do modelo, que foi bem segmentada, sendo de fácil identificação onde houve a coleta da amostra nessa seção do perfil, e na Figura 29 o *input* (Figura 29a) foi de uma nova imagem, que não foi segmentada de maneira a identificar onde na imagem houve a coleta da amostra (Figura 29b), delimitada em vermelho na imagem.



(a) (b)

Figura 28: Exemplos de imagens usadas no treinamento do modelo (a) imagem original - imagem usada no treinamento com amostra destacada em vermelho; (b) imagem segmentada - retorno do algoritmo.
Fonte: Elaboração Própria.



(a) (b)

Figura 29: Exemplo de imagem utilizada para a seleção do modelo (a) imagem original - imagem nova para o modelo com amostra destacada em vermelho; (b) imagem segmentada - retorno do algoritmo.
Fonte: Elaboração Própria.

4.2.2 Modelo SVM

O modelo SVM utilizava mesma organização dos dados de entrada e também apresentou resultados parecidos, mesmo após tunning descrito na Tabela 4 do item 3.2, o modelo se apresentava *overfitted* segmentando as imagens de treinamento, mas não conseguindo segmentar novas imagens.

4.3 Treinamento do algoritmo de Classificação e Localização

Devido aos resultados apresentados nos itens anteriores, foi necessário realizar uma mudança no algoritmo de classificação, optando por um algoritmo de rede convolucional, que faria a localização das amostras, e não apenas a sua classificação. O treinamento foi feito com o *dataset* completo e com a adição dos dados da *bounding box* em cada imagem que continha um artefato de coleta de amostra. O esquema do modelo com suas camadas é ilustrado na Figura 30.

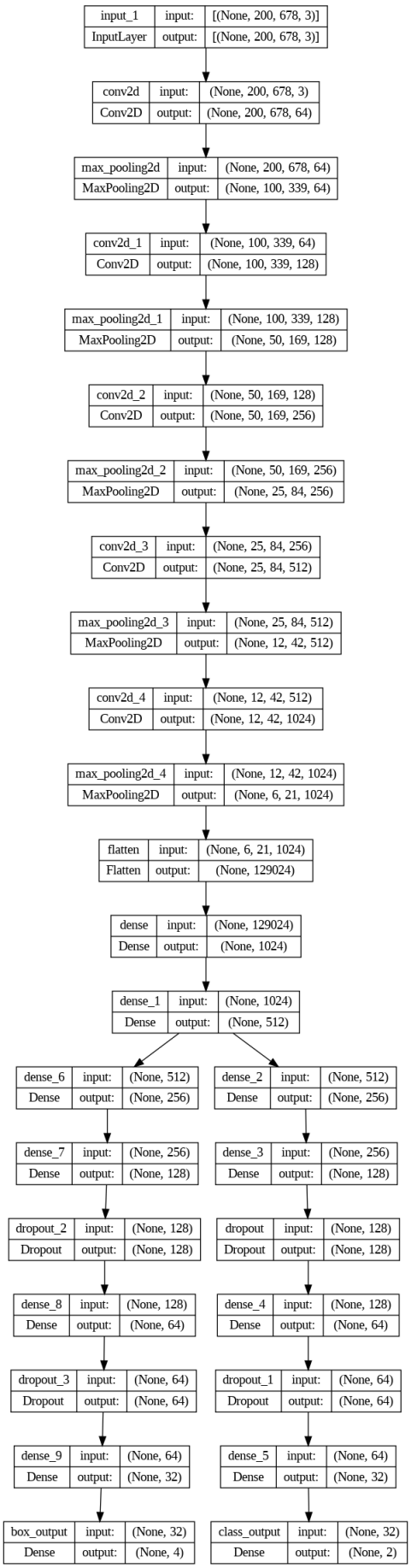


Figura 30: Diagrama do modelo implementado.
Fonte: Elaboração Própria

O modelo foi programado para treinar por 150 épocas, mas devido a função de parada aplicada no treinamento o modelo foi treinado por 131 épocas. Os resultados das medidas das funções de *loss*, acurácia e MSE do modelo foram plotados em um gráfico durante as épocas de treinamento e são apresentados na Figura 31. O modelo final apresentou classificação indevida de imagens, pois onde houve a coleta de amostras as classificando como se não existisse coleta de amostra na imagem. Na Figura 31a é possível observar um problema no modelo na qual a acurácia dos valores de saída da classe não melhoram com o passar das épocas, podendo ser um sintoma de *overfitting* do modelo. Outro fator são os valores de *loss* altos que o modelo apresentou nos gráficos das Figuras 31b 31c e 31d.

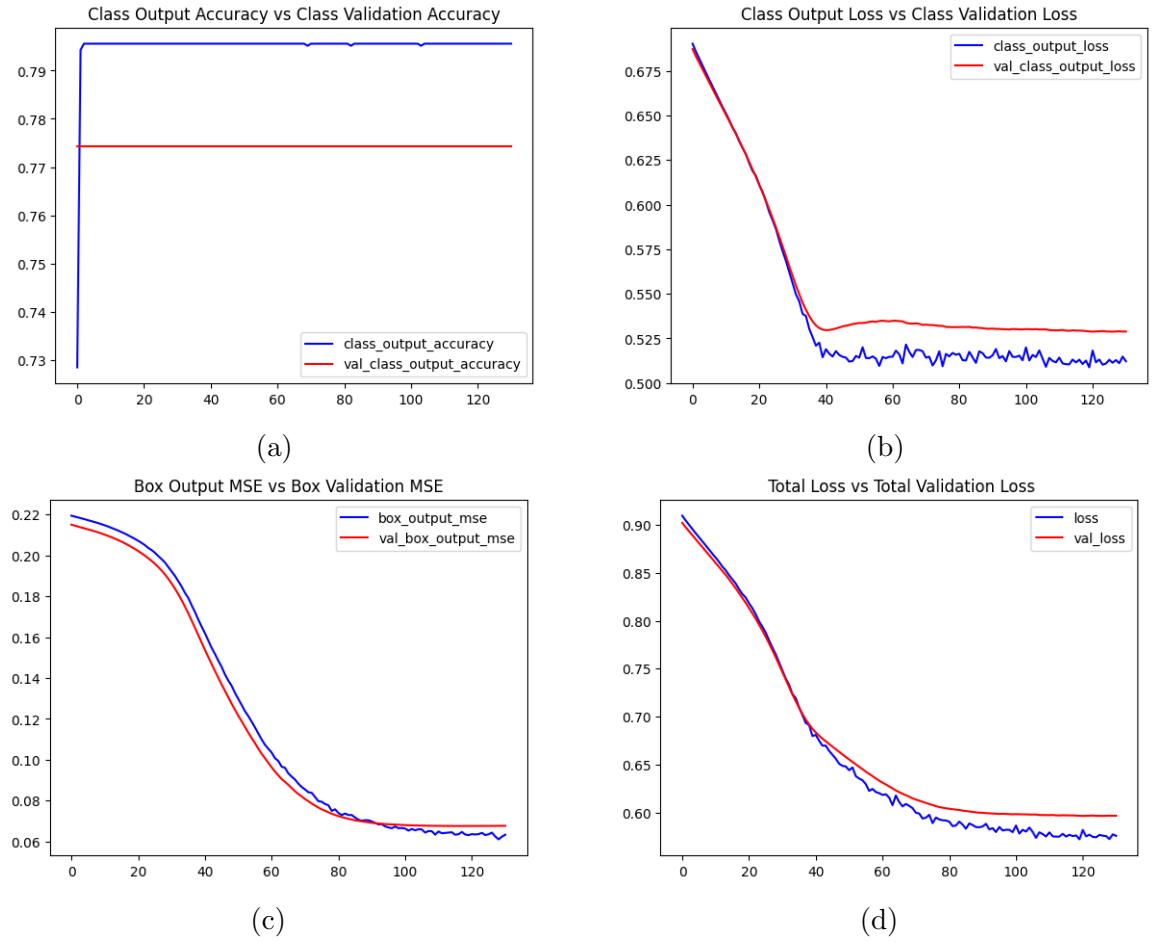
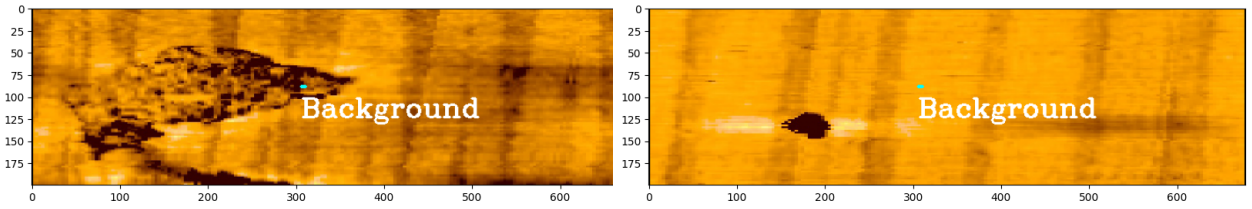


Figura 31: Gráficos de acurácia e *loss* do modelo ao longo das épocas de treinamento
Fonte: Elaboração Própria.

Também, a bounding box não conseguiu identificar a amostra, retornando valores muito próximos para os pontos p_1 e p_2. Devido a esses resultados em modelos de teste, foram alterados alguns hiperparâmetros e arquitetura da CNN, mas os modelos continuavam com o mesmo comportamento. Outro fator ressaltado durante as pesquisas foi

o desbalanceamento das classes no banco de dados, que caracteriza o *dataset* montado nesse TCC. Algumas soluções eram *data augmentation* ou a aplicação de peso para as classes do modelo a qual ajudariam o modelo a dar maior importância a uma das classes do banco de dados. A primeira solução foi aplicada, entretanto, a aplicação de peso nas classes não foi possível implementar devido a limitações técnicas na biblioteca utilizada para o desenvolvimento do modelo.

A Figura 32 apresenta o resultado da classificação de imagens, uma que não contém amostra (Figura 32a) e outra que contém amostra (Figura 32b). Como é possível notar na imagem, ambas têm a mesma classificação com valores muito próximos da *bounding box*.



(a) (b)

Figura 32: (a) Exemplo de *output* do modelo para uma imagem sem amostra; (b) Exemplo de *output* do modelo para uma imagem com amostra.
Fonte: Elaboração Própria.

Algumas das possíveis causas para esse comportamento são:

- Modelo muito complexo, com elevado número de hiperparâmetros, o que facilitaria o seu *overfitting*.
- Conjunto de dados de treinamento e de validação são destoantes, pertencendo a domínios de hipóteses diferentes que não seriam generalizáveis.
- Alta taxa de aprendizado para o treinamento do modelo.

5 CONCLUSÃO

Ao avaliar os objetivos iniciais do trabalho, foi possível atingir os pontos de obtenção de dados de perfilagem e de classificação do perfil de imagem acústica ao criar os bancos de dados utilizados nos modelos. A implementação de algoritmos de ML foi concluída, com o pré-processamento dos dados para dar entrada no modelo, entretanto, os modelos desenvolvidos não foram capazes de fazer a classificação correta de regiões onde houve a coleta de amostras laterais no poço. Sua acurácia se manteve constante desde as primeiras épocas de treinamento, caracterizando um possível *overfitting* do modelo e além disso, as funções de *loss* das saídas e do modelo se estabilizaram em um valor alto.

Algumas possíveis causas levantadas foram o elevado número de parâmetros treináveis do modelo, a divergência entre os dados de treinamento e validação e uma elevada taxa de aprendizado empregada no treinamento do modelo.

Apesar dos desafios enfrentados durante a execução deste TCC, é essencial ressaltar a importância crucial do aprendizado de máquina na classificação de imagens, em especial no contexto da perfilagem de poços. Embora a obtenção de resultados satisfatórios tenha demandado uma busca por instâncias adicionais de modelos, a viabilidade do ML como uma ferramenta valiosa para a classificação na área de perfilagem de poços permanece evidente.

A flexibilidade e adaptabilidade dos algoritmos de ML oferecem uma abordagem promissora para lidar com a complexidade das imagens obtidas durante a perfilagem de poços. A habilidade destes modelos em assimilar padrões intrínsecos e extrair informações pertinentes das imagens não apenas contribui para aprimorar a eficiência, mas também para otimizar as análises na indústria do petróleo, promovendo uma abordagem mais ágil e refinada nesta área de estudo.

Embora o presente trabalho tenha enfrentado desafios, a aplicação de ML na classificação de perfis de imagem continua a representar uma via promissora. Ainda é um método viável, capaz de proporcionar *insights* valiosos para a interpretação geológica e

avaliação de reservatórios. O aprimoramento constante dos modelos, com o acréscimo de instâncias e ajustes necessários, abre perspectivas animadoras para a contínua evolução e aplicação bem-sucedida dessas técnicas no campo da perfilagem de poços e de ML como uma ferramenta robusta e promissora na análise e classificação de imagens na indústria do petróleo.

6 APRENDIZADOS E PROPOSTAS PARA CONTINUAÇÃO DO TRABALHO

6.1 Aprendizados e Desafios na Realização do Trabalho

Durante a execução deste estudo, uma série de aprendizados e desafios foram enfrentados, resultando em um desenvolvimento tanto no âmbito acadêmico quanto profissional. No que tange aos aprendizados, destaca-se a assimilação dos fundamentos do aprendizado de máquina, proporcionando uma compreensão das técnicas e abordagens aplicadas na área. Adicionalmente, a implementação prática de algoritmos de aprendizado de máquina traduziu os conhecimentos teóricos em aplicações concretas, consolidando a proficiência na manipulação de ferramentas e frameworks utilizados nessa área.

Por outro lado, os desafios inerentes à complexidade do projeto incluíram a montagem do banco de dados e a definição de estratégias para a rotulação das imagens, requerindo planejamento e busca por diferentes soluções e aplicações que auxiliem nessa tarefa. A organização do código revelou-se desafiadora, exigindo que fossem feitas diversas refatorações para manter a clareza e sua funcionalidade. Por fim, o treinamento do modelo e a otimização constante dos hiperparâmetros foram etapas com adversidades, demandando uma abordagem iterativa e aprofundada na tentativa de alcançar resultados satisfatórios. Esses desafios, embora exigentes, proporcionaram importantes aprendizados e experiências na implementação e manutenção de algoritmos de ML.

6.2 Propostas para Continuação do Trabalho

Em trabalhos futuros, seria relevante explorar as seguintes áreas para aprimorar a capacidade de classificação do modelo:

- Aplicar pesos às classes, seja por meio da adaptação da função de *loss* ou pela

aplicação de uma ferramenta alternativa capaz de incorporar esse parâmetro. Essa estratégia tem o intuito de aprimorar a sensibilidade do modelo em relação a diferentes classes, promovendo uma classificação mais equilibrada.

- Refinar a arquitetura proposta neste Trabalho de Conclusão de Curso (TCC), especialmente otimizando as cabeças de saída do modelo. Adicionar camadas que mitiguem o *overfitting*, como camadas de *dropout*, e realizar o ajuste dos hiperparâmetros pode contribuir para a performance do modelo.

- Explorar a aplicação de modelos pré-treinados, uma estratégia que pode beneficiar a classificação de imagens e reduzir o tempo de treinamento das saídas relacionadas à *bounding box* e classe. Essa abordagem pode aproveitar o conhecimento adquirido por modelos pré-existentes, adaptando-o ao contexto específico da perfilagem de poços.

REFERÊNCIAS

"SCHLUMBERGER". [*S. l.: s. n.*]. Disponível em: ⟨https://www.slb.com⟩. Acesso em: 2022.

AL-ABDULJABBAR, A.; MAHMOUD, A. A.; ELKATATNY, S.; ABUGHABAN, M. Artificial neural networks-based correlation for evaluating the rate of penetration in a vertical carbonate formation for an entire oil field. en. **Journal of Petroleum Science and Engineering**, v. 208, p. 109693, jan. 2022. ISSN 09204105. DOI: ⟨10.1016/j.petrol.2021.109693⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S0920410521013218⟩. Acesso em: 14 nov. 2023.

ALBUMENTATIONS Documentation. [*S. l.: s. n.*]. Disponível em: ⟨https://albumentations.ai/docs/⟩. Acesso em: 2023.

ALPAYDIN, E. **Introduction to machine learning**. 2nd ed. Cambridge, Mass: MIT Press, 2010. 537 p. (Adaptive computation and machine learning). OCLC: ocn317698631. ISBN 978-0-262-01243-0.

ASSAIFE, R. F. Perfilagem durante a perfuração de poços de petróleo: estudo bibliográfico. **Revista Científica Multidisciplinar Núcleo do Conhecimento**, p. 51–80, 31 mar. 2022. ISSN 24480959. DOI: ⟨10.32749/nucleodoconhecimento.com.br/engenharia-mecanica/perfuracao-de-pocos⟩. Disponível em: ⟨https://www.nucleodoconhecimento.com.br/engenharia-mecanica/perfuracao-de-pocos⟩. Acesso em: 21 jul. 2022.

BISHOP, C. M. **Pattern recognition and machine learning**. New York: Springer, 2006. 738 p. (Information science and statistics). ISBN 978-0-387-31073-2.

DATASCIENCE Foundation. [*S. l.: s. n.*]. Disponível em: ⟨https://datasciencefoundation.org/⟩. Acesso em: 2022.

DIERSEN, S.; LEE, E.-J.; SPEARS, D.; CHEN, P.; WANG, L. Classification of Seismic Windows Using Artificial Neural Networks. **Procedia Computer Science**, v. 4, p. 1572–1581, 2011. ISSN 18770509. DOI: ⟨10.1016/j.procs.2011.04.170⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S1877050911002286⟩. Acesso em: 10 mai. 2022.

ELGENDI, M.; NASIR, M. U.; TANG, Q.; SMITH, D.; GRENIER, J.-P.; BATTE, C.; SPIELER, B.; LESLIE, W. D.; MENON, C.; FLETCHER, R. R.; HOWARD, N.; WARD, R.; PARKER, W.; NICOLAOU, S. The Effectiveness of Image Augmentation in Deep Learning Networks for Detecting COVID-19: A Geometric Transformation Perspective. **Frontiers in Medicine**, v. 8, p. 629134, mar. 2021. ISSN 2296-858X. DOI: ⟨10.3389/fmed.2021.629134⟩. Disponível em: ⟨https://www.frontiersin.org/articles/10.3389/fmed.2021.629134/full⟩. Acesso em: 7 nov. 2023.

FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. **Inteligência artificial: uma abordagem de aprendizado de máquina**. Rio de Janeiro: Grupo Gen - LTC, 2011. OCLC: 854567299. ISBN 978-85-216-1880-5 978-85-216-2015-0. Disponível em: ⟨http://site.ebrary.com/id/10707120⟩. Acesso em: 31 mai. 2022.

GÉRON, A. **Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems**. Second edition. Beijing [China] ; Sebastopol, CA: O'Reilly Media, Inc, 2019. 819 p. ISBN 978-1-4920-3264-9.

JIANG, M.; WANG, J.; HU, L.; HE, Z. Random forest clustering for discrete sequences. en. **Pattern Recognition Letters**, s0167865523002507, set. 2023. ISSN 01678655. DOI: ⟨10.1016/j.patrec.2023.09.001⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S0167865523002507⟩. Acesso em: 14 set. 2023.

KHAN, M. Y.; QAYOOM, A.; NIZAMI, M. S.; SIDDIQUI, M. S.; WASI, S.; RAAZI, S. M. K.-R. Automated Prediction of Good Dictionary EXamples (GDEX): A Comprehensive Experiment with Distant Supervision, Machine Learning, and Word Embedding-Based Deep Learning Techniques. en. Edição: Shahzad Sarfraz. **Complexity**, v. 2021, p. 1–18, set. 2021. ISSN 1099-0526, 1076-2787. DOI: ⟨10.1155/2021/2553199⟩. Disponível em: ⟨https://www.hindawi.com/journals/complexity/2021/2553199/⟩. Acesso em: 14 set. 2023.

KOROTEEV, D.; TEKIC, Z. Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future. en. **Energy and AI**, v. 3, p. 100041, mar. 2021. ISSN 26665468. DOI: ⟨10.1016/j.egyai.2020.100041⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S2666546820300410⟩. Acesso em: 13 nov. 2023.

MITCHELL, T. M. **Machine learning**. Nachdr. New York: McGraw-Hill, 2013. 414 p. (McGraw-Hill series in Computer Science). ISBN 978-0-07-115467-3 978-0-07-042807-2.

OLIVATTO, T. F. **Identificação Automática De Rampas De Acessibilidade Apoiada Por Visão Computacional A Partir De Imagens Panorâmicas Street-level**. 2021. Mestrado – Universidade Federal de São Carlos, São Carlos - SP.

Disponível em: ⟨https://www.researchgate.net/publication/357132228_Identificacao_automatica_de_rampas_de_acessibilidade_apoiada_por_visao_computacional_a_partir_de_imagens_panoramicas_street-level⟩.

PARAB, M. A.; MEHENDALE, N. D. Red Blood Cell Classification Using Image Processing and CNN. en. **SN Computer Science**, v. 2, n. 2, p. 70, abr. 2021. ISSN 2662-995X, 2661-8907. DOI: ⟨10.1007/s42979-021-00458-2⟩. Disponível em: ⟨http://link.springer.com/10.1007/s42979-021-00458-2⟩. Acesso em: 8 nov. 2023.

PICCININI, G. The First Computational Theory of Mind and Brain: A Close Look at McCulloch and Pitts's "Logical Calculus of Ideas Immanent in Nervous Activity". **Synthese**, v. 141, n. 2, p. 175–215, 2004. Publisher: Springer. ISSN 00397857, 15730964. Disponível em: ⟨http://www.jstor.org/stable/20118476⟩. Acesso em: 14 nov. 2023.

QIN, Q.; HUANG, Z.; ZHOU, Z.; CHEN, C.; LIU, R. Crude oil price forecasting with machine learning and Google search data: An accuracy comparison of single-model versus multiple-model. en. **Engineering Applications of Artificial Intelligence**, v. 123, p. 106266, ago. 2023. ISSN 09521976. DOI: ⟨10.1016/j.engappai.2023.106266⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S0952197623004505⟩. Acesso em: 14 nov. 2023.

RUMELHART, D. E.; HINTON, G. E.; WILLIAMS, R. J. Learning Internal Representations by Error Propagation. *In:* PARALLEL Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations. Cambridge, MA, USA: MIT Press, 1986. P. 318–362. ISBN 026268053X.

SHEIKHOUSHAGHI, A.; GHARAEI, N. Y.; NIKOOFARD, A. Application of Rough Neural Network to forecast oil production rate of an oil field in a comparative study. en. **Journal of Petroleum Science and Engineering**, v. 209, p. 109935, fev. 2022. ISSN 09204105. DOI: ⟨10.1016/j.petrol.2021.109935⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S0920410521015503⟩. Acesso em: 14 nov. 2023.

SHORTEN, C.; KHOSHGOFTAAR, T. M. A survey on Image Data Augmentation for Deep Learning. en. **Journal of Big Data**, v. 6, n. 1, p. 60, dez. 2019. ISSN 2196-1115. DOI: ⟨10.1186/s40537-019-0197-0⟩. Disponível em: ⟨https://journalofbigdata.springeropen.com/articles/10.1186/s40537-019-0197-0⟩. Acesso em: 6 nov. 2023.

SIRCAR, A.; YADAV, K.; RAYAVARAPU, K.; BIST, N.; OZA, H. Application of machine learning and artificial intelligence in oil and gas industry. **Petroleum Research**, v. 6, n. 4, p. 379–391, 2021. ISSN 20962495. DOI: ⟨10.1016/j.ptlrs.2021.05.009⟩. Disponível em: ⟨https://linkinghub.elsevier.com/retrieve/pii/S2096249521000429⟩. Acesso em: 11 abr. 2022.

YAMASHITA, R.; NISHIO, M.; DO, R. K. G.; TOGASHI, K. Convolutional neural networks: an overview and application in radiology. en. **Insights into Imaging**, v. 9, n. 4, p. 611–629, ago. 2018. ISSN 1869-4101. DOI: ⟨10.1007/s13244-018-0639-9⟩. Disponível em: ⟨https://insightsimaging.springeropen.com/articles/10.1007/s13244-018-0639-9⟩. Acesso em: 8 nov. 2023.

ANEXO A – ARTIGO SINTESE

Universidade de São Paulo

Engenharia de Petróleo — Escola Politécnica

Número USP: 10772157 **Data: 05/01/2024**

Aplicação de *Machine Learning* na Classificação de Perfis de Imagem Acústica de Poços de Petróleo

Cecília Garcia da Silveira

Orientador: Profa. Dra. Elsa Vásquez Alvarez

Co-orientador: Geo. Érica Kato Pacheco Ferraz

Artigo Sumário referente à disciplina PMI3349 — Trabalho de Conclusão de Curso II
Este artigo foi preparado como requisito para completar o curso de Engenharia de Petróleo na Escola Politécnica da USP. Template Latex versão 2021v00.

Resumo

A indústria de petróleo é responsável pela geração de diversos tipos de dados cruciais para a administração e tomada de decisões e atualmente ela está incorporando inteligência artificial e aprendizado de máquina como ferramentas para aprimorar a tomada de decisões e a gestão. O objetivo deste Trabalho de Conclusão de Curso(TCC) é fazer a classificação de seções de perfis de imagem acústica e implementar algoritmos de aprendizado de máquina para classificar e localizar pontos de coleta de amostras nesse tipo de perfil. Inicialmente, foram empregados algoritmos de aprendizado supervisionado tradicionais, e posteriormente foi implementado modelos de Redes Convolucionais para fazer a localização das amostras no perfil. Os dados do poço utilizados foram disponibilizados pela ANP (SEI nº 48610.233579/2023-38) e ao todo foram utilizadas 2566 imagens para treinamento e validação dos modelos. Os resultados finais do modelo não convergiram para a classificação e localização das amostras, entretanto os dados classificados para treinamento foram organizados em um *dataset* (banco de dados) para ser utilizado em projetos futuros.

Palavras-Chave – Aprendizado de Máquina, Classificação de Imagens, Engenharia de Petróleo, Perfilagem de Poços, Perfil de Imagem Acústico.

Abstract

The oil industry is responsible for generating various types of crucial data for administration and decision-making, and it is currently incorporating artificial intelligence and machine learning as tools to enhance decision-making and management. The objective of this Undergraduate Thesis is to classify sections of acoustic image logs and implement machine learning algorithms to classify and locate sample collection points in this type of image log. Initially, traditional supervised learning algorithms were employed, and later Convolutional Neural Network models were implemented to locate the samples in the profile. Well data used were provided by ANP (SEI No. 48610.233579/2023-38), and a total of 2566 images were used for model training and validation. The final model results did not converge for sample classification and localization; however, the classified training data was organized into a dataset to be used in future projects.

Keywords – Machine Learning, Image Classification, Petroleum Engineering, Well Logging, Acoustic Image Log.

1 Introdução

Atualmente, mesmo com todos os avanços e investimentos em energias renováveis, o petróleo segue como uma das fontes de energia mais utilizadas no mundo. A exploração do combustível fóssil acumulado em bacias sedimentares se inicia pela identificação de áreas promissoras, aquisição de dados sísmicos até a perfuração dos primeiros poços exploratórios. A perfilagem destes poços agrega uma série de informações petrofísicas que irão alimentar o modelo geológico do reservatório. A indústria de petrolífera enfrenta diversos desafios na organização de banco de dados e no processamento das informações. A análise técnica apropriada dos dados é de fundamental importância no desempenho da indústria de hidrocarbonetos e atualmente ela ainda é feita em grande parte por especialistas, que fazem a classificação e interpretação dos dados.

A utilização de Inteligência Artificial (IA), tem se mostrado uma ferramenta importantíssima para as tomadas de decisões na indústria de óleo e gás (Koroteev & Tekic, 2021), e com o subcampo dela, o *machine learning* (aprendizado de máquina), que deverá apresentar um crescimento ainda maior nos próximos anos, contribuindo com seu valor para o desenvolvimento desta indústria, pois a suas técnicas tem o potencial de aprimorar inúmeras ações críticas realizadas por administradores e engenheiros nesse setor.

Para a aquisição de dados petrofísicos, perfis de poços de petróleo são essenciais para o desenvolvimento e melhor compreensão dos campos de produção. As ferramentas de perfilagem descidas durante ou após a perfuração dos poços apresentam diferentes propriedades para a aquisição de uma série de dados essenciais para a exploração, como propriedades elétricas, acústicas, mecânicas, radioativas, entre outras, que são registradas no perfil geofísico. Durante a coleta de dados de poços, podem ser retiradas amostras laterais que são utilizadas para identificação de dados diretos da rocha, de grande importância para a calibração de perfis do poço. Uma das dificuldades relacionadas a coleta de amostras laterais é a identificação da profundidade real de coleta em imagens acústica devido a imprecisão da ferramenta quanto à profundidade que a amostra foi coletada. A identificação dessas amostras, por intermédio de técnicas computadorizadas seria de extrema ajuda para os especialistas que fazem sua classificação, podendo reduzir consideravelmente o tempo investido na identificação dessas amostras permitindo que eles utilizem melhor o seu tempo para a interpretação dos dados das amostras ou do perfil estudado. Visando isto, o presente trabalho de Conclusão de Curso (TCC) tem como objetivo a identificação de pontos de coleta de amostras laterais no poço 7-BUZ-12-RJS utilizando aprendizado de máquina.

2 Revisão Bibliográfica

A perfilagem de poço é a prática de realizar registro detalhado das formações geológicas atravessadas durante a perfuração do poço. Os perfis geofísicos são medidas indiretas, ou seja, são uma interpretação feita com base nas propriedades físicas das rochas, que fornece dados das rochas e fluidos da formação. Já as medidas diretas, que são obtidas através de ensaios feitos em amostras da formação, através da coleta de amostras do poço, podendo ser amostras laterais ou testemunhos (Assaife, 2022).

Meio a tantos dados indiretos obtidos em uma perfilagem, as amostras laterais se destacam como dados diretos da rocha, importantíssimos para a calibração dos demais perfis petrofísicos e melhor compreensão das rochas. É inerente à aquisição dessas amostras, que haja imprecisão quanto à profundidade de retirada, que se dá pelo esticamento do cabo e movimentação do guincho que sustenta a ferramenta. Dessa forma, para a melhor correlação dessas imagens com os demais perfis e posicionamento preciso na coluna geológica, utiliza-se a identificação visual das amostras retiradas nos perfis de imagem acústica adquiridos após a coleta.

2.1 *Machine Learning* na classificação de imagens

Uma das principais distinções entre os tipos de algoritmos de aprendizado de máquina é a forma que o algoritmo aprende e treina. Os tipos de aprendizados mais comuns são o aprendizado supervisionado, e o

não supervisionado.

O aprendizado supervisionado é utilizado por alguns algoritmos, e consiste na utilização dos resultados do conjunto de dados para ajustar a hipótese conforme ocorre o treinamento e esse tipo de aprendizado é utilizado para resolver tarefas de previsão onde é encontrada uma hipótese ou função a partir dos dados de treinamento para ser utilizado com conjuntos de novos dados para prever uma resposta. Esses algoritmos caracterizam modelos preditivos (Faceli et al., 2011).

Para tarefas de descrição, o objetivo do algoritmo é explorar e identificar formas de representar um conjunto de dados. Por tanto, algoritmos utilizados para esse tipo de tarefa não fazem a comparação entre os dados de saída e do rótulo dos dados de treino, caracterizando o aprendizado não supervisionado (Faceli et al., 2011).

Em tarefas de classificação, como as de imagens, são utilizados algoritmos de classificação, cujo objetivo é a partir de um conjunto de dados de entrada definir um resultado de um conjunto de respostas com valor discreto (Bishop, 2006). Após a fase de treinamento o algoritmo pode determinar uma regra de descriminação entre as classes. Essa regra é uma função que separa os dados de treinamento para as diferentes classes de classificação, e se essa regra conseguir ser abstraída ela pode fazer a classificação de novos dados que o algoritmo ainda não teve contato (Alpaydin, 2010).

Adicionalmente às tarefas de classificação de imagens, é possível fazer também a localização e detecção de objetos. Nesses tipos de algoritmo, além do resultado da classificação da imagem é identificando a distribuição espacial da classe. Na segmentação de image é feito o particionamento de uma imagem em múltiplos segmentos (Géron, 2019) e na localização e detecção de objetos é feita a identificação de um objeto de interesse utilizando uma caixa delimitadora. A Figura 1 ilustra estes diferentes tipos de algoritmos utilizados na classificação de imagens.

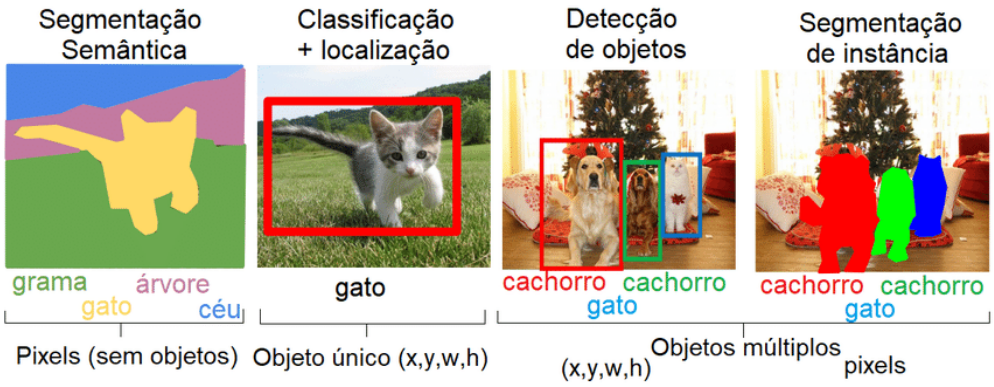


Figura 1 – Exemplos de algoritmos de classificação de imagens
Fonte : Olivatto (2021).

2.1.1 Redes Neurais

As redes neurais, fundamentadas na inspiração do funcionamento do cérebro humano, são estruturas complexas e interconectadas de unidades computacionais, conhecidas como neurônios artificiais. Cada neurônio recebe sinais de entrada, que são tratados pela sua função de ativação e retornam um sinal de saída. As funções de ativação são únicas de cada neurônio e seus *links* têm um peso próprio ditando a importância de cada sinal de entrada para a função de ativação. Os sinais de saída dos neurônios são utilizados como entrada para outros neurônios (Diersen et al., 2011).

As redes neurais são dispostas em camadas, sendo apenas uma a *Input Layer* (camada de entrada), apenas uma *Output Layer* (camada de saída) e podendo ter *n Hidden Layers* (camadas intermediárias)

que são compostas por neurônios e ligamentos com suas respectivas funções de ativação e pesos (Diersen et al., 2011).

2.1.2 Redes Neurais Convolucionais (CNN)

As redes convolucionais (CNNs) tiveram sua origem nos estudos sobre o córtex visual do cérebro humano e têm sido aplicadas no reconhecimento de imagens desde a década de 1980. Essas redes apresentam camadas distintas em relação às redes neurais tradicionais, incluindo camadas convolucionais que nomeiam a rede e camadas de *pooling* (Géron, 2019).

Nas camadas convolucionais, os neurônios não se conectam diretamente a cada pixel da imagem de entrada, mas, em vez disso, respondem a uma pequena área retangular específica. Essa arquitetura permite que as camadas iniciais da rede se foquem em detalhes de baixo nível na imagem. À medida que as convoluções ocorrem, esses detalhes se agregam para identificar características mais complexas e de alto nível nas camadas subsequentes. A identificação de aspectos é feita por filtros com dimensões pré-definidas que permitem a extração de novas informações da imagem original, transformando-as em um *feature map* (mapa de aspectos) (Géron, 2019).

Outra camada fundamental nas CNNs é a camada de *pooling* (agrupamento), a qual é empregada para reduzir as dimensões dos *feature maps* (mapas de aspectos) enquanto ainda retém as características previamente mapeadas. Nessa camada, os neurônios não possuem pesos; em vez disso, eles realizam uma agregação dos valores de entrada utilizando uma função de agregação, podendo ser a média dos valores de entrada ou, mais comumente, o valor máximo entre os elementos que serão agregados (Géron, 2019).

3 Materiais e Métodos

Para a realização do presente trabalho foi necessário solicitar à ANP (Agência Nacional do Petróleo, Gás Natural e Biocombustíveis) dados de um poço que continha perfis de imagem acústica para que fossem feitas a identificação de coletas de amostras laterais. Dos dados solicitados do poço 7-BUZ-12-RJS, localizado na Bacia de Santos (processo SEI nº 48610.233579/2023-38), foi utilizado o perfil de imagem acústica. O fluxograma apresentado na Figura 2 foi elaborado para descrever o trabalho.

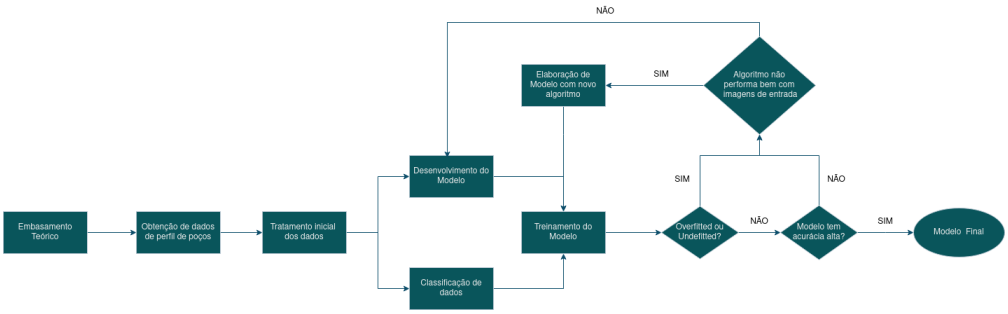


Figura 2 – Fluxograma de desenvolvimento do trabalho.
Fonte: Elaboração Própria.

3.1 Mineração de dados e Desenvolvimento dos Modelos

Inicialmente, foi realizado o processamento dos dados do perfil de imagem acústica. Foi feita a correção de velocidade, na qual o perfil é ajustado em profundidade para corrigir possíveis distorções causadas pelo cabo, em seguida foi realizado o alinhamento em relação ao norte magnético e correção de excentricidade, que visa corrigir descentralizações potenciais da ferramenta durante a aquisição da imagem. Por último foi feita a harmonização da imagem e normalização dos dados e distribuição de cores na imagem. O perfil

processado foi então exportado como arquivo PDF. Essa escolha foi motivada pela facilidade de manipulação do arquivo e pela capacidade de recortar a imagem original em imagens menores, simplificando assim seu manuseio e sua classificação.

O arquivo original PDF foi convertido para um arquivo PNG que foi então recortado em arquivos menores de mesmo tipo. Esses arquivos foram utilizados como dados de entrada para os modelos de ML. Em seguinda, foi feita a classificação das imagens, separando em imagens que contêm a coleta de amostra, e imagens onde não houve coleta de amostra.

Os primeiros modelos de classificação foram modelos de segmentação de imagem. Neles foram utilizadas imagens da mesma seção do perfil de imagem acústica estático e do perfil de tempo de trânsito. Junto das imagens originais foi feita a sua extração de aspectos utilizando filtros. Os filtros utilizados foram: Gabor, Canny Edge, Roberts, Sobel, Scharr, Prewitt, Farid, Gaussano com $\sigma = 3$. Além disso foram criadas máscaras para serem utilizadas como rótulos para validação durante a fase de treinamento do modelo.

Posteriormente, foram desenvolvidos os modelos de *Random Forest* (RF) e de *Suport Vector Machine* (SVM), e foi realizado o *tunning* (ajuste dos hiperparâmetros) do modelo. Os modelos após *tunning* apresentava os hiperparâmetros listados na Tabela 1.

Tabela 1 – Hiperparâmetros dos modelos de segmentação de imagens.

Random Forest		**SVM**	
Hiperparâmetro	Valor após *tuning*	Hiperparâmetro	Valor após *tuning*
n_estimators	600	*C*	0.1
min_samples_split	5	*gamma*	1
min_samples_leaf	1	*kernel*	rbf
max_features	None		
max_depth	True		

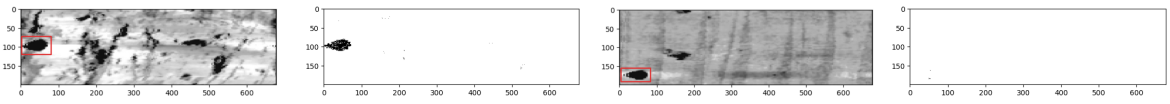
Após os modelos de segmentação de imagens foi desenvolvido um modelo de rede neural convolucional para ser feita a classificação e localização da imagem. Para isso foi necessário alterar os dados de entrada, e criar *bounding boxes* (caixa delimitadora) que foram utilizadas para fazer a localização das amostras nas imagens. A *bounding box* utilizada é compostas de dois pontos p_1 (x_1, y_1) e p_2 (x_2, y_2) que descrevem o ponto superior esquerdo e inferior direito respectivamente da *bounding box* na imagem. Para imagens onde não há coleta de amostra, foram adotados p_1 (0, 0) e p_2 (0, 0). Foi feita então a augmentação do dados com amostras, aplicando inversão horizontal, vertical e aumentando o contraste de cores das imagens. O modelo desenvolvido contém 5 camadas convolucionais, 5 camadas de *pooling* máximo, duas camadas densas e uma camada *flatten*, e duas cabeças de saída, uma sendo para os dados da *bounding box*, e outra para os dados da classificação da imagem. As duas contêm 4 camadas densas, 2 camadas *dropout* e uma camada de *output*.

Para o treinamento do modelo foi utilizado o otimizador *Stochastic Gradient Descent* (Gradiente Descendente Estocástico) com momento de 0.5 e *learning rate* de 10^{-4}. As funções de *loss* aplicadas aos *outputs* foram erro quadrático médio (MSE) e *Cross-Entropy* (Entropia Cruzada) para as cabeças da *bounding box* e da classe respectivamente. O peso das funções de loss foram iguais a 1 e as métricas utilizada durante o treinamento do modelo foram a acurácia para o *output* da classe, e o MSE para o *output* da *bounding box*. O modelo foi programado para treinar por 150 épocas com *batch size* de 50. Além disso, foram aplicadas funções de parada caso o modelo não evoluísse a função de loss para os dados de validação.

4 Resultados

O banco de dados ao final da classificação de todas as imagens continha 2142 imagens, com 106 imagens que apresentavam amostras e nas quais foram descritas a suas respectivas *bounding box*.

Após o treinamento dos modelos de segmentação de imagem, eles ainda não conseguiam fazer a segmentação de novas imagens e estava *overfitted*, pois ele conseguia classificar muito bem as imagens originais de treinamento, mas não conseguia extrapolar para imagens que não havia entrado em contato antes. As Figuras 3a e 3c apresentam a imagem original que foi utilizada como dado de entrada do modelo em conjunto com suas variantes e seção do perfil de tempo de transição, e nas Figuras 3b 3d é apresentado o *output* do modelo. Nas Figuras 3a e 3b o *input* foi de uma imagem utilizada no treinamento do modelo, que foi bem segmentada, sendo de fácil identificação onde houve a coleta da amostra nessa seção do perfil, e nas Figuras 3c e 3d o *input* foi de uma nova imagem, que não foi segmentada de maneira a identificar onde na imagem houve a coleta da amostra, delimitada em vermelho na imagem (c).

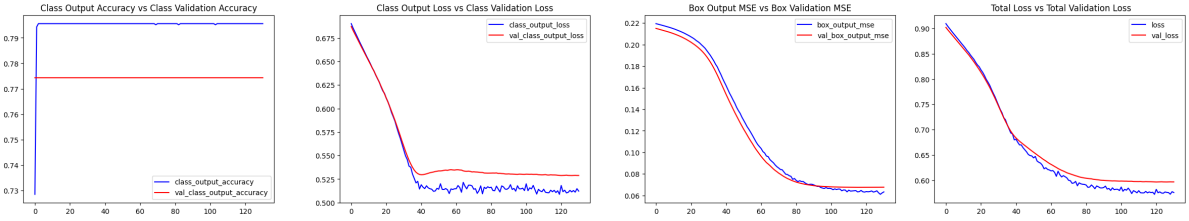


(a) (b) (c) (d)

Figura 3 – (a) imagem original - imagem usada no treinamento com amostra destacada em vermelho; (b) imagem segmentada - retorno do algoritmo. (c) imagem original - imagem nova para o modelo com amostra destacada em vermelho; (d) imagem segmentada - retorno do algoritmo.
Fonte: Elaboração Própria.

O modelo de CNN foi programado para treinar por 150 épocas, mas devido a função de parada aplicada no treinamento o modelo foi treinado por 131 épocas. Os resultados das medidas das funções de loss, acurácia e MSE do modelo foram plotados em um gráfico durante as épocas de treinamento e foram apresentados na Figura 4.

O modelo final apresentou classificação indevida de imagens onde houve a coleta de amostras as classificando como se não existisse coleta de amostra na imagem. Na Figura 4a é possível observar um problema no modelo na qual a acurácia dos valores de saída da classe não melhoram com o passar das épocas, podendo ser um sintoma de *overfitting* do modelo. Outro fator são os valores de *loss* altos que o modelo apresentou nos gráficos das Figuras 4b 4c e 4d.



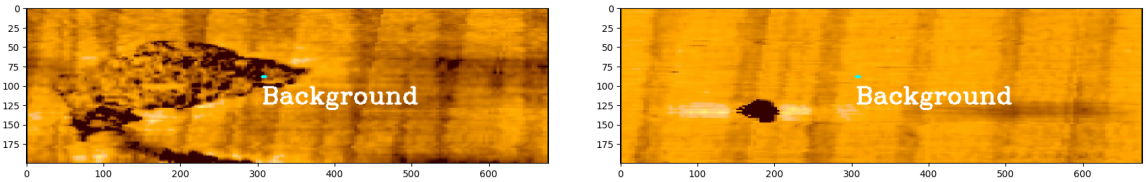
(a) (b) (c) (d)

Figura 4 – Gráficos de acurácia e *loss* do modelo ao longo das épocas de treinamento
Fonte: Elaboração Própria.

A bounding box também não conseguiu identificar a amostra, retornando valores muito próximos para os pontos p_1 e p_2. Devido a esses resultados em modelos de teste, foram alterados alguns hiperparâmetros e arquitetura da CNN, mas os modelos continuavam com o mesmo comportamento. Outro fator ressaltado durante as pesquisas foi o desbalanceamento das classes no banco de dados, que caracteriza o *dataset* montado nesse TCC. Algumas soluções eram *data augmentation* ou a aplicação de peso para as classes

do modelo a qual ajudariam o modelo a dar maior importância a uma das classes do banco de dados. A primeira solução foi aplicada, entretanto, a aplicação de peso nas classes não foi possível implementar devido a limitações técnicas na biblioteca utilizada para o desenvolvimento do modelo.

A Figura 5 apresenta o resultado da classificação de imagens, uma que não contém amostra (Figura 5a) e outra que contém amostra (Figura 5b). Como é possível notar na imagem, ambas têm a mesma classificação com valores muito próximos da *bounding box*.



(a) (b)

Figura 5 – (a) Exemplo de *output* do modelo para uma imagem sem amostra; (b) Exemplo de *output* do modelo para uma imagem com amostra.
Fonte: Elaboração Própria.

Algumas das possíveis causas para esse comportamento são: Modelo muito complexo, com elevado número de hiperparâmetros, o que facilitaria o seu *overfitting*; Conjunto de dados de treinamento e de validação são destoantes, pertencendo a domínios de hipóteses diferentes que não seriam generalizáveis; Alta taxa de aprendizado para o treinamento do modelo.

5 Conclusão

Ao avaliar os objetivos iniciais do trabalho, foi possível atingir os pontos de obtenção de dados de perfilagem e de classificação do perfil de imagem acústica ao criar os bancos de dados utilizados nos modelos. A implementação de algoritmos de ML foi concluída, com o pré-processamento dos dados para dar entrada no modelo, entretanto, os modelos desenvolvidos não foram capazes de fazer a classificação correta de regiões onde houve a coleta de amostras laterais no poço. Sua acurácia se manteve constante desde as primeiras épocas de treinamento, caracterizando um possível *overfitting* do modelo e além disso, as funções de *loss* das saídas e do modelo se estabilizaram em um valor alto.

Algumas possíveis causas levantadas foram o elevado número de parâmetros treináveis do modelo, a divergência entre os dados de treinamento e validação e uma elevada taxa de aprendizado empregada no treinamento do modelo.

6 Referências Bibliográficas

ALPAYDIN, E. (2010). *Introduction to machine learning*, Adaptive computation and machine learning, 2nd ed edn, MIT Press. OCLC: ocn317698631.

ASSAIFE, R. F. (2022). Perfilagem durante a perfuração de poços de petróleo: estudo bibliográfico, pp. 51–80.
URL: *https://www.nucleodoconhecimento.com.br/engenharia-mecanica/perfuracao-de-pocos*

BISHOP, C. M. (2006). *Pattern recognition and machine learning*, Information science and statistics, Springer.

DIERSEN, S., LEE, E.-J., SPEARS, D., CHEN, P., WANG, L. (2011). Classification of seismic windows using artificial neural networks, **4**: 1572–1581.
URL: *https://linkinghub.elsevier.com/retrieve/pii/S1877050911002286*

FACELI, K., LORENA, A. C., GAMA, J., CARVALHO, A. C. (2011). *Inteligência artificial: uma abordagem de aprendizado de máquina*, Grupo Gen - LTC. OCLC: 854567299.
URL: *http://site.ebrary.com/id/10707120*

GÉRON, A. (2019). *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems*, second edition edn, O'Reilly Media, Inc.

KOROTEEV, D., TEKIC, Z. (2021). Artificial intelligence in oil and gas upstream: Trends, challenges, and scenarios for the future, *Energy and AI* **3**: 100041.
URL: *https://linkinghub.elsevier.com/retrieve/pii/S2666546820300410*

OLIVATTO, T. F. (2021). *Identificação Automática De Rampas De Acessibilidade Apoiada Por Visão Computacional A Partir De Imagens Panorâmicas Street-level*, Mestrado, Universidade Federal de São Carlos, São Carlos - SP.
URL: *https://www.researchgate.net/publication/357132228*