

Universidade Federal do Espírito Santo
Centro Tecnológico
Programa de Pós-Graduação em Engenharia Elétrica

Catherine Salvador Henrique

Monitoramento de Processos Industriais sob o Efeito de
Concept Drift Utilizando Análise Externa e Aprendizado
Ativo

Vitória

2020

Catherine Salvador Henrique

**Monitoramento de Processos Industriais sob o Efeito de
Concept Drift Utilizando Análise Externa e Aprendizado
Ativo**

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Universidade Federal do Espírito Santo

Centro Tecnológico

Programa de Pós-Graduação em Engenharia Elétrica

Orientador: Patrick Marques Ciarelli

Vitória

2020

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

S182m Salvador Henrique, Catherine, 1995-
Monitoramento de Processos Industriais sob o Efeito de Concept Drift Utilizando Análise Externa e Aprendizado Ativo / Catherine Salvador Henrique. - 2020.
62 f. : il.

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

1. Controle de processo. 2. Modelos de engenharia. 3. Análise multivariada. 4. Aprendizado de máquinas. 5. Processos industriais. I. Marques Ciarelli, Patrick. II. Universidade Federal do Espírito Santo. Centro Tecnológico. III. Título.

CDU: 621.3

Catherine Salvador Henrique

**Monitoramento de Processos Industriais sob o Efeito de
Concept Drift Utilizando Análise Externa e Aprendizado
Ativo**

Dissertação apresentada ao Programa de PósGraduação em Engenharia Elétrica do Centro Tecnológico da Universidade Federal do Espírito Santo, como requisito parcial para obtenção do Grau de Mestre em Engenharia Elétrica.

Trabalho aprovado. Vitória, 24 de março de 2020:



Prof. Dr. Patrick Marques Ciarelli Orientador



Prof. Dr. Marco Antonio de Souza

Leite Cuadros

Membro Externo



Prof. Dr. Klaus Fabian Côco

Membro Externo

Vitória

2020

Dedico este trabalho à minha família.

Agradecimentos

Agradeço a Deus pelo dom da vida e por todas as oportunidades colocadas em meu caminho durante esses anos. Agradeço à Nossa Senhora, por toda a intercessão recebida nesta caminhada.

Um agradecimento especial aos amores da minha vida e razão de tudo o que tenho construído, meus pais Paulo e Terezinha e meu irmão Júnior, por todo o apoio incondicional e carinho. Agradeço também aos meus avós Luiza, José, Maria e Antônio (que agora olha por nós lá do céu), por todas as orações e me incentivarem a ir atrás dos meus sonhos.

Agradeço às amigadas que construí e cultivei em Vitória, as quais considero família, por me suportarem nas dificuldades e por compartilharem também tantos momentos incríveis durante a jornada do mestrado.

Aos amigos do LabCisne e do LCI, sem os quais eu não teria conseguido concluir esta etapa, muito obrigada por todo o conhecimento compartilhado, por toda a cumplicidade e por proporcionarem tanta alegria no ambiente de trabalho durante os anos de mestrado. São todos pesquisadores incríveis, que se tornaram minha inspiração.

Aos sempre presentes amigos de Muniz Freire, por toda a preocupação, apoio e carinho. Obrigada por estarem há tantos anos enchendo minha vida de alegria e amor. Também aos amigos de Viçosa, que mesmo tão longes e espalhados pelo mundo, torcem por mim.

Aos amigos de São Paulo, que me incentivaram a terminar o mestrado mesmo com as dificuldades do caminho. Um agradecimento especial a Vitor, por me apoiar e por compreender a importância dessa etapa na minha vida.

Muito obrigada a todos os professores que me ensinaram e ajudaram a ser uma profissional melhor, em particular ao professor Patrick Marques Ciarelli, pelos anos de orientação, compreensão e paciência. Agradeço também ao professor Celso José Munaro pela oportunidade de projeto.

Por fim, agradeço à FAPES e à VALE S.A. pelo financiamento desta pesquisa. Agradeço também à UFES e ao Programa de Pós-Graduação em Engenharia Elétrica (PPGEE) por todo o suporte recebido.

“Ainda que eu tivesse o dom da profecia, o conhecimento de todos os mistérios e de toda a ciência; ainda que eu tivesse toda a fé, a ponto de transportar montanhas, se não tivesse o amor, eu não seria nada.”

1 Coríntios 13,2

“Remember to look up at the stars and not down at your feet. Try to make sense of what you see and wonder about what makes the universe exist. Be curious. And however difficult life may seem, there is always something you can do and succeed at. It matters that you don't just give up.”

Stephen Hawking

“Olhe todas as que vieram antes de nós, não há segredo, a potência de ser mulher atravessa nossas veias. Somos fortalezas”

Rayane Leão

Resumo

O monitoramento adequado de processos industriais é de suma importância para detectar, prever e diagnosticar falhas, além de garantir que os produtos possuam qualidade e estejam dentro das especificações esperadas. Desta forma, é importante detectar distúrbios no processo, mau funcionamento de equipamentos ou outros eventos prejudiciais o quanto antes, para que, em seguida, seja possível identificar a causa do comportamento indesejado e ações corretivas sejam tomadas. Existem diversos métodos na literatura que tratam do monitoramento. Dentre eles, a utilização de técnicas de análise externa e métodos multivariados é bem difundida, porém, existem mudanças no processo às quais esses métodos não são robustos o suficiente para adequar e monitorar, gerando falsos alarmes em situações de normalidade. Assim, neste trabalho é proposto um sistema de união das técnicas de PCA (*Principal Component Analysis*) e análise externa com um método de aprendizado ativo através de detecções de *concept drift* para gerar um sistema de monitoramento que se ajuste às mudanças de comportamento de um processo de uma planta industrial. O método foi aplicado a um problema real de um ventilador industrial e os resultados apresentados mostraram que o modelo se adequou às mudanças do processo sem armazenar dados excessivos para treinamento do modelo e com um número menor de falsos alarmes em relação a um modelo sem atualização.

Palavras-chave: Monitoramento de Processos, Aprendizado Ativo, *Concept Drift*

Abstract

Appropriate industrial processes monitoring is of paramount importance to detect, predict and diagnose faults, in addition to ensuring that the products have quality and are within the expected specifications. Thus, it is important to detect disturbances in the process, equipment malfunction or other harmful events as soon as possible, so that afterward, it is possible to identify the cause of the unwanted behavior and take corrective actions. There are several methods in the literature that deal with monitoring. Among them, the use of external analysis techniques and multivariate methods is widespread. However, there are process changes to which these methods are not robust enough to adapt and monitor, generating false alarms in normal situations. Thus, this work proposes a system of joining PCA (Principal Component Analysis) and external analysis techniques with an active learning method through Concept Drift detections to generate a monitoring system that adjusts to changes in the behavior of an industrial plant process. The method was applied to a real problem of an industrial fan and the results presented showed that the model was adapted to the process changes without storing excessive data for training the model and with a lower number of false alarms in relation to a model without updating.

Keywords: Process Monitoring, Active Learning, Concept Drift.

Lista de ilustrações

Figura 1 – Classificação dos algoritmos de detecção e diagnóstico de falhas.	21
Figura 2 – Ilustração da carta de Shewhart.	25
Figura 3 – Distribuição normal e intervalos de confiança.	26
Figura 4 – Controle de qualidade de duas variáveis que ilustram a não-aplicabilidade de gráficos univariados em processos multivariados.	27
Figura 5 – Descrição do modelo de análise externa	31
Figura 6 – Hiperplano de separação e projeção das margens virtuais com limiar de incerteza de 0,95 e 0,90.	33
Figura 7 – Modelo de Aprendizado Passivo (1) Ativo (2)	35
Figura 8 – Funcionamentos dos algoritmos hierárquicos aglomerativos e divisivos.	37
Figura 9 – Exemplo de dendograma.	38
Figura 10 – Seleção de dados de treino.	41
Figura 11 – Criação do modelo e limiares de monitoramento.	42
Figura 12 – Fluxograma do monitoramento dinâmico com uso de <i>Concept Drift</i>	43
Figura 13 – Fluxograma do monitoramento dinâmico com uso de <i>Concept Drift</i> e clusterização	44
Figura 14 – Fluxograma do processo de pelotização.	45
Figura 15 – Esquema de um ventilador industrial.	46
Figura 16 – Variáveis de monitoramento: rotação, produção, vibrações LA e LNA e de carcaça, potência e torque.	48
Figura 17 – Detalhamento do momento em que os <i>drift</i> são detectados no modelo “Polinomio de grau 2 - sem clusterização”.	52
Figura 18 – Comparação dos momentos de adição de dados de treino para o modelo polinomial de 2º Grau.	53
Figura 19 – Sinais de rotação e vibrações dos dados com falha.	54
Figura 20 – Sinais de rotação e vibrações dos dados com falha.	54
Figura 21 – Momento em que se inicia a falha.	55
Figura 22 – Momento da detecção de <i>drift</i> nos dados de falha.	56

Lista de tabelas

Tabela 1 – Variáveis do Modelo.	47
Tabela 2 – Resultados (30 dias) sem atualização de modelo - Dados de normalidade. . .	49
Tabela 3 – Resultados (30 dias) com atualização de modelo - Dados de normalidade. . .	49
Tabela 4 – Resultados (30 dias) com atualização de modelo e clusterização - Dados de normalidade.	50
Tabela 5 – Resultados (10 dias) com atualização de modelo- Dados de normalidade. . .	50
Tabela 6 – Resultados (10 dias) com atualização de modelo e clusterização de dados - Dados de normalidade.	51
Tabela 7 – Variáveis do Modelo.	53
Tabela 8 – Resultados do monitoramento de dados com falha.	54

Sumário

1	INTRODUÇÃO	20
1.1	Justificativa	22
1.2	Objetivos	23
1.3	Estrutura da Dissertação	23
2	REFERENCIAL TEÓRICO	24
2.1	Monitoramento Estatístico de Processos	24
2.1.1	Análise Univariada	24
2.1.2	Análise Multivariada	27
2.1.2.1	Análise de Componentes Principais	28
2.1.3	Deteção de falhas utilizando PCA	29
2.2	Análise Externa	30
2.3	<i>Concept Drift</i>	32
2.4	Aprendizado Ativo	34
2.5	Clusterização	35
2.5.1	<i>K-means</i>	35
2.5.2	Clusterização Hierárquica	36
3	METODOLOGIA PROPOSTA	39
3.1	Seleção de variáveis	40
3.2	Treinamento	40
3.3	Validação	41
3.4	Teste	42
3.4.1	Monitoramento estático do processo	42
3.4.2	Monitoramento dinâmico com uso de <i>Concept Drift</i>	43
3.4.3	Monitoramento dinâmico com uso de <i>Concept Drift</i> e clusterização de dados	43
4	ESTUDO DE CASO, RESULTADOS E DISCUSSÕES	45
4.1	Ventilador Industrial	45
4.1.1	Situação de normalidade	46
4.1.2	Situação de falha	53
5	CONCLUSÃO E TRABALHOS FUTUROS	57
	REFERÊNCIAS	59

1 Introdução

As operações industriais vêm se tornando cada vez mais complexas e ao mesmo tempo têm facilitado a captura de grandes quantidades de dados de processo, o que impulsiona novos conceitos e ferramentas estatísticas de controle e monitoramento das mesmas (GE; SONG, 2012).

Para que os processos industriais sejam seguros e produzam consistentemente produtos de alta qualidade, é necessário que o monitoramento deles seja feito de forma adequada (SATOYAMA; FUJIWARA; KANO, 2016). Tradicionalmente, tal monitoramento é feito através da construção de complexos modelos matemáticos e, em seguida, tais modelos são otimizados e/ou usados para monitoramento de processo e detecção de falhas. No entanto, a construção desses modelos ou a personalização dos modelos gerais existentes para uma planta específica exigem tempo e esforço (KOURTI, 2002).

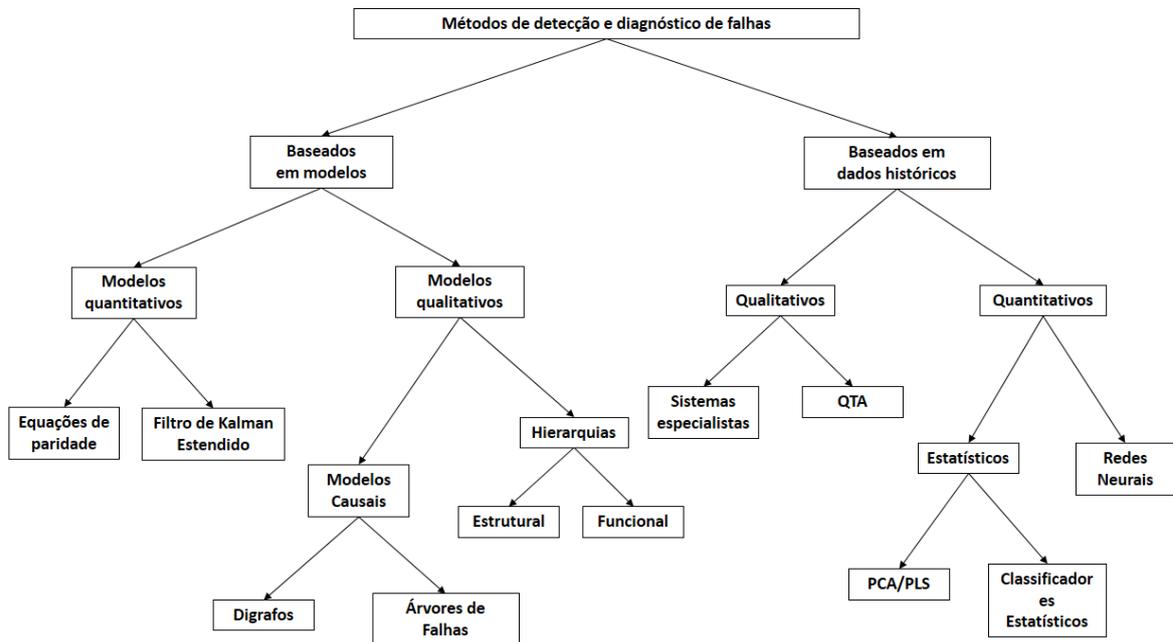
Existem diversos trabalhos na literatura que tratam sobre monitoramento de processos, variando de métodos analíticos a inteligência artificial e abordagens estatísticas (VENKATASUBRAMANIAN et al., 2003b). Pode-se classificar os métodos de detecção e diagnóstico em dois grandes grupos: os baseados em modelos, que geralmente são desenvolvidos com base em algum entendimento fundamental da física e química do processo (seja quantitativo ou qualitativo), e modelos baseados em dados históricos, onde apenas a disponibilidade de uma grande quantidade de dados do processo é necessária (VENKATASUBRAMANIAN; RENGASWAMY; KAVURI, 2003; VENKATASUBRAMANIAN et al., 2003a). A Figura 1 possui um esquema com exemplos dos tipos de métodos de detecção e diagnósticos de falhas.

A metodologia baseada em dados históricos vem recebendo atenção na área de pesquisa, bem como em aplicações industriais. Uma técnica consolidada nesse contexto é a análise multivariada do processo.

O controle estatístico multivariado de processos (*Multivariate Statistic Process Control* - MSPC) unido a técnicas de modelagem permite a detecção de falhas (CHIANG; RUSSELL; BRAATZ, 2001). Neste tipo de abordagem, geralmente o processo é comparado com um modelo, de forma que um desvio significativo no processo possa ser indicado como falha. E tal união das técnicas de modelagem com as técnicas de controle estatístico de processos *online* é uma ferramenta poderosa para alcançar a estabilidade, de forma a produzir sempre produtos com qualidade (MONTGOMERY; RUNGER, 2010).

Técnicas importantes e clássicas de MSPC são as baseadas em variância/covariância, que se mostram mais apropriadas para aplicações de monitoramento de processos. Essas técnicas são a Análise de Componentes Principais (*Principal Component Analysis* - PCA)

Figura 1 – Classificação dos algoritmos de detecção e diagnóstico de falhas.



Fonte: Adaptado de Venkatasubramanian et al. (2003b)

e Mínimos Quadrados Parciais (*Partial Least Squares* -PLS) (KRUGER; XIE, 2012).

Tais abordagens obtêm um bom desempenho quando o processo opera em somente um único ponto de operação (KANO et al., 2004). Entretanto, muitos processos industriais reais possuem diversas regiões de operação, assim, além de detectar eficientemente os pontos de falha, é de suma importância que uma mudança de ponto de operação não indique falsos alarmes. Neste contexto, surge a análise externa, onde as variáveis do processo são divididas em dois grupos, e assume-se que as variáveis principais podem ser explicadas pelas variáveis externas através de um modelo (ZHAO; ZHANG; XU, 2004). Os resíduos medidos entre a diferença das variáveis medidas e estimadas são utilizados para a detecção e diagnósticos de falhas. Assim, as variáveis passam a ser independentes da variação da região de operação.

Além disso, as variáveis do processo podem passar por mudanças ao longo do tempo, como efeito de envelhecimento natural, mudança de matéria prima, corrosão, etc. O que causa uma descalibração do modelo de monitoramento estatístico previamente obtido. Essas mudanças dinâmicas nas variáveis, tais que elas não possuam mais distribuições estacionárias, são chamadas de *concept drift* (SOBHANI; BEIGY, 2011). Assim, modelagens e controles estatísticos podem não ser robustos o suficiente para monitorar os processos se eles não forem atualizados periodicamente.

Métodos amplamente utilizados para atualização de modelos são, por exemplo, a utilização de Mínimos Quadrados Recursivos (YOUNG, 2011) e PCA Recursivo (PORTNOY

et al., 2016).

Um método que tem ganho destaque para detecção de anomalias em processos é o de *Concept Drift Detection*, com exemplo de aplicações em (WANG; ABRAHAM, 2015), (SOBHANI; BEIGY, 2011), (GAMA et al., 2004), (RU; ANDROMEDA; MARSONO, 2014), (LOO; MARSONO, 2015) e (FARID; RAHMAN, 2012). Desta forma, essa é uma técnica que também pode ser utilizada como gatilho para atualização de modelos de monitoramento.

Sabe-se que o problema de detecção de falhas em um processo industrial pode ser dividido em dois tipos de análise: uma primeira, que detecta falhas abruptas ou eventos catastróficos, e que deve ser feita de forma muito eficiente, porém, há uma segunda, e de interesse crucial, que deve detectar falhas menores (graduais ou incipientes) causadas por motivos como os descritos anteriormente, que afetam o processo sem fazer com que ele pare, mas que é necessária para evitar a ocorrência subsequente de eventos mais catastróficos (BASSEVILLE; NIKIFOROV et al., 1993).

Logo, deve-se criar modelos que sejam robustos e adaptativos, para abranger todos os tipos de falhas e situações de operação. Uma forma de atacar essa questão é utilizar um método de aprendizado ativo, em que, a partir da detecção de uma anomalia no processo, seja enviado um alerta para um operador da planta, e o mesmo possa identificá-lo como uma situação de normalidade ou não. Dessa forma, o modelo pode ser aprimorado com o tempo, mesmo que ele não tenha sido obtido inicialmente para abranger todos os possíveis diferentes pontos de operação e o processo esteja sob efeito de *concept drift*. Com isso, o monitoramento não deixa de alertar as falhas e gera um menor número de falsos alarmes a medida que ele adapta às características do processo.

Neste trabalho será proposta uma metodologia de monitoramento de falhas baseada em aprendizado ativo, que utiliza PCA e modelos de análise externa, aplicada aos dados reais de um ventilador industrial.

1.1 Justificativa

Nos trabalhos desenvolvidos em Assis (2018) e Faria (2018), que utilizaram dados do mesmo ventilador industrial usado nesta Dissertação, foram estudados métodos de monitoramento e detecção de falhas em processos que possuem múltiplos pontos de operação. As metodologias propostas por ambos se mostraram eficientes e especialmente acuradas para aplicação em uma situação de monitoramento real, porém, elas se tratam de modelos estáticos e, desta forma, este trabalho vem com a proposta de aliar as metodologias propostas anteriormente à uma forma de atualização que permita a adequação constante às possíveis mudanças de processo que não sejam contempladas pela análise externa. Em especial às mudanças de distribuições estatísticas de variáveis, que se caracterizam como

um *Concept Drift*.

1.2 Objetivos

O objetivo deste trabalho consiste em obter um sistema de monitoramento de processos industriais que consiga se adequar às mudanças de processos ocorridas ao longo do tempo, de forma a gerar o mínimo de falsos positivos e que ainda assim seja acurado na detecção das falhas. Para atingir tal objetivo, é proposto um sistema que combine a metodologia clássica de análise externa à metodologia de análise de *concept drift* com a utilização de aprendizado ativo.

1.3 Estrutura da Dissertação

A dissertação está estruturada da seguinte forma: no Capítulo 2 é feita uma revisão teórica de todas as técnicas utilizadas no trabalho. No Capítulo 3 é descrita a metodologia proposta. No Capítulo 4, a metodologia é aplicada a duas situações distintas de um estudo de caso de um forno industrial. E, por fim, no Capítulo 5 se encontram as conclusões e sugestões de trabalhos futuros.

2 Referencial Teórico

2.1 Monitoramento Estatístico de Processos

Nos processos industriais modernos, uma grande quantidade de dados é gerada e coletada, sendo assim, para monitorar essa grande base de dados, conseguindo identificar possíveis situações de falha, é necessário criar modelos que expliquem o processo. Diferentemente dos métodos tradicionais baseados nas equações que regem o processo, os métodos baseados em dados históricos raramente precisam de qualquer conhecimento profundo prévio acerca do mesmo. Extrair as informações úteis dos dados registrados de um processo, os modelos baseados em dados também são capazes de modelar o relacionamento entre diferentes variáveis (GE; SONG, 2012). Essa metodologia vem recebendo atenção na área de pesquisa, bem como em aplicações industriais.

O controle estatístico de processos (*Statistical Process Control* - SPC) é uma das formas de monitoramento baseada em dados, que consiste em uma variedade de técnicas para monitorar as variáveis de determinado processo, fornecendo insumos para melhoria contínua da operação. Dentre as etapas do método, é fundamental empregar um conjunto de ferramentas estatísticas para estabelecer os limites de controle para um processo e, caso alguma situação de anormalidade ocorra, ela deve ser indicada pelo conjunto de monitoramento (ZHANG, 2000).

As primeiras técnicas a serem desenvolvidas foram as análises univariadas, que levam em consideração as variáveis de forma independente e ignoram as correlações entre as mesmas, sendo aplicáveis para sistemas mais simples e se tornando menos aplicáveis em sistemas mais complexos. Neste contexto, o Controle Estatístico de Processo Multivariado (*Multivariate Statistical Process Control* - MSPC), que analisa as variáveis de forma mútua e permite uma visão ampla do processo. Nas subseções a seguir elas são descritas com mais detalhes.

2.1.1 Análise Univariada

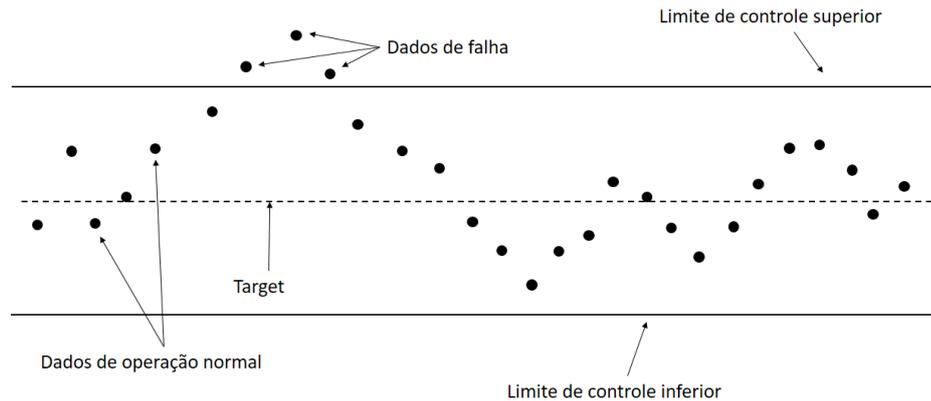
Uma análise estatística univariada pode ser usada para determinar os limites para cada variável do processo separadamente, onde tais limites encontrados definem a região de uma operação normal e uma violação desse limite indica uma situação de falha no processo.

Os gráficos de controle estatístico univariado (SPC), como a carta de Shewhart, *Cumulative Sum Control Chart* (CUSUM) e *Exponentially Weighted Moving Average* (EWMA) são amplamente utilizadas para monitorar variáveis-chave do produto, a fim de

detectar a ocorrência de algumas anormalidades (MACGREGOR; KOURTI, 1995).

Na Figura 2 encontra-se uma ilustração da carta de Shewhart, onde os pontos são as amostras do processo e as linhas representam os limiares de operação obtidos.

Figura 2 – Ilustração da carta de Shewhart.



Fonte: Adaptado de Chiang, Russell e Braatz (2001)

Tais limites de controle são dados pela Equação 2.1, assumindo-se que os dados possuem uma distribuição normal, onde μ representa a média e σ o desvio padrão dos dados:

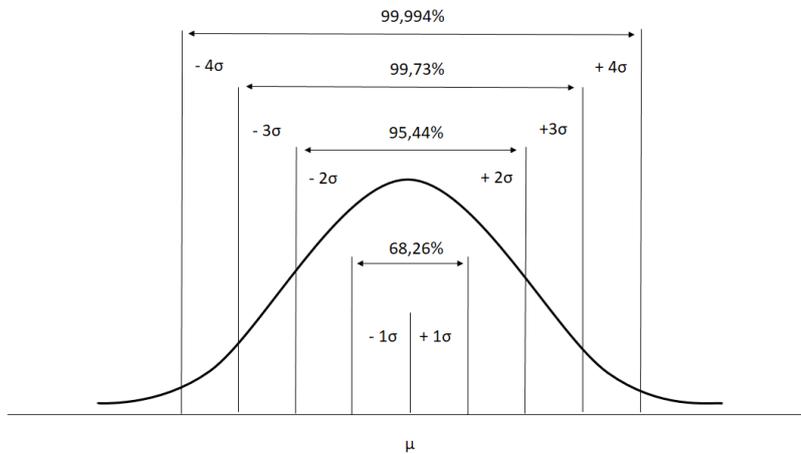
$$\mu - c_{\frac{\alpha}{2}}\sigma \leq x \leq \mu + c_{\frac{\alpha}{2}}\sigma \quad (2.1)$$

onde $c_{\frac{\alpha}{2}}$ representa a amplitude do desvio padrão do nível de confiança desejado, que é definido por $(1 - \alpha)$, sendo α o nível de significância da distribuição, ou seja, a probabilidade de se obter um falso positivo. O número de desvios padrões a serem utilizados é calculado analisando-se o gráfico de distribuição acumulada da variável (CHIANG; RUSSELL; BRAATZ, 2001). Na Figura 3 é mostrada a densidade de probabilidade de uma distribuição normal, com seus intervalos de confiança.

Para o Gráfico de Controle de Soma Cumulativa (*Cumulative Sum Control Chart - CUSUM*), ao invés de plotar a distribuição das amostras, é plotada a soma cumulativa das observações, de forma que o sistema fica mais sensível a pequenas variações. Para o gráfico de média móvel exponencial (*Exponentially Weighted Moving Average - EWMA*) é dado menos peso às amostras mais antigas e mais peso às amostras mais recentes, de forma a tornar o sistema mais robusto às mudanças de processo que ocorram com o tempo (ZHANG, 2000).

As técnicas de análise univariada (Shewhart, CUSUM e EWMA) determinam os limites para cada variável de observação individualmente, sem considerar as informações

Figura 3 – Distribuição normal e intervalos de confiança.



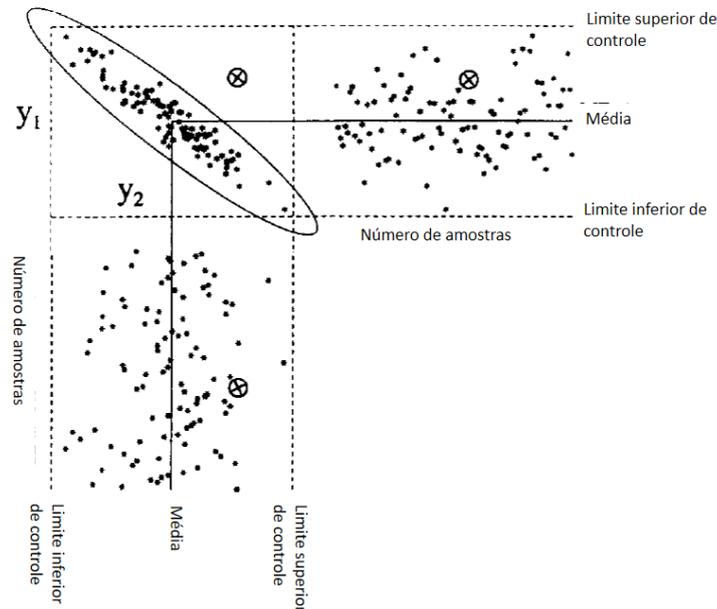
Fonte: Autora

das demais variáveis. Dado que esses métodos ignoram a correlação entre as variáveis, eles não caracterizam com assertividade o comportamento das variáveis dos processos industriais complexos (CHIANG; RUSSELL; BRAATZ, 2001).

Um exemplo do problema em se utilizar a análise univariada para processos multivariados é encontrado na Figura 4, na qual foram plotadas apenas duas variáveis de qualidade (y_1 , y_2) de um produto. O processo está sob controle estatístico, y_1 e y_2 seguem uma distribuição normal e são correlacionadas ($\rho_{y_1, y_2} = -0,94$), como ilustrado no gráfico conjunto de $y_1 \times y_2$. A elipse representa um contorno de controle multivariado do processo, com limite de confiança de 99%, e os pontos representam o conjunto de observações da distribuição. As mesmas distribuições são plotadas individualmente na mesma figura com gráficos de Shewhart com y_1 e y_2 versus o número de amostras (tempo), com seus respectivos limites de controle (com 99% de limite de confiança). Entre os dados do processo existe um problema relacionado a um produto que é representado por \otimes . Analisando-se individualmente os gráficos de Shewhart de cada variável, percebe-se que o produto problemático está dentro dos limites de controle e, portanto, em situação de normalidade. Porém, quando analisa-se a plotagem multivariada $y_1 \times y_2$ vê-se que o produto indicado por \otimes está claramente fora dos limites de controle (MACGREGOR; KOURTI, 1995).

No exemplo mostrado na Figura 4 ocorre um erro Tipo-II, onde não ocorre uma indicação de falha, mesmo com a falha tendo acontecido no sistema real, são os chamados falsos negativos. Um erro Tipo-I existe quando é indicada uma falha, mas ela não ocorreu na realidade, são chamados de falsos positivos (CHIANG; RUSSELL; BRAATZ, 2001).

Figura 4 – Controle de qualidade de duas variáveis que ilustram a não-aplicabilidade de gráficos univariados em processos multivariados.



Fonte: Adaptado de [MacGregor e Kourti \(1995\)](#)

2.1.2 Análise Multivariada

Como descrito anteriormente, as técnicas univariadas não possuem um bom desempenho em sistemas mais complexos, onde existem, por exemplo, diversos sensores que fazem a medição de uma mesma variável, ou ainda quando é necessário monitorar variáveis interdependentes. A finalidade da análise univariada é realizar uma exploração descritiva das variáveis, entendendo as estatísticas primárias e demais informações relevantes de cada variável individualmente, sendo, portanto, um primeiro passo na análise das variáveis do processo, que para serem analisadas em sua completude necessitam de ferramentas mais adequadas. Desta forma, o Controle Estatístico de Processo Multivariado (MSPC), que engloba um conjunto de ferramentas poderosas capazes de compactar dados e reduzir sua dimensionalidade, deixando que as informações essenciais sejam retidas e se tornem mais fáceis de analisar do que o enorme conjunto de dados original, além de serem capazes de lidar com ruído e correlação para extrair informações de maneira eficaz ([VENKATASUBRAMANIAN et al., 2003a](#)). Métodos de projeção multivariados, como Análise de Componentes Principais (*Principal Component Analysis - PCA*), Mínimos Quadrados Parciais (*Partial Least-Squares - PLS*), Análises de Componentes Independentes (*Independent Component Analysis - ICA*) e Discriminante Linear de Fisher (*Fisher discriminant analysis - FDA*) são amplamente utilizados ([GE et al., 2008](#)).

2.1.2.1 Análise de Componentes Principais

A Análise de Componentes Principais (*Principal Component Analysis* - PCA) é uma técnica que produz uma representação dos dados com uma menor dimensão, mantendo a estrutura de correlação entre as variáveis e é ideal para capturar a variabilidade dos mesmos (CHIANG; RUSSELL; BRAATZ, 2001). É uma combinação linear das variáveis originais em um espaço dimensional reduzido (ZHANG, 2000).

Considere uma matriz de dados $X^{n \times m}$, onde m é o número de variáveis e n o número de amostras (como na Equação 2.2), normalizada com média zero e variância unitária.

$$X = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1m} \\ x_{21} & x_{22} & \cdots & x_{2m} \\ \vdots & \vdots & & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{nm} \end{bmatrix} \quad (2.2)$$

De acordo com Portnoy et al. (2016), para o método PCA, calcula-se a matriz S de covariância de X , conforme a Equação 2.3:

$$S = \frac{1}{n-1} X^T X \quad (2.3)$$

E a decomposição espectral de S é dada pela Equação 2.4:

$$S = V \Lambda V^T \quad (2.4)$$

onde $\Lambda^{m \times m}$ é uma matriz diagonal contendo os autovalores ordenados de forma decrescente e $V^{m \times m}$ é composta pelos autovetores correspondentes. É selecionado o número a de componentes principais a ser utilizado. Definindo-se $P^{m \times a}$ como a Matriz de Carregamento, contendo as primeiras a colunas (componentes retidas) de V . As projeções dos dados da matriz X no espaço de menor dimensão estão contidas na matriz de pontuação T (Equação 2.5):

$$T = XP \quad (2.5)$$

E a projeção de T de volta ao espaço de observações m -dimensional original é dada pela Equação 2.6:

$$\hat{X} = TP^T. \quad (2.6)$$

A diferença entre X e \hat{X} é a matriz residual E (Equação 2.7):

$$E = X - \hat{X}. \quad (2.7)$$

O subespaço contido na matriz E possui uma baixa relação sinal-ruído, e a remoção desse espaço de X pode produzir uma representação mais precisa do processo, que está armazenada em \hat{X} .

A seleção do número de variáveis principais é parte fundamental do processo de modelagem. Para a seleção das a componentes existem diversas técnicas na literatura, dentre elas as descritas por [Chiang, Russell e Braatz \(2001\)](#):

- Variância percentual acumulada: esse método determina o número a de componentes principais através do cálculo do menor número de vetores de carregamento necessários para explicar um percentual mínimo escolhido da variância total do sistema, escolhido arbitrariamente.
- Análise paralela: a dimensionalidade é determinada comparando-se o perfil de variação total com o perfil obtido assumindo observações independentes das variáveis. A ordem de redução é determinada como o ponto no qual os dois perfis se cruzam.
- *Scree test*: parte da ideia de que a variância corresponde a um ruído aleatório que possui um perfil linear. A dimensionalidade será determinada localizando o valor onde a variância não possui mais um perfil linear. É um método difícil de automatizar por ser ambíguo.
- Validação cruzada: é construída utilizando o valor de previsão da soma dos quadrados (*Prediction Residual Sum of Squares* – PRESS) (Equação 2.8):

$$PRESS(a) = \frac{1}{mn} \|X - \hat{X}\|_F^2 \quad (2.8)$$

onde a representa o número de componentes retidas que são utilizadas no cálculo de \hat{X} , e $\|\cdot\|_F^2$ é dada pela norma de Forbenius. Para a utilização desta técnica, o conjunto de dados é dividido em vários blocos. Cada vez que um bloco é deixado de fora, é feita uma análise PCA sobre o restante dos dados e o cálculo de *PRESS* é feito no bloco que foi retirado. Tal procedimento é repetido até que todos os blocos tenham sido deixados de fora pelo menos uma vez. O valor de a associado com o menor valor médio da estatística *PRESS* determina o número de componentes selecionadas.

2.1.3 Detecção de falhas utilizando PCA

A técnica PCA, aliada à uma estatística multivariada, que leva em consideração as correlações entre as variáveis, gera uma solução capaz de monitorar os dados e indicar quando os mesmos estiverem fora dos limites de normalidade.

Um método muito difundido para análise estatística multivariada é a estatística T^2 de *Hotelling* ([CHIANG; RUSSELL; BRAATZ, 2001](#)).

A estatística T^2 de *Hotelling* é dada pela Equação 2.9:

$$T^2 = \mathbf{x}^\top P \Lambda_a^{-1} P^\top \mathbf{x} \quad (2.9)$$

onde Λ_a contém os autovalores referentes as componentes principais selecionadas.

Para determinar se o processo está em estado de normalidade, é necessário determinar limiares de controle baseados em dados de onde não existam falhas, os mesmos usados para construir o modelo PCA. Para a estatística em questão, o limiar de controle é dado pela Equação 2.10, onde n é a quantidade de amostras e α é o nível de significância desejado.

$$T_\alpha^2 = \frac{a(n-1)}{n-1} F_{a,n-a,\alpha} \quad (2.10)$$

Desta forma, a utilização dos dados gerados pelo modelo PCA, junto à estatística T^2 levam a um monitoramento do processo, onde dados que saíam do limite de confiança são indicados como falhas.

2.2 Análise Externa

Os processos industriais comumente operam em múltiplas regiões de operação, como em diferentes valores de *setpoints*. Essas variáveis são alteradas intencionalmente, segundo as necessidades, mas geram mudanças em todo o processo, podendo indicar erroneamente falhas. O modelo PCA é eficaz somente para um único ponto de operação quando em estado estacionário, logo, é necessário combinar outras técnicas para viabilizar a aplicação em múltiplas regiões de operação.

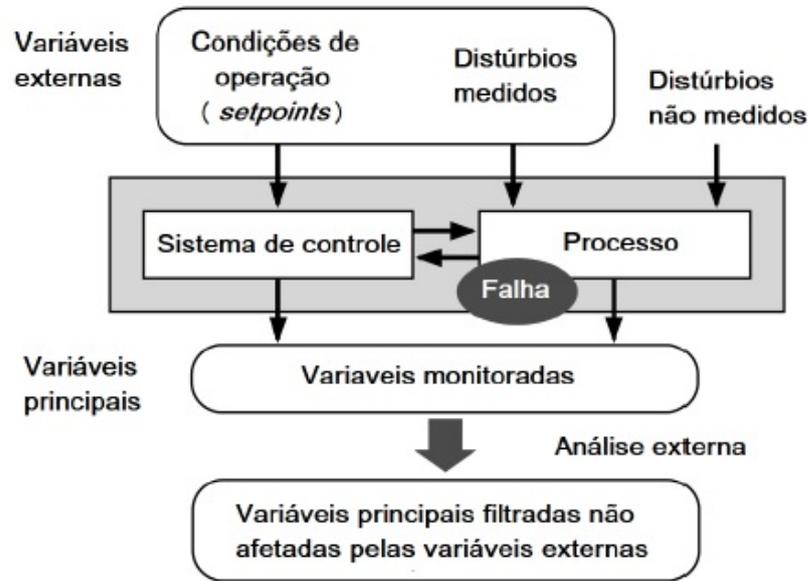
Assim, [Kano et al. \(2004\)](#) propõem divisão das variáveis de processo em duas categorias: as externas, que são as responsáveis pelas condições de operação e distúrbios, e as principais, que são alteradas por essas mudança, são as variáveis monitoradas. Esse processo é ilustrado na Figura 5.

Para que as falhas sejam distinguidas das alterações de ponto de operação, é necessário retirar as influências das variáveis externas sobre as variáveis principais. Considere um conjunto de dados $Z^{n \times m}$, onde m é o número de variáveis e n o número de amostras, decomposto de acordo com a Equação 2.11, onde $[]$ representa a concatenação dos vetores:

$$Z = [Y U] \quad (2.11)$$

em que Y é o conjunto de variáveis principais e U é o conjunto de variáveis externas. A matriz Y deve ser decomposta em duas, parte explicada pelas variáveis externas e parte não explicada, chamada de resíduos. Para isso, [Kano et al. \(2004\)](#) usam um modelo linear

Figura 5 – Descrição do modelo de análise externa



Fonte: Adaptado de Kano et al. (2004)

obtido por mínimos quadrados, utilizando U como entrada, Y como saída e ϵ a componente não explicada por U , conforme Equação 2.12:

$$Y = U\Theta + \epsilon \quad (2.12)$$

onde Θ é estimado pela Equação 2.13

$$\hat{\Theta} = (U^T U)^{-1} U^T Y \quad (2.13)$$

os resíduos, que são os valores reais subtraídos dos estimados, são dados por:

$$E = Y - U \hat{\Theta} \quad (2.14)$$

tais resíduos são usados para obter o modelo PCA e a estatística T^2 de *Hottelling*.

Na maioria das aplicações reais, as relações entre as variáveis não são lineares, portanto, é importante adequar os modelos de análise externa a tais condições de não-linearidade. O modelo mais simples, polinomial de grau 1, é dado pela Equação 2.15:

$$y = a\mathbf{x} + b \quad (2.15)$$

onde a e b são calculados pela Equação 2.13, sendo

$$\hat{\Theta} = [\mathbf{a} \quad \mathbf{b}] \quad (2.16)$$

Acrescentando graus ao polinômio, para permitir a adequação à não-linearidade do sistema, a Equação 2.15 se torna a Equação 2.17.

$$y = \mathbf{b} + \mathbf{a}_0 \mathbf{x}^1 + \mathbf{a}_1 \mathbf{x}^2 + \mathbf{a}_2 \mathbf{x}^3 + \dots + \mathbf{a}_{p-1} \mathbf{x}^p \quad (2.17)$$

onde p é a ordem do polinômio que se deseja utilizar.

Tal representação de não linearidade também pode ser representada por meio de redes neurais, neste trabalho sendo utilizada a *General Regression Neural Network* (GRNN), com descrição e estrutura dada por Wasserman (1993) e Specht et al. (1991), que usa *kernels* Gaussianos para obter uma estimativa da saída desejada por meio de uma ponderação da distância entre os dados de treinamento e a entrada da rede. E também o método de Support Vector Regression (SVR), que se assemelha ao Support Vector Machine (SVM), porém com a diferença de ao invés maximizar a distância das amostras em relação a um hiperplano, ele busca minimizar a distância das amostras em relação a um hiperplano (VAPNIK, 2013).

2.3 Concept Drift

A aplicação da estatística para monitoramento de processos parte do pressuposto de que as características das variáveis do processo são constantes a menos que ocorra uma falha no sistema. De tal forma, é assumido que propriedades estatísticas, como média e desvio padrão, são sempre iguais sob as mesmas condições operacionais (CHIANG; RUSSELL; BRAATZ, 2001), o que para um sistema real nem sempre é aplicável, pois existem variações lentas com o tempo, tais como desgaste e envelhecimento de equipamentos. Desta forma, a menos que o modelo de monitoramento seja atualizado constantemente, existirão alarmes falsos (PORTNOY et al., 2016).

Concept Drift ocorre quando as estatísticas dos dados que estão sendo coletados mudam de tempos em tempos após um período mínimo de estabilidade (KADWE; SURYAWANSHI, 2015). Quando o *concept drift* é detectado, o modelo atual precisa ser atualizado para manter a precisão (KADWE; SURYAWANSHI, 2015).

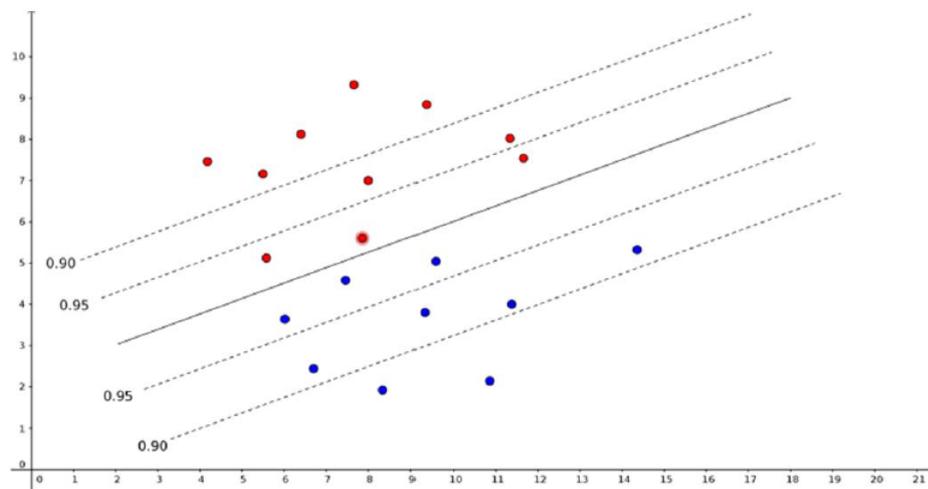
O problema de detecção de falhas pode ser encarado como um problema de classificação, onde dados são rotulados como falha ou situação de normalidade. Os métodos tradicionais de aprendizado de máquina abordam problemas de classificação e previsão com base em duas suposições: (1) dados rotulados representativos estão disponíveis em abundância para descrever os problemas; e (2) as propriedades estatísticas dos dados são similares aos dados de treinamento e não se alteram ao longo do tempo, porém, se tratando de monitoramento de processos industriais é muito difícil que todas as possíveis situações de falhas ou diferentes regiões de operação sejam previamente rotuladas, especialmente por conta do alto custo envolvido, por ser um processo que conta com inúmeras variáveis e situações de imprevisibilidade e que pode contar com o *concept drift* (COSTA; ALBUQUERQUE; SANTOS, 2018).

Para lidar com *concept drift*, muitos métodos foram propostos, que podem ser divididos em dois grupos: métodos cegos e informados (SAYED-MOUCHAWEH, 2016).

Métodos cegos não fornecem um mecanismo explícito para detectar o *concept drift*. Eles atualizam os modelos independentemente da ocorrência de desvios. Nesse caso, geralmente há um alto custo computacional envolvido. Já na segunda abordagem, aplicada por exemplo por Gama e Castillo (2006), Baena-Garcia et al. (2006), Pinage e Santos (2015) e Sethi e Kantardzic (2015), é utilizado um mecanismo para realizar a tarefa de detecção e ajustes de classificadores, porém, ainda sim, a maioria dos métodos desenvolvidos assume que os dados rotulados estarão disponíveis após a classificação, pra poder gerar melhorias e ajustes no desempenho do classificador. Porém, existem diversas situações em que tais rótulos não estão disponíveis e, nestes casos, uma abordagem de Aprendizado Ativo é interessante, por poder detectar explicitamente desvios com pouca interferência humana. Assim, os rótulos verdadeiros das novas instâncias são necessários somente após uma falsa detecção (seja de falha ou de normalidade) e são usadas para atualizar o classificador (COSTA; ALBUQUERQUE; SANTOS, 2018).

Dentre as várias técnicas presentes no aprendizado ativo, a estratégia de Incerteza Fixa mostra-se interessante, devido aos bons resultados já alcançados (ŽLIOBAITĖ et al., 2013). Para usar a estratégia de incerteza fixa na seleção de amostras para detecção de *concept drift* é necessário definir o conceito de margens virtuais. Na proposta de Costa, Albuquerque e Santos (2018) as margens virtuais são interpretadas como a projeção de hiperplanos equidistantes a um hiperplano que modela a relação entre as variáveis, com base em um limiar de incerteza (λ) definido pelo usuário. A Figura 6 mostra margens virtuais definidas para $\lambda = 0,95$ e $\lambda = 0,90$. Essa figura mostra projeções lineares de margens virtuais, no entanto, margens virtuais não lineares também podem ser atingidas.

Figura 6 – Hiperplano de separação e projeção das margens virtuais com limiar de incerteza de 0,95 e 0,90.



Fonte: (COSTA; ALBUQUERQUE; SANTOS, 2018)

A projeção de margens virtuais cria um subespaço do espaço de entrada que é a região de confiança do algoritmo de regressão. Esta região é definida pelo valor de λ .

Quanto maior o valor de λ , mais estreita a região de confiança ao redor do valor previsto pelo modelo de regressão.

Uma vez definidas as margens virtuais, é possível descrever como esse conceito é usado para detectar *drifts*. Esse processo ocorre analisando lotes de dados usando o modelo de regressão escolhido. Se, para um lote de dados, for obtida uma densidade de amostras menor do que um limiar (α), então o modelo não está conseguindo modelar adequadamente o processo e, possivelmente, pode ter acontecido *drift* nos dados (COSTA; ALBUQUERQUE; SANTOS, 2018). O valor da densidade é a porcentagem de amostras do lote que estão dentro da região de confiança.

Neste trabalho os limiares de incerteza serão dados pela estatística T^2 de *Hottelling*, com $\alpha = 0,90$.

2.4 Aprendizado Ativo

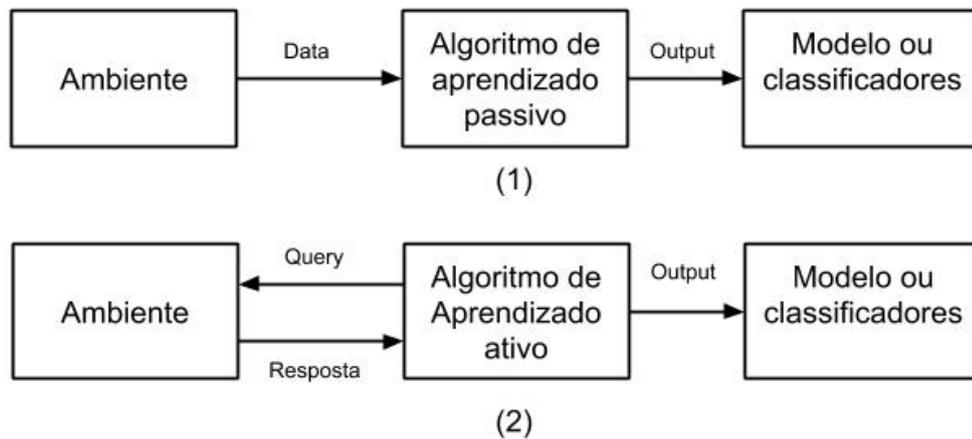
Além dos modelos construídos serem fieis à realidade do processo, o fluxo de dados apresenta desafios adicionais, pois a distribuição estatística dos dados pode mudar ao longo do tempo e os modelos precisam se adaptar, sendo o aprendizado ativo uma alternativa pra esse problema (ŽLIOBAITĖ et al., 2013).

A maioria dos cenários de aprendizado de máquina geralmente se enquadra em uma das duas tarefas de aprendizado: aprendizado supervisionado ou aprendizado não supervisionado (TONG, 2001). No aprendizado supervisionado, o objetivo é aprender uma regra geral que mapeie as entradas para as saídas, um exemplo clássico é a tarefa de classificação. Já no aprendizado não-supervisionado, nenhum tipo de rótulo é dado ao algoritmo de aprendizado, deixando-o sozinho para encontrar a estrutura dos dados fornecidos, um exemplo é a tarefa de clusterização.

Para todas essas tarefas de aprendizado supervisionado e não supervisionado, geralmente é reunida uma quantidade significativa de dados que são amostrados aleatoriamente na distribuição da população e, em seguida, é induzido um classificador ou modelo. Essa metodologia é chamada de aprendizado passivo. No aprendizado ativo, são reunidas informações sobre o mundo fazendo perguntas (*queries*) e recebendo respostas. Em seguida, ele gera um classificador ou modelo, dependendo da tarefa para a qual está sendo usado, como mostra a Figura 7 (TONG, 2001).

A aprendizagem ativa estuda o ciclo de um algoritmo de aprendizado, selecionando ações ou fazendo consultas que influenciam quais dados são adicionados ao seu conjunto de treinamento. A principal motivação para a aprendizagem ativa vem do tempo ou do custo em se obter exemplos de treinamento rotulados. Em alguns domínios, como na modelagem de processos industriais, um único exemplo de treinamento pode exigir vários dias e causar

Figura 7 – Modelo de Aprendizado Passivo (1) Ativo (2)



Fonte: Adaptado de Tong (2001)

muitos prejuízos (SCHOHN; COHN, 2000).

2.5 Clusterização

Clusterização se refere a um conjunto de técnicas para agrupar dados, de forma que as amostras em cada grupo sejam bastante semelhantes entre si, enquanto observações de grupos diferentes sejam diferentes entre si, baseado em determinado critério de similaridade (JAMES et al., 2013).

2.5.1 *K-means*

O algoritmo *k-means* é explicado por Bishop (2006) como descrito a seguir: suponha que exista uma base de dados $\{x_1, x_2, \dots, x_n\}$ que consiste em n observações de uma variável aleatória \mathbf{x} com dimensão D . A ideia é separar a base de dados em em um número K de clusters, de forma que a distância entre os pontos do mesmo grupo seja menor do que a distância aos pontos fora do grupo.

Desta forma, é introduzido um conjunto de vetores $\boldsymbol{\mu}_k$, onde $k = 1, \dots, K$, e cada $\boldsymbol{\mu}_k$ é o centro do k -ésimo cluster. Deve-se, então, encontrar o conjunto de observações pertencente a cada cluster, bem como um conjunto de vetores $\{\boldsymbol{\mu}_k\}$, de forma que o quadrado da distância (distância Euclidiana) de cada observação ao seu vetor $\boldsymbol{\mu}_k$ correspondente seja o menor entre o conjunto de vetores.

Para cada amostra da base de dados x_n da base de dados é introduzido um conjunto correspondente de variáveis binárias $r_{nk} \in \{0, 1\}$, onde $k = 1, \dots, K$ descreve a qual cluster k a amostra x_n pertence, onde se x_n pertence ao cluster k , $r_{nk} = 1$ e $r_{nj} = 0$ para $j \neq k$.

A função objetivo (a ser minimizada) é dada pela Equação 2.18:

$$J = \sum_{n=1}^N \sum_{k=1}^K r_{nk} \|x_n - \boldsymbol{\mu}_k\|^2 \quad (2.18)$$

que representa a soma do quadrado das distâncias de cada ponto ao seu respectivo vetor $\boldsymbol{\mu}_k$. Deve-se encontrar os valores de $\{\boldsymbol{\mu}_k\}$ e $\{r_{nk}\}$ que minimizam J . Isso pode ser feito por meio de um processo iterativo, onde cada iteração envolve dois passos, no primeiro passo é feita a minimização de J com relação a r_{nk} , mantendo $\{\boldsymbol{\mu}_k\}$ fixo. No segundo passo é feita a minimização de J com relação a $\{\boldsymbol{\mu}_k\}$, mantendo r_{nk} fixo. Isso é feito até a convergência.

Considerando primeiramente a determinação de r_{nk} podemos expressar da seguinte forma (Equação 2.19):

$$r_{nk} = \begin{cases} 1 & \text{se } k = \operatorname{argmin}_j \|x_n - \boldsymbol{\mu}_j\|^2 \\ 0 & \text{caso contrário} \end{cases} \quad (2.19)$$

Agora otimiza-se $\boldsymbol{\mu}_k$ com r_{nk} fixo. A função objetivo J é uma função quadrática de $\boldsymbol{\mu}_k$, logo, pode ser otimizada por meio de sua derivada igualada a zero, como mostra a Equação 2.20

$$2 \sum_{n=1}^N r_{nk} (x_n - \boldsymbol{\mu}_k) = 0 \quad (2.20)$$

cujas soluções são dadas por 2.21

$$\boldsymbol{\mu}_k = \frac{\sum_{n=1}^N r_{nk} x_n}{\sum_{n=1}^N r_{nk}} \quad (2.21)$$

onde percebe-se facilmente que o valor de $\boldsymbol{\mu}_k$ é dado pelo valor da média dos dados x_n pertencentes ao cluster k .

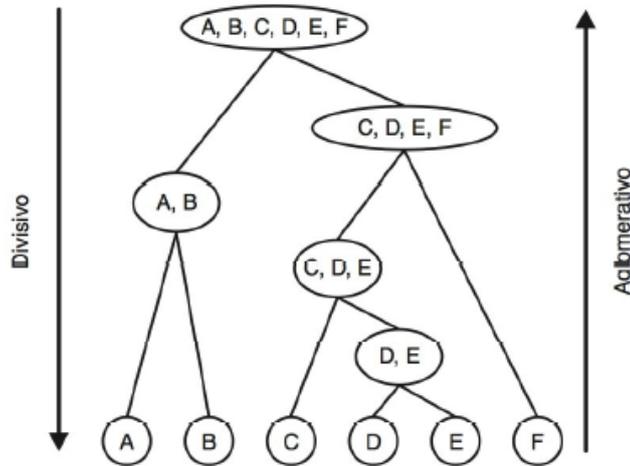
Os passos são repetidos até que não haja mais mudanças de atribuição de clusters para os dados (ou quando seja atingido um número máximo de iterações pré-determinado).

2.5.2 Clusterização Hierárquica

Como é descrito por Faceli et al. (2011), um algoritmo de agrupamento (clusterização) hierárquico gera, a partir de uma matriz de proximidade, uma sequência de partições alinhadas. O algoritmo pode ser dividido em dois tipos: o aglomerativo (com uma estratégia *bottom-up*), que começa com n grupos, onde cada amostra é o seu próprio *cluster* e forma uma sequência de passos agrupando tais grupos sucessivamente, e a divisiva (com uma estratégia *top-down*), que começa com um único *cluster* contendo todos os elementos e forma uma sequência dividindo os *clusters* sucessivamente, como mostrado na Figura 8.

Para agrupar ou dividir os *clusters*, cada algoritmo considera uma das alternativas de distância/similaridade dadas pelas métricas de integração (*linkage metrics*), que estão

Figura 8 – Funcionamentos dos algoritmos hierárquicos aglomerativos e divisivos.



Fonte: (FACELI et al., 2011)

dentre as abordagens mais clássicas. Esses algoritmos não possuem função objetivo global, são baseados em decisões locais.

Dados dois *clusters* $C_1 = \{x_{11}, x_{21}, \dots, x_{n1}\}$ e $C_2 = \{x_{12}, x_{22}, \dots, x_{n2}\}$, com os respectivos centroides $\bar{x}^{(1)}$ e $\bar{x}^{(2)}$, para medir a distância entre os *clusters*, podem ser utilizadas métricas como a distância euclidiana ou de Manhattan entre os centroides dos *clusters*, $d(\bar{x}^{(1)}, \bar{x}^{(2)})$. Deve-se calcular a distância entre os possíveis objetos (x_i, x_j) sendo cada um pertencente a um dos clusters (C_1 e C_2), e depois aplicar uma operação, como a média, mínimo ou máximo, para traduzir essas distâncias entre os elementos nas distâncias dos clusters. Cada uma dessas distâncias entre os grupos é uma métrica de integração.

A Equação 2.22 ilustra a distância entre os grupos, que é dada pela distância mínima entre quaisquer dois objetos, um de cada cluster. Essa distância é utilizada pelo algoritmo hierárquico com distância mínima (*single-link*). Já na Equação 2.23 é apresentada a distância média (*average-link*) entre os pontos dos clusters e na Equação 2.24 é demonstrada a distância máxima (*complete-link*), que utiliza os elementos mais distantes.

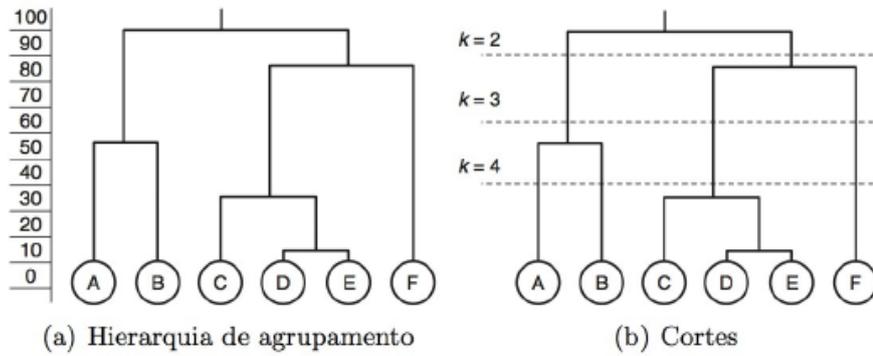
$$d(C_1, C_2) = \min_{x_i \in C_1, x_j \in C_2} d(x_i, x_j) \quad (2.22)$$

$$d(C_1, C_2) = \frac{1}{n_1 n_2} \sum_{x_i \in C_1, x_j \in C_2} d(x_i, x_j) \quad (2.23)$$

$$d(C_1, C_2) = \max_{x_i \in C_1, x_j \in C_2} d(x_i, x_j) \quad (2.24)$$

Geralmente as soluções são ilustradas por dendogramas como mostra a Figura 9(a), onde o conjunto de dados era formado por $\{A, B, C, D, E, F\}$. A altura das ramificações,

Figura 9 – Exemplo de dendograma.



Fonte: (FACELI et al., 2011)

em geral, é proporcional à distância entre os grupos. Cortes em diferentes níveis representam diferentes números de *clusters*. Na Figura 9(b) os cortes representam as seguintes partições:

- $\{A, B\}, \{C, D, E, F\}$, com $k = 2$;
- $\{A, B\}, \{C, D, E\}, \{F\}$, com $k = 3$;
- $\{A\}, \{B\}, \{C, D, E\}, \{F\}$, com $k = 4$;

Esse algoritmo não requer a especificação prévia do número de clusters, mas ao mesmo tempo são mais sensíveis a ruídos e outliers.

3 Metodologia Proposta

Como proposto por [Kano et al. \(2004\)](#), serão utilizados modelos de análise externa de variáveis para monitoramento de um processo industrial. Os modelos de análise externa que serão utilizados são tanto lineares, quanto não-lineares. Aliada ao método de análise externa, também será aplicada a técnica PCA e posterior aplicação da estatística T^2 de *Hotelling* para detecção de possíveis pontos de anormalidade de operação. Os limiares obtidos com tais técnicas serão utilizados como limiares de incerteza (*Fixed Uncertainty*) e o rótulo verdadeiro da amostra será solicitado para o operador se o mesmo for ultrapassado, comprovando se a anormalidade representa ou não uma falha.

Primeiramente foi feito o monitoramento dos dados de teste, sem atualizações dos modelos, de forma que uma vez gerados, eles permanecem estáticos.

Sabe-se que utilizar um modelo estático de monitoramento só se aplica nos casos em que a distribuição estatística das variáveis permanece sempre a mesma. No entanto, processos reais geralmente possuem comportamentos variáveis no tempo, por conta, por exemplo, de troca de equipamentos, desgastes de sensores, troca de fornecedores de matéria-prima, manutenção e limpeza. Muitos falsos alarmes geralmente ocorrem quando um modelo invariante no tempo é utilizado para monitorar tais processos. Para melhorar a robustez do monitoramento é essencial uma estratégia adaptativa ([ZHANG, 2000](#)).

Sendo assim, como proposto por [Costa, Albuquerque e Santos \(2018\)](#), os modelos serão atualizados baseados na detecção de *concept drift*, através de um método de aprendizado ativo em que é utilizada a variação de densidade dos dados como um gatilho para detecção de *concept drift*, como discutido no Capítulo 2. Essa variação de densidade é calculada a partir das margens virtuais projetadas no espaço (limiares estatísticos), ou seja, é calculado um limiar estatístico para os dados, e se a porcentagem de dados (densidade dos dados) fora desse limiar for superior a um valor estabelecido, o *drift* é detectado.

É importante ressaltar que como mais de uma variável será monitorada pela análise externa, os resíduos do monitoramento também serão compostos de mais de um conjunto de dados, por isso, é necessária a utilização da PCA e da estatística T^2 , de forma a obter-se apenas um limiar e um conjunto de monitoramento.

Além disso, como proposto por [Sobhani e Beigy \(2011\)](#), um método apropriado para problemas reais sujeitos a *concept drift* deve, além de se adaptar às novas distribuições de variáveis, também ser capaz de fazê-lo sem a necessidade de armazenar todos os dados. Desta forma, foi incorporada à metodologia a clusterização dos dados para inclusão nos modelos de treinamento.

Então, três cenários estão sendo criados para testar a metodologia proposta:

1. Monitoramento estático do processo;
2. Monitoramento dinâmico com uso de *Concept Drift*;
3. Monitoramento dinâmico com uso de *Concept Drift* e clusterização de dados;

Todos os modelos propostos seguiram as mesmas etapas iniciais, descritas a seguir.

3.1 Seleção de variáveis

Para criar um sistema de monitoramento estático de processo, primeiramente fez-se a seleção das variáveis a serem utilizadas. As variáveis externas e principais foram selecionadas unindo-se duas abordagens: a entrevista com o operador da planta, que indicou quais eram fundamentais baseado em sua experiência e a consulta nos trabalhos de que obtiveram bons resultados com os dados escolhidos (ASSIS, 2018; FARIA, 2018).

3.2 Treinamento

Após a seleção de variáveis, os dados de treinamento foram selecionados seguindo a metodologia explicada pela Figura 10.

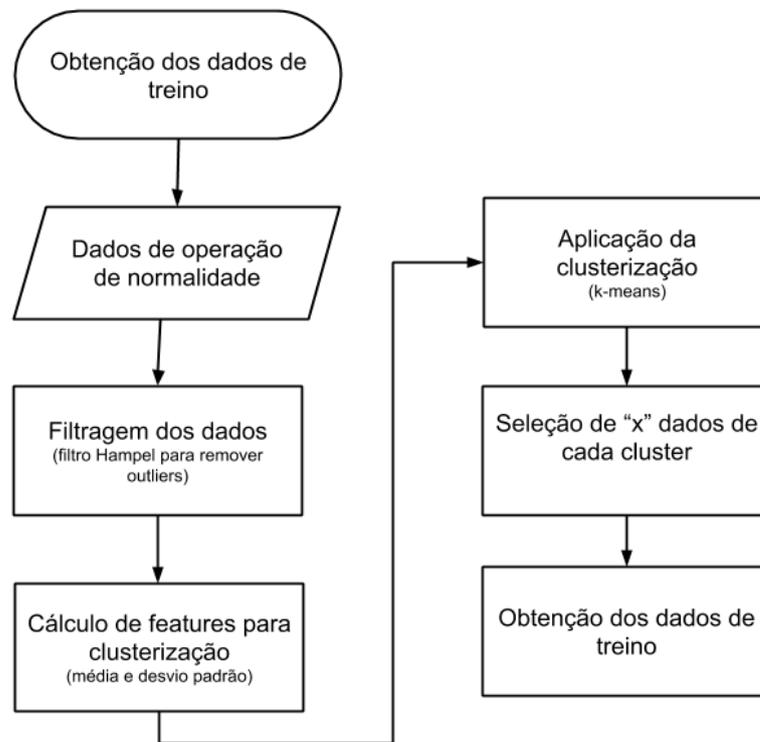
Os dados de treinamento foram retirados do conjunto de dados de normalidade de operação, ou seja, onde não existem falhas. Após uma seleção aleatória desses dados, empiricamente chegou-se a um tamanho de amostra ideal para a resolução do estudo de caso.

Após a seleção dos dados de treinamento, os mesmos passaram por uma filtragem para retirar *outliers*. O filtro utilizado foi o Hampel (PEARSON et al., 2015), que se aplica da seguinte forma: para cada amostra de x , a função calcula a mediana de uma janela composta pela amostra e suas k amostras circundantes, $\frac{x}{2}$ por lado. Ele também estima o desvio padrão de cada amostra em relação à sua mediana da janela usando o desvio absoluto médio. Se uma amostra diferir da mediana em mais de três desvios padrão, ela é substituída pela mediana.

Após a filtragem, os dados passaram pela clusterização utilizando o *k-means*, de forma a agrupar conjuntos de dados que tivessem médias e desvios padrões similares, resultando em x clusters que representem os pontos de operação dos dados selecionados.

Após a seleção e tratamento dos dados para treino, como descrito anteriormente, os mesmos geraram os modelos de análise externa que serão utilizados para as etapas de validação e teste (monitoramento) descritas a seguir.

Figura 10 – Seleção de dados de treino.



Fonte: Autora

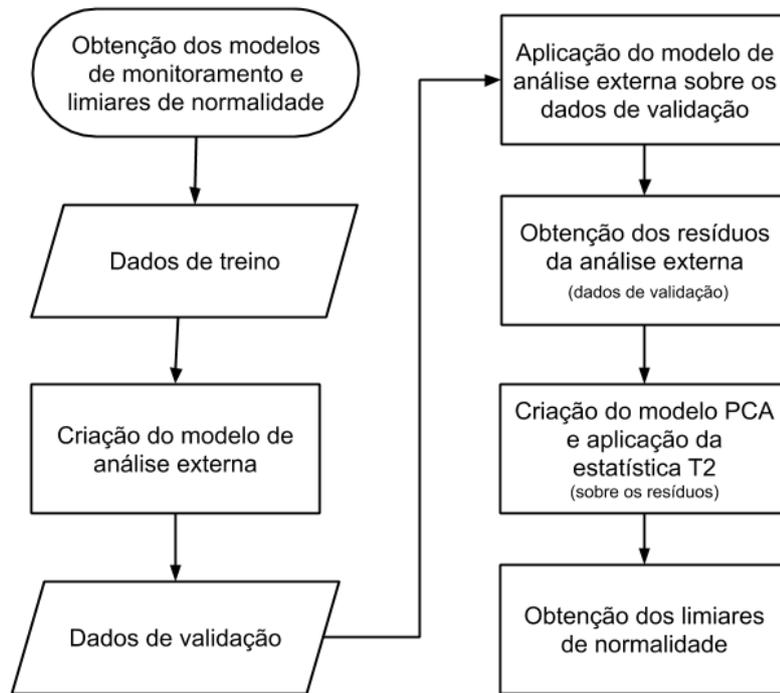
3.3 Validação

Após os dados de treino selecionados, foi feita a seleção dos dados de validação, os quais determinam os limites e operação de normalidade. Tais dados foram separados do conjunto de teste (e posteriormente retirados dos mesmos, para não haver dados de validação no conjunto de teste) e também passaram por filtragem para retirada de possíveis *outliers* que pudessem interferir na criação dos limites.

A Figura 11 demonstra tanto o momento de obtenção dos modelos de análise externa, quanto a aplicação dos mesmos para a obtenção dos limites de operação normal.

Nesse momento, os dados de validação são aplicados ao modelo gerado anteriormente pelo treinamento, assim, obtém-se os resíduos da análise externa feita sob os dados de validação, onde são aplicadas as técnicas PCA e a estatística T^2 , sendo este último valor obtido pelo método estatístico, que será utilizado como limite de operação.

Figura 11 – Criação do modelo e limiares de monitoramento.



Fonte: Autora

3.4 Teste

Tendo obtido os modelos de análise externa, PCA e limiares de normalidade, os dados do processo (dados de teste) são analisados em pacotes (*batches*) de forma sequencial, simulando um monitoramento em tempo real (*online*) dos dados.

Para cada um dos três cenários testados, o monitoramento dos dados de teste é tratado de forma diferente.

3.4.1 Monitoramento estático do processo

No monitoramento estático, os possíveis pontos de anormalidade (nesse caso *drifts*) são detectados mediante a análise de cada pacote: se a quantidade de pontos fora do limiar T^2 for superior a 10% (uma densidade de pontos dentro do limiar menor que 90%), é detectado um *drift*. Lembrando que, para todos os cenários aqui tratados uma ultrapassagem apenas é contabilizada se ela acontecer 5 vezes seguidas, para evitar que erros de medição influenciem nos resultados.

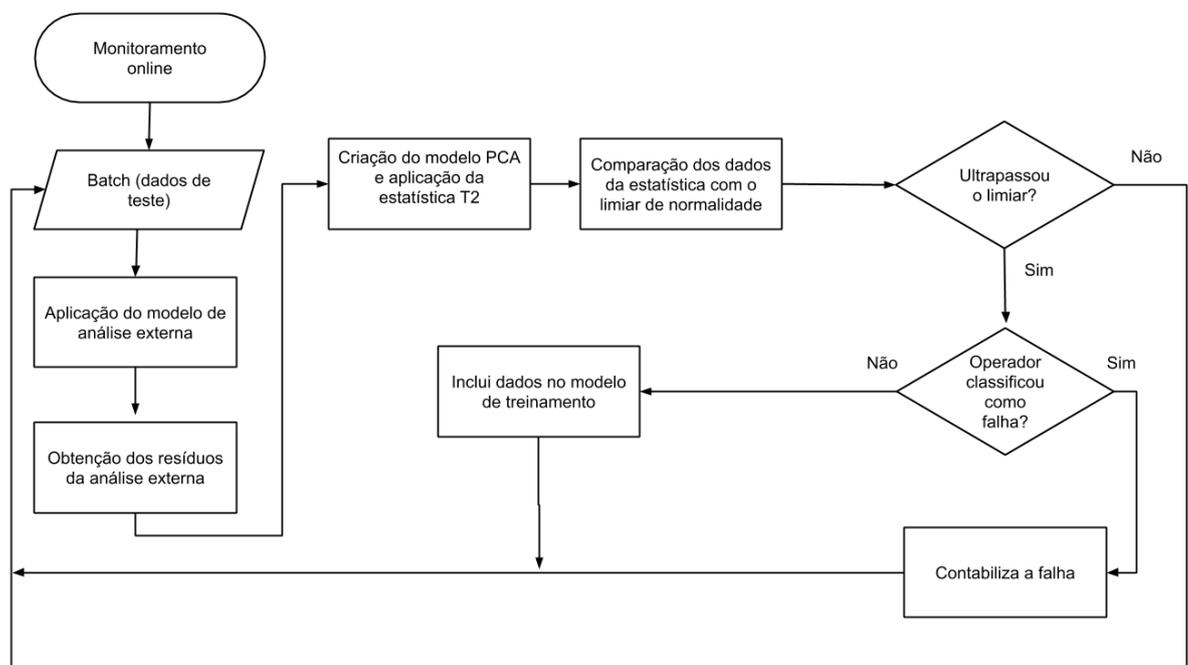
Neste modelo, os *drift* são apenas contabilizados, mas os modelos não são atuali-

zados.

3.4.2 Monitoramento dinâmico com uso de *Concept Drift*

No monitoramento dinâmico, os modelos são atualizados. Na análise de cada pacote, se a quantidade de pontos fora do limiar T^2 for superior a 10% (uma densidade de pontos dentro do limiar menor que 90%), é detectado um *drift* e o operador é alertado. Se o operador identificar como normalidade, os dados são incluídos no modelo de treinamento. O modelo de variáveis externas é retreinado usando os dados originais de treinamento com os novos dados obtidos. E segue-se com o monitoramento, como mostra a Figura 12.

Figura 12 – Fluxograma do monitoramento dinâmico com uso de *Concept Drift*.



Fonte: Autora

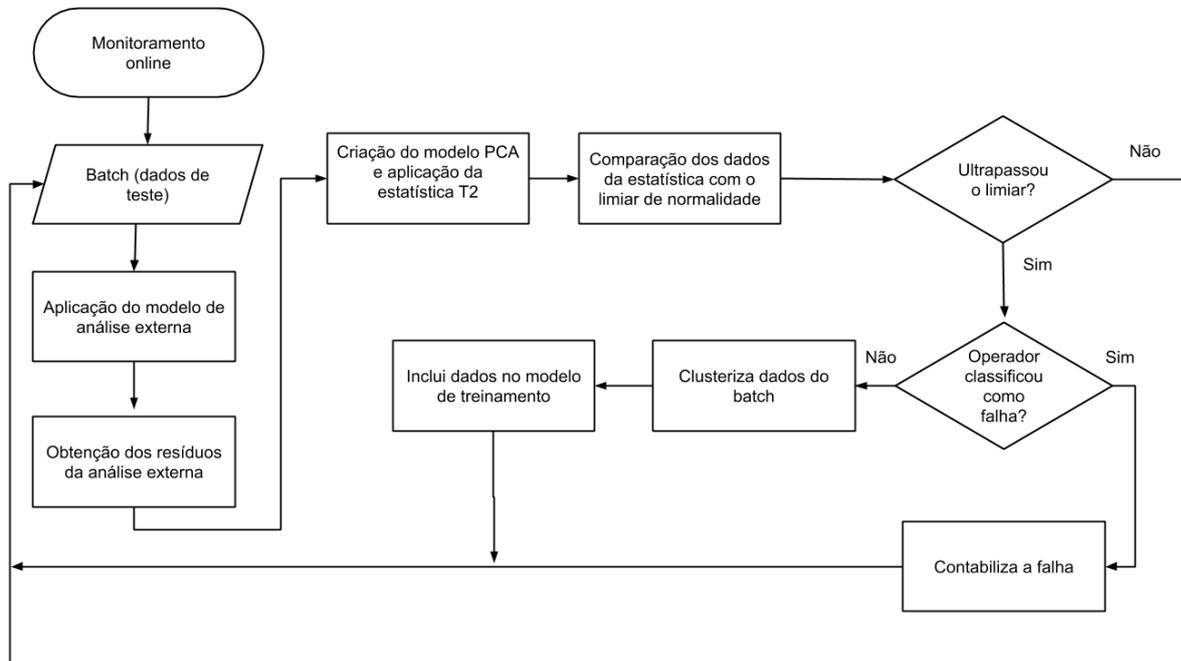
Entretanto, essa solução pode ser não-escalável com o tempo, dado que os dados de treino podem aumentar indefinidamente.

3.4.3 Monitoramento dinâmico com uso de *Concept Drift* e clusterização de dados

A fim de solucionar o problema de crescimento indefinido dos dados de treino, propõe-se considerar apenas alguns pontos de interesse do pacote analisado. Para isso, é utilizado o algoritmo de clusterização hierárquica que agrupa as instâncias em diferentes grupos de regiões de operação, a fim de selecionar apenas um conjunto reduzido de dados

de cada região de operação, o que reduz a quantidade de dados usados para treinamento, como mostra a Figura 13.

Figura 13 – Fluxograma do monitoramento dinâmico com uso de *Concept Drift* e clusterização



Fonte: Autora

É importante frisar que neste caso os dados de treino do modelo também passaram por mais uma etapa de clusterização, e foram diminuídos, de forma que os dados que fossem acrescentados ao longo do monitoramento tivessem um impacto significativo no modelo, para esta etapa também foi utilizado o *k-means*.

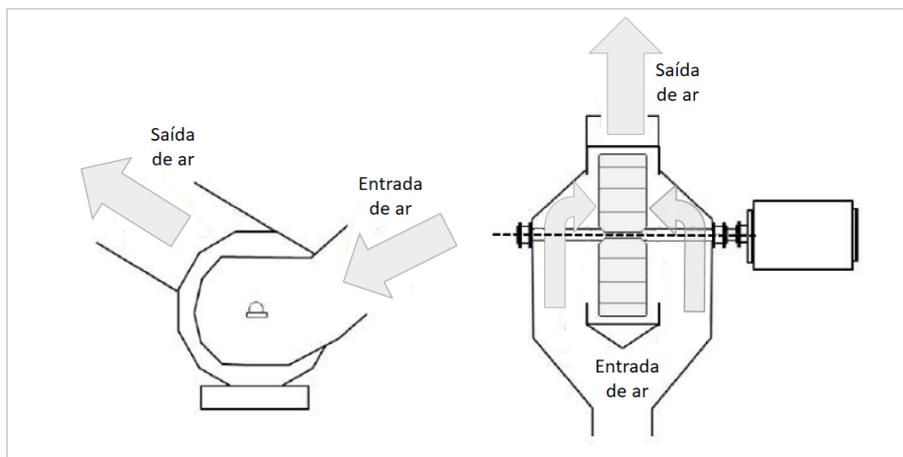
Durante o monitoramento é utilizada a clusterização hierárquica e não o *k-means*, como na etapa de treino, porque utilizar o *k-means* seria muito custoso e demorado, e como se trata de um monitoramento *online*, o desempenho do algoritmo de clusterização é fundamental.

a etapa de homogeneização, é feita uma filtragem, o pelotamento e o peneiramento do material, até que o mesmo chegue na etapa de queima. O processo de queima é de suma importância para que ocorra o aumento da resistência mecânica do produto para atender o processo posterior de geração de aço em siderúrgicas.

O forno de pelotização é do tipo grelha móvel que recebe uma camada de pelotas e se movimenta ao longo do forno, passando por grupos de vários queimadores de combustíveis. A queima das pelotas é realizada através de fluxos de gases quentes que passam entre as pelotas, que são gerados por grandes ventiladores de processo (FARIA, 2018).

Este trabalho tem como objeto de estudo os ventiladores presentes no processo de queima. Tais ventiladores são grandes máquinas, que possuem acionamento por inversores de frequência e motores elétricos de alta tensão, na faixa de 2-8MW de potência. Os ventiladores estudados são como os mostrados na Figura 15.

Figura 15 – Esquema de um ventilador industrial.



Fonte: Adaptado de Faria (2018)

4.1.1 Situação de normalidade

No primeiro momento será tratada uma situação de normalidade do forno, em que não existem falhas, com o objetivo de observar o número de falsos positivos. A metodologia foi aplicada ao ventilador de processo PF01 de uma usina de pelotização localizada em Vitória-ES, da mineradora Vale S. A.

Foram utilizadas 2000 observações multivariadas, com uma taxa de amostragem de 1 segundo. O monitoramento foi feito durante um mês de operação (cerca de 700 horas), equivalente a 2,5 milhões de observações.

Das de 600 variáveis medidas no forno de pelletização, foram selecionadas 8, de acordo com a metodologia descrita no Capítulo 3. A divisão das variáveis entre principais e externas foi feita com base nos trabalhos de Faria (2018) e Assis (2018). Tais variáveis monitoradas se encontram na Tabela 1.

Tabela 1 – Variáveis do Modelo.

Classe	Variáveis
Variáveis principais	Vibração do mancal acoplado ao ventilador (LA) [mm/s ²] Vibração do mancal não acoplado ao ventilador (LNA) [mm/s ²] Vibração da carcaça do ventilador [mm/s ²]
Variáveis externas	Nível de produção [Kg/h] Abertura dos <i>dumpers</i> [%] Rotação do motor [RPM] Potência do motor [KW] Torque do motor [N.m]

Parte das variáveis (excluindo-se a abertura dos *dumpers* que é dada em porcentagem), estão mostradas na Figura 16.

Para o ventilador industrial, variou-se os parâmetros da análise para testar o modelo que teria o melhor desempenho em um caso de aplicação real. Os parâmetros variados foram:

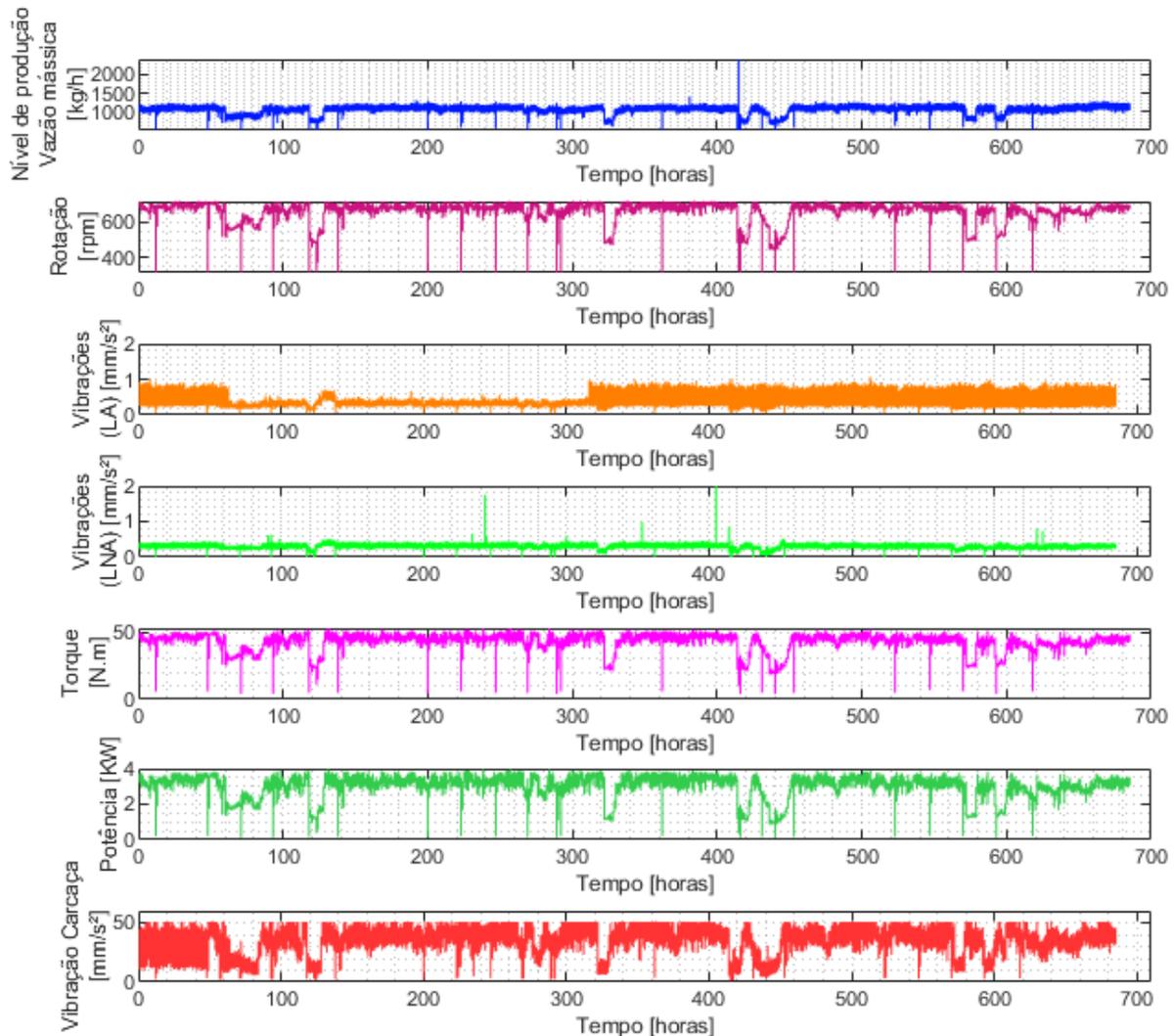
- Quantidade de dias de monitoramento: 10 ou 30;
- Modelos de análise externa: modelos polinomiais, GRNN e SVR;
- Utilização do aprendizado ativo ou não;
- Utilizar a clusterização de dados ou não (com a metodologia hierárquica);

Os resultados de monitoramento foram avaliados usando as seguintes métricas:

- Falsos positivos: quantidade percentual de vezes que os dados ultrapassaram o limiar (considerando um falso positivo quando o limiar é ultrapassado 5 vezes seguidas);
- Nº de alarmes: representa a quantidade de vezes que o sistema alarmou, ou seja, que possivelmente ocorreu um *drift* nos dados. Todo alarme precisa de uma interação com um especialista;
- Ajuste ao modelo (*FIT*): indica o quanto o modelo construído se ajusta aos dados reais, sendo indicada pela Equação 4.1:

$$FIT = 100 * \left(1 - \left(\frac{\|r\|}{\|t\|} \right) \right) \quad (4.1)$$

Figura 16 – Variáveis de monitoramento: rotação, produção, vibrações LA e LNA e de carcaça, potência e torque.



Fonte: Autora

em que r são os resíduos do modelo e t são os dados reais de teste. Quanto maior o valor, maior o ajuste do modelo aos dados (como são monitorados 3 modelos de análise externa, o valor indicado é a média dos três);

- Aumento dos dados de treino: quantidade de pontos incluídos no pacote de treinamento, dadas as iterações do operador;
- Tempo de simulação: tempo total de simulação, em segundos, de execução do algoritmo sobre os dados de teste.

Em condições de operação normal, quanto menor o número de falsos positivos, menor

número de alarmes, menor aumento de dados de treino e maior *FIT*, melhor o sistema.

Na Tabela 2 são exibidos os resultados dos modelos sem atualização, como descrito na Seção 3.4.1 (Monitoramento Estático do Processo), para 30 dias de dados de monitoramento. Os melhores resultados para as métricas estão em negrito.

Tabela 2 – Resultados (30 dias) sem atualização de modelo - Dados de normalidade.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)
Polinômio de grau 1	9	81,03%	0,18%	221,57
Polinômio de grau 2	6	79,48%	0,22%	213,65
Polinômio de grau 3	4	35,89%	0,32%	284,2
GRNN	33	72,05%	0,34%	1429.54
SVR	20	69,49%	0,20%	821.44

Combinando-se todas as métricas, o modelo “Polinômio de Grau 2” apresenta o melhor resultado deste monitoramento, apresentando 1 alarme a cada 5 dias. Já o modelo “GRNN” apresenta o pior resultado, alarmando a uma taxa de 1,1 vezes por dia no período tratado.

Os resultados para 30 dias de dados de monitoramento, com os modelos sendo atualizados, seguindo a metodologia proposta na Seção 3.4.2 (Monitoramento Dinâmico com uso de *Concept Drift*) são apresentados na Tabela 3. Os melhores resultados para as métricas estão em negrito.

Tabela 3 – Resultados (30 dias) com atualização de modelo - Dados de normalidade.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)	Dados de treino inseridos
Polinômio de grau 1	7	83,09%	0,17%	119.15	7007
Polinômio de grau 2	7	75,83%	0,26%	122.67	7007
Polinômio de grau 3	5	48,6%	0,23%	120.65	5005
GRNN	11	76,18%	0,21%	4449.15	9009
SVR	22	63,54%	0,22%	4975.56	22022

Já neste caso, é visível que o monitoramento teve melhoras para alguns modelos, como o “Polinômio de Grau 1” e “GRNN”, mas que teve uma ligeira piora na acurácia dos demais. E além disso, os dados de treino foram incrementados significativamente, o que a longo prazo não é sustentável.

Por sua vez, os resultados para 30 dias de dados de monitoramento, com os modelos sendo atualizados e usando o método de clusterização de dados, seguindo a metodologia proposta na Seção 3.4.3 (Monitoramento Dinâmico com uso de *Concept Drift* com Clusterização de Dados) são apresentados na Tabela 4. Os melhores resultados para as métricas estão em negrito. Lembrando que neste caso os dados de treino passaram por outra etapa de clusterização anterior ao início do monitoramento, e foram reduzidos para 200.

Tabela 4 – Resultados (30 dias) com atualização de modelo e clusterização - Dados de normalidade.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)	Dados de treino inseridos
Polinômio de grau 1	6	81,5%	0,18%	410,3	12
Polinômio de grau 2	4	76,89%	0,19%	208,0	8
Polinômio de grau 3	4	50,36%	0,40%	387,0	8
GRNN	13	70,06%	0,27%	634,7	26
SVR	19	69,21%	20,21%	338,9	38

Já neste caso, percebe-se que todos os modelos tiveram melhoras tanto em alarmar menos vezes, quanto no número de dados de treinamento, que aumentou muito pouco, tornando o modelo sustentável por longos períodos de tempo.

Para 10 dias de monitoramento, ainda seguindo a metodologia proposta na Seção 3.4.2 (Monitoramento dinâmico com uso de *Concept Drift*), tem-se os resultados mostrados na Tabela 5. Os melhores resultados para as métricas estão em negrito.

Tabela 5 – Resultados (10 dias) com atualização de modelo- Dados de normalidade.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)	Dados de treino inseridos
Polinômio de grau 1	3	77,60%	0,24%	45,60	3003
Polinômio de grau 2	3	77,98%	0,40%	51,67	3003
Polinômio de grau 3	1	34,86%	0,29%	57,7	1001
GRNN	6	69,77%	0,46%	2165,3	6006
SVR	9	64,71%	0,42%	1707,18	9009

Para 10 dias de monitoramento, ainda seguindo a metodologia proposta na Seção 3.4.3 (Monitoramento dinâmico com uso de *Concept Drift* com clusterização de dados), tem-se os resultados mostrados na Tabela 6. Os melhores resultados para as métricas estão em negrito.

Tabela 6 – Resultados (10 dias) com atualização de modelo e clusterização de dados - Dados de normalidade.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)	Dados de treino inseridos
Polinômio de grau 1	2	78,55%	0,24%	37,4	4
Polinômio de grau 2	3	71,15%	0,15%	41,56	6
Polinômio de grau 3	1	40,86%	0,28%	59,7	2
GRNN	6	70,33%	0,48%	319,8	12
SVR	8	67,22%	0,37%	100.1	16

Como pode ser percebido, com um terço dos dias de monitoramento, o algoritmo alarmou menos vezes, tanto na metodologia com clusterização, quanto na que não possui clusterização. O que demonstra que, de fato, os alarmes acabam sendo proporcionais aos dias de monitoramento e são distribuídos de forma uniforme entre os dias. O algoritmo não demonstra picos de alarmes em situações de mudanças de ponto de operação.

Pode-se observar que o método de análise externa, com modelo polinomial de grau 2 com inclusão de pacotes para treino através de clusterização “Polinômio de Grau 2 - com clusterização” retorna o melhor resultado médio para as diferentes métricas: possui um baixo incremento na quantidade de dados armazenados (cerca de centenas de vezes menor do que os modelos sem a clusterização), o segundo menor índice de falsos positivos, um número baixo de alarmes e um ajuste aos dados (*FIT*) razoável.

A rede neural GRNN e a SVR não apresentaram bons resultados, apresentando um número alto de alarmes.

O modelo “Polinômio de Grau 2- com clusterização” deu 4 alarmes em cerca de 700 horas de operação, aproximadamente um alarme a cada 7,5 dias, em contraste ao modelo “Polinômio de Grau 1 - sem atualização”, que fornecia, em média, uma alarme a cada 3,33 dias. Isto realça a importância de sistemas que possam ser atualizados ao longo do tempo.

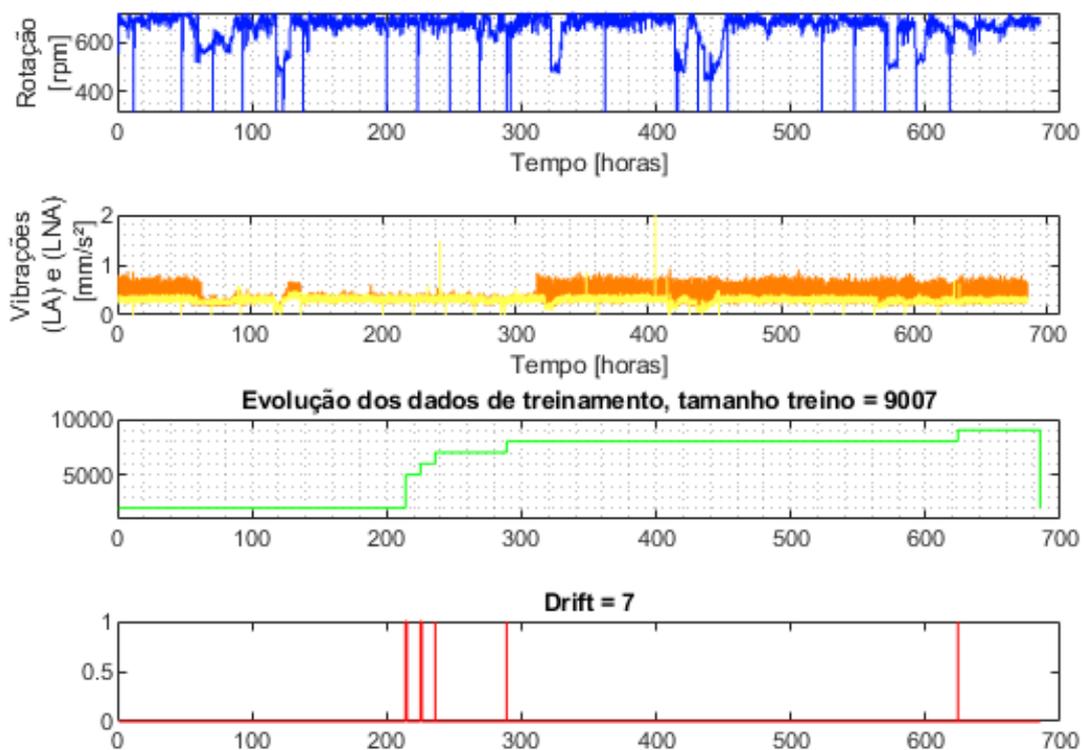
Também é importante ressaltar que o gatilho para atualização dos modelos ser a detecção de *drifts* torna o modelo mais robusto do que se fosse simplesmente utilizado, por exemplo, a ocorrência de um falso positivo para a atualização, porque mesmo com os falsos positivos sendo muito baixos, a menor taxa encontrada foi de 0,18%, o que ainda representa $4,65 \times 10^3$ pontos de dados.

Um outro ponto de atenção aos resultados é a análise do tempo de monitoramento. O maior valor de tempo encontrado foi de 4975,56 segundos, o que representa aproximadamente 83 minutos gastos para o monitoramento de um mês de operação (que contém aproximadamente 720 horas). Isto mostra que o algoritmo possui capacidade de operar em tempo real, dado que o tempo de processamento de cada dado é menor do que o tempo de

chegada de novos dados.

Na Figura 17 é apresentado o comportamento de algumas variáveis e o momento em que os *drifts* foram detectados. Para o modelo ‘Polinômio de Grau 2 - sem atualização’ é evidente que os momentos de detecção de *drifts* não ocorreram com as mudanças de ponto de operação, o que mostra a robustez do modelo à tais mudanças. Uma observação importante é que nos dois primeiros momentos de detecção, ocorreram dois consecutivos.

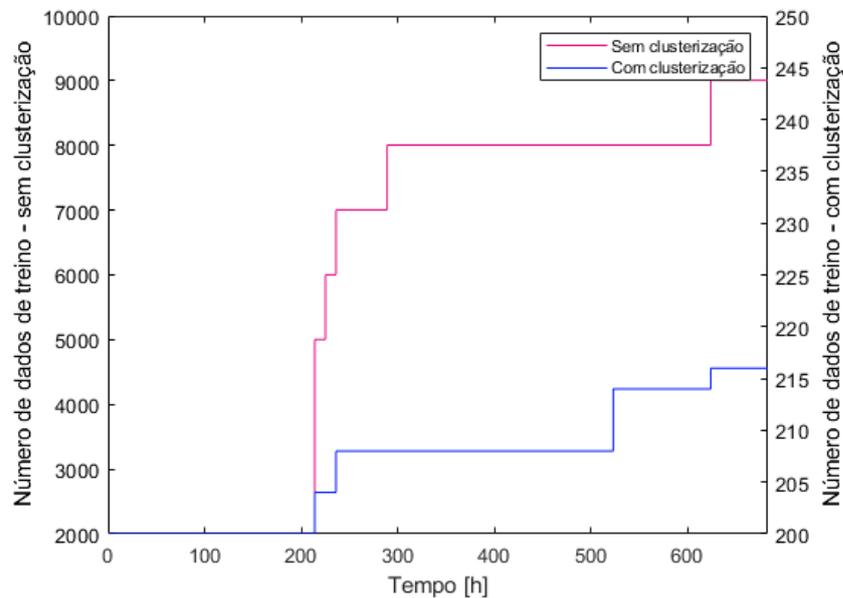
Figura 17 – Detalhamento do momento em que os *drift* são detectados no modelo “Polinômio de grau 2 - sem clusterização”.



Fonte: Autora

Para fins de comparação dos modelos com e sem clusterização, na Figura 18 são mostrados os momentos em que cada alarme foi detectado e, conseqüentemente o momento em que mais dados de treino foram incluídos aos modelos de 2º Grau (com e sem clusterização). Devido a diferença de magnitude, o eixo vertical à esquerda é referente ao gráfico do modelo sem clusterização, enquanto o eixo à direita é referente ao gráfico do modelo com clusterização. Pode ser observada na figura a diferença entre a quantidade de dados de treino entre os modelos. O modelo com clusterização tem uma taxa de crescimento bem mais reduzida ao longo do tempo.

Figura 18 – Comparação dos momentos de adição de dados de treino para o modelo polinomial de 2º Grau.



Fonte: Autora

4.1.2 Situação de falha

Para testar se o algoritmo possui a capacidade de identificar de forma assertiva situações de falha, aplicou-se a metodologia à uma situação real de falha que também ocorreu em um forno industrial. Especificamente no ventilador industrial 5Q7VT de uma usina de pelletização localizada em Vitória-ES, da mineradora Vale S. A. O tempo para reparo deste equipamento gerou 210,23 horas de parada de produção da usina.

O tempo de amostragem é de 10 segundos. Para treino foram selecionadas 16 horas de operação antes do início da falha. E por se tratar de uma aplicação em caso real, só existe a falha em um ponto de operação.

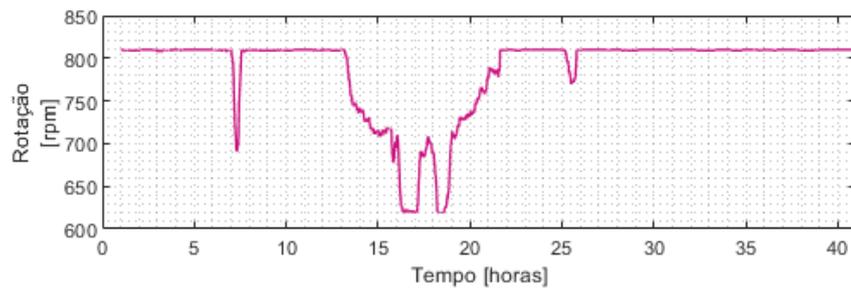
Para o caso deste ventilador, as variáveis disponíveis para análise eram outras, sendo assim, a seleção de variáveis externas e principais foi modificada, como pode-se perceber na Tabela 7.

Tabela 7 – Variáveis do Modelo.

Classe	Variáveis
Variáveis principais	Vibração do mancal acoplado ao ventilador (LA) [mm/s ²] Vibração do mancal não acoplado ao ventilador (LNA) [mm/s ²]
Variáveis externas	Rotação do motor [RPM]

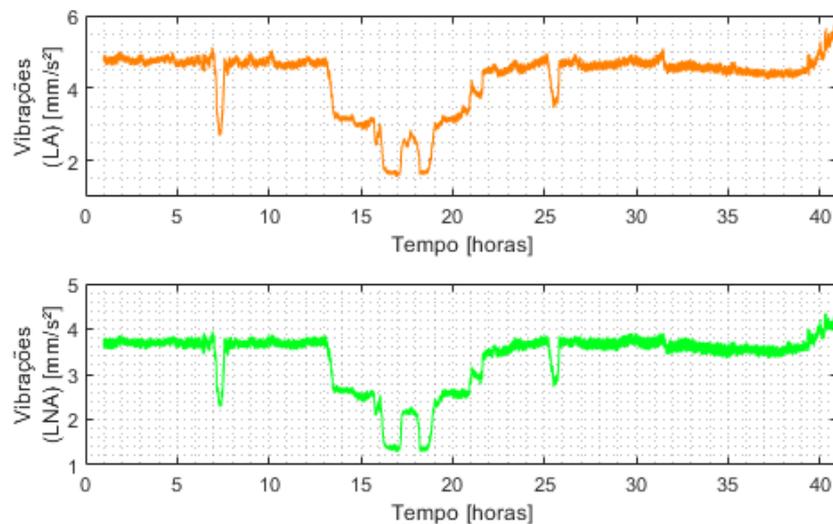
Os dados de monitoramento da situação com falha se encontram nas Figura 19 que mostra a rotação e Figura 20 com as vibrações.

Figura 19 – Sinais de rotação e vibrações dos dados com falha.



Fonte: Autora

Figura 20 – Sinais de rotação e vibrações dos dados com falha.



Fonte: Autora

Existe um momento em que o comportamento das vibrações podem indicar situações de anormalidade, é a região em que há uma elevação grande da vibração posterior ao instante de 38h, quando ocorre a parada do equipamento, como mostra a Figura 21.

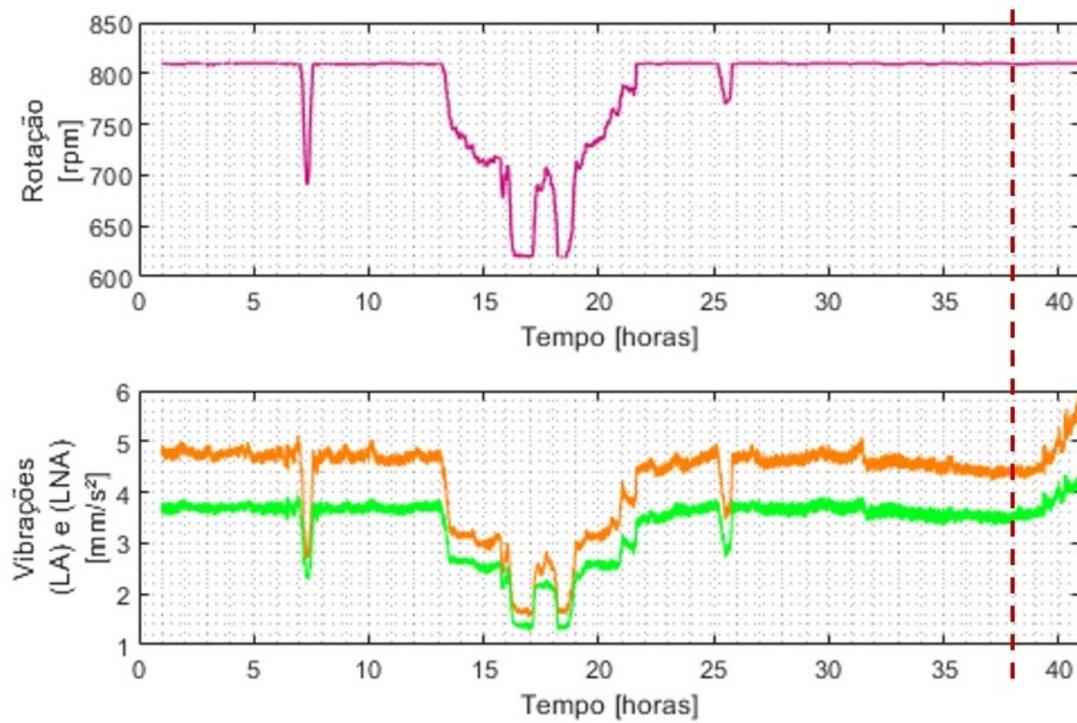
Foi aplicado o modelo “Polinômio de Grau 2” nesses dados de monitoramento para verificar se a falha seria detectada. Foram analisados também o número de falsos positivos e o tempo de monitoramento a *FIT* do modelo, como mostra a Tabela 8.

Tabela 8 – Resultados do monitoramento de dados com falha.

	Nº de alarmes	<i>FIT</i> do modelo	Falsos Positivos	Tempo (seg)
Polinômio de grau 2	1	72,17%	2,06%	0,73

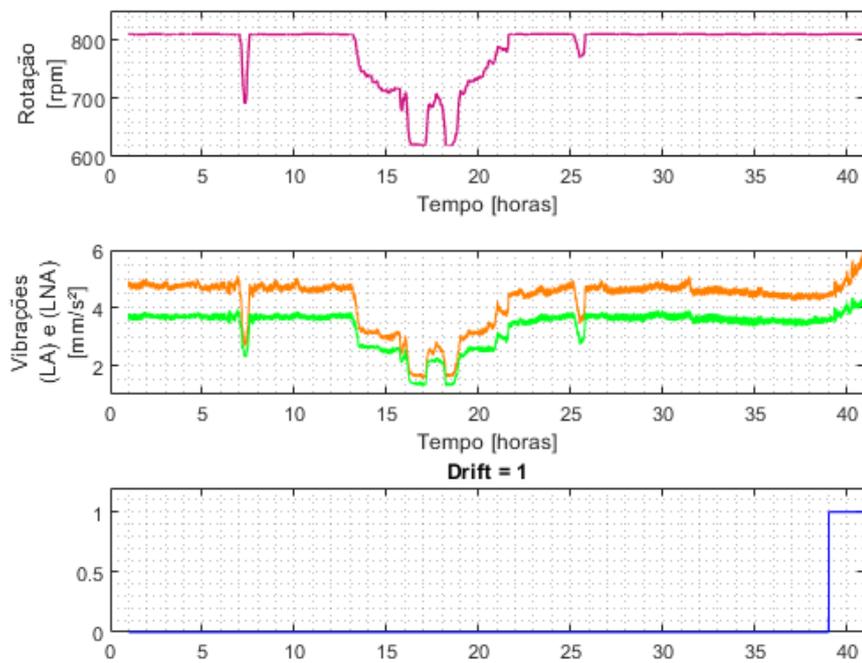
O momento de detecção do *drift* é mostrado na Figura 22. Uma observação em escalas diferentes nos gráficos.

Figura 21 – Momento em que se inicia a falha.



Fonte: Autora

Pode perceber-se que a falha foi detectada pelo algoritmo no instante 39h, antes que a situação se tornasse crítica, aproximadamente no instante 40h. Ou seja, conseguiu-se detectar a falha logo após o início da mesma e antes que a situação se tornasse severa o suficiente para ter parada da planta.

Figura 22 – Momento da detecção de *drift* nos dados de falha.

Fonte: Autora

5 Conclusão e Trabalhos Futuros

Diversos trabalhos na literatura têm tratado da detecção de falhas em processos onde existem diversos pontos de operação e muitas variáveis envolvidas, utilizando especialmente técnicas através de monitoramentos multivariados e análise externa, como citados nos Capítulos 1 e 2. Também existem muitos estudos que tratam do aprendizado ativo de classificadores, que se aplicam especialmente nos problemas onde o número de dados rotulados são escassos e os dados podem estar sujeitos a mudanças ao longo do tempo, como descrito na Seção 2.4.

A proposta deste trabalho foi unir os dois cenários de estudo, propondo um algoritmo robusto para monitoramentos reais, que tanto fosse acurado na detecção de falhas, quanto pudesse se adaptar de forma sustentável ao longo do tempo às mudanças ocorridas no processo.

Portanto, a metodologia para monitoramento de processos em diferentes regiões de operação, baseada em aprendizado ativo através de detecção de *concept drift* e modelagem por análise externa foi construída e testada em um sistema real e apresentou bons resultados durante um mês de operação, detectando poucos possíveis distúrbios (falsos positivos), que foram identificados pelo operador como situações de normalidade. Graças ao modelo de clusterização somente um conjunto reduzido de dados são armazenados a cada detecção de *drift*, tornando o sistema apto a operar em longos períodos de tempo.

Além disso, em uma situação de falha, o modelo proposto conseguiu detectar a mesma com acurácia, demonstrando que é um modelo que pode apresentar bom desempenho em situações reais, porém, para a confirmação dessa hipótese, mais testes seriam necessários, em outras bases de dados que possuam mais situações de falhas.

Otimizar o custo computacional envolvido no treinamento e constantes atualizações de modelos não era a finalidade principal do trabalho, porém mesmo assim, tal questão conseguiu ser tratada e a metodologia proposta se mostrou aplicável e com baixo custo computacional nos modelos que envolviam clusterização de dados.

Trabalhos futuros que possam aprimorar a metodologia proposta incluem testes de demais modelos de análise externa, tais como outras redes neurais, além disso, entender melhor o motivo pelo qual as redes neurais testadas não tiveram bons resultados. Testar demais técnicas de processamento de dados, aliadas às técnicas de *MSPC*, tais como Wavalet e PCA. Pode-se também utilizar outros métodos estatísticos além da estatística T^2 *Hottelling* para o monitoramento. Além disso, é importante propor outras formas de atualizações de modelos. Também existem diversos estudos na literatura sobre aprendizado ativo que podem ser explorados. Também existem diversas outras formas de se fazer a

detecção do *drifts*, então é interessante que outras sejam exploradas. Além disso, é de suma importância que as metodologias desenvolvidas sejam colocadas em prática na operação, de forma a conseguir ajustar como o operador faria a marcação dos dados como falha e normalidade e como isso impactaria o monitoramento em caso de demora de tal rotulação (como em fins de semana e na parte da noite, onde podem não existir funcionários cuidando deste processo). E além disso, pode-se também desenvolver o trabalho com a finalidade de prever tais falhas, e também de conseguir identificá-las.

Referências

- ASSIS, A. A. *Avaliação de técnicas baseadas em análise externa para detecção de falhas em processos industriais operando em múltiplos modos*. Dissertação (Mestrado) — UFES, 2018. Citado 3 vezes nas páginas 22, 40 e 47.
- BAENA-GARCIA, M.; CAMPO-ÁVILA, J. del; FIDALGO, R.; BIFET, A.; GAVALDA, R.; MORALES-BUENO, R. Early drift detection method. In: *Fourth international workshop on knowledge discovery from data streams*. [S.l.: s.n.], 2006. v. 6, p. 77–86. Citado na página 33.
- BASSEVILLE, M.; NIKIFOROV, I. V. et al. *Detection of abrupt changes: theory and application*. [S.l.]: prentice Hall Englewood Cliffs, 1993. v. 104. Citado na página 22.
- BISHOP, C. M. *Pattern recognition and machine learning*. [S.l.]: springer, 2006. Citado na página 35.
- CHIANG, L. H.; RUSSELL, E. L.; BRAATZ, R. D. *Fault detection and diagnosis in industrial systems*. [S.l.]: Springer Science & Business Media, 2001. Citado 6 vezes nas páginas 20, 25, 26, 28, 29 e 32.
- COSTA, A. F. J.; ALBUQUERQUE, R. A. S.; SANTOS, E. M. dos. A drift detection method based on active learning. In: IEEE. *2018 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2018. p. 1–8. Citado 4 vezes nas páginas 32, 33, 34 e 39.
- FACELI, K.; LORENA, A. C.; GAMA, J.; CARVALHO, A. C. P. d. L. et al. *Inteligência Artificial: Uma abordagem de aprendizado de máquina*. [S.l.: s.n.], 2011. Citado 3 vezes nas páginas 36, 37 e 38.
- FARIA, J. C. O. *Detecção de falhas em processos industriais em múltiplos pontos de operação via análise externa linear e não-linear*. Dissertação (Mestrado) — UFES, 2018. Citado 4 vezes nas páginas 22, 40, 46 e 47.
- FARID, D. M.; RAHMAN, C. M. Novel class detection in concept-drifting data stream mining employing decision tree. In: IEEE. *2012 7th International Conference on Electrical and Computer Engineering*. [S.l.], 2012. p. 630–633. Citado na página 22.
- GAMA, J.; CASTILLO, G. Learning with local drift detection. In: SPRINGER. *International Conference on Advanced Data Mining and Applications*. [S.l.], 2006. p. 42–55. Citado na página 33.
- GAMA, J.; MEDAS, P.; CASTILLO, G.; RODRIGUES, P. Learning with drift detection. In: SPRINGER. *Brazilian symposium on artificial intelligence*. [S.l.], 2004. p. 286–295. Citado na página 22.
- GE, Z.; SONG, Z. *Multivariate statistical process control: Process monitoring methods and applications*. [S.l.]: Springer Science & Business Media, 2012. Citado 2 vezes nas páginas 20 e 24.

- GE, Z.; YANG, C.; SONG, Z.; WANG, H. Robust online monitoring for multimode processes based on nonlinear external analysis. *Industrial & Engineering Chemistry Research*, ACS Publications, v. 47, n. 14, p. 4775–4783, 2008. Citado na página 27.
- JAMES, G.; WITTEN, D.; HASTIE, T.; TIBSHIRANI, R. *An introduction to statistical learning*. [S.l.]: Springer, 2013. v. 112. Citado na página 35.
- KADWE, Y.; SURYAWANSHI, V. A review on concept drift. *IOSR J. Comput. Eng.*, v. 17, n. 1, p. 20–26, 2015. Citado na página 32.
- KANO, M.; HASEBE, S.; HASHIMOTO, I.; OHNO, H. Evolution of multivariate statistical process control: application of independent component analysis and external analysis. *Computers & chemical engineering*, Elsevier, v. 28, n. 6-7, p. 1157–1166, 2004. Citado 4 vezes nas páginas 21, 30, 31 e 39.
- KOURTI, T. Process analysis and abnormal situation detection: from theory to practice. *IEEE control systems magazine*, IEEE, v. 22, n. 5, p. 10–25, 2002. Citado na página 20.
- KRUGER, U.; XIE, L. *Statistical monitoring of complex multivariate processes: with applications in industrial process control*. [S.l.]: John Wiley & Sons, 2012. Citado na página 21.
- LOO, H. R.; MARSONO, M. N. Online data stream classification with incremental semi-supervised learning. In: *Proceedings of the Second ACM IKDD Conference on Data Sciences*. [S.l.: s.n.], 2015. p. 132–133. Citado na página 22.
- MACGREGOR, J. F.; KOURTI, T. Statistical process control of multivariate processes. *Control Engineering Practice*, Elsevier, v. 3, n. 3, p. 403–414, 1995. Citado 3 vezes nas páginas 25, 26 e 27.
- MONTGOMERY, D. C.; RUNGER, G. C. *Applied statistics and probability for engineers*. [S.l.]: John Wiley & Sons, 2010. Citado na página 20.
- PEARSON, R. K.; NEUVO, Y.; ASTOLA, J.; GABBOUJ, M. The class of generalized hampel filters. In: IEEE. *2015 23rd European Signal Processing Conference (EUSIPCO)*. [S.l.], 2015. p. 2501–2505. Citado na página 40.
- PINAGE, F. A.; SANTOS, E. M. dos. A dissimilarity-based drift detection method. In: IEEE. *2015 IEEE 27th International Conference on Tools with Artificial Intelligence (ICTAI)*. [S.l.], 2015. p. 1069–1076. Citado na página 33.
- PORTNOY, I.; MELENDEZ, K.; PINZON, H.; SANJUAN, M. An improved weighted recursive pca algorithm for adaptive fault detection. *Control Engineering Practice*, Elsevier, v. 50, p. 69–83, 2016. Citado 3 vezes nas páginas 22, 28 e 32.
- RU, L. H.; ANDROMEDA, T.; MARSONO, M. Online data stream learning and classification with limited labels. *Proceeding of the Electrical Engineering Computer Science and Informatics*, v. 1, n. 1, p. 161–164, 2014. Citado na página 22.
- SATOYAMA, Y.; FUJIWARA, K.; KANO, M. Variable elimination-based contribution for accurate fault identification. *IFAC-PapersOnLine*, Elsevier, v. 49, n. 7, p. 383–388, 2016. Citado na página 20.

SAYED-MOUCHAWEH, M. *Learning from data streams in dynamic environments*. [S.l.]: Springer, 2016. Citado na página 32.

SCHOHN, G.; COHN, D. Less is more: Active learning with support vector machines. In: CITESEER. *ICML*. [S.l.], 2000. v. 2, n. 4, p. 6. Citado na página 35.

SETHI, T. S.; KANTARDZIC, M. Don't pay for validation: Detecting drifts from unlabeled data using margin density. In: *INNS Conference on Big Data*. [S.l.: s.n.], 2015. v. 53, p. 103–112. Citado na página 33.

SOBHANI, P.; BEIGY, H. New drift detection method for data streams. In: SPRINGER. *International conference on adaptive and intelligent systems*. [S.l.], 2011. p. 88–97. Citado 3 vezes nas páginas 21, 22 e 39.

SPECHT, D. F. et al. A general regression neural network. *IEEE transactions on neural networks*, v. 2, n. 6, p. 568–576, 1991. Citado na página 32.

TONG, S. *Active learning: theory and applications*. [S.l.]: Stanford University USA, 2001. v. 1. Citado 2 vezes nas páginas 34 e 35.

VALE S.A. *Fluxo do processo de pelotização de minério de ferro*. 2020. Disponível em: <<http://www.vale.com/brasil/pt/aboutvale/news/paginas/entenda-funciona-processo-pelotizacao-usinas.aspx>>. Acesso em: 20 fev. 2020. Citado na página 45.

VALE S.A. *Qual a importância da mineração para a economia do país?* 2020. Disponível em: <<http://www.vale.com/brasil/pt/aboutvale/news/paginas/qual-a-importancia-da-mineracao-para-a-economia-do-pais.aspx>>. Acesso em: 22 jun. 2020. Citado na página 45.

VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 2013. Citado na página 32.

VENKATASUBRAMANIAN, V.; RENGASWAMY, R.; KAVURI, S. N. A review of process fault detection and diagnosis: Part ii: Qualitative models and search strategies. *Computers & chemical engineering*, Elsevier, v. 27, n. 3, p. 313–326, 2003. Citado na página 20.

VENKATASUBRAMANIAN, V.; RENGASWAMY, R.; KAVURI, S. N.; YIN, K. A review of process fault detection and diagnosis: Part iii: Process history based methods. *Computers & chemical engineering*, Elsevier, v. 27, n. 3, p. 327–346, 2003. Citado 2 vezes nas páginas 20 e 27.

VENKATASUBRAMANIAN, V.; RENGASWAMY, R.; YIN, K.; KAVURI, S. N. A review of process fault detection and diagnosis: Part i: Quantitative model-based methods. *Computers & chemical engineering*, Elsevier, v. 27, n. 3, p. 293–311, 2003. Citado 2 vezes nas páginas 20 e 21.

WANG, H.; ABRAHAM, Z. Concept drift detection for streaming data. In: IEEE. *2015 International Joint Conference on Neural Networks (IJCNN)*. [S.l.], 2015. p. 1–9. Citado na página 22.

WASSERMAN, P. D. *Advanced methods in neural computing*. [S.l.]: John Wiley & Sons, Inc., 1993. Citado na página 32.

YOUNG, P. C. *Recursive estimation and time-series analysis: An introduction for the student and practitioner*. [S.l.]: Springer Science & Business Media, 2011. Citado na página 21.

ZHANG, H. *Statistical process monitoring and modelling using PCA and PLS*. Dissertação (Mestrado) — University of Alberta, 2000. Citado 4 vezes nas páginas 24, 25, 28 e 39.

ZHAO, S. J.; ZHANG, J.; XU, Y. M. Monitoring of processes with multiple operating modes through multiple principle component analysis models. *Industrial & engineering chemistry research*, ACS Publications, v. 43, n. 22, p. 7025–7035, 2004. Citado na página 21.

ŽLIOBAITĖ, I.; BIFET, A.; PFAHRINGER, B.; HOLMES, G. Active learning with drifting streaming data. *IEEE transactions on neural networks and learning systems*, IEEE, v. 25, n. 1, p. 27–39, 2013. Citado 2 vezes nas páginas 33 e 34.