UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

CENTRO TECNOLÓGICO

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

JOSÉ MARQUES DE OLIVEIRA JÚNIOR

Orientador: Prof. Dr. Celso José Munaro

Coorientador: Prof. Dr. Patrick Marques Ciarelli

ESTIMAÇÃO DE TEOR DE ÓLEOS E GRAXAS EM ÁGUA DESCARTADA NO MAR USANDO MODELOS BASEADOS EM DADOS

Vitória, ES

2022

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO
CENTRO TECNOLÓGICO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

JOSÉ MARQUES DE OLIVEIRA JÚNIOR

ESTIMAÇÃO DE TEOR DE ÓLEOS E GRAXAS EM ÁGUA DESCARTADA NO MAR USANDO MODELOS BASEADOS EM DADOS

Dissertação apresentada ao Curso de Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Espírito Santo como parte dos requisitos necessários para a obtenção do grau de Mestre em Engenharia Elétrica.

Orientador: Prof. Dr. Celso José Munaro
Coorientador: Prof. Dr. Patrick Marques Ciarelli

Vitória, ES
2022

Ficha catalográfica disponibilizada pelo Sistema Integrado de Bibliotecas - SIBI/UFES e elaborada pelo autor

---

Oliveira Júnior, José Marques de, 1993-
O48e Estimação de teor de óleos e graxas em água descartada no mar usando modelos baseados em dados / José Marques de Oliveira Júnior. - 2022.
75 f. : il.

Orientador: Celso José Munaro.
Coorientador: Patrick Marques Ciarelli.
Dissertação (Mestrado em Engenharia Elétrica) - Universidade Federal do Espírito Santo, Centro Tecnológico.

1. Teor de óleos e graxas. 2. Monitoramento ambiental. 3. Monitoramento de processo. 4. Ciência de Dados. 5. Redes neurais recorrentes. I. Munaro, Celso José. II. Ciarelli, Patrick Marques. III. Universidade Federal do Espírito Santo. Centro Tecnológico. IV. Título.

CDU: 621.3

José Marques de Oliveira Júnior

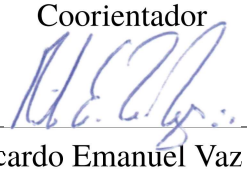
ESTIMAÇÃO DE TEOR DE ÓLEOS E GRAXAS EM ÁGUA DESCARTADA NO MAR USANDO MODELOS BASEADOS EM DADOS

Dissertação apresentada ao Curso de Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal do Espírito Santo como parte dos requisitos necessários para a obtenção do grau em Mestre em Engenharia Elétrica.

Aprovada em Vitória, 04 de abril de 2022.

Prof. Dr. Celso José Munaro
Universidade Federal do Espírito Santo
Orientador

Prof. Dr. Patrick Marques Ciarelli
Universidade Federal do Espírito Santo
Coorientador



Dr. Ricardo Emanuel Vaz Vargas
Petróleo Brasileiro S. A., ES
Examinador

Prof. Dr. Luiz Affonso Henderson Guedes de Oliveira
Universidade Federal do Rio Grande do Norte (UFRN)
Examinador

À minha mãe, Luzia Marques de Oliveira.

Agradecimentos

Aos meus pais, José e Luzia, grandes exemplos de força e persistência.

Às minhas irmãs Gabriely, Adriana, Ana Paula, Sandra e Sara, pelo apoio fundamental e contínuo. Aos familiares, profunda gratidão pois essa jornada foi construída por todos.

À Mayra, pelo amor e incentivo, sobretudo por compreender as ausências e por surgir no céu uma estrela a cada vez que sorri.

Aos Prof. Celso, Prof. Patrick, Ricardo e Karina, pela paciência, pelos conselhos, pelo grandes esforços empreendidos e por acreditarem no projeto. O crescimento acadêmico, profissional e pessoal foi incrível graças à ótima equipe.

Aos amigos da Petrobras, principalmente da P-57, pelo suporte, encorajamento e companheirismo durante todos esses anos.

À Petrobras, pela permissão para publicar e participar de congressos, e à equipe do TOGy, por compartilhar os dados e as experiências, ambos essenciais para realização deste trabalho.

À UFES e ao PPGEE, por proporcionar ensino público de qualidade para todos.

A todos vocês, de coração, muito obrigado.

"O que sabemos é uma gota; o que ignoramos é um oceano".

Resumo

Água produzida, em plataformas marítimas, é um dos efluentes recuperados de poços em conjunto com petróleo e gás natural, sendo o principal resíduo gerado nesse processo. O Teor de Óleos e Graxas (TOG) é considerado um dos principais parâmetros de controle do descarte de água produzida no mar, com limites diários e mensais definidos pela legislação vigente. A medição de TOG usada como referência pelo IBAMA é feita pelo método gravimétrico, com amostras de água coletadas diariamente e enviadas para laboratório acreditado, que fornece os resultados com defasagem de alguns dias a partir da data de amostragem. A necessidade de ações corretivas em caso de valores acima do limite tem motivado o uso de métodos alternativos que gerem estimativas com maior frequência. Neste trabalho, modelos baseados em dados são criados para obtenção de estimativas do TOG. Variáveis de processo de tratamento de água produzida, informações sobre produtos químicos e dados sobre produção diária de uma plataforma *offshore* foram coletados, tratados e utilizados para treinar, validar e testar esses modelos. Além disso, foram aplicadas técnicas de otimização de hiperparâmetros e seleção de atributos. Os resultados obtidos mostraram que os modelos baseados em redes neurais recorrentes (LSTM e CNN+LSTM) alcançaram desempenhos superiores se comparados aos sistemas de monitoramento online existentes.

Palavras-chave: Estimação de teor de óleos e graxas, Redes neurais recorrentes, Engenharia de Atributos.

Abstract

Produced water, on offshore platforms, is one of the effluents recovered from wells together with oil and natural gas, being the main waste generated in this process. The Total Oil and Grease (TOG) is considered one of the main parameters for controlling the disposal of produced water at sea, with daily and monthly limits defined by current legislation. The TOG measurement used as a reference by IBAMA is done by the gravimetric method, with water samples collected daily and sent to an accredited laboratory, which provides the results with a lag of a few days from the sampling date. The need for corrective actions in case of values above the limit has motivated the use of alternative methods that generate estimates more frequently. In this work, data-based models are created to obtain TOG estimates. Produced water treatment process variables, chemical information and daily production data from an offshore platform were collected, treated and used to train, validate and test these models. In addition, hyperparameter optimization and feature selection techniques were applied. The results obtained showed that the models based on recurrent neural networks (LSTM and CNN+LSTM) achieved superior performances compared to existing online monitoring systems.

Keywords: Estimation of oil and grease content, Recurrent neural networks, Attribute Engineering.

Lista de Ilustrações

Lista de Tabelas

Lista de Abreviaturas e Siglas

ANP	Agência Nacional do Petróleo, Gás Natural e Biocombustíveis
AM	Aprendizado de Máquina
AP	Água Produzida
BDO	Boletim Diário de Operação
BSW	*Basic sediments and water*
CNN	*Convolutional neural network*
CONAMA	Conselho Nacional do Meio Ambiente
CV	Coeficiente de variação
IBAMA	Instituto Brasileiro do Meio Ambiente e dos Recursos Naturais Renováveis
LSTM	*Long short-term memory*
RF	*Random forest*
RNA	Redes neurais artificiais
RNN	Redes neurais recorrentes
SM	*Standard Methods For the Examination of Water and Wastewater*
TOG	Teor de Óleos e Graxas

Sumário

1 Introdução

1.1 Introdução

O petróleo é ainda hoje uma das maiores fontes na matriz energética mundial. Entretanto, sua produção pode afetar o meio ambiente de diversas maneiras, sendo a Água Produzida (AP) o maior efluente do processo (AMINI et al., 2012). A AP é trazida à superfície juntamente com o óleo e o gás durante a produção desses fluídos e contém contaminantes como o próprio óleo, produtos químicos e gases dissolvidos (BAYATI; SHAYEGAN; NOORJAHAN, 2011). O tratamento e o descarte corretos são fatores críticos nessa indústria pois falhas nesse estágio podem levar a danos extremos ao ambiente marítimo e prejuízos financeiros à empresa.

A métrica de qualidade da água amplamente utilizada é o Teor de Óleos e Graxas (TOG), por vezes chamada, em inglês, de *Total Oil and Grease* ou *Oil-in-Water*. No Brasil, a Resolução CONAMA 393/2007 (BRASIL, 2007) estabelece a metodologia sobre o descarte contínuo de água de processo ou de produção em plataformas marítimas de petróleo e gás natural. Destacam-se os seguintes artigos:

> Art. 5º O descarte de água produzida deverá obedecer à concentração média aritmética simples mensal de óleos e graxas de até 29 mg/L, com valor máximo diário de 42 mg/L.
>
> Art. 6º A concentração de óleos e graxas a que se refere o Art. 5º desta Resolução deverá ser determinada pelo método gravimétrico.
>
> Art. 11º Os métodos de coleta e de análise são os especificados em normas técnicas cientificamente reconhecidas.

A obrigatoriedade do uso do método gravimétrico para análise do TOG impõe desafios. Em plataformas de produção *offshore*, o balanço inerente às embarcações dificulta a execução das análises por esse método (YANG, 2011). Assim, as amostras são enviadas a laboratórios *onshore* e, devido à logística, os resultados são disponibilizados com defasagem, o que impossibilita o controle da planta de processamento em função desses resultados.

Para permitir um melhor acompanhamento do processo, a AP é analisada a bordo da unidade por meio de análises de laboratório e instrumentos *online*, estes últimos instalados no processo. Porém, vale frisar que o TOG é uma medida método-dependente, ou seja, os valores encontrados só têm significado quando é informado o método aplicado (YANG, 2011).

Dentro desse contexto, o que se propõe neste trabalho de dissertação de mestrado é a preparação de dados e aplicação de modelos, baseados em métodos estatísticos e de aprendizado de máquina, capazes de estimar os valores de TOG. A referência para tais modelos são os valores obtidos com o método gravimétrico SM 5520 B, utilizado por órgãos públicos para fins de

fiscalização. Os dados de entrada foram obtidos de diversas fontes (instrumentos e equipamentos já instalados na planta de produção), por exemplo: transmissores de pressão, temperatura, vazão e nível; corrente elétrica de tratores eletrostáticos; posições de válvulas de controle e segurança; informações sobre produtos químicos injetados, entre outros.

Os objetivos específicos serão focados nas fases iniciais da divisão do fluxo de construção de conhecimento a partir de dados (ROKACH; MAIMON, 2014). Tais etapas contemplam: aprofundamento da base teórica relativa ao tratamento da AP, os componentes do TOG e suas formas de medição; a seleção, pré-processamento, limpeza e transformação dos dados disponíveis; seleção de atributos e investigação de quais variáveis possuem maior impacto nas saídas; testes de modelos para estimação do TOG, bem como a otimização de hiperparâmetros. Desse forma, pretende-se solidificar os alicerces para a construção do estimador de TOG, com embasamento científico e domínio sobre o problema.

1.2 Justificativa

Segundo dados da ANP (Agência Nacional do Petróleo, Gás Natural e Biocombustíveis), em maio de 2020 foram gerados quase $1,7$ milhão de barris por dia (ANP, 2020) de AP no Brasil. No mundo, esse volume chega a 250 milhões de barris (JIMÉNEZ et al., 2018). Com o amadurecimento do campo, a relação de água/óleo (chamada de BSW, *Basic Sediments and Water*) tende a aumentar, tornando seu gerenciamento ainda mais importante. A Figura 1.1 apresenta a evolução da produção nacional de petróleo e água entre 2009 e 2019. Vale observar que, no geral, a produção de óleo só tem incrementos com a perfuração de novos poços, enquanto que o aumento de AP ocorre com o amadurecimento dos campos.

Tendo em vista o volume desse efluente e sua tendência de crescimento, o impacto ambiental causado pela AP é notório, principalmente relacionado a um fator: sua toxicidade. Pode-se atribuir tais efeitos, sobretudo, a: óleo (disperso e dissolvido), alta salinidade (concentração de sólidos dispersos), produtos químicos utilizados na operação, metais pesados e radioisótopos (FAKHRU'L-RAZI et al., 2009).

Além disso, há evidências de que o descarte da AP no mar causa alterações em espécies. Gagnon (2011) avaliou três espécies de peixes nas proximidades de unidades de produção na Austrália, sendo observadas mudanças fisiológicas. Os resultados sugerem que a carga de contaminantes (volume de AP $\times$ concentração de hidrocarbonetos) gera uma resposta mais clara nos biomarcadores (instrumentos que possibilitam identificar a substância tóxica ou uma condição adversa antes que sejam evidenciados danos à saúde (AMORIM, 2003)). Em outro estudo, Meier et al. (2010) expuseram peixes em diversas fases de desenvolvimento (embrionária, larval e juvenil) à diferentes concentrações de AP. A concentração de 1% de AP interferiu na pigmentação das larvas e, após se desenvolverem, a maior parte não se alimentou adequadamente, morrendo por inanição. Observou-se que tal fato se relacionou com a aumentada taxa de deformidades nas

Figura 1.1 – Comparação da série histórica da produção de petróleo e AP no Brasil.

Fonte: Construído pelo autor a partir dos dados disponibilizados em (ANP, 2020).

mandíbulas.

A Operação Ouro Negro, iniciada no primeiro semestre de 2017, envolveu o Ministério Público do Trabalho, a Polícia Federal, o IBAMA e a Agência Nacional do Petróleo. Durante uma investigação a respeito do descarte de AP no mar pela Plataforma P-51 da Petrobras, foram gerados autos de infração pelo IBAMA, com multas superiores a R$ 14 milhões. A principal divergência era em relação ao método de análise do TOG. A Resolução CONAMA 393/2007 (BRASIL, 2007), no seu Art. 6º, determina que a concentração de óleos e graxas seja obtida por meio do método gravimétrico. A empresa, até então, empregava o método SM 5520 F (BAIRD et al., 2017), que também é gravimétrico mas o entendimento do IBAMA foi que o método correto seria o SM 5520 B (BAIRD et al., 2017), mais restritivo.

Em consequência desses fatos, um Termo de Compromisso foi firmado entre Petrobras e IBAMA com o objetivo de "disciplinar as ações e medidas necessárias durante o período de transição para adequação de 28 plataformas marítimas de produção da Petrobras nele listadas, em relação ao descarte de água de produção, regulado pelo artigo 5º da Resolução CONAMA nº 393/2007, mediante a realização das análises gravimétricas a partir do método *Standard Methods* (SM) 5520-B" (BRASIL, 2018). Como medida compensatória, foi aplicada uma multa de R$ 100 milhões.

Em caso de descumprimentos dos requisitos legais por parte de qualquer operadora, os órgãos reguladores podem aplicar sanções às empresas, que podem variar de não-conformidades leves, multas e até mesmo interdição da unidade de produção. Os riscos ambientais e os prejuízos

financeiros provenientes de uma falha no processo são gravíssimos. Em caso de interdição da plataforma, supondo que o volume produzido seja 100 mil barris de petróleo por dia, o lucro cessante seria maior que 47 milhões de reais por dia[1].

Há grandes desafios na implementação do método SM 5520 B no controle das plantas de tratamento de AP. A realização das análises de TOG por meio de tal método nas plataformas é inviável, já que são necessárias medições de peso (justificando o nome gravimétrico) e, até o momento, não há equipamentos disponíveis no mercado que realizem tal tarefa em embarcações, tendo em vista o balanço a que estão submetidas. Com isso, é necessário que as amostras sejam coletadas, desembarcadas, levadas até o laboratório em terra e, só então, o resultado da análise é disponibilizado.

Dada tal situação, o panorama atual de monitoramento e controle do TOG na área operacional se divide em duas formas: análises no laboratório na própria unidade ou por instrumentos *online*. Entretanto, deve-se considerar que o TOG é um parâmetro método-dependente (YANG, 2011), isto é, cada método consegue medir determinados componentes de hidrocarbonetos e não há garantia de correlação entre eles, seja linear ou não-linear. O SM 5520 B, que passou a ser adotado como desdobramento da Operação Ouro Negro, detecta tanto os hidrocarbonetos dissolvidos quanto os dispersos; o SM 5520 F (utilizado anteriormente), o fotométrico, o infravermelho e os demais medidores *online* não conseguem detectar a parte do óleo dissolvida na água. Com isso, a comparação entre a medição fiscalizada pelos órgãos ambientais e valores disponibilizados para a área operacional possuem correlações lineares e não-lineares baixas.

Apesar dessa dificuldade específica com os analisadores de TOG, há uma variedade de outros sensores em todo o processo de produção de óleo e gás. O uso de instrumentos inteligentes, como transmissores de pressão, temperatura, vazão, nível e válvulas de controle, tem aumentado, gerando dados provenientes da planta em operação. Tais dados, inicialmente concentrados nos CLPs (Controlador Lógico Programável), SDCDs (Sistema Digital de Controle Distribuído) e sistemas supervisórios, são historiados em um nível mais alto da pirâmide da automação (SOUZA et al., 2005), tornando-os acessíveis mesmo a grandes distâncias do chão-de-fábrica.

Diante de tal situação (de um lado, a necessidade de um melhor monitoramento do tratamento da AP; de outro, a grande disponibilidade de dados), é razoável a tentativa de criação de modelos para previsão de TOG baseados em métodos estatísticos e de Aprendizado de Máquina. Para tanto, são necessários alguns passos preliminares, como aprofundamento no conhecimento sobre o processo, integração, pré-processamento e visualização dos dados, identificação das variáveis críticas que influenciam o TOG e estabelecimento de métricas de avaliação dos modelos. Com tais recursos, o caminho para obtenção de bons estimadores se consolida com embasamento científico.

A perspectiva do uso de uma ferramenta que estime com boa precisão o Teor de Óleos e

[1] Cotações referentes ao dia 16/02/2022 - Brent: 91,93 US$/bbl; Câmbio: 5,14 R$/US$.

Graxas (em um intervalo de tempo mais curto do que a realidade atual) abarca a possibilidade da redução do risco de danos ambientais causados pela indústria de óleo e gás, reduzindo as chances do descarte inadequado da AP. Com a preocupação em relação a sustentabilidade do empreendimento cada vez maior, vale citar que também por parte do mercado financeiro, a empresa obtém ganhos na imagem ao ser vista como uma organização que realmente se atenta aos impactos potenciais de sua operação no meio ambiente e na sociedade. Ademais, tal ferramenta de predição poderia evitar multas aplicadas pelos órgãos regulamentadores, que diminuem a lucratividade do negócio.

1.3 Objetivos

O objetivo principal do presente trabalho é avaliar a utilização de variáveis do processo por meio de modelos baseados em dados para estimar o TOG em água de descarte na produção de petróleo.

Para tal, propõe-se uma metodologia cujos objetivos específicos são:

- selecionar e apresentar os dados, detalhando as fontes, frequência de amostragem e aspectos específicos;
- visualizar os dados, utilizando ferramentas que facilitem a análise destes a fim de torná-los úteis e de simples uso, obter as correlações entre os atributos e identificar as transformações necessárias;
- executar o pré-processamento (em conjunto com o item anterior), efetuando a limpeza, descartando os atributos dispensáveis e aplicando técnicas de seleção de atributos;
- propor e avaliar modelos para estimação de TOG, bem como técnicas para seleção de atributos para simplificação do modelo e aumento do desempenho.

1.4 Organização do Trabalho

Após este capítulo introdutório que discorre sobre o problema, as justificativas e os objetivos deste trabalho, o Capítulo 2 apresenta a revisão bibliográfica. Nele, são apresentadas as publicações relevantes sobre o tema e o embasamento teórico sobre o processo de tratamento da AP, as análises de TOG, as técnicas de pré-processamento de dados e as estratégias para criação de bons modelos de aprendizado de máquina. O Capítulo 3 apresenta a metodologia adotada, os dados selecionados, as transformações executadas, as correlações e estatísticas exploradas, as formas de validação cruzada, teste e métricas de avaliação dos modelos. O Capítulo 4 contém o estudo de caso a partir da metodologia empregada, apresentando a visualização dos dados, os efeitos obtidos com o pré-processamento, as variáveis mais importantes para os modelos e

o conhecimento gerado sobre o problema com o uso desta estrutura. Por fim, o Capítulo 5 traz as conclusões, avaliando as soluções providas e apontando um direcionamento para trabalhos futuros.

2 Revisão bibliográfica

Este capítulo contém a revisão bibliográfica, abarcando os temas de aprendizado de máquina (2.1), métricas de avaliação de modelos (2.1.1), regressão linear (2.1.3), árvores de regressão e *random forest* (2.1.4), redes neurais (2.1.5), regularização (2.1.6), seleção de atributos (2.1.7) e otimização de hiperparâmetros (2.1.8). Além disso, na Seção 2.2 são abordadas questões relativas ao tratamento da água produzida em unidades de produção *offshore* e às diferentes medições de TOG.

2.1 Aprendizado de Máquina

Existem algumas definições de Aprendizado de Máquina (AM). Para Géron (2019), é a ciência da programação de computadores para que eles possam aprender com os dados. Uma abordagem mais genérica é adotada por Samuel (1959): é o campo de estudo que dá aos computadores a habilidade de aprender sem ser explicitamente programados. Outra definição poderia ser dada como um programa de computador que aprende pela experiência E em relação a algum tipo de tarefa T e alguma medida de desempenho P se o seu desempenho em T, conforme medido em P, melhora com a experiência E (MITCHELL, 1997).

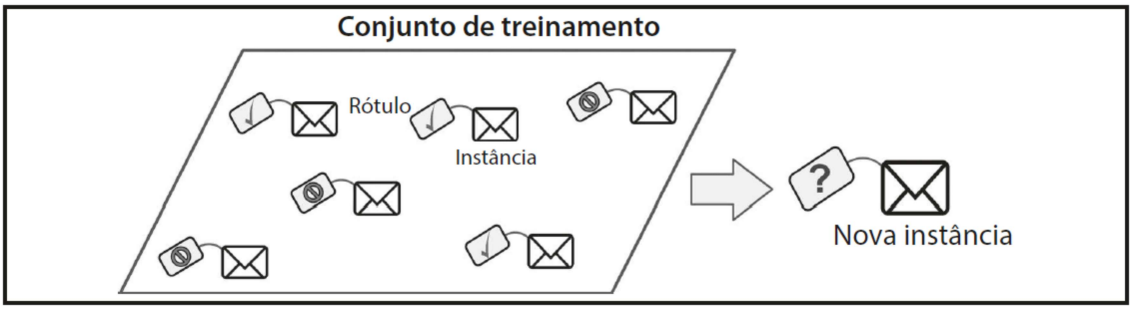
Para Géron (2019), a solução de problemas por meio de Aprendizado de Máquina é adequada quando:

- problemas para os quais as soluções existentes exigem muita configuração manual ou longas listas de regras: um algoritmo de AM geralmente simplifica e melhora o código;
- problemas complexos para os quais não existe uma boa solução quando é utilizada uma abordagem tradicional: as melhores técnicas de AM podem encontrar uma solução melhor;
- ambientes flutuantes: um sistema de AM pode se adaptar a novos dados;
- compreensão de problemas complexos e grandes quantidades de dados.

Usualmente, considera-se rótulo o valor que se deseja obter na saída do modelo. Por exemplo, em um problema de identificação de *spam*, deseja-se classificar a qual das classes o e-mail avaliado pertence (*spam* ou *não spam*). Outro exemplo seria predizer o valor de determinada ação do mercado financeiro. Em ambos os casos, a base de treinamento deve possuir claramente o alvo desejado, isto é, o rótulo para cada instância. Quando os dados de determinado problema possuem rótulos, ele é chamado de *aprendizado supervisionado* (LORENA; GAMA; FACELI, 2000).

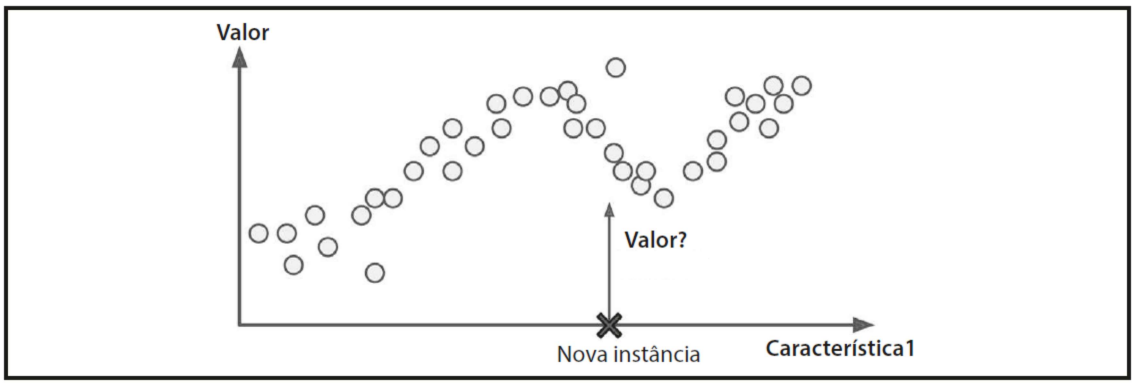
Dentro do conceito de aprendizado supervisionado, há a subdivisão entre classificação (os rótulos são classes, ou seja, valores discretos) e regressão (os rótulos são valores contínuos) (GÉRON, 2019). As Figuras 2.1 e 2.2 exemplificam cada caso. Na classificação (Figura 2.1), cada instância na base de treinamento está associada a um rótulo (ϕ ou $\surd$) e deseja-se identificar qual o rótulo na nova instância. Na regressão (Figura 2.2), o objetivo é estimar o valor numérico contínuo de saída a partir das características de entrada. Este trabalho é focado na tarefa de regressão.

Figura 2.1 – Conceito de classificação em aprendizado supervisionado.



Fonte: obtido em (GÉRON, 2019).

Figura 2.2 – Conceito de regressão em aprendizado supervisionado.



Fonte: obtido em (GÉRON, 2019).

2.1.1 Métricas de avaliação

Uma das principais medidas para avaliação de desempenho em regressões é o RMSE (sigla em inglês para *Root Mean Square Error*), que fornece uma ideia da quantidade de erros gerados pelo modelo em suas previsões, com um peso maior para grandes erros (GÉRON, 2019). O RMSE é calculado pela Equação 2.1:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{m}(y_i - \hat{y}_i)^2}{m}}, \tag{2.1}$$

onde:

- m: número de instâncias do conjunto avaliado;
- y_i: saída gerada pelo modelo para a instância i;
- $\hat{y}_i$: valor verdadeiro para a instância i;

Além do RMSE existem outras métricas, como o MAE (*Mean Absolute Error*). O MAE não pondera erros grandes com maior peso, diferentemente do RMSE. Dessa forma, considera-se o RMSE mais sensível a *outliers* do que o MAE. Com isso, neste trabalho adotou-se o RMSE e não se utilizou o MAE.

2.1.2 Divisão dos Dados em Conjuntos de Treino, Validação e Teste

A fim de averiguar o poder de generalização de um modelo, é necessário testá-lo em dados nunca antes vistos durante o treinamento. Uma das táticas usadas é separar o conjunto de dados em dois: dados exclusivos para treinamento e para teste. É importante que os dados desses conjuntos não se misturem, impedindo que amostras contidas na base de teste estejam presentes no conjunto de treinamento.

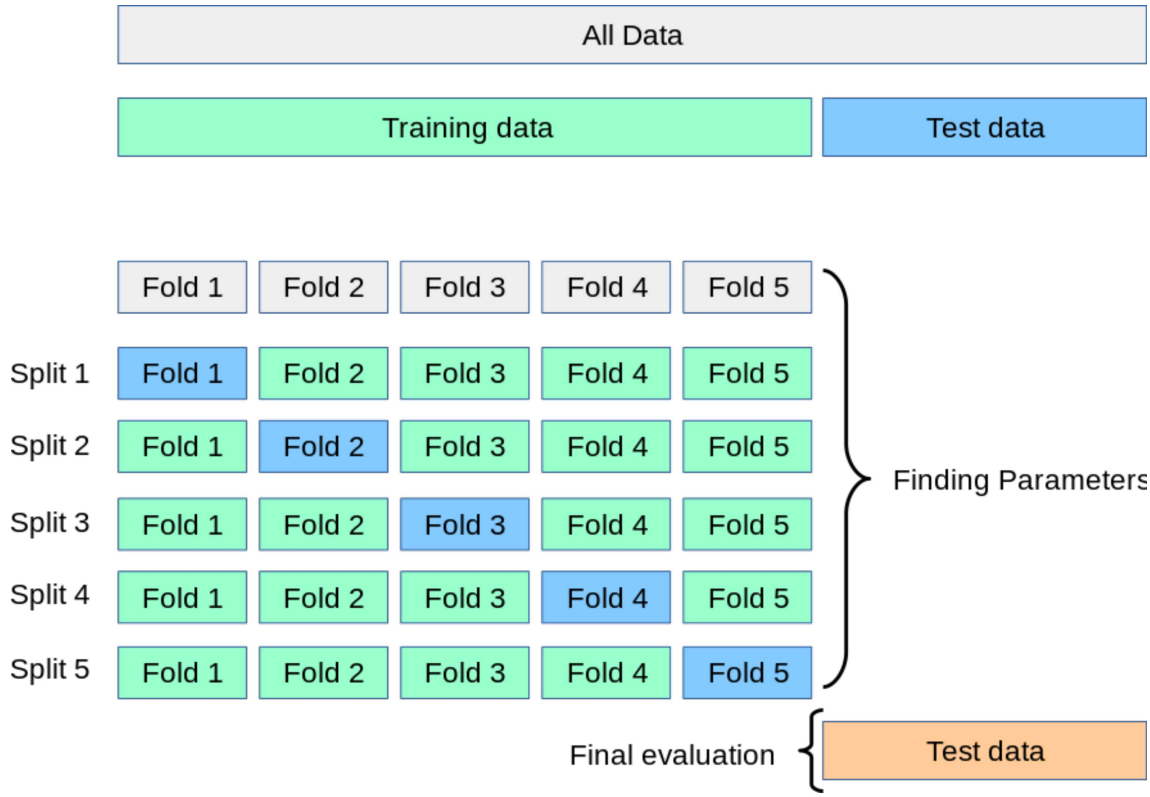
Quando se deseja encontrar os melhores hiperparâmetros para um modelo, é comum utilizar um outro conjunto de dados, chamado de conjunto de validação. Nesses casos, o modelo tem seus parâmetros ajustados utilizando a base de treinamento, a partir de um conjunto de hiperparâmetros pré-definidos (configuração), e a avaliação do desempenho dessa configuração é feita com os dados de validação. Uma vez definida a melhor configuração, verifica-se o comportamento do modelo treinado no conjunto de teste (PEDREGOSA et al., 2021).

Existem diferentes formas de avaliar um modelo pelo uso de conjuntos de treino, validação e teste. Uma das mais utilizadas é a validação cruzada *k-folds* (PEDREGOSA et al., 2021). A Figura 2.3 contém um exemplo desta validação de modelo. O conjunto total dos dados (barra superior cinza) é dividida em dois grupos: dados de treinamento (verde) e dados de teste (em azul). Em seguida, o conjunto de treinamento é dividido em k partições (neste exemplo adotou-se $k = 5$). O seguinte procedimento é executado para cada *fold*:

- o modelo é treinado utilizando dados de $k - 1$ *folds*, representados pela cor verde;
- o modelo resultante é validado com a parte restante dos dados, representados pela cor azul;
- o desempenho obtido pela validação cruzada *k-folds* é a média dos valores obtidos nos k conjuntos.

O método anterior é utilizado caso as amostras das bases de dados sejam independentes umas das outras. Entretanto, este não é o caso de dados de séries temporais. Alguns cuidados devem ser tomados quanto à seleção de dados de treinamento e teste.

Figura 2.3 – Exemplo de divisão de dados para execução da validação cruzada.

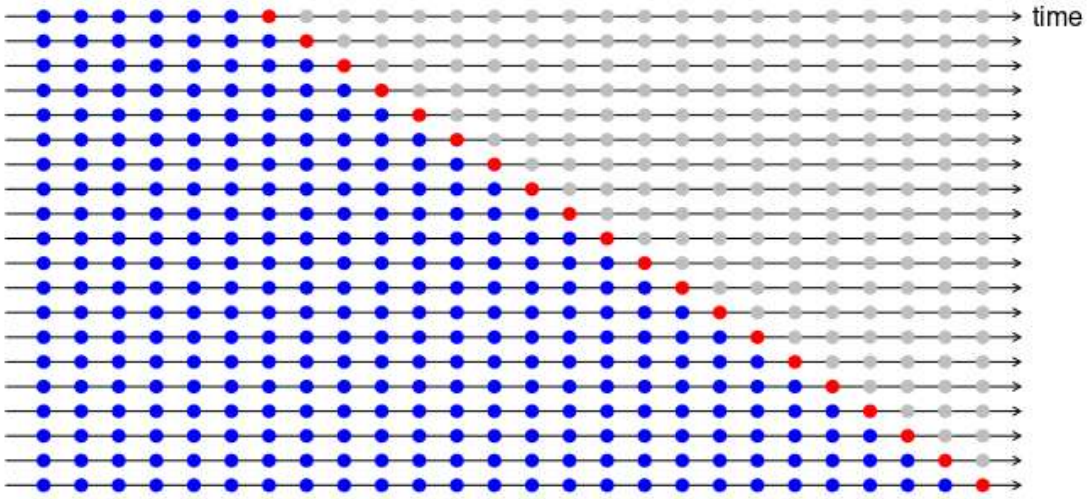


Fonte: obtido em (PEDREGOSA et al., 2021).

A análise de séries temporais lida com a extração de informações de dados coletados ao longo do tempo (DEISTLER, 2002). Os conceitos estudados nessa área têm como objetivo, baseando-se em valores do passado, estimar ou prever valores futuros ou não disponíveis, assim como entender os fenômenos que influenciam o processo em questão (BONTEMPI; TAIEB; BORGNE, 2012).

Em séries temporais, o erro de um modelo deve ser calculado com valores futuros (HYNDMAN; ATHANASOPOULOS, 2018). Por isso, a validação cruzada apresentada anteriormente viola o princípio de não usar valores futuros para treinamento do modelo. Hyndman e Athanasopoulos (2018) apresentam um exemplo da divisão dos dados para validação de modelos de séries temporais, conforme Figura 2.4. Assim como na validação cruzada *k-folds*, são gerados diferentes subconjuntos e o resultado final considerado é a média do desempenho em tais subconjuntos. Vale ressaltar novamente que as instâncias do conjunto de treinamento consistem apenas de observações realizadas anteriormente às do conjunto de teste. Este é o tipo de divisão de dados que será usado neste trabalho.

Figura 2.4 – Validação cruzada para séries temporais. Em azul, dados de treinamento; em vermelho, dados de teste.



Fonte: obtido em (HYNDMAN; ATHANASOPOULOS, 2018).

2.1.3 Regressão Linear

A análise de regressões lineares pode ser útil em várias aplicações, visto que é comum a exploração das relações existentes entre duas ou mais variáveis. A regressão linear simples considera uma única variável de entrada x (regressora ou preditora) e uma variável dependente ou de resposta y (MONTGOMERY; RUNGER, 2010). O modelo ajustado (por vezes chamado de linha de regressão estimada) é:

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x \tag{2.2}$$

onde:

- $\hat{y}$: valor de saída do modelo para a entrada x ou variável dependente;
- $\hat{\theta}_0$ e $\hat{\theta}_1$: parâmetros ou coeficientes da regressão, ajustados durante o treinamento;
- x: variável independente.

Entretanto, muitas das aplicações de análise regressiva envolvem mais de uma variável de entrada, o que caracteriza a regressão linear múltipla (MONTGOMERY; RUNGER, 2010). A Equação 2.3 mostra um exemplo com n variáveis regressoras (x_1, x_2 a x_n), coeficientes ou parâmetros do modelo ($\hat{\theta}_0$, $\hat{\theta}_1$, $\hat{\theta}_2$ a $\hat{\theta}_n$) e o valor estimado pela regressão ($\hat{y}$).

$$\hat{y} = \hat{\theta}_0 + \hat{\theta}_1 x_1 + \hat{\theta}_2 x_2 + ... + \hat{\theta}_n x_n \tag{2.3}$$

A forma vetorizada da Equação 2.3 é escrita na Equação 2.4:

$$\hat{y} = \theta^T \cdot \mathbf{x} \tag{2.4}$$

onde θ^T é o vetor transposto dos parâmetros do modelo, contendo os pesos das características, e $\mathbf{x}$ é vetor de entrada (ou vetor de características).

Com os dados de entrada e saída do modelo desejado, parte-se para o ajuste dos parâmetros, que consiste em minimizar o erro entre a previsão do modelo e os valores observados. Uma abordagem muito utilizada para encontrar os valores dos parâmetros é o uso de uma fórmula fechada (GÉRON, 2019), mostrada na Equação 2.5:

$$\hat{\theta} = (\mathbf{X}^T \cdot \mathbf{X})^{-1} \cdot \mathbf{X}^T \cdot \mathbf{y}, \tag{2.5}$$

onde $\hat{\theta}$ são os parâmetros calculados, $\mathbf{X}$ é a matriz de dados de entrada e $\mathbf{y}$ é o vetor de resposta.

2.1.4 Árvores de regressão e *random forest*

A ideia básica por trás de algoritmos baseados em árvores é dividir um problema complexo em problemas mais simples, aos quais a mesma estratégia recursivamente é aplicada (LORENA; GAMA; FACELI, 2000). Tais modelos são formados por dois tipos de nós: de divisão, com dois ou mais sucessores, e folhas, que caracterizam o fim daquele ramo.

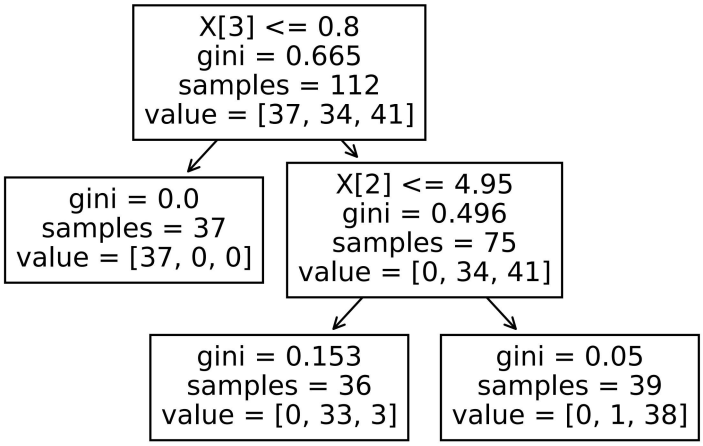
Um nó de divisão contém um teste condicional em que o atributo e o valor de comparação são encontrados de forma heurística. O objetivo é minimizar uma função de custo, comumente o ganho de informação, para tarefas de classificação, ou o erro quadrático médio (*Mean Squared Error* - MSE), para tarefas de regressão, calculados com base na variável alvo do problema. A função de custo indica quão bem a regra de divisão, composta pelo atributo e ponto de corte, consegue segregar as instâncias. A criação de nós de decisão prossegue encadeando-se até que seja atingido um dos critérios de parada, que pode ser profundidade máxima da árvore, número mínimo de amostras por folha, decremento mínimo da função de custo, entre outros.

Chega-se a um nó folha quando já não é mais possível a formação de nós de decisão em determinado ramo. Um nó folha não contém testes condicionais e suas propriedades estão relacionadas às instâncias do conjunto de treinamento que chegam até aquela ramificação. Nas árvores de regressão, o valor da saída do modelo para a instância que chegou em um nó folha será a média da variável alvo das amostras de treinamento que também chegaram nesta folha.

Uma das vantagens de modelos baseados em árvores é a facilidade de representação visual do modelo. No mesmo sentido, de acordo com Molnar (2019), tais modelos possuem boa interpretabilidade e conseguem capturar interações entre os atributos. Por outro lado, árvores falham ao lidar com relações lineares pois as divisões sempre criam uma função degrau, perdendo suavidade na saída.

A Figura 2.5 contém a representação de uma árvore de classificação. Os nós de divisão possuem um teste condicional relativos $X[n]$ onde n é um atributo. Os nós que não apresentam um teste condicional representam as folhas. No caso específico do exemplo, a função objetivo do algoritmo é o coeficiente de Gini, que pode ser substituído por outras métricas, como o MSE para problemas de regressão. Também há a indicação de quantas amostras são englobadas pelo nó e a contagem de cada classe.

Figura 2.5 – Exemplo de árvore de decisão.



Fonte: próprio autor.

A importância de um atributo em uma árvore pode ser expressa da seguinte maneira: ao percorrer todas as divisões para as quais o atributo foi usado, calcula-se o quanto ele reduziu a variância (regressão) ou índice de Gini (classificação) em comparação com o nó pai (MOLNAR, 2019). A importância é normalizada de modo que a soma total seja igual a 1. Isso significa que cada importância pode ser interpretada como uma parcela da importância geral do modelo.

De acordo com Lorena, Gama e Faceli (2000), a seleção de atributos em cada nó de divisão, inerente ao treinamento do algoritmo, produz modelos que tendem a ser bastante robustos contra a adição de atributos irrelevantes e redundantes. Contudo, estimadores baseados em árvores podem criar modelos extremamente complexos e não ter boa capacidade de generalização. Para evitar tal problema, mecanismos como poda da árvore e a adição de critérios de parada foram criados. Além disso, em virtude da tendência de *overfitting* (abordado na Seção 2.1.8), alguns algoritmos surgiram a fim de melhorar os desempenhos dos modelos criados, como o *random forest* (RF) (BREIMAN, 2001).

O objetivo de métodos *ensemble*, que é o caso do RF, é combinar a predição de diversos estimadores bases, construídos com um determinado algoritmo de aprendizado, para melhorar a capacidade de generalização e a robustez em relação a um estimador único. O RF é uma combinação de árvores de decisão ou de regressão em que cada árvore é construída usando um subconjunto dos atributos de entrada e em um subconjunto das instâncias de treinamento,

ambos escolhidos aleatoriamente. A saída do modelo é a média das predições de cada estimador individual.

Em uma investigação sobre detecção de falhas em bombas submarinas, Oliveira-Santos et al. (2016) obtiveram resultados ligeiramente superiores com o RF quando comparados com outros modelos. Em outro estudo, Branco e Gomide (2021) usaram RF para estimar a taxa de perfuração de poços do pré-sal a partir de entradas do processo.

Assim como em árvores individuais, em RF também é possível quantificar a importância relativa de cada entrada do modelo. O procedimento é feito da mesma forma: calcula-se o quanto cada atributo contribuiu para redução do erro nos nós em que foi usado; depois, soma-se toda a sua contribuição em todos os nós de todas os estimadores individuais. A importância será obtida dividindo a parcela de contribuição de cada atributo pela soma total de redução do erro.

Attanasi, Freeman e Coburn (2020) apresentam uma aplicação de RF em que os mecanismos que influenciam a produção em poços terrestres são investigados. Com isso, conseguiram identificar que as características mais importantes variavam de acordo com a localização geográfica da instalação, com particulares para cada poço. Hidayat e Astsauri (2022) utilizaram o RF para eliminação dos atributos irrelevantes na estimação da recuperação de óleo em processos com injeção de água no reservatório. Diversas características, como temperatura do reservatório, temperatura da água injetada e concentração de íons, foram utilizados como entrada e identificou-se que os sulfatos e o volume injetado tinham as maiores importâncias para recuperação de óleo.

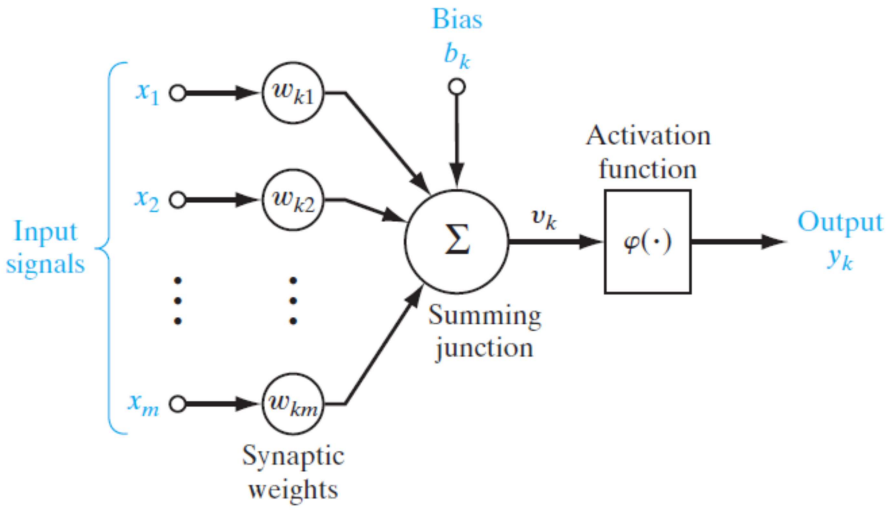
2.1.5 Redes Neurais, Recorrentes e Convolucionais

As redes neurais, por vezes chamadas de redes neurais artificiais (RNA), são técnicas de Aprendizado de Máquina baseados em otimização, isto é, otimizam uma função em seu treinamento (BRAGA, 2000; HAYKIN, 2010), ao contrário de algoritmos formados por árvores, que são considerados modelos baseados em procura.

De acordo com Haykin (2010), um neurônio é uma unidade básica de processamento de informação para a operação de uma rede neural. No diagrama da Figura 2.6 é mostrado o modelo de um neurônio, que pode ser descrito com os seguintes elementos básicos:

- conjunto de sinapses: caracterizados por pesos, um sinal x será multiplicado por um peso sináptico w;
- somador: soma as entradas x ponderadas pelos pesos sinápticos w. Esta operação é descrita como uma combinação linear;
- função de ativação: limita a amplitude da saída do neurônio. Há diversos tipos, podendo ser linear ou não;
- bias: tem o efeito de aumentar ou diminuir a entrada da rede para a função de ativação.

Figura 2.6 – Modelo de um neurônio - unidade básica das redes neurais.



Fonte: obtido em (HAYKIN, 2010).

Matematicamente, sabendo que x_j é o j-ésimo sinal de entrada (de um total de m entradas), w_{kj} é o peso do neurônio para a j-ésima entrada, u_k é a combinação linear das entradas, b_k é o bias, $\phi(\cdot)$ é a função de ativação, o sinal de saída y_k será calculado conforme as Equações 2.6 e 2.7.

$$u_k = \sum_{j=1}^{m} w_{kj} x_j \tag{2.6}$$

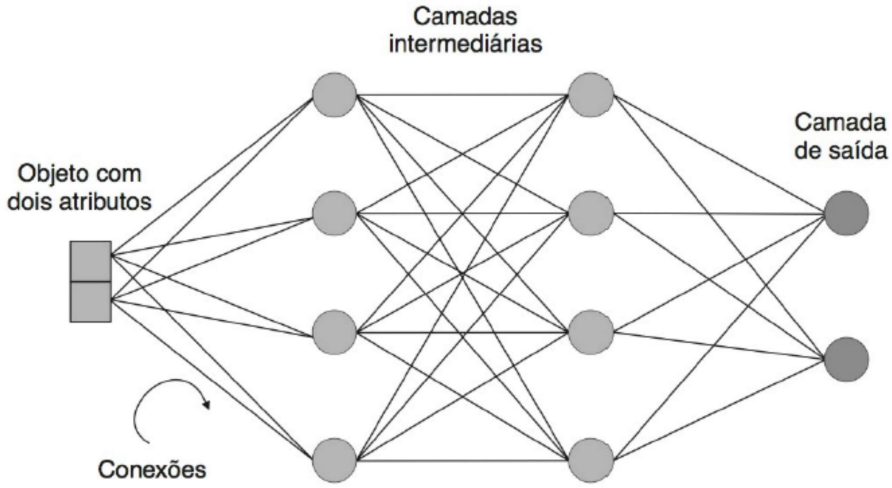
$$y_k = \phi(u_k + b_k) \tag{2.7}$$

O algoritmo de treinamento mais comum das RNA envolve uma regra de correção de erros, na qual se recorre à otimização de uma função quadrática do erro entre as respostas da RNA e os rótulos dos exemplos (LORENA; GAMA; FACELI, 2000). A função que é minimizada, chamada de função de perda ou *loss function*, pode ser customizada para cada problema. Em geral, para regressão, utiliza-se o MSE, que faz uma quantificação de quão bem a rede consegue estimar a saída verdadeira.

Segundo Lorena, Gama e Faceli (2000), os neurônios em uma RNA podem estar dispostos em uma ou mais camadas. Quando duas ou mais camadas são utilizadas, um neurônio pode receber em seus terminais de entrada valores de saída de neurônios da camada anterior e/ou enviar seu valor de saída para terminais de entrada de neurônios da camada seguinte. Esse esquema de interligação entre camadas é mostrado na Figura 2.7.

As RNA convencionais, como a representada na Figura 2.7 e classificada como *feed-forward*, não possuem retroalimentação. A informação flui da camada de entrada da rede para os

Figura 2.7 – Exemplo de RNA multicamadas.



Fonte: obtido em (LORENA; GAMA; FACELI, 2000).

neurônios da camada de saída. Para os casos de multicamadas, esse fluxo ocorre de camada em camada (LORENA; GAMA; FACELI, 2000).

Entretanto, a arquitetura *feedforward* (Figura 2.7) não lida satisfatoriamente com muitas tarefas de reconhecimento de padrões dinâmicos (BRAGA, 2000), como reconhecimento de voz, detecção de movimentos, processamento de sinais entre outros. Diante disso, as redes neurais recorrentes (RNN) surgem como boa alternativa, visto que foram projetadas especialmente para aplicações em dados sequenciais.

De acordo com Yu et al. (2019), as camadas recorrentes são formadas por células em que os estados são afetados simultaneamente pelos estados passados e pelas entradas atuais. A estrutura é parecida com a de um neurônio convencional *feedforward*, com a diferença de possuir retroalimentação. A Figura 2.8 ilustra, à esquerda, uma representação de um neurônio recebendo as entradas, gerando a saída e realizando a retroalimentação. O mesmo neurônio é mostrado de forma estendida, à direita, em que os diferentes *time steps* são explicitados.

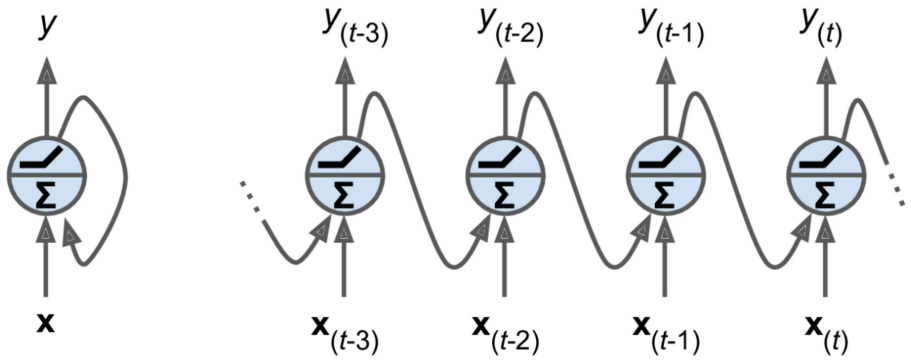
A saída do neurônio recorrente y_t no instante t será:

$$y_t = \phi(w_x x_t + w_h h_{t-1} + b) \tag{2.8}$$

onde $h_{t-1} = y_{t-1}$, sendo esta a informação da recorrência, x_t é a entrada no instante t, w_h e w_x são os pesos e b é o bias. Percebe-se que a saída do neurônio é reconectada, passando novamente pela multiplicação dos pesos e, por fim, pela função de ativação, repetindo esse processo por todos os *time steps* da entrada.

A repetição de operações mostrada na Equação 2.8 gera um problema: os gradientes tendem a explodir ou desvanecer (*exploding / vanishing gradients*) (HOCHREITER; SCHMIDHUBER, 1997; BENGIO; SIMARD; FRASCONI, 1994). A evolução temporal do erro

Figura 2.8 – À esquerda, um neurônio recorrente; à direita, o mesmo neurônio estendido ao longo do tempo.



Fonte: obtido em (GÉRON, 2019).

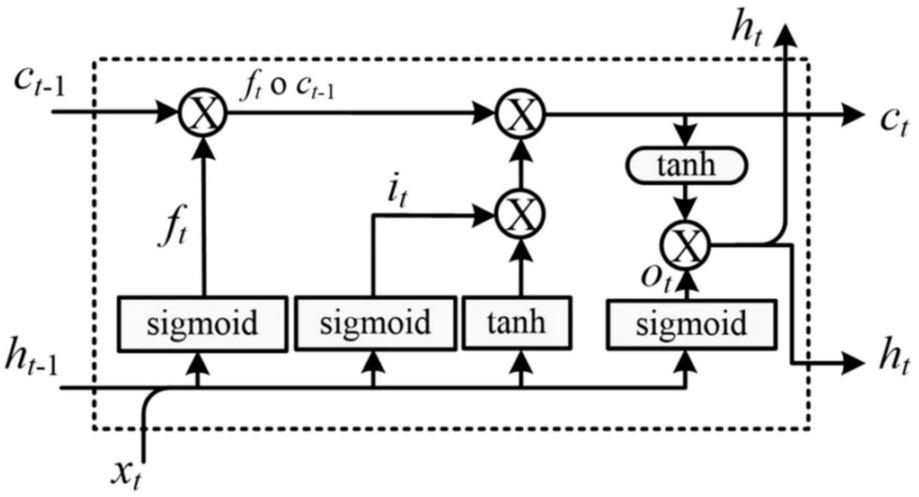
retropropagado durante o treinamento depende do valor dos pesos (HOCHREITER, 1991). Quando o gradiente aumenta exageradamente, os pesos podem oscilar de forma descontrolada; em caso de desvanecimento, o aprendizado de problemas com grandes *time lags* leva um tempo de execução proibitivo ou, em certos casos, não funciona.

Com isso, a rede neural recorrente LSTM (*Long Short-Term Memory*) foi apresentada por Hochreiter e Schmidhuber (1997), projetada para superar os problemas na propagação do erro. A ideia chave é que a rede tem a capacidade de aprender o que deve ser armazenado no estado de longo prazo e o que deve ser dispensado Géron (2019).

A Figura 2.9 contém um diagrama da estrutura detalhada de uma célula LSTM. Seguindo a explicação fornecida em (GÉRON, 2019), o estado de longo prazo c_{t-1} transpassa a célula da esquerda para direita, passando antes por um *forget gate* (f_t), onde algumas memórias são dispensadas e, então, adiciona novas memórias (i_t) pelo *input gate*. O resultado c_t é enviado diretamente para a saída, sem nenhuma transformação adicional. Assim, em cada etapa de tempo, algumas memórias são adicionadas e outras são excluídas no estado de longo prazo c_t. Além disso, o estado de longo prazo é copiado e passa por uma função de ativação tangente hiperbólica ($tanh$). Em seguida, o resultado é filtrado no *output gate* (o_t) por uma função sigmoide, em que a entrada é uma concatenação entre as entradas x_t e o estado de curto prazo h_{t-1}. Isso produz a saída da célula y_t, que é igual ao estado de curto prazo h_t.

Outra arquitetura que tem sido utilizada nos últimos anos são as redes neurais convolucionais (CNN - *Convolutional Neural Networks*). Em diversos problemas, CNN se destacam em relação às redes neurais convencionais com camadas totalmente conectadas quando são comparadas a tolerância a translações e distorções locais das entradas (LECUN; BENGIO; HINTON, 2015). De acordo com LeCun, Bengio et al. (1995), outra vantagem das CNN é que tal arquitetura não ignora a tipologia das entradas. Em imagens, sequências ou séries temporais há fortes estruturas locais, isto é, as entradas (ou *pixels*) mais próximas entre si espacialmente ou

Figura 2.9 – Estrutura detalhada de uma célula LSTM.



Fonte: obtido em (VIJAYAPRABAKARAN; SATHIYAMURTHY, 2021).

temporalmente são altamente correlacionadas. Os filtros convolucionais forçam a extração de características locais, capturando as relações entre as regiões (HAYKIN, 2010).

Em cada camada da CNN, a entrada passa por uma operação de convolução com um vetor de pesos (também chamado de filtro) para criar um *feature map* (BOROVYKH; BOHTE; OOSTERLEE, 2017). De outra forma, o vetor de pesos translada pela entrada e calcula-se o produto escalar entre a entrada e a matriz de pesos. O resultado esperado é que a camada convolucional consiga extrair as características necessárias e insensíveis à translações de forma automatizada.

As aplicações de CNN em séries temporais possuem um vasto horizonte. Liu, Hsaio e Tu (2018) propuseram uma arquitetura capaz de lidar com entradas multivariadas e atributos com atrasos. Já Zhong et al. (2019) aplicaram CNN para previsão de permeabilidade em poços de petróleo.

2.1.6 Generalização e regularização

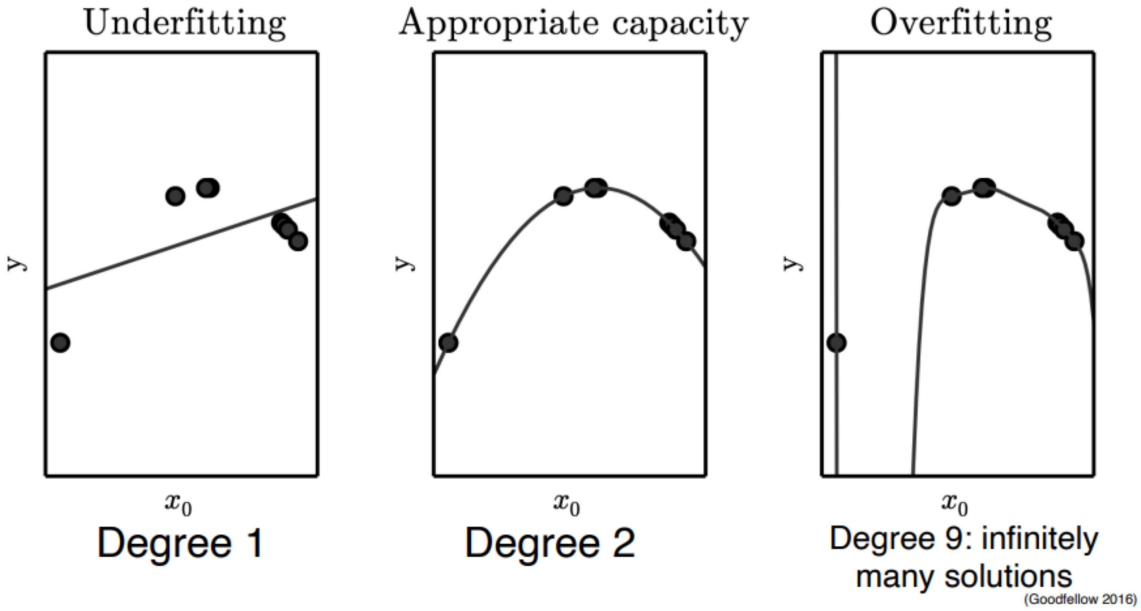
De acordo com Goodfellow, Bengio e Courville (2016), o principal desafio em Aprendizado de Máquina é o algoritmo funcionar bem com entradas não vistas anteriormente, habilidade esta chamada de generalização. Tipicamente, na construção de modelos baseados em dados, separam-se grupos de treinamento e teste; o erro no conjunto de teste, ou seja, em amostras não vistas pelo algoritmo durante o treinamento, é chamado de erro de generalização ou erro de teste.

Durante a fase de treinamento do modelo (ajuste dos parâmetros), há dois focos principais: tornar o erro no treinamento o menor possível e tornar a diferença entre os erros de treinamento e teste o menor possível (GOODFELLOW; BENGIO; COURVILLE, 2016). Tais objetivos estão associados a dois tópicos recorrentes: *underfitting* e *overfitting*. *Underfitting* (ou subajuste, em

português) acontece quando o modelo é incapaz de obter um erro suficientemente baixo no conjunto de treinamento. *Overfitting* (ou sobreajuste, em português) ocorre quando a diferença entre os erros dos conjuntos de treinamento e teste é alta.

O *bias-variance trade off* entre a tendência a *overfitting* e a *underfitting* é feito pelo controle da capacidade do modelo (BELKIN et al., 2019). Modelos com baixa capacidade têm mais dificuldade em se adaptar aos dados de treinamento; já modelos com alta capacidade tornam-se mais propensos a apenas memorizar o conjunto de treinamento, o que caracteriza *overfitting*. A Figura 2.10 contém uma representação gráfica dos conceitos abordados. Dados sintéticos foram gerados a partir da amostragem aleatória de certa função quadrática. À esquerda, é apresentada uma tentativa de ajuste dos dados a uma função linear (grau 1) e observa-se a presença de *underfitting*, ou seja, a função não é capaz de reproduzir a curvatura presente nos dados. Mais à direita, realizou-se o ajuste com uma função polinomial de grau 9 e verificou-se o *overfitting*, isto é, o modelo não é capaz de generalizar em caso de avaliação de instâncias não presentes no treinamento. Entre os extremos, um modelo com capacidade intermediária é capaz de passar por todos os pontos de treinamento e generalizar bem para amostras não vistas anteriormente.

Figura 2.10 – *Underfitting* e *overfitting* em modelos.



Fonte: obtido em (GOODFELLOW; BENGIO; COURVILLE, 2016).

Além disso, outra observação válida é que o sistema não terá bom desempenho se o conjunto de treinamento for muito pequeno ou se os dados não forem representativos, apresentarem ruídos ou com muitas características irrelevantes (GÉRON, 2019). A qualidade dos dados que são fornecidos para o modelo está intimamente relacionada ao potencial de sucesso.

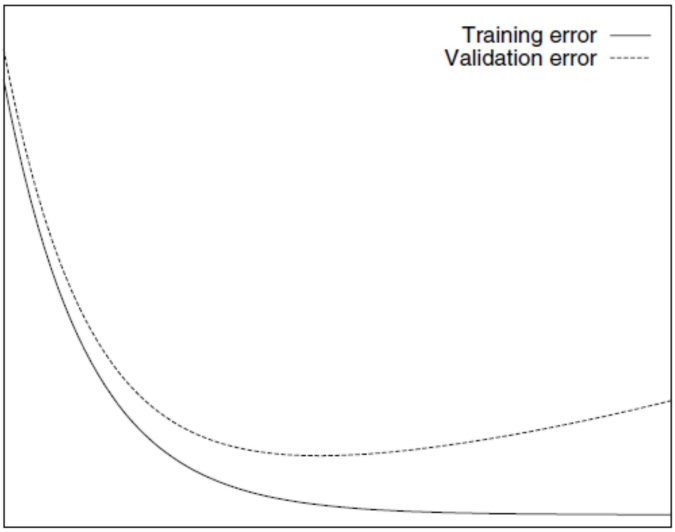
A regularização é a estratégia para restringir um modelo para simplificá-lo e reduzir

o risco de *overfitting* (GÉRON, 2019). Em regressões lineares, por exemplo, pode-se atribuir penalidades para evitar valores de parâmetros extremamente elevados. Isto faz com que o modelo torne-se menos flexível. Em algoritmos baseados em árvores, a poda é uma técnica que reduz o tamanho da árvore ao remover trechos com menor importância ou repetidos.

Dessa forma, o coeficiente de regularização em um algoritmo deve ser definido previamente e não é alterado durante o treinamento. Assim, tal coeficiente é considerado um hiperparâmetro, isto é, é um parâmetro do algoritmo e não do modelo (GÉRON, 2019).

Em redes neurais, uma das técnicas mais usadas para prevenção do *overfitting* é o *early stopping* (parada antecipada, em português). A Figura 2.11 apresenta uma curva típica dos erros dos conjuntos de treinamento e validação durante o treinamento ao longo das épocas, isto é, ao longo dos ciclos em que todas as amostras de treinamento são passadas pela rede. Sabendo da tendência ao *overfitting* dessa arquitetura devido a sua alta capacidade de adaptação à diferentes entradas, o erro no conjunto de treinamento cai continuamente (PRECHELT, 1998). Porém, no conjunto de validação isso não ocorre: a partir de dado momento, o erro de generalização começa a subir, criando um valor mínimo. A ideia do *early stopping* é identificar a partir de qual momento o treinamento do modelo deixa de ser efetivo e pará-lo, mantendo o modelo com os parâmetros que melhor atingiram o objetivo de aumentar o poder de generalização.

Figura 2.11 – Curvas de erros idealizados no conjunto de treinamento e teste. Eixo vertical: erros; eixo horizontal: épocas.

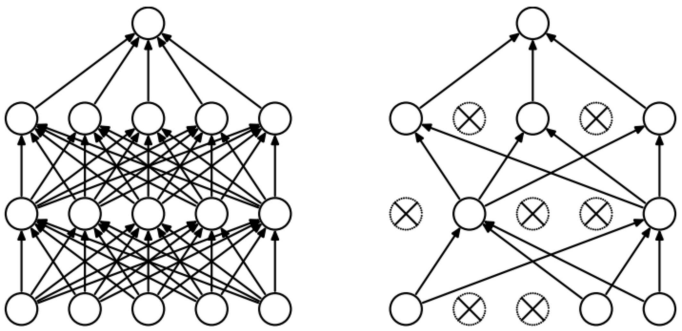


Fonte: obtido em (PRECHELT, 1998).

Além disso, especificamente para redes neurais, outra estratégia muito usada para aumento da generalização do modelo é o *dropout*, proposto inicialmente por Hinton et al. (2012) e Srivastava et al. (2014). A ideia básica é, durante o treinamento, a cada época desativar neurônios aleatoriamente com uma probabilidade definida anteriormente p. O propósito é evitar coadaptações complexas entre os neurônios, deixando assim os neurônios mais independentes uns dos

outros. A Figura 2.12 mostra as diferenças entre duas redes neurais sem e com a aplicação de *dropout*.

Figura 2.12 – Exemplificação da regularização *dropout*: a) rede neural padrão; b) rede neural depois do *dropout* em determinada época do treinamento.



Fonte: obtido em (SRIVASTAVA et al., 2014).

2.1.7 Seleção de atributos

Em aplicações industriais, a quantidade de dados obtidos do processo é grande e normalmente são armazenados em bancos de dados, sendo facilmente acessados. Entretanto, a inclusão de variáveis sem nenhum tipo de critério de qualidade em modelos de estimação baseados em dados pode deteriorar o desempenho almejado.

De acordo com Chen et al. (2020), as vantagens de realizar a seleção de atributos incluem redução do tempo de treinamento do modelo e aprimoramento na habilidade de generalização (CHANDRASHEKAR; SAHIN, 2014; GUYON; ELISSEEFF, 2003; LIU; YU, 2005). Os métodos de seleção de atributos são classificados em três grupos:

- filtros: ordenam os atributos calculando um critério independente de modelos; aqueles atributos com o valor obtido abaixo de determinado limiar são excluídos. O critério utilizado pode variar com a aplicação e o domínio do problema, sendo possível citar ganho de informação, correlação entre atributos, variância e coeficiente de variação (CV), entre outros (BOMMERT et al., 2020) (ERTUĞRUL; TAĞLUK, 2017);
- embutidos: a seleção do subconjunto de entradas é embutida ou integrada no próprio processo de aprendizagem do algoritmo. As árvores de decisão e demais algoritmos baseados em árvores são exemplos desse tipo de seleção (LORENA; GAMA; FACELI, 2000), assim como as regressões Lasso (TIBSHIRANI, 1996);
- *wrappers*: utiliza o modelo de aprendizado como uma caixa-preta para a seleção. Para cada possível subconjunto, o algoritmo é treinado e o subconjunto que apresentar a melhor com-

binação entre redução do erro e redução do número de atributos é selecionado (LORENA; GAMA; FACELI, 2000; CHEN et al., 2020).

A característica intrínseca de modelos *random forest* (RF) de buscar as melhores entradas para estimação da saída permite sua aplicação direta como seleção de atributos do tipo embutido. Dessa forma, encontra-se na literatura diversos exemplos do uso dessa técnica. Hasan et al. (2016) observaram que, em um problema de detecção de intrusão em redes de computadores, a taxa de acerto foi maior com um subconjunto dos atributos selecionados por meio do RF, além do menor tempo de execução.

2.1.8 Otimização de hiperparâmetros

Em Aprendizado de Máquina, distingue-se os conceitos de parâmetros do modelo, aqueles ajustados durante o processo de treinamento, e hiperparâmetros, aqueles que controlam o processo de aprendizado e como o modelo será estruturado. Como exemplo, em redes neurais, os valores dos pesos em cada neurônio são parâmetros e o número de neurônios em cada camada é um hiperparâmetro.

Com o uso de modelos cada vez mais complexos e hiperparâmetros podendo assumir uma grande faixa de valores (FEURER; HUTTER, 2019), a otimização nesse campo pode dispender grandes esforços. Escolhas ruins de hiperparâmetros podem fazer com que modelos não atinjam os resultados desejados.

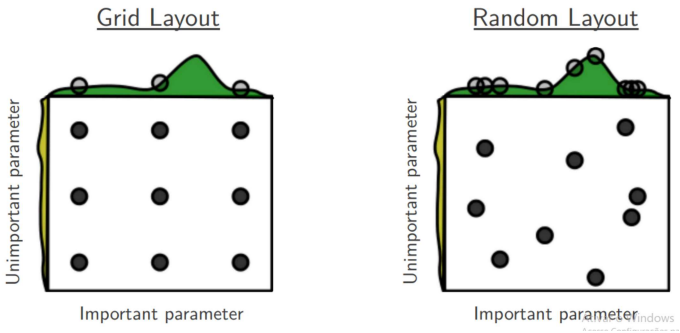
Com isso, diversas técnicas de otimização de hiperparâmetros surgiram. A primeira delas, chamada de busca exaustiva ou *grid search*, consiste na determinação do melhor conjunto de hiperparâmetros por meio da avaliação de todas as combinações possíveis no espaço de busca pré-determinado (SILVA, 2021). Essa técnica demanda um grande tempo de execução e perde eficiência à medida que a dimensão do espaço de busca aumenta, pois o número de pontos de grade cresce exponencialmente (YANG; SHAMI, 2020).

Diante dessas limitações, Bergstra e Bengio (2012) apresentaram uma alternativa chamada de *random search*, demonstrando viabilidade de implementação para problemas de alta dimensionalidade. Essa técnica busca aproveitar a ideia da baixa dimensionalidade efetiva, que ocorre quando uma função é mais sensível a mudanças em algumas direções do que em outras. Matematicamente, isso ocorre quando uma função f de duas variáveis pode ser aproximada por outra função de apenas uma variável: $f(x_1, x_2) \approx g(x_1)$.

A Figura 2.13 mostra os métodos *grid search* e *random search* para nove tentativas de otimização de uma função $f(x, y) = g(x) + h(y) \approx g(x)$. Acima, a área verde corresponde à precisão do modelo em função de $g(x)$; na lateral, é mostrado em amarelo a precisão em função de $h(x)$. No *grid search*, nove execuções são realizadas em apenas três posições diferentes para $g(x)$. Com o *random search*, todas as nove execuções exploram valores distintos de $g(x)$. Percebe-se que a grade de pontos cobre uma área igualmente distribuída no espaço original 2D,

porém as projeções em x_1 e x_2 demonstram ineficiência no subespaço. De outra forma, os pontos aleatórios são distribuídos de forma menos uniforme no espaço original. Entretanto, a efetividade é demonstrada nos subespaços.

Figura 2.13 – *Grid search* e *random search* para nove tentativas de otimização de uma função.



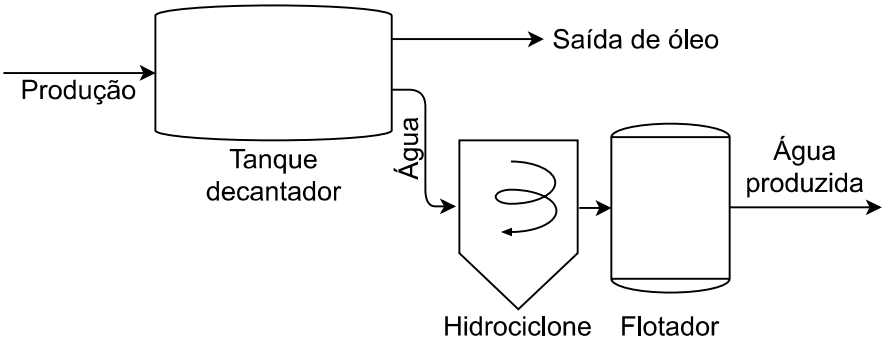
Fonte: obtido em (BERGSTRA; BENGIO, 2012).

2.2 Tratamento de água produzida em instalações *offshore* e medição de TOG

Um típico sistema de tratamento de água produzida em instalações *offshore* é apresentado na Figura 2.14. Nesse sistema, o fluido produzido (composto basicamente por óleo, água e gás residual), que já passou por processos de separação de gás, é enviado primeiramente para um tanque decantador (*settling tank*), onde ocorre a separação por gravidade. As principais influências nesta etapa são o nível da interface água/óleo, o tamanho das gotículas de óleo e a temperatura do fluido (STEWART; ARNOLD, 2011).

Posteriormente, a água é bombeada para hidrociclones, onde uma força centrífuga é aplicada para separar líquidos de diferentes densidades (VEIL, 2011). O desempenho desse

Figura 2.14 – Diagrama simplificado de um típico sistema de tratamento de água produzida.



Fonte: próprio autor.

equipamento é influenciado sobretudo pela taxa de rejeito e pela vazão de entrada (HUSVEG et al., 2007). Em (HUSVEG et al., 2007) são detalhadas curvas sobre a eficiência de hidrociclones. Stewart e Arnold (2011) apresentam princípios de operação de hidrociclones. Além disso, destaca-se que incrustações podem ocorrer no interior das linhas, sendo necessárias manutenções periódicas e injeção de inibidor de incrustação, a fim de manter a eficiência do sistema, a qual também pode ser verificada por variáveis de processo obtidas em tempo real.

Por fim, a água contendo menos contaminantes chega a etapa de flotação. Pequenas bolhas de gás são injetadas na água na entrada do vaso flotador. Ao subirem, tais bolhas associam-se a gotículas de óleo e partículas sólidas e, na superfície, esses elementos são removidos por um processo chamado *skimming* (VEIL, 2011). Dessa forma, a eficiência do tratamento está vinculada ao tamanho das bolhas e à vazão de gás, bem como à vazão de entrada de água (STEWART; ARNOLD, 2011).

Ademais, o TOG é afetado diretamente por produtos químicos utilizados, tais como: inibidores de incrustação, desemulsificantes, antiespumantes e polieletrólitos (AL-GHOUTI et al., 2019; JIMÉNEZ et al., 2018).

De acordo com Yang (2011), os métodos para medição de TOG podem ser separados em três grupos: referência, bancada (ou laboratório de bordo) e em linha (instrumento *online*), como mostrado na Figura 2.15. Os métodos de referência são utilizados por instituições governamentais e estabelecem medidas de referência abrangentes e padronizadas. Já os métodos de bancada e *online* têm ganhado espaço por possuírem procedimentos menos complexos, menores custos e geração de estimativas com menor tempo (CIRNE et al., 2016).

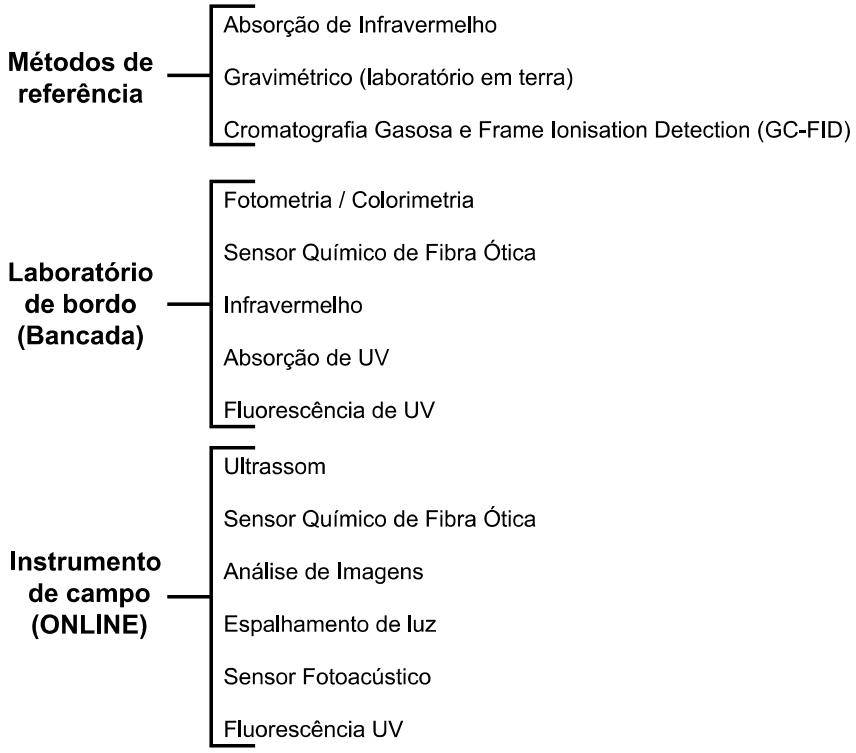
O CONAMA (BRASIL, 2007) estabelece que as concentrações de TOG devem ser determinadas pelo método gravimétrico (RICE; BAIRD; EATON, 2017). Além disso, a média mensal de TOG na água produzida descartada é limitada à 29 mg/l, com valor máximo diário de 42 mg/l (BRASIL, 2007). Esse valor médio mensal é obtido a partir de amostras diárias e estas são compostas por subamostras coletadas ao longo do dia, tipicamente sendo usadas 4 subamostras. O resultado somente é conhecido posteriormente, o que inviabiliza o controle do processo a tempo.

A medição de TOG, como enfatizado por Yang (2011), é um parâmetro método-dependente, ou seja, uma mesma amostra avaliada por diferentes métodos gera quase sempre diferentes resultados. O autor ressalta que a forma como é realizada a amostragem pode aumentar significativamente as incertezas das análises.

Com o intuito de evitar valores altos de TOG na água descartada, e controlar e otimizar o processo, são realizadas análises por outros métodos nas próprias unidades de produção *offshore*, utilizando analisadores *online* e de bancada.

No estudo em questão, dados de três métodos de análise de TOG foram utilizados como entradas do modelo: fotométrico (HACH, 2021), infravermelho (ERALYTICS, 2021) e um

Figura 2.15 – Métodos para determinação do Teor de Óleos e Graxas.



Fonte: adaptado de (YANG, 2011).

analisador *online* (Advanced Sensors, 2021), que usa a fluorescência UV como princípio de medição. Em relação aos dois primeiros, por serem de bancada, foram coletadas amostras a cada duas horas e realizada a análise no próprio laboratório de bordo. Cabe ressaltar que o monitoramento de TOG *online* tem sido um desafio nas últimas décadas, visto que apresentam dificuldades associadas a diversos fatores (principalmente em aspectos de confiabilidade e continuidade operacional) e sensibilidade a temperatura do fluido, pressão, presença de gás e particulados, tipo de óleo, tamanho das gotículas, vazão de processo e do ponto de amostragem, procedimento de coleta de amostra, entre outros.

A importância da estimativa do TOG tem resultado em contribuições recentes ao tema na literatura. Roverso (2009) propôs a criação de sensores virtuais a partir do agrupamento de redes neurais em uma unidade de produção *offshore*. Usando como referência os valores diários (mistura das quatro coletas com intervalos de seis horas entre si), o autor descreveu as dificuldades associadas às diferentes taxas de amostragem das entradas e à definição de metodologia para sincronizar os dados de processo com as medidas laboratoriais. Por fim, optou pelo emprego de médias móveis em torno dos horários de coletas.

Em outra abordagem, Júnior e Pereira (2020) investigaram a criação de modelos capazes de estimar em tempo real o TOG fotométrico e usaram como alvo as análises realizadas no laboratório de bordo. Os autores extraíram características em diferentes janelas de tempo (5, 20 e

60 minutos) antecedentes a cada coleta, avaliaram diferentes algoritmos para regressão (árvores de regressão, *bagged trees*, SVMs e redes neurais) e alcançaram os melhores resultados com *bagged trees*.

Filho et al. (2020) criaram modelos baseados em árvores de decisão e redes neurais com o objetivo de gerar previsões do TOG gravimétrico a cada cinco minutos. Considerando que a variável alvo é resultante de análises em terra e com frequência diária, a estratégia para obtenção de valores com período de cinco minutos foi a aplicação de uma técnica de interpolação. Os resultados auferidos para dados de duas plataformas indicaram a viabilidade e utilidade dos modelos baseados em dados.

3 Metodologia proposta

Antes de detalhar a metodologia proposta, algumas definições de terminologia utilizadas neste trabalho são feitas para facilitar o entendimento:

- variáveis de processo: entradas brutas recebidas do processo, como valores de instrumentos e produtos químicos;
- atributos: após a etapa de engenharia de atributos, as variáveis são transformadas em atributos. Os atributos são criados a partir de operações sobre as variáveis de processo (detalhamento na Seção 3.2);
- características: usadas como entradas para os modelos de regressão linear e de *random forest* (detalhamento na Seção 3.5). A partir de atributos na forma de séries temporais, características são extraídas para sincronização das entradas com o período de amostragem da saída (TOG gravimétrico), que é diário.

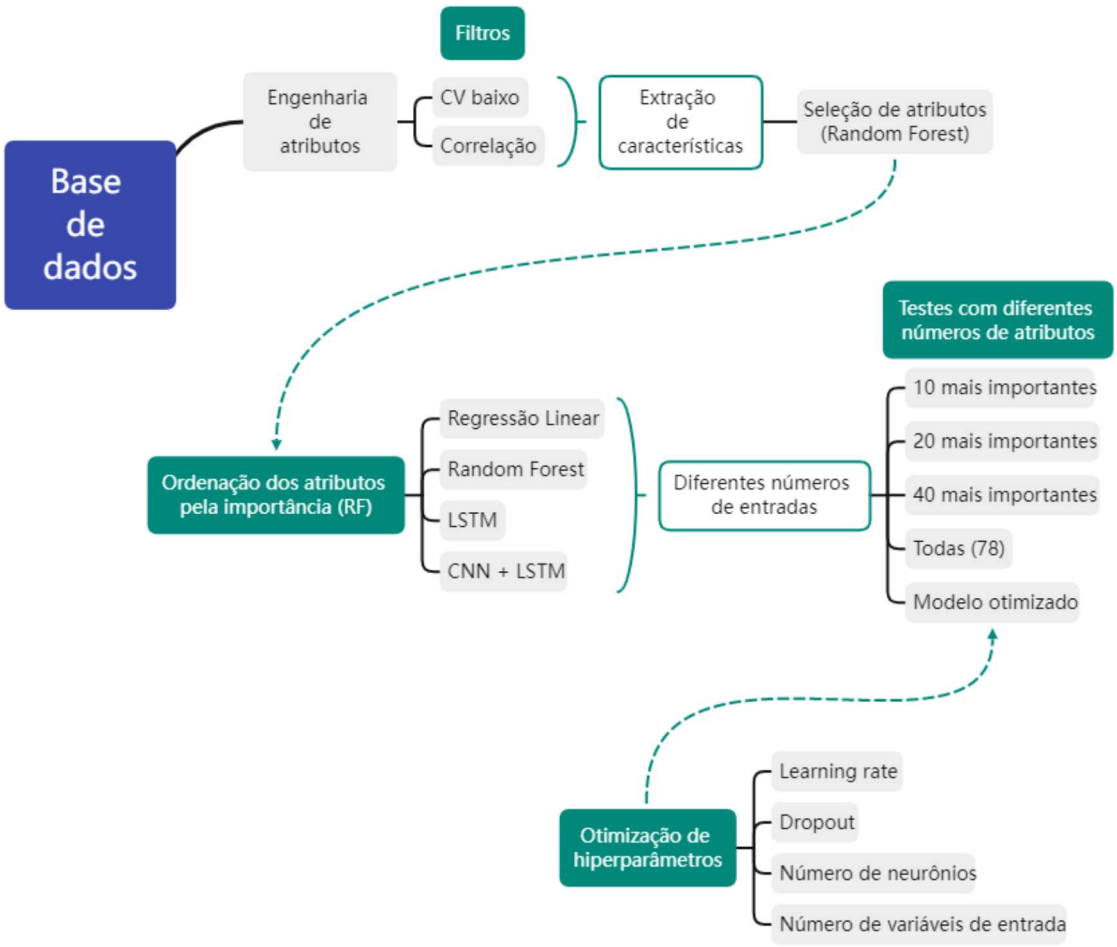
A síntese da metodologia adotada é mostrada na Figura 3.1. As próximas seções detalharão cada etapa mostrada na figura, incluindo a descrição da base de dados utilizada no trabalho. Resumidamente, o conjunto de dados passa por um tratamento inicial chamado de engenharia de atributos, em que há uma manipulação das variáveis a fim de prepará-las adequadamente para os passos seguintes. Em seguida, são realizados dois filtros para exclusão de atributos: um visa identificar atributos redundantes por meio da correlação entre eles e outro busca anomalias nas entradas por meio do coeficiente de variação (CV) baixo. Então, após a filtragem, ocorre a extração de características, etapa predecessora para criação de um *ranking* de atributos ordenados de acordo com a importância obtida via *random forest*. Com o uso do *ranking*, foram treinados quatro modelos distintos com diferentes número de entradas. Além disso, realizou-se também a otimização de hiperparâmetros e a comparação estatística dos resultados alcançados pelos modelos.

3.1 Base de dados e pré-seleção de variáveis

O conjunto de dados utilizado nos experimentos é proveniente de uma plataforma de produção *offshore* e refere-se à operação entre janeiro de 2018 e fevereiro de 2021.

Neste trabalho, as análises de TOG são relativas sempre a coletas de água descartada, isto é, da água produzida após tratamento. Apesar de serem realizadas análises em outros pontos da planta, definiu-se que essas não seriam incluídas, já que ocorrem de forma ocasional.

Figura 3.1 – Fluxograma da metodologia adotada.



Fonte: próprio autor.

Para obtenção do TOG gravimétrico, variável de interesse, subamostras coletadas durante o dia (em geral, em intervalos de seis horas entre si) são misturadas e analisadas em terra, gerando um único valor diário. O resultado é disponibilizado com defasagem de alguns dias devido a logística, justificando assim a importância de se ter outras medidas ou estimativas do TOG para eventuais ações corretivas.

No laboratório de bordo são obtidas medidas do TOG fotométrico e infravermelho (análises individuais para cada amostra) e os resultados são gerados em questão de minutos. Entre janeiro de 2018 e março de 2020, apenas a análise fotométrica estava disponível. A partir de março de 2020, o analisador com o princípio de medição por infravermelho começou a ser testado.

O analisador de TOG *online* funciona com a fluorescência ultravioleta (UV) e possui sistema de autolimpeza na câmara de medição (Advanced Sensors, 2021). Ainda assim, é comum a necessidade de manutenções preventivas e corretivas com profissionais especializados. O sinal do instrumento é enviado em tempo real para o sistema de automação, que armazena o histórico

em um PIMS (*Plant Information Management Systems*). Para a formação do conjunto de dados utilizado neste trabalho, optou-se por realizar a amostragem a cada cinco minutos, desse sinal e das variáveis de processo. A extração foi feita com a interpolação de cada variável realizada pelo próprio PIMS.

A seleção inicial das variáveis de processo incluídas no conjunto de dados foi realizada por um comitê multidisciplinar, incluindo especialistas em processamento de água, óleo e gás, ciência de dados e medições de fluidos. Além dos instrumentos relacionados à planta de tratamento da água produzida, foram coletadas medidas de variáveis relacionadas a outros trechos e equipamentos da planta de produção: chegada dos poços, separadores bifásicos, separador de teste, tratadores eletrostáticos de óleo e tanque de resíduos, totalizando 104 variáveis.

Assim como o TOG *online*, as demais variáveis de processo passam pelo sistema de automação e são armazenadas em um historiador, onde podem ser acessadas de forma segura. Apesar de serem atualizadas no historiador em questão de segundos, a frequência de amostragem utilizada foi de cinco minutos, seguindo as orientações do grupo multidisciplinar de especialistas.

Os Boletins Diários de Operação (BDOs) contêm resumos da operação da plataforma no dia. Eles possuem dados específicos de cada poço (produção líquida e bruta simuladas, BSW (*Basic Sediments and Water*) e vazão de água produzida por poço calculada com base nos valores anteriormente citados), além de informações totalizadas da unidade (produção bruta e líquida, água produzida e água descartada). Também estão inclusos dados sobre os produtos químicos injetados, a saber: desemulsificante (*topsides* e *subsea*), inibidor de incrustação e polieletrólitos. Os valores representam o volume adicionado ao processo durante o dia em questão.

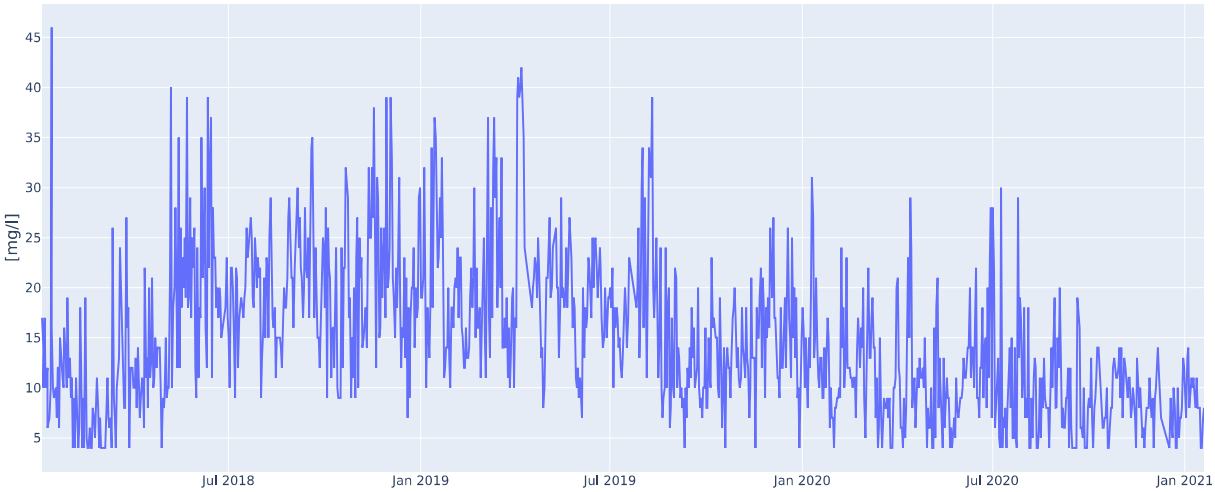
Nos momentos que o analisador de TOG *online* esteve em manutenção, sua saída ficou constante em zero. Para estes casos, os valores zerados foram excluídos, transformando-os em valores faltantes.

A Figura 3.2 contém os valores de TOG gravimétrico, variável alvo do problema, desde janeiro de 2018 a janeiro de 2021. Já na Tabela 3.1 são mostradas a quantificação da base de dados crua, indicando valores totais de dias, número de amostras, valores faltantes e quantidades de variáveis iniciais.

Tabela 3.1 – Estatísticas referentes a base de dados.

Total de dias	1113
Número de amostras	1087
Valores faltantes	26
Quantidade de variáveis	116

Figura 3.2 – Gráfico de tendência do TOG gravimétrico.



Fonte: próprio autor.

3.2 Engenharia de atributos

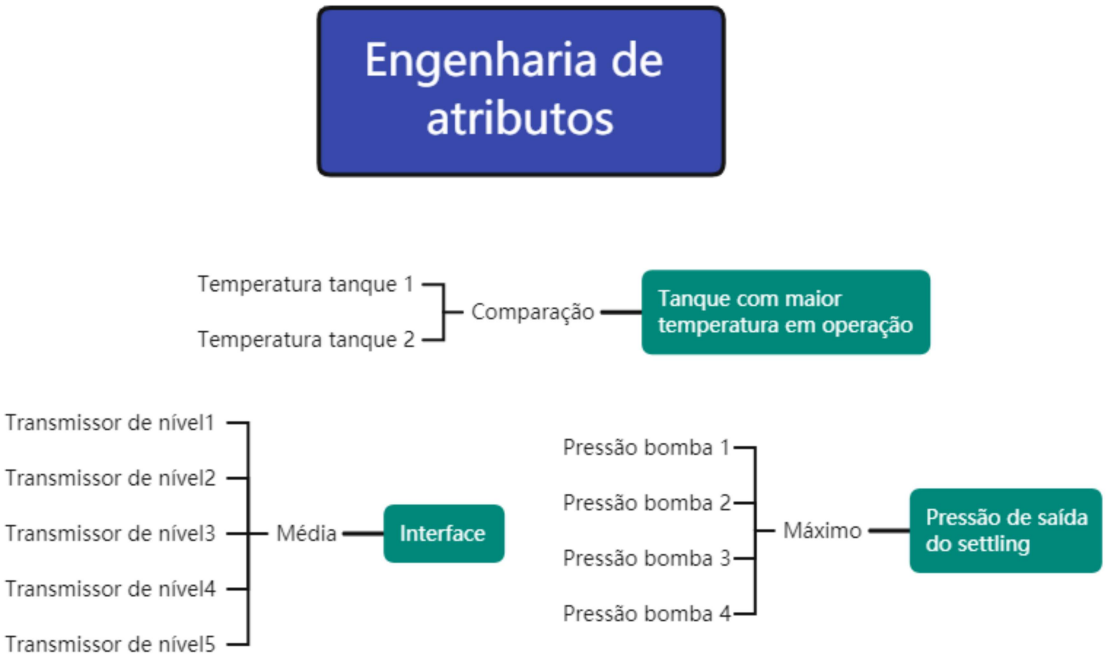
Inicialmente, executa-se a engenharia de atributos, que consiste na manipulação dos sinais de entrada para criação de atributos. Essa etapa, realizada de forma manual, tem como objetivo, por meio do conhecimento prévio do processo e das análises dos dados, a criação de novas entradas que conseguiriam expressar de forma melhor as informações contidas em dados brutos. Em alguns casos, variáveis isoladas não representam informações úteis, entretanto, ao se combinar com outros sinais do processo, o novo atributo passa a conter valores mais representativos da realidade.

A análise buscou tratar melhor processos em que há mais de um equipamento ou sistema para a mesma função. Frisando que a situação seguinte ocorre também com bombas e demais equipamentos. Por exemplo: há dois tanques decantadores, entretanto, apenas um deles opera por vez e outro permanece em *stand-by*. Entende-se que inserir dados de equipamentos fora de operação pode deteriorar o desempenho dos modelos. Com isso, realizou-se uma combinação de variáveis de entrada de modo que apenas aquelas dos tanques em operação naquele instante fossem passadas para as etapas seguintes.

Outro ponto avaliado foi quando há mais de um tramo para o mesmo processo, isto é, o fluxo é tratado de forma paralela. A título de exemplo, esse caso ocorre na medição do óleo produzido, em que há três medidores de vazão e a soma deles já contém toda a informação necessária, uma vez que não há diferença prática no valor de TOG se há mais óleo passando por um medidor de vazão do que em outros.

A Figura 3.3 mostra exemplos das técnicas utilizadas para criação de atributos. Adotou-se a combinação de variáveis de entrada por meio de somas, comparações e médias.

Figura 3.3 – Exemplos de operações realizadas na engenharia de atributos.



Fonte: próprio autor.

Todos os atributos criados são detalhados na sequência abaixo e ilustradas na Figura 3.3 . As variáveis geradoras dos novos atributos foram excluídas, mitigando, dessa forma, a presença de informações redundantes no conjunto de dados.

(a) Em cada um dos dois tanques decantadores há cinco analisadores de BSW em diferentes alturas. Um novo atributo foi criado para cada tanque, sendo o valor dele a média dos cinco analisadores. Este atributo é usado para inferência do nível da interface água/óleo. O novo atributo é um valor contínuo no intervalo entre 0% a 100% de preenchimento do tanque. As 10 variáveis originais de analisadores foram descartadas após isto;

(b) Um atributo foi criado para identificar qual tanque está em operação (somente um deles opera por vez). O tanque em operação é identificado como aquele com a maior temperatura. O novo atributo tem valor binário (0 para o tanque A, 1 para o tanque B);

(c) Um atributo foi adicionado referente à pressão de saída da bomba de água do tanque em operação. O valor deste atributo é o da variável de pressão com o maior valor entre as quatro (são duas bombas por tanque). O novo atributo tem valor contínuo na unidade de quilo Pascal (kPa). As variáveis originais de pressão de bomba foram descartadas após isto;

(d) Foi adicionado um atributo referente à temperatura do tanque em operação. O valor deste atributo é o da variável de temperatura do tanque em operação, conforme definido em (b).

Este valor é contínuo e em graus Celsius (°C). As variáveis originais de temperatura do tanque foram descartadas após isto;

(e) Um atributo foi criado para indicar a pressão de gás do tanque em operação, conforme definido em (b). O valor dele é contínuo na unidade de quilo Pascal (kPa). As variáveis originais foram descartadas após isto;

(f) Para os seguintes pontos de medição, há dois tramos: óleo do separador de teste, água do separador de produção e óleo também do separador de produção. Foram criados três atributos referentes à média das vazões dos dois tramos de cada ponto de medição, cujo valores são contínuos na unidade m^3/h. As variáveis originais foram descartadas após isto;

(g) Criado um atributo referente à soma das três vazões de óleo após tratamento, sendo o valor contínuo na unidade m^3/h. As variáveis originais foram descartadas após isto;

(h) Adicionado um atributo referente à média das últimas sete amostras diárias disponíveis de TOG gravimétrico, considerando a defasagem existente entre a coleta e o resultado da análise em terra. Seu valor é contínuo e a unidade é mg/l.

3.3 Filtragem de variáveis de entrada por CV baixo

Alimentar modelos de Aprendizado de Máquina com informações pouco úteis pode deteriorar o desempenho, além de aumentar o custo computacional. Um dos métodos para filtragem de atributos aplicados neste trabalho foi a eliminação de variáveis com baixo coeficiente de variação (CV). A análise foi executada utilizando apenas os dados de treinamento do primeiro *fold* (672 dias de um total de 1056), não havendo inclusão dos conjuntos de validação e teste. Um detalhamento melhor dessa repartição é apresentado na Figura 3.7, na Seção 3.6.

O CV foi calculado por meio da Equação 3.1:

$$CV = \frac{s}{|\overline{x}|} \times 100, \tag{3.1}$$

onde s é o desvio-padrão e $|\overline{x}|$ é o valor absoluto da média da variável no período avaliado.

Após o cálculo dessa estatística para todas as variáveis do conjunto de dados, criou-se uma lista ordenada para avaliação das entradas com menores CVs. Sabendo que é uma medida da variabilidade dos dados, um CV baixo pode indicar algum tipo de problema com o atributo, como um instrumento com valor constante durante grandes períodos de tempo, o que pode representar uma anomalia e provavelmente não contribuir para estimação do TOG.

Com isso, foi possível realizar uma análise detalhada das variáveis com menores CV, inspecionando o gráfico de tendência e o histórico de manutenção. O intuito foi identificar e eliminar variáveis não relevantes para a predição de TOG.

3.4 Filtragem de variáveis de entrada por alta correlação

Esta filtragem visou eliminar atributos muito correlacionados entre si, o que pode reduzir o esforço computacional e diminuir o número de parâmetros do modelo, minimizando a chance de *overfitting*. De acordo com Mukaka (2012), o coeficiente de correlação de Pearson r entre uma amostra de dados bivariados $(x, y) = (x_i, y_i)$ para $i = 1, 2, ...n$, onde $\overline{x}$ e $\overline{y}$ são as médias amostrais, é calculado pela Equação 3.2, :

$$r = \frac{\sum_{i=1}^{n}(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \overline{x})^2} \cdot \sqrt{\sum_{i=1}^{n}(y_i - \overline{y})^2}} \tag{3.2}$$

Calculou-se o coeficiente r para cada par de atributos. Para os pares com $r > 0,90$, excluiu-se o atributo com menor CV, conforme o calculado da Seção 3.3. Realizou-se também avaliações específicas para aqueles pares com as maiores correlações, a fim de averiguar os motivos do alto valor encontrado. A fim de realizar uma inspeção visual do experimento, criou-se uma figura que contém as correlações entre os 10 atributos de entrada escolhidos arbitrariamente.

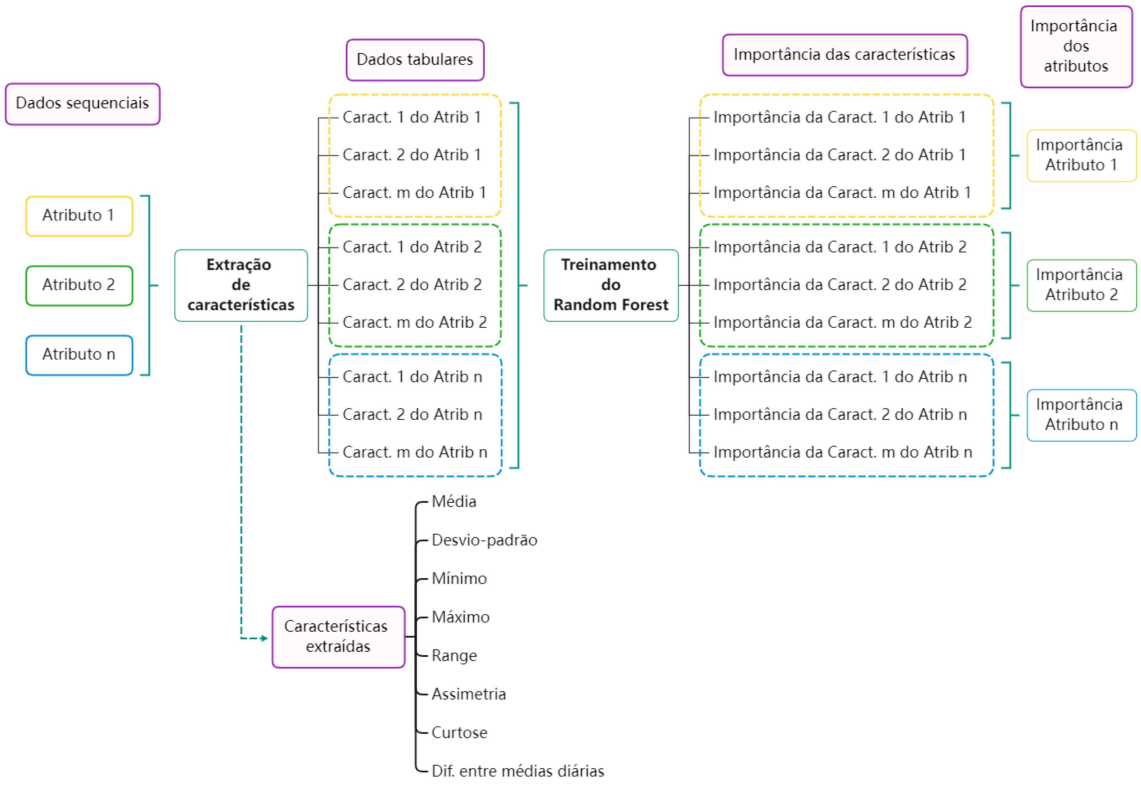
3.5 Extração de características e seleção de atributos com *random forest*

Após a exclusão das variáveis com baixo CV e das altamente correlacionadas entre si, ocorre a extração de características dos atributos. O fluxograma da Figura 3.4 mostra os passos para a extração de características e o cálculo da importância de cada atributo. De forma resumida, a partir dos atributos, até então formatados como dados sequenciais, são extraídas oito características, tornando os dados tabulares. Tais características servem como entrada para treinamento de um modelo *random forest*, sendo o TOG gravimétrico o alvo. Assim, com o modelo treinado, realiza-se a quantificação da importância das características e dos atributos. Os demais parágrafos desta seção detalham cada passo do fluxograma.

Inicialmente, os dados estão com a taxa de amostragem original de 5 minutos e, para as análises posteriores de seleção de atributos via *random forest*, são necessários dados tabulares, ou seja, características com valores únicos a serem associados à amostra diária. Portanto, para associar a valores diários do TOG gravimétrico, foram extraídas características referentes às 48 horas anteriores da amostra a ser obtida, a saber:

- média;
- desvio padrão;
- máximo;
- mínimo;

Figura 3.4 – Fluxo para criação do *ranking* de importância dos atributos via *random forest*.



Fonte: próprio autor.

- *range* (diferença entre máximo e mínimo);
- assimetria;
- curtose;
- diferença entre as médias diárias.

Com isso, pode-se usar os dados, agora tabulares, para treinamento de um modelo *random forest*. Vale reafirmar que, para essa etapa, bem como para filtragem por CV e por correlação, os dados utilizados referem-se ao conjunto de treinamento do primeiro *fold*, representado na Figura 3.7. Tal escolha visou não violar a ordem temporal da série, isto é, validação e testes foram realizados sempre com dados futuros aos do pré-processamento e treinamento do modelo.

Com o modelo *random forest* treinado, é possível obter a quantificação da importância para cada característica dos atributos (média, desvio-padrão, assimetria, etc.). Em cada nó da árvore há o número de amostras contidas e o MSE (*Mean Squared Error*), calculado com a comparação entre os valores verdadeiros do alvo e o valor médio das amostras do nó. Sabe-se que a média ponderada dos erros dos nós descendentes será sempre inferior ao erro do nó superior.

Essa redução do erro é atribuída à entrada utilizada como divisora para aquele nó. Somando a redução do erro para cada entrada individual em todas as árvores e dividindo pela redução do erro de todas as entradas, obtemos a importância da entrada, isto é, como cada uma contribuiu para explicação da variável alvo.

Ao somar a importância das características pertencentes a cada atributo, abre-se horizonte para uma análise em relação ao processo e para identificar os elementos com maior influência no TOG. Assim, ordenou-se os atributos um função da importância relativa, gerando um *ranking* que será aplicado posteriormente na seleção de entradas para os modelos.

Com o intuito de verificar se o modelo *random forest* está coerente e aderente ao conhecimento prévio do processo, gerou-se a representação de um dos estimadores do modelo, uma árvore de regressão. Por meio da análise da estrutura da árvore consegue-se verificar quais atributos estão localizados em posições mais altas, o que indica maior importância para a estimação da saída. Outros pontos de verificação são as regras aplicadas para segmentação em cada ramo da árvore (comparações do tipo "maior que" e "menor que") e os respectivos valores encontrados ao fim do treinamento.

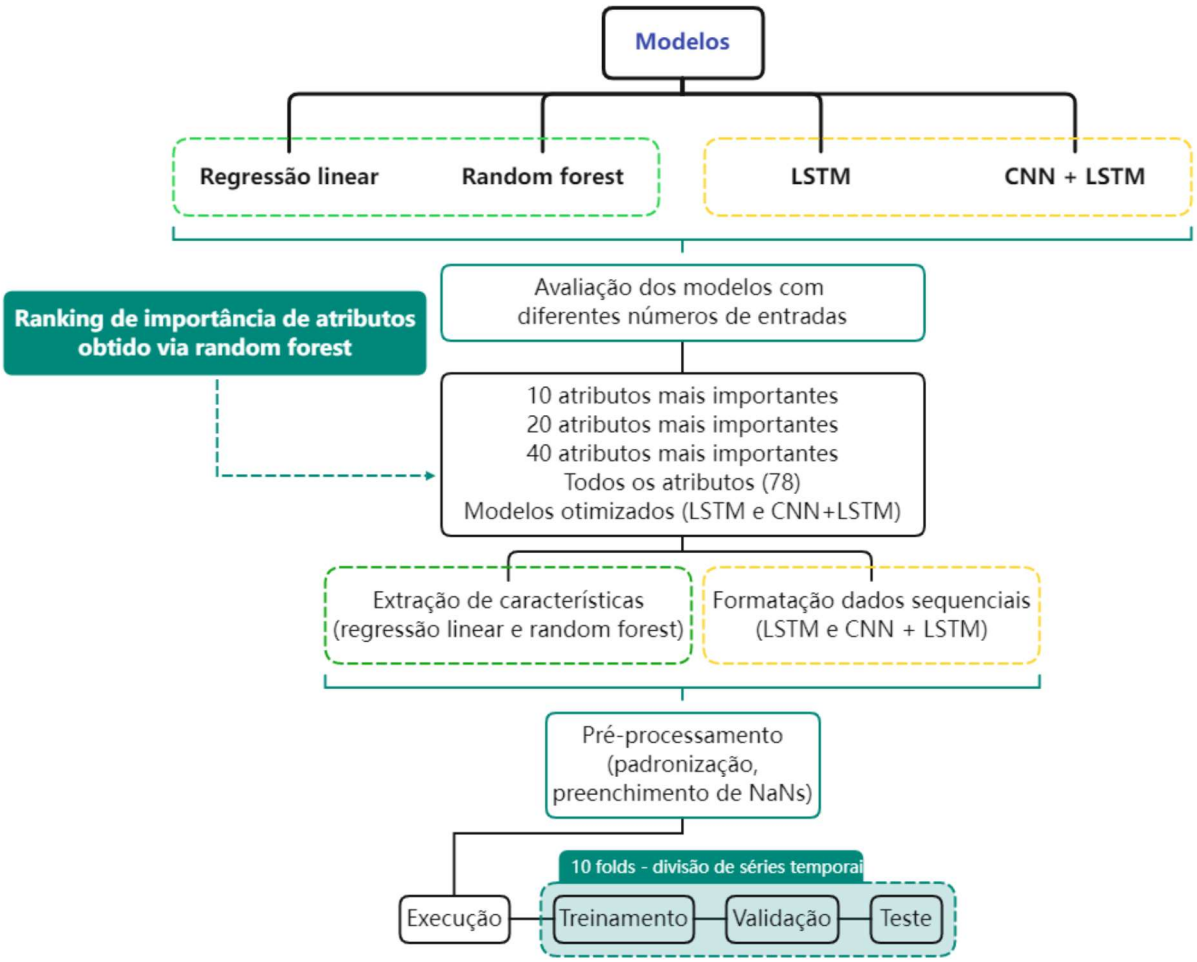
3.6 Avaliação dos modelos

Após a criação do *ranking* de importância dos atributos, é realizada a comparação de quatro modelos distintos: regressão linear, *random forest*, LSTM e CNN+LSTM. A proposta é avaliar o desempenho dos modelos com diferentes números de atributos, testando com quatro valores pré-definidos dos n atributos mais importantes: 10, 20, 40 e 78, além de modelos que passaram por otimização de hiperparâmetros. A escolha de testar, além da LSTM e CNN+LSTM, os algoritmos de regressão linear e *random forest* baseia-se em verificar se modelos mais simples atingiriam os mesmos resultados que os modelos mais complexos.

A Figura 3.5 contém um fluxograma da avaliação dos diferentes modelos. Há dois agrupamentos iniciais: modelos com entradas tabulares (regressão linear e *random forest*), circundados pela linha pontilhada verde; e modelos com entradas sequenciais (LSTM e CNN+LSTM), envoltos pela linha pontilhada amarela. Após a escolha do modelo desejado e usando o *ranking* de importância de atributos resultante do diagrama da Figura 3.5, chega-se à configuração dos atributos de entrada selecionados, cujo número pode variar entre 10, 20, 40 e 78 (todos). Especificamente para LSTM e CNN+LSTM, há também o teste de modelos com hiperparâmetros otimizados, sendo que o número de atributos de entrada foi um dos alvos da otimização.

Na etapa seguinte, há uma bifurcação: para os algoritmos que requerem entradas tabulares, há a extração de características; para aqueles que usam entradas sequenciais, procede-se a formatação adequada (Figura 3.6). Para o primeiro caso, por exemplo, se a opção "10 atributos mais importantes" for selecionada, serão extraídas as oito características de cada atributo e, portanto, o modelo teria 80 entradas.

Figura 3.5 – Fluxograma mostrando o processo de avaliação dos diferentes modelos.

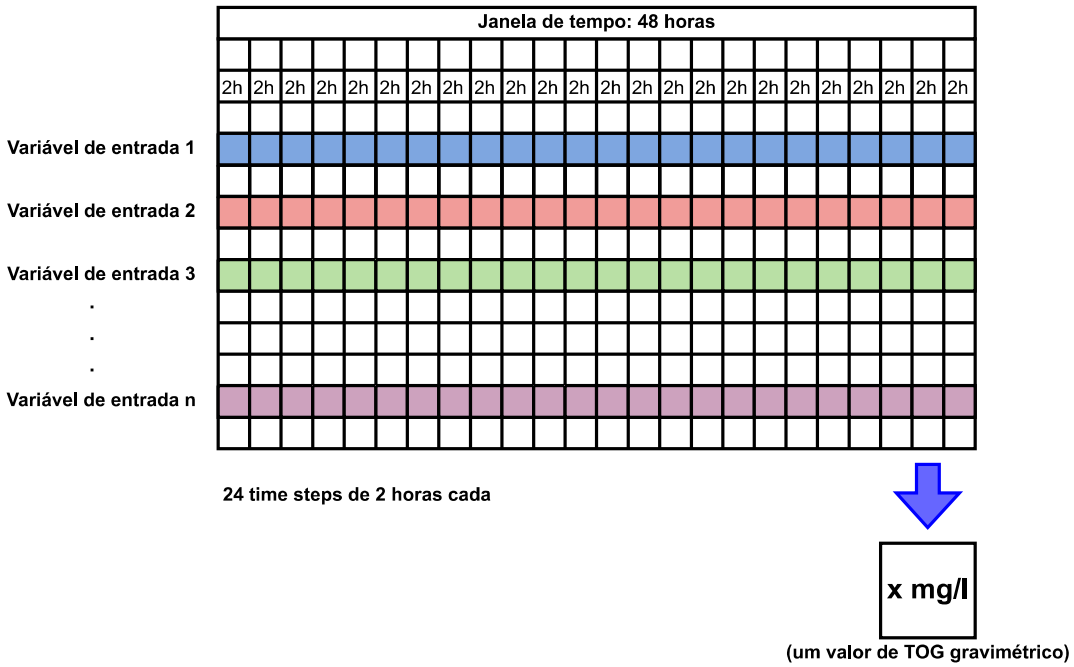


Fonte: próprio autor.

Já a formatação dos dados sequenciais, em amarelo na Figura 3.5, aplicáveis a LSTM e a CNN+LSTM, foi realizada de acordo com o indicado na Figura 3.6. Optou-se por padronizar em 2 horas a frequência de amostragem de todos os atributos de entrada. Para as variáveis de processo, que são amostradas a cada 5 minutos, assim como para os atributos calculados a partir delas (descritos na Seção 3.2), foi usada uma janela deslizante sem sobreposição para o cálculo de suas médias a cada 2 horas. Para os dados diários de produção, seus valores foram repetidos por 12 *time steps*, de forma a ter um valor a cada duas horas. Para cada valor de TOG gravimétrico, associaram-se os dados das 48 horas anteriores, totalizando assim 24 *time steps* (ver Figura 3.6). Foi associada uma matriz de dados deste formato para cada amostra de TOG gravimétrico, que é disponibilizada diariamente.

Após a formatação específica para entradas tabulares e sequenciais, há uma unificação dos procedimentos das etapas seguintes para avaliação dos modelos. Os procedimentos descritos a seguir, referentes ao pré-processamento e a execução em si do treinamento, são aplicáveis aos

Figura 3.6 – Exemplo de matriz de entrada contendo n atributos, divididos em 24 *time steps* de 2 horas cada, totalizando uma janela temporal de 48 horas, que se associa a um único valor de TOG gravimétrico.



Fonte: próprio autor.

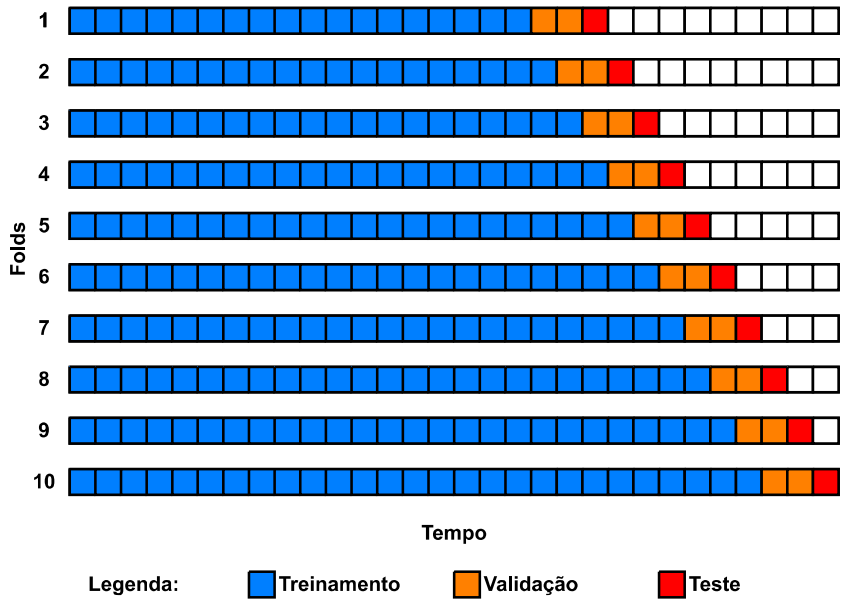
quatro tipos de modelos.

Divisão dos dados: a separação dos dados em treinamento, validação e teste foi realizada conforme a Figura 3.7, uma vez que os valores diários de TOG formam uma série temporal. Na representação gráfica, cada bloco corresponde a um conjunto de 32 dias, isto é, 32 amostras de TOG gravimétrico, e cada amostra do TOG está associada a uma matriz de entrada (como a da Figura 3.6). É importante destacar que o conjunto de validação é sempre subsequente ao de treinamento, assim como o de teste é subsequente ao de validação, a fim que de que o algoritmo não seja treinado com dados futuros e testado com dados passados. Além disto, o tamanho do conjunto de treinamento cresce ao passar para o *fold* seguinte.

Pré-processamento: os dados de entrada são padronizados com *z-score* (subtrai-se a média e divide-se pelo desvio padrão de cada atributo). Ressalta-se que a média e o desvio padrão são obtidos sobre o conjunto de treinamento e a transformação aprendida é aplicada nos conjuntos de validação e de teste. Na sequência, por escolha de projeto, os valores calculados são limitados a uma faixa entre -7 e 7 para evitar extremos, e, por último, os dados faltantes são substituídos pelo valor -10, difenciando-os do restante.

Modelos com entradas tabulares: a regressão linear foi ajustada por meio do método dos mínimos quadrados, não havendo, portanto, hiperparâmetros. Já o *random forest* foi configurado com: número de estimadores = 500; profundidade máxima = 7; mínimo de amostras por folha =

Figura 3.7 – Separação dos dados entre treinamento, validação e teste para os 10 *folds*.



Fonte: próprio autor.

3.

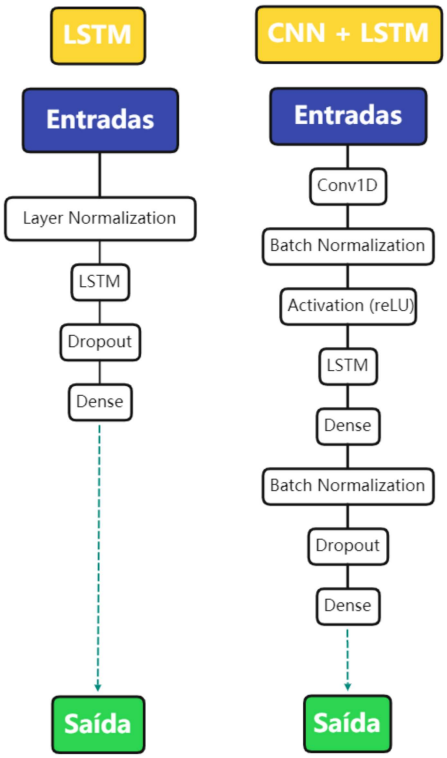
Modelos com entradas sequenciais: foram propostos dois modelos baseados em LSTM (Figura 3.8), sendo que a principal diferença entre eles é a existência ou não de uma camada convolucional na entrada. O modelo com essa camada é referenciado neste documento por CNN+LSTM e o outro por LSTM.

Na Figura 3.8 existem as seguintes camadas:

- Conv1D, camada responsável pela convolução dos dados de entrada CNN+LSTM, buscando realizar a extração de características de forma automática (ALZUBAIDI et al., 2021) (LECUN; BENGIO et al., 1995);
- *Layer Normalization* e *Batch Normalization*, responsáveis por normalizar os dados de entrada, tornando o treinamento das redes neurais mais estável (ALZUBAIDI et al., 2021) (IOFFE; SZEGEDY, 2015);
- *Dropout*, que auxilia na regularização da rede para evitar *overfitting*;
- *Activation*, que usa a função de ativação não-linear ReLU (*Rectified Linear Unit*), capaz de diminuir a chance de *overfitting* e reduzir o tempo de treinamento (KRIZHEVSKY; SUTSKEVER; HINTON, 2012) e (NAIR; HINTON, 2010);
- LSTM, camada referente à RNN em si, que recebe os dados de forma sequencial (24 *time steps*) e processa de forma a aprender os padrões nos sinais de entrada;

- *Dense*, também chamada por vezes de *fully connected layer*, camada de neurônios similar ao de uma rede neural clássica, que consegue se ajustar a mapeamentos não-lineares entre entrada e saída.

Figura 3.8 – Modelos com entradas sequenciais: “LSTM”, ‘a esquerda, e “CNN+LSTM”, ‘a direita.



Fonte: próprio autor.

Para o treinamento, foi usado um número máximo de 5000 épocas, função de perda MSE (*Mean Squared Error*) e *max norm* nas camadas LSTM e *Dense* com o valor $4,0$. Foi usado *early stopping* para interromper o treinamento caso o erro na validação não diminua após 100 épocas.

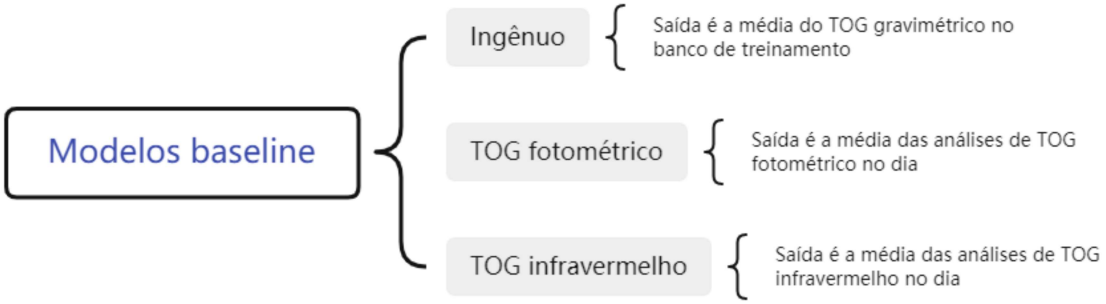
Antes da otimização dos hiperparâmetros, os modelos sequenciais LSTM e CNN+LSTM foram testados de acordo com a configuração mostrada na Tabela 3.2.

Tabela 3.2 – Hiperparâmetros utilizados para os modelos sequenciais LSTM e CNN+LSTM.

Hiperparâmetro	LSTM	CNN+LSTM
Learning rate	10^{-3}	$5 \cdot 10^{-5}$
Dropout	0,4	0,4
Neurônios LSTM	256	16
Neurônios dense	256	16
Número de filtros	-	16
Tamanho do *kernel*	-	3
Strides	-	2

Modelos *baselines*: os modelos propostos foram comparados a três outros: um preditor *baseline* ingênuo, que fornece como previsão a média dos valores de TOG gravimétrico do conjunto de treinamento (valor constante em todas as previsões do *fold*); outro que faz a previsão usando o valor médio do TOG fotométrico do dia e, por fim, um que faz a previsão usando o valor médio do TOG infravermelho no dia. Vale ressaltar que o TOG fotométrico e o infravermelho são utilizados pela equipe de operação para monitorar o processo de tratamento de AP, sendo essas as duas melhores estimativas do TOG gravimétrico viáveis atualmente.

Figura 3.9 – Modelos *baselines* adotados.



Fonte: próprio autor.

Métricas: as métricas de avaliação utilizadas foram o RMSE (*Root Mean Square Error)* e a qualidade do modelo (fit). O RMSE de determinado modelo é calculado ao se comparar a saída gerada pelo modelo e o valor da variável alvo (TOG gravimétrico). Já a qualidade do modelo é obtida por meio da Equação 3.3:

$$fit = \left(1 - \frac{\overline{e}_M}{\overline{e}_B}\right) \times 100, \tag{3.3}$$

em que $\overline{e}_M$ e $\overline{e}_B$ são as médias do RMSE nos 10 *folds* de teste do modelo avaliado e do preditor *baseline* ingênuo, respectivamente. Quanto maior o valor dessa métrica, melhor é o ajuste entre o valor de referência de TOG gravimétrico e o valor gerado pelo modelo em questão.

Para avaliar a dispersão dos erros e permitir a comparação das medianas, foram gerados *boxplots* dos erros RMSE calculados para os 10 *folds* de teste para os modelos propostos e para os três preditores *baselines* (preditor ingênuo, TOG fotométrico e TOG infravermelho). Além disso, para comparação das médias, foi realizado testes de Wilcoxon para amostras pareadas.

3.7 Otimização de hiperparâmetros

A parte final do experimento consistiu em realizar a otimização dos hiperparâmetros dos modelos LSTM e CNN+LSTM. Sabendo das desvantagens apresentadas pelo método *grid search*, principalmente devido ao número de combinações de valores crescer exponencialmente com a dimensão do espaço de busca (FEURER; HUTTER, 2019), optou-se por uma abordagem com o

random search (BERGSTRA; BENGIO, 2012). Foram realizadas 120 execuções e escolheu-se como melhor conjunto de hiperparâmetros aquele que o modelo treinado atingiu a menor média do RSME dos 10 *folds* do conjunto de validação.

Para a LSTM, os hiperparâmetros otimizados foram: *learning rate*, *dropout*, número de neurônios da LSTM e número de neurônios da camada *dense*. Para a CNN+LSTM, entraram na otimização: número de filtros, tamanho do *kernel* e número de *strides*.

Em ambos os casos, optou-se também por otimizar o número de atributos de entrada, ou seja, esse fator entrou como um hiperparâmetro. Este poderia assumir um dos valores presentes numa lista pré-definida (5, 10, 20, 30, 40, 50, 60, 70 e 78) e representaria o número de atributos usados no modelo, utilizando os n mais importantes, seguindo a ordenação do *ranking* de importância resultante do treinamento do *random forest*. Por fim, gerou-se um gráfico *boxplot* contendo os melhores modelos de *random forest*, LSTM e CNN+LSTM que atingiram resultados melhores que os *baselines*.

4 Estudo de caso

Este capítulo apresenta os resultados provenientes do emprego dos métodos abordados no Capítulo 3. As Seções 4.1, 4.2, 4.3 e 4.4 referem-se ao tratamento inicial dos dados, filtragem e seleção de atributos. Por fim, a Seção 4.5 expõe os resultados dos modelos treinados.

4.1 Engenharia de atributos

A Tabela 4.1 apresenta o número de variáveis de entrada separados por tipos, antes e depois da engenharia de atributos. Observa-se que as alterações ocorreram apenas nas variáveis de processo.

Tabela 4.1 – Número de variáveis por tipo antes e depois da engenharia de atributos.

Tipo	Antes	Depois
Medições de TOG	3	3
Variáveis de processo	105	85
Boletins Diários de Operação	4	4
Produtos químicos	4	4
Total	116	96

Destaca-se que já nesta fase foi possível eliminar algumas variáveis a partir das operações descritas na Seção 3.2 (comparações, cálculo de médias e somas). Partindo do pressuposto que a criação de um atributo por meio de manipulações de variáveis condensaria as informações de determinado processo, optou-se por excluir as variáveis usadas no cálculo do novo atributo.

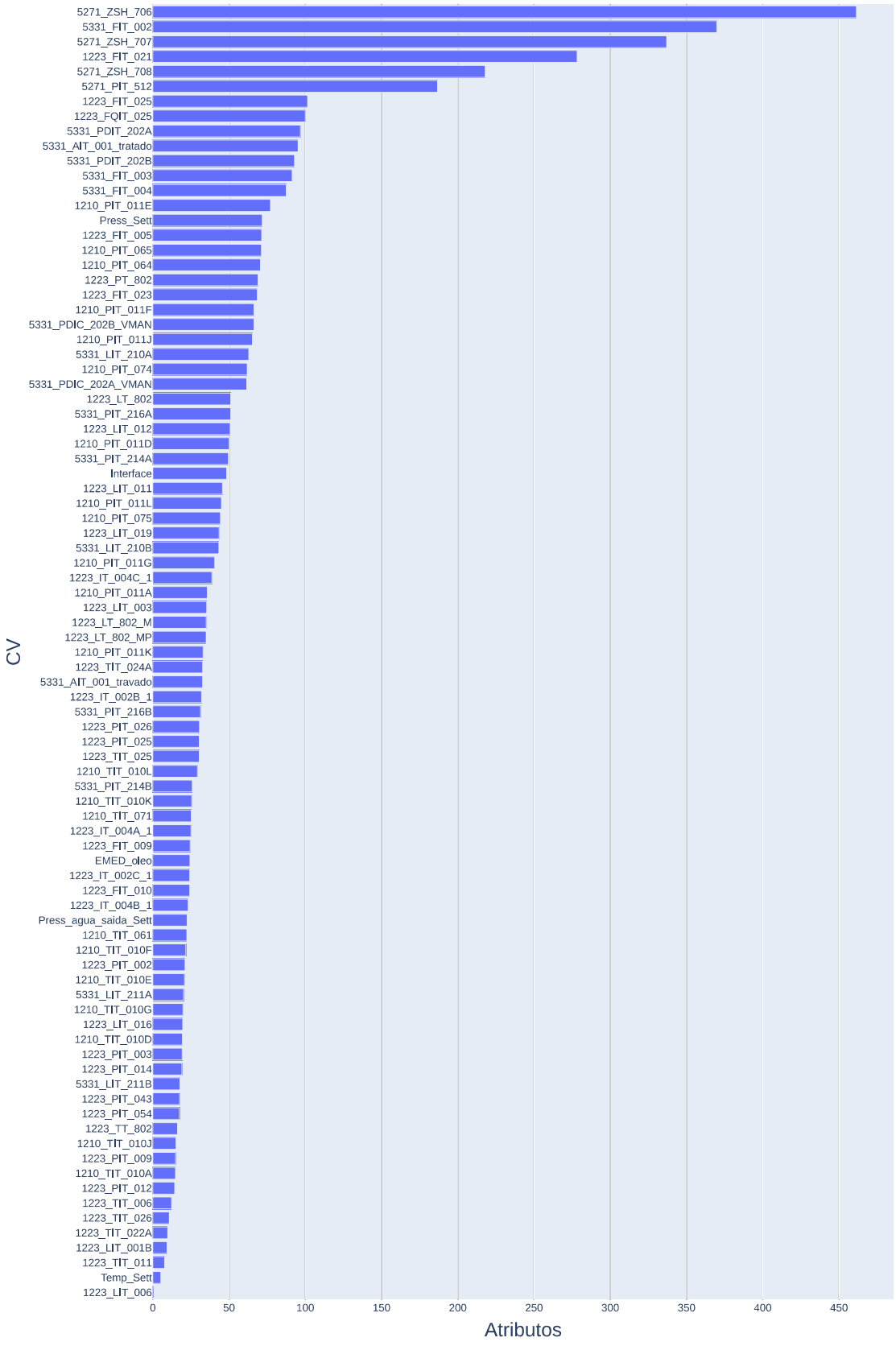
As variáveis de processo, inicialmente 105, obtidas de instrumentos e válvulas desde a chegada dos poços até o tratamento final da água e do óleo, puderam ser reduzidas a 85, representando a eliminação de 20 entradas. Levando em consideração todos os tipos de dados, as 116 variáveis de entrada iniciais chegaram a 96 ao fim desta etapa.

4.2 Filtragem de variáveis de entrada por CV baixo

Realizou-se o cálculo do CV para todos os atributos de entrada, e os resultados são mostrados na Figura 4.1. Para melhor visualização, os valores dos cinco menores CV e os respectivos atributos são mostrados na Tabela 4.2.

O gráfico da variável com menor CV, 1223_LIT_006, foi avaliado e verificado seu comportamento ao longo do tempo (Figura 4.2). Observou-se que seu valor ficou praticamente constante durante a maior parte do período de treinamento. Ao verificar o histórico de manutenção do equipamento, confirmou-se que o transmissor de nível estava avariado e só voltou a operar

Figura 4.1 – Coeficientes de variação (CV) dos atributos.



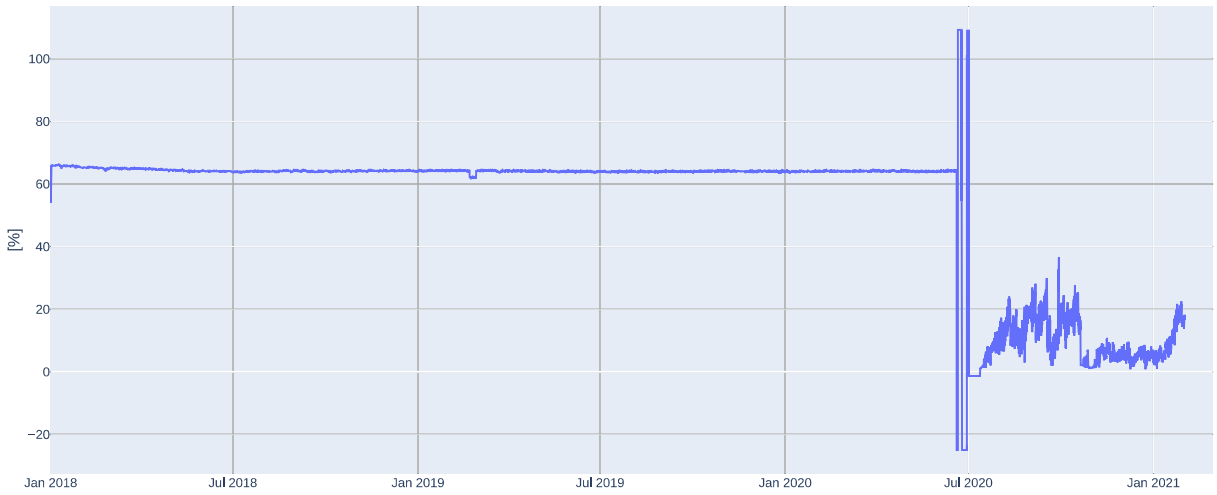
Fonte: próprio autor.

Tabela 4.2 – Variáveis com menores coeficientes de variação (CV).

Variável de entrada	CV
1223_LIT_006	0,81
$Temp\_Sett$	5,19
1223_TIT_011	7,70
1223_LIT_001B	9,28
1223_TIT_022A	9,76

normalmente a partir de julho de 2020. Desta forma, tal entrada foi eliminada e não passou para as etapas seguintes.

Figura 4.2 – Gráfico de tendência do atributo 1223_LIT_006, que apresentou o menor CV.

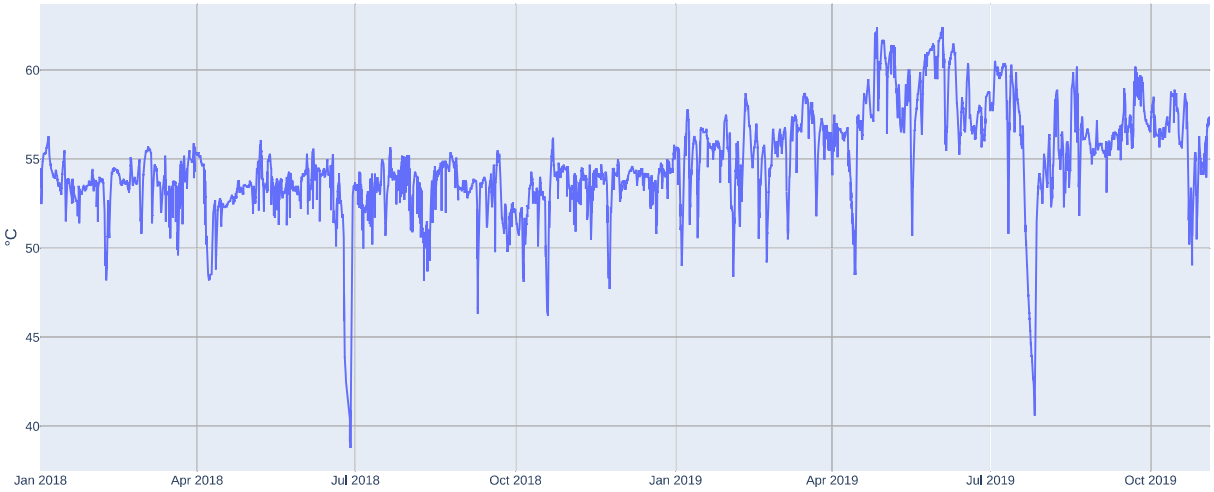


Fonte: próprio autor.

A variável com o segundo menor CV foi o atributo criado $Temp\_Sett$ (Figura 4.3), relativo à medição de temperatura do tanque decantador em operação. Verificou-se que seu comportamento não apresentava anomalias e o CV baixo está relacionado às próprias características do tanque (grande capacitância) e também à grandeza medida, cabendo ressaltar que a temperatura, em geral, varia com menor velocidade do que outras grandezas (como vazão e nível, por exemplo). Com isso, optou-se por manter o atributo $Temp\_Sett$ para as etapas seguintes.

Na avaliação dos demais atributos com baixo CV não foram encontradas mais anomalias. Optou-se por uma abordagem mais conservadora em relação à exclusão de atributos nessa etapa, sabendo que mesmo com um CV baixo, certa entrada pode ser útil na criação de modelos preditivos. Com isso, a filtragem por CV baixo foi capaz de eliminar uma variável confirmadamente anômala.

Figura 4.3 – Gráfico de tendência do atributo $Temp\_Sett$, que apresentou o segundo menor CV.



Fonte: próprio autor.

4.3 Filtragem de variáveis de entrada por alta correlação

Calculou-se uma matriz com o módulo da correlação entre todos os pares de atributos. A título de exemplo e com o intuito de permitir uma visualização gráfica do experimento, foi gerada a Figura 4.4. Para esta figura, foram utilizados apenas 10 atributos escolhidos arbitrariamente (o uso de todas as variáveis tornaria inviável a visualização e bom entendimento). Nela, quanto maior a intensidade do vermelho, maior a correlação absoluta entre os atributos da linha e coluna do quadrante. De forma contrária, quadrantes com cores brancas representam uma baixa correlação.

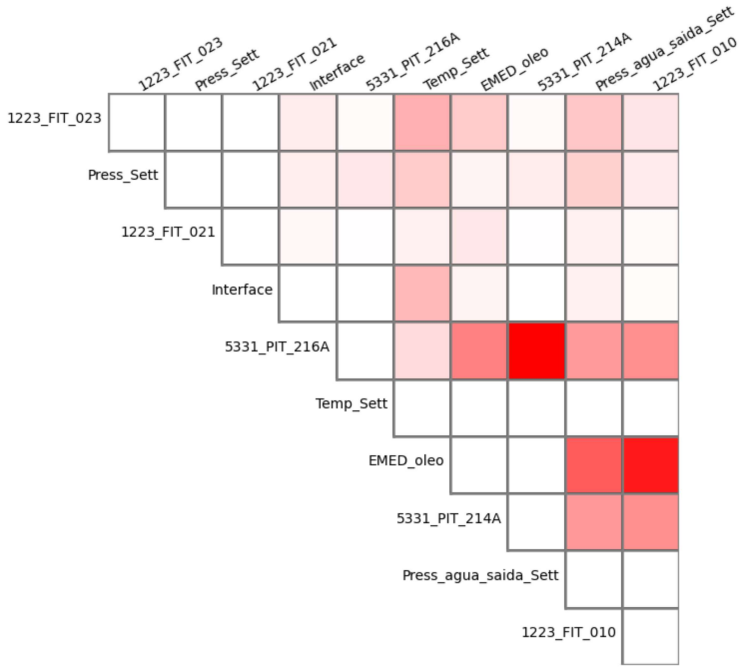
Percebe-se que dois pares se destacam: 5331_PIT_214A e 5331_PIT_216A, que apresentaram uma correlação de $1,00$; e o par $EMED\_oleo$ e 1223_FIT_010, com o valor $0,89$. Os demais pares de atributos apresentaram correlações menores, como pode ser identificado por meio de inspeção visual.

A lista com os pares de variáveis com correlação entre si maior que o limiar estabelecido 0,9 é apresentada na Tabela 4.3. Para manter somente uma das variáveis correlacionadas, optou-se, por critério de projeto, por excluir aquela com menor coeficiente de variação, conforme Seção 4.2.

O primeiro par de variáveis altamente correlacionados são referentes às medições de nível de um mesmo tanque, 1223_LT_802M e 1223_LT_802MP. Entretanto, a segunda mostra o nível em termos de ulagem (distância do topo do tanque ao nível de líquido). Dessa forma, já se esperava uma alta correlação entre elas e a inclusão de apenas uma delas é suficiente para adicionar os dados referentes ao nível do tanque, sem perdas significativas de informação.

O segundo par com maior correlação, 1223_PIT_026 e 1223_PIT_025, referem-se aos

Figura 4.4 – Matriz de correlação criada a título de exemplo com 10 atributos de entrada.



Fonte: próprio autor.

transmissores de pressão. De acordo com o diagrama de processo, os dois instrumentos estão instalados no mesmo vaso e também já se esperava que ambos indicassem os mesmos valores. Uma análise parecida pode ser efetuada para o restante dos pares contidos na Tabela 4.3.

Assim, a análise da correlação entre as variáveis possibilitou a exclusão de 17 variáveis que acrescentariam poucas informações aos modelos de estimação de TOG. A Tabela 4.4 apresenta uma sumarização do número de variáveis excluídas em cada etapa até o momento, que foram: engenharia de atributos, filtro por baixo CV e filtro por alta correlação. Portanto, das 116 iniciais, 78 atributos foram utilizados nos procedimentos seguintes.

4.4 Extração de características e seleção de atributos com *random forest*

Com os atributos restantes após as exclusões nas etapas anteriores, chega-se a extração de características e criação do *ranking* de importância. Para a conversão dos dados sequenciais para dados tabulares, que é o tipo de entrada utilizado no modelo *random forest*, extraiu-se as oito características dos 78 atributos com uma janela temporal de 48 horas. Cabe aqui ressaltar que o TOG gravimétrico, variável que deseja-se estimar, é representado por um valor diário e este é obtido por análise de uma mistura de quatro subamostras, coletadas ao longo do dia.

A Figura 4.5 mostra uma das árvores que formam o modelo *random forest* treinado. As

Tabela 4.3 – Pares de variáveis com coeficiente de correlação maior que o limiar 0,9.

Variável 1	Variável 2	Correlação
1223_LT_802M	1223_LT_802MP	1,00
1223_PIT_026	1223_PIT_025	1,00
5331_PIT_216A	5331_PIT_214A	1,00
1223_PIT_003	1223_PIT_014	0,99
1223_LT_802	1223_LT_802M	0,99
1223_PIT_014	1223_PIT_043	0,99
1223_PIT_009	1223_PIT_012	0,99
PROD BRUTA MÉDIA	PROD LÍQUIDA OLEO	0,99
1223_FIT_025	1223_$FQIT$_025	0,94
1223_LIT_011	1223_LIT_003	0,92
5331_$PDIT$_202B	5331_FIT_004	0,92
5271_ZSH_707	5271_PIT_512	0,91
1223_IT_004A	1223_IT_004B	0,91
1223_FIT_009	1223_FIT_010	0,91
PROD LÍQUIDA OLEO	EMED OLEO	0,91
1223_FIT_005	1223_LIT_011	0,90
1223_TIT_022A	1223_TIT_011	0,90

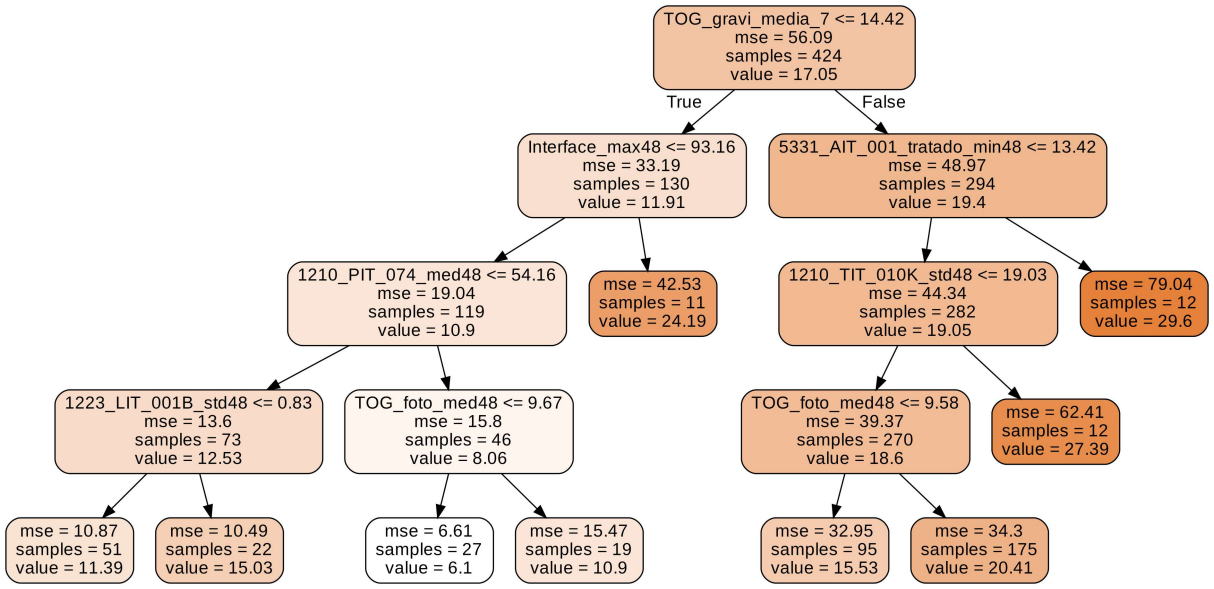
Tabela 4.4 – Número de atributos antes e depois dos métodos de seleção.

Etapa	**Antes**	**Depois**	**Variáveis excluídas**
Engenharia de atributos	116	96	20
Filtro por baixo CV	96	95	1
Filtro por alta correlação	95	78	17

células representam dois tipos de nós: de decisão e folhas. Os nós de decisão, que possuem ao menos um sucessor, contêm testes condicionais das características extraídas e o resultado da comparação pode ser verdadeiro ou falso, o que indicará o caminho a ser seguido. Os nós folhas, que não têm subdivisões, geram a saída do modelo, que é a média da variável alvo das amostras daquela folha.

Analisando o primeiro nó de decisão, na parte superior, pode-se observar que foi selecionada a característica $TOG\_gravi\_media\_7$, que é a média das últimas sete amostras disponíveis de TOG gravimétrico, e o teste condicional é se seu valor é $\leq$ 14,42. Outras informações presentes na célula são: o número de amostras contidas (424); o valor médio das amostras (17,05) e o MSE, ou *Mean Squared Error*, quando os valores verdadeiros do alvo são comparados com o valor médio das amostras do nó. Das 424 amostras presentes inicialmente, o teste condicional resultou em 130 positivos, que seguem pela ramificação esquerda, e 294 falsos, que tomam o caminho da direita. Prosseguindo com a análise e tomando a ramificação esquerda, pode-se perceber que a característica obtida pelo modelo foi $Interface\_max48$ (valor máximo do nível da interface nas últimas 48 horas) e o teste condicional foi $\leq$ 93,16. Da mesma forma, há duas possibilidades para o fluxo continuar. Se o teste resultar um valor verdadeiro, a criação de novos nós de decisão prossegue. Tal processo se repete até que seja atingido um critério de parada, o que ocorreu no

Figura 4.5 – Exemplo de uma árvore que compõe o modelo *random forest* após treinamento.



Fonte: próprio autor.

caso de valor falso da célula $Interface\_max48$, chegando a um nó folha, isto é, uma terminação da árvore. Esse nó englobou 11 amostras e o valor médio do TOG gravimétrico delas foi de 24,19. Dessa forma, na estimação de novas instâncias, se uma amostra possuir as características que a levem a esse nó folha, o TOG gravimétrico estimado pelo modelo *random forest* será 24,19.

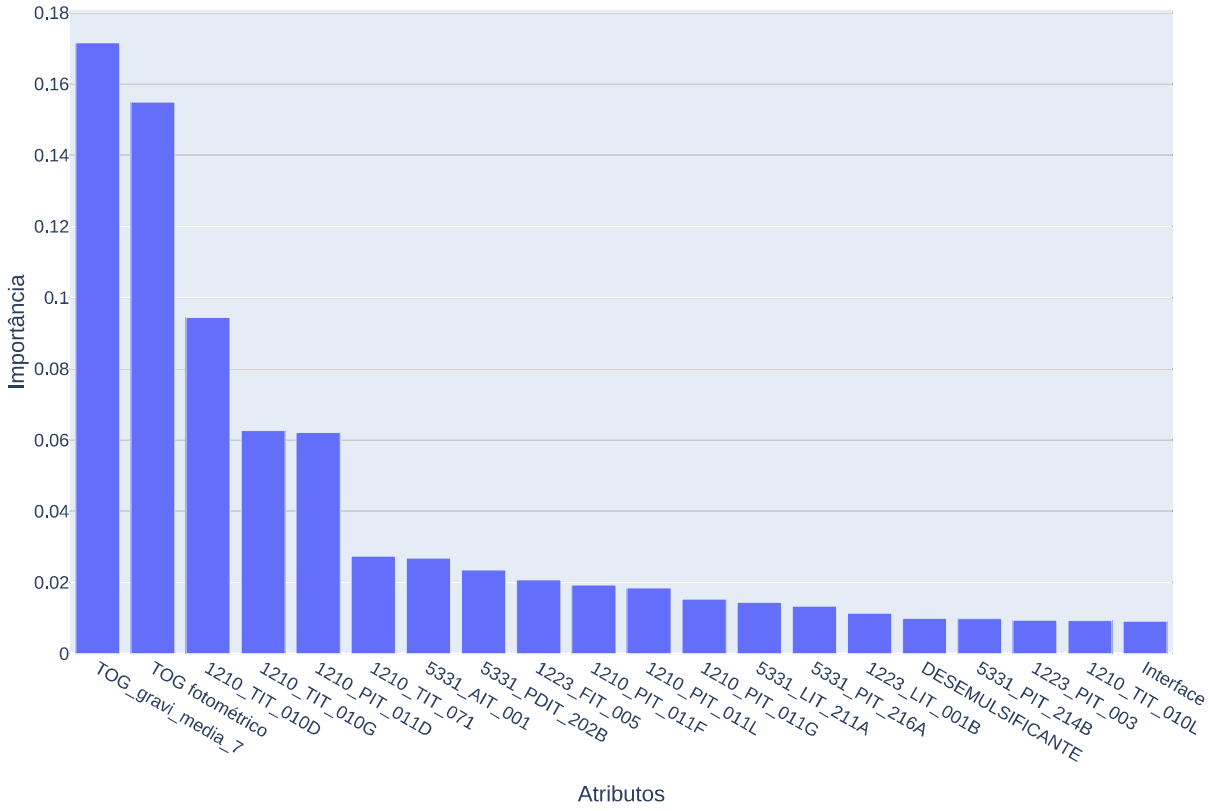
Uma propriedade da Figura 4.5 é que os tons das células variam de acordo com o valor médio da variável alvo das amostras que as compõem. Quanto mais escura a célula, maior o TOG gravimétrico daquela célula. Isso pode facilitar a verificação se o modelo está condizente com o esperado e gerar novos *insights* sobre o processo e a influência de cada atributo no TOG.

Após o treinamento do *random forest*, é possível obter a quantificação da importância de cada entrada, ou seja, das características extraídas. A obtenção da importância usa a ideia de verificar o quanto cada característica conseguiu reduzir o erro com sua utilização. Por exemplo, o nó de decisão composto pela característica $1210\_PIT\_074\_MED48$ possui um valor de MSE de $19,04$. Seus nós descendentes possuem MSE de $13,60$ (73 amostras) e $15,80$ (46 amostras). A média ponderada dos erros dos nós descendentes é de $14,45$, uma redução de $4,59$. O somatório da redução de erro de cada entrada em todas as árvores dividido pelo somatório da redução de todas as entradas fornece o valor da importância da característica extraída. Essa propriedade refere-se à parcela de contribuição da entrada para redução do erro total e, como passa por normalização, a soma de todas as importâncias é sempre igual a 1. Assim, pode-se chegar à importância do atributo ao somar as importâncias de todas as suas características.

Com isso, gerou-se um *ranking* de importância dos atributos. A Figura 4.6 mostra os 20

primeiros atributos desse *ranking*.

Figura 4.6 – Os 20 atributos com maior importância de acordo com o *random forest*.



Fonte: próprio autor.

Destaca-se que o $TOG\_gravi\_media\_7$, atributo criado usando a média das sete amostras mais recentes da variável de saída, alcançou a primeira posição. Em segundo está o $TOG fotométrico$, umas da medições executadas no laboratório de bordo. Além desses, o analisador de TOG *online* (5331_AIT_001) também obteve boa colocação.

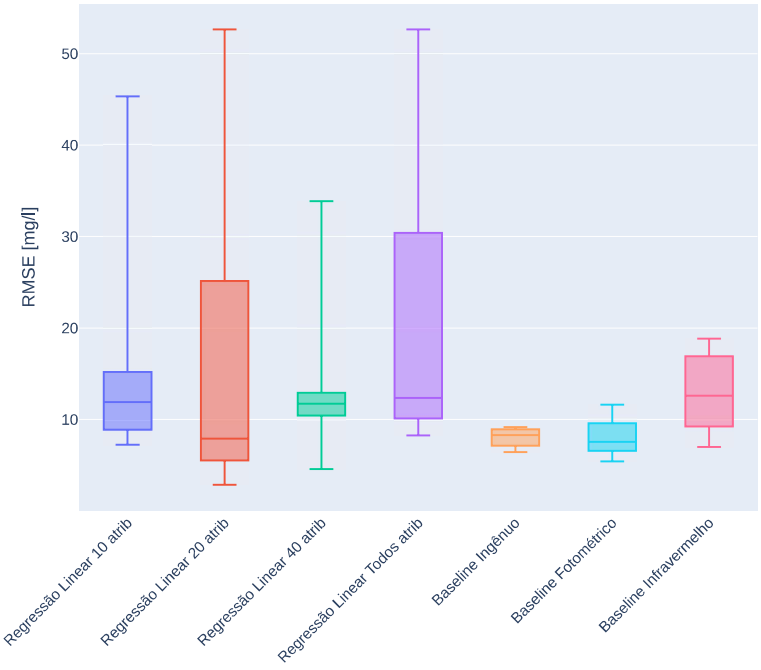
4.5 Resultados dos modelos

Nas Figuras 4.7, 4.8, 4.9 e 4.10 são apresentados os *boxplots* dos RMSE dos modelos nos 10 *folds* de teste. São mostrados também os *boxplots* dos *baselines* ingênuo, TOG fotométrico e TOG infravermelho. As legendas são referentes aos modelos que usaram os n atributos mais importantes, de acordo com o *ranking* obtido via *random forest* (Seção 4.4).

Os modelos de regressão linear apresentaram desempenhos ruins, com valores de *fit* negativos, ou seja, não conseguiram superar nenhum dos três *baselines* (Tabela 4.6). As médias e os desvios padrão também foram piores que os dos *baselines*. A variação da quantidade de atributos por ordem de importância não melhorou significativamente o RMSE dos modelos. Os

resultados não satisfatórios da regressão linear justificam a tentativa de aplicação de modelos mais complexos para estimação do TOG gravimétrico.

Figura 4.7 – *Boxplot* dos erros dos 10 *folds* de teste para o modelo regressão linear ao variar o número de entradas.



Fonte: próprio autor.

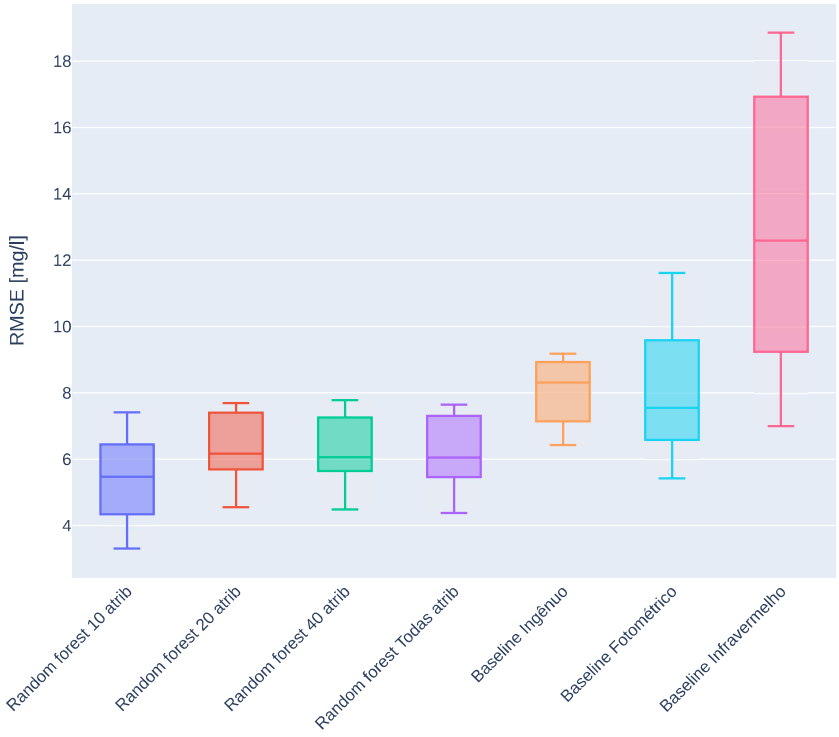
Os resultados do *random forest* (Figura 4.8) foram melhores que os da regressão linear. Percebe-se que o primeiro *boxplot* à esquerda, referente ao modelo treinado com os 10 atributos mais importantes, possui a menor mediana. Além disto, foi o único modelo que seu *boxplot* não sobrepôs os limites dos *boxplots* dos *baselines* ingênuo e fotométrico.

Cabe aqui reforçar que a criação do *ranking* de importância dos atributos foi baseada no próprio *random forest* e, com isso, já se esperava que acrescentar atributos menos importantes não melhoraria os resultados do modelo. Para cada árvore, uma determinada fração das características de entrada do modelo são selecionadas aleatoriamente para treinamento do preditor individual. Esse processo se repete até que o valor máximo de árvores seja treinado. Portanto, a inclusão de atributos menos importantes diminui a chance de atributos mais importantes serem selecionados para criação das árvores e, assim, diminui a força dos preditores individuais que compõe o *random forest*.

Para os modelos LSTM e CNN+LSTM, foi executada a otimização de hiperparâmetros por meio da técnica *random search*, realizando o treinando com 120 conjuntos distintos de hiperparâmetros para cada modelo. O espaço de busca e os melhores hiperparâmetros encontrados são mostrados na Tabela 4.5.

Os resultados dos modelos LSTM são apresentados na Figura 4.9. Observa-se que nenhum

Figura 4.8 – *Boxplot* dos erros dos 10 *folds* de teste para o modelo *random forest* ao variar o número de entradas.



Fonte: próprio autor.

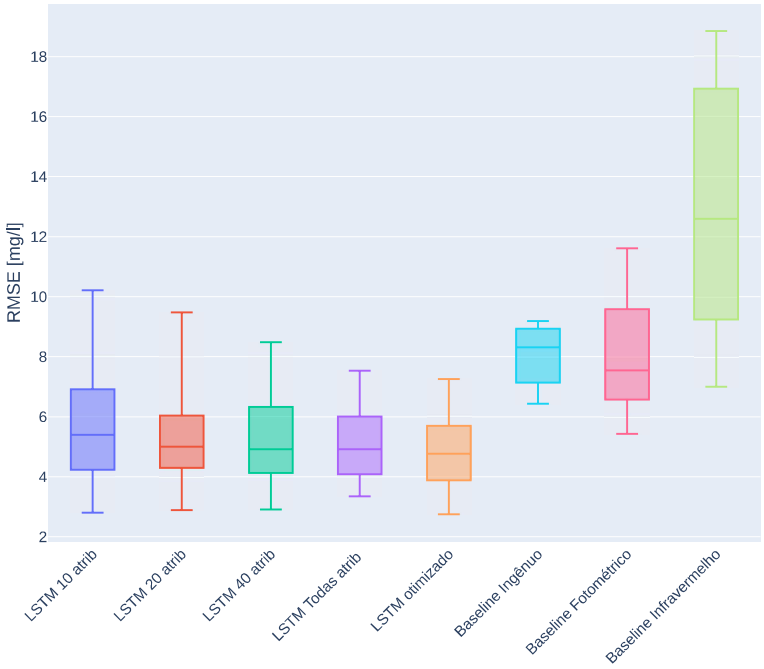
Tabela 4.5 – Espaço de busca e valores encontrados na otimização dos hiperparâmetros dos modelos.

Hiperparâmetro	Espaço de busca	LSTM	CNN+LSTM
Learning rate	$[10^{-4}; 10^{-1}]$	0,005	0,05
Dropout	[0; 0,8]	0,0	0,8
Neurônios LSTM	[4; 2048]	1024	64
Neurônios *dense*	[4; 2048]	512	256
Número de filtros	[2; 256]	-	128
Tamanho do *kernel*	[3; 11]	-	7
Strides	[1; 3]	-	2
Número de atributos	[5; 78]	50	10

dos *boxplots* dos modelos treinados se sobrepôs aos dos modelos *baselines*. Entretanto, em relação aos modelos treinados, há sobreposições entre si. Dessa forma, não é possível concluir qual dos modelos treinados apresenta o melhor resultado. Em valores absolutos, o LSTM otimizado atingiu a menor média e desvio padrão, quando comparado aos demais modelos (Tabela 4.6).

Para a LSTM, a variação do número de atributos de entrada de acordo com a importância calculada via *random forest* não provocou diminuição dos erros médios. A otimização dos hiperparâmetros alcançou os melhores resultados utilizando os 50 atributos mais importantes. Tal comportamento pode ser explicado pelo uso de técnicas de regularização e controle de *overfitting* no algoritmo (*dropout* e *early stopping*), o que torna o modelo menos sensível à inclusão de

Figura 4.9 – *Boxplot* dos erros dos 10 *folds* de teste para o modelo LSTM ao variar o número de entradas.



Fonte: próprio autor.

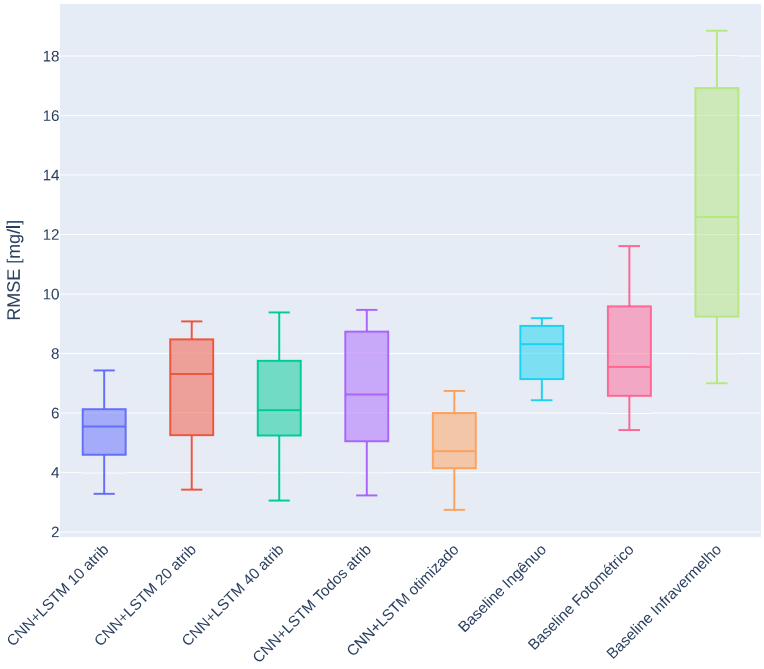
atributos menos importantes e sem grandes perdas de desempenho.

Para os modelos CNN+LSTM (Figura 4.10), verifica-se que os *boxplots* dos que usaram 20, 40 e todos os atributos se sobrepõem aos dos *baselines* ingênuo e fotométrico e, por isso, não é possível afirmar que há diferenças estatisticamente significativas. Somente nos *boxplots* do modelo com 10 atributos e do otimizado não há interseção com os dos *baselines* e, por isso, há evidências suficientes de que a mediana do RMSE de ambos é inferior aos dos *baselines*.

Destaca-se que o modelo CNN+LSTM otimizado alcançou o melhor resultado utilizando os 10 atributos mais importantes. Observa-se também que os modelos que utilizaram mais atributos (20, 40 e todos) tiveram o desempenho degradado, apesar de usar mecanismos de regularização assim como as LSTM. Uma diferença notada é que, devido às camadas convolucionais na entrada, o número de parâmetros treinados em modelos CNN+LSTM é maior do que nas LSTM. Isso, por um lado, pode ser benéfico à medida em que tais camadas podem realizar a extração de características de forma automática e aumentar a flexibilidade do modelo; por outro lado, as chances de *overfitting* são maiores, como verificado quando se adiciona atributos com menor importância.

Por fim, a Figura 4.11 ilustra os *boxplots* dos modelos *random forest*, LSTM e CNN+LSTM com as menores médias do RMSE nos 10 *folds* de teste (Tabela 4.6) e os modelos *baselines* ingênuo, TOG fotométrico e TOG infravermelho. Os resultados da regressão linear não foram incluídos pois seus valores de erro são muito maiores que os demais e sua inclusão prejudicaria a

Figura 4.10 – *Boxplot* dos erros dos 10 *folds* de teste para o modelo CNN+LSTM ao variar o número de entradas.



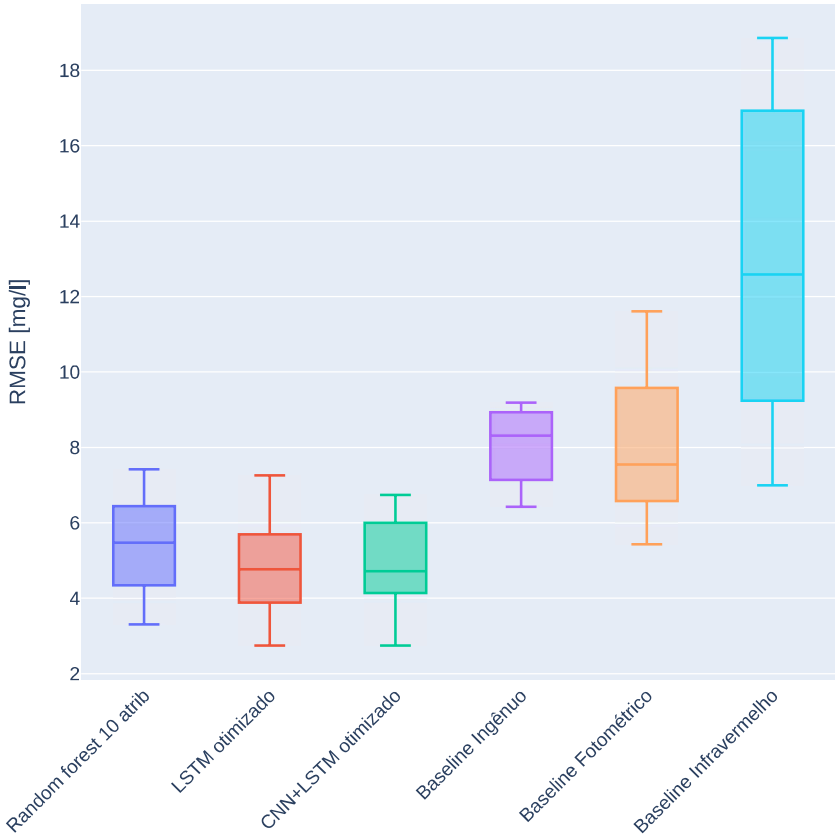
Fonte: próprio autor.

interpretação do gráfico devido a mudança de escala. Verifica-se na figura que os *boxplots* dos modelos treinados não sobrepõem os dos modelos *baselines*, isto é, há indícios de que existe diferenças entre eles.

O teste não-paramétrico de Wilcoxon foi utilizado para a comparação das 10 médias dos erros (RMSE) dos modelos, mostradas na Tabela 4.6 e cujos gráficos *boxplot* são mostrados na Figura 4.11. Em cada teste, a hipótese nula H_0 de que as médias eram iguais foi testada contra a hipótese alternativa H_1 de que as médias eram diferentes. No caso de serem diferentes, assume-se que o melhor desempenho seja do modelo com menor erro entre os dois, mostrado na Tabela 4.6. Na Figura 4.12 são mostrados graficamente os resultados dos testes entre todos os pares de modelos, com a cor indicando o p-valor correspondente a cada teste.

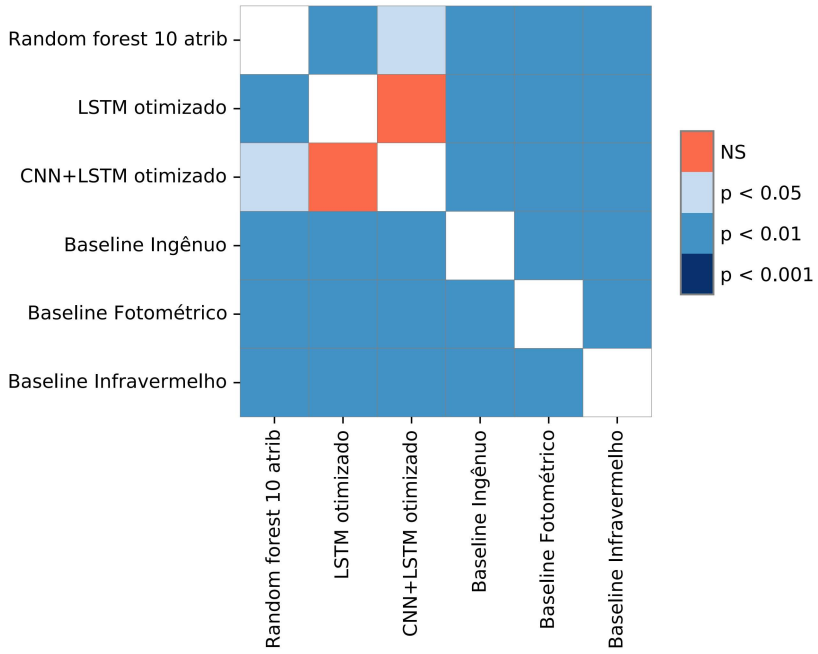
Os modelos treinados (*random forest*, LSTM e CNN+LSTM), quando comparados aos modelos *baselines*, obtiveram resultados melhores, com p-valores sempre inferiores a 0.01, ou seja, houve diferenças estatisticamente significativas. Já quando o *random forest* foi comparado ao LSTM e ao CNN+LSTM, houve diferenças estatisticamente significativas entre os modelos, indicando que LSTM e CNN+LSTM alcançaram melhores desempenhos (p-valores de 0.007 e 0.037, respectivamente). Por fim, a comparação entre LSTM e CNN+LSTM não permitiu afirmar que há diferença entre eles, com p-valor de 0.721.

Figura 4.11 – Comparação do RMSE entre os melhores modelos testados e os *baselines*.



Fonte: próprio autor.

Figura 4.12 – Teste estatístico não-paramétrico de Wilcoxon para comparação do RMSE dos modelos.



Fonte: próprio autor.

Tabela 4.6 – Resultados obtidos pelos modelos nos conjuntos de teste (10 *folds*).

Modelo		**Média do RMSE [mg/l]**	**Desvio-padrão do RMSE [mg/l]**	**Qualidade do fit [%]**
Baseline	Ingênuo	8,08	0,96	0,00%
	TOG fotométrico	7,93	1,96	-0,15%
	TOG infravermelho	12,76	4,06	-60,09%
Regressão Linear	10 variáveis	17,80	14,46	-117,88%
	20 variáveis	15,93	16,05	-105,47%
	40 variáveis	**14,28**	**8,61**	**-83,78%**
	Todas variáveis	19,56	14,92	-154,71%
Random forest	**10 variáveis**	**5,50**	**1,33**	**31,63%**
	20 variáveis	6,31	1,00	21,78%
	40 variáveis	6,28	1,09	22,26%
	Todas variáveis	6,19	1,06	23,32%
LSTM	10 variáveis	5,83	2,33	28,44%
	20 variáveis	5,34	1,91	33,34%
	40 variáveis	5,27	1,65	34,52%
	Todas variáveis	5,13	1,31	35,73%
	Otimizado	**4,99**	**1,28**	**37,66%**
CNN + LSTM	10 variáveis	5,49	1,29	32,08%
	20 variáveis	6,78	2,00	15,93%
	40 variáveis	6,32	1,99	20,91%
	Todas variáveis	6,82	2,10	14,37%
	Otimizado	**4,84**	**1,35**	**39,23%**

5 Conclusão

Neste trabalho, investigou-se o problema da estimação de TOG gravimétrico em água descartada no mar por plataforma *offshore* de produção de petróleo, ressaltando a característica método-dependente do TOG e o tempo necessário para se obter os resultados pelo método de referência aceito pelo órgão fiscalizador devido ao envio das amostras para laboratórios *onshore*. O objetivo principal foi avaliar a utilização de modelos baseados em dados para estimar o TOG gravimétrico.

A metodologia proposta foi criar modelos (regressão linear, *random forest*, LSTM e CNN+LSTM) e testá-los com diferentes números de atributos, ordenados previamente a partir da importância obtida por meio do modelo *random forest*. Além disso, também foram avaliados modelos após a otimização de hiperparâmetros via *random search*.

Os erros dos modelos criados foram comparados com os erros referentes às estimativas obtidas pelos métodos fotométrico e infravermelho, além de outro *baseline* calculado como a média do TOG gravimétrico do banco de treinamento.

Os resultados alcançados mostraram que os modelos LSTM otimizado e CNN+LSTM otimizado superaram os dos modelos *baselines* (ingênuo, fotométrico e infravermelho), além do melhor modelo *random forest*. Cabe destacar que um dos hiperparâmetros otimizados foi o número de atributos, ou seja, com um número menor de atributos foram alcançados os melhores resultados, caracterizando a seleção de atributos. Os resultados obtidos neste trabalho indicam a viabilidade e a utilidade de modelos baseados em dados para monitorar o TOG.

Perspectivas futuras englobam a aplicação de técnicas mais avançadas de otimização de hiperparâmetros, avaliação de outros tamanhos de janela de entrada e do período de amostragem, uso de outros modelos de regressão, além da criação de *ensembles*.

Referências

Advanced Sensors. *Advanced Sensors EX-100/1000*. 2021. Disponível em: <https://www.advancedsensors.co.uk/products/ex-100-1000>.

AL-GHOUTI, M. A.; AL-KAABI, M. A.; ASHFAQ, M. Y.; DA'NA, D. A. Produced water characteristics, treatment and reuse: A review. *Journal of Water Process Engineering*, Elsevier, v. 28, p. 222–239, 2019.

ALZUBAIDI, L.; ZHANG, J.; HUMAIDI, A. J.; AL-DUJAILI, A.; DUAN, Y.; AL-SHAMMA, O.; SANTAMARíA, J.; FADHEL, M. A.; AL-AMIDIE, M.; FARHAN, L. Review of deep learning: concepts, cnn architectures, challenges, applications, future directions. *Journal of Big Data*, v. 8, n. 53, p. 1–74, 2021.

AMINI, S.; MOWLA, D.; GOLKAR, M.; ESMAEILZADEH, F. Mathematical modelling of a hydrocyclone for the down-hole oil–water separation (dows). *Chemical Engineering Research and Design*, v. 90, n. 12, p. 2186 – 2195, 2012. ISSN 0263-8762.

AMORIM, L. C. A. Os biomarcadores e sua aplicação na avaliação da exposição aos agentes químicos ambientais. *Revista Brasileira de Epidemiologia*, SciELO Public Health, v. 6, p. 158–170, 2003.

ANP. *Produção por plataforma*. Rio de Janeiro, RJ, 2020. Disponível em: <http://www.anp.gov.br/arquivos/dados-ep/historico-producao-plataforma.xlsx>.

ATTANASI, E. D.; FREEMAN, P. A.; COBURN, T. C. Well predictive performance of play-wide and subarea random forest models for bakken productivity. *Journal of Petroleum Science and Engineering*, Elsevier, v. 191, p. 107150, 2020.

BAIRD, R. B.; EATON, A. D.; RICE, E. W.; BRIDGEWATER, L. et al. *Standard methods for the examination of water and wastewater*. [S.l.]: American Public Health Association Washington, DC, 2017.

BAYATI, F.; SHAYEGAN, J.; NOORJAHAN, A. Treatment of oilfield produced water by dissolved air precipitation/solvent sublation. *Journal of Petroleum Science and Engineering*, Elsevier, v. 80, n. 1, p. 26–31, 2011.

BELKIN, M.; HSU, D.; MA, S.; MANDAL, S. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, National Acad Sciences, v. 116, n. 32, p. 15849–15854, 2019.

BENGIO, Y.; SIMARD, P.; FRASCONI, P. Learning long-term dependencies with gradient descent is difficult. *IEEE transactions on neural networks*, IEEE, v. 5, n. 2, p. 157–166, 1994.

BERGSTRA, J.; BENGIO, Y. Random search for hyper-parameter optimization. *Journal of machine learning research*, v. 13, n. 2, 2012.

BOMMERT, A.; SUN, X.; BISCHL, B.; RAHNENFÜHRER, J.; LANG, M. Benchmark for filter methods for feature selection in high-dimensional classification data. *Computational Statistics & Data Analysis*, Elsevier, v. 143, p. 106839, 2020.

BONTEMPI, G.; TAIEB, S. B.; BORGNE, Y.-A. L. Machine learning strategies for time series forecasting. In: SPRINGER. *European business intelligence summer school*. [S.l.], 2012. p. 62–77.

BOROVYKH, A.; BOHTE, S.; OOSTERLEE, C. W. Conditional time series forecasting with convolutional neural networks. *arXiv preprint arXiv:1703.04691*, 2017.

BRAGA, A. d. P. *Redes neurais artificiais: teoria e aplicações*. [S.l.]: Livros Técnicos e Científicos, 2000.

BRANCO, A.; GOMIDE, J. Previsão de taxa de perfuração em poços de petróleo offshore utilizando aprendizado de máquina. In: SBC. *Anais do XVIII Encontro Nacional de Inteligência Artificial e Computacional*. [S.l.], 2021. p. 504–515.

BRASIL. Resolução conama nº 393/2007. *Diário Oficial da União*, Brasília, DF, n. 153, p. 72–73, 2007. Disponível em: <http://www2.mma.gov.br/port/conama/legiabre.cfm?codlegi=541>.

BRASIL. Extrato de compromisso. *Diário Oficial da União*, Brasília, DF, n. 47, p. 100, 2018. Disponível em: <http://pesquisa.in.gov.br/imprensa/jsp/visualiza/index.jsp?data=09/03/2018&jornal=530&pagina=100>.

BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, n. 1, p. 5–32, 2001.

CHANDRASHEKAR, G.; SAHIN, F. A survey on feature selection methods. *Computers & Electrical Engineering*, Elsevier, v. 40, n. 1, p. 16–28, 2014.

CHEN, C.-W.; TSAI, Y.-H.; CHANG, F.-R.; LIN, W.-C. Ensemble feature selection in medical datasets: Combining filter, wrapper, and embedded feature selection results. *Expert Systems*, Wiley Online Library, v. 37, n. 5, p. e12553, 2020.

CIRNE, I.; BOAVENTURA, J.; GUEDES, Y.; LUCAS, E. Methods for determination of oil and grease contents in wastewater from the petroleum industry. Publishing House of Lviv Polytechnic National University, 2016.

DEISTLER, M. System identification and time series analysis: Past, present, and future. In: *Stochastic Theory and Control*. [S.l.]: Springer, 2002. p. 97–109.

ERALYTICS. *Eracheck Eco*. 2021. Disponível em: <https://eralytics.com/instruments/oil-in-water-testers/eracheck-eco>.

ERTUĞRUL, Ö. F.; TAĞLUK, M. E. A fast feature selection approach based on extreme learning machine and coefficient of variation. *Turkish Journal of Electrical Engineering & Computer Sciences*, The Scientific and Technological Research Council of Turkey, v. 25, n. 4, p. 3409–3420, 2017.

FAKHRU'L-RAZI, A.; PENDASHTEH, A.; ABDULLAH, L. C.; BIAK, D. R. A.; MADAENI, S. S.; ABIDIN, Z. Z. Review of technologies for oil and gas produced water treatment. *Journal of hazardous materials*, Elsevier, v. 170, n. 2-3, p. 530–551, 2009.

FEURER, M.; HUTTER, F. Hyperparameter optimization. In: *Automated machine learning*. [S.l.]: Springer, Cham, 2019. p. 3–33.

FILHO, C. F. A.; VARGAS, R. E. V.; BUCKER, E. B.; DELAIBA, V. H. B. Monitoramento de teor de óleos e graxas em água descartada no mar usando ciência de dados. *Rio Oil & Gas Conference*, Instituto Brasileiro de Petróleo e Gás, 2020.

GAGNON, M. Evidence of exposure of fish to produced water at three offshore facilities, north west shelf, australia. In: ____. *Produced water: environmental risks and advances in mitigation technologies*. [S.l.: s.n.], 2011. p. 295–309.

GÉRON, A. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems*. [S.l.]: O'Reilly Media, 2019.

GOODFELLOW, I.; BENGIO, Y.; COURVILLE, A. *Deep Learning*. [S.l.]: MIT Press, 2016. <http://www.deeplearningbook.org>.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. *Journal of machine learning research*, v. 3, n. Mar, p. 1157–1182, 2003.

HACH. *DR6000 Laboratory Spectrophotometer*. 2021. Disponível em: <https://www.hach.com/spectrophotometers/dr6000-laboratory-spectrophotometer/family?productCategoryId=35547203833>.

HASAN, M. A. M.; NASSER, M.; AHMAD, S.; MOLLA, K. I. Feature selection for intrusion detection using random forest. *Journal of information security*, Scientific Research Publishing, v. 7, n. 3, p. 129–140, 2016.

HAYKIN, S. *Neural networks and learning machines, 3/E*. [S.l.]: Pearson Education India, 2010.

HIDAYAT, F.; ASTSAURI, T. M. S. Applied random forest for parameter sensitivity of low salinity water injection (lswi) implementation on carbonate reservoir. *Alexandria Engineering Journal*, Elsevier, v. 61, n. 3, p. 2408–2417, 2022.

HINTON, G. E.; SRIVASTAVA, N.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. R. Improving neural networks by preventing co-adaptation of feature detectors. *arXiv preprint arXiv:1207.0580*, 2012.

HOCHREITER, S. Untersuchungen zu dynamischen neuronalen netzen. *Diploma, Technische Universität München*, v. 91, n. 1, 1991.

HOCHREITER, S.; SCHMIDHUBER, J. Long short-term memory. *Neural computation*, MIT Press, v. 9, n. 8, p. 1735–1780, 1997.

HUSVEG, T.; RAMBEAU, O.; DRENGSTIG, T.; BILSTAD, T. Performance of a deoiling hydrocyclone during variable flow rates. *Minerals Engineering*, Elsevier, v. 20, n. 4, p. 368–379, 2007.

HYNDMAN, R. J.; ATHANASOPOULOS, G. *Forecasting: principles and practice*. [S.l.]: OTexts, 2018.

IOFFE, S.; SZEGEDY, C. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In: PMLR. *International conference on machine learning*. [S.l.], 2015. p. 448–456.

JIMÉNEZ, S.; MICÓ, M.; ARNALDOS, M.; MEDINA, F.; CONTRERAS, S. State of the art of produced water treatment. *Chemosphere*, Elsevier, v. 192, p. 186–208, 2018.

JúNIOR, J. M. O.; PEREIRA, M. d. A. Forecasting Total Oil and Grease in produced water using Machine Learning methods in an oil extraction plant. *Marine Systems & Ocean Technology*, Springer, v. 15, p. 124–134, 2020.

KRIZHEVSKY, A.; SUTSKEVER, I.; HINTON, G. E. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, v. 25, p. 1097–1105, 2012.

LECUN, Y.; BENGIO, Y.; HINTON, G. Deep learning. *nature*, Nature Publishing Group, v. 521, n. 7553, p. 436–444, 2015.

LECUN, Y.; BENGIO, Y. et al. Convolutional networks for images, speech, and time series. *The handbook of brain theory and neural networks*, v. 3361, n. 10, p. 1995, 1995.

LIU, C.-L.; HSAIO, W.-H.; TU, Y.-C. Time series classification with multivariate convolutional neural network. *IEEE Transactions on Industrial Electronics*, IEEE, v. 66, n. 6, p. 4788–4797, 2018.

LIU, H.; YU, L. Toward integrating feature selection algorithms for classification and clustering. *IEEE Transactions on knowledge and data engineering*, IEEE, v. 17, n. 4, p. 491–502, 2005.

LORENA, A. C.; GAMA, J.; FACELI, K. *Inteligência artificial: uma abordagem de aprendizado de máquina*. [S.l.]: Grupo Gen-LTC, 2000.

MEIER, S.; MORTON, H.; NYHAMMER, G.; GROSVIK, B.; MAKHOTIN, V.; GEFFEN, A.; BOITSOV, S.; KVESTAD, K.; BOHNE-KJERSEM, A.; GOKSøYR, A.; FOLKVORD, A.; KLUNGSøYR, J.; SVARDAL, A. Development of atlantic cod (gadus morhua) exposed to produced water during early life stages effects on embryos, larvae, and juvenile fish. *Marine environmental research*, v. 70, p. 383–94, 12 2010.

MITCHELL, T. Machine learning. McGraw hill Burr Ridge, 1997.

MOLNAR, C. *Interpretable Machine Learning*: A guide for making black box models explainable. [S.l.: s.n.], 2019.

MONTGOMERY, D. C.; RUNGER, G. C. *Applied statistics and probability for engineers*. [S.l.]: John Wiley & Sons, 2010.

MUKAKA, M. M. A guide to appropriate use of correlation coefficient in medical research. *Malawi medical journal*, v. 24, n. 3, p. 69–71, 2012.

NAIR, V.; HINTON, G. E. Rectified linear units improve restricted boltzmann machines. In: *Icml*. [S.l.: s.n.], 2010.

OLIVEIRA-SANTOS, T.; RAUBER, T. W.; VAREJAO, F. M.; MARTINUZZO, L.; OLIVEIRA, W.; RIBEIRO, M. P.; RODRIGUES, A. Submersible motor pump fault diagnosis system: A comparative study of classification methods. In: IEEE. *2016 IEEE 28th International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.], 2016. p. 415–422.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. *Cross-validation: evaluating estimator performance*. 2021. Disponível em: <https://scikit-learn.org/stable/modules/cross_validation.html>. Acesso em: 30 dezembro 2021.

PRECHELT, L. Early stopping-but when? In: *Neural Networks: Tricks of the trade*. [S.l.]: Springer, 1998. p. 55–69.

RICE, E.; BAIRD, R.; EATON, A. *Standard methods for the examination of water and wastewater*. 23. ed. [S.l.]: American Public Health Association and American Water Works Association and others, 2017.

ROKACH, L.; MAIMON, O. *Data Mining With Decision Trees: Theory and Applications*. 2nd. ed. USA: World Scientific Publishing Co., Inc., 2014. ISBN 9789814590075.

ROVERSO, D. Empirical ensemble-based virtual sensing-a novel approach to oil-in-water monitoring. In: *Oil-in-Water Monitoring Workshop, Aberdeen, UK*. [S.l.: s.n.], 2009.

SAMUEL, A. L. Some studies in machine learning using the game of checkers. *IBM Journal of research and development*, IBM, v. 3, n. 3, p. 210–229, 1959.

SILVA, R. S. *Aprendizado de máquina aplicado ao planejamento de lavra de curto prazo para o aumento do desempenho operacional de equipamentos de mina.* Dissertação (Mestrado) — Universidade de São Paulo, 2021.

SOUZA, A. J. de; BEZERRA, C. G.; FEIJó, R. H.; ANDRADE, W. L. S. de; LEITãO, G. B. P.; GUEDES, L. A.; MEDEIROS, A. A. D.; MAITELLI, A. L. Gerência de informação de processos industriais - um estudo de caso na produção de petróleo e gás. In: VII SBAI / II IEEE LARS. São Luís, Brasil, 2005.

SRIVASTAVA, N.; HINTON, G.; KRIZHEVSKY, A.; SUTSKEVER, I.; SALAKHUTDINOV, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, JMLR. org, v. 15, n. 1, p. 1929–1958, 2014.

STEWART, M.; ARNOLD, K. *Produced water treatment field manual*. [S.l.]: Gulf Professional Publishing, 2011.

TIBSHIRANI, R. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B (Methodological)*, Wiley Online Library, v. 58, n. 1, p. 267–288, 1996.

VEIL, J. A. Produced water management options and technologies. In: *Produced water*. [S.l.]: Springer, 2011. p. 537–571.

VIJAYAPRABAKARAN, K.; SATHIYAMURTHY, K. Neuroevolution based hierarchical activation function for long short-term model network. *Journal of Ambient Intelligence and Humanized Computing*, Springer, v. 12, n. 12, p. 10757–10768, 2021.

YANG, L.; SHAMI, A. On hyperparameter optimization of machine learning algorithms: Theory and practice. *Neurocomputing*, Elsevier, v. 415, p. 295–316, 2020.

YANG, M. Measurement of oil in produced water. In: *Produced water*. [S.l.]: Springer, 2011. p. 57–88.

YU, Y.; SI, X.; HU, C.; ZHANG, J. A review of recurrent neural networks: Lstm cells and network architectures. *Neural computation*, MIT Press One Rogers Street, Cambridge, MA 02142-1209, USA journals-info . . . , v. 31, n. 7, p. 1235–1270, 2019.

ZHONG, Z.; CARR, T. R.; WU, X.; WANG, G. Application of a convolutional neural network in permeability prediction: A case study in the jacksonburg-stringtown oil field, west virginia, usa. *Geophysics*, Society of Exploration Geophysicists, v. 84, n. 6, p. B363–B373, 2019.