JOSÉ VINÍCIUS SANTOS DE ARAÚJO

ALARMES INTELIGENTES EM PROCESSOS DE DESMINERALIZAÇÃO DA ÁGUA PARA A PRODUÇÃO DE VIDROS BASEADO EM DESCOBERTA DO CONHECIMENTO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica - PPGEE, da Universidade Federal da Paraíba - UFPB, como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Juan Moisés Maurício Villanueva

JOÃO PESSOA

2024

JOSÉ VINÍCIUS SANTOS DE ARAÚJO

ALARMES INTELIGENTES EM PROCESSOS DE DESMINERALIZAÇÃO DA ÁGUA PARA A PRODUÇÃO DE VIDROS BASEADO EM DESCOBERTA DO CONHECIMENTO

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica - PPGEE, da Universidade Federal da Paraíba - UFPB, como requisito parcial para a obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Juan Moisés Maurício Villanueva

JOÃO PESSOA

2024

Catalogação na publicação
Seção de Catalogação e Classificação

A663a Araújo, José Vinícius Santos de.
Alarmes inteligentes em processos de desmineralização da água para a produção de vidros baseados em descoberta do conhecimento / José Vinícius Santos de Araújo. - João Pessoa, 2024.
84 f. : il.

Orientação: Juan Moisés Maurício Villanueva.
Coorientação: Euler Cássio Tavares de Macedo.
Dissertação (Mestrado) - UFPB/CEAR.

1. Engenharia elétrica - Data mining. 2. Alarmes preditivos. 3. Extração de regras. 4. Detecção de anomalias. I. Villanueva, Juan Moisés Maurício. II. Macedo, Euler Cássio Tavares de. III. Título.

UFPB/BC CDU 621.3(043)

Elaborado por WALQUELINE DA SILVA ARAUJO - CRB-15/514

UNIVERSIDADE FEDERAL DA PARAÍBA – UFPB

CENTRO DE ENERGIAS ALTERNATIVAS E RENOVÁVEIS – CEAR

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA – PPGEE

A Comissão Examinadora, abaixo assinada, aprova a Dissertação

ALARMES INTELIGENTES EM PROCESSOS DE DESMINERALIZAÇÃO DA ÁGUA PARA A PRODUÇÃO DE VIDROS BASEADO EM DESCOBERTA DO CONHECIMENTO

Elaborada por

JOSÉ VINÍCIUS SANTOS DE ARÁUJO

como requisito parcial para obtenção do grau de

Mestre em Engenharia Elétrica.

COMISSÃO EXAMINADORA

PROF. DR. JUAN MOISES MAURICIO VILLANUEVA
Orientador – UFPB

Documento assinado digitalmente
gov.br EULER CASSIO TAVARES DE MACEDO
Data: 02/12/2024 21:43:50-0300
Verifique em https://validar.iti.gov.br

PROF. DR. EULER CÁSSIO TAVARES DE MACEDO
Coorientador– UFPB

Documento assinado digitalmente
gov.br CLEONILSON PROTASIO DE SOUZA
Data: 30/11/2024 15:53:48-0300
Verifique em https://validar.iti.gov.br

PROF. DR. CLEONILSON PROTÁSIO DE SOUZA
Examinador Interno – UFPB

Documento assinado digitalmente
gov.br IVANOVITCH MEDEIROS DANTAS DA SILVA
Data: 03/12/2024 05:43:44-0300
Verifique em https://validar.iti.gov.br

PROF. DR. IVANOVITCH MEDEIROS DANTAS DA SILVA
Examinador Externo – UFRN

João Pessoa/PB, 29 de novembro de 2024

Aos meus pais e irmão que me deram apoio emocional fundamental,

À turma do EMBRAPII que sempre estiveram na torcida

A meus familiares e amigos queridos

Dedico

AGRADECIMENTOS

Agradeço meus pais e irmão, por sempre serem o suporte fundamental todos os dias, a meus amigos pelos momentos de felicidade e apoio, dentre eles, especialmente meus amigos de longa data da UFPB e o pessoal da EMBRAPII que me deram toda a confiança e estrutura para que esse trabalho fosse possível.

“Se avexe não! Toda caminhada começa no primeiro passo, a natureza não tem pressa, segue seu compasso, inexoravelmente chega lá.”

Flávio José.

SUMÁRIO

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

CSV	*Comma Separated Values*
IHM	Interface Homem Máquina
KDD	*Knowledge Discovery in Databases*
LOF	*Local Outliers Factor*
NaN	Não Numéricos
OR	Osmose Reversa
RL	Regressão Linear
SCADA	*Supervisory, Control and Data Acqusition System*
TLM	Trocador de Leito Misto
LSTM	*Long Short Term Memory*
ARIMA	*AutoRegressive Integrated Moving Average*

RESUMO

ALARMES INTELIGENTES EM PROCESSOS DE DESMINERALIZAÇÃO DA ÁGUA PARA A PRODUÇÃO DE VIDROS BASEADO EM DESCOBERTA DO CONHECIMENTO

A antecipação de deterioração no comportamento desejado de uma variável de processo é de fundamental importância para que sistemas operem de maneira eficiente e com segurança. Nesse contexto, alarmes são essenciais para captarem a atenção de operadores da necessidade de ações corretivas imediatas no processo. Entretanto, os alarmes tradicionais baseados em controle estatístico de processo seguem metodologias que há muito tempo não são atualizados de acordo com as novas demandas das indústrias como, por exemplo, a detecção preditiva de anomalias que utilizam apenas métodos do controle estatístico de processos, ficando restritos à observação dos limites inferior e superior da variável de processo. Assim, como uma forma de melhorar a relação entre a sinalização de alarmes e a variação da dinâmica de processos industriais, neste trabalho é proposto uma metodologia para criação de alarmes inteligentes baseados na combinação de algoritmos de detecção de tendências (regressão linear) com algoritmos de extração de regras (baseados em data mining). A partir da aplicação deste método, espera-se criar um sistema de alarmes que indique tendências de deterioração de uma variável de processo antes que esta chegue a atingir limites críticos, com o objetivo de proporcionar tempo hábil para tomada de ações corretivas e, posteriormente, por meio de um modelo de classificação de árvore de decisão, extrair conhecimento dos dados para que se tenha sugestões acerca de qual processo deve ser corrigido para que a tendência de deterioração seja cessada. A combinação das duas técnicas resultou em um alarme capaz de detectar desvios progressivos e informações relevantes para a operação.

Palavras-chave: *data mining;* alarmes preditivos; extração de regras; detecção de anomalias.

ABSTRACT

INTELLIGENT ALARMS IN WATER DEMINERALIZATION PROCESSES FOR GLASS PRODUCTION BASED ON KNOWLEDGE DISCOVERY

The anticipation of deterioration in the desired behavior of a process variable is crucial for systems to operate efficiently and safely. In this context, alarms are essential for capturing operators' attention to the need for immediate corrective actions in the process. However, traditional alarms based on statistical process control follow methodologies that have long remained unchanged, failing to keep pace with modern industrial demands, such as predictive anomaly detection that relies solely on statistical process control methods, limited to observing the upper and lower bounds of the process variable. Therefore, as a way to improve the relationship between alarm signaling and variations in industrial process dynamics, this work proposes a methodology for creating intelligent alarms based on the combination of trend detection algorithms (linear regression) with rule extraction algorithms (based on data mining). Through the application of this method, the goal is to create an alarm system that indicates deterioration trends in a process variable before it reaches critical limits, providing sufficient time for corrective actions. Additionally, using a decision tree classification model, it is intended to extract knowledge from the data to offer suggestions regarding which process should be adjusted to halt the deterioration trend. The combination of these two techniques resulted in an alarm capable of detecting progressive deviations and providing valuable information for operations.

Keywords: data mining; predictive alarms; rule extraction; anomaly detection.

1 INTRODUÇÃO

1. INTRODUÇÃO

A fabricação de espelhos é um processo que exige controle altamente preciso dos insumos que estão sendo utilizados devido ao elevado grau de qualidade que as normas regulamentadoras, como a NBR14696 (ABNT, 2015), exigem como pré-requisito mínimo de qualidade.

Destes variados processos que compreendem a fabricação dos espelhos, um elemento fundamental é a qualidade da água. A água, nas condições ideais, será responsável por auxiliar na pré-lavagem do vidro, polimento, sensibilização, ativação, prateação e adesão da prata ao vidro (VIVIX, 2024). Em cada um desses processos, é necessário que a água tenha um padrão mínimo de qualidade em termos de condutividade, alcalinidade e dureza, conforme foi apontado em Düffer's F. (1996). A influência destas propriedades químicas afeta a aderência à superfície do vidro em processos de revestimento.

1.1 PERTINÊNCIA E MOTIVAÇÃO DO TRABALHO

A indústria de espelhos sendo pertencente a um contexto fabril exigente, o monitoramento e controle constante das variáveis do processo e a previsibilidade de eventos adversos na planta são de fundamental importância para manter a qualidade em todas as fases da produção do espelho.

É nesse contexto que se insere um dos aspectos da indústria 4.0: o *Data Mining.* Por meio do *Data Mining*, dados se tornam informação e conhecimento que são os pilares de suporte às tomadas de decisões realizadas no dia a dia de empresas (Wang, 2012).

Muitos trabalhos já exploraram os conceitos de previsibilidade de eventos adversos ou anomalias e a utilização de conceitos de *Data Mining*. Em Koutroulis, et al. (2023), por exemplo, foi aplicado algoritmos de detecção de anomalias e causalidade para tomadas de decisões ágeis a ataques cibernéticos em sistemas de controle industriais, tendo sido validado em um sistema de tratamento de água.

O trabalho de Perales Gómez, et al. (2020) foi outro que também desenvolveu e implementou um algoritmo para detecção de ataques cibernéticos em um sistema de tratamento de água, utilizando a metodologia nomeado por eles de *MADICS.*

Assim, considerando a relevância do tema e a necessidade de manter os parâmetros de qualidade da água na fabricação de espelhos sempre elevados, neste trabalho será apresentado os resultados do processo de *Data Mining* para uma fábrica de vidros, especificamente, analisando o setor de desmineralização da água utilizada na fabricação de espelhos.

Neste processo, será proposto, primeiramente, uma solução para detecção de operações anômalas para categorização automática de uma variável de processo nos estados de *operação normal*, *em deterioração* e *perigo*. Essa abordagem, em contraposição às metodologias clássicas de classificação da variável de processo em *warning* e *alarm* levará em consideração não apenas os limites inferior e superior de operação, mas também sua tendência como forma de categorizar o estado atual da variável do processo. Além disso, fornecerá também uma estimativa de tempo para aquela variável extrapolar os limites normais de operação, caso nenhuma ação de correção seja tomada.

Agregado a essa solução de alarmes, é implementada uma solução de extração de conhecimento para conectar o estado da variável de processo com a causalidade deste, de modo a fornecer à operação não apenas uma notificação de alarme, mas também sugestões de causalidade para que seja facilitada a tomada de decisão para correção dos parâmetros.

1.2 OBJETIVO GERAL

Este trabalho tem como objetivo desenvolver uma metodologia baseado em aprendizado de máquinas para detecção de operações anômalas apoiado a técnicas de *Data Mining* para extração de conhecimento, com o fim de fornecer suporte operacional para ações corretivas aos operadores de um fábrica de vidro, em especial, no setor de desmineralização da água que é utilizada na fabricação de espelhos.

1.2.1 Objetivos Específicos

Para cumprir com o objetivo geral, se espera concluir os seguintes objetivos específicos:

- Verificar a possibilidade de detectar desvios operacionais, por meio de algoritmo que seja capaz de antecipar a deterioração da variável do processo antes dela atingir os limites de segurança;
- Aplicar um algoritmo de classificação baseado em regras para extração de conhecimento e causalidade que indique o motivo da deterioração da variável do processo;
- Desenvolver uma representação visual que seja de fácil compreensão para operadores visualizarem alarmes e sugestões de ações corretivas baseados nas regras aprendidas pelo algoritmo de classificação e possam auxiliar em tomadas de decisão mais assertivas.

1.3 ESTRUTURA E ORGANIZAÇÃO DO TRABALHO

Além desta seção introdutória, este trabalho está organizado em mais 5 seções, descritos a seguir:

- Capítulo 2: este capítulo abrange os trabalhos relacionados, considerando abordagens similares que serviram como inspiração para o presente trabalho;
- Capítulo 3: neste capítulo é apresentado a fundamentação teórica necessária para compreensão da metodologia adotada;
- Capítulo 4: neste capítulo é apresentado a metodologia passo a passo do que foi desenvolvida, bem como o detalhamento da planta utilizada como estudo de caso;
- Capítulo 5: neste capítulo são apresentados os resultados e discussões da solução implementada, de modo a analisar seus pontos positivos e negativos;
- Capítulo 6: por fim, tem-se o capítulo de conclusão à qual estará expresso as considerações finais do autor quanto a solução desenvolvida bem como apresentar possíveis trabalhos futuros inseridos nessa mesma temática.

2. REVISÃO DA LITERATURA

2. REVISÃO DA LITERATURA

Em Munirathinam (2021) foi explorado a temática da necessidade de se avançar em novas abordagens para o monitoramento de variáveis industriais, em substituição as soluções mais antigas utilizando os conceitos de limites do Controle Estatístico de Processos, argumentando que esse tipo de monitoramento tende a ser ineficiente, por alertar tardiamente desvios na variável de processo e, consequentemente, não possibilitar ações preventivas na operação.

Nesse contexto, o autor propôs utilizar uma regressão linear dupla para identificação de *drifts* (desvios) agressivos e progressivos nos dados, em tempo real, bem como um método de *boxplot* para detecção de *outliers* tanto em dados com distribuições simétricas quanto assimétricas. O autor ainda acrescenta um parâmetro para calcular a gravidade do desvio, considerando o tempo de duração do desvio. Todavia, em seus estudos, não foi relacionado os limites mínimos e máximos da variável de processo ao seus *status* de alarme criados utilizando apenas os conceitos de normal (variação branda), *updrift, downdrift* e *outlier*.

Já Shi, et al. (2024) fundamentou seu trabalho baseado em detecção de anomalias em processos industriais apoiado no conceito de *Data Mining*, tendo em vista as dificuldades impostas em dados produzidos em ambiente industrial, como a presença de ruídos, redundâncias e assincronia entre dados. O método proposto combina tendência de dados e seu valor médio para proporcionar uma descrição compreensiva de eventos operacionais ocorridos. Ele inclui também um método para descobrir padrões comportamentais e sequências de informações que antecedem os eventos. O procedimento proposto é, primeiramente, a seleção dos dados, seguido de rotulação dos eventos pela tendência histórica (o autor propôs 7 tipos de eventos), aplicação de um modelo de árvore de decisão de classificação para aprender o que provoca estes eventos e, consequentemente, extrair conhecimento. Por último, a própria detecção do evento. Ao comparar toda sua metodologia proposta e sua precisão na detecção de eventos quaisquer, foi constatado que o algoritmo desenvolvido foi superior. No entanto, seu trabalho também não utiliza como referência os limites de controle da variável monitorado como suporte na detecção das anomalias.

No trabalho de Garmaroodi, et al. (2021) foi realizado um estudo de detecção de anomalias de um sistema de purificação de água, utilizando *Data Mining*. Segundo os autores, a detecção antecipada de falhas pode diminuir significantemente os custos de manutenção e garantir uma maior segurança na qualidade final da água. Para conseguir antecipar operações anômalas, foi proposto duas possíveis abordagens: a primeira é quando já se conhece as classes de anomalias e então se alimenta um algoritmo de aprendizagem supervisionada de classificação aprender a prever estes eventos; já na segunda abordagem os autores selecionaram dos dados, apenas aqueles que representavam a operação normal do sistema para treinar um modelo de regressão. Para determinar se o sistema estava em uma condição de falha, foi realizado o cálculo de resíduo entre a variável de processo real e a prevista pelo modelo, de modo que uma diferença maior que um gatilho pré-definido indicaria que o sistema estava operando em uma condição anômala. Como resultado, as duas abordagens cumpriram os seus propósitos de detecção de falhas, no entanto, devido as técnicas utilizadas (redes neurais artificiais e *Support Vector Machines*), não houve solução para interpretabilidade da causa de operações anômalas.

Já no estudo de Dix, et al. (2021), foi comparada a utilização de três topologias de redes neurais artificiais para predição de anomalias com base na reconstrução do erro. Para obter o efeito preditivo desejado, as topologias de redes neurais foram inicialmente treinadas apenas com dados de operações normais, que não estavam contaminados com situações de alarme pré-definidas. Posteriormente, após o treinamento, o modelo foi exposto a todas as condições operacionais, incluindo as operações anômalas. Quando o erro da previsão e o valor real da variável ultrapassavam um determinado limiar, uma operação anômala é reportada pelo modelo. Por fim, os autores propuseram uma metodologia de *Explainable* AI para determinar a causa raiz da falha, obtendo uma acurácia de até 99% nos 1.387 casos de alarme. Abordagens semelhantes foram utilizadas em Malhotra, et al. (2016) e Liu, et al. (2021), onde uma rede LSTM (*Long Short-Term Memory*) e ARIMA (*Autoregressive Integrated Moving Average*), respectivamente, foram aplicadas como modelos de decisão para prever eventos de alarme baseados em um gatilho (*trigger)* de erro.

Já estudo de Dix, et al. (2022), a proposta foi comparar o desempenho preditivo de três topologias de redes neurais para previsão de eventos de alarme com

antecedência de 10, 20, 40, 60 e 120 minutos, considerando que os eventos de alarme já eram conhecidos. A estimativa de tempo do modelo para a ocorrência do alarme mostrou-se promissora, capaz de prever um alarme com duas horas de antecedência e com uma precisão de aproximadamente 76%. No entanto, nenhuma proposição foi feita em relação à causalidade dos alarmes. Essa mesma abordagem foi realizada em Malhotra, et al. (2016), quando os autores treinaram uma rede LSTM para prever alarmes com base na reconstrução de sinais originais.

De maneira similar, o estudo de Boldt, et al. (2021), comparou modelos de aprendizado de máquina, como Random Forest, Support Vector Machines, Extreme Gradient Boosting e Redes Neurais, utilizando 231 variáveis como entrada em suas respectivas capacidades de prever a ocorrência de um alarme no futuro.

O tempo de antecedência analisado para os alarmes variou de 10 minutos a 48 horas, alcançando uma precisão de aproximadamente 52% para o maior tempo de antecedência e uma precisão ligeiramente inferior à de Dix, et al. (2022), com cerca de 72% para a previsão de 2 horas. Em conclusão, os autores forneceram uma classificação de importância das variáveis como meio de explicar os alarmes.

Em Bahar-Gogani, et al. (2017), um novo método é proposto para determinar se uma variável de controle está em estado de alarme. A abordagem substitui o gatilho de alarme tradicional, baseado em um limite pré-definido fixo, por um disparo dinâmico baseado em métricas estatísticas e funções de densidade de probabilidade. A principal conquista do estudo foi a redução de alarmes falsos e de alarmes oscilando entre estados normal e anormal. No entanto, o trabalho não abordou soluções para a causalidade dos alarmes.

Uma abordagem interessante foi proposta por Villalobos, et al. (2021), que utilizou duas topologias de redes neurais, LSTM (Long Short Term Memory) e ResNet (Rede Neural Residual), para detecção de alarmes. O método consistiu em duas etapas: na primeira, o LSTM previu uma janela de dados futura para que, posteriormente, suas previsões servissem de entrada para a ResNet para determinar se há um alarme presente e qual é o tipo de alarme.

Dentro desse contexto, percebe-se que trabalhos já exploraram esta temática de criação de um alarme inteligente, que utilize técnicas de *machine learning* para melhorar os alarmes atuais baseados apenas nos limites inferior e superior. No entanto, percebe-se que nos trabalhos não se conecta a informação de limites de

segurança da variável de processo com a tendência de desvio, nem se estima o tempo de degradação da variável até atingir estes limites.

Outros trabalhos seguiram mais a abordagem de previsão antecipada de um alarme para que houvesse um tempo hábil para a realização de manobras corretivas. No entanto, estes trabalhos pressupõem uma condição de alarme já bem definida.

Foi visto também que alguns dos trabalhos foram apoiados em técnicas de *Data Mining* para realizar a extração do conhecimento, mas ainda assim nenhum relacionou os elementos de limites da variável de processo, sua tendência histórica junto da extração do conhecimento. Na Tabela 1 é apresentado o resumo dos trabalhos avaliados.

TABELA 1: RESUMO DOS TRABALHOS RELACIONADOS.

Autor	**Tecnica Adotada**	**Estimativa de tempo**	**Abordagem adotada**
(Munirathinam, 2021)	Regressão Linear	Não consta	Rotulou a variável de processo em 5 condições a partir dos ângulos da regressão linear dupla.
(Shi, et al., 2024)	Valor Médio e Árvore de Decisão	Não consta	Rotulou a variável de processo em 7 condições a partir de seu valor médio e outros parâmetros estatísticos. Utilizou uma árvore de decisão para extrair causalidade para cada um dos 7 rótulos.
(Garmarodi, et al., 2021)	Redes Neurais e SVM	Não consta	Através de modelos de regressão treinados apenas com dados contendo operações normais era previsto condição de alarme quando o modelo errava mais que um certo limiar.
(Dix, et al., 2021)	LSTM e ARIMA	Não consta	Através de modelos de regressão treinados apenas com dados contendo operações normais era previsto condição de alarme quando o modelo errava mais que um certo limiar. Aplicaram Explanaible AI para visualização de

			conhecimento.
(Dix, et al., 2022)	Redes Neurais	Previsão de janelas temporais de até 2 horas	Treinou os modelos para preverem se um alarme aconteceria num futuro curto.
(Malhotra, et al., 2016)	LSTM	Previsão de alarmes em tempos variados	Previsão de alarme baseado na reconstrução do sinal original.
(Boldt, et al., 2021)	Random Forest, SVM, Extreme Gradient Boosting e Redes Neurais	Previsão de alarmes em tempos variados (minutos até 2 dias)	Os modelos previam se haveriam alarmes dentro da janela de dados definidos.
(Bahar-Gogani, et al., 2017)	Métricas Estatísticas	Não consta	Propõe uma nova forma para denunciar uma operação anômala como alarme baseado em métricas estatísticas.
(Villalobos, et al., 2021)	LSTM + ResNet	Não consta	Treina um LSTM para prever valores futuros e ResNet prever baseado na previsão da LSTM se há um alarme.

Tendo em vista estas abordagens adotadas nas pesquisas analisadas, este trabalho visa apresentar uma metodologia que crie um alarme preditivo que embarque as funcionalidades de rotulação de alarme baseado na tendência da variável, estime o tempo para alcançar um valor limite e tenha uma camada de interpretabilidade com capacidade de integração uma ferramenta *SCADA (Supervisory, Control and Data Acquistion)* de modo a auxiliar na tomada de ações corretivas operacionais.

3. FUNDAMENTAÇÃO TEÓRICA

3. FUNDAMENTAÇÃO TEÓRICA

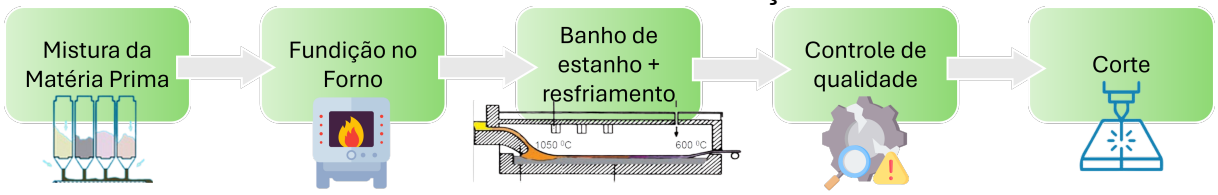
A fundamentação teórica está organizada da seguinte forma: na primeira parte será contextualizado o processo de fabricação de espelhos e como os parâmetros químicos da água impactam na qualidade final do espelho produzido; na segunda parte serão exploradas as problemáticas que envolvem sistemas de desmineralização de água e os principais equipamentos que são utilizados para fazer o trabalho de desmineralização da água; por fim, será abordada a teoria e algoritmos de *data mining* e como esse ramo da ciência de dados tem sido utilizado para beneficiar a operação de processos industriais; por fim

3.1 PROCESSO DE FABRICAÇÃO DE ESPELHOS

A fabricação de espelhos é uma etapa posterior do processo de fabricação de vidros planos, uma vez que os espelhos são um dos possíveis produtos resultantes da transformação do vidro. Segundo Freire (2016) as principais matérias-primas utilizadas são sílica (70%), barrilha (15%), calcário (10%), dolomita (2%), feldspato (2%) e aditivos compostos por sulfato de sódio, ferro e outros.

Assim, o processo de fabricação do vidro plano começa por meio da mistura de insumos e sua fundição em um forno; em seguida, o material fundido passa por um processo chamada de flutuação, que consiste em depositar o vidro fundido em um banho de estanho líquido que, com o devido tempo e controle adequado no resfriamento, é capaz formar uma lâmina de vidro de 5 a 6 mm de espessura, conforme foi resumido no trabalho de (Costa, 2006). Por fim, tem-se o controle de qualidade para descartar trechos que apresentem quaisquer tipos de defeitos indicados na norma NBR NM294 (ABNT, 2004) e o corte do vidro. Esse processo está ilustrado na Figura 1.

FIGURA 1: PROCESSO DE FABRICAÇÃO DE VIDRO.



A parcela de vidro que é reprovada no controle de qualidade costuma ser 100% reciclada e reutilizada como matéria-prima diminuindo os desperdícios do processo. Assim, uma vez que se tem vidros planos dentro das adequações exigidas nas normas técnicas, o vidro passa por uma segunda fase produtiva que envolve transformar o vidro em um novo produto, isto é, em espelhos, laminados, temperados, autolimpates entre outros (ABRAVIDRO, 2023).

Dessa forma, a partir do momento em que o vidro é cortado, quaisquer problemas que surgirem na sua transformação em espelho é problema exclusivamente de processos posteriores. Não se aplica a transformação de nenhuma peça de vidro defeituosa para evitar desperdícios no resto na cadeia produtiva.

Assim sendo, é iniciado o processo de fabricação de espelhos, que neste trabalho, será destacado o processo de espelhamento por meio da prateação do vidro, uma vez que é o método utilizado na planta do estudo de caso. No Brasil, os requisitos mínimos de controle que são exigidos na qualidade do espelho de prata seguem a norma NBR 14696 (ABNT, 2015).

O processo para fabricação começa pela limpeza da superfície do vidro, uma vez que a atmosfera possui contaminações que podem ser prejudiciais na qualidade final. Neste trabalho, o termo contaminações irá se referir a qualquer material e/ou energia sobre a superfície como contaminação, podendo se apresentar na forma líquida, sólida, iônica, orgânica ou inorgânica (Pulker, et al., 1999).

Sendo um dos processos essenciais, a limpeza da superfície é um processo complexo e delicado. A escolha da solução limpante deve ser forte o suficiente para reduzir a aderência da contaminação com a superfície do vidro: se a água utilizada for pura, por exemplo, em alguns tipos de contaminantes a remoção seria menos eficiente; entretanto, utilizando-se água quente com detergente estes mesmos contaminantes poderiam ser removidos eficientemente (Pulker, et al., 1999).

Uma vez limpo, a superfície deve ser enxaguada em água corrente ou destilada para posteriormente ser aplicada à solução de prata – que também leva água – de maneira bem distribuída sobre a superfície até secar (Curtis, 1911). Se a limpeza tiver sido adequada, a prata terá aderido bem a superfície e só restará a limpeza final para transformar em um produto comercial. O processo de fabricação do espelho está ilustrado na Figura 2.

FIGURA 2: PROCESSO DE PRODUÇÃO DE ESPELHOS.

3.2 SISTEMA DE DESMINERALIZAÇÃO DA ÁGUA

A água, como visto na seção anterior, é um dos pilares na eficiência na produção de espelhos, estando presente em vários processos, conforme vista na Seção 3.1. Dessa forma, é fundamental que a água fornecida para estes processos seja da qualidade adequada, uma vez que suas propriedades químicas possuem grande impacto no resultado.

Por este motivo, é incluso um sistema de desmineralização de água em indústrias de vidro, de modo a garantir que para cada fase do processo de espelhamento se tenha água com as propriedades químicas adequadas.

A respeito das propriedades químicas da água tem-se como as mais relevantes, no contexto da fabricação de espelhos, as seguintes propriedades, conforme definido em Belan (1981) e em Veolia (2024).

- **Alcalinidade:** refere-se ao conteúdo total de substâncias na água que causam o aumento de concentração de íons OH^-, a habilidade natural da água de neutralizar um ácido. Provocam corrosões em equipamentos industriais. Pode ser tratado por meio de tratamento ácido e desmineralização por troca de ânions;
- **Dureza:** é definido como a concentração de cálcio e magnésio na água. Pode provocar incrustações em equipamentos industriais.
- **Concentração de sílica:** é definido como a concentração de ácido sílico (H_2SiO_3) na água. Quando não é controlada, pode formar depósitos em superfícies difíceis de serem removidos. Pode ser controlado por meio de processos de aumento de temperatura da solução, uso de resinas de troca de ânions e osmose reversa.

- **pH:** refere-se à concentração de íon de hidrogênio. Varia de acordo com a acidez ou a alcalinidade na água. Pode ser tratada por meio do balanço de bases (aumentam o pH) e ácidos (diminuem o pH).

A água bruta, isto é, a água comumente encontrada em fontes naturais como rios, lagos e reservatórios de água também costumam ser a fonte para uso em processos industriais. Sua qualidade depende de vários fatores ambientais, como chuva, a água armazenada sob a superfície do solo, do uso, entre outros (Veolia, 2024), o que torna os desafios particulares para cada contexto de tratamento de águas industriais e, por isso, sendo fundamental que os equipamentos de tratamento sejam adequadamente projetados de acordo com as necessidades da indústria.

3.2.1 Equipamentos para desmineralização da água

Para controlar a qualidade da água, existe alguns processos físicos e químicos comuns de serem encontrados em sistemas de desmineralização da água. De forma a introduzir os processos mais importantes (e que estão presentes no estudo de caso), nas seções a seguir é elencado os equipamentos mais importantes, que são os filtros, a osmose reversa e trocador iônico e como eles afetam a qualidade da água.

a) Filtros

Segundo Nicholas, et al. (2002), o processo de filtração é fundamental, sendo responsável por separar partículas sólidas suspensas na água. Todo equipamento de filtragem opera fazendo a solução atravessar membranas porosas, de modo a reter partículas sólidas (como areia, metais e outros minerais). Costuma ser sempre o primeiro processo de tratamento da água antes dos processos que alteram de maneira mais precisa as propriedades químicas da água. Na Figura 3 tem-se diferentes tipos de filtros, cada qual com uma capacidade de filtração de um material diferente.

FIGURA 3: TIPOS DE FILTROS.

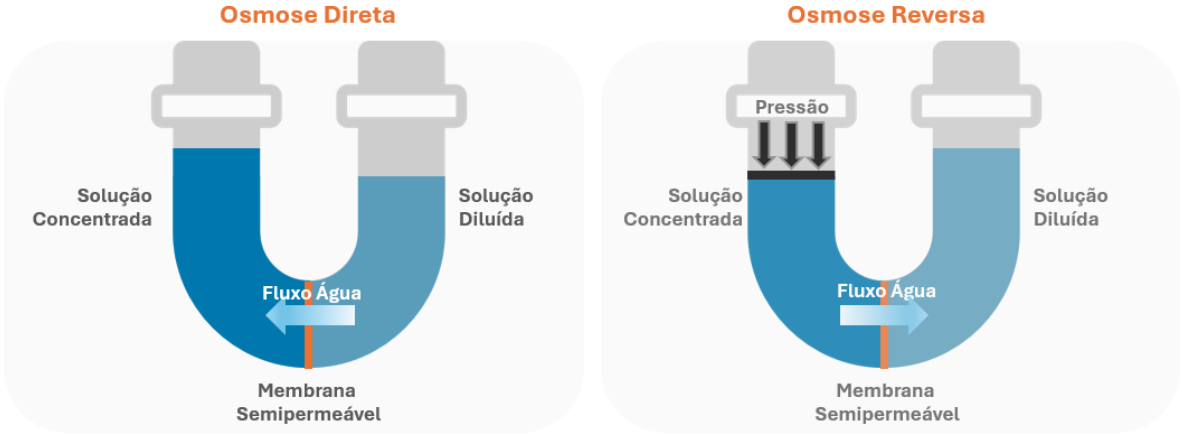
Fonte: Norspec, 2024.

Nesse contexto, os filtros não são capazes de remover sais dissolvidos que contribuem para a dureza e condutividade da água, nem afetam o pH da água.

b) Osmose Reversa

O processo de Osmose Reversa (OR) funciona exatamente como o próprio nome sugere. Na osmose direta, a água naturalmente tende ir do meio menos concentrado para o mais concentrado quando dividido por uma membrana semipermeável, para equilibrar as quantidades de soluto e solvente. Já na osmose reversa, o contrário acontece, por meio da aplicação de uma pressão no lado mais concentrado o que faz com que a concentração de soluto diminua. Na Figura 4 tem-se a ilustração deste processo.

FIGURA 4: PROCESSO DE OSMOSE REVERSA.



No contexto da desmineralização, a água bruta carregada de solutos ao atravessar o processo de osmose reversa e ser pressionada sobre a membrana semipermeável retém os solutos e permite a passagem do solvente, entregando uma água mais diluída para os processos posteriores.

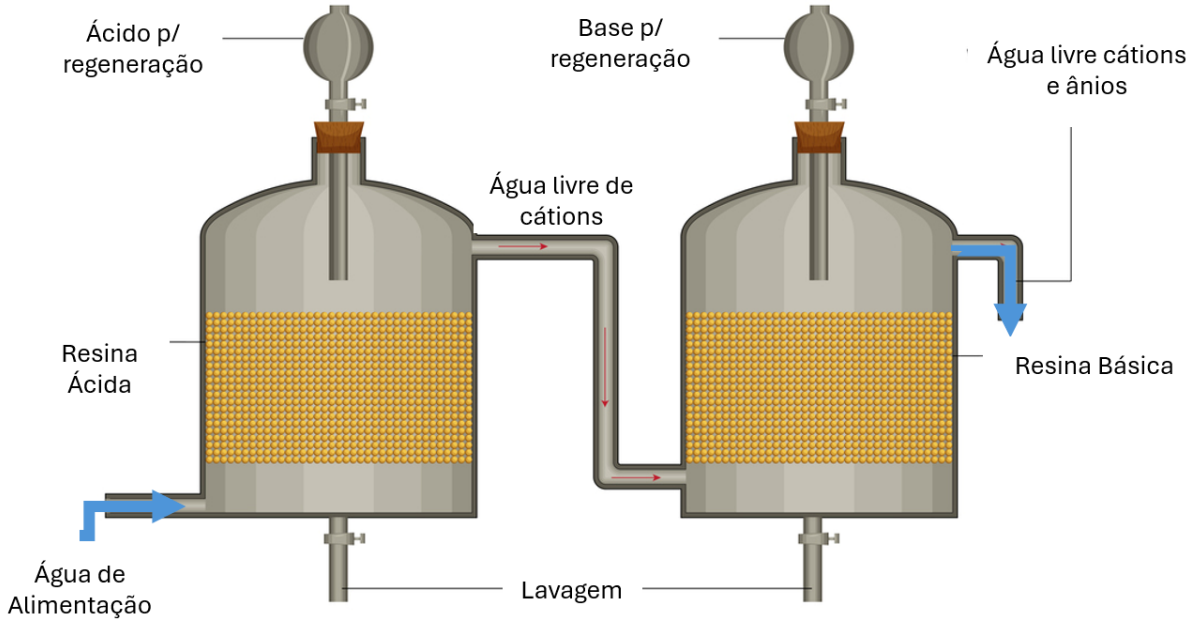
Segundo Nicholas, et al. (2002), a osmose reversa é capaz de rejeitar certas partículas de íons, moléculas e partículas sólidas que influenciam na dureza da água. A eficiência dessa rejeição depende de fatores com o tipo dos sólidos, os tipos de membranas a pressão e o pH da água de alimentação, sendo o ideal que a operação seja o máximo pressurizada o possível.

Os maiores problemas inerentes as máquinas de OR são a composição físico-química da água de alimentação e a necessidade de substituição das membranas periodicamente. Além disso, a OR tende a ser sensível a dureza da água, não sendo eficientes para lidarem com águas com alta concentração de ferro, manganês que podem danificar a membrana, além de possuírem sensibilidade a temperatura também. Ainda assim são capazes de reduzir boa parte da quantidade de sais dissolvidos que contribuem com a condutividade e dureza da água.

c) Trocador de Leito Misto

De acordo com Nicholas, et al. (2002), os trocadores de íons são exclusivamente adequados para a remoção iônica (ou redução de condutividade elétrica) da água. No contexto na desmineralização, neste processo, a água atravessa uma resina de troca catiônica (que requer um forte ácido) melhorando sua condutividade. No entanto, ao entrar em contato com essa resina a água estará parcialmente misturada com minerais ácidos. Assim, a água passa por uma resina absolvente de ácido (uma base fraca). Na Figura 5 é ilustrado o processo de troca iônica, que tem como resultado uma água livre de partículas ionizadas.

FIGURA 5: ILUSTRAÇÃO DE TROCADOR IÔNICO.



Fonte: EAIWATER, 2024.

No trocador de leito misto (TLM), as trocas catiônicas e aniônicas acontecem ao mesmo tempo dentro de um mesmo vaso, sendo um tipo de trocador iônico especial, mas tendo a mesma finalidade. Geralmente, costumam ser utilizados para fazer o polimento da água, isto é, fazer a água alcançar níveis de qualidade que os demais processos sozinhos não são capazes de fazer, entregando níveis de condutividade e contaminantes dissolvidos baixos.

Sua principal vantagem é sua capacidade de regeneração, não necessitando processos de substituição das resinas internas de maneira periódica conforme necessário no processo de osmose reversa, na substituição das membranas. No entanto, o tempo de regeneração também deve ser levado em consideração.

De maneira geral, são efetivos na redução da condutividade elétrica e dureza da água, mas podem ser prejudicados com uma água de alimentação com altos níveis de ferro e manganês, saturando mais rapidamente as resinas e acelerando o processo de regeneração.

3.3 *DATA MINING AND KNOWLEDGE DISCOVERY*

Segundo Wang (2012), *Data Mining* (Mineração de dados) e o KDD (*Knowledge Discovery from Databases* ou, traduzindo literalmente, Descoberta do Conhecimento em Banco de Dados) são tecnologias capazes de extrair informações e conhecimentos a partir de dados, que podem ser utilizados para o desenvolvimento de sistemas de espaços de estados.

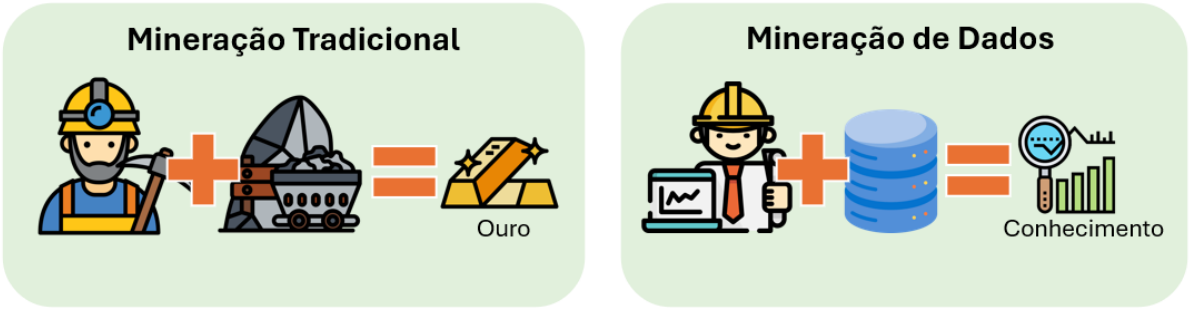
Já de acordo com Maimon, et al. (2005) o KDD é uma forma automática de realizar análise e modelagem de grandes conjuntos dados, sendo o *Data Mining* o motor do processo de descoberta do conhecimento, que envolve algoritmos de inferência, descoberta de padrões desconhecidos e entendimento de fenômenos a partir dos dados.

Seguindo uma definição mais simplificada e genérica, em Layton (2015), *Data Mining* é definido como uma metodologia que pode treinar computadores para tomar decisões a partir de dados, envolvendo ferramentas de ciências de dados como algoritmos, estatísticas, engenharia e otimização.

Por fim, em Han, et al. (2022), *Data Mining* é definido como um processo de descoberta de conhecimento e padrões de interesse a partir de um grande volume de dados, fazendo parte de uma metodologia para se alcançar o KDD.

Dentre todas essas definições, percebe-se que o principal conceito que converge entre elas é que o processo de *Data Mining* tem como o objetivo a extração automática de conhecimento a partir de dados, de modo que auxilie na tomada de decisões mais eficientes; pode-se entender melhor este conceito comparando, de maneira lúdica, a mineração de dados com a mineração de pedras preciosas, conforme ilustração da Figura 6. Enquanto na mineração tradicional deseja-se extrair minerais valiosos, na mineração de dados o artefato valioso que se busca são informações e conhecimentos.

FIGURA 6: MINERAÇÃO TRADICIONAL X MINERAÇÃO DE DADOS.



Para alcançar o KDD, a realização de algumas etapas prévias é necessária, tendo em vista a extração do conhecimento do objeto de estudo de maneira mais eficiente. Dessa forma, o *workflow* ideal detalhado em Wang (2012) sugere que há 9 etapas até a extração do conhecimento, a qual podem ser resumidas em 6 tarefas, conforme ilustrado na Figura 7.

FIGURA 7: WORKFLOW PARA DESCOBERTA DE CONHECIMENTO.

Essas 6 tarefas, de forma mais detalhada representam os seguintes processos:

1. Entendimento do objeto de estudo, registrando os conhecimentos prévios existentes e os objetivos do usuário final;
2. Coleta de dados a partir dos diferentes bancos de dados, agrupando todas as variáveis de interesse em um único ambiente (*Data Warehouse*);
3. Limpeza de dados, realizando operações para tratamento/remoção de ruídos e *outliers* e aplicação de estratégias contra dados faltantes e com diferentes tempos de amostragem;

4. Transformação dos dados através de métodos de normalização de dados, redução de dimensionalidade e criação de novos atributos;
5. Aplicação de técnicas de *Data Mining* de acordo com o objetivo do processo de descoberta de conhecimento, seja ele sumarização, classificação, regressão, clusterização, detecção de padrões, extração de regras, detecção de anomalias etc.;
6. Interpretação, consolidação e apresentação dos conhecimentos minerados aos quais devem ser validados com os usuários, checando os possíveis conflitos existentes entre o conhecimento extraído e o conhecimento prévio.

3.3.1 Pré-Processamento de dados

A fase de pré-processamento dos dados é uma das etapas mais custosas nas tarefas de *machine learning*, uma vez que os dados precisam ser devidamente analisados e adequados antes de serem utilizados pois, primeiramente, é primordial que os dados estejam sincronizados no tempo, principalmente se a modelagem que está relacionada com comportamentos dinâmicos e temporais. Além disso, o tratamento dos dados é importante para extrair dos modelos o máximo desempenho que são prejudicados na presença de dados faltantes e outliers.

Entre as principais técnicas de pré-processamento que serão usadas neste trabalho estão a média móvel e Local Outlier Factor, as quais serão descritas em seguida.

a) Média Móvel

A média móvel é uma das possibilidades que se tem quando se deseja, suavizar *outliers*, preencher dados não numéricos (NaN) via interpolação ou simplesmente para aplicação de uma reamostragem nos dados. Por meio de uma janela deslizante de dados, se aplica a média em uma série temporal, tomando uma janela com n amostras de dados para aplicação da média que represente aquele pequeno conjunto de dados. Mas, como a média é “móvel” a seleção da janela para cálculo sempre vai deslizando sequencialmente. Assim, o cálculo do termo de uma

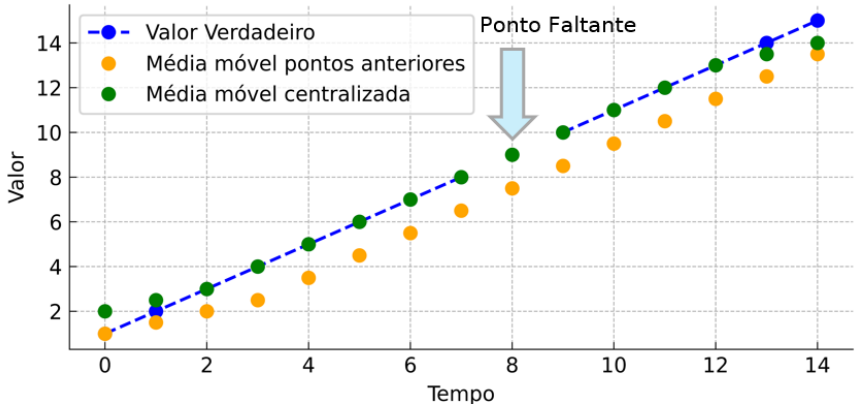
sequência é dado pela Equação 1, onde n é o número de amostra de dados, p uma sequência de valores numéricos e $\overline{p_{n+1}}$ a média móvel calculada.

$$\overline{p_{n+1}} = \frac{1}{n} \cdot \sum_{j=1}^{n} p_j \tag{1}$$

Existe um aspecto importante, no entanto, que deve ser considerado na aplicação da média móvel. A posição do ponto a qual será aplicado a média móvel em relação a janela de dados selecionada. Pode-se aplicar a média em relação as últimas *n* amostras e substituir a amostra atual ou aplicar a média móvel nas *n/2* últimas amostras junto das *n/2* amostras seguintes, de modo que o valor da média estará centrado no meio entre essas duas janelas.

Do ponto de vista de um comportamento temporal mais realístico, essa segunda abordagem deve ser preterida por ser capaz de levar em consideração o comportamento futuro da variável, conforme pode ser visto a Figura 8.

FIGURA 8: EFEITOS DA MÉDIA MÓVEL PARA PREENCHIMENTO DE DADOS FALTANTES.

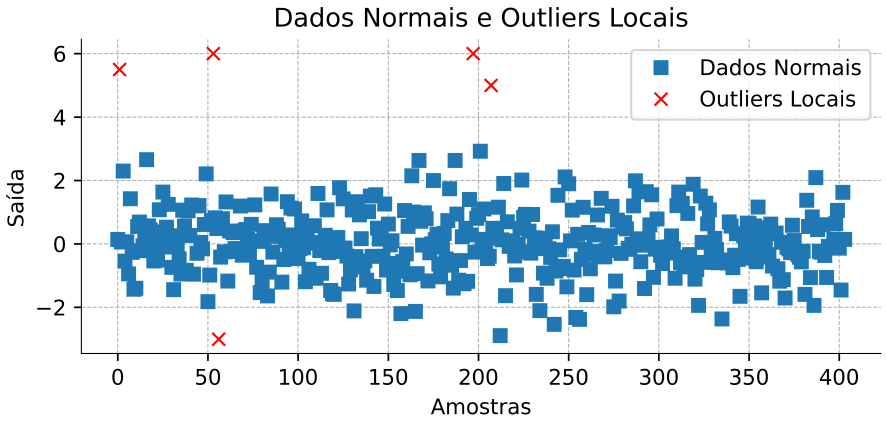


b) Local Outlier Factor

O *Local Outlier Factor* (LOF) é um algoritmo baseado em distância que vem sendo utilizado na literatura para o tratamento de *outliers* (Alghushairy, et al., 2021). Quanto mais afastado uma certa amostra de dado estiver de sua distribuição, maiores são as chances de ser apontado como um *outlier*, de acordo com este algoritmo (Breunig, et al., 2000).

Para definir *outlier,* estão sendo seguidas as definições enunciadas em Hawkins (1980) que define *outlier* como sendo “uma observação que se desvia tanto de outras observações a ponto de levantar suspeitas de que foi gerada por um mecanismo diferente”. Isso fica mais claro ao se analisar a Figura 9, onde percebe-se que uma grande distribuição de dados centrados dentro de uma certa média e desvio padrão, ao mesmo tempo que se observa dados que estão muito afastados dessa zona, sendo estes sendo considerados *outliers*.

FIGURA 9: EXEMPLOS DE *OUTLIERS*.



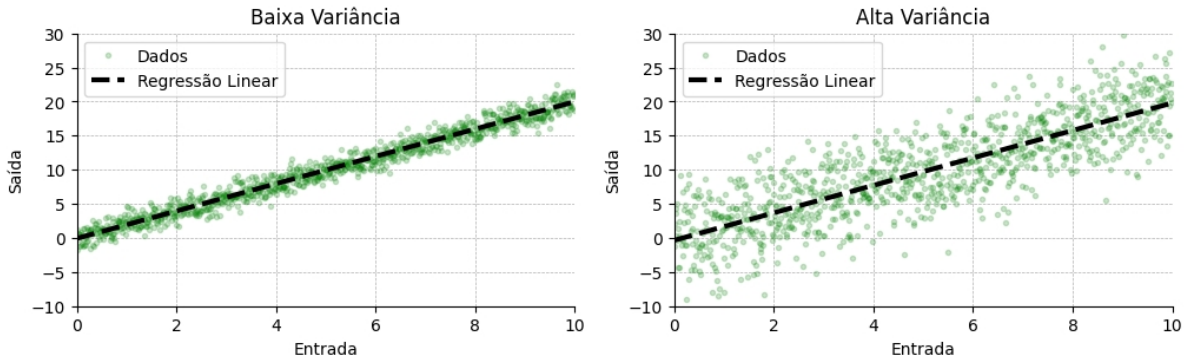
3.3.2 Regressão Linear

De acordo com Bonaccorso (2018), o modelo de regressão linear (RL) é baseado na hipótese de que é possível aproximar os valores de uma variável de saída por meio de uma combinação linear das entradas, conforme a equação x, onde $\tilde{y_k}$ é o valor de saída estimado do valor de y_k calculado a partir de equação polinomial baseada nos valores de entrada x_k. Ainda se tem as constantes a_0 e a_i representando os coeficientes que relacionam as entradas com as saídas e n representando o número máximo de entradas.

$$\tilde{y_k}\left(\alpha_0, \ldots, \alpha_n\right) = a_0 + \sum_{i=1}^{n} a_i \cdot x_k \tag{2}$$

Para usar este algoritmo e ter um desempenho satisfatório para a aplicação, deve-se levar em consideração que a relação entre os dados deve ser baseada em estruturas planas (como linhas, planos e hiperplanos), uma vez que não linearidade e alta dispersão prejudicam o desempenho deste modelo. Na Figura 10 é apresentado um exemplo do comportamento da regressão linear para uma certa função $y = f(x)$ acrescido de ruídos com variâncias diferentes (no gráfico da esquerda, a variância é menor que no da direita).

FIGURA 10: REGRESSÃO LINEAR PARA DADOS COM BAIXA E ALTA VARIÂNCIAS.



a) b)

Na Figura 10a) é possível notar uma relação entre a entrada e a saída devido à baixa variação dos dados enquanto na direita uma relação mais fraca, dada pela dispersão nos dados. Nota-se também que, no exemplo na Figura 10b), apesar da regressão linear não ter aproximado tão precisamente sua estimativa em relação ao caso da esquerda, a tendência da saída foi capturada, pois se for comparadas as duas retas geradas, tanto suas inclinações quanto suas constante são aproximadas.

Para ilustrar os conceitos bases da regressão linear, nas seções a seguir serão apresentados os processos de treinamento para encontrar a regressão linear baseado em mínimos quadrados e a interpretação visual associada a tendência ou variação de uma determinada variável.

a) Processo de Treinamento

Para aproximar as previsões do modelo com a saída real prevista, a regressão linear usa o método dos mínimos quadrados, que tem o objetivo minimizar os resíduos

da soma dos quadrados (Weisberg, 2005), dada pela Equação (3). Nesse exemplo, só foi considerado uma única entrada e uma única saída para facilitar a compreensão, onde α_0 é coeficiente de interceptação e α_1 o coeficiente de inclinação; J representa a função de custo que precisa ser minimizada, a qual $\tilde{y}$ representa o valor estimado pelo algoritmo dado a Equação (3), y e x os valores de saídas e entradas reais respectivamente.

$$J\left(\alpha_0, \alpha_1\right) = \frac{1}{2 \cdot m} \sum_{i=1}^{m} \left(\tilde{y}_{\,i\,(x_i)} - y_i\right)^2 \tag{3}$$

Assim, por meio da função, os coeficientes são atualizados até que se minimize ao máximo a função de custo dada pelo método dos mínimos quadrados. Para fazer isso, pode-se aplicar diferentes técnicas de minimização, mas um método comumente utilizado (James, et al., 2013), onde $\bar{x}$ é a média dos valores conforme as Equações (4) e (5).

$$\alpha_1 = \frac{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)\left(y_i - \bar{y}\right)}{\sum_{i=1}^{n} \left(x_i - \bar{x}\right)^2}, \tag{4}$$

$$\alpha_0 = \bar{y} - \alpha_1 \cdot \bar{x} \tag{5}$$

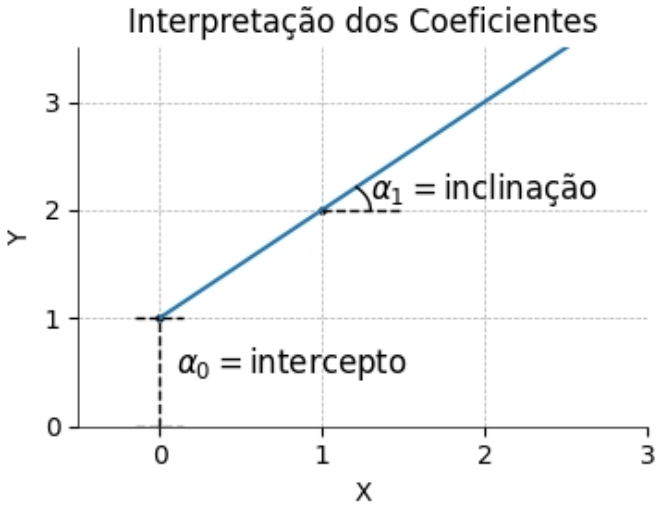
b) Interpretação do algoritmo

A regressão linear se destaca por ter coeficientes matemáticos simples de serem interpretados e rápido processo de treinamento quando comparados a demais modelos de *machine learning* como redes neurais artificiais. Quando se faz a modelagem considerando apenas uma única entrada e saída, sua interpretação fica direta, uma vez se tem apenas um coeficiente representando um *offset* e outro a inclinação da reta que relaciona a proporção da qual uma entrada modifica uma saída.

Quando o coeficiente de inclinação é negativo, significa que o aumento/diminuição no valor da variável de entrada está relacionado com a diminuição/aumento da variável de saída. O módulo do coeficiente de inclinação ditará ainda a intensidade na qual uma variável de saída é modificada, quanto maior/menor for o coeficiente maior/menor será a variação na variável de saída.

A interpretação visual deste algoritmo é dada na Figura 11. Neste exemplo considerando uma única entrada x e uma única saída y, tem-se os coeficientes a_0 também chamado de intercepto ou interceptação e a_1 que representação a inclinação da reta.

FIGURA 11: COEFICIENTES DA REGRESSÃO LINEAR.



3.3.3 Árvores de Decisão

De acordo com Bonaccorso (2018), as árvores de decisão são estruturas baseadas num processo de decisão sequencial. Conceitualmente, imita a estrutura de uma árvore, como as abstrações de raiz, galhos e folhas além de utilizar expressões condicionais (do tipo *se* <*condição*> *então* <...> *senão* <...>) para aproximar a saída prevista coma saída real.

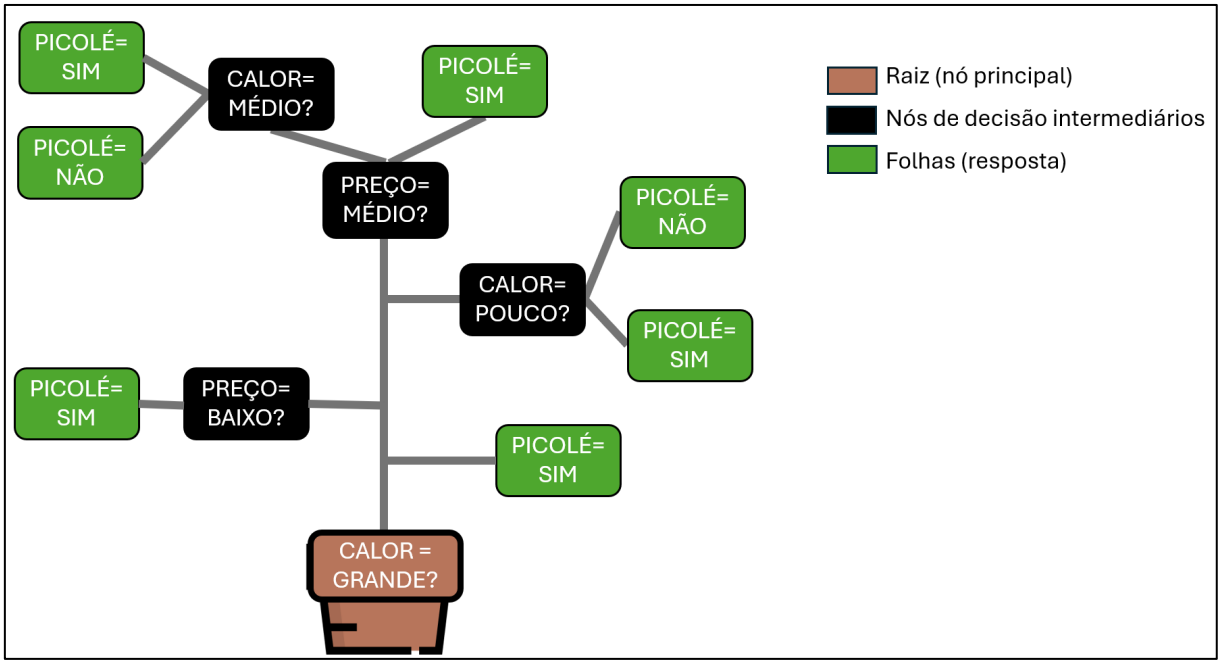
No contexto do algoritmo, tem-se os seguintes parâmetros Bonaccorso (2018):

- Raiz: é o nó inicial da árvore de decisão, sendo o atributo principal dentre as entradas que é utilizada como referência para a primeira tomada de decisão condicional;

- Galhos: representam os caminhos que ligam os nós de decisão, na qual cada galho é resultado da decisão criada no nó;
- Nós intermediários: todo ponto de decisão é um nó, a qual o primeiro se chama raiz e os demais são os intermediários;
- Folhas: as folhas representam os nós finais de decisão, a qual se obtém a resposta final da classificação, após o caminho de decisão percorrido na árvore.
- Profundidade: refere-se à quantidade de ramificações máximas que o algoritmo pode criar, isto é, o número de nós de decisão em sequência, além da raiz, que podem ser criados.

Para trazer maior compreensão, na Figura 12 são apresentados esses parâmetros numa árvore de decisão fazendo alusão dos elementos de uma árvore ao fluxograma resultado do treinamento do modelo de árvore de decisão. Como pode ser visto, a interpretabilidade do algoritmo é simplificada permitindo que não especialistas facilmente compreendam o processo de tomada de decisão do algoritmo e, por consequência, aprendam a dinâmica de um processo complexo.

FIGURA 12: ÁRVORE DE DECISÃO ESQUEMATIZADO.



Algumas das outras vantagens deste algoritmo estão na sua capacidade de conseguir lidar tanto com entradas numéricas quanto com entradas categóricas sem perder desempenho; além disso, não necessita que os dados de entradas sejam transformados em processos de normalização de dados. Três pontos sobre este algoritmo serão detalhados a seguir: o processo de treinamento, a interpretação do algoritmo e formas de validar desempenho.

a) Processo de treinamento

Assim como outros algoritmos de *machine learning,* a árvore de decisão possui sua própria metodologia para minimizar o erro entre a saída prevista e a real dos dados de treinamento. Seu processo de treinamento consiste dos seguintes passos, conforme descrito em Han, et al. (2011), a qual o objetivo principal é encontrar a melhor maneira de criar os nós de decisão mais assertivos, que melhor dividem a informação de saída. A principal equação que funciona como função de perda, ou objetivo, é o índice de impureza de $Gini$, onde $\mathrm{p}^2\left(\mathrm{c_i}\right)$ representa a probabilidade de que uma classe c_i esteja fora da população e n é o número de classes possíveis, representado na Equação (6).

$$Gini = 1 - \sum_{i=1}^{n} p^2\left(c_i\right) \tag{6}$$

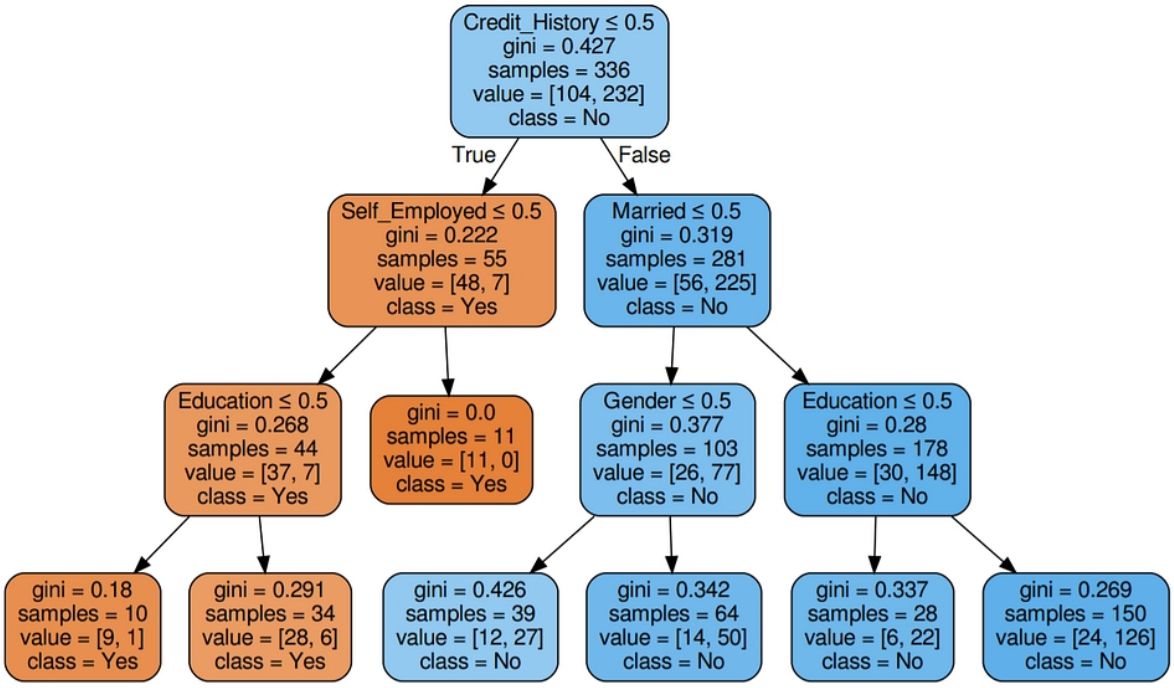
Um alto valor de $Gini$ implica uma população heterogênea (muitas amostras de várias classes), enquanto um baixo valor de $Gini$ indica uma população homogênea (composta principalmente de uma única classe). Assim, o objetivo é alcançar o índice $Gini$ igual a zero, a qual para aquele nó de decisão o resultado é uma única classe.

b) Interpretação do algoritmo

Como citado anteriormente, uma das qualidades existentes no algoritmo de árvore de decisão está na sua fácil interpretabilidade, pois, uma vez que a árvore já está treinada é possível ter uma visualização de seu processo de inferência no formato de um fluxograma, conforme pode ser visto na Figura 13. Neste exemplo qualquer,

tem-se como entradas as variáveis “Credit_History”, “Self_employed”, “Married”, “Education” e “Genre” que devem ser utilizadas como regras para classificar a saída “class” que assume os valores “Yes” e “No”. A informação de “samples” indica quantos exemplos de dados aquele nó de decisão inclui. Ou seja, para o nó raiz de exemplo, tem-se um total de 336 amostras que, a partir da decisão condicional, são divididos para dois nós intermediários abaixo, um com 55 amostras e outro com 281 amostras. O termo *class* em cada nó indica a classe dominante naquele nó. Já a informação de *value* em cada nó indica a proporção de classes presentes em cada nó. Por fim, tem-se o gini, que quanto menor mais homogêneo aquele nó de decisão é em relação a classificação, pois o objetivo final de um modelo perfeito de árvore de decisão é que as folhas contenham apenas uma única classe.

FIGURA 13: FLUXOGRAMA DE ESTIMAÇÃO DE UMA ÁRVORE DE DECISÃO.



Fonte: Revelo, 2024.

c) Validação da classificação

Para validar a precisão do modelo em relação ao seu aprendizado, neste trabalho serão adotadas duas métricas: a acurácia e a acurácia balanceada. A primeira por medir um desempenho geral do modelo em relação a todo o conjunto de dados. No entanto, em conjuntos de dados desbalanceados, esse tipo de métrica pode acabar sendo enviesada pela classe majoritária.

Dessa forma, para compensar o desbalanceamento, também será utilizada a métrica de acurácia balanceada, prevendo que o conjunto de dados possui desbalanceamento considerando que operações anômalas ocorram com bem menos frequências que operações normais. Na Equações (7) e (8) têm-se as representações matemáticas destas duas métricas, onde VP e VN significam verdadeiros positivos e negativos, respectivamente e FP e FN falsos positivos e negativos respectivamente.

$$acuracia = \frac{VP + VN}{VP + FN + VN + FP} \tag{7}$$

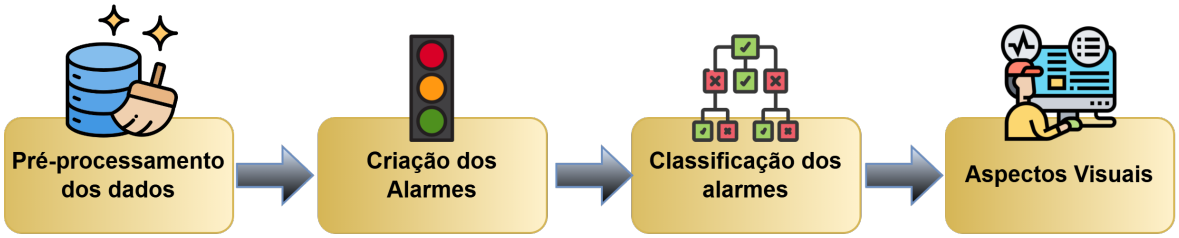
$$acuracia\ balanceada = \frac{1}{2} \cdot \left(\frac{VP}{VP + FN} + \frac{VN}{VN + FP} \right) \tag{8}$$

4. METODOLOGIA

4. METODOLOGIA

Neste capítulo, será descrita a metodologia adotada para o desenvolvimento deste trabalho bem como o detalhamento do cenário de estudo. Para dar uma primeira visão geral acerca dos procedimentos que foram desenvolvidos e que serão detalhadamente explicados nas seções a seguir, é apresentado na Figura 14 um esquemático das atividades realizadas e como cada uma delas combinam entre si.

FIGURA 14: METODOLOGIA PROPOSTA NO TRABALHO.



Assim, conforme ilustração da Figura 14, a solução proposta foi desenvolvida com o seguinte procedimento:

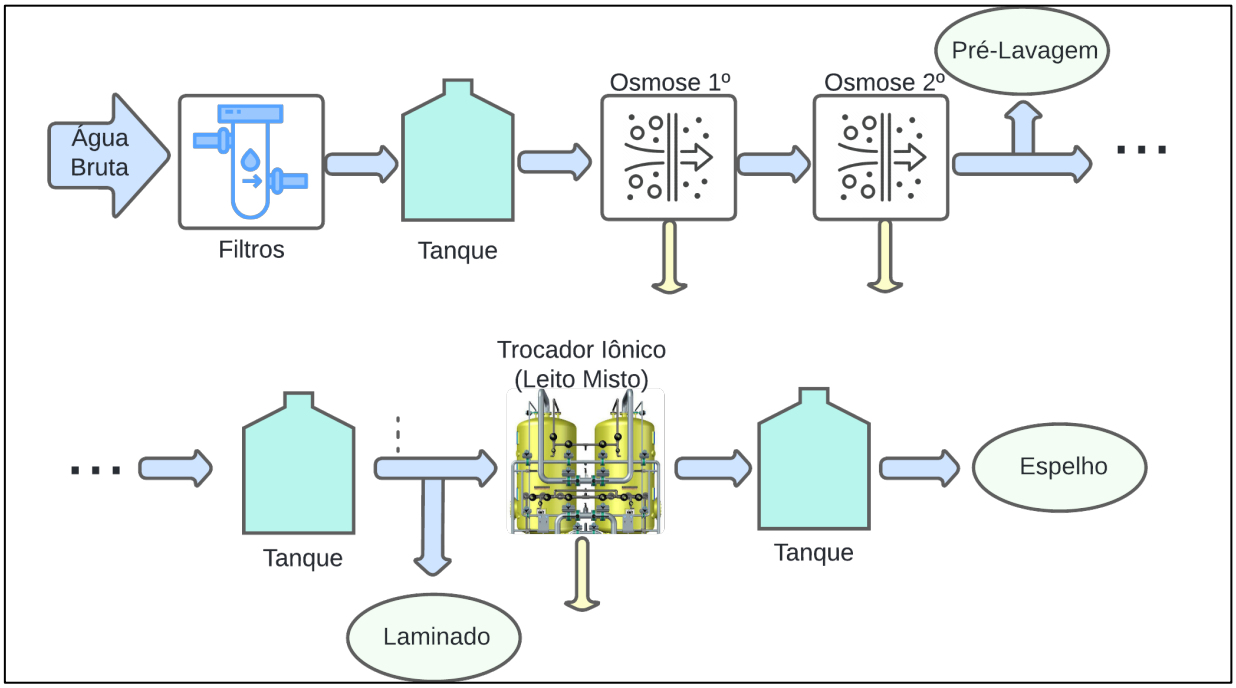
i. Pré-processamento automático dos dados: preparação dos dados para entrada no modelo de regressão linear;
ii. Criação de alarmes pelo algoritmo de regressão linear: usar a inclinação da reta gerada pela regressão linear para prever a tendência de desvio de uma variável de processo;
iii. Classificação dos alarmes utilizando algoritmos baseados em regras: interpretação das causas dos alarmes baseado nas regras criadas;
iv. Análise dos resultados e explicabilidade dos alarmes: implementação do algoritmo completo integrado a um SCADA implementando recursos visuais de simples interpretação para tomadas de decisão.

4.1 MATERIAIS

Para fazer o estudo de caso deste trabalho, foi utilizado o setor de desmineralização de água de uma fábrica de vidros, localizada no Brasil, responsável por tratar a água bruta que entra neste sistema para tornar apropriada sua utilização nos processos de laminação e espelhamento dos vidros planos, que exigem qualidades químicas na água específicas.

Esse sistema de desmineralização, tal qual esquematizado de maneira simplificada na Figura 15, conta com várias etapas de tratamento especializados por tratar cada uma das qualidades discutidas anteriormente: a dureza, condutividade e alcalinidade (pH) da água. Esse sistema é composto por filtros na entrada, dois processos de osmose reversa e um trocador de leito misto; a água, representada pelas setas azuis, é enviada para três processos principais que são o processo de pré-lavagem do espelho, o de laminação do vidro e o espelhamento, representados pelos círculos amarelos. Por fim, as setas amarelas representam pontos na qual a água é rejeitada.

FIGURA 15: CENÁRIO DE ESTUDO SIMPLIFICADO.



Nesse sistema encontram-se distribuídos sensores de diferentes tipos, sejam eles conectados diretamente na linha do fluxo da água ou nos tanques. São sensores

de medição de pressão manométrica e diferencial, níveis de tanque, vazão no trecho, além dos parâmetros de qualidade da água de pH, condutividade e dureza, com intervalos entre amostras de aproximadamente 10 segundos. Para resumir todas essas variáveis, na Tabela 8 presente no Anexo 1, são apresentadas todas as variáveis medidas consideradas.

Sobre os limites operacionais, mínimos e máximos de segurança na Tabela 2 são apresentados estes valores para cada um dos sensores de qualidade da água, em especial, destacando o sensor de pH localizado na saída do TLM, que mede a qualidade da água que chega para o processo de fabricação do espelho o qual serão apresentados os resultados mais detalhados neste trabalho.

TABELA 2: LIMITES SUPERIOR E INFERIOR DAS VARIÁVEIS DE PROCESSOS.

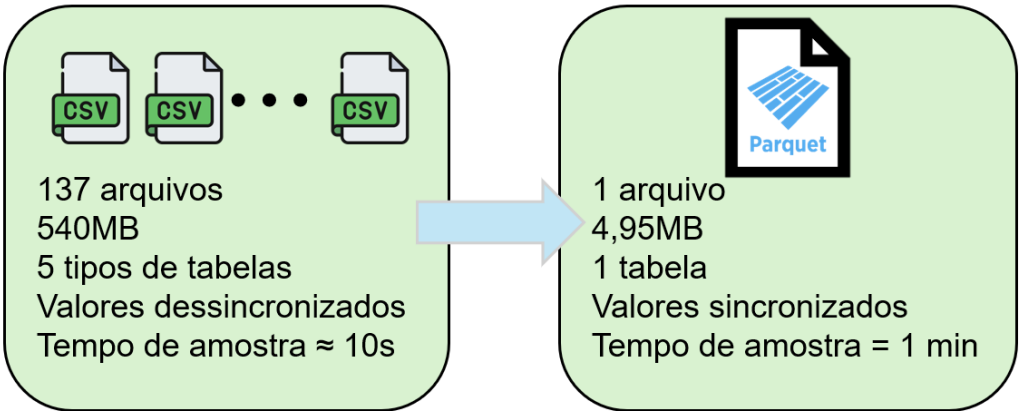
	AIT602 (pH)	AIT616 (pH)	AIT603 (uS/cm)	AIT604 (ppb)	AIT610 (pH)	AIT607 (uS/cm)	AIT611 (pH)	AIT614 (uS/cm)	AIT605 (uS/cm)
Limite Superior	8	7,5	0	0	5	**0**	5	0	0
Limite Inferior	9	9	60	3,5	7	**1**	6	0,8	10
Localiza-ção	TANQUE 1	OR2	OR2	OR2	TANQUE 3	**TLM**	ESPELHO	ESPELHO	TLM

4.1.1 Extração e pré-processamento dos dados

A extração dos dados foi realizada por meio do *software* Grafana (Grafana Labs, 2018), que contém conexão com os diversos bancos de dados da fábrica conectados em seu serviço. Assim, os dados eram baixados a partir de vários arquivos em formato CSV (*Comma Separeted* Values) e unidos pela mesma amostragem de tempo.

De maneira geral, os dados eram amostrados a cada 10 segundos e muitos deles não possuíam sincronia entre si, tendo em vista a presença de fontes de aquisição dos sensores diferentes, que dependiam da configuração dos IHMs (Interface Homem Máquina) localizados na planta. Na Figura 16, é resumido a características dos dados e como eles foram agrupados para um único arquivo de extensão “parquet”.

FIGURA 16: PRÉ-TRATAMENTO DOS DADOS.



Dada a natureza do processo que envolve reações químicas e físicas mais lentas quando comparado a processos que envolvem controle constante e preciso da variável de processo, todos os dados foram reamostrados para o intervalo de um minuto usando a média móvel, para não haver muita perda informação, suavizar *outliers* e, principalmente, fazer uma sincronia artificial entre os dados.

Por fim, para que a presença de *outliers* fosse reduzida de maneira adequada, também foi acoplada a técnica de detecção de *outliers* LOF, que por meio de sua detecção foram removidas. Assim, os dados ficaram preparados para serem utilizados nas tarefas de *machine learning* propostas.

4.2 MÉTODOS

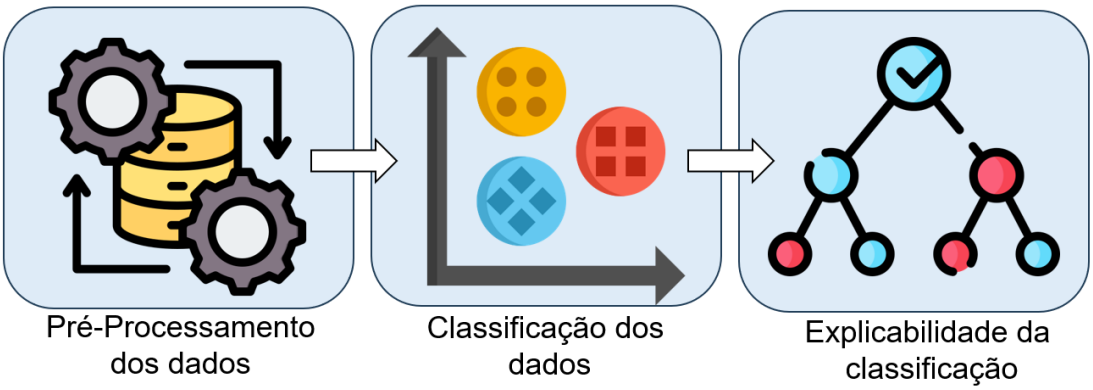
Nas seções a seguir serão elucidadas as técnicas aplicadas para o desenvolvimento do trabalho, iniciando pela criação dos alarmes, classificação via árvores de decisão, validação e até a apresentação da extração do conhecimento adquirido.

4.2.1 Criação de alarmes inteligentes

Para a construção dos alarmes inteligentes foi adotada os princípios da metodologia de extração do conhecimento a partir do *Data Mining*, desde o pré-processamento até a fase de interpretação e extração do conhecimento gerado.

Assim, a abordagem adotada para se alcançar o conhecimento seguiu o esquema ilustrado na Figura 17, iniciando com a preparação dos dados, para aplicação algoritmo de regressão linear e rotulação dos dados em três categorias: operação normal, deterioração e perigo; e, por fim, a etapa de tornar os resultados do algoritmo úteis no aspecto de sua interpretação, utilizando um algoritmo baseado em regras que sugira a motivação do desvio.

FIGURA 17: METODOLOGIA DE CRIAÇÃO DE ALARMES INTELIGENTES



De maneira geral, a proposta deste processo permite que se automatize a criação de alarmes inteligentes que possam ser explicáveis no dia a dia da operação. Pois, apesar de ser interessante o conhecimento antecipado das tendências de desvios de uma variável crítica para a qualidade do processo, o conhecimento da motivação do porquê aquela variável estar naquele estado enriquece ainda mais a informação do alarme. Essa explicabilidade será proporcionada por meio de um algoritmo classificador baseado em regras.

Assim, nas seções a seguir serão detalhadas o procedimento para detecção desvios, como a regressão linear auxilia nesta etapa e como funções matemáticas auxiliares são acopladas ao modelo para tornar o sistema de detecção mais robusto.

a) Detecção de tendência de desvios

Uma vez que determinada as variáveis dos processos que deverão ser monitoradas, buscou-se alternativas para criação de um alarme inteligente que fosse capaz de antecipar a tendência de deterioração de um parâmetro medido com um

mínimo de apresentação de falsos alarmes, que poderiam provocar decisões precipitadas por parte da operação ao se deparar com este tipo de sinalização.

Portanto, os requisitos mínimos que este algoritmo detector de alarmes deveria atender são os seguintes:

- Categorizar a variável de processo em 3 classes: *operação normal*, *em deterioração* e *perigo;*
- Levar em consideração os limites mínimos e máximos da variável de processo;
- Considerar a tendência de subida ou descida da variável de processo a partir de uma janela temporal;
- Quanto mais próximo o valor da variável de processo estiver de seus limites de segurança menor precisa ser a tendência para que o detector acuse estado de deterioração;
- Estimar o tempo que a variável de processo deverá alcançar um estado crítico se mantiver a tendência;
- Evitar sinalização da variável de processo como *em deterioração* ao se deparar com desvios típicos de mudanças bruscas (ou *outliers*);
- Sugerir o subprocesso causador de um desvio na variável de processo.

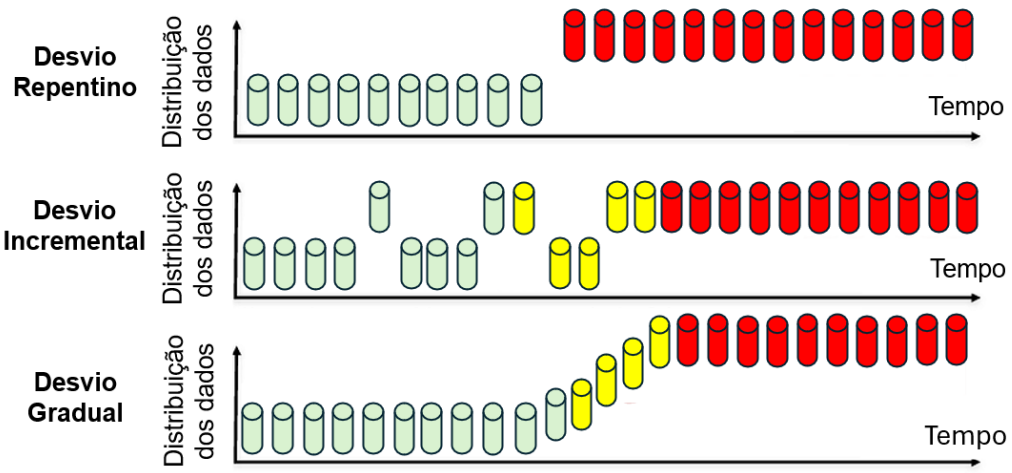
De maneira ilustrativa na Figura 18, é desejado que o algoritmo seja capaz de se comportar conforme os dados apresentados no gráficos da figura que, para cada diferente comportamento da variável de processo, a acusação de alarme seja diferente, considerando verde para sinalizar operação normal, amarelo para tendência de deterioração e vermelho para perigo, estado crítico alcançado.

O estado de operação normal, indica que a variável de controle está dentro de seus limites superior e inferior e não está com tendência de superar um desses dois limites; já o estado de deterioração indica que a variável de controle está dentro de seus limites, mas sua tendência sugere que ela irá superar os limites em breve; por fim, o estado crítico acontece quando a variável de controle está fora de seus limites.

No primeiro tipo de desvio, optous-e por não haver um estado intermediário devido a possível urgência que esse comportamento possa significar; já no desvio incremental tem-se desvios típicos de *outliers* e transição de operações que se optou por não considerar importante a não ser que estes desvios fiquem mais consistentes,

por fim, no desvio gradual se deseja que seja percebível os sinais de deterioração antecipadamente e acusado prontamente como alarme.

FIGURA 18: COMPORTAMENTO ESPERADO DO ALGORITMO.



A alternativa proposta para cumprir com esta etapa do trabalho, foi a utilização de uma regressão linear como ferramenta matemática para classificação do estado de uma variável de processo. Em termos gerais, é considerada como variável de entrada as últimas *x* horas de dados e como saída as últimas *y* medições da variável (dentro das *x* horas de dados), de modo que um modelo de regressão linear seja desenvolvido em relação a esta janela de dados gerando os coeficientes que formam a representação da equação reta.

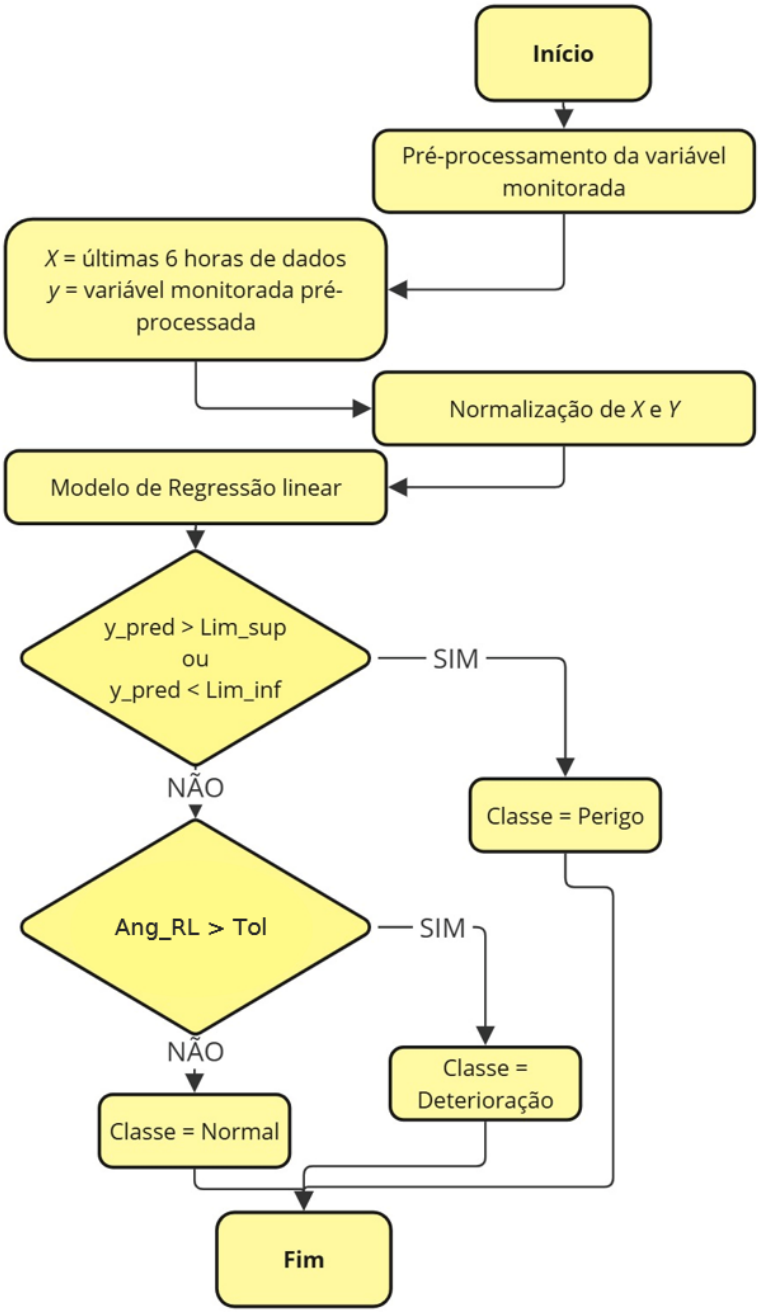
Dessa forma, o ponto-chave da utilização deste algoritmo está nos coeficientes gerados pela regressão linear, em especial, o coeficiente que representa o ângulo de inclinação da reta. Foi por meio dele, que foi possível categorizar o estado da variável quanto a sua tendência de desvio de seus limites de atuação, de modo que se o valor absoluto dessa inclinação calculada superasse um certo valor máximo de tolerância, se assumiria que aquela variável estava em tendência de deterioração.

Assim a tolerância é o gatilho de decisão para estabelecer em qual estado a variável de controle está no momento. A tolerância é um valor máximo, a qual o ângulo de inclinação da regressão linear não pode ultrapassar. Ao ser ultrapassado, se determina que a variável de controle está em processo de deterioração.

Conceitualmente, essa lógica de deterioração segue o fluxograma da Figura 19, onde 3 condicionais que ditam a classe da variável de processo de acordo com

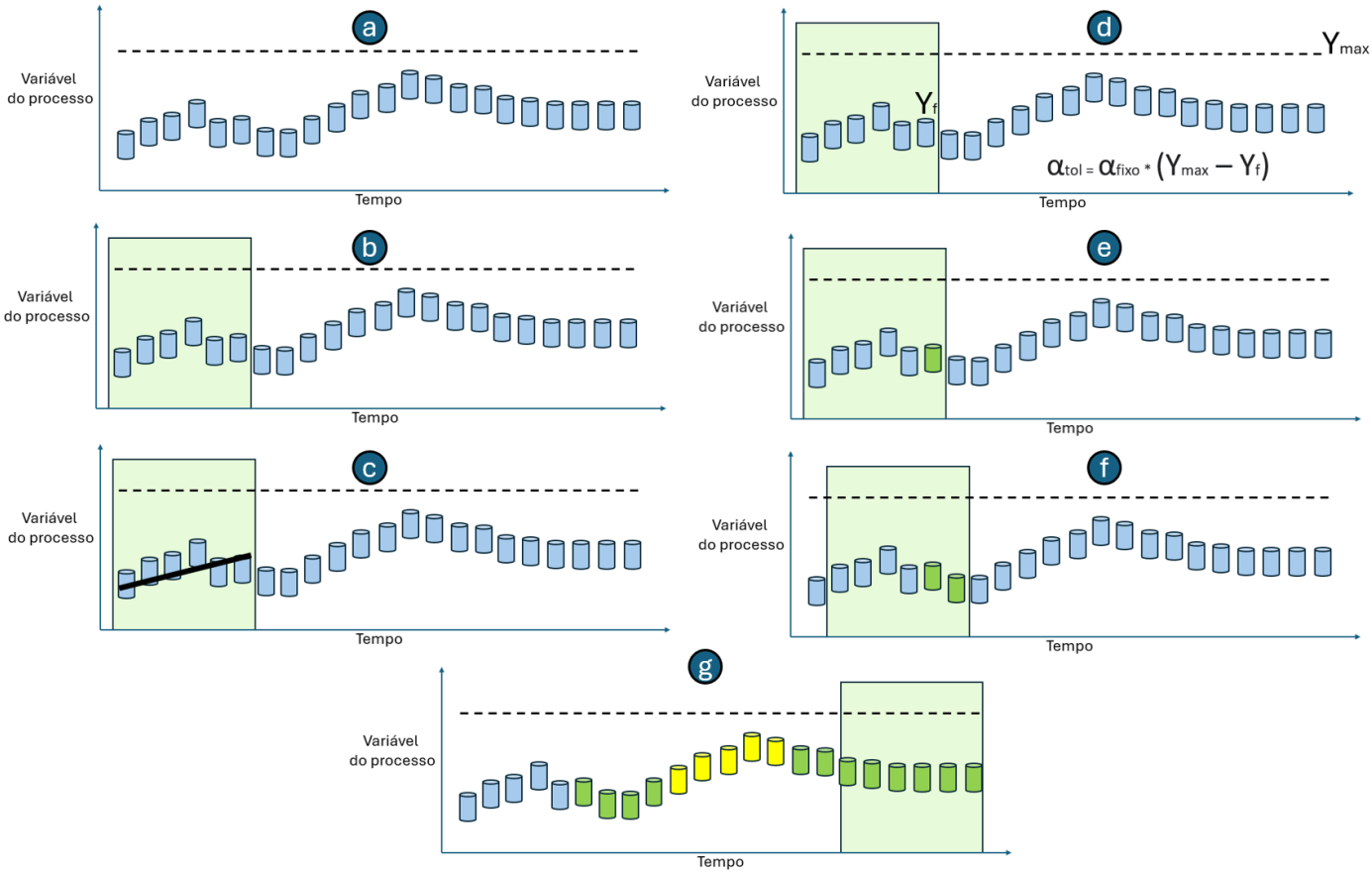
seu comportamento histórico recente. Na ilustração foi apresentado demais operações que precisam ser realizadas para preparar os dados à modelagem via regressão linear, como pré-processamento dos dados, a seleção das últimas 6 horas de amostras de dados (esse número é discutido em seções seguintes) e a normalização antes da modelagem via regressão linear. Além disso, as variáveis *Lim_sup* e *Lim_inf* referem-se aos limites máximos e mínimos que a variável de processo deve estar contida para não entrar num estado crítico para a operação.

FIGURA 19: FLUXOGRAMA LÓGICO DO ALGORITMO DETECTOR DE ALARMES.



Já na Figura 20, é apresentado este passo a passo de maneira mais visual, introduzindo os conceitos com dados fictícios de modo a emular o processo de funcionamento do algoritmo, desde a seleção da janela deslizante (passo a e b), cálculo do coeficiente de inclinação da reta (passo c), cálculo da tolerância dinâmica (passo d, mais detalhado na seção seguinte) até a determinação se aquele dado pertence a um estado normal (verde), deterioração (amarelo) e alarme (vermelho não contido na Figura).

FIGURA 20: ILUSTRAÇÃO DO ALGORITMO DE DETERMINAÇÃO DE ALARME.
A) DADOS MEDIDOS EM TEMPO REAL; B) SELEÇÃO DE JANELA DESLIZANTE; C) ESTIMATIVA DA INCLINAÇÃO DA RETA VIA REGRESSÃO LINEAR; D) CÁLCULO DA TOLERÂNCIA; E) CLASSIFICAÇÃO DO ESTADO DA VARIÁVEL; F) REPETIÇÃO DO MÉTODO NO INSTANTE N+1; G) REPETIÇÃO DO PROCESSO ATÉ O ESTADO ATUAL.



b) Utilizando a Regressão Linear

Os parâmetros da regressão linear são fundamentais no processo de detecção dos alarmes, uma vez que por meio da inclinação da reta gerada pelo modelo, é possível obter uma estimativa acerca da tendência da variável de processo. Em outras

palavras, também é possível estimar o tempo que levará para aquela variável alcançar os limites de controle pré-definidos.

Usando a própria equação da reta, onde se é conhecido o valor atual da variável, o valor dos limites inferior e superior, a inclinação da reta e considerando que o tempo inicial é igual a zero, pode-se considerar que o tempo estimado para a variável de processo chegar em um estado crítico é igual a Equação (9), onde $t - t_o$ representam o tempo estimado para alcançar o valor limite, y_o o valor atual, y_f o valor do limite e m a inclinação do regressão linear.

$$t - t_o = \frac{y_f - y_o}{m} \tag{9}$$

Assim, unindo a lógica de análise de tendência desenvolvida com o modelo de regressão linear com a estimativa de tempo de deterioração, tem-se uma parcela do algoritmo de alarmes concluída. No entanto, um novo conceito foi adicionado para este algoritmo com o intuito de torná-lo mais robusto: quanto mais próximo a variável de processo estiver de extrapolar os limites de controle, menor deve ser a tolerância ao valor de inclinação da regressão linear.

Por meio dessa preposição, foi criado um mecanismo matemático (por meio de funções auxiliares) para tornar a tolerância dinâmica, de modo que para valores mais próximos dos limites basta uma pequena inclinação para que seja acusado deterioração da variável de processo. Essa lógica é mais bem detalhada na seção a seguir.

c) Funções Auxiliares

Uma vez que a lógica do modelo de regressão linear foi criada, foi desenvolvido em seguida a lógica para tornar a tolerância da inclinação da reta mais sensível. Essa lógica foi planejada para que à medida que a variável de processo estivesse mais próxima dos limites críticos de operação, a tolerância a tendência fosse cada vez menor. E isso foi realizado por meio da experimentação de duas funções matemáticas. A primeira delas, foi uma função linear e a segunda uma função exponencial.

Considerando essas duas funções auxiliares estudadas, tem-se pelas Equações (10) e (11) a representação da função auxiliar linear, onde y_{sup} e y_{inf} são os limites inferior e superior da variável de processo, y_{pred} o valor previsto pela regressão linear m a inclinação da reta de regressão linear.

$$\theta_{din} = tang\left(\alpha_1\right)^{-1} \cdot \left|y_{sup} - y_{pred}\right|, \; se \; y_{pred} > \frac{y_{sup} + y_{inf}}{2} \tag{10}$$

$$\theta_{din} = tang\left(\alpha_1\right)^{-1} \cdot \left|y_{inf} - y_{pred}\right|, \; se \; y_{pred} < \frac{y_{sup} + y_{inf}}{2} \tag{11}$$

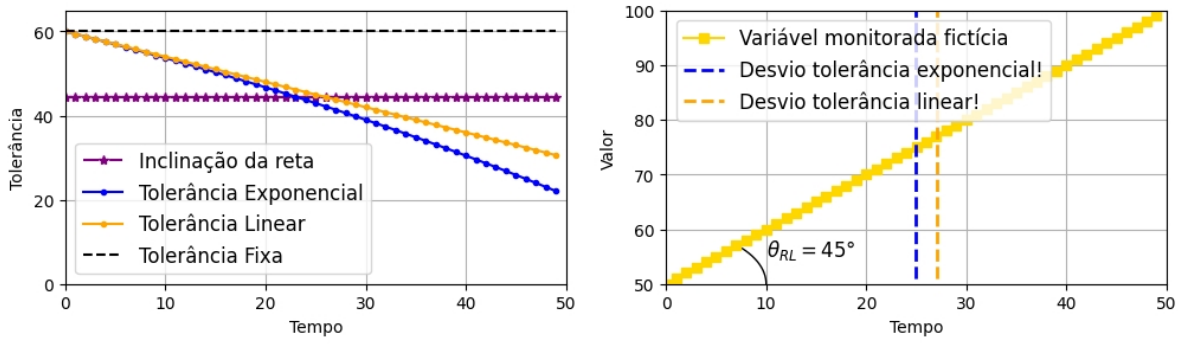
De maneira similar, tem-se para a função exponencial, onde agora a tolerância está em função de um exponencial, conforme as Equações (12) e (13).

$$\theta_{din} = tang\left(\alpha_1\right)^{-1} \cdot e^{-\left|y_{sup} - y_{pred}\right|}, \; se \; y_{pred} > \frac{y_{sup} + y_{inf}}{2} \tag{12}$$

$$\theta_{din} = tang\left(\alpha_1\right)^{-1} \cdot e^{-\left|y_{inf} - y_{pred}\right|}, \; se \; y_{pred} < \frac{y_{sup} + y_{inf}}{2} \tag{13}$$

Para exemplificar melhor, tomemos como referência os gráficos apresentados na Figura 21, considerando os limites inferior e superior iguais a 0 e 100, respectivamente. No gráfico da Figura 21b), tem-se uma reta, que emula uma variável qualquer tendendo a superar seu valor limite superior. Foi emulado, no Figura 21, o cálculo do ângulo dinâmico seguindo as três metodologias propostas: tolerância fixa, linear e exponencial.

FIGURA 21: COMPARAÇÃO ENTRE TOLERÂNCIAS DINÂMICAS.

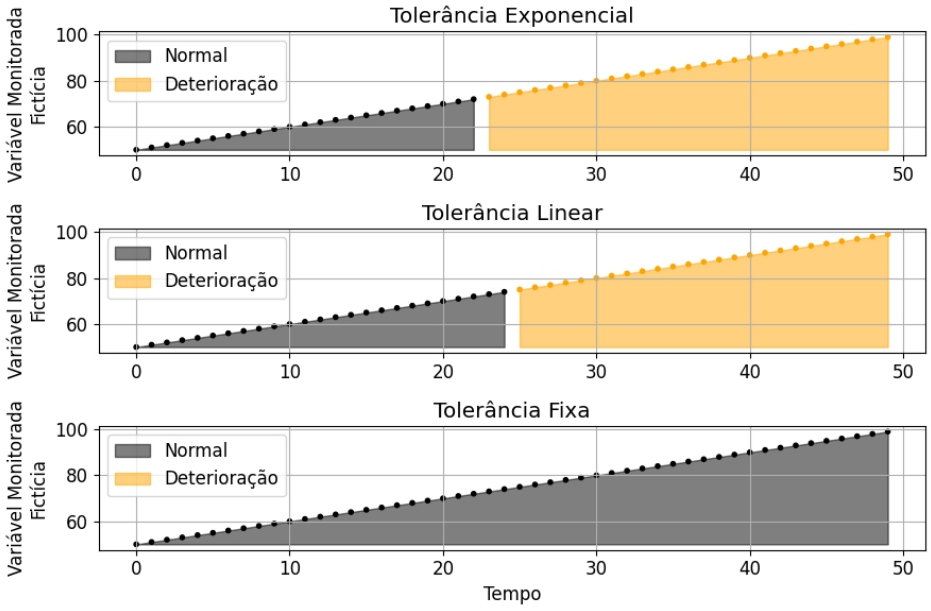


a) b)

É possível notar que a tolerância dinâmica vai diminuindo à medida que a variável do processo atinge valores mais próximos de seus limites, como pode ser observado na Figura 21a). Nota-se que, apesar da inclinação da reta calculada pela regressão linear permanecer sempre constante e tendendo a ultrapassar o limite superior, a utilização de uma tolerância fixa não acusou em nenhum momento a deterioração. Por outro lado, pelas metodologias da tolerância exponencial e linear, observa-se que ambas apontaram deterioração da variável de processo, tendo a exponencial apontado mais cedo. Isto mostra a importância destas funções matemáticas auxiliares para que a detecção de alarmes fique mais sensível à medida que a variável de processo se aproxime de seus limites.

Já o gatilho de acionamento destas 3 metodologias pode ser observado na Figura 22, a qual tem-se a comparação entre as 3 abordagens. Por esta forma de analisar, fica evidente que a tolerância fixa possui um defeito a qual as tolerâncias dinâmicas conseguem suplantar.

FIGURA 22: GATILHO DE ACIONAMENTO DAS TOLERÂNCIAS DINÂMICAS.



Essas duas lógicas de tolerância, serão denominadas como tolerâncias dinâmicas, justamente pela variação de acordo com o valor atual da variável de controle.

4.2.2 Classificação dos alarmes:

Uma vez que se tem uma maneira de predizer o estado da variável de processo, deseja-se saber qual dos subsistemas do processo de desmineralização da fábrica está provocando o seu estado de deterioração/perigo, com o objetivo de facilitar uma tomada de decisão por parte dos operadores.

Nesse sentido, investiu-se em algoritmos de classificação que fossem facilmente interpretáveis na maneira como eles fazem suas predições. Assim, o algoritmo que foi selecionado foi a árvore de decisão, considerando sua capacidade de realizar predições baseados em regras simples de serem interpretadas, no formato de *SE - SENÃO*.

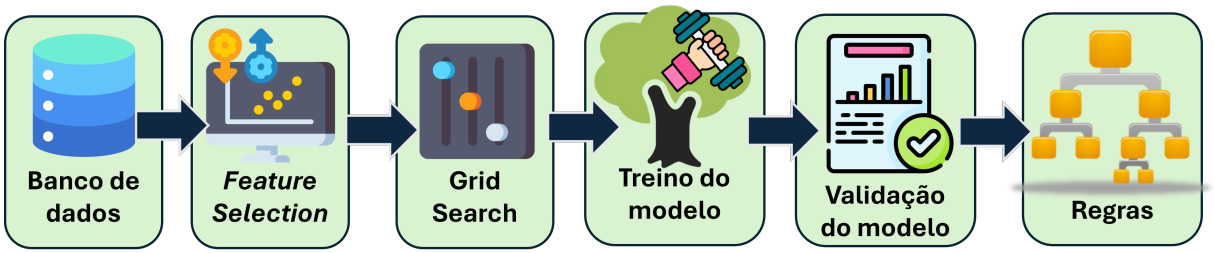
a) Abordagem planejada

Por meio dos dados históricos da operação, primeiramente foram criadas as classes utilizando o algoritmo de regressão linear, de modo que todos os instantes no tempo receberam um rótulo quanto o estado da variável de processo em questão, isto é, um rótulo de *operação normal*, *em deterioração* e *perigo*.

Uma vez que estes rótulos foram criados, foi possível realizar a atividade de aprendizagem supervisionada de classificação, levando em consideração a influência das variáveis de processos anteriores no estado atual da variável de processo em questão.

Assim, o procedimento completo da classificação seguiu-se da forma ilustrada na Figura 23. Aproveitando a capacidade da árvore de decisão em ranquear as variáveis mais influentes, o *feature importance* das variáveis foi realizado utilizando todo o volume de dados disponíveis para ser estabelecido as variáveis mais importantes; depois a árvore de decisão é novamente treinada, agora, apenas com as variáveis mais relacionadas e acoplada ao hiperparametrizador *Grid Search* e o desempenho da classificação é obtido. Por fim, o mapa de árvore é tratado para que fique simples de ser interpretado e consultado.

FIGURA 23: PROCESSO DA CLASSIFICAÇÃO COM ÁRVORE DE DECISÃO.



b) Utilizando Árvores de Decisão

Após a criação das categorias por meio do algoritmo de detecção de alarmes, os dados foram divididos em 2 parcelas para treinamento do algoritmo classificador: dados de treinamento (70%) e validação (30%). Com o objetivo de maximizar o desempenho foi realizado uma sintonia de seus hiperparâmetros por meio da técnica *Grid Search*, tendo sempre em vista a manutenção da interpretabilidade. Assim, para as árvores de decisão, os hiperparâmetros sintonizados seguiram os valores apresentados na Tabela 3.

TABELA 3: HIPERPARÂMETROS A SEREM SINTONIZADOS DA ÁRVORE DE DECISÃO.

Hiperparâmetros da Árvore de Decisão	*Grid Search*
Profundidade máxima	[2,3,4]
Mínimos exemplos por nó	[2, 10, 20, 200]
Mínimos exemplos por folha	[1,5,10,20]

4.2.3 Visualização dos resultados

O principal objetivo de algoritmos de extração de regras é que eles extraiam conhecimento interpretativo que forneçam suporte à tomada de decisões. Nesse sentido, essa seção ampliará a discussão acerca da visualização dos alarmes e sua interpretabilidade.

4.2.4 Alarmes

Para melhor identificação do estado de uma variável de processo, seguiu-se os princípios gerais da norma (ANSI/ISA18-2, 2016), que estabelece que o principal

objetivo de “um sistema de alarme é identificar operações não seguras ou anormais, avisos ou mal-funcionamento” e que possua “estados de alarmes que sejam fáceis de usar”. A ISO também recomenda que o alarme tenha cores e símbolos visíveis, não necessitando de cores para alarmes a qual a variável de processo esteja operando de maneira normal. Nesse contexto, o indicativo de alarme proposto segue conforme a Figura 24, onde se adotou o padrão de cores clássico de semáforos para as categorias de normal (verde), em deterioração (amarelo), perigo (vermelho).

FIGURA 24: PADRÃO DE ALARMES PROPOSTO.

4.3 RECURSOS E PACOTES PYTHON UTILIZADOS

Todos os algoritmos, análises e resultados desenvolvidos neste trabalho foram realizados utilizando a linguagem de programação Python, na versão 3.10.13, e seu ecossistema de pacotes auxiliares. Assim as soluções desenvolvidas fizeram uso dos seguintes recursos:

- **Pandas:** para leitura de arquivos com extensão CSV, criação de arquivos com extensão “parquet”, junção de dados e pré-tratamento via média móvel;
- **Numpy:** para operações vetorizados dos dados, execução de cálculos matemáticos;
- **Scikit-Learn:** para desenvolvimento do modelo de regressão linear para criação dos alarmes e árvores de decisão para modelo de classificação interpretativo; ferramentas de normalização de dados; métricas estatísticas;
- **Graphviz:** para visualização das regras criadas pelo modelo de árvore de decisão em formato de fluxograma;
- **Re:** para manipulação de textos (junto do Graphviz para visualização das regras com um visual agradável e compreensível);
- **Matplotlib:** para análises gráficas dos resultados.

Vale citar também que outras ferramentas também foram utilizadas para as tarefas de coleta de dados (Grafana) e a ferramenta SCADA da empresa para integração dos algoritmos (Ignition Automation).

O Python foi escolhido como ferramenta de trabalho devido primeiramente, a facilidade do autor de programar utilizando esta linguagem, mas principalmente devido a sua fácil integração com o *software* SCADA da empresa que é capaz de executar Python nativamente, através do compilador Jython. Isso significa que os scripts a serem desenvolvidos devem ser baseados em Python 2.x, em vez da versão atual do Python 3.x, impossibilitando o uso de bibliotecas populares como Scikit-Learn, Pandas e Numpy.

Dessa forma, uma das etapas a serem apresentadas nos resultados será a parte da conversão da solução desenvolvida para o compilador Jython e, posteriormente, a estética da tela apresentada.

5 RESULTADOS

5. RESULTADOS

Os resultados que serão apresentados seguirão a ordem da metodologia, começando pelo pré-processamento das variáveis conforme apresentado na Seção 4.1.1, para a comparação dos alarmes (baseado na Seção 4.2.1), a construção da árvore de decisão e seu desempenho para classificação (baseado na Seção 4.2.2) até a visualização do conhecimento gerado (baseado na Seção 4.2.3).

5.1 PRÉ-PROCESSAMENTO DOS DADOS

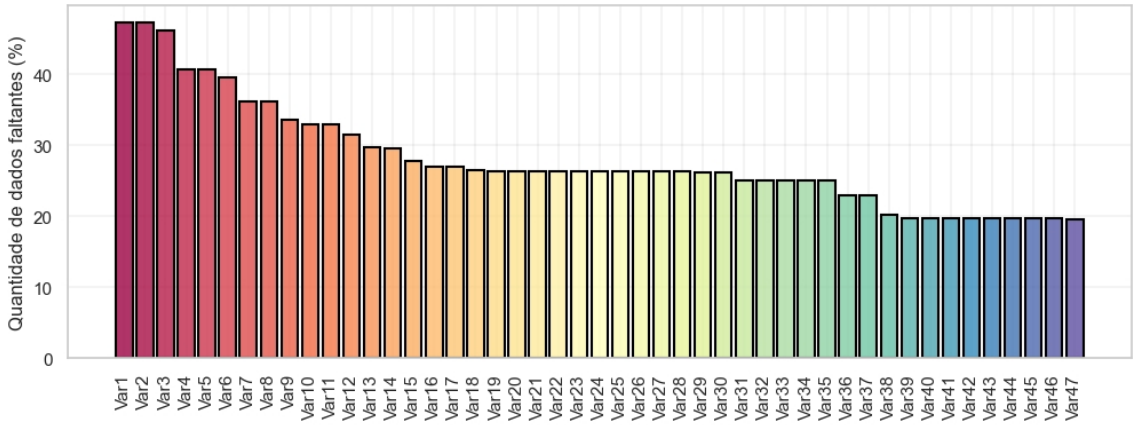
O começo do pré-processamento dos dados começou pela junção dos arquivos CSVs anteriormente citados em único parquet. Por meio de um diagnóstico inicial, foi realizado as seguintes constatações:

- Um total de 47 variáveis adquiridas relacionadas com as medições de vazão, pH, condutividade, dureza, pressão barométrica, pressão diferencial e nível de tanques;
- Dados separados a cada 1 segundo, mas com muita presença de NaN (representando cerca de 79,78% do total), devido a taxa de aquisição dos sensores serem a cada aproximadamente 10 segundos;
  Um total de cerca de 2.081.106 linhas de dados;
- Dados medidos durante o período de 21/02/2024 até 05/05/2024.

Assim, a primeira ação foi a aplicação da reamostragem dos dados para a cada um minuto, conforme justificado na metodologia, reduzindo o conjunto de dados para 108.000 amostras caindo o percentual de NaN para 28,01%. A eliminação de todas as linhas NaN implicaria numa redução de 108.000 para amostras, representando quase 70% dos dados que restaram após o tratamento. Para evitar uma eliminação tão grande de dados, foi verificado quais variáveis continham a maior quantidade de NaN para se retirar do conjunto total, partindo da suposição que estas ausências devem ser frequentes e, portanto, prejudicariam a intermitência de uma solução que dependa destes dados.

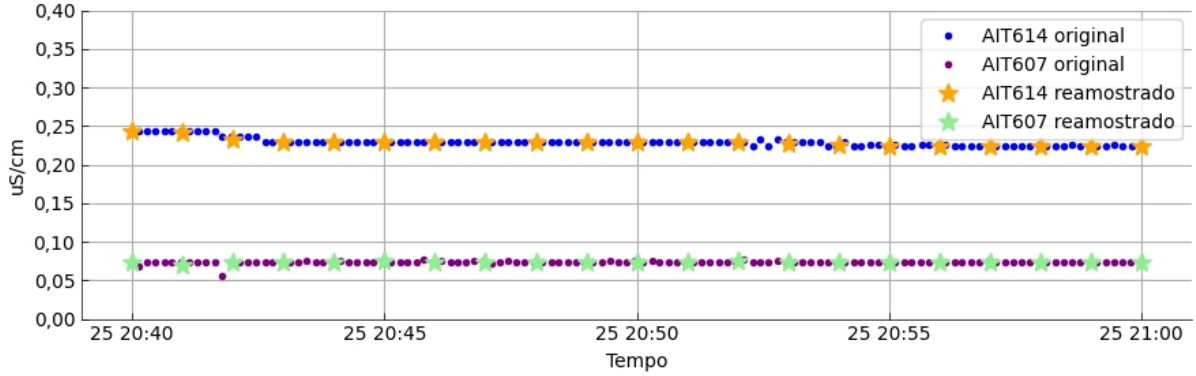
Assim calculando o percentual de NaN para cada uma das variáveis, decidiu-se eliminar de maneira empírica, variáveis com mais de 25% de dados faltantes, conforme a ilustrado no gráfico da Figura 25 representando 30 variáveis das 47 que foram pré-selecionadas para o procedimento de *Data Mining*. Além disso, mais duas variáveis foram eliminadas, pois apesar de terem medição, apresentaram desvio padrão igual à zero, sem qualquer variação durante o período de dados.

FIGURA 25: PERCENTUAL DE DADOS FALTANTES POR VARIÁVEL.



Eliminando todas as linhas com dados faltantes dentre as 15 variáveis que restaram, foi obtido um conjunto de dados de tamanho 78.567 amostras, ainda representando uma considerável quantidade de dados. Outra preocupação era quanto a qualidade dos dados do ponto de vista do seu perfil de distribuição após a aplicação da média móvel e a eliminação dos dados apontados. Só para nível de comparação na Figura 26, tem-se uma comparação do sensor AIT607 de condutividade no TLM e AIT614 de condutividade no ESPELHO, antes de depois da aplicação da média nos dados. É possível notar que a média móvel funcionou como esperado ajudando a sincronizar os dados sem perda de informação relevante.

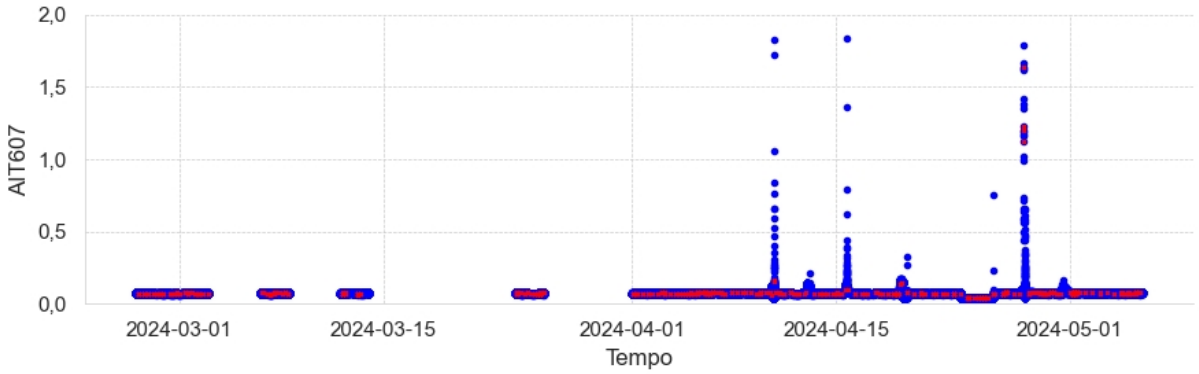
FIGURA 26: IMPACTO DA REAMOSTRAGEM NOS DADOS.



Para confirmar que as variáveis não perderam seu comportamento original, avaliou-se também suas características estatísticas como a média e o desvio padrão, os quais para o AIT614, por exemplo, as médias e desvio padrão pós reamostragem saíram de 0,3473 para 0,3473 e de 0,1236 para 0,1232 de respectivamente.

Para tratar *outliers* foi aplicada a técnica LOF em todo o conjunto de dados, de modo a encontrar dados que não se repetem como um padrão natural. Assim, na Figura 27, são apresentados os pontos que foram eliminados pelo algoritmo que foram considerados *outliers* ou avarias nas medições. Como é possível notar, os pontos marcados com o X vermelho, representam variações drásticas anormais no comportamento de uma variável, tendo sido rotulados com *outlier* pelo algoritmo. No total, cerca de 1% dos dados foram removidos, restando uma massa de 63.451 amostras de dados.

FIGURA 27: ELIMINAÇÃO DE OUTLIERS.

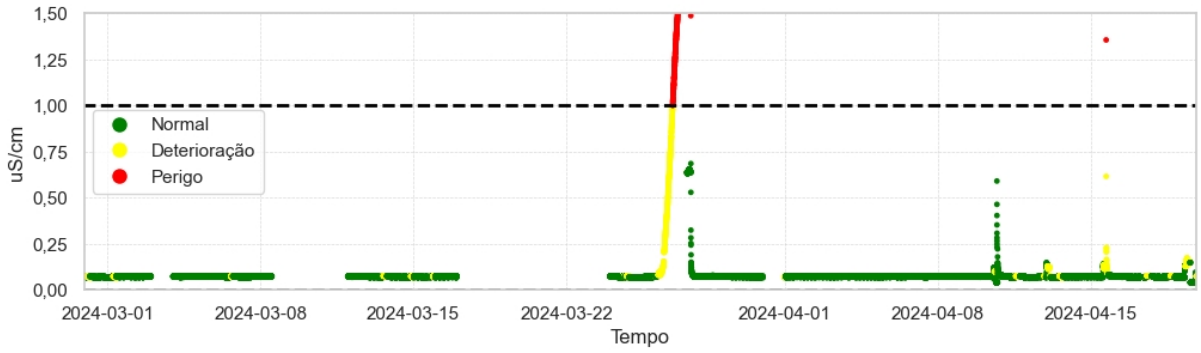


5.2 ALARME INTELIGENTE X TRADICIONAL

Após o tratamento dos dados, a construção do alarme inteligente da variável de processo foi avaliada comparando as funções matemáticas auxiliares. O processo de definição do ângulo de tolerância e o tamanho da janela temporal foi realizado de maneira empírica, acompanhando as visualizações gráficas que mais se encaixavam com a expectativa de detecção dos desvios pré-definidos.

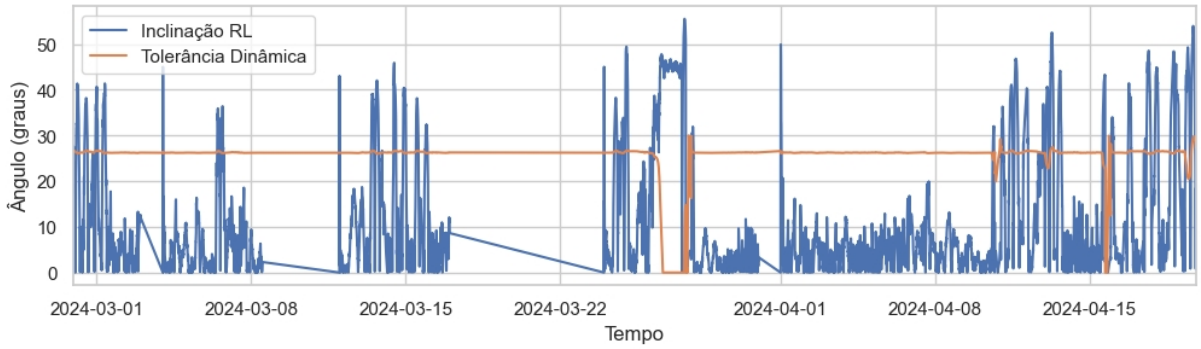
Assim, considerando as funções lineares e exponencial, para a variável de processo de condutividade AIT607, foi escolhido o método de função auxiliar linear com ângulo de tolerância igual à 30º e janela deslizante de 360 minutos como a mais apropriada para detecção de comportamentos anômalos. Pode-se notar, pela Figura 28, que esta abordagem foi capaz de evitar sinalização de desvios quando não fossem necessários, embora alguns momentos apresentassem falsa sinalização. Por outro lado, quando a tendência de desvio aconteceu entre os dias 22 de março até 1º de abril, o algoritmo cumpriu com seu propósito base de prever a deterioração da variável de processo, conforme era desejado.

FIGURA 28: DETECÇÃO DE DESVIOS NO ESTUDO DE CASO.



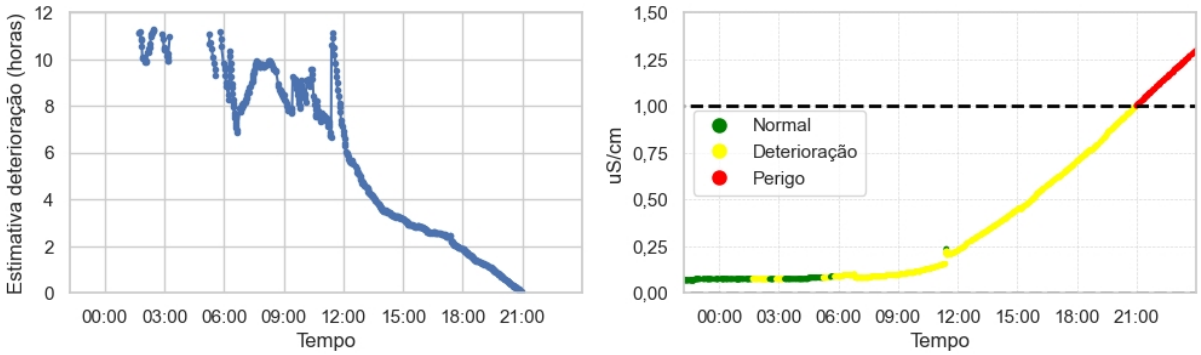
Na Figura 29, nota-se a comparação entre os ângulos de inclinação da regressão linear e a tolerância dinâmico, baseados na Equações (2), (6) e (7), onde sempre que a inclinação supera a tolerância, é acionado uma sinalização de deterioração. É possível notar que no mesmo período que o algoritmo acusou deterioração na Figura 28, a tolerância dinâmica na Figura 29 variou para um menor valor.

FIGURA 29: VARIAÇÃO DO TOLERÂNCIA DINÂMICA AO LONGO DO TEMPO NO ESTUDO DE CASO.



Tomando como referência esta mesma janela temporal da variável de processo, foi avaliado também o tempo estimado, baseado na Equação (5) pelo algoritmo para que a deterioração alcance um estado crítico para a operação. Nesse sentido, na Figura 30, é apresentado um caso da estimativa do algoritmo em relação ao momento que ocorreu na deterioração.

FIGURA 30: ESTIMATIVA DE DETERIORAÇÃO NO ESTUDO DE CASO.



a) b)

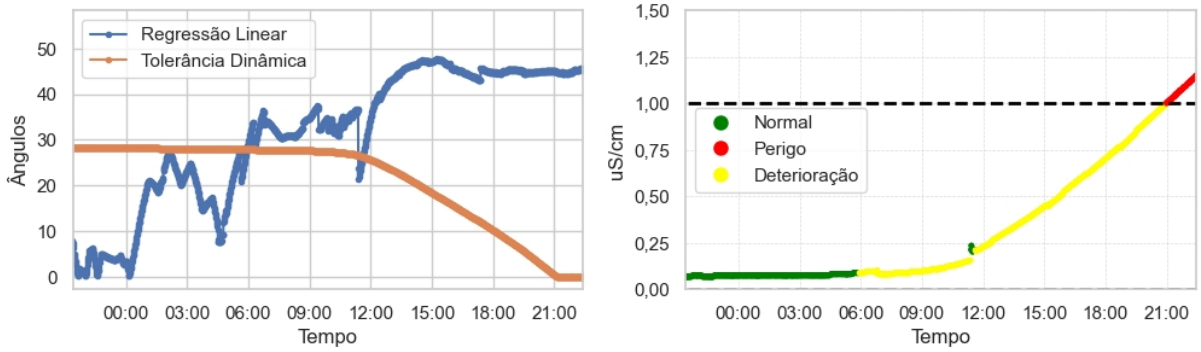
Pode-se notar que o algoritmo fez estimativas de tempo que variaram de acordo com a dinâmica da variável de processo, ficando mais precisa à medida que a deterioração era consumada. Por exemplo, às 12 horas, o algoritmo estimou que em aproximadamente 7,5 horas a variável do processo atingiria seu estado crítico, que acabou de fato acontecendo por volta das 21 horas. A estimativa de tempo seguiu a matemática apresentada na Equação (9), conforme pode ser visto exemplificado na Figura X, onde 0,225 era o valor atual da variável de controle, 1 o limite superior, 35 graus o valor do ângulo estimado pela RL e o 360, representando as 6 horas de janela deslizante utilizado.

FIGURA 31: CÁLCULO DO TEMPO ESTIMADO PARA AS 12 HORAS DA JANELA ANALISADA.

$$t - t_o = \frac{(y_f - y_o)}{m} \Rightarrow t - 0 = \frac{(1 - 0.225)}{35,09° \cdot \pi/180} \cdot 360 \Rightarrow t \simeq 7,59 horas$$

Clarificando, novamente, o comportamento do algoritmo proposto, na Figura 32, é apresentado para esse mesmo recorte temporal a variação do ângulo de inclinação da regressão linear em relação à tolerância dinâmica. Este é um ótimo exemplo para ilustrar o funcionamento do algoritmo. Nota-se que a tolerância dinâmica diminui à medida em que o valor da variável de processo começa a tender a seu limite crítico, até próximo das 21:00 a qual há consumação da deterioração. Pode-se observar também que a acusação de alarme ocorre apenas após a inclinação da regressão linear ultrapassa a tolerância dinâmica.

FIGURA 32: RECORTE DO FUNCIONAMENTO DO ALGORITMO EM UMA SITUAÇÃO DE DETERIORAÇÃO.



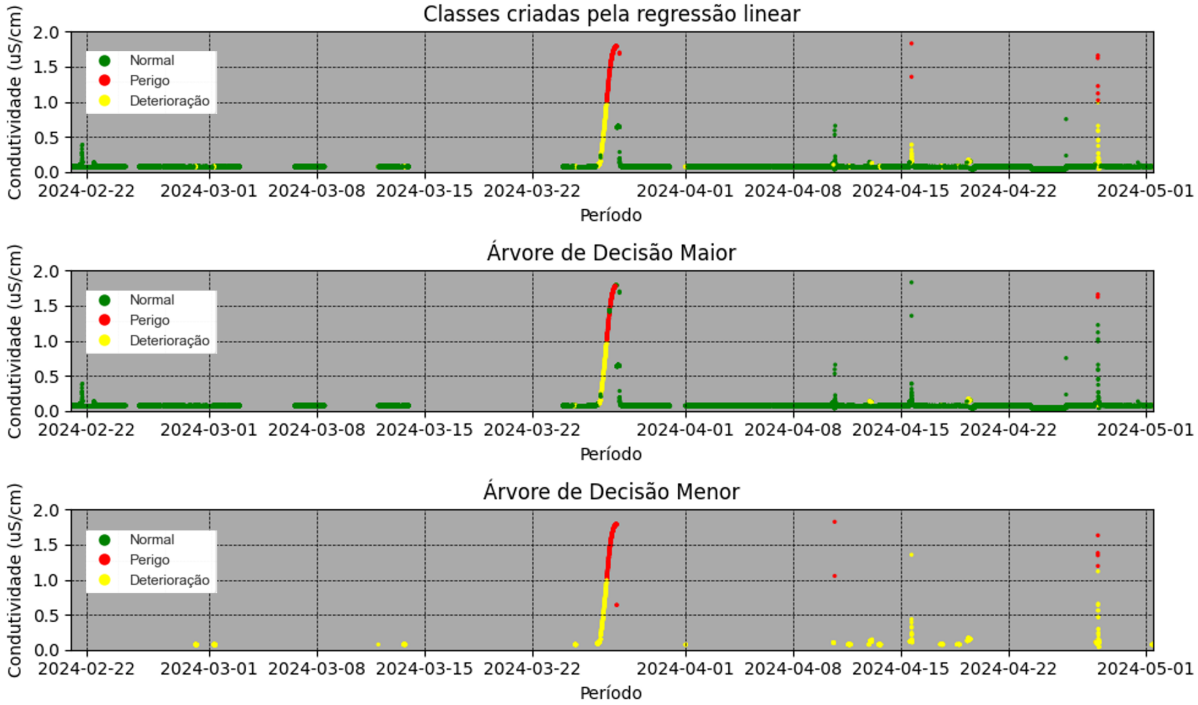
a) b)

5.3 DESEMPENHO DA CLASSIFICAÇÃO

Uma vez que todo o conjunto de dados recebeu seus respectivos rótulos de *status* da variável de processo com a regressão linear, foi iniciado o processo de treinamento do modelo de árvore de decisão. Para isso, este trabalho se propõe a discutir os resultados para duas abordagens de classificação: a primeira que aprende o comportamento de todos os dados rotulados e o segundo que foca em apenas aprender os rótulos relacionados aos *status* de *em deterioração* e *perigo.*

Para ilustrar melhor essa diferença entre a rotulação de alarme inteligente criada pela regressão linear com o aprendizado das duas abordagens de modelos de árvore de decisão, na Figura 33 é apresentado 3 gráficos a qual se pode comparar o quão próximo a previsão das árvores estão do valor real rotulado pela regressão linear. Isto mostra o quão aderente estão os modelos para as situações de alarmes acusadas pela regressão linear. Nos parágrafos a seguir são mais bem detalhados como os modelos chegaram a este desempenho.

FIGURA 33: JANELA TEMPORAL DE PREVISÃO DAS ÁRVORES DE DECISÃO.



De acordo com a rotulação realizada pelo alarme inteligente baseado em regressão linear, a distribuição dos rótulos de *status* da variável de processo ficou com um percentual de 92,98% para *normal*, 5,19% para *em deterioração* e 1,26% para *perigo.* Isso, em termos absolutos representam 73.501, 4071 e 995 amostras respectivamente para cada um dos *status.*

Assim, separando de maneira aleatória, o conjunto de dados de treinamento e teste, foi desenvolvido o primeiro modelo de árvore de decisão que, após conclusão de seu processo de hiperparametrização *Grid Search*, teve como melhor combinação de hiperparâmetros os apresentados na Tabela 4.

TABELA 4: HIPERPARÂMETROS ÓTIMOS PARA ÁRVORE DE DECISÃO MAIOR.

Profundidade máxima	Mínimo de exemplos por folhas	Mínimo de exemplo para nós
4	10	10

As variáveis que obtiveram maior pontuação no *rank* da árvore de decisão foram as mostradas na Tabela 5. Pode-se notar que as variáveis mais influentes, foram variáveis mais próximos do processo de espelhamento mesmo, com menor influência de variáveis de processos anteriores. Interessante notar que as variáveis de qualidade de água que mais estão relacionadas com o estado da condutividade da água para o espelho foram o pH do espelho e a dureza da água no segundo passo da OR.

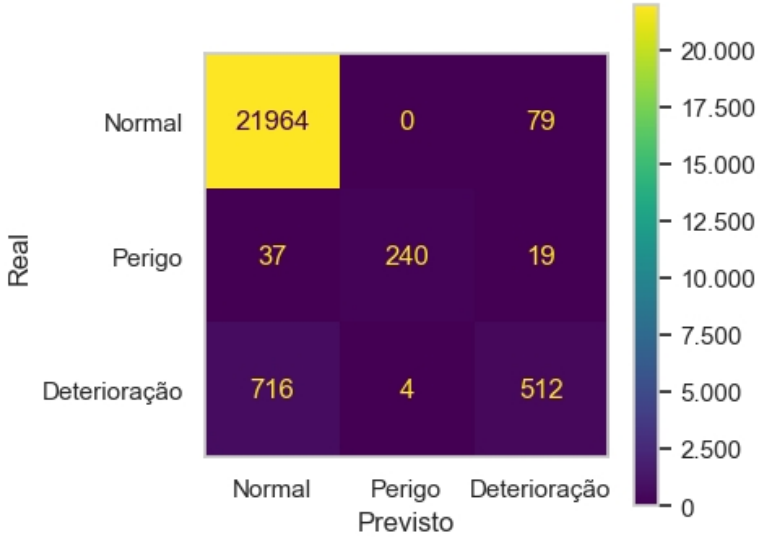
TABELA 5: VARIÁVEIS MAIS IMPORTANTES PELA ÁRVORE DE DECISÃO MAIOR.

Variável de entrada	Importância
LIT606	0,431725
AIT604	0,178389
AIT610	0,141738
FIT606	0,072324
AIT605	0,050886
PIT613	0,038137
LIT601	0,037105
PIT605	0,031227
PIT607	0,009947
AIT603	0,008521
AIT602	0,000000
LIT607	0,000000
PIT611	0,000000
AIT616	0,000000

Avaliando este modelo segundo as métricas de desempenho, foi obtido uma acurácia de 96,37%, ao passo que foi obtida uma acurácia balanceada com um desempenho menor na proporção de 74,09%. Isto se dá, pois o modelo de árvore de decisão teve desempenho superior para classificar operações *normais* do que as demais operações, que pode ser devido a proporção de cada um dos modos de operações nos dados. Para ilustrar melhor a relação de acertos e erros do modelo foi também produzido a matriz de confusão para os 3 rótulos e suas respectivas acurácias

isoladas na Figura 34, a qual pode-se notar um desempenho bem maior do *status* de *normal*, em relação as demais. Destaca-se também, uma dificuldade em distinguir operação *normal* de *em deterioração*.

FIGURA 34: MATRIZ DE CONFUSÃO PARA OS TRÊS STATUS.



Tendo em vista um desempenho aquém em classificar a variável de processo no estado de *em deterioração*, repetiu-se o procedimento de treinamento do modelo de classificação, agora, para aprendizagem apenas dos *status* de *em deterioração* e *perigo*. Assim, foi desenvolvido uma nova árvore de classificação, utilizando os mesmos dados do primeiro treinamento, mas retirando os dados com *status* de operação *normal.* Neste caso, os hiperparâmetros ótimos apontado pelo hiperparametrizador foram diferentes, conforme pode ser visto na Tabela 6.

TABELA 6: HIPERPARÂMETROS ÓTIMOS PARA ÁRVORE DE DECISÃO MENOR.

Profundidade máxima	Mínimo de exemplos por folhas	Mínimo de exemplo por nós
3	20	2

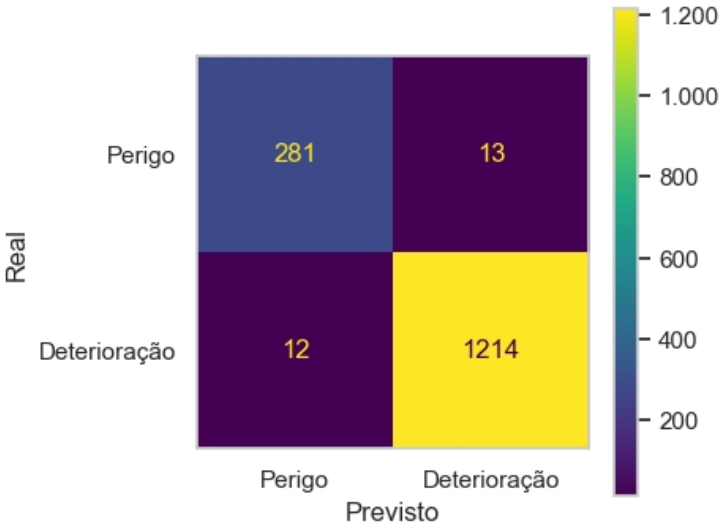
Neste cenário, foi observado que as variáveis de maior influência na variável de processo mudaram parcialmente em relação a primeira abordagem, conforme pode ser visto na Tabela 7. Isto deve se dar devido a mudança geral da distribuição dos dados considerando apenas conjuntos com operações anômalas. Esse fato é bem positivo, tendo em vista o novo conjunto de conhecimento que este novo modelo pôde fornecer.

TABELA 7: VARIÁVEIS MAIS IMPORTANTES PELA ÁRVORE DE DECISÃO MENOR.

Variável de entrada	Importância
AIT610	0,801837
AIT602	0,147195
AIT616	0,029515
PIT611	0,021207
LIT607	0,000246
LIT601	0,000000
AIT603	0,000000
PIT607	0,000000
PIT605	0,000000
FIT606	0,000000
AIT604	0,000000
LIT606	0,000000
AIT605	0,000000
PIT613	0,000000

Mais uma vez, a matriz de confusão foi produzida para ter um panorama do desempenho do modelo em relação a sua tarefa de classificação, conforme pode ser analisada na Figura 35. Nesse caso, pode ser visto que o modelo melhorou sua capacidade de prever os *status* propostos em comparação com a primeira abordagem. Isso mostra que esta abordagem é mais eficiente, quando se trata apenas de criar informações que são prejudiciais na operação.

FIGURA 35: MATRIZ DE CONFUSÃO PARA 2 STATUS.



Os resultados dos conhecimentos gerados por ambas de árvores de decisão produzidas são detalhados na Seção seguinte.

5.4 INTERPRETAÇÃO VISUAL DOS RESULTADOS

Os dois modelos de árvores de decisão obtiveram os seguintes conhecimentos gerados, conforme as ilustrações das Figura 36 e Figura 37, que representam as árvores de decisão maior e menor respectivamente. É possível notar que a interpretabilidade da árvore menor está mais simples de ser entendida, considerando ter uma menos nós e profundidade. No entanto, a árvore maior proporciona a visualização mais completa, conforme pode ser visto na Figura 36.

FIGURA 36: EXTRAÇÃO DE CONHECIMENTO DA ÁRVORE MAIOR.

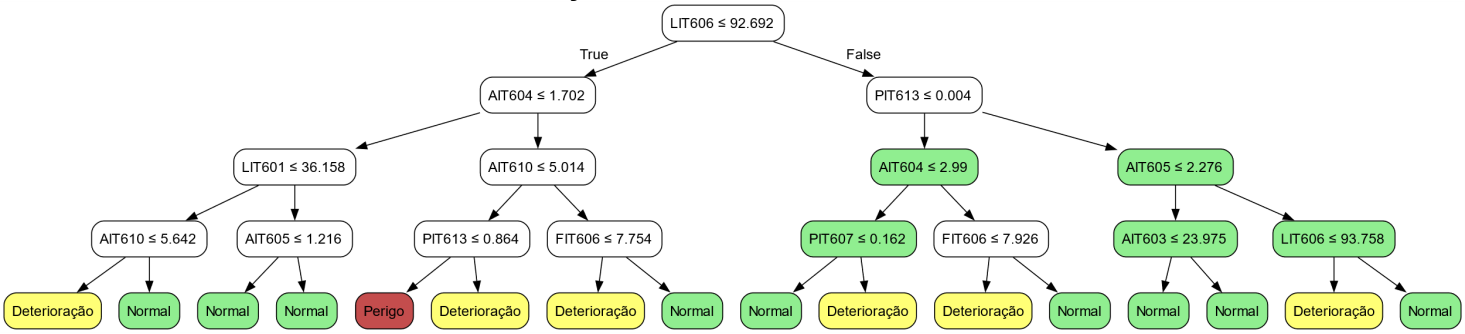
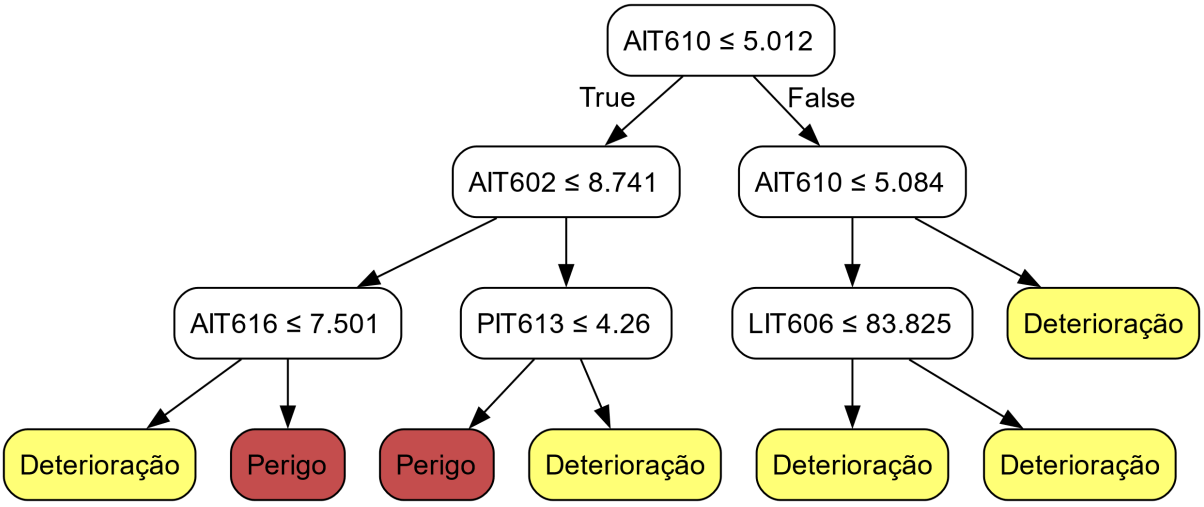


FIGURA 37: EXTRAÇÃO DE CONHECIMENTO DA ÁRVORE MENOR.



Além disso, nota-se que a árvore maior se mostra mais interessante quanto ao conhecimento, devido a justamente incluir a noção de operações normais no meio de

seu aprendizado, pois pode-se compreender mais facilmente o que se espera do comportamento das variáveis mais relacionadas dado um certo *status* operacional da variável monitada. No entanto, de maneira mais precisa, a segunda árvore é mais sensível para notar a influência de processos anteriores no *status* atual.

Assim, tomando ainda a Figura 37, pode-se fazer sugestões mais práticas aos operadores sobre onde intervir para fazer ações corretivas na operação. Por exemplo, caso o pH na OR2 (representado pelo sensor AIT616) estiver com valor maior que 7,5 isso significa que a água está em um estado crítico e correções da OR2 são criticamente necessárias, como a substituição das resinas da osmose. Por outro lado, tem-se também que se a pressão medida pós OR2 (representado pelo sensor PIT613) estiver menor que 4,26 bar há um indicativo de que, novamente, a OR2 precisa de manutenção, considerando uma baixa pressão que pode ser motivada pela perda na eficiência da resina que começa a ficar mais entupida, diminuindo a pressão medida.

5.5 IMPLEMENTAÇÃO DA SOLUÇÃO EM SCADA

Após todos os resultados coletados, foi a vez de converter toda a solução lógica desenvolvida para o software SCADA da fábrica. O Ignition oferece um conjunto de funções que permitem interação com o banco de dados do servidor, possibilitando que as operações matemáticas necessárias pelos modelos de *machine learning* sejam viabilizadas, desde que as devidas compatibilidades sejam satisfeitas.

Assim, foram desenvolvidos dois scripts para a conversão das lógicas propostas. O primeiro relacionado ao diagnóstico de deterioração a partir da regressão linear e o segundo a determinação da causalidade de deterioração a partir da árvore de decisão.

Dessa forma, na Figura 38 é apresentado o script adaptado para a regressão linear. É possível notar que o processo de normalizar e a criação do modelo de regressão teve que ser feito todo de maneira manual, substituindo a utilização da biblioteca Scikit-Learn. Percebe-se também, que o treinamento da regressão linear foi facilitado utilizando as Equações 4 e 5.

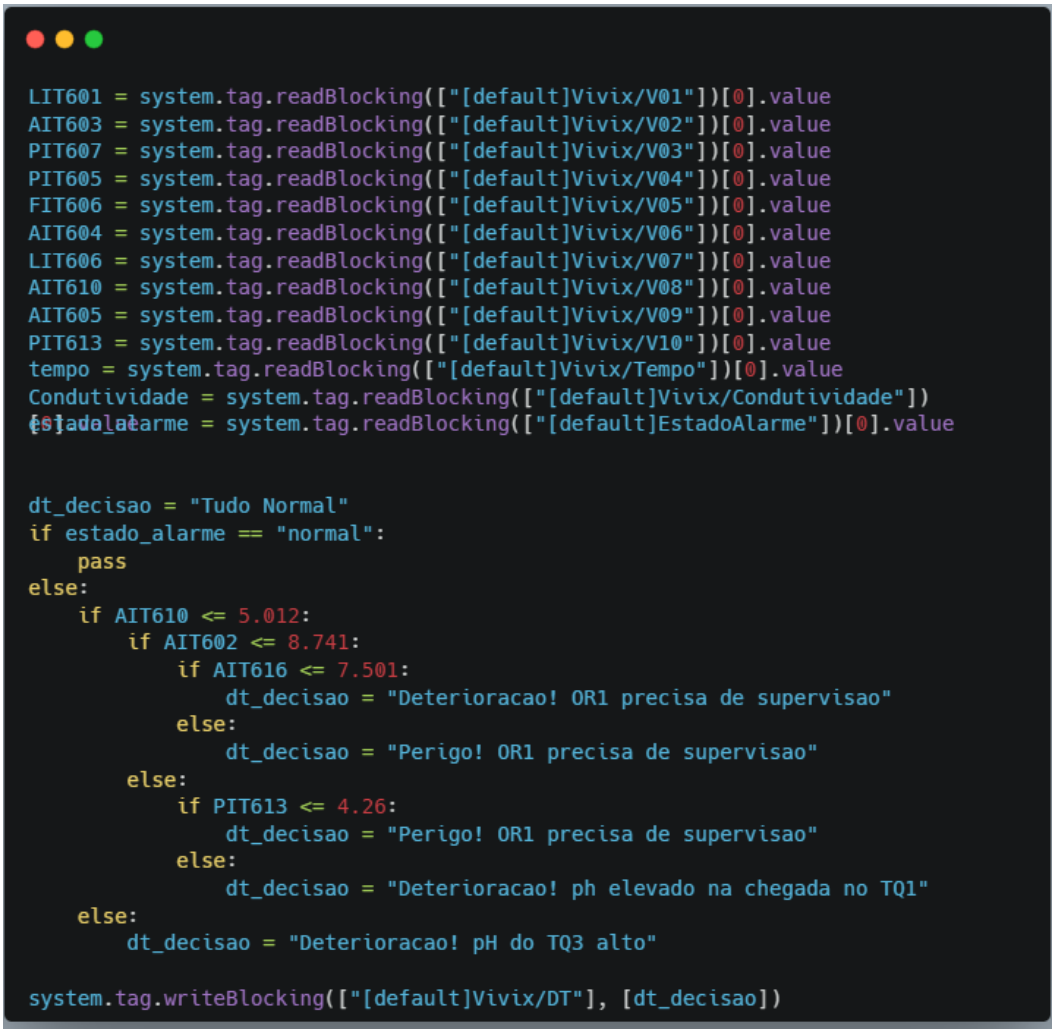
FIGURA 38: CONVERSÃO DO SCRIPT DE REGRESSÃO LINEAR PARA O AMBIENTE DO IGNITION.

```
import math

path_x = "[default]Vivix/Condutividade"
path_y = "[default]Vivix/Tempo"

# Constantes de acordo com a variável analisada
limsup_y = 1
liminf_y = 0
incl = 30
horas = 6

endTime = system.date.now()
startTime = system.date.addMinutes(endTime, -horas)

# Coletar dados da saida
dataset = system.tag.queryTagHistory([path_x], startDate=startTime, endDate=endTime)
rows1 = system.dataset.toPyDataSet(dataset)

dataset = system.tag.queryTagHistory([path_y], startDate=startTime, endDate=endTime)
rows2 = system.dataset.toPyDataSet(dataset)

index_y = [system.date.parse(row[1], 'yyyy-MM-dd hh:mm:ss') for row in rows2] # Tempo em milisegundos
desde 1970
valor_y = [row[1] for row in rows1] # Valores

diff_temp = [(index_y[i].time -index_y[0].time)/(60*1000) for i in range(len(index_y))]

# Normalizacao
ScalerFunc = lambda z: (z - min(z))/(max(z) - min(z))
x_norm = [(diff_temp[i] - min(diff_temp))/(max(diff_temp) - min(diff_temp)) for i in
range(len(diff_temp))]
y_norm = [(valor_y[i] - min(valor_y))/(max(valor_y) - min(valor_y)) for i in range(len(valor_y))]

# Coeficientes da RL
x_mean = sum(x_norm)/len(x_norm)
y_mean = sum(y_norm)/len(y_norm)
numerador = sum([(x_norm[i]-x_mean)*(y_norm[i]-y_mean) for i in range(len(valor_y))])
denominador = sum([(x-x_mean)**2 for x in x_norm])
a1 = numerador/denominador
angulo = math.atan(a1)*360/(2*3.14)
a0 = y_mean - a1*x_mean

# Calculo da tolerancia
y_pred_norm = a0 + a1*x_norm[-1]
y_pred = y_pred_norm*(max(valor_y) - min(valor_y)) + min(valor_y)
k = 1-abs(y_pred_norm)
tol = incl*k

# Calculo da tolerancia
estado_alarme = "normal"
if angulo > tol:
    estado_alarme="deterioracao"

if valor_y[-1] > limsup_y or valor_y[-1] < liminf_:
    estado_alarme = "perigo"

# Tempo estimado para deterioração
if estado_alarme == "deterioracao":
    tempo = (limsup_y - valor_y[-1])/a1*diff_temp[-1]
else:
    tempo = 0

path_estado = "EstadoAlarme.value"
path_tempo = "EstimativaDeterioracao.value"

system.tag.writeBlocking([path_estado], [estado_alarme])
system.tag.writeBlocking([path_tempo], [tempo])
```

Por outro lado, na Figura 39 é apresentado o script para a criação do modelo de árvores de decisão. Nesse caso, foi necessário apenas o carregamento das

variáveis do banco e o desenvolvimento de lógicas de if-else próprios da lógica criada pelo modelo durante a fase de treinamento. Foi escolhido modelo de árvore de decisão menor, devido a seu melhor desempenho em classificar os estados de deterioração e perigo.

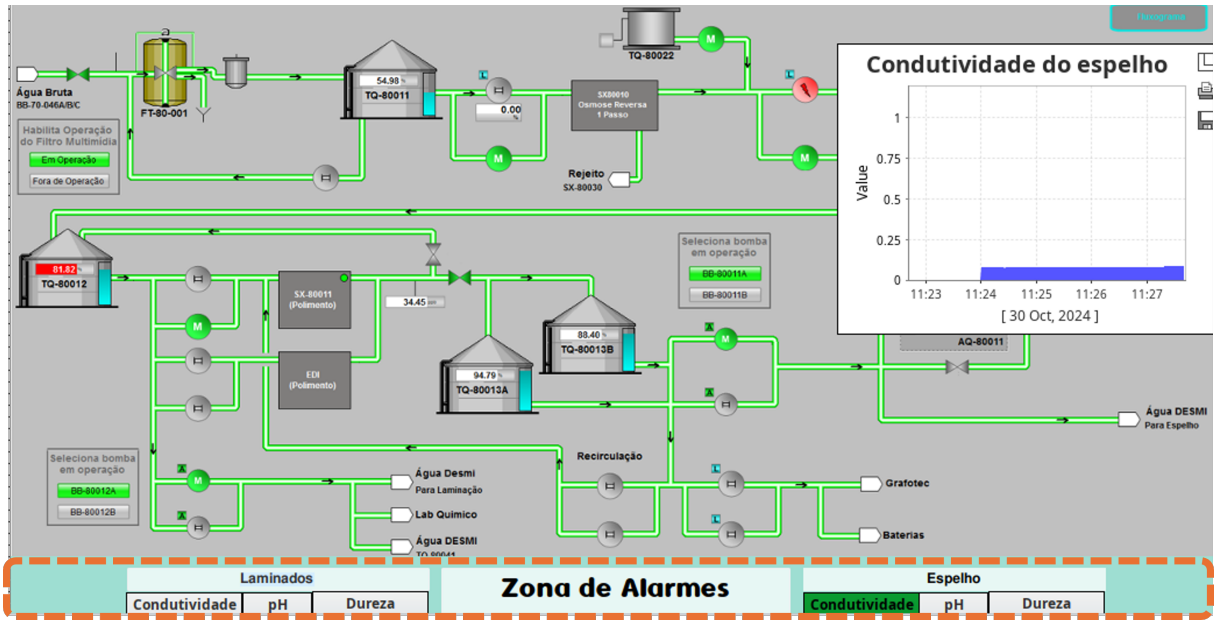
FIGURA 39: CONVERSÃO DO SCRIPT DE PREVISÃO DA ÁRVORE DE DECISÃO PARA O AMBIENTE DO IGNITION.



```
LIT601 = system.tag.readBlocking(["[default]Vivix/V01"])[0].value
AIT603 = system.tag.readBlocking(["[default]Vivix/V02"])[0].value
PIT607 = system.tag.readBlocking(["[default]Vivix/V03"])[0].value
PIT605 = system.tag.readBlocking(["[default]Vivix/V04"])[0].value
FIT606 = system.tag.readBlocking(["[default]Vivix/V05"])[0].value
AIT604 = system.tag.readBlocking(["[default]Vivix/V06"])[0].value
LIT606 = system.tag.readBlocking(["[default]Vivix/V07"])[0].value
AIT610 = system.tag.readBlocking(["[default]Vivix/V08"])[0].value
AIT605 = system.tag.readBlocking(["[default]Vivix/V09"])[0].value
PIT613 = system.tag.readBlocking(["[default]Vivix/V10"])[0].value
tempo = system.tag.readBlocking(["[default]Vivix/Tempo"])[0].value
Condutividade = system.tag.readBlocking(["[default]Vivix/Condutividade"])
estado_alarme = system.tag.readBlocking(["[default]EstadoAlarme"])[0].value


dt_decisao = "Tudo Normal"
if estado_alarme == "normal":
    pass
else:
    if AIT610 <= 5.012:
        if AIT602 <= 8.741:
            if AIT616 <= 7.501:
                dt_decisao = "Deterioracao! OR1 precisa de supervisao"
            else:
                dt_decisao = "Perigo! OR1 precisa de supervisao"
        else:
            if PIT613 <= 4.26:
                dt_decisao = "Perigo! OR1 precisa de supervisao"
            else:
                dt_decisao = "Deterioracao! ph elevado na chegada no TQ1"
    else:
        dt_decisao = "Deterioracao! pH do TQ3 alto"

system.tag.writeBlocking(["[default]Vivix/DT"], [dt_decisao])
```

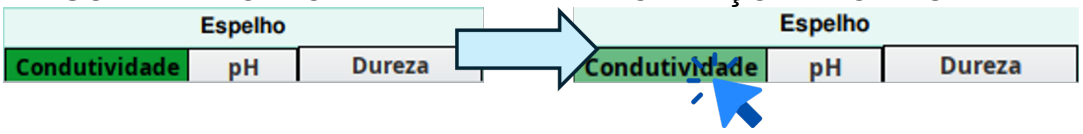
Para a proposição da tela SCADA, foi utilizado como background a própria interface utilizada como IHM da fábrica para o sistema de desmineralização, adicionando apenas um rodapé com uma seção chamada “Zona de Alarmes”. Nessa região teria botões coloridos, de acordo com a situação das variáveis de controle monitoradas, que indicam a qualidade da água para o espelho e os laminados, as etapas produtivas de interesse. Na Figura 40 pode ser visto o rodapé desenvolvido.

FIGURA 40: RODAPÉ ZONA DE ALARMES DESENVOLVIDO.



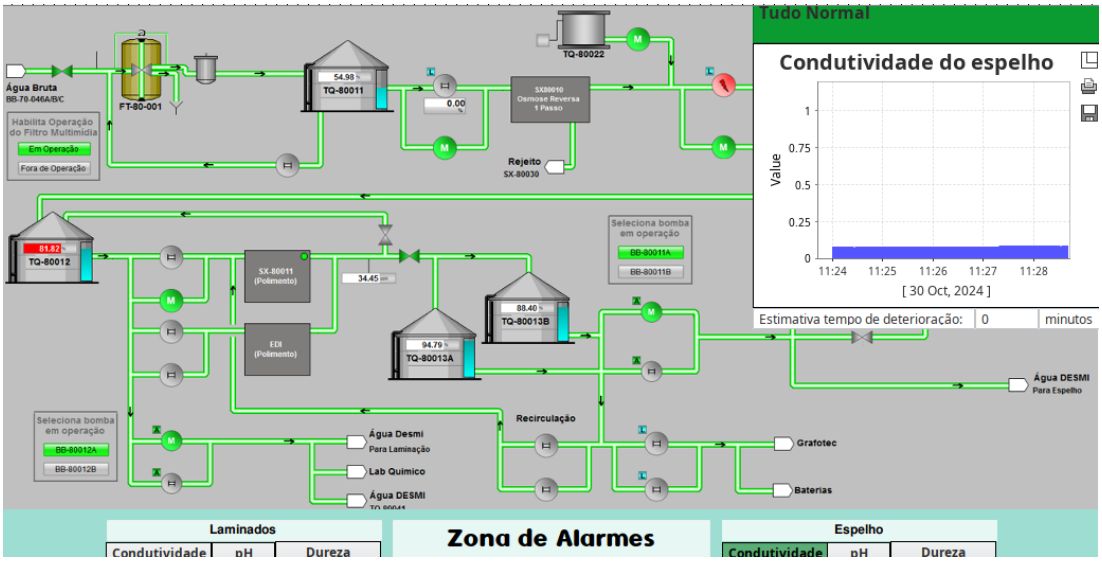
O gráfico incluso na Figura 40 representa uma simulação do algoritmo funcionando diante do mesmo recorte temporal apresentado nos resultados da Figura 30. Mais adiante iremos explorá-lo um pouco mais. Pode-se notar também o que há um retângulo verde na zona de alarmes em espelho, representando um botão. Este botão pode ser clicado conforme pode ser visto na Figura 41, liberando mais informações acerca da ocorrência daquele alarme, que são as informações de causalidade e tempo estimado para que a variável alcance um nível crítico de perigo.

FIGURA 41: BOTÃO PARA LIBERAR INFORMAÇÕES DOS SCRIPTS.



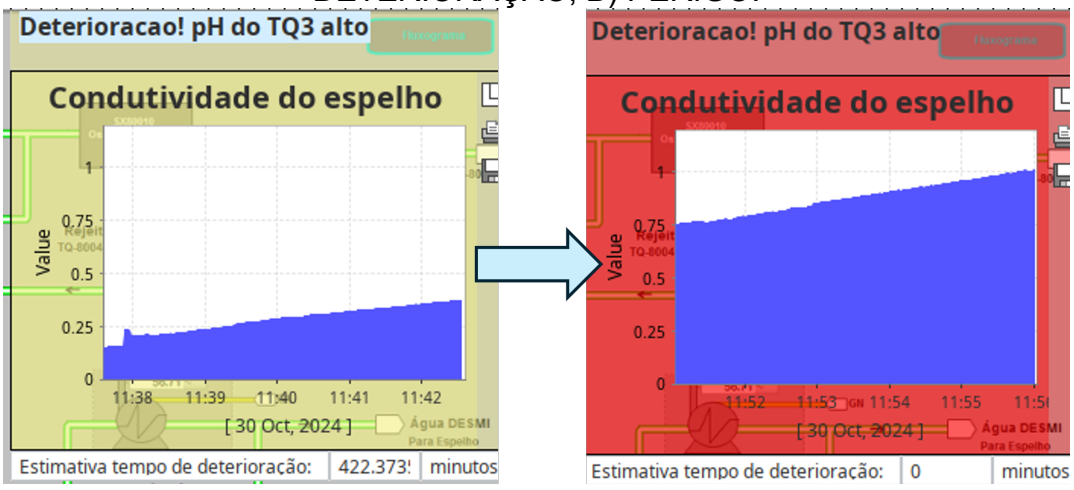
Uma vez que o botão é clicado, as informações do estado da operação são reveladas, imediatamente acima e abaixo do gráfico da variável, conforme pode ser visto na Figura 42. Naquele instante, a variável de controle estava bem estável, distante de seu limite superior e sem tendência de subida, o que fez a estimativa de tempo de deterioração ser igual à zero minutos e apresentação de estado normal acima do gráfico.

FIGURA 42: INFORMAÇÕES DE ESTADO DA VARIÁVEL MONITORADA APÓS CLICK DE BOTÃO.



Por fim, na Figura 43a e 76Figura 43b, pode ser visto as mudanças da visualização da interface à mudança de estado da variável de controle, isto é, quando esta apresenta um estado de deterioração e perigo, respectivamente. Nota-se que, primeiramente, a estimativa de tempo é computada apenas durante o período de deterioração, uma vez que quando ela chega no estado de “perigo” não há mais necessidade de se estimar tempo. Depois, é possível notar a mudança de cor nos gráficos e na questão da causalidade.

FIGURA 43:MUDANÇA NO ESTADO DA VARIÁVEL DE CONTROLE. A) DETERIORAÇÃO; B) PERIGO.



a) b)

6 CONCLUSÕES

CONCLUSÕES

Este trabalho apresentou uma metodologia para o desenvolvimento de alarmes inteligentes que levem em consideração a tendência de uma variável de processo para detecção de deterioração, bem como a aplicação de conceitos de *data mining* para extração de conhecimentos da operação por meio da utilização de um modelo de classificação baseado em árvore de decisão.

Para validar a metodologia, foram utilizados dados de operação de uma fábrica de vidros, mais especificamente do processo de desmineralização da água que é utilizada para a produção de espelhos. Estes dados eram referentes a medições realizadas entre o período de fevereiro até meados de maio de 2024, com diferentes tempos aquisição.

Assim, uma vez que estes dados foram coletados e tratados, foi aplicado um modelo de regressão linear para detecção de tendência de subida e descida da variável de processo, de modo que a inclinação resultante do modelo de regressão linear aliado à proximidade do valor medido com os limites da operação indicou o *status* como *normal, deterioração* ou *perigo.*

Posteriormente, após a rotulação da variável de processo de acordo com seu *status,* um modelo de árvore de decisão foi treinado para se extrair conhecimento de causalidade do *status*, se transformando numa ferramenta de auxílio na tomada de ações corretivas, ao sinalizar qual processo antecedente deve-se corrigir para que a variável de processo retorne para sua faixa desejada.

Por fim o algoritmo foi adaptado, se mostrando factível de ser utilizado em softwares SCADA até pela simplicidade de adaptação do código. Para o Scada-LTS, por exemplo, a qual os scripts são baseados em Javascript, esta solução poderia ser facilmente adaptada também, considerando as lógicas matemáticas e condicionais comuns a essas duas linguagens.

Futuros trabalhos poderão explorar maneiras de melhorar ainda mais o algoritmo, de modo a deixá-lo mais automático acerca da determinação de seus hiperparâmetros de tamanho da janela deslizante e o ângulo de tolerância, bem como a consideração de outras funções matemáticas auxiliares para o cálculo da tolerância dinâmica.

Realização de benchmarks com outros *datasets* também podem contribuir para análise de robustez da solução proposta, bem como o teste de outros algoritmos de cálculos de tendência e criação de regras, como ARIMA e lógica fuzzy, respectivamente. Outra sugestão, é metrificar o erro de estimativa de tempo ou até mesmo aplicar outras abordagens para aumentar a precisão da estimativa de tempo.

REFERÊNCIAS

ABNT. 2015. *Espelhos de prata - Requisitos e métodos de ensaio.* Rio de Janeiro : s.n., 2015.

—. 2004. *Vidro foat.* Rio de Janeiro : s.n., 2004.

ABRAVIDRO. 2023. Panorama Abravidro 2023 - ABRAVIDRO — abravidro.org.br. *Panorama Abravidro 2023 - ABRAVIDRO — abravidro.org.br.* 2023. [Accessed 03-07-2024].

Alghushairy, Omar, et al. 2021. A Review of Local Outlier Factor Algorithms for Outlier Detection in Big Data Streams. 2021, Vol. 5.

ANSI/ISA18-2. 2016. ISA 18-2:2016 - Management of Alarm Systems for the Process Industries. *ISA 18-2:2016 - Management of Alarm Systems for the Process Industries.* 2016. Disponível em: \url{https://www.isa.org/intech-home/2016/may-june/departments/isa18-alarm-management-standard-updated}.

Bahar-Gogani, Mahdi, Aslansefat, Koorosh e Shoorehdeli, Mahdi Aliyari. 2017. A novel extended adaptive thresholding for industrial alarm systems. 2017, pp. 759–765.

Belan, F. I. 1981. *Water Treatment.* s.l. : Mir, 1981. ISBN: 9780714717715LCCN: gb82038341.

Boldt, Martin, et al. 2021. Alarm prediction in cellular base stations using data-driven methods. 2021, Vol. 18, pp. 1925–1933.

Bonaccorso, Giuseppe. 2018. *Machine Learning Algorithms: Popular algorithms for data science and machine learning.* s.l. : Packt Publishing Ltd, 2018.

Breunig, Markus M., et al. 2000. LOF: identifying density-based local outliers. May de 2000, Vol. 29, pp. 93–104.

Costa, Herbert Rodrigues do Nascimento. 2006. *Aplicação de técnicas de inteligência artificial em processos de fabricação de vidro.* Universidade de São Paulo. 2006. Ph.D. dissertation.

Curtis, Heber D. 1911. *Methods of silvering mirrors.* s.l. : JSTOR, 1911. pp. 13–32.

Dix, Marcel, et al. 2022. An AI-based alarm prediction in industrial process control systems. 2022, pp. 242–245.

Dix, Marcel, et al. 2021. Anomaly detection in the time-series data of industrial plants using neural network architectures. 2021, pp. 222–228.

Düffer's F., Paul. 1996. Glass Reactivity and its Potential Impact on Coating Processes. *Glass Reactivity and its Potential Impact on Coating Processes.* 1996.

Freire, Laura Lúcia Ramos. 2016. A indústria de vidros planos. *A indústria de vidros planos.* s.l. : Banco do Nordeste do Brasil, 2016.

Garmaroodi, Mohammad Sadegh Sadeghi, et al. 2021. Detection of Anomalies in Industrial IoT Systems by Data Mining: Study of CHRIST Osmotron Water Purification System. 2021, Vol. 8, pp. 10280-10287.

Grafana Labs. 2018. *Grafana Documentation.* 2018.

Halkidi, Maria, Batistakis, Yannis e Vazirgiannis, Michalis. 2001. On clustering validation techniques. 2001, Vol. 17, pp. 107–145.

Han, J., Kamber, M. e Pei, J. 2011. *Data Mining: Concepts and Techniques.* s.l. : Elsevier Science, 2011. ISBN: 9780123814807LCCN: 2011010635.

Han, Jiawei, Pei, Jian e Tong, Hanghang. 2022. *Data mining: concepts and techniques.* s.l. : Morgan kaufmann, 2022.

Hawkins, D. M. 1980. *Identification of Outliers.* s.l. : Chapman and Hall, 1980. ISBN: 9780412219009LCCN: 81192025.

James, Gareth, et al. 2013. *An introduction to statistical learning.* s.l. : Springer, 2013. Vol. 112.

Koutroulis, Georgios, Mutlu, Belgin e Kern, Roman. 2023. A Causality-Inspired Approach for Anomaly Detection in a Water Treatment Testbed. 2023, Vol. 23.

Layton, Robert. 2015. *Learning data mining with python.* s.l. : Packt Publishing Ltd, 2015.

Liu, Keying, et al. 2021. Online anomaly detection with streaming data based on fine-grained feature forecasting. 2021, pp. 454–459.

MacQueen, James e others. 1967. Some methods for classification and analysis of multivariate observations. 1967, Vol. 1, pp. 281–297.

Maimon, Oded e Rokach, Lior. 2005. *Data mining and knowledge discovery handbook.* s.l. : Springer, 2005. Vol. 2.

Malhotra, Pankaj, et al. 2016. LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection. *LSTM-based Encoder-Decoder for Multi-sensor Anomaly Detection.* 2016.

Montgomery, Douglas C. 2009. *Statistical quality control.* s.l. : Wiley New York, 2009. Vol. 7.

Munirathinam, Sathyan. 2021. Drift detection analytics for iot sensors. 2021, Vol. 180, pp. 903–912.

Nicholas, P. Cheremisionoff e Cheremisinoff, A. 2002. *Handbook of water and wastewater treatment technologies.* 2002. pp. 33–37. Vol. 1.

Perales Gómez, Ángel Luis, et al. 2020. MADICS: A Methodology for Anomaly Detection in Industrial Control Systems. 2020, Vol. 12.

Pulker, Hans e Pulker, H. K. 1999. *Coatings on glass.* s.l. : Elsevier, 1999. Vol. 20.

Shi, Yilin, et al. 2024. Novel approach for industrial process anomaly detection based on process mining. 2024, Vol. 136, p. 103165.

Thorndike, Robert L. 1953. Who belongs in the family? 1953, Vol. 18, pp. 267–276.

Veolia. 2024. Handbook of Industrial Water Treatment | Veolia — watertechnologies.com. *Handbook of Industrial Water Treatment | Veolia — watertechnologies.com.* 2024. [Accessed 12-07-2024].

Villalobos, Kevin, Suykens, Johan e Illarramendi, Arantza. 2021. A flexible alarm prediction system for smart manufacturing scenarios following a forecaster–analyzer approach. 2021, Vol. 32, pp. 1323–1344.

VIVIX. 2024. *Entrevista concedida a José Vinícius.* 2024. Pernambuco, Fevereiro de 2024.

Wang, Xue Z. 2012. *Data mining and knowledge discovery for process monitoring and control.* s.l. : Springer Science & Business Media, 2012.

Weisberg, Sanford. 2005. *Applied linear regression.* s.l. : John Wiley & Sons, 2005. Vol. 528.

Zheng, Shaodong e Zhao, Jinsong. 2020. A new unsupervised data mining method based on the stacked autoencoder for chemical process fault diagnosis. 2020, Vol. 135, p. 106755.

ANEXO 1

TABELA 8: VARIÁVEIS E SUAS UNIDADES DE MEDIDAS.

Variáveis disponíveis no setor de desmineralização da água					
ENTRADA	FIT80045	m^3/h	**LEITO MISTO**	FIT80606	m^3/h
FILTROS E TANQUE 1	PDIT80602	bar		AIT80609	pH
	PIT80603	bar		AIT80605	uS/cm
	PDT80601	bar		AIT80608*	ppb
	PDT80603	bar		LIT80605	%
	AIT80602	pH		AIT80607	uS/cm
	AIT80613	uS/cm	**TANQUE 3**	AIT80610	pH
	AIT80601	ppb		AIT80606*	ppb
OSMOSE 1	AIT80617*	pH		FIT80606	m^3/h
	LIT80601	%	**PRÉ-LAVAGEM**	PIT80613	bar
	FIT80601	m^3/h		FIT80605	m^3/h
	PIT80605	bar	**LAMINADO**	PIT80006	bar
	PIT80607	bar	**ESPELHO**	FIT80041	m^3/h
	FIT80602	m^3/h		FIT80042	m^3/h
OSMOSE 2	AIT80616	pH		PIT80007	bar
	AIT80603	uS/cm		AIT80614	uS/cm
	AIT80604	ppb		AIT80611	pH
	FIT80601	m^3/h			
	PIT80611	bar	* Não estão funcionando		
	PIT80613	bar			
	FIT80605	m^3/h			
TANQUE 2	FIT80604	m^3/h			