Pontifícia Universidade Católica
do Rio de Janeiro

Guilherme Carneiro Meziat

Desenvolvimento de modelos utilizando Inteligência Artificial para detecção e diagnóstico de falhas em sistemas de óleo e gás marítimos

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Engenharia Química, de Materiais e Processos Ambientais, do Departamento de Engenharia Química e de Materiais da PUC-Rio.

Orientador: Prof. Brunno Ferreira dos Santos

Rio de Janeiro
Abril de 2025

Pontifícia Universidade Católica
do Rio de Janeiro

Guilherme Carneiro Meziat

Desenvolvimento de modelos utilizando Inteligência Artificial para detecção e diagnóstico de falhas em sistemas de óleo e gás marítimos

Dissertação apresentada como requisito parcial para obtenção do grau de Mestre pelo Programa de Pós–graduação em Engenharia Química, de Materiais e Processos Ambientais da PUC-Rio. Aprovada pela Comissão Examinadora abaixo:

Prof. Brunno Ferreira dos Santos
Orientador
Departamento de Engenharia Química e de Materiais – PUC-Rio

Prof. Roberto Bentes de Carvalho
Departamento de Engenharia Quimica e Materiais – PUC-Rio

Prof. Rejane Barbosa Santos
Instituto Federal de Educação, Ciência e Tecnologia do Sul de Minas Gerais

Rio de Janeiro, 24 de Abril de 2025

Guilherme Carneiro Meziat

Graduado em Engenharia Química pela PUC-Rio.

Ficha Catalográfica

Carneiro Meziat, Guilherme

Desenvolvimento de modelos utilizando Inteligência Artificial para detecção e diagnóstico de falhas em sistemas de óleo e gás marítimos / Guilherme Carneiro Meziat; orientador: Brunno Ferreira dos Santos. – 2025.

65 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Engenharia Química e de Materiais, 2025.

Inclui bibliografia

1. Engenharia Química – Teses. 2. Detecção de falhas. 3. Monitoramento de Poços de Petróleo. 4. Machine Learning. I. Ferreira dos Santos, Brunno. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Engenharia Química e de Materiais. III. Título.

CDD: 620.11

À minha namorada, meus amigos e minha família
pelo apoio e encorajamento.

Agradecimentos

Ao meu orientador, Professor Brunno Ferreira dos Santos, por ter apresentado esta linha de pesquisa e pela parceria fundamental para a realização deste trabalho.

À minha companheira, Ana Clara, por estar ao meu lado do começo ao fim da produção desta dissertação, me apoiando ao longo de todo este processo e sempre trazendo luz aos meus dias, mesmo quando à distância.

Aos meus amigos e colegas dentro e fora do curso com os quais dividi conhecimento, experiências e oportunidades.

Aos meus pais, Armando e Denise, pelo apoio e pelas conversas durante as situações difíceis ao longo deste último ano.

À ANP, ao CNPq e à PUC-Rio, pelos auxílios concedidos, sem os quais este trabalho não poderia ter sido realizado.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

Resumo

Carneiro Meziat, Guilherme; Ferreira dos Santos, Brunno. **Desenvolvimento de modelos utilizando Inteligência Artificial para detecção e diagnóstico de falhas em sistemas de óleo e gás marítimos**. Rio de Janeiro, 2025. 65p. Dissertação de Mestrado – Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

A indústria de óleo e gás, atualmente uma das mais complexas e maduras, é a principal motora da economia e energia nacional e mundial. Atividades relacionadas a esse ramo envolvem assuntos técnica e tecnologicamente avançados, com destaque para a exploração de óleo e gás *offshore*. Completamente diferente de como era em 1950, época da construção dos primeiros poços *offshore*, a crescente complexidade desse setor em prol de uma maior produtividade vem desafiando as metodologias tradicionais de monitoramento, controle e segurança de processos. A adaptação dessas técnicas ao potencial de processamento de dados apresentado por tecnologias baseadas em Inteligência Artificial (IA) apresenta-se atualmente como uma das maiores oportunidades de inovação no setor. Nesse cenário, o presente trabalho teve como objetivo o desenvolvimento de modelos para detecção e diagnóstico de falhas (anomalias) em poços de produção de petróleo *offshore* utilizando IA. Os modelos foram baseados na parcela real de dados do 3W *dataset*, pertencente ao domínio público desde 2019. A metodologia proposta divide os dados em três subconjuntos, de tal forma a realizar o treinamento de modelos de *Machine Learning* (ML) com validação cruzada, testar seus desempenhos e, finalmente, validá-los frente a uma simulação de situação real. Os modelos de Árvore de Decisão (*Decision Tree*, DT) e *Multi-Layer Perceptron* (MLP) foram treinados para duas situações distintas: detecção binária de falha e diagnóstico multi-classe de anomalias. Além disso, os modelos também foram treinados em uma versão do conjunto na qual os dados de operação normal e operação anômala foram balanceados. As métricas para avaliação dos modelos foram selecionadas levando em consideração tanto o desempenho do modelo quanto a sua capacidade de operar de forma segura. Assim, as métricas escolhidas foram o f1-score, o recall e a AUC. Outras métricas, como a precisão e a acurácia, também foram consideradas para possibilitar a comparação dos resultados com uma parcela maior da literatura existente. Os resultados encontrados foram satisfatórios e demonstram que os modelos avaliados são capazes de detectar e diagnosticar anomalias, em alguns casos com desempenhos superiores aos encontrados na literatura.

Palavras-chave

Detecção de falhas, Monitoramento de Poços de Petróleo, Machine Learning.

Abstract

Carneiro Meziat, Guilherme; Ferreira dos Santos, Brunno (Advisor). **Artificial intelligence model development for failure detection and diagnosis in offshore oil and gas extraction systems**. Rio de Janeiro, 2025. 65p. Dissertação de Mestrado – Departamento de Engenharia Química e de Materiais, Pontifícia Universidade Católica do Rio de Janeiro.

The oil and gas industry, currently one of the most complex and mature industries, is the primary driver of national and global economy and energy. Activities related to this area tackle advanced technical and technological topics, offshore oil and gas exploration being one of its highlights. A far cry from what it used to be like in the 1950s, when the first offshore wells were being built, the growing complexity of this sector towards greater productivity has been challenging traditional process monitoring, control, and safety methodologies. Adapting these techniques to harness the data processing potential offered by Artificial Intelligence (AI) technologies currently stands as one of the greatest opportunities for innovation in the sector. In this context, the present work aimed to develop fault (anomaly) detection and diagnosis models in offshore oil and gas production wells using AI. The models were based on real data instances from the 3W dataset, publicly available since 2019. The proposed methodology involved dividing the data into three subdatasets to train Machine Learning (ML) models with cross-validation, test their performances, and validate them in a simulated real-world scenario. The ML models – Decision Tree (DT) and Multi-Layer Perceptron (MLP) – were trained for two distinct situations: one-class binary fault detection and multi-class anomaly diagnosis. Additionally, the models were trained on a version of the data where the representation of normal and anomalous operation data was balanced. Metrics for model evaluation were selected based on both performance and ability to operate safely. Thus, the chosen metrics were f1-score, recall, and AUC. Other metrics such as precision and accuracy were also considered to enable the study to compare results to a broader segment of existing literature. The results obtained were satisfactory and demonstrate that the evaluated models are capable of detecting and diagnosing anomalies, in some cases outperforming those found in the literature.

Keywords

Fault detection, Oil Well Monitoring, Machine Learning.

Sumário

Lista de Figuras

Lista de Tabelas

1
Introdução

A indústria de óleo e gás atualmente contribui para 80 % da produção energética mundial e, com sua demanda ainda em alta pelo menos até 2030 tanto no mundo quanto no Brasil, deverá continuar nos holofotes até o final da década (Figura 1.1) (Freitas; Almeida; Fernández, 2023; Iea, 2024; Schernikau; Smith, 2022).

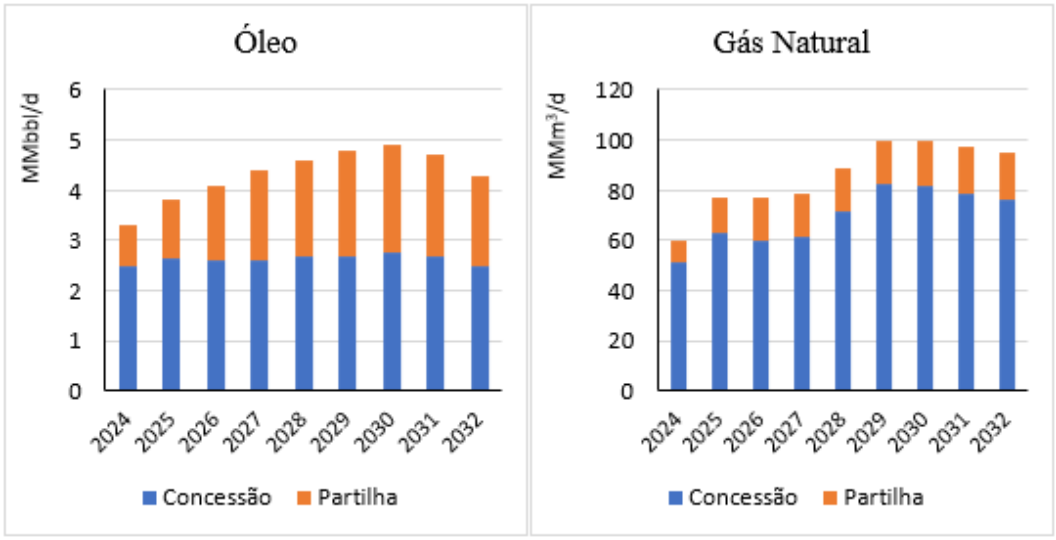


Figura 1.1: Demanda de Petróleo e Gás Natural no cenário nacional. Adaptado de: (Freitas; Almeida; Fernández, 2023)

O Brasil se apresenta como um dos maiores produtores de petróleo, ocupando a oitava posição em 2023, uma acima do ano anterior (Energy Institute, 2023). A situação atual do país se deve majoritariamente à descoberta do pré-sal em 2006 pela Petrobras. Dados da Agência Nacional do Petróleo, Gás Natural e Biocombustíveis (ANP) demonstram que, no Brasil, dos 4,3 milhões de barris equivalentes de petróleo produzidos diariamente em média, mais de três quartos provêm do pré-sal e por volta de 95 % provêm de poços marítmos (Figura 1.2) (ANP, 2024). Tanto mundialmente quanto nacionalmente, a prosperidade do setor de óleo e gás é resultado de constantes inovações tecnológicas e demandas energéticas. Mesmo países que possuíam claras alternativas, como o próprio Brasil com o programa Proálcool, eventualmente penderam novamente para o caminho dos combustíveis fósseis.

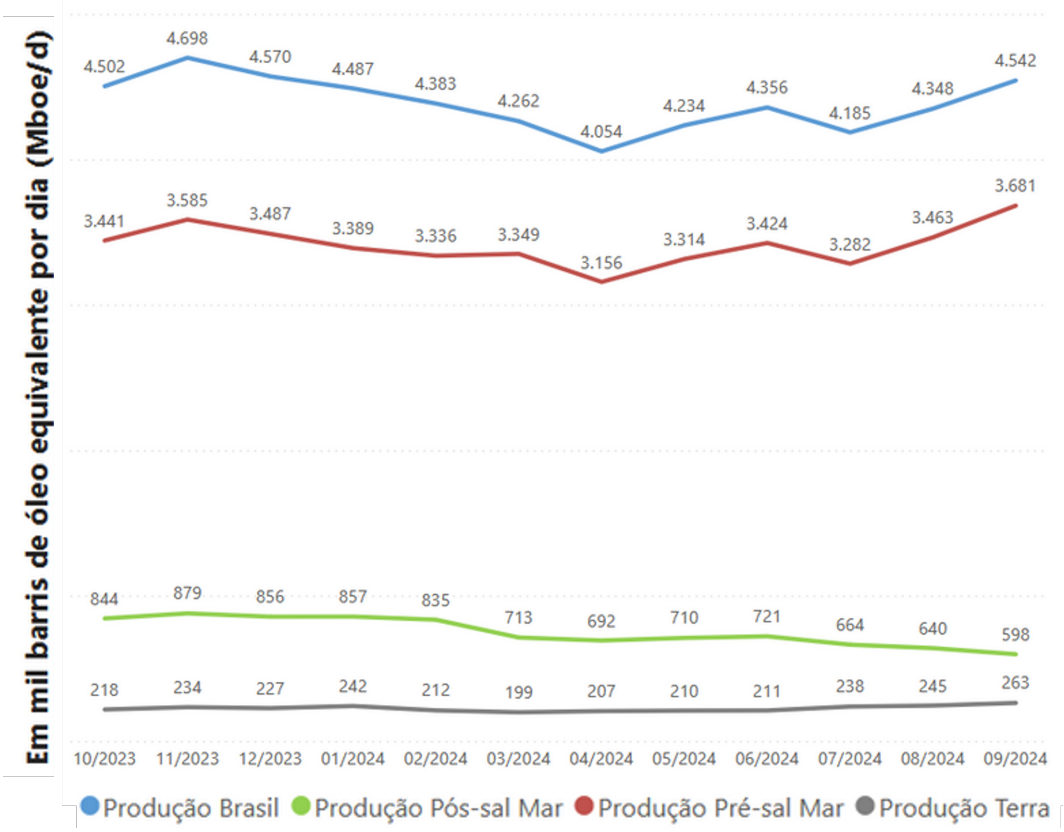


Figura 1.2: Evolução da produção onshore e offshore – Pré-sal x "Pós-sal".
Adaptado de: ANP, 2024

As constantes inovações no setor de óleo e gás em prol de uma maior produtividade trouxeram consigo um nível profundo de complexidade para as suas plantas, especialmente offshore, tornando-as extremamente sensíveis a variações operacionais. Em 2022, o quarto acidente mais comunicado em plataformas de produção foi "interrupção não programada superior a 24 horas decorrente de incidente operacional", com uma frequência média de mais de três comunicações por mês (Figura 1.3) (ANP, 2022). De acordo com Júnior (2022), a interrupção da operação de uma plataforma gera prejuízos de mais de 1 milhão de dólares por dia. Ressaltando outra consequência importante das circunstâncias do setor, o quinto acidente mais comunicado, também com uma frequência média de quase três comunicações mensais, foi "Ferimento com afastamento por mais de três dias"(ANP, 2022), denotando o desafio de se operar as plataformas com segurança.

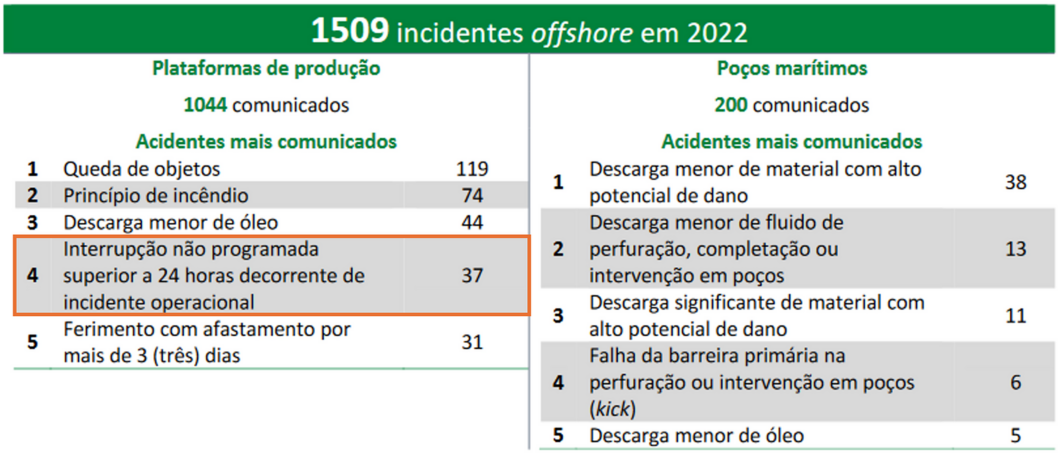


Figura 1.3: Incidentes reportados em plataformas *offshore* nacionais em 2022. Adaptado de: ANP, 2022

A base da metodologia moderna de monitoramento e controle de processos envolve uma coleta extensiva de dados sobre o processo e a necessidade de um amplo conhecimento da unidade monitorada. Ao verificar de forma constante a saúde de equipamentos e correntes do processo, é possível definir quais são as áreas de avaliação mais críticas, além de valores-limite para determinar quando o processo está em tendência de colapso (Goode; Roylance; Moore, 2000; Wang, 2000).

Nesse contexto, uma tecnologia emergente possui ótima sinergia com a ampla quantidade de dados gerados durante o monitoramento de processos e é tida como uma das mais promissoras para superar os desafios na indústria de óleo e gás: a Inteligência Artificial (IA) (Schweidtmann et al., 2021). Mais especificamente, a sua aplicação na forma de *Machine Learning* (ML) pode gerar modelos capazes de identificar anomalias em processos, algumas das quais podem levar a falhas mais graves na produção.

Diversos algoritmos de ML já foram testados com sucesso para a identificação de anomalias na indústria de óleo e gás. Xu, Du e Zhang (2019) propuseram um algoritmo baseado em Redes Neurais Artificiais (RNAs) para prever falhas em dutos, alcançando uma AUC de quase 90%. Zhang et al. (2022) conseguiram ajustar um modelo usando um algoritmo baseado em Árvores de Decisão (ADs), que apresentou uma AUC melhor (91,7%), mas com uma pontuação F1 menor (76% em comparação com os 82% de Xu).

O ML também foi testado com sucesso em outras áreas da indústria de óleo e gás. Por exemplo, em refinarias petroquímicas e operações de perfuração de poços, foi demonstrado que modelos de manutenção preditiva baseados em RNAs são viáveis e possuem grande potencial (Alkinani; Al-hameedi; Dunnnorman, 2020; Suursalu, 2017; Helmiriawan, 2018). Na área de extração de petróleo, as Árvores de Decisão (ADs) se mostraram mais capazes do que outros métodos ao definir correlações não lineares entre parâmetros usados para injeção de água de baixa salinidade, uma técnica de recuperação avançada de petróleo (Salimova; Pourafshary; Wang, 2021)

1.1 Objetivos

De acordo com o contexto apresentado acima, o presente trabalho tem como objetivo principal desenvolver modelos de Machine Learning (ML) para detectar e diagnosticar anomalias com segurança e eficiência.

O trabalho também possui objetivos específicos, dispostos abaixo:

1. Identificar condições favoráveis e desfavoráveis para os modelos testados
2. Comparar o desempenho de diferentes modelos de ML
3. Identificar o tipo de modelo de ML mais apto para detectar e diagnosticar anomalias no processo de produção de petróleo *offshore*

Este trabalho visa atingir os objetivos citados acima oferecendo um ambiente próprio para modelos de ML, ou seja: com uma ampla variedade de dados com formatação consistente e ordenada. Os dados disponíveis serão organizados de diversas formas distintas e as métricas utilizadas para avaliar os modelos levarão em consideração tanto questões de desempenho quanto de segurança.

1.2 Estrutura da dissertação

Este documento está estruturado da seguinte forma: O Capítulo 1 introduziu e contextualizou o assunto da dissertação, bem como apresentou sua motivação e objetivos. O Capítulo 2 contém a fundamentação teórica necessária para o problema em questão, bem como trabalhos relacionados. O Capítulo 3 explica a metodologia utilizada, seguida pelos resultados no Capítulo 4. No Capítulo 5, por fim, apresentam-se a conclusão e possíveis trabalhos futuros.

2
Fundamentação Teórica

Este capítulo descreve diferentes conceitos, métodos e técnicas cujo entendimento básico é necessário para uma compreensão completa da dissertação. Tópicos relacionados a diferentes esferas de estudo serão apresentados a um nível de profundidade adequado para o presente trabalho. A aplicação destes tópicos será definida no capítulo 3.

2.1
Produção de Petróleo *Offshore*

O termo "produção" é utilizado alternadamente com "elevação" para se referir às operações realizadas dentro do poço de petróleo diretamente relacionadas ao ato de retirar o petróleo das camadas profundas da crosta terrestre e fazê-lo fluir até a plataforma, fazendo parte das operações *upstream*. Em poços *offshore*, os equipamentos de completação do poço e o *manifold*, também chamado de árvore de natal, precisam ser posicionados no leito marinho e conectados à plataforma na superfície (Thomas, 2004; Laik, 2018). A Figura 2.1 abaixo apresenta um esquema de produção de petróleo *offshore* por um FPSO (*Floating Production and Storage Offload, Navio-Plataforma*).

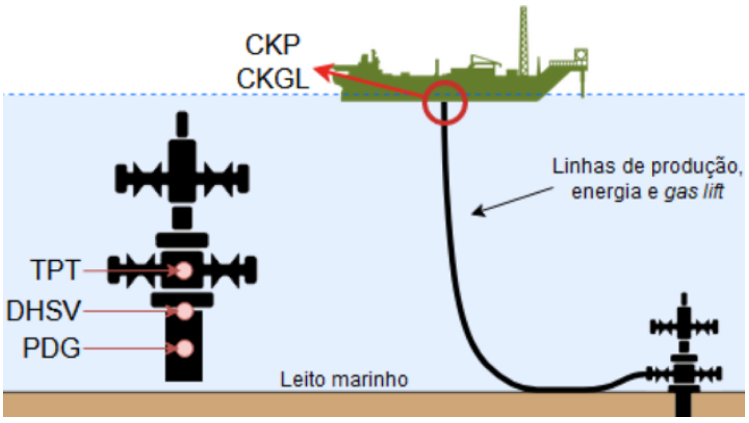


Figura 2.1: Esquema da estrutura básica dos sistemas produtores de onde os dados do 3W *dataset* foram coletados. Adaptado de: (Vargas et al., 2019)

As condições do poço de petróleo afetam diretamente os procedimentos normais de produção. Poços que possuem pressão o suficiente para provocar o escoamento de petróleo até a superfície são denominados surgentes e operados através da elevação natural fornecida pelo poço. A operação destes poços é mais simples e, consequentemente, mais lucrativa. Em contrapartida, poços que não possuem pressão suficiente para levar o petróleo até a superfície são denominados não-surgentes e precisam ser operados através de procedimentos de elevação artificial. Estes poços podem ser poços maduros que perderam surgência ao longo dos anos, ou poços que já possuem um baixo índice de produtividade desde a sua descoberta (Thomas, 2004).

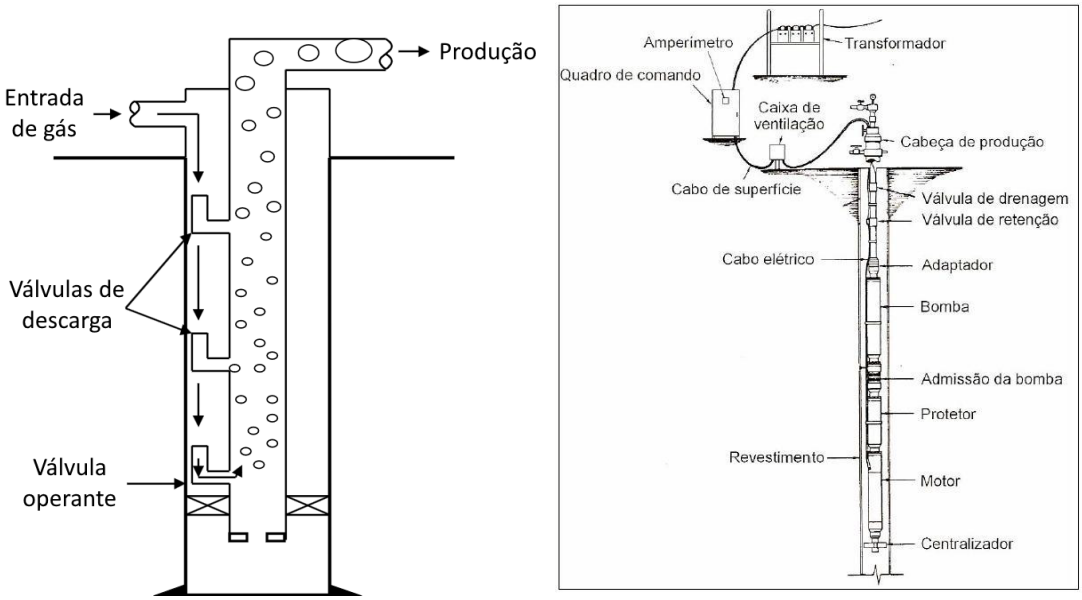
Existem algumas formas de se operar um poço de petróleo *offshore* através de elevação artificial. Dois dos mecanismos mais comuns são o Bombeio Centrífugo Submerso (BCS) e o *gas lift* (Thomas, 2004; Bai; Bai, 2016).

A elevação por BCS é um mecanismo puramente mecânico, no qual o sistema de bombeamento é submerso diretamente no poço de produção, abaixo da árvore de natal. A energia fornecida ao sistema é transmitida diretamente para a bomba por um cabo elétrico. O sistema possui múltiplos estágios, é tipicamente utilizado em poços que produzem majoritariamente petróleo ao invés de gás e possui melhor desempenho ao bombear fluidos de baixa viscosidade, ou seja, com alto teor de água. Bombas projetadas para o BCS possuem capacidade de elevação de até 5000 metros e podem ser utilizadas em poços de alta ou baixa vazão, oferecendo flexibilidade na variação do número de estágios da bomba. Assim como em qualquer outro sistema de bombeamento, a pressão de entrada, perda de carga e limites mecânicos devem ser levados em consideração durante a seleção do tipo de bomba e seus acessórios para a realização do BCS (Thomas, 2004; Bai; Bai, 2016).

A elevação por *gas lift*, por sua vez, é relativamente mais complexa, se tratando de uma transferência de energia menos direta. Apesar disso, é considerado um mecanismo mais versátil do que o BCS, levando preferência em poços em menor profundidade, com maior parcela de gás natural e fluidos mais viscosos e arenosos. Este mecanismo é capaz de produzir até 2600 metros de elevação e pode ser utilizado em poços com capacidade de vazão de 1 a 1700 m^3/d (Thomas, 2004; Bai; Bai, 2016).

O sistema de *gas lift* é composto por um compressor localizado na superfície da plataforma que bombeia gás para o fundo da coluna de produção. Este bombeio pode ser realizado de forma contínua, gaseificando o fluido e diminuindo a perda de carga ao longo da coluna de produção devido à menor densidade, ou de forma intermitente, induzindo golfadas de fluido que são escoadas pelo gás em alta pressão até a superfície. O *gas lift* intermitente é utilizado em poços que não possuem pressão estática o suficiente para produzir petróleo de forma satisfatória através do *gas lift* contínuo, sendo mais típico em poços de baixa produção (Thomas, 2004; Bai; Bai, 2016; Guo; Lyons; Ghalambor, 2007).

A Figura 2.2 a seguir apresenta o funcionamento básico de ambos os mecanismos de elevação.



(a) Adaptado de: (Guo; Lyons; Ghalambor, 2007)

(b) Fonte: (Maitelli et al., 2008)

Figura 2.2: Métodos de elevação artifical por (a) *Gas Lift* e (b) BCS.

2.1.1
Garantia de Escoamento e Eventos Indesejados na Produção de Petróleo

Mencionado pela primeira vez nos anos 1990, a garantia de escoamento é atualmente apontada como o maior desafio técnico para a produção de petróleo em plataformas *offshore* profundas. Isso se dá pela alta variabilidade de condições do processo, que precisa escoar o fluido desde o leito marinho até a superfície, tornando-o suscetível a falhas de instrumentação e formação de diversos tipos de incrustações (Gudmundsson, 2017; Laik, 2018).

De acordo com Júnior (2022), a interrupção operacional de plataformas produtoras devido a falhas e a subsequente utilização de sondas para realizar a manutenção e reparos necessários causam prejuízos milionários aos produtores em escala diária. Além disso, as plataformas correm risco de sofrer uma grande variedade de anomalias, atingindo tanto o reservatório, quanto os sistemas de integridade do poço e até mesmo o escoamento em si do petróleo desde o reservatório até a plataforma (Figura 2.3).

Escoamento	Integridade		Reservatório
	Completação seca	Completação submarina	
– Hidratos – Parafinas – Asfaltenos – Sulfato de bário – Sulfato de estrôncio – Sulfato de cálcio – Carbonato de cálcio – Materiais radioativos de ocorrência natural (NORM)	– Cabeça de poço – Revestimento de produção – Coluna de produção – Válvula de segurança (DHSV) – Válvula de segurança anular (ASV) – Válvula de *gas lift* (VGL) – Cimento – *Packer*	– Arvore de Natal Molhada (ANM) – Base adaptadora de produção (BAP) – Válvula de segurança (DHSV) – Revestimento de produção – Coluna de Produção – *Flowline*	– Produção excessiva de água – Produção excessiva de gás – Produção de areia – Migração de finos

Figura 2.3: Eventos que geram intervenção de manutenção em poço caracterizados por falhas ou anomalias. Adaptado de: (Suaznabar; Moroocka; Miura, 2020)

Esta dissertação utilizará o banco de dados intitulado *3W dataset*, criado pela Petrobras e publicado por Vargas et al. (2019). O banco de dados possui este nome por estar relacionado a dados de monitoramento de poços (*Wells*, em inglês) e devido às três diferentes fontes de dados que o compõem: dados reais, gerados por simulações e desenhados à mão por especialistas no ramo.

De acordo com o *3W dataset*, eventos indesejados que colocam a operação de produção em risco podem ser agrupados em 8 classes de anomalia:

1. Aumento abrupto de BSW *(Basic Sediment and Water)* - o BSW é o teor de água e sedimentos dentro do fluido produzido, ou seja, diz respeito à parte do fluido que não é de interesse ao processo. Métodos de elevação artificial introduzem turbulência no sistema e podem interferir no leito do poço, o que tende a aumentar o BSW e, consequentemente, diminuir a produção de petróleo e gás. Além disso, o excesso de BSW induz uma maior perda de carga e formação de incrustações ao longo do escoamento do fluido (Vargas et al., 2019);

2. Fechamento espúrio da DHSV (*DownHole Safety Valve*) - a DHSV é uma válvula de segurança localizada na coluna de produção e opera normalmente aberta, fechando em ocasiões de emergência para evitar fluxo descontrolado do fluido ao longo da coluna. Esta válvula pode ser atuada de forma espúria, sem ser detectado pelo supervisório, gerando uma parada de operação;

3. Intermitência Severa (do inglês, *Severe Slugging*) - é caracterizada por golfadas periódicas e intensas na produção, também chamadas de *Slugging*, provocadas por uma alternância entre bolsões de líquido e grandes bolhas de gás ao longo da coluna. É um tipo de evento crítico, provocando variações na pressão, vazão e composição do fluido capazes de danificar equipamentos em linha (Vargas et al., 2019);

4. Instabilidade de Fluxo - é caracterizada pela variação de pelo menos uma das variáveis monitoradas dentro de amplitudes toleráveis, podendo ser identificado como um precursor do evento de Intermitência Severa;

5. Perda Rápida de Produtividade - Também chamado de perda de surgência, esta anomalia diz respeito a uma alteração das condições do poço e propriedades do fluido escoado de tal forma que o fluxo seja reduzzido de

forma crítica ou interrompido. Pode estar correlacionado a outras classes de anomalia aqui descritas, como as classes 1, 4, 7 e 8 (Guo; Lyons; Ghalambor, 2007; Hausler; Krishnamurthy; Sherar, 2015; Vargas et al., 2019);

6. Restrição Rápida na Válvula Choke de Produção (CKP) - a Válvula CKP controla o escoamento do fluido na sua chegada à superficie e pode ser operada manualmente, se necessário. Nesses momentos, erros operacionais podem causar uma restrição desnecessária do fluido;

7. Incrustações na Válvula CKP - devido à grande variação das condições de temperatura e pressão ao longo da coluna de produção, o ponto de restrição na válvula CKP se torna um ponto altamente suscetível à formação de incrustações, diminuindo a capacidade de vazão e controle da vazão do fluido;

8. Hidrato em Linha de Produção - Assim como as incrustações descritas na classe acima, hidratos podem se formar e acumular na tubulação próxima ao leito marinho. As condições de alta pressão e baixa temperatura favorecem a reação entre a água e hidrocarbonetos de cadeia curta ou gases de baixo peso molecular, formando estruturas cristalinas que se assemelham a gelo (VerÇosa et al., 2018).

As classes de anomalia apresentadas acima se aplicam tanto a poços surgentes quanto a poços não-surgentes, pois os equipamentos envolvidos são fundamentais para ambos os tipos de poço.

2.2 *Machine Learning*

O aprendizado de máquina (*Machine Learning* – ML), uma subárea da Inteligência Artificial (IA), consiste no desenvolvimento de algoritmos que ajustam seu desempenho automaticamente a partir da análise de dados. O aprendizado acontece na medida em que o modelo ajustado se torna capaz de descrever com sucesso o sistema analisado a partir da generalização de uma visão limitada, fornecida pelos dados coletados. Esse aprendizado pode ser:

- supervisionado: o modelo compara de forma direta os resultados obtidos a dados já rotulados com o atributo de interesse;
- não supervisionado: o modelo identifica padrões intrínsecos à estrutura dos dados analisados, sem rotulação definida;
- por reforço: o modelo é direcionado ao ponto desejado de forma iterativa a partir da maximização de uma função-objetivo específica ao cenário explorado (Igual; SeguÍ, 2017).

No contexto desta dissertação, os dados analisados já possuem rótulos definidos e há um resultado específico desejado, configurando um problema de classificação supervisionada no qual as classes identificadas pelos modelos treinados deverão coincidir com os rótulos dos dados analisados. A escolha dos algoritmos de ML a serem utilizados levará isso em consideração (Igual; SeguÍ, 2017; Bramer, 2020).

2.2.1
Mutual Information

Mutual Information (MI) se baseia no conceito de entropia computacional, que se assemelha intimamente à entropia termodinâmica em termos probabilísticos. Dada uma variável discreta aleatória X, a sua entropia está diretamente relacionada à quantidade de valores (ou estados) x_i que X pode assumir, bem como a probabilidade de X assumir cada um dos valores x_i. Como um exemplo, para uma distribuição uniforme de X, sua entropia se simplifica a H(X) = ln M, onde M é a quantidade de estados que x_i pode assumir (Bishop, 2006).

No contexto computacional, a redução da entropia pode ser traduzida em ganho de informação. Quanto menor a entropia, menor a quantidade de estados possíveis, o que implica na redução da incerteza do comportamento da variável em mãos. A partir desta noção, a informação mútua (ou MI) entre duas variáveis X e Y é definida como sendo a redução da entropia (ou o ganho de informação) de X a partir da observação de Y. Este conceito pode ser descrito pela equação abaixo:

$$MI[X,Y] = -\int\int p(X,Y) ln\left(\frac{p(X)p(Y)}{p(X,Y)}\right) dXdY = H[X] - H[X|Y]$$

Se X e Y forem independentes entre si, então p(X,Y) = p(X)p(Y) e a integral acima resultará em zero. Assim, percebe-se que o MI é uma forma de avaliar a dependência das variáveis X e Y entre si, representando a redução na incerteza de X dado a observação Y. Sob uma ótica probabilistica, o MI quantifica a diferença entre as distribuições p(x) e p(X|Y). Se X e Y forem variáveis independentes, então p(X|Y) = p(X) e não há ganho de informação sobre a variável X ao observar Y, ou seja, as variáveis não possuem informação mútua (Bishop, 2006).

A partir das definições apresentadas acima, o MI pode ser utilizado como um critério de Feature Selection (FS) ao quantificar a dependência entre as variáveis em um banco de dados e a classe auferida. Dessa forma, MI mede o nível de contribuição da presença ou ausência de cada variável para a classificação do dado analisado ao comparar a distribuição da variável em cada classe. Quanto menor a diferença entre essas distribuições, menor o ganho de informação acarretado pela observação da variável, logo menor o valor de MI e menos importante será a variável no banco de dados(Bishop, 2006; Manning; Raghavan; SchÜtze, 2008).

2.2.2
Análise de Componentes Principais

A Análise de Componentes Principais (APC) é um dos métodos mais amplamente utilizados na estatística de multivariáveis, com evidências deste método sendo utilizado há mais de dois séculos. O intuito de uma APC é analisar um conjunto de dados com variáveis dependentes e correlacionadas entre si e, a partir desta análise, gerar um novo conjunto contendo apenas a informação relevante presente no conjunto antigo. Dessa forma, o conjunto de dados pode ser simplificado e seu número de variáveis reduzido, chamado de

redução de dimensionalidade, com perda mínima de qualidade (Abdi; Williams, 2010).

O conjunto gerado é composto por Componentes Principais (CPs), baseados em combinações lineares das variáveis originais, definidos a partir dos autovetores da matriz de covariância do banco de dados. O primeiro componente captura a maior variância possível dos dados analisados, o segundo componente, a segunda maior variância e o componente N captura a N maior variância do banco. Estes componentes são todos ortogonais, ou seja, não-correlacionados, e por serem combinações lineares das variáveis originais, podem ser interpretados como uma rotação dos eixos do espaço amostral original, de forma a maximizar a informação com menos variáveis. Os CPs obtidos podem ainda ser comparados às variáveis originais como uma avaliação da contribuição de cada variável para a informação presente nos dados analisados(Abdi; Williams, 2010).

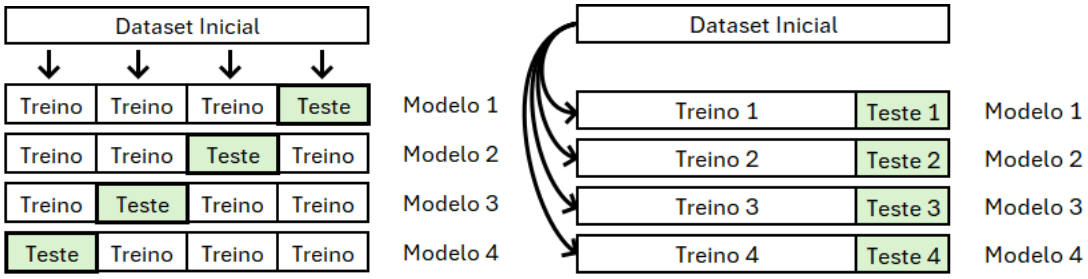
2.2.3 Segmentação de dados

Geralmente, devido às características de um algoritmo de ML, deseja-se utilizar a maior quantidade possível dos dados disponíveis para o ajuste do modelo, mantendo uma parcela razoável de dados disponíveis para a sua validação. As duas técnicas mais comuns de segmentação de dados são chamadas de *Holdout* e *Cross-Validation* (Bishop, 2006; Aggarwal, 2015).

Na técnica de *Holdout*, os dados são divididos em dois conjuntos de forma aleatória, onde a parcela de dados amostrados para o treino do modelo geralmente recebe uma quantidade maior de dados (entre dois terços e três quartos do banco). É uma técnica simples, de baixa demanda computacional e facilmente reprodutível para uma estimativa mais confiável, porém não é recomendado para bancos de dados desbalanceados. Por exemplo, em um banco de dados binário no qual apenas 0,1% dos dados pertencem a uma das classes, o risco de se gerar um conjunto de teste, ou até mesmo de treino, com dados apenas de uma das classes aumenta consideravelmente, prejudicando ou impedindo completamente a avaliação do modelo treinado. Nesse caso, a aleatoriedade da divisão pode ser parcialmente restrita ao amostrar cada classe de forma isolada, ao invés de aplicar a amostragem ao banco de dados completo (Aggarwal, 2015).

O *K-fold Cross-Validation* (CV), por sua vez, envolve o particionamento do banco de dados original em um número K de segmentos. Destes K segmentos, um deles é escolhido como o conjunto de teste, enquanto os outros são utilizados para treinar o modelo, gerando uma configuração similar à gerada na técnica de *Holdout*. A principal diferença é que, no CV, o procedimento acima é repetido até que todos os K segmentos tenham sido selecionados uma vez como o conjunto de teste do modelo, essenciamente reproduzindo a técnica de *Holdout* K vezes. A estimativa apresentada de desempenho do modelo geralmente é mais confiável e robusta do que na técnica de *Holdout*, sendo a média de todos os resultados obtidos, porém demanda uma quantidade significativamente maior de recursos computacionais, crescendo exponencialmente com o valor de K (Bishop, 2006; Aggarwal, 2015).

A Figura 2.4 demonstra as similaridades e diferenças entre as duas técnicas apresentadas. É importante ressaltar que ambas as técnicas podem ser utilizadas simultaneamente, criando uma situação na qual se cria um conjunto *Holdout* e o CV é aplicado nos dados restantes de treino. Essa combinação é uma prática recomendada, pois permite testar os modelos em dados completamente não vistos após uma validação cruzada robusta. (Aggarwal, 2015).



(a) *K-Fold* CV (b) *Holdout*

Figura 2.4: Técnicas de segmentação de dados.

2.2.4 Redes Neurais Artificiais

Redes neurais artificiais (RNAs) são algoritmos de ML análogos ao sistema nervoso humano. RNAs contêm uma rede de unidades computacionais chamadas neurônios, com conexões entre si que possuem pesos diferentes. A forma mais simples de uma RNA é denominada perceptron, cuja estrutura pode ser vista na Figura 2.5. O perceptron é composto por uma camada de entrada e uma camada de saída. A camada de entrada contêm os valores de entrada das variáveis e o peso de cada variável, enquanto a camada de saída é uma camada computacional na qual os valores caracterizados pelos pesos são acumulados e transformados em um ou mais sinais de saída por meio de funções de ativação (Bishop, 2006; Aggarwal, 2018).

O treinamento das RNAs ocorre por meio do algoritmo de retropropagação do erro (*backpropagation*), que ajusta iterativamente os pesos com base no gradiente do erro calculado entre a saída prevista e a saída real, permitindo que a rede aprenda a generalizar seu aprendizado para novos exemplos não vistos. O perceptron também pode incluir neurônios de *bias* para auxiliar a modelagem feita com distribuições desbalanceadas de dados ao adicionar um valor fixo ao somatório ponderado das entradas, funcionando como um deslocamento da função de ativação (Aggarwal, 2018).

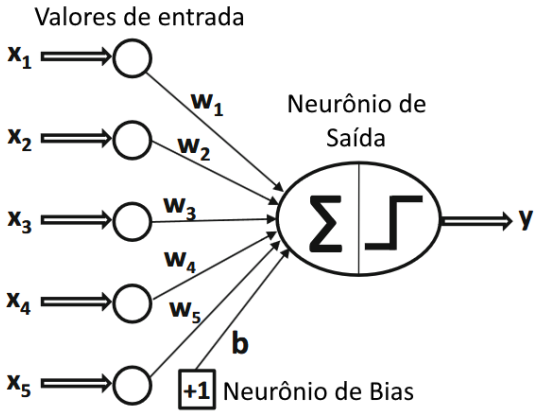


Figura 2.5: Estrutura básica de um Perceptron, onde x_i representa os valores de entrada, w_i representa os pesos atribuídos a cada valor, b é o *bias* aplicado e y representa o valor de saída. Adaptado de (Aggarwal, 2018)

RNAs que possuem camadas intermediárias, tambem chamadas de multicamada, são denominadas *Multi-Layer Perceptron* (MLP). Em um MLP, as camadas intermediárias são compostas por mais neurônios, cada um com suas entradas, saídas e funções de ativação, formando uma arquitetura conhecida como *feed-forward*, conforme mostrado na Figura 2.6 (Aggarwal, 2018).

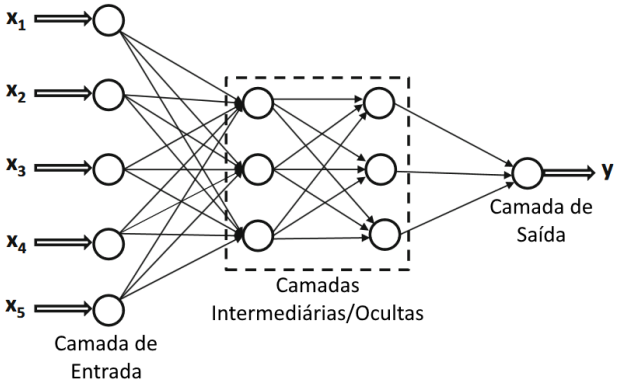


Figura 2.6: Estrutura básica de um MLP, revelando as camadas intermediárias. Adaptado de (Aggarwal, 2018)

A escolha da função de ativação é essencial no *design* de redes neurais, pois define como os valores de entrada são transformados nas saídas. A Figura 2.7 dispõe algumas funções comuns de ativação. MLPs costumam obter melhor desempenho com funções de ativação não lineares, como sigmoide ou tangente hiperbólica, permitindo que a rede modelada consiga detectar relações mais complexas nos dados (Aggarwal, 2018).

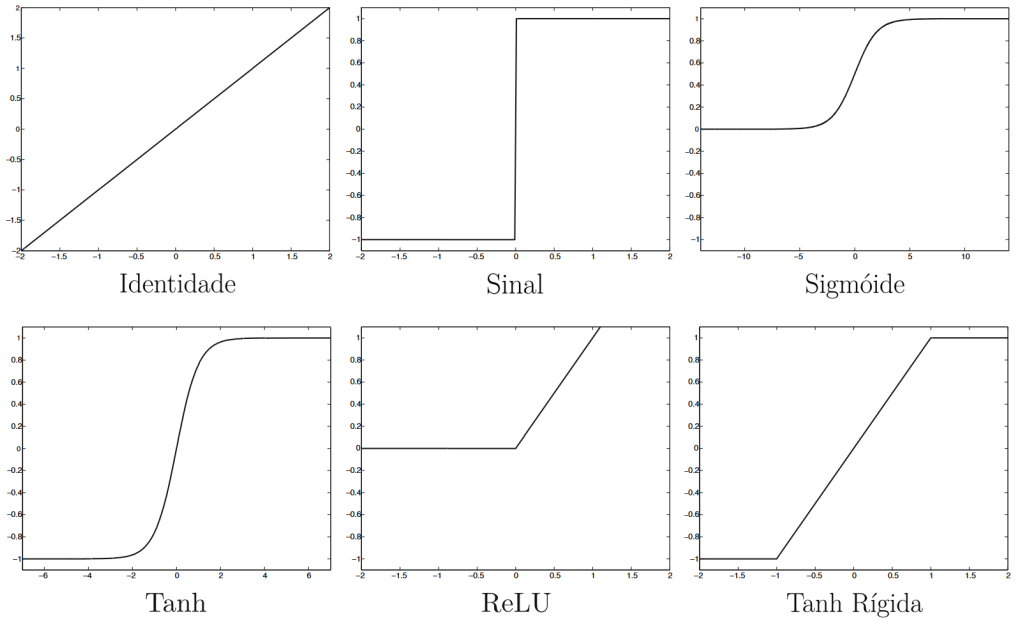


Figura 2.7: Funções de ativação comumente usadas para o treinamento de redes neurais. Adaptado de (Aggarwal, 2018)

2.2.5
Árvore de Decisão

A árvore de decisão (AD) é um tipo de algoritmo *"Divide and Conquer"*, no qual um conjunto de dados é dividido progressivamente até que os grupos resultantes sejam simples, ou puros, o bastante para serem classificados pelo algoritmo. Cada ponto de divisão recebe o nome de nó e a impureza de um nó, refere-se ao grau de mistura de classes nos dados que o compõem, no qual um grau de mistura menor confere uma pureza maior ao nó. Um nó é considerado puro quando todos os exemplos pertencem à mesma classe. As divisões podem ser binárias ou multiclasse e os nós que definem a classificação dos dados são chamados de terminais, ou folhas, como a folha de uma árvore. (Dinov, 2018).

O processo de classificação ocorre iterativamente, do primeiro nó aos nós terminais, identificando características e padrões na estrutura dos dados analisados que possibilitam a sua segmentação em classes distintas. Essa segmentação é guiada por critérios baseados em funções que dependem da distribuição dos dados, além da comparação dos resultados previstos pelo modelo aos dados fornecidos como entrada (Dinov, 2018). Cada iteração adiciona uma camada à AD, logo o número de camadas da árvore é igual ao número total de iterações realizadas até se chegar ao último nó terminal.

Um exemplo de uma árvore de decisão pode ser visto abaixo, na Figura 2.8. Esta figura ilustra os tipos de decisão tomadas por um jogador de golfe nos dias em que costuma praticar. Percebe-se que, para a situação de tempo nublado, o jogador não necessita de mais nenhuma informação para tomar sua decisão, caracterizando um nó puro logo na primeira camada da árvore. Para o tempo ensolarado ou chuvoso, mais uma camada é necessária para a

tomada de decisão do jogador, buscando informações sobre a umidade e o vento, respectivamente.

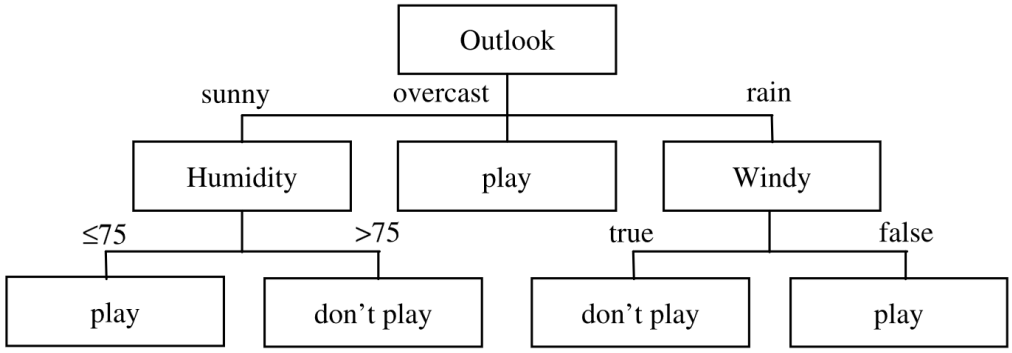


Figura 2.8: Árvore de Decisão de um jogador de golfe Sobre jogar golfe. Adaptado de (Bramer, 2020)

ADs tem como objetivo sempre minimizar a impureza, ou maximizar o ganho de informação sobre os dados analisados em cada divisão realizada. Esse princípio se assemelha bastante ao princípio da utilização do MI como um método de FS. Uma diferença é que, na AD, geralmente se utiliza uma outra métrica além da entropia para avaliar a melhor divisão disponível: o índice de Gini. Enquanto a entropia é definida como sendo o nível de incerteza de um conjunto de dados, o índice de Gini mede a probabilidade de um dado aleatoriamente selecionado ser classificado erroneamente pelo modelo. Dependendo das características dos dados avaliados, como balanceamento e espaço amostral, tanto a entropia quanto o índice de Gini podem ser mais apropriado para treinar o modelo de AD (Dinov, 2018).

A divisão dos dados pode ser encerrada de algumas formas:

- quando não é mais possível dividí-los, ou seja, quando o grupo em questão possui somente um dado;
- quando o conjunto é julgado completamente puro, ou seja, todos os dados pertencem à mesma classe;
- quando o conjunto possui pureza acima de um nível de tolerância previamente definido;
- quando a árvore atinge um limite máximo de divisões ou camadas de profundidade previamente definidos.

Para evitar o sobreajuste dos modelos, estratégias de poda podem ser aplicadas, consistindo na eliminação de divisões julgadas ineficientes pelo modelo. A poda pode acontecer durante a modelagem, sendo chamada de pré-poda, ou após a modelagem, chamada de pós-poda. A pré-poda geralmente consiste em uma limitação baseada na redução da impureza causada por uma divisão e ocorre durante a criação do modelo. A pós-poda, por sua vez, reduz o tamanho da árvore após sua criação inicial, identificando divisões potencialmente prejudiciais ao modelo. A pós-poda tende a ser mais eficaz, porém mais computacionalmente intensiva, pois permite que a árvore seja completamente desenvolvida antes de simplificá-la (Dinov, 2018).

2.2.5.1 *Random Forest*

Random Forest (RF) é um algoritmo baseado na AD. Ele é um algoritmo *ensemble*, ou seja, se utiliza de múltiplas árvores de decisão treinadas com subconjuntos distintos dos dados e variáveis, combinadas por meio de votação por maioria. Como o nome sugere, os classificadores-base em uma RF são ADs. A junção de múltiplas ADs em um mesmo modelo serve para aumentar a robustez e evitar sobreajuste (Igual; SeguÍ, 2017).

Como um exemplo, a Figura abaixo demonstra a aplicação do RF em um banco de dados. Percebe-se que os resultados de cada uma das N ADs difere em alguns nós, uma perspectiva que não seria possível de se obter com apenas uma AD. Seguindo a figura 2.9 para um N = 1000, se 100 ADs classificam o dado como sendo da classe verde e as outras 900 classificam o dado como da classe azul, a classe azul ganha na votação por maioria.

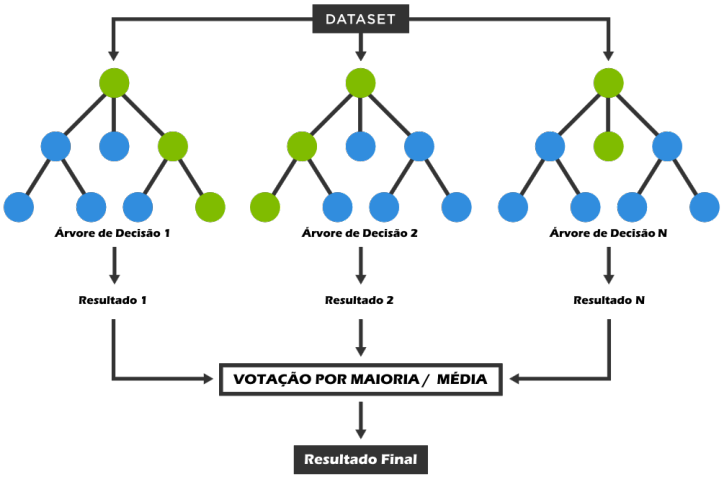


Figura 2.9: Exemplo de um *Random Forest*

2.3 Avaliação de Modelos de ML

A avaliação de modelos de ML depende majoritariamente de métricas relacionadas à sua concordância com o resultado desejado. Para o caso de um problema de classificação, grande parte dessas métricas são calculadas a partir de dois tipos de erro que o modelo pode cometer: Falsos Positivos (FPs) e Falsos Negativos (FNs). A forma mais comum de representar esses erros, juntamente às classificações corretas, ou seja, Verdadeiros Positivos (VPs) e Verdadeiros Negativos (VNs), é a partir do que é denominado matriz de confusão (Figura 2.10) (Ertel, 2017; Kubat, 2017):

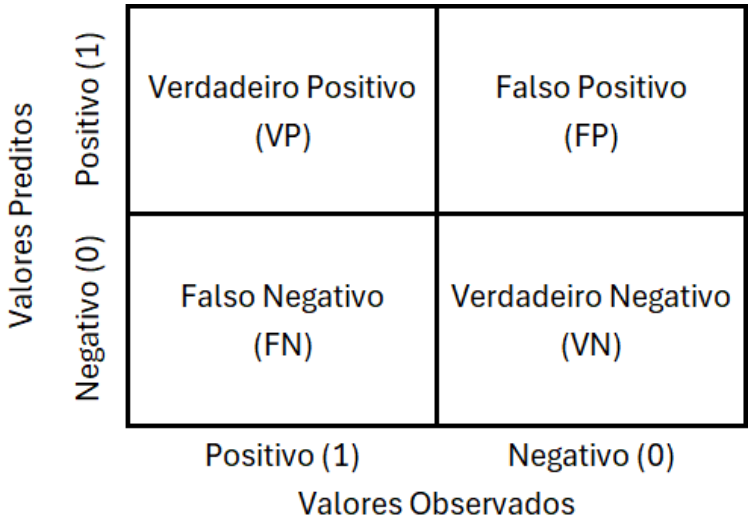


Figura 2.10: Exemplo de uma matriz de confusão

2.3.1 Matriz de Confusão

As métricas que podem ser calculadas a partir da matriz de confusão possuem um nível variado de especificidade. Destas, a mais geral é a acurácia, definida como a probabilidade de uma classificação correta ou, em outras palavras, a frequência de acertos do modelo ao classificar o banco de dados alimentado. Para obter este valor, basta dividir o número de classificações corretas pelo número de dados dentro do banco (Figura 2.10) (Kubat, 2017; Dinov, 2018).

$$Acurácia = \frac{VP + VN}{VP + VN + FP + FN}$$

A precisão e o *recall*, também chamado de sensibilidade, são duas métricas mais específicas e, de certa forma, opostas entre si. A precisão define a probabilidade de o modelo estar correto ao classificar um dado como positivo, enquanto o *recall* define a probabilidade de um dado positivo ser classificado pelo modelo como tal. Enquanto o *recall* mede a razão entre o número de VPs e o número total de dados rotulados como positivo (VP + FN), a precisão mede a razão entre o número de VPs e o número total de dados classificados como positivo (VP + FP). Suas fórmulas estão dispostas abaixo. (Kubat, 2017; Dinov, 2018)

$$precisão = \frac{VP}{VP + FP}$$

$$recall = sensibilidade = \frac{VP}{VP + FN}$$

Vale notar que, apesar de a matriz apresentada na Figura 2.10 levar em consideração um problema de classificação binária, a matriz de confusão pode ser estendida para múltiplas classes, formando uma matriz N $\times$ N, onde N é o número de classes.

Um modelo que possui baixa precisão porém alto *recall* está enviesado para a detecção de FPs, ou seja, apesar de o modelo reconhecer e classificar

dados positivos como tal, classifica errôneamente como positivo um grande conjunto de dados negativos. De forma análoga, um modelo que possui baixo *recall* porém alta precisão está enviesado para a detecção de FNs, ou seja, costuma acertar ao classificar um dado como positivo porém não reconhece uma grande parcela de dados positivos como tal. Essas métricas se tornam mais relevantes em relação à acurácia na medida em que o balanceamento da distribuição dos dados analisados piora. (Kubat, 2017). Ao alimentar com um conjunto de dados no qual 98% dos dados estão rotulados como negativo a um modelo incapaz de classificar dado algum como positivo, a acurácia obtida ainda é de 98%, apesar de tanto a precisão quanto o recall serem zero.

Com a hipótese acima em mente, consideremos como exemplo um estudo estatístico de uma metodologia de detecção de doenças. Na prática, a grande maioria dos testes realizados para a detecção de uma doença retornam negativos. Nesse caso, a acurácia geralmente não é uma boa métrica pelo motivo já citado acima e, consequentemente, por causar um entendimento incorreto do resultado obtido. Para esta situação, a métrica mais útil é a de *recall*, por oferecer uma visão melhor da frequência de falsos negativos, críticas para a contenção e o tratamento da doença avaliada.

Precisão e *recall* têm importâncias diferentes dependendo do sistema analisado e do que se deseja do modelo em questão. O *recall* se torna mais importante uma vez que o contexto do sistema está envolvido em questões críticas de segurança ou saúde. Nesses casos, deseja-se evitar FNs o máximo possível pois eles podem levar a consequências severas, como a falha em diagnosticar uma doença, ou a não-detecção de uma falha crítica em uma fábrica. Já a precisão possui maior importância em contextos onde é necessária uma relação de confiança ou uma operação contínua. Aqui, deseja-se evitar FPs pois eles introduzem obstáculos desnecessários e incerteza, como em sistemas de recomendação de produtos em páginas de *e-commerce*, ou na operação de uma planta industrial (Kubat, 2017).

Precisão e *recall* podem ser consolidadas em uma métrica denominada F_β. Essa métrica permite o usuário impor um peso maior na precisão ou no *recall* a partir do valor da constante β. Quanto maior o seu valor, maior o peso na precisão. $\beta = 0$ reduz a expressão ao valor do *recall*, enquanto um $\beta = 1$ confere peso igual às duas métricas, sendo o valor mais comumente utilizado(Kubat, 2017; Dinov, 2018).

$$F_\beta = \frac{(\beta^2 + 1) \times precisão \times recall}{\beta^2 \times precisão + recall}$$

$$F_1 = \frac{2 \times precisão \times recall}{precisão + recall} = \frac{2VP}{2VP + FP + FN}$$

2.3.2 Curva ROC

Outra métrica interessante provém da curva Característica de Operação do Receptor, do inglês Receiver Operating Characteristic Curve (Curva ROC). Essa curva geralmente é utilizada para analisar a priorização de VPs frente à detecção de FPs, ou vice versa, sendo uma visualização da qualidade da

classificação do modelo e também relacionada à métrica de *recall*.

A curva ROC geralmente é representada em um gráfico unitário onde o eixo X é a taxa de classificação de dados rotulados negativos como positivo e o eixo Y é a taxa de classificação de dados rotulados positivos como positivo. Percebe-se que o eixo Y é equivalente ao *recall*. Cada ponto da curva representa o comportamento do modelo frente a valores crescentes do limiar para o modelo classificar um dado como positivo ou negativo. Como consequência, as curvas ROC sempre começam no ponto (0, 0) e terminam no ponto (1, 1) (Aggarwal, 2015; Ertel, 2017). Exemplos de curvas ROC podem ser vistos na figura 2.11.

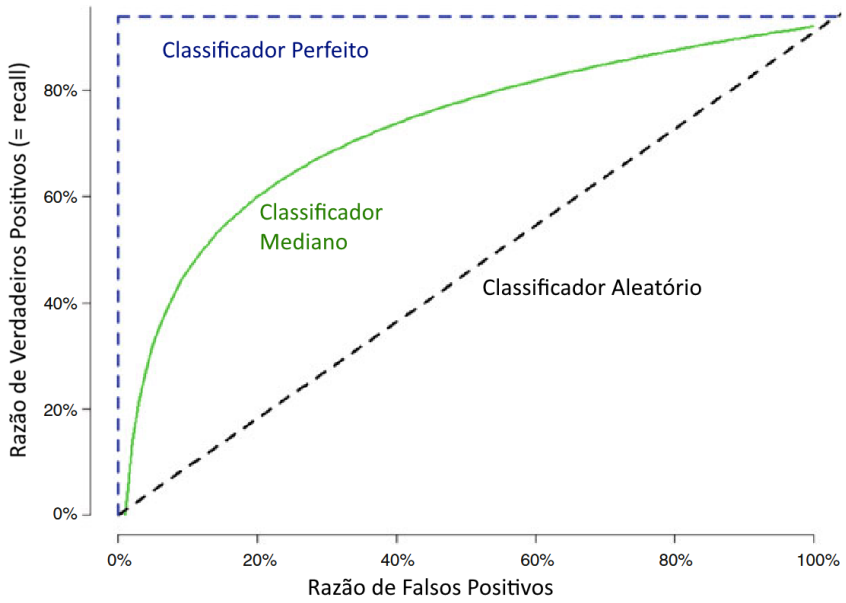


Figura 2.11: Exemplos de curvas ROC para classificadores de diferentes qualidades. Adaptado de (Dinov, 2018)

Um modelo possui bom desempenho quando é capaz de classificar VPs sem classificar quase nenhum FP, ou seja, uma curva ROC que salta do ponto (0, 0) para o ponto (1, 0) é considerada a curva de um classificador perfeito. Assim, na curva ROC, quanto mais perto do ponto (1, 0) a curva se aproxima, melhor o desempenho do modelo. Para quantificar a proximidade da curva ao ponto (1, 0), é utilizada a métrica de área sob a curva ROC, do inglês *Area Under Curve* (AUC). Essa métrica varia de 0.5 a 1 e, quanto mais próxima de 1, mais próximo está o modelo de um classificador perfeito (Skiena, 2017). Uma AUC de 0.5 implica em uma taxa de detecção de VPs e FPs idêntica, por sua vez implicando em um classificador aleatório.

Um outro tipo de curva que pode ser utilizada é a curva Precision-Recall (PvR). A curva PvR é popularmente utilizada no lugar da curva ROC em *datasets* desbalanceados pela crença de que sua AUC é uma métrica superior à AUC de uma curva ROC nestes casos. No entanto, McDermott et al (2025) demonstraram que, em boa parte dos casos onde há desbalanceamento nos dados estudados, a curva ROC continua sendo preferível sobre a curva PvR.

Especificamente para o caso desta dissertação, onde a métrica de *recall* possui grande importância conforme descrito anteriormente e também na seção de Metodologia a seguir, a curva PvR não é uma boa escolha pois prioriza

a maximização de verdadeiros positivos, ao invés da minimização de falsos negativos. Além disso, para o caso de diagnósticos multiclasse, a AUC da curva PvR poderia incentivar mudanças no modelo de forma a favorecer classes desproporcionalmente ao invés de visar uma melhora geral. A curva ROC, por sua vez, oferece um incentivo mais uniforme e mais centralizado no *recall* do que na precisão (Mcdermott et al., 2025).

Apesar de todas estas indagações, na seção de metodologia a seguir também será revelado que, no cenário mais desafiador para os modelos, foi incluída uma etapa envolvendo o balanceamento dos dados utilizados. Dessa forma, grande parte das questões entre a curva ROC e a curva PvR se tornam irrelevantes para os resultados principais obtidos.

3
Metodologia

Este capítulo descreve a metodologia utilizada para explorar diferentes cenários de interesse para a dissertação. O capítulo está estruturado da seguinte forma: primeiro, o banco de dados utilizado é descrito, bem como o sistema avaliado percentence ao banco. Em seguida, os cenários explorados serão explicados em ordem cronológica. Finalmente, as métricas de avaliação utilizadas serão apresentadas.

Os procedimentos para cada cenário foram realizados utilizando Python 3.11.4 através do pacote *scikit-learn* (*sklearn*) e das bibliotecas *pandas* (pd), *numpy* (np), *seaborn* (sns), *prince* e *matplotlib*.

3.1
Descrição do sistema

Todos os dados utilizados neste trabalho pertencem à parcela real de dados do 3W *dataset*, disponibilizado publicamente por Vargas et al. (2019). Ele contém informações de monitoramento em tempo real de mais de vinte poços de petróleo *offshore* da Petrobras, totalizando por volta de 10 milhões de vetores de dados coletados. As parcelas de dados simulados e desenhados por especialistas no ramo não serão utilizadas.

O 3W *dataset* é um compilado de instâncias do funcionamento dos poços, agrupados pelo tipo de anomalia ocorrido. Instâncias que apresentam apenas operação normal, sem falhas, são agrupadas na classe 0. Devido à arquitetura do 3W *dataset*, as instâncias que apresentam anomalias também contêm pequenas parcelas de dados da classe 0, que servirão como grupo de controle nos cenários.

Cada instância divide a classe de anomalia relatada em duas: anomalia transiente, representada pelo número 10X, e anomalia estável, representada pelo número X, onde X é o número da classe da anomalia de acordo com a lista abaixo.

1. Aumento de impurezas (*Basic Sediment and Water*, BSW) no petróleo;
2. Fechamento espúrio da DHSV (*DownHole Safety Valve*);
3. Intermitência severa;
4. Fluxo instável;
5. Perda de produtividade ou perda de surgência do poço;
6. Fechamento errôneo ou acidental da válvula CKP (*ChoKe* de Produção);
7. Inscrustações na válvula CKP;
8. Formação de hidrato em linha de *gas lift*.

O estado de anomalia transiente é definido como sendo o espaço de tempo no qual as dinâmicas resultantes da anomalia ainda estão se estabilizando no processo e podem ser interpretadas como um período pré-anomalia, após o qual o processo entra novamente em um estado estável anômalo (Vargas et al., 2019). Cada linha no dataset possui a data e hora dos dados coletados, a classe auferida e dados de oito sensores disponíveis em cada poço:

- P-PDG - Pressão no *Permanent Downhole Gauge* (PDG);
- T-TPT - Temperatura no Transdutor de Pressão e Temperatura (TPT);
- P-TPT - Pressão no TPT;
- P-MON-CKP - Pressão à MONtante do CKP;
- T-JUS-CKP - Temperatura à JUSante do CKP;
- P-JUS-CKGL - Pressão à JUSante do ChoKe de Gas Lift (CKGL);
- T-JUS-CKGL - Temperatura à JUSante do CKGL;
- QGL - Vazão, Q, de *Gas Lift*.

Os últimos três sensores dizem respeito ao procedimento de *gas lift*, que pode ser realizado em alguns dos poços presentes no banco de dados.

A Figura 2.1 representa o processo de extração de petróleo ao qual pertence o 3W dataset. Conforme demarcado, abaixo da árvore de natal estão localizadas a válvula de segurança DHSV e o PDG, enquanto o TPT se encontra dentro da árvore. O resto dos instrumentos se encontram na superfície, onde o a tubulação umbilical encontra a tubulação de produção da plataforma fixa ou FPSO, bem como seus elementos finais de controle (CKP e CKGL). As variáveis do processo são os sensores destes instrumentos, listados acima.

A Figura 3.1 demonstra o comportamento de uma das instâncias do 3W *dataset*. Neste caso está sendo observada a variável T-JUS-CKP para a anomalia de classe 1. Percebe-se que o sistema está no estado estacionário desejável até o momento em que a anomalia acontece, momento a partir do qual a variável entra em um estado transiente pelo tempo necessário até que se chegue a um novo estado estacionário, este desfavorável para a operação.

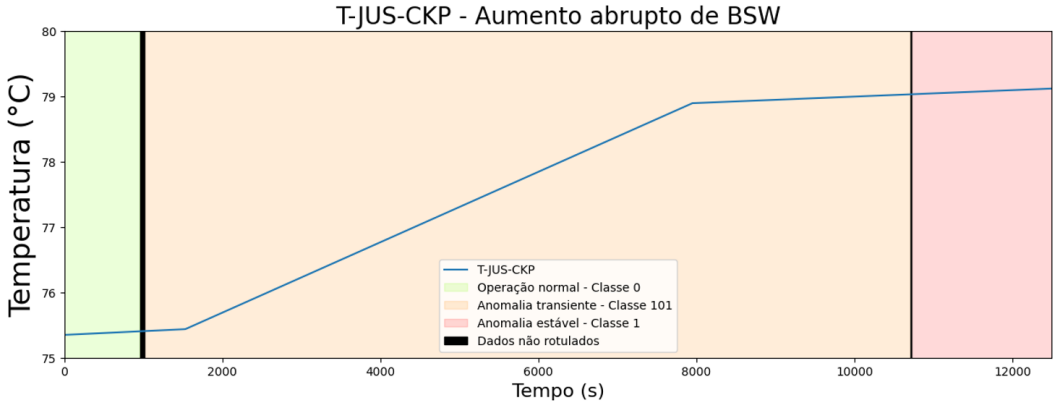


Figura 3.1: Visualização do comportamento da variável T-JUS-CKP frente a uma anomalia de classe 1.

3.2 Cenário 1

Neste primeiro cenário, deseja-se compreender a estrutura e o comportamento dos dados, além do desempenho dos modelos treinados frente à modificação do *dataset* estudado. Reduzir a dimensionalidade de um banco de dados desta extensão pode resultar em uma grande redução na demanda computacional para treinar os modelos de ML, porém isso deve ser feito sem prejudicar a qualidade dos dados e, consequentemente, o desempenho dos modelos. Para esse cenário, um *subdataset* será criado utilizando apenas os dados pertencentes à instâncias reais do 3W *dataset* que apresentam anomalias de classe 1.

A primeira etapa nesse cenário foi o pré-processamento dos dados, de forma a lidar majoritariamente com conjuntos de pontos flagrantemente inválidos e ausentes. O objetivo principal dessa etapa é ajustar e ordernar o *subdataset* de forma a possibilitar a aplicação das técnicas de interesse.

Uma matriz de correlação foi criada a partir do *subdataset* resultante para análise de variáveis similares. Em seguida, foram aplicados algumas técnicas de redução de dimensionalidade: APC, uma técnica de FE, e MI, uma técnica de FS.

Para a aplicação da APC conforme descrito por Abdi e Williams (2010), o *subdataset* foi padronizado e o método foi aplicado através da biblioteca *prince* (prince.PCA.fit). A biblioteca também gera tabelas de contribuição dos componentes principais (CPs) e de correlação entre as variáveis de processo e cada CP, que foram utilizadas.

Uma modificação foi realizada ao final da aplicação da APC, na qual um valor mínimo de correlação (0,75, nesse caso) entre as CPs consideradas e as variáveis de processo foi definido para a inclusão ou não das variáveis no *subdataset* final, abandonando os CPs ao final deste processo. Isso foi feito pois, se as variáveis possuem altos níveis de correlação com as CPs, elas podem continuar sendo utilizadas, preservando a contextualização do problema em mãos. Dessa forma, a aplicação da APC se assemelha mais a uma técnica de FS do que de FE. O método MI, por sua vez, foi aplicado através da função SelectKBest do pacote *sklearn*, com k=3.

Os *subdatasets* resultantes foram então normalizados, divididos em uma razão treino:validação:teste de 70:15:15 e modelos de Regressão Logística (RLog) foram treinados e testados utilizando o *sklearn*, de acordo com a sua capacidade de detectar anomalias de classe 1.

3.3 Cenário 2

O Cenário 2 é mais simples e direto, visando comparar o desempenho entre modelos de ML mais simples, como a RLog, a modelos mais aprofundados, baseados em RNAs e ADs. Além disso, explorar este cenário é essencial para definir as melhores condições de desempenho para os modelos de ML, a partir do conhecimento obtido no Cenário 1. Para esse cenário, um *subdataset* será criado utilizando apenas os dados pertencentes à instâncias reais do 3W *dataset* que apresentam anomalias de classe 1 e 3.

De forma similar ao primeiro cenário, na primeira etapa foi realizado o pré-processamento dos dados, com princípios e objetivos similares: organizar o *subdataset* de forma que não haja presença de dados inválidos e ausentes para possibilitar a aplicação dos algoritmos de ML. Em seguida, RF foi aplicado como um método de FS através do *sklearn* (RFC), ou seja, modelos de AD foram rodados com diferentes subconjuntos das variáveis disponíveis de forma a classificar e selecionar as variáveis mais relevantes ao problema em questão. Por fim, o *subdataset* resultante foi dividido em uma razão treino:teste de 70:30. Por fim, modelos de RLog, MLP e DT foram treinados e seus desempenhos foram avaliados de acordo com a capacidade de diferenciar anomalias de classe 1 e 3 de instâncias de operação normal.

3.4 Cenário 3

Este cenário tem como objetivo juntar o conhecimento adquirido até então para criar um ambiente especificamente para a avaliação dos modelos de ML a serem treinados. Todo o conhecimento obtido sobre a qualidade dos dados, a melhor forma de se processar o *dataset*, os modelos que oferecem maior potencial e, sobretudo, como criar uma situação inesperada para o modelo treinado, será utilizado no Cenário 3 para avaliar modelos de ML a partir de uma perspectiva ampla e imparcial. Para esse cenário, um *subdataset* será criado utilizando apenas os dados pertencentes à instâncias reais do 3W *dataset* que apresentam anomalias de classe 1, 2, 6, 7 e 8.

Para o pré-processamento do *subdataset* gerado nesse cenário, o objetivo principal foi manter o maior número possível de dados válidos, além de evitar a remoção completa de dados de um ou mais poços dentro do *subdataset*. Para isso, duas características importantes foram analisadas: o número de pontos inválidos, ausentes ou congelados em cada variável, bem como a possibilidade de estimar estes pontos baseado no contexto do sistema. Isso evita a exclusão de linhas do conjunto de dados devido a um único ponto inválido de um dos sensores, o que, por sua vez, evita a remoção de dados potencialmente relevantes de outros instrumentos além do que está apresentando a falha na mesma linha, preservando a qualidade dos dados e melhorando o desempenho do modelo.

A finalização deste passo consiste na remoção de todos os dados de anomalia estável do *subdataset* para armazenamento em outro conjunto de dados para uso posterior. Dessa forma, os dados do *subdataset* resultante estão prontos para o treino e teste os modelos.

O *subdataset* foi dividido em uma razão treino:teste de 70:30 utilizando o *sklearn* e uma metodologia em duas etapas foi criada para avaliar dois modelos de ML: Multi-Layer Perceptron (MLP), um algoritmo baseado em RNA, e Árvore de Decisão (AD). Para a primeira etapa, o subconjunto de dados foi simplificado de forma a representar todas as anomalias por uma classe arbitrariamente definida como 1, e os modelos foram treinados e testados com base em suas capacidades de detecção de falhas. Seguindo para a segunda etapa, todos os dados de classe 0 foram descartados, as classes originais das anomalias foram restauradas e novos modelos foram treinados e testados para avaliar a capacidade de diagnóstico de falhas. Dessa forma, tanto o DT quanto

o MLP possuem um conjunto de modelos que lidam separadamente com a detecção e o diagnóstico de falhas. Os modelos foram treinados usando a função GridSearchCV do *sklearn* com k=5 e critérios conforme descrito nas Tabelas 3.1 e 3.2.

Modelo	alpha	hidden_layer_sizes	learning_rate_init
Detecção de Falhas	0,00012, 0,0001, 0,00008	(20,), (12,8), (9,5,3)	0,01, 0,007, 0,005
Diagnóstico de Falhas	0,0001, 0,0003, 0,0005	(4,), (6,), (8,), (4,2), (4,4)	0,003, 0,005, 0,007

Tabela 3.1: Hiperparâmetros do GridSearchCV para encontrar o melhor ajuste de MLP

max_depth	Criterion	class_weight
1, 2, ..., 10	gini, entropy, log_loss	None, balanced

Tabela 3.2: Hiperparâmetros do GridSearchCV para encontrar o melhor ajuste de AD

Após a finalização e teste dos modelos, os dados de anomalias em estado estável previamente removidos foram unidos com uma amostra aleatória do conjunto de dados da classe 0 e alimentados aos modelos para uma etapa final de validação de forma semelhante: primeiro, os dados foram simplificados em um conjunto de dados de detecção de falhas para a primeira etapa e, em seguida, os dados da classe 0 foram desconsiderados e as classes originais das anomalias foram recuperadas para a segunda etapa. Esta é uma camada adicional de precaução contra a possibilidade de sobreajuste, bem como um teste de robustez, já que os dados de anomalias em estado estável representam dados fora da perspectiva de treino do modelo, simulando então uma situação real. A Figura 3.2 mostra um esquema do cenário realizado.

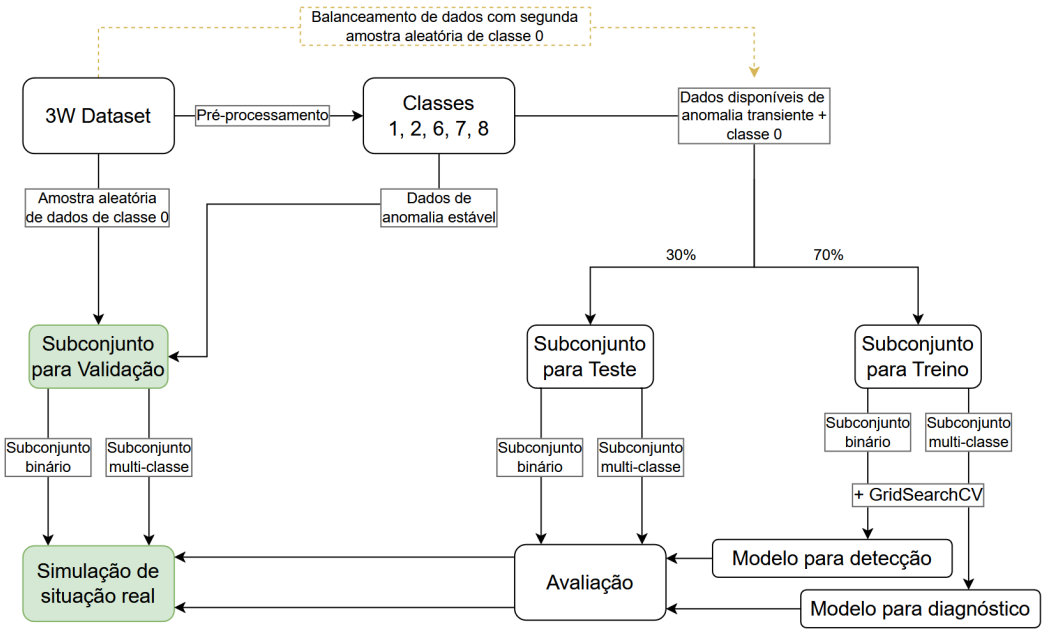


Figura 3.2: Visualização da metologia proposta para o Cenário 3.

A partir da seleção dos dados, pode-se supor que o *subdataset* usado para treinar os modelos possui uma representatividade desbalanceada entre dados que apresentam condições anômalas e condições normais de operação. Como medida para criar um melhor equilíbrio entre estes dados, uma nova amostra aleatória do conjunto de dados da classe 0 foi adicionada ao *subdataset* e, em seguida, o programa acima foi executado novamente.

3.5 Métricas de Avaliação

Os modelos foram avaliados principalmente através da geração de matrizes de confusão, conforme a função *confusion_matrix* no pacote *sklearn*, que foram apresentadas através da função sns.heatmap. Dessa forma, diversas métricas podem ser avaliadas, como precisão, *recall*, F_1 e acurácia. Destas, o *recall* e o F_1 são as métricas mais interessantes, uma vez que estão relacionadas à falsos negativos e, consequentemente, ligados a uma das principais preocupações do estudo.

Os modelos também foram avaliados através da geração das suas curvas ROC após o ajuste aos bancos de dados analisados. Apesar de as curvas ROC medirem primariamente o desempenho de classificadores binários, a criação de um conjunto de curvas ROC no estilo One-versus-Rest (OvR) é viável para a avaliação de classificadores multivariável, caso dos modelos de diagnóstico nos cenários 2 e 3. As AUCs foram calculadas e utilizadas para avaliar o desempenho geral dos modelos.

Para todos os cenários, as matrizes de confusão e curvas ROC para os *datasets* de treino não serão apresentados. Como estes *datasets* são justamente os conjuntos de dados utilizados para o ajuste dos modelos, as métricas resultantes podem oferecer perspectivas incorretas sobre o desempenho do modelo. Assim, para os cenários 1 e 2, os resultados expostos dizem respeito ao desempenho do modelo no *dataset* de teste. Como o Cenário 3 possui um nível adicional de complexidade, os resultados serão apresentados detalhando qual etapa está sendo abordada.

A partir da metodologia descrita nesse capítulo, planejada para evoluir em sofisticação e realismo com cada cenário explorado, os algoritmos de ML serão aplicados de forma estruturada, culminando na avaliação da robustez dos modelos em cenários operacionais simulando um ambiente industrial de alta complexidade. Os resultados obtidos serão detalhados a seguir, onde o desempenho dos modelos frente aos desafios impostos será confrontado com as métricas definidas neste capítulo.

4
Resultados e Discussão

4.1
Cenário 1

4.1.1
Pré-processamento dos dados

O *subdataset* analisado contém cinco instâncias de anomalias de classe 1, com um total de 118294 linhas e 8 variáveis. Durante a análise de dados do *subdataset*, foi observado que, em algumas instâncias, um ou mais instrumentos não estavam funcionando ou não estavam em uso. Diante dessa observação, três variáveis foram descartadas: duas delas por excesso de valores inválidos, sendo elas P-PDG (> 34 % dos dados) e T-JUS-CKGL (100 % dos dados), e uma por não possuir nenhum valor diferente de zero, QGL.

Como o sensor de QGL está diretamente relacionado à realização de Gas Lift, a falta de atividade detectada pelo sensor demonstra que esse procedimento não foi realizado. Isso tem implicações diretas em outros dados, servindo como outra razão para o descarte de T-JUS-CKGL, além de ser um motivador para o descarte de P-JUS-CKGL. Entretanto, essa variável possui valores válidos, diferentes de 0 e não-congelados em algumas instâncias do banco de dados. Como o objetivo dessa etapa é simplesmente lidar com dados inválidos e preparar o *subdataset* para aplicação das técnicas de FS e FE, P-JUS-CKGL foi mantida por enquanto.

Ademais, uma pequena parcela de dados na interface entre duas classificações diferentes não possui uma classe atribuída. Por não influenciarem significativamente na qualidade do *subdataset*, esses dados foram removidos, resultando em um *dataset* com 117275 linhas e 5 variáveis.

4.1.2
Matriz de Correlação

A matriz de correlação de Spearman foi gerada com o *subdataset* pré-processado, conforme disposto na Figura 4.1. Preferiu-se a utilização da correlação de Spearman sobre Kendall e Pearson por dois motivos. Primeiro, a distribuição dos dados da base não é normal, premissa necessária para a aplicação da correlação de Pearson. Segundo, as variáveis estudadas contêm valores repetidos e são puramente cardinais, enquanto o coeficiente de correlação de Kendall foi criado para distribuições de dados não-vinculados e ordinais, logo sua aplicação também é prejudicada.

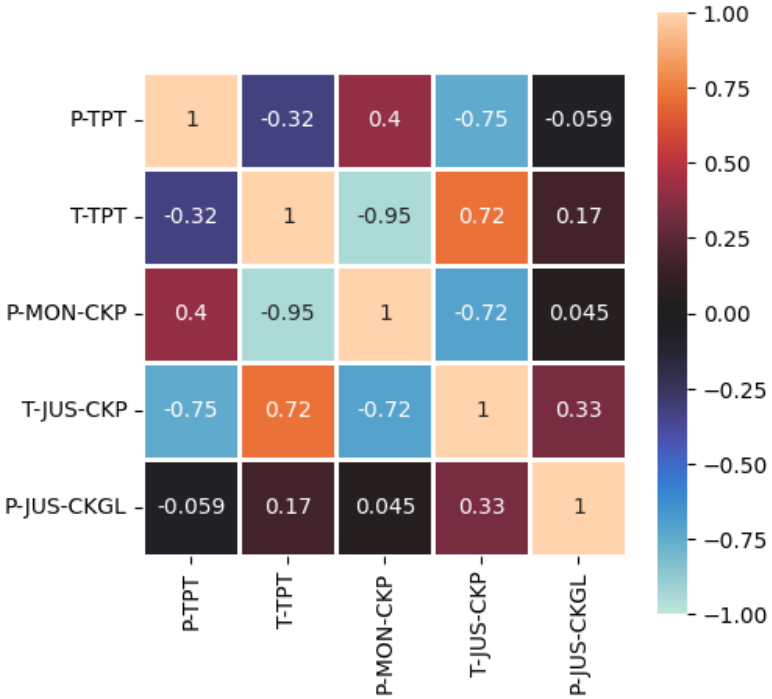


Figura 4.1: Matriz de correlação de Spearman para o *subdataset* pré-processado

Alguns pontos podem ser destacados nessa matriz, como a forte correlação negativa entre P-MON-CKP e T-TPT (-0,95) e as correlações elevadas de T-JUS-CKP com todas as outras variáveis exceto P-JUS-CKGL, que não possui correlação significativa com nenhuma outra variável.

Devido à grande distância que separa P-MON-CKP e T-TPT, a forte correlação entre essas duas variáveis é interessante. Sendo um indicador de proporcionalidade inversa, é uma porta para um modelo capaz de se antecipar a alguma variação propagada para P-MON-CKP a partir do que aconteça na árvore de natal, onde T-TPT está localizado. Essa interação pode beneficiar a detecção de anomalias das classes 1, 2 e 5, por estarem relacionadas a variações na árvore de natal.

P-JUS-CKGL, por sua vez, possui correlações próximas a zero com todas as outras variáveis. Esse é, potencialmente, outro fator favorável para seu descarte do *subdataset*, no entanto a etapa atual visa somente entender melhor como os dados se correlacionam. Os métodos de redução de dimensionalidade foram aplicados a partir da seção seguinte.

4.1.3 Análise de Componentes Principais

Para o ACP realizado foram utilizados os mesmos dados da matriz de correlação, dessa vez padronizados em relação às médias e variâncias de cada variável. Esse passo é necessário pois, sendo um método baseado na dispersão do sistema, o ACP é tendencioso para dados de maior magnitude. A padronização evita com que esses dados, geralmente com maior variância absoluta, dominem o resultado da análise.

Na Tabela 4.1, nota-se que os três primeiros componentes principais (CPs) juntos explicam mais de 98% da variância total dos dados, logo esse será o número de componentes considerados.

Componente Principal	Autovalor	Variância explicada	Variância explicada (acumulada)
1	2,195	43,89 %	43,89 %
2	1,844	36,89 %	80,78 %
3	0,870	17,41 %	98,19 %
4	0,082	1,64 %	99,83 %
5	0,009	0,17 %	100,00 %

Tabela 4.1: CPs ranqueados por porcentagem de variância explicada

Pela tabela, os CPs com a maior contribuição estão ordenados. CP1 explica 43,89 % da variância no banco de dados e CP2 explica 36,89 %, apenas 7 % a menos do que CP1, indicando uma complexidade relativa. CP3, o último CP considerado, possui explicabilidade de 17,41 %.

Considerando a modificação da técnica descrita na metodologia, A Tabela 4.2 conecta a associação entre cada CP e cada variável de processo através de uma tabela de correlações (ρ). Nesse caso, ρ é proporcional ao nível de informações contidas na variável relacionadas a um CP, o que impacta diretamente na significância da variável.

Variável	CP1	CP2	CP3	CP4	CP5
P-MON-CKP	0,948	0,214	0,107	-0,209	0,013
T-TPT	-0,808	-0,192	-0,533	-0,160	0,020
P-JUS-CKGL	0,413	0,638	-0,647	0,042	-0,041
T-JUS-CKP	-0,211	0,974	0,038	0,048	0,061
P-TPT	0,654	-0,637	-0,393	0,094	0,051

Tabela 4.2: Matriz de correlações entre os CPs gerados e as variáveis do *subdataset*

Utilizando como valor de corte $|\rho| = 0{,}75$ para as correlações e considerando os três primeiros CPs conforme visto na Tabela 4.2, percebe-se que P-JUS-CKGL e P-TPT podem ser eliminados. Voltando à Figura 4.1, percebe-se também que essas duas variáveis possuem as correlações mais fracas na matriz de correlação, tanto entre si quanto com as outras variáveis.

4.1.4 Mutual Information

A Figura 4.2 abaixo mostra o resultado da aplicação do MI ao *subdataset* pré-processado. As três variáveis com maior pontuação foram P-MON-CKP, T-TPT e P-TPT, logo elas serão mantidas para a próxima parte do cenário.

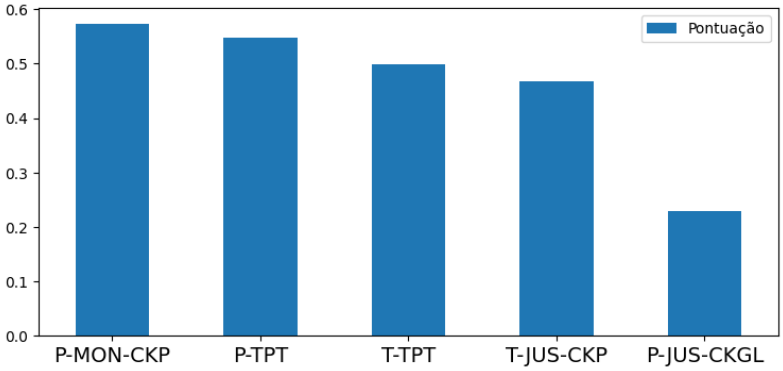


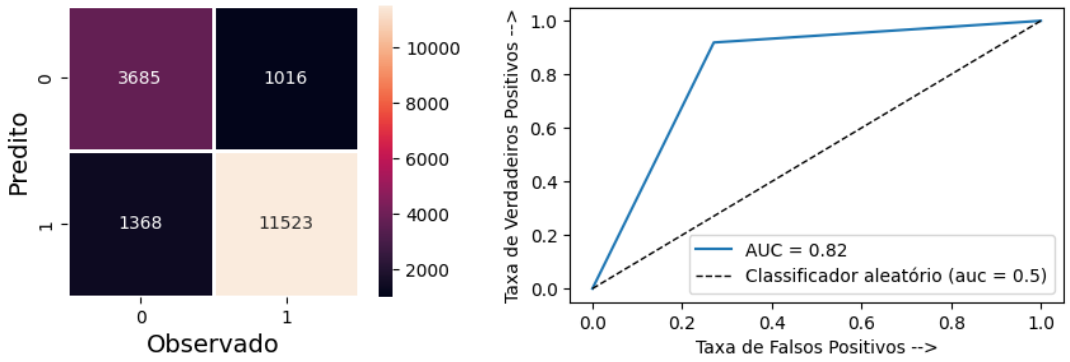
Figura 4.2: Resultados da aplicação do MI nas variáveis do *subdataset* pré-processado

Como esperado, mesmo mantendo P-JUS-CKGL após as discussões nas seções anteriores, ela foi eliminada tanto no ACP quanto no MI, demonstrando que, de fato, essa variável não possui informação relevante o bastante para a modelagem.

Uma diferença notável entre as duas técnicas é a troca de T-JUS-CKP por P-TPT. É importante lembrar que a APC modificada leva em consideração as variáveis que estão mais proximamente relacionadas a 3 dos 5 CPs gerados, sendo uma comparação entre variáveis de processo e um conjunto de componentes ortogonais (ou seja, sem correlação alguma) entre si. O MI, por sua vez, compara variáveis de processo a si mesmas, que dependem umas das outras, o que pode levar a resultados diferentes.

4.1.5
Modelagem e Desempenho

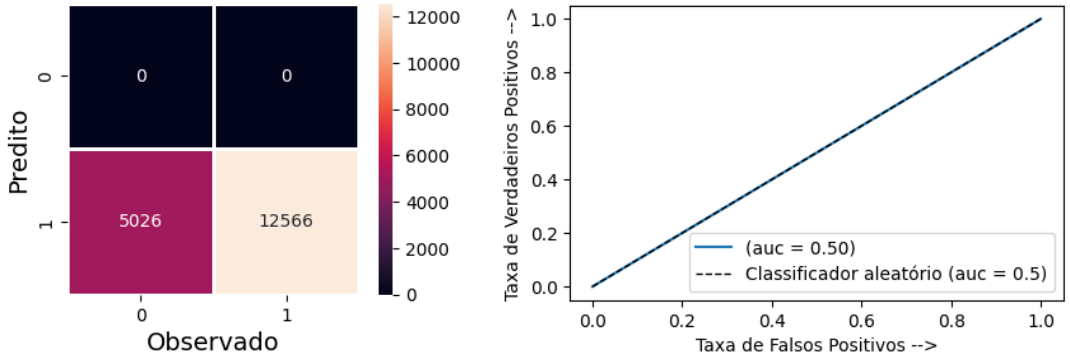
As figuras 4.3, 4.4 e 4.5 abaixo expõem as matrizes de confusão e as curvas ROC dos modelos de RLog gerados para cada configuração analisada:



(a) Matriz de confusão

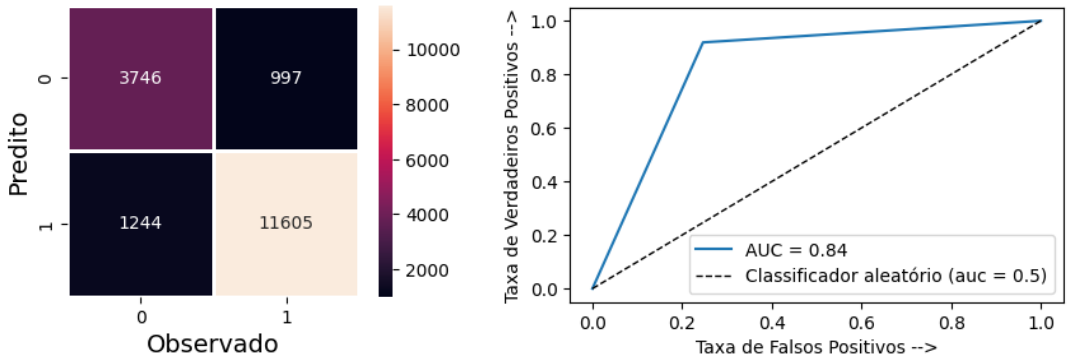
(b) Curva ROC

Figura 4.3: Resultados do ajuste do modelo de RLog ao *dataset* sem alterações.



(a) Matriz de confusão (b) Curva ROC

Figura 4.4: Resultados do ajuste do modelo de RLog ao *dataset* após processamento dos dados por APC



(a) Matriz de confusão (b) Curva ROC

Figura 4.5: Resultados do ajuste do modelo de RLog ao *dataset* após processamento dos dados por MI

Comparado ao desempenho do modelo gerado para o *subdataset* sem alterações, a matriz de confusão para o *subdataset* processado pelo MI produziu resultados similares, o que implica na manutenção de todas as métricas avaliadas, confirmado pela Tabela 4.3. Mais especificamente, o modelo gerado após a aplicação do MI apresentou acurácia (0,87) e AUC (0,84) levemente superiores aos do modelo para o *subdataset* sem alterações (0,86 e 0,82). Isso indica que o processamento resultante da técnica de MI ao menos manteve a qualidade dos dados analisados.

Além disso, percebe-se que a matriz de confusão no *subdataset* processado pelo MI possui melhor equilíbrio, com uma razão FP/FN de 1,25 contra 1,35 para o *subdataset*. Isso, no entanto, não será necessariamente um ponto forte nos próximos cenários, dado o objetivo de criar modelos que, além de oferecer bom desempenho, sejam seguros de se utilizar.

Métrica	Nenhum	MI	APC
Acurácia	0,86	0,87	0,71
AUC	0,82	0,84	0,5

Tabela 4.3: Métricas de desempenho do ajuste do modelo de RLog para os *subdatasets* após a aplicação das técnicas propostas de processamento de dados

O modelo gerado para o *subdataset* configurado pela APC, por sua vez, mostrou-se incapaz de detectar níveis normais de operação dos poços, classificando todas as linhas analisadas como classe 1, resultando em uma acurária ce 0,71, ou seja, a representatividade dos dados de classe 1 dentro do *subdataset*. Pode-se observar na Figura 4.4b que a curva ROC nesse caso é a reta x=y, demonstrando a incapacidade de funcionamento do modelo e indicando uma AUC de 0,5. Na visão do projeto, apenas um modelo que classificaria todas as observações como classe 0 teria um desempenho inferior.

Sendo assim, conclui-se que, em um cenário preliminar, técnicas de FS podem ser aplicadas com sucesso no 3W *dataset* para classificadores binários. Essa é uma conclusão importante, pois a redução da demanda computacional para o ajuste de modelos de ML pode ser crucial para a sua viabilização. No entanto, nem todas as técnicas funcionam, como pôde ser observado nos resultados do modelo ajustado pelo *subdataset* configurado pela APC.

4.2 Cenário 2

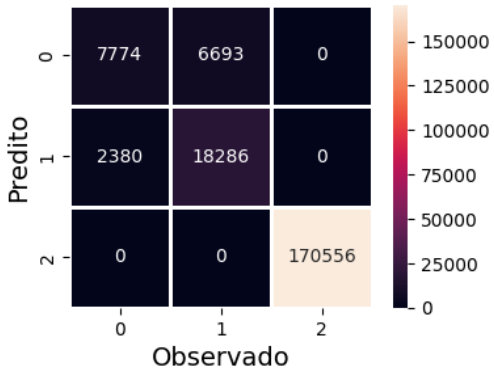
4.2.1 Processamento de dados

O *subdataset* analisado neste cenário contem as instâncias de classe 1 já exploradas pelo Cenário 1, adicionadas às instâncias de classe 3 disponíveis, totalizando 687446 linhas e 8 variáveis. Conforme discutido no Cenário 1, T-JUS-CKGL não possui dados, então foi descartado. No entanto, a decisão tomada foi diferente para P-PDG e QGL: os valores de P-PDG foram igualados a 0 e a ambas as variáveis foram mantidas. A razão para tal vem da expectativa de que as variáveis serão descartadas durante o FS, servindo como um teste sanitário tanto das técnicas utilizadas quanto dos modelos a serem treinados.

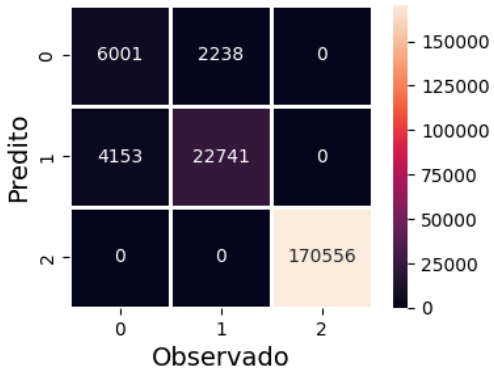
Assim como no Cenário 1, uma pequena parcela de dados não estavam rotulados e foram descartados, bem como quaisquer outros dados inválidos dispersos pelo *subdataset*, resultando em uma contagem de 685627 linhas e 7 variáveis. A aplicação do RFC resultou na remoção de três variaveis, sendo elas: QGL, P-PDG e P-MON-CKP. Nota-se que as duas variáveis previamente mantidas apesar de problemáticas foram removidas durante o FS. Assim, a contagem final deste *subdataset* é de 685627 linhas e 4 variáveis.

4.2.2
Modelagem e Desempenho

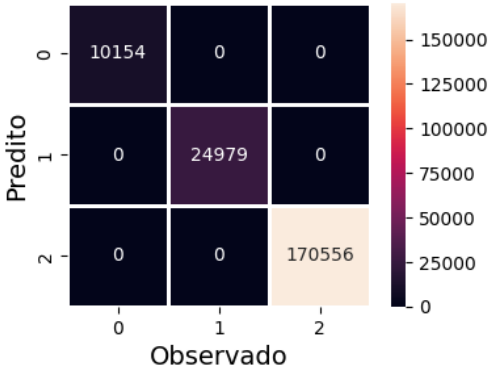
As matrizes de confusão, curvas ROC e demais métricas de desempenho dos modelos gerados neste cenário estão dispostas nas Figuras 4.6 e 4.7:



(a) RLog

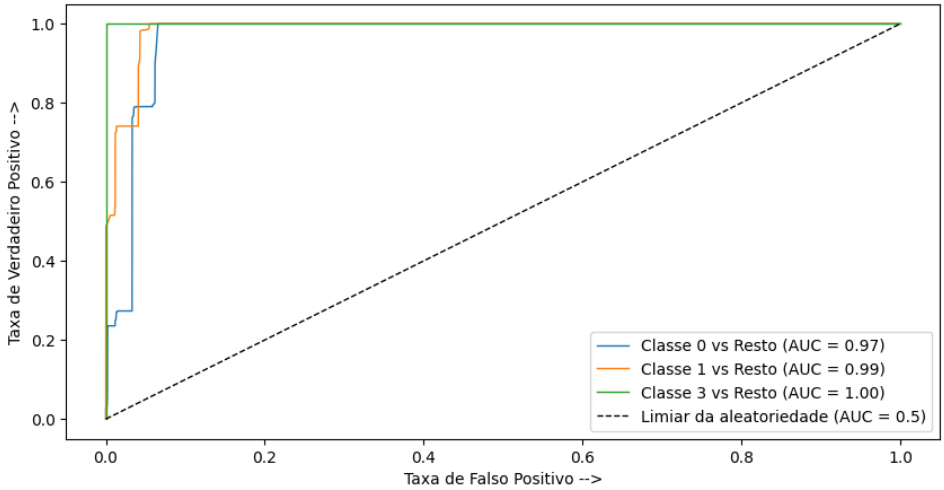


(b) MLP

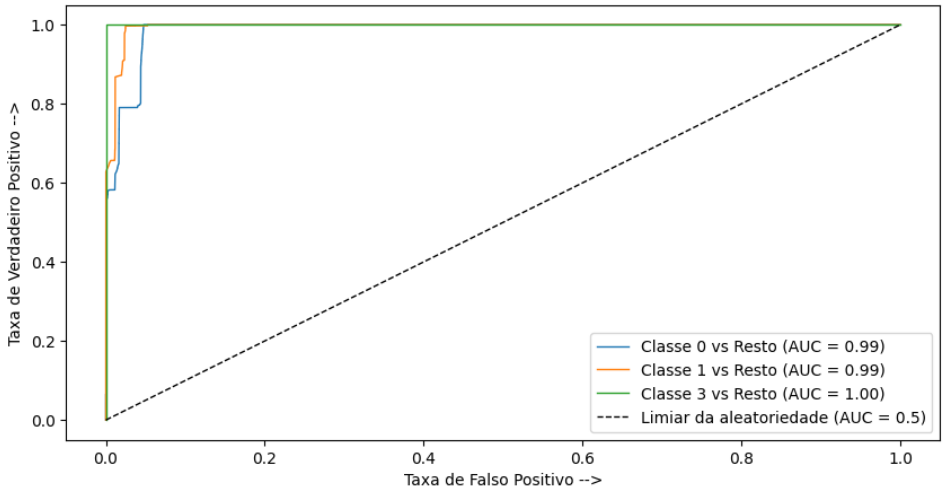


(c) DT

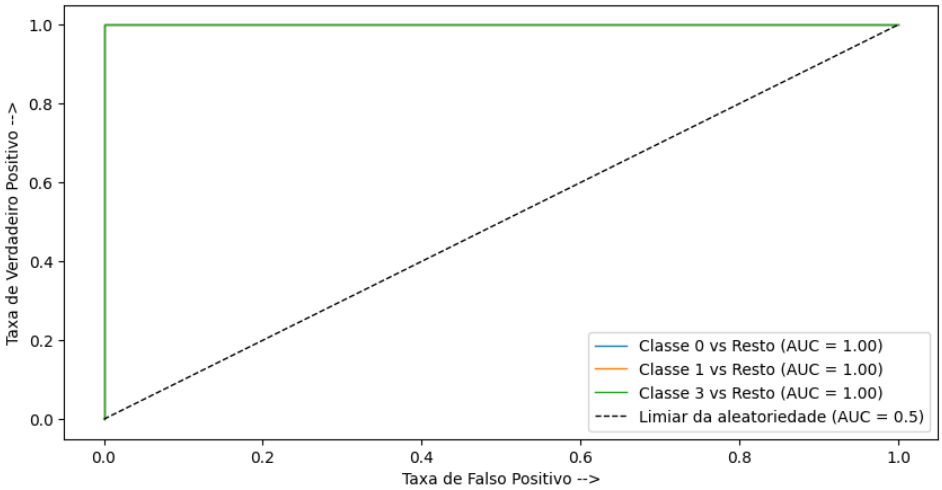
Figura 4.6: Matrizes de confusão dos modelos ajustados ao *subdataset* de treino após a aplicação da técnica de FS por RFC



(a) RLog



(b) MLP



(c) DT

Figura 4.7: Curvas ROC dos modelos ajustados ao *subdataset* de treino após a aplicação da técnica de FS por RFC

Métrica	RLog	MLP	AD
Precisão	0,81	0,86	1,00
Recall	0,83	0,83	1,00
F_1	0,81	0,84	1,00
Acurácia	0,96	0,97	1,00

Tabela 4.4: Métricas de desempenho dos modelos ajustados ao *subdataset* de treino após a aplicação da técnica de FS por RFC

Nota-se que todos os modelos, inclusive o RLog, foram capazes de diferenciar eventos de classe 3 perfeitamente das outras classes analisadas. Esse resultado implica que eventos de classe 3 são facilmente indentificáveis, o que faz sentido, já que a classe 3 se trata de eventos de intermitência severa, com periodicidade bem definida e amplitude notável. Além disso, o modelo de AD obteve desempenho perfeito ao diagnosticar tanto eventos de classe 1 quanto de classe 3 e diferenciá-los de eventos de operação normal, ressaltando a compatibilidade do algoritmo com o 3W *dataset*.

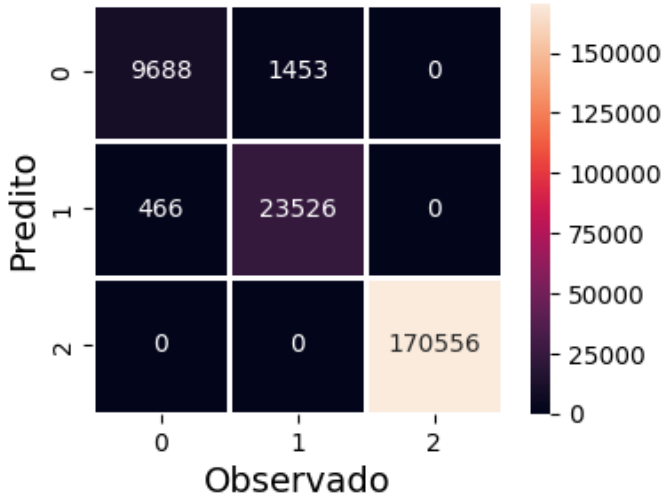
O modelo de RLog obteve o pior desempenho dos três, demonstrando que modelos relacionados a um aprendizado mais profundo, como o MLP e o DT, são mais apropriados para o nível de complexidade do 3W *dataset*. Apesar disso, o modelo de RLog também aparenta apresentar valores surpreendentes de acurácia e AUC, todos acima de 0,95.

Ao observar a Figura 4.6 novamente, percebe-se que o *subdataset* resultante está significativamente desbalanceado, com uma representatividade muito maior de eventos da classe 3 do que qualquer outro evento (4/5 dos dados), justamente a classe mais fácil de se identificar. Isso quer dizer que até mesmo um modelo que classifica todos os dados como classe 3 teria uma acurácia de 80% apesar de ser incapaz de desempenhar seu papel como um classificador. Daí, os valores inflados de acurácia não só do RLog, como do MLP também. A métrica de AUC também é afetada de forma similar, sendo enviesada por *datasets* desbalanceados.

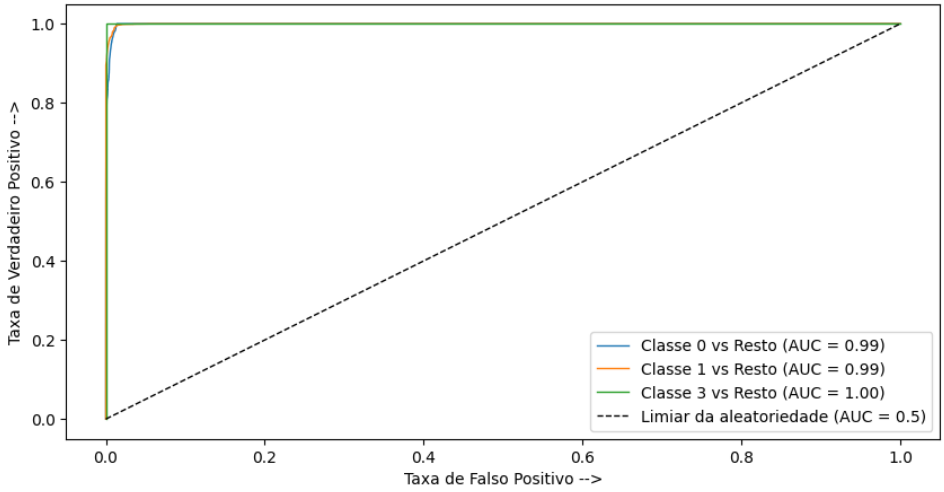
Observando a tabela 4.5, o modelo de MLP não se sobressaiu de forma convincente ao modelo de RLog, com uma precisão, *recall* e F_1 de 0,86, 0,83 e 0,84, contra 0,81, 0,83 e 0,81 do RLog. A Tabela 4.5 e a Figura 4.8 abaixo ilustram o ajuste de um novo modelo de MLP para o *subdataset* sem alterações por técnicas de FS, para motivos de comparação.

Métrica	Nenhum	RFC
Precisão	0,95	0,86
Recall	0,97	0,83
F_1	0,96	0,84
Acurácia	0,99	0,97

Tabela 4.5: Métricas de desempenho do modelo de MLP ajustado ao *subdataset* de treino sem alterações em comparação ao modelo previamente ajustado



(a) Matriz de confusão



(b) Curva ROC

Figura 4.8: Resultados do modelo de MLP ajustado ao *subdataset* de treino sem alterações

A melhora no desempenho do modelo é notável, agora apresentando valores acima de 0,95 em todas as métricas. A acurácia e a AUC não sofreram grandes mudanças por já estarem infladas pelos dados classificados corretamente como de classe 3, porém as métricas de precisão, *recall* e F_1 subiram mais de um ponto decimal em média, agora em 0,95, 0,97 e 0,96.

Em suma, os modelos de ML conseguem desempenhar bem em *datasets* desbalanceados, porém a aplicação de técnicas de FS se torna menos viável e as métricas de desempenho precisam ser interpretadas com mais cuidado. Modelos de ML mais avançados ou de *deep learning* aparentam ser mais compatíveis ao 3W *dataset* e seu elevado nível de complexidade. Além disso, visto a dificuldade de dois dos modelos de diferenciar entre a operação normal de um poço e certas classes de anomalia, é provável que abranger operações normais e anomalias ao mesmo tempo com apenas um modelo não seja a melhor forma de se abordar este problema. Levando esses pontos em consideração, a metodologia utilizada

foi alterada para o Cenário 3, conforme a seção 3.4.

4.3 Cenário 3

4.3.1 Processamento de dados

O *subdataset* inicial neste cenário continha 687.446 linhas e 8 variáveis. A partir de conhecimentos adquiridos nos cenários anteriores, sabe-se que T-JUS-CKGL não possui dados e P-PDG está congelado ou apresentando valores sem sentido físico pela maior parte do *subdataset*. Sendo assim, estas duas variáveis foram removidas, reduzindo o número de variáveis para 6. Quanto às variáveis T-JUS-CKP, P-JUS-CKGL, P-MON-CKP e QGL, algumas medidas foram tomadas para preservá-las sem a exclusão de uma grande quantidade de linhas.

T-JUS-CKP apresentou valores inválidos em todas as instâncias de classe 8 e na maioria das instâncias de classe 2. Para preservar a variável juntamente aos dados das outras variáveis nas linhas onde T-JUS-CKP era inválido, a decisão tomada foi de definir T-JUS-CKP como zero para a anomalia 8 e preencher o restante dos dados ausentes dessa variável com a média dos valores disponíveis na anomalia 2. De forma similar, dados faltantes das variáveis P-JUS-CKGL e P-MON-CKP foram preenchidos com a média dos valores disponíveis. Para o QGL, todos os dados ausentes foram definidos como zero, a partir de conclusões tomadas no Cenário 1. Outra similaridade ao Cenário 1 é a presença de linhas não rotuladas no *subdataset* analisado, que foram removidas sob a premissa de que o efeito na qualidade dos dados seria insignificante, conhecimento também adquirido durante Cenário 1.

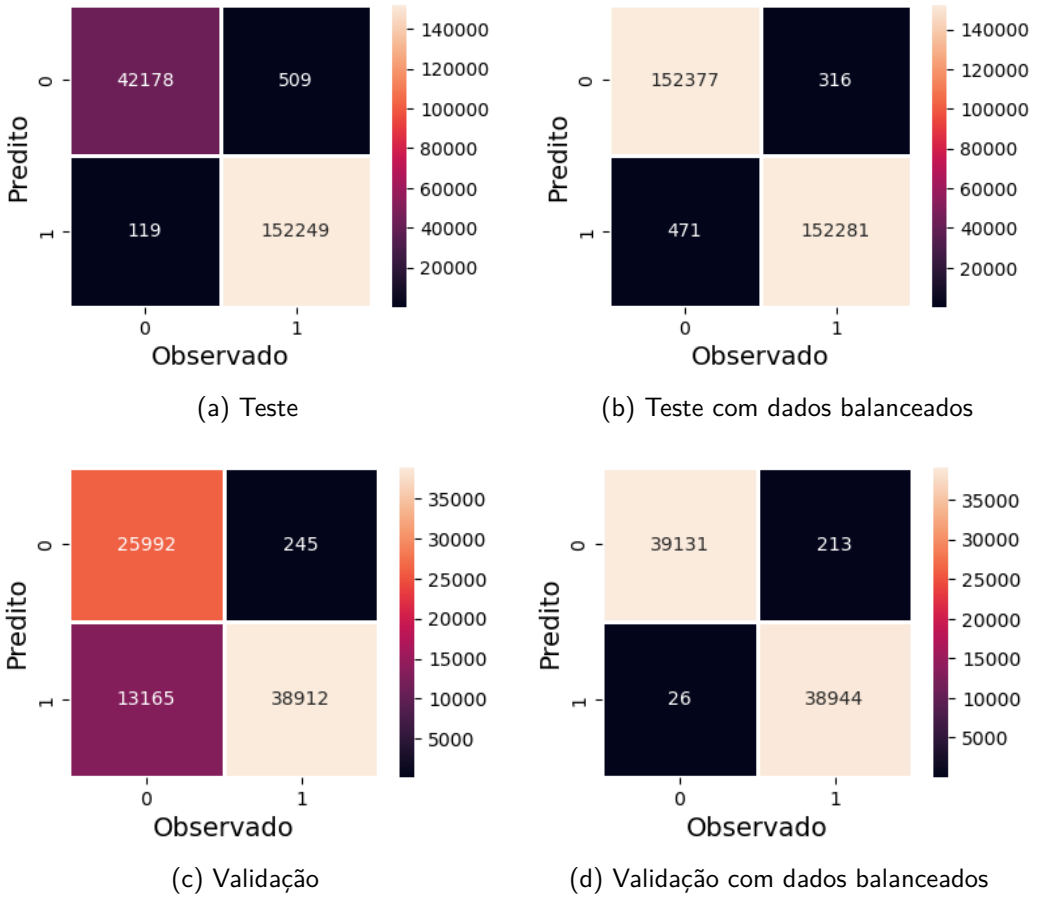
Após o pré-processamento dos dados, a transferência dos dados de anomalia em estado estável para outro armazenamento resultou em um novo *dataset* com 78.304 linhas, considerando o seu balanceamento com uma amostra de dados da classe 0. Após a divisão, restaram 650.181 linhas no *subdataset* utilizado para treinar e testar os modelos. Para a segunda parte do cenário, na qual o *subdataset* de treinamento é balanceado com nova amostra de dados da classe 0, conforme descrito na metodologia, um terceiro *dataset* foi gerado, totalizando 1,01 milhão de linhas.

4.3.2 Modelagem e Desempenho - Detecção de falhas

4.3.2.1 Árvore de Decisão

As matrizes de confusão, curvas ROC e demais métricas de desempenho dos modelos de AD gerados para detecção de falhas estão dispostas nas Figuras 4.9 e 4.10, bem como nas Tabelas 4.6 e 4.7. As curvas ROC da AD na fase de teste já apresentam AUCs muito próximos ou iguais a 1, logo elas foram

omitidas destes gráficos para permitir uma visualização mais clara das curvas ROC da fase de validação:



(a) Teste

(b) Teste com dados balanceados

(c) Validação

(d) Validação com dados balanceados

Figura 4.9: Matrizes de confusão do modelo de AD ajustado ao *subdataset* de treinamento para detecção de falhas

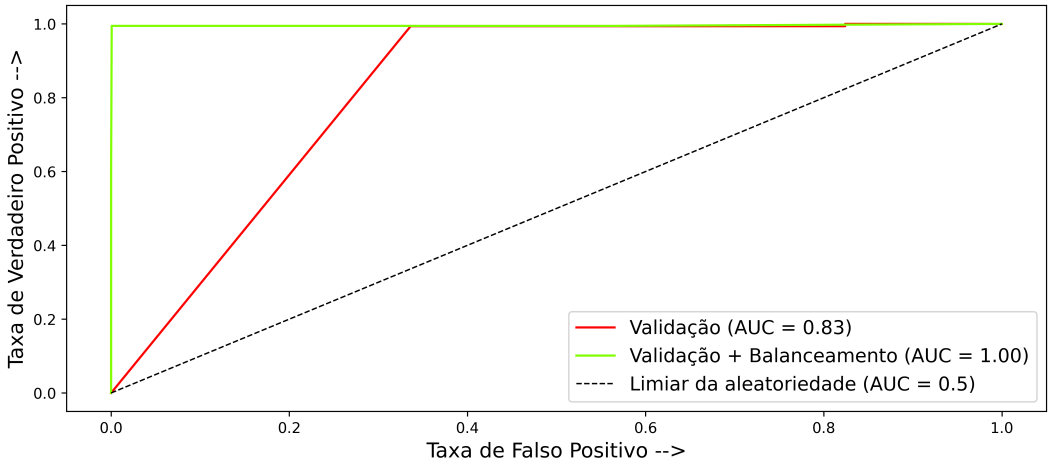


Figura 4.10: Curvas ROC do modelo de AD ajustado ao *subdataset* de treinamento para detecção de falhas

AD	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
0	0,99	1,00	0,99	0,99	0,66	0,79
1	1,00	1,00	1,00	0,75	0,99	0,85

Tabela 4.6: Métricas de desempenho de detecção de falhas do modelo de AD ajustado ao *subdataset* de treinamento

AD	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
0	0,99	1,00	0,99	0,99	0,66	0,79
1	1,00	1,00	1,00	0,75	0,99	0,85

Tabela 4.7: Métricas de desempenho de detecção de falhas do modelo de AD ajustado ao *subdataset* de treinamento com dados balanceados

É possível observar que o balanceamento do conjunto de dados influenciou os modelos de AD positivamente durante a fase de detecção, especialmente durante a etapa de validação. Os modelos treinados em dados balanceados apresentaram desempenho quase perfeito em todas as métricas avaliadas para detecção de falhas. Sem esse balanceamento, a AD ainda apresentou um desempenho satisfatório, obtendo um *recall* próximo de 1 para a classe de falha em todas as etapas, o que significa que a utilização deste modelo não ofereceria riscos adicionais de segurança.

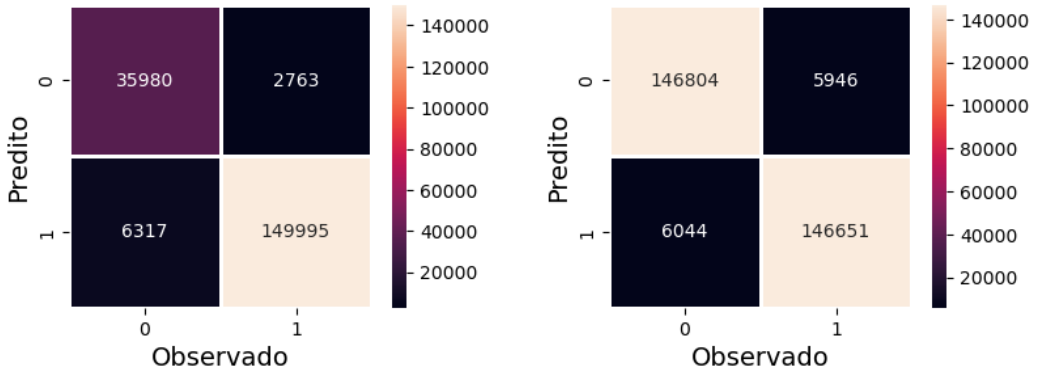
Observando as matrizes de confusão, percebe-se como o balanceamento de dados ajudou na criação de um modelo mais equilibrado. O modelo de AD treinado no *dataset* desbalanceado acabou com um desempenho fortemente enviesado para a detecção de FPs na etapa de validação, o que prejudica métricas como o F_1 quando comparado a um modelo equilibrado com a mesma acurácia. Já o modelo treinado no *dataset* balanceado apresentou melhor equilíbrio, porém enviesado para a detecção de FNs. Apesar disso, os valores próximos de 1.00 em todas as métricas avaliadas eliminam quaisquer dúvidas em relação tanto ao desempenho quanto aos riscos de operação deste modelo.

Na avaliação das curvas ROC, fica evidente que o balanceamento do conjunto de dados foi eficaz na melhoria do desempenho de detecção de falhas do modelo DT, elevando o AUC de 0,83 para 1,00 na etapa de validação.

A ideia de adicionar uma amostra de dados da classe 0 aos conjuntos de treinamento e teste surgiu do conhecimento de que os dados disponíveis da classe 0, embutidos nos conjuntos de anomalias, não eram suficientemente representativos (correspondendo a apenas um quinto do conjunto de dados resultante). O balanceamento dos dados em termos numéricos é uma técnica conhecida por melhorar o desempenho dos modelos. Além disso, essa abordagem provavelmente ampliou a perspectiva dos modelos, pois incluiu dados da classe 0 fora do contexto das anomalias avaliadas, aumentando sua robustez e preparando-os melhor para aplicações no mundo real.

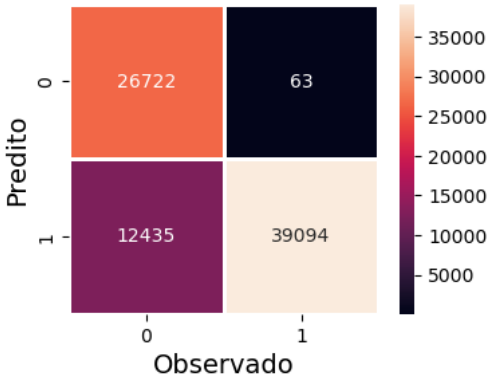
4.3.2.2 Multi-Layer Perceptron

As matrizes de confusão, curvas ROC e demais métricas de desempenho dos modelos de MLP gerados para detecção de falhas estão dispostas nas Figuras 4.11 e 4.12, bem como nas tabelas 4.8 e 4.9 :

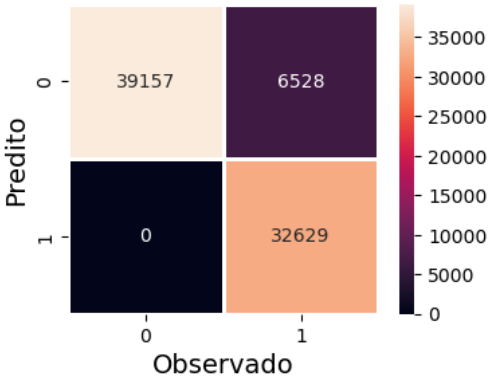


(a) Teste

(b) Teste com dados balanceados



(c) Validação



(d) Validação com dados balanceados

Figura 4.11: Matrizes de confusão do modelo de MLP ajustado ao *subdataset* de treinamento para detecção de falhas

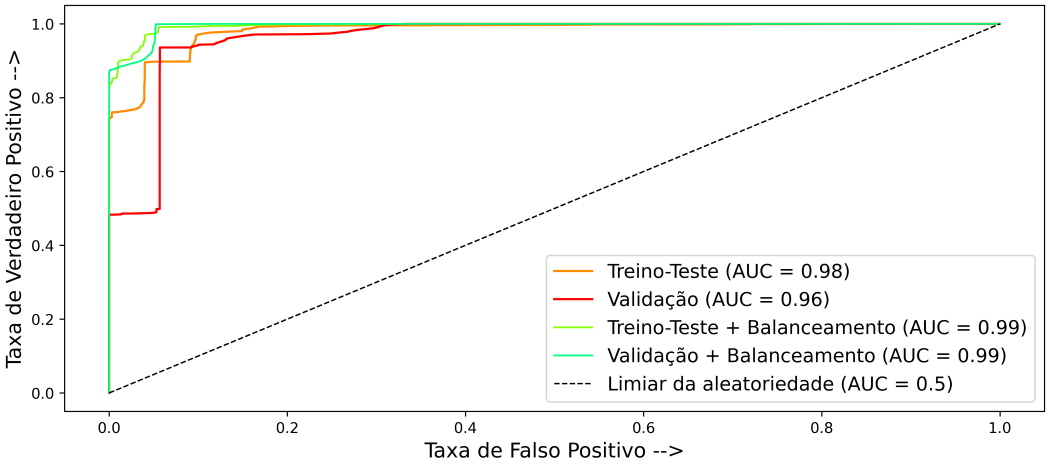


Figura 4.12: Curvas ROC do modelo de MLP ajustado ao *subdataset* de treinamento para detecção de falhas

MLP	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
0	0,93	0,85	0,89	1,00	0,68	0,81
1	0,96	0,98	0,97	0,76	1,00	0,86

Tabela 4.8: Métricas de desempenho de detecção de falhas do modelo de MLP ajustado ao *subdataset* de treinamento

MLP	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
0	0,96	0,96	0,96	0,86	1,00	0,92
1	0,96	0,96	0,96	1,00	0,83	0,91

Tabela 4.9: Métricas de desempenho de detecção de falhas do modelo de MLP ajustado ao *subdataset* de treinamento com dados balanceados

O balanceamento dos dados também propiciou melhorias no desempenho do modelo de MLP, a ser observado no aumento dos valores de f1 em todas as etapas. No entanto, os valores de *recall* para a classe de falha foram reduzidos, especialmente durante a etapa de validação, com uma queda substancial de 1,00 para 0,83. Ao visualizar as curvas ROC, torna-se mais evidente a melhora do desempenho geral do modelo, com valores maiores de AUC. A AUC na etapa de teste subiu de 0,98 para 0,99, enquanto a AUC na etapa de validação subiu de 0,96 também para 0,99.

Apesar de o balanceamento de dados ter potencializado o modelo de AD de forma clara, o mesmo não pode ser concluído para o modelo de MLP. O impacto positivo do balanceamento no desempenho geral do modelo foi contraposto pelo comprometimento da segurança inerente do modelo. Pode-se

argumentar, entretanto, que os baixos valores de F_1 na etapa de validação para o modelo sem balanceamento de dados implicam que o desempenho do modelo é insuficiente do ponto de vista de operacionalidade da plataforma.

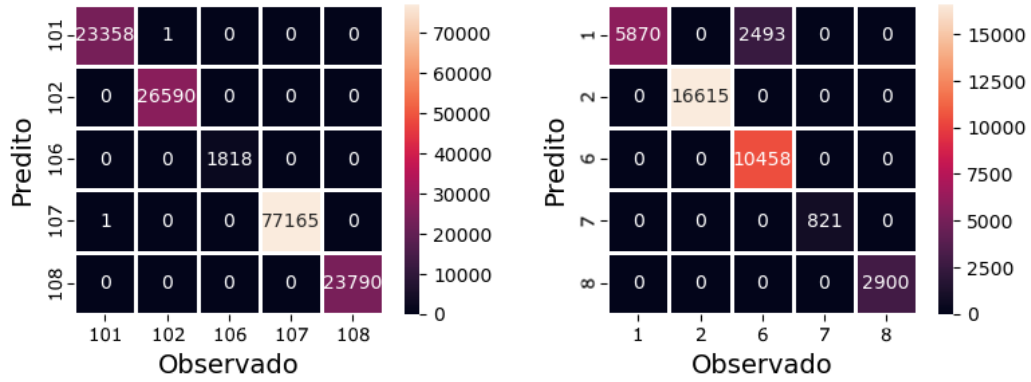
Olhando primeiramente para a Figura 4.8c, o enviesamento do modelo de MLP sem dados balanceados para a detecção de FPs é maior ainda do que o apresentado pelo modelo de AD, detectando 12435 FPs contra apenas 63 FNs. Esse viés foi prenunciado pela matriz de confusão de teste do modelo, que também já apresentava a mesma tendência. Já na Figura 4.8d, percebe-se que o balanceamento de dados jogou o viés do modelo completamente para o lado oposto, sem detectar nenhum FP, porém mais de 6 mil FNs. Estas observações auxiliam na interpretação dos dados das tabelas 4.8 e 4.9 e na conclusão de que o balanceamento de dados neste caso levou a uma troca entre desempenho e segurança. É interessante notar também que a matriz de confusão de teste indicava um ótimo equilíbrio do modelo, ressaltando a importância de etapas adicionais de validação como a que está sendo realizada neste trabalho.

4.3.3
Modelagem e Desempenho - Diagnóstico de falhas

Nesta seção, apenas as curvas ROC da etapa de validação serão apresentadas, pois tanto o modelo de MLP quanto o modelo de DT apresentaram AUCs muito próximas de, ou igual a 1, para todas as classes analisadas na etapa de teste.

4.3.3.1
Árvore de Decisão

As matrizes de confusão, curvas ROC e demais métricas de desempenho dos modelos de AD gerados para diagnóstico de falhas estão dispostas nas Figuras 4.13 e 4.14, bem como na Tabela 4.10:



(a) Teste (b) Validação

Figura 4.13: Matrizes de confusão do modelo de AD ajustado ao *subdataset* de treinamento para diagnóstico de falhas

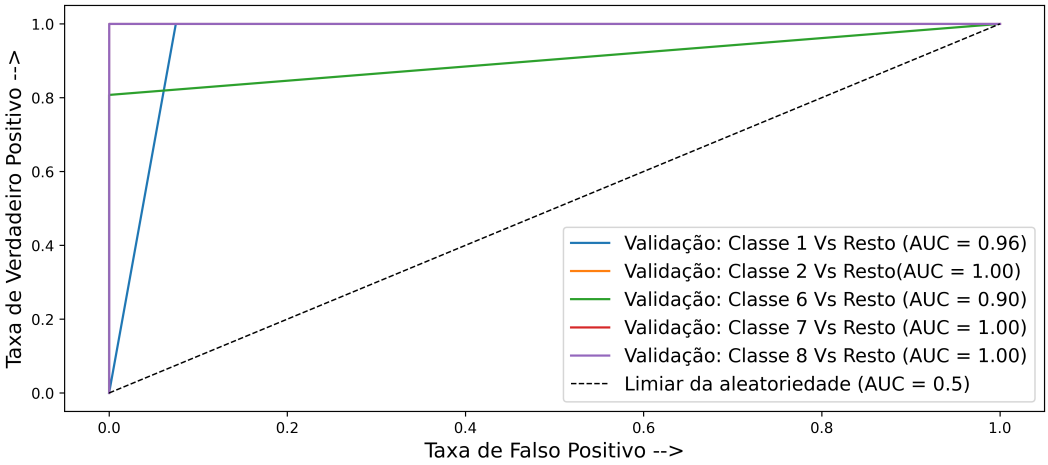


Figura 4.14: Curvas ROC do modelo de AD ajustado ao *subdataset* de treinamento para diagnóstico de falhas

AD	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
1	1,00	1,00	1,00	0,70	1,00	0,82
2	1,00	1,00	1,00	1,00	1,00	1,00
6	1,00	1,00	1,00	1,00	0,81	0,89
7	1,00	1,00	1,00	1,00	1,00	1,00
8	1,00	1,00	1,00	1,00	1,00	1,00

Tabela 4.10: Métricas de desempenho de diagnóstico de falhas do modelo de AD ajustado ao *subdataset* de treinamento

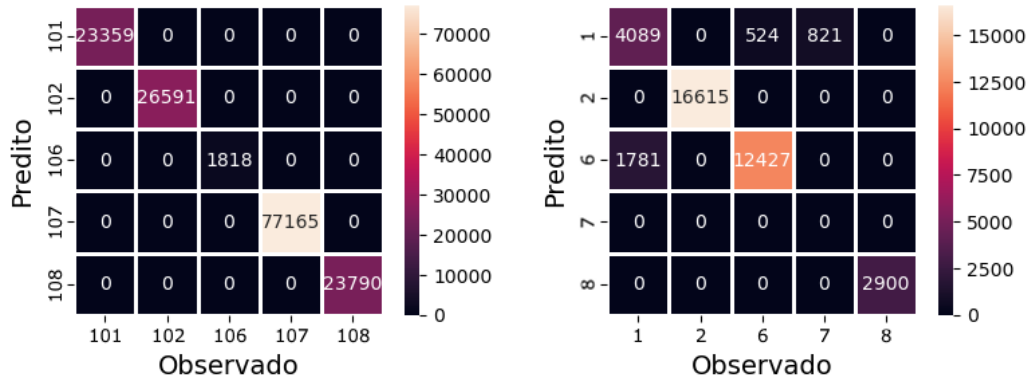
Vale notar que, como este modelo foi treinado apenas em dados de anomalia nesta fase, não é possível realizar o balanceamento de dados como anteriormente pois não existem mais dados das classes anômalas disponíveis para tal. Apesar disso, o modelo de AD obteve bom desempenho ao corretamente diagnosticar a classe de anomalia em questão, com acurácia absoluta para as classes 2, 7 e 8 e o menor recall sendo de 0,81 para diagnosticar anomalias da classe 6. Ao observar a matriz de confusão do modelo, é possível concluir que a principal dificuldade do modelo foi em diferenciar entre as classes 1 e 6, sem demais detecções falsas em nenhuma outra classe.

A análise da curva ROC oferece uma visualização mais clara das classes que ofereceram certa dificuldade para o modelo de AD treinado. Dito isso, os valores das AUCs foram todas igual ou acima de 0,9, demonstrando mais uma vez o bom desempenho deste modelo.

4.3.3.2 Multi-Layer Perceptron

As matrizes de confusão, curvas ROC e demais métricas de desempenho dos modelos de MLP gerados para diagnóstico de falhas estão dispostas nas

Figuras 4.15 e 4.16, bem como na tabela 4.11:



(a) Teste (b) Validação

Figura 4.15: Matrizes de confusão do modelo de MLP ajustado ao *subdataset* de treinamento para diagnóstico de falhas

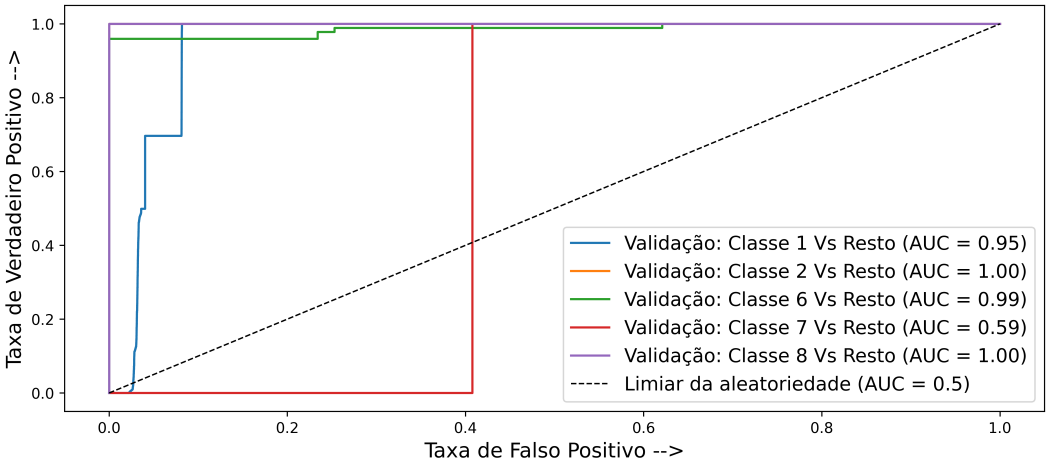


Figura 4.16: Curvas ROC do modelo de MLP ajustado ao *subdataset* de treinamento para diagnóstico de falhas

MLP	Teste			Validação		
Classe	Precisão	*Recall*	F_1	Precisão	*Recall*	F_1
1	1,00	1,00	1,00	0,75	0,70	0,72
2	1,00	1,00	1,00	1,00	1,00	1,00
6	1,00	1,00	1,00	0,87	0,96	0,92
7	1,00	1,00	1,00	0,00	0,00	0,00
8	1,00	1,00	1,00	1,00	1,00	1,00

Tabela 4.11: Métricas de desempenho de diagnóstico de falhas do modelo de AD ajustado ao *subdataset* de treinamento

Apesar de todas as questões de balanceamento de dados, o MLP também aparenta ter bom desempenho, se sobressaindo levemente acima do modelo de DT em algumas métricas, como os valores de *recall* e F_1 para a classe 6 no conjunto de validação. No entanto, o modelo falhou completamente em diagnosticar a anomalia da classe 7. Olhando a matriz de confusão, percebe-se que todas as suas ocorrências foram classificadas como classe 1. Houve uma certa confusão entre as classes 1 e 6 também, de forma similar à AD.

Uma explicação possível para tal é a falta de dados de classe 7 no *dataset*. Como a anomalia da classe 7 ocorre ao longo do tempo, devido a incrustações, sua dinâmica é lenta, o que significa que leva mais tempo para o cenário da anomalia se estabilizar. Isso resulta em uma representação aumentada de dados em anomalias transiente e, consequentemente, uma menor representatividade de anomalias em estado estável (<5%, neste caso) no *dataset*.

Analisando as Curvas ROC do modelo, percebe-se que todas as curvas possuem boas AUC, exceto a curva das falhas de classe 7, com um valor de 0,59, um valor próximo ao limiar de aleatoriedade.

4.3.4 Resultados comparativos e discussão

Os resultados obtidos neste cenário final, especialmente referentes aos modelos de AD, superam a maior parte dos resultados presentes na literatura. Os valores perfeitos obtidos de acurácia e F_1 nesta dissertação para o diagnóstico de anomalias de classe 2 superam as métricas obtidas por Machado et al (2022) de até 0,936 após ajustarem um modelo de Long Short-Term Memory (LSTM) aos dados. O *dataset* de classe 2 possui uma das maiores proporções de linhas com dados inválidos, as quais foram todas descartadas no estudo de Machado et al. (2022), provavelmente prejudicando o desempenho do modelo ajustado por eles.

Os valores obtidos nesta dissertação de F_1 para detecção de falhas sem balanceamento de dados durante a etapa de validação ficaram ligeiramente aquém do valor alcançado por Fernandes, Komati e Gazolli (2024) com o seu modelo de Local Outlier Factor (LOF) que também foi treinado apenas em instâncias reais (AD: 0,83; MLP: 0,84; LOF(Jr; Komati; Gazolli, 2024): 0,87). No entanto, após o balanceamento dos dados, os modelos de AD e MLP apresentaram resultados amplamente superiores em comparação, o que corrobora os efeitos positivos da adição de uma amostra de dados da classe 0 aos *datasets* de treinamento e teste.

Dificuldades em diferenciar eventos anômalos de classe 1, 6 também foram enfrentadas por Marins et al (2021), que treinou modelos de *Random Forest* em dados simulados e reais do 3W *dataset*. Dito isso, as acurácias obtidas pelos modelos de AD e MLP nesta dissertação para estas classes superam significativamente as métricas obtidas por Marins et al, de 0,508 e 0,710.

O *dataset* de classe 6 possui um problema similar ao *dataset* de classe 7: ele possui poucos dados de anomalia transiente (enquanto faltam dados de anomalia estável de calsse 7). Isso ocorre devido à rápida dinâmica da anomalia da classe 6, que é causada pelo fechamento acidental incorreto da válvula CKP, um evento rápido em contraste com as outras anomalias estudadas. O *dataset* de classe 6 também é o menor entre os utilizados, com apenas c. 54 mil linhas,

o que torna o diagnóstico dessa classe mais difícil em geral. Como não existem dados extras desta classe para poder realizar algum tipo de balanceamento, diferentemente dos dados de operação normal, os desequilíbrios entre cada classe de anomalia, especialmente para as classes 1, 6 e 7 nesta dissertação, continuam sendo um problema sem uma solução clara.

5
Considerações Finais

Os modelos de ML ajustados para detecção e diagnóstico de falhas em poços de produção *offshore* ao longo deste trabalho se demonstraram satisfatórios em grande parte. Em especial, os modelos de AD possuem ótima compatibilidade com o *dataset* estudado e apresentaram os melhores resultados nas métricas de *recall* e F_1, consideradas as mais importantes para este tipo de problema. Ademais, tanto AD quanto MLP produziram resultados significativamente superiores à RLog, demonstrando que algoritmos de complexidade ao menosmoderada são preferíveis para este problema.

A decisão por utilizar uma dupla de modelos para detecção e diagnóstico de falhas, em vez de um modelo só para realizar ambos os papéis, demonstrou-se necessária para simplificar tanto o problema quanto os modelos resultantes. Consequentemente, o desempenho e robustez dos modelos melhorou de forma significativa, demonstrado pela etapa de validação com uma parcela de dados reais inéditos na visão dos modelos.

Quanto à parte comparativa dos resultados, os modelos ajustados de MLP e AD no último experimento superaram consistentemente outros resultados disponíveis na literatura, demonstrando que os cuidados tomados durante o pré-processamento de dados não foram em vão.

Um estudo aprofundado do *dataset* a ser analisado, seguido por um pré-processamento bem realizado e a decisão em utilizar, ou não, técnicas de FS ou FE são de suma importância para o êxito dos modelos ajustados. A seleção dos modelos de forma fundamentada também é, evidentemente, necessário para obter êxito no estudo realizado, porém dar passos extras para preservar tanto a qualidade quanto a quantidade dos dados durante o pré-processamento possui um valor não apreciado.

Em suma, finalizando a resposta para a pergunta principal exposta na seção 1.1, esta dissertação mostra que modelos de ML são capazes, em grande parte, de detectar e diagnosticar anomalias de forma segura em situações reais de produção de petróleo e gás *offshore*, com o algoritmo de AD superando neste caso até mesmo modelos de *deep learning*, como o MLP.

5.1
Trabalhos e Desafios Futuros

Visto os desafios enfrentados não somente nesta dissertação mas também na literatura quanto a diagnosticar classes 1, 6 e 7, a procura e obtenção de uma base melhor de dados destas classes é importante para possibilitar o ajuste de modelos mais robustos. Trabalhar intencionalmente com *datasets* menores para auxiliar em um balanceamento melhor entre estas e outras classes também é uma possibilidade, porém o efeito na capacidade geral de detecção e diagnóstico será negativo.

Outros desafios a serem considerados envolvem a implementação destes modelos em poços operantes em tempo real, bem como a utilização destes modelos em poços diferentes dos utilizados para a sua criação e validação.

Conforme descrito no capítulo 3, o 3W *dataset* possui dados de mais de 20 poços diferentes, porém não é possível a partir do material aqui apresentado concluir sobre o desempenho dos modelos aqui ajustados em poços que não estão presentes neste *dataset*.

Um ponto final a se considerar é a própria estrutura do 3W *dataset*. As definições das classes de anomalias utilizadas provém de uma análise realizada por uma empresa, neste caso a Petrobras. Outras empresas e organizações possuem estruturas diferentes de dados, possivelmente discrepâncias nas classificações de anomalias, adicionando uma camada extra de complexidade na implementação deste tipo de modelo em operações de produção em tempo real.

Levando estes pontos em consideração, um possível trabalho futuro consiste na modelagem em ML de poços que não estão no escopo do 3W *dataset*, ou mesmo a utilização de dados de outros poços nos modelos criados neste trabalho. Vale notar que, neste caso, além da necessidade da detecção de padrões de anomalia em tempo hábil, os dados precisam também ser rearranjados de uma forma interpretável para os modelos.

Outra possibilidade para trabalhos futuros seria dar um passo adiante na metodologia de validação com dados reais descrita nesta dissertação, criando um ambiente online onde os dados são alimentados aos modelos da mesma forma que seriam em uma situação real de produção. a Junção dados de outros *benchmarks* além do 3W *dataset*, experimentando com a unificação destes dados em um formato para interpretação pelos modelos também pode cobrir outro aspecto dos desafios acima apontados que ainda necessitam ser abordados.

6
Referências Bibliográficas

Abdi, H.; Williams, L. J. Principal component analysis. **WIREs Computational Statistics**, v. 2, n. 4, p. 433–459, 2010. Disponível em: <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.101>.

Aggarwal, C. C. Data classification. In: _____. **Data Mining: The Textbook**. Cham: Springer International Publishing, 2015. p. 285–344.

Aggarwal, C. C. An introduction to neural networks. In: _____. **Neural Networks and Deep Learning: A Textbook**. Cham: Springer International Publishing, 2018. p. 1–52. ISBN 978-3-319-94463-0.

Agência Nacional do Petróleo, Gás Natural e Biocombustíveis - ANP. **Boletim da Produção de Petróleo e Gás Natural - Setembro/2024**. 2024. Disponível em: <https://www.gov.br/anp/pt-br/centrais-de-conteudo/publicacoes/boletins-anp/boletins/arquivos-bmppgn/2024/setembro.pdf>.

Predicting the Risk of Lost Circulation Using Support Vector Machine Model, v. 54th U.S. Rock Mechanics/Geomechanics Symposium de **U.S. Rock Mechanics/Geomechanics Symposium**, (U.S. Rock Mechanics/Geomechanics Symposium, v. 54th U.S. Rock Mechanics/Geomechanics Symposium). ARMA-2020-1154 p.

Bai, Y.; Bai, Q. **Sistemas Marítimos de Produção de Petróleo**. Elsevier, 2016. (Engenharia de Petróleo). ISBN 9788535273205.

Bishop, C. M. **Pattern Recognition and Machine Learning**. Cham: Springer International Publishing, 2006. (Information Science and Statistics). ISBN 9781493938438.

Bramer, M. **Principles of Data Mining**. Cham: Springer International Publishing, 2020. (Undergraduate Topics in Computer Science). ISBN 9781447174936.

Dinov, I. D. Decision tree divide and conquer classification. In: _____. **Data Science and Predictive Analytics: Biomedical and Health Applications using R**. Cham: Springer International Publishing, 2018. p. 307–343. ISBN 978-3-319-72347-1.

Energy Institute. **Statistical Review of World Energy**. Energy Institute, 2023. ISBN 9781787254084.

Ertel, W. Introduction. In: _____. **Introduction to Artificial Intelligence**. Cham: Springer International Publishing, 2017. p. 1–21. ISBN 978-3-319-58487-4.

Freitas, F.; Almeida, E. de; Fernández, E. F. y. Perspectivas para arrecadação de participações governamentais no setor de Óleo e gás no brasil. **Ensaio Energético**, 2023. Disponível em: <https://ensaioenergetico.com.br/perspectivas-para-arrecadacao-de-participacoes-governamentais-no-setor-de-oleo-e-gas-no-brasil/>.

Goode, K.; Roylance, B.; Moore, J. Development of model to predict condition monitoring interval times. **Ironmaking & Steelmaking**, v. 27, n. 1, p. 63–68, 2000.

Gudmundsson, J. S. **Flow Assurance Solids in Oil and Gas Production**. CRC Press - Taylor Francis Group, 2017. ISBN 9781315185118.

Guo, B.; Lyons, W. C.; Ghalambor, A. **Petroleum production engineering - a computer-assisted approach**. Elsevier Science Technology Books, 2007. ISBN 0750682701.

Hausler, R. H.; Krishnamurthy, R. M.; Sherar, B. W. Observation of productivity loss in large oil wells due to scale formation without apparent production of formation brine. In: Nace international. **NACE CORROSION**. 2015. p. NACE–2015–6147.

Helmiriawan, H. **Scalability Analysis of Predictive Maintenance Using Machine Learning in Oil Refineries**. Dissertação (Mestrado) — Delft University of Technology, 2018. Disponível em: <https://resolver.tudelft.nl/uuid:dbf3e77c-e624-47ef-b951-3f1948b1609a>.

Iea. **Oil 2024**. 2024.

Igual, L.; Seguí, S. Supervised learning. In: _____. **Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications**. Cham: Springer International Publishing, 2017. p. 67–96. ISBN 9783319500171.

Jr, W. F.; Komati, K. S.; Gazolli, K. A. de S. Anomaly detection in oil-producing wells: a comparative study of one-class classifiers in a multivariate time series dataset. **J Petrol Explor Prod Technol**, v. 14, p. 1–21, 2024.

Junior, W. F. **Comparação de classificadores para detecção de anomalias em poços produtores de petróleo**. Dissertação (Mestrado) — Instituto Federal do Espírito Santo, 2022.

Kubat, M. Performance evaluation. In: _____. **An Introduction to Machine Learning**. Cham: Springer International Publishing, 2017. p. 211–229.

Laik, S. **Offshore Petroleum Drilling and Production**. CRC Press - Taylor Francis, 2018. ISBN 9781315157177.

Machado, A. P. F. et al. Improving performance of one-class classifiers applied to anomaly detection in oil wells. **Journal of Petroleum Science and Engineering**, v. 218, p. 110983, 2022. ISSN 0920-4105. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0920410522008348>.

Maitelli, A. et al. Simulação computacional para poços de petróleo com método de elevação artificial por bombeio centrífugo submerso. In: **Rio Oil Gas 2008**. [S.l.: s.n.], 2008. p. 8.

Manning, C. D.; Raghavan, P.; SchÜtze, H. Text classification and naive bayes. In: _____. **Introduction to Information Retrieval**. Cambridge University Press, 2008. p. 234–265.

Marins, M. A. et al. Fault detection and classification in oil wells and production/service lines using random forest. **Journal of Petroleum Science and Engineering**, v. 197, p. 107879, 2021. ISSN 0920-4105. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0920410520309372>.

Mcdermott, M. B. A. et al. **A Closer Look at AUROC and AUPRC under Class Imbalance**. 2025. Disponível em: <https://arxiv.org/abs/2401.06091>.

Salimova, R.; Pourafshary, P.; Wang, L. Data-driven analyses of low salinity waterflooding in carbonates. **Applied Sciences**, v. 11, n. 14, 2021. ISSN 2076-3417. Disponível em: <https://www.mdpi.com/2076-3417/11/14/6651>.

Schernikau, L.; Smith, W. Climate impacts of fossil fuels in todays electricity systems. **Journal of the Southern African Institute of Mining and Metallurgy**, v. 122(3), p. 133–145, 2022.

Schweidtmann, A. M. et al. Machine learning in chemical engineering: A perspective. **Chemie Ingenieur Technik**, v. 93, n. 12, p. 2029–2039, 2021. Disponível em: <https://onlinelibrary.wiley.com/doi/abs/10.1002/cite.202100083>.

Skiena, S. S. Mathematical models. In: _____. **The Data Science Design Manual**. Cham: Springer International Publishing, 2017. p. 201–236.

Causas de perda de produção que geram intervenção de manutenção em poço, v. 8° Congresso Brasileiro de Pesquisa e Desenvolvimento em Petróleo e Gás - 8°PDPETRO.

Superintendência de Segurança Operacional e Meio Ambiente - SSM. **Relatório Anual de Segurança Operacional das Atividades de Exploração e Produção de Petróleo e Gás Natural**. Agência Nacional do Petróleo, Gás Natural e Biocombustíveis - ANP, 2022.

Suursalu, S. **Predictive Maintenance Using Machine Learning Methods in Petrochemical Refineries**. Dissertação (Mestrado) — Delft University of Technology, 2017. Disponível em: <http://resolver.tudelft.nl/uuid:e95f39a4-569a-470e-a431-962b9766a302>.

Thomas, J. E. **Fundamentos de engenharia de petróleo**. Interciência, 2004. ISBN 9788571930995.

Vargas, R. E. V. et al. A realistic and public dataset with rare undesirable real events in oil wells. **Journal of Petroleum Science and Engineering**, v. 181, p. 106223, 2019. ISSN 0920-4105. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0920410519306357>.

VerÇosa, I. J. F. L. et al. Formação de hidratos em perfurações de poços em Águas profundas e ultra profundas. In: Universidade federal de campina grande. **III CONEPETRO**. 2018. ISSN 2446-8339.

Wang, W. A model to determine the optimal critical level and the monitoring intervals in condition-based maintenance. **International Journal of Production Research**, Taylor & Francis, v. 38, n. 6, p. 1425–1436, 2000.

Xu, P.; Du, R.; Zhang, Z. Predicting pipeline leakage in petrochemical system through gan and lstm. **Knowledge-Based Systems**, v. 175, p. 50–61, 2019. ISSN 0950-7051. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0950705119301340>.

Zhang, J. et al. A machine learning method for the risk prediction of casing damage and its application in waterflooding. **Sustainability**, v. 14, n. 22, 2022. ISSN 2071-1050. Disponível em: <https://www.mdpi.com/2071-1050/14/22/14733>.