INSTITUTO FEDERAL DO ESPÍRITO SANTO
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

LUDMILA JUNCA LOPES

SISTEMA INTELIGENTE DE APOIO A DECISAO PARA A FASE RAMPA DE ACELERAÇÃO DO PROCESSO DE PARTIDA A QUENTE DE POÇOS DE ELEVAÇAO BCSS

SERRA
2018

LUDMILA JUNCA LOPES

SISTEMA INTELIGENTE DE APOIO A DECISAO PARA A FASE RAMPA DE ACELERAÇÃO DO PROCESSO DE PARTIDA A QUENTE DE POÇOS DE ELEVAÇAO BCSS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Controle e Automação do Instituto Federal do Espírito Santo, como requisito parcial para obtenção do Título de Mestre em Engenharia de Controle e Automação.

Orientador: Prof. Dr. Gustavo Maia de Almeida

SERRA

2018

Dados Internacionais de Catalogação na Publicação (CIP)

L864s 2018

Lopes, Ludmila Junca
Sistema inteligente de apoio a decisão para a fase rampa de aceleração do processo de partida a quente de poços de elevação BCSS / Ludmila Junca Lopes. - 2018.
81 f.; il.; 30 cm

Orientador: Prof. Dr. Gustavo Maia de Almeida.
Dissertação (mestrado) - Instituto Federal do Espírito Santo, Programa de Pós-graduação em Engenharia de Controle de Automação, 2018.

1. Bombas centrífugas. 2. Petróleo - Produção. 3. Dinâmica dos fluidos. 4. Algoritmos genéticos. 5. Válvulas. I. Almeida, Gustavo Maia de. II. Instituto Federal do Espírito Santo. III. Título.

CDD 665.5

Bibliotecária Rogeria Gomes Belchior - CRB6/ES 417

MINISTÉRIO DA EDUCAÇÃO

INSTITUTO FEDERAL DO ESPÍRITO SANTO

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE CONTROLE E AUTOMAÇÃO

LUDMILA JUNCA LOPES

SISTEMA INTELIGENTE DE APOIO A DECISÃO PARA A FASE RAMPA DE ACELERAÇÃO DO PROCESSO DE PARTIDA A QUENTE DE POÇOS DE ELEVAÇÃO BCSS

Dissertação apresentada ao Programa de Pós-Graduação em Engenharia de Controle e Automação do Instituto Federal do Espirito Santo, como requisito parcial para obtenção de título de Mestre em Engenharia de Controle e Automação.

Aprovado em 28 de fevereiro de 2018

COMISSÃO EXAMINADORA

Prof. Dr. Gustavo Maia de Almeida
Instituto Federal do Espírito Santo
Orientador

Prof. Dr. Wagner Teixeira da Costa
Instituto Federal do Espírito Santo
Membro Interno

Prof. Dr. Edilson Luiz do Nascimento
Instituto Federal do Espírito Santo
Membro externo

AGRADECIMENTOS

Em primeiro lugar agradeço a Luzineth, minha mãe, e a Natália, minha irmã, pelo amor e incentivo incondicionais, por sempre acreditarem em mim e, sobretudo, por todos os ensinamentos que recebi ao longo da minha vida e continuo a receber diariamente. Sem vocês a minha vida não teria sentido.

Agradeço também ao prof. Gustavo Maia de Almeida por toda a jornada até esse momento. Só nós sabemos dos desafios que enfrentamos ao longo dessa caminhada. Suas orientações, contrapontos, apoio e compreensão foram imprescindíveis para que eu chegasse até aqui.

Meu muito obrigada aos colegas Jean Araújo e Otávio Borges por me ensinarem sobre BCSS; ao Ricardo Vargas por me apresentar suas ideias sobre aprendizado de máquina com séries temporais, ao Heber Barbosa e ao Leonardo Baraúna pelo apoio quando da minha decisão em inscrever nesse programa de mestrado; novamente ao Leonardo Baraúna por todo apoio ao longo dessa trajetória; e à Kellen Suamy, por sua fiel amizade e conselhos.

Meu sincero agradecimento à Ana, companheira de todas as horas, mesmo isto significando abdicar de fins de semana e feriados para que eu pudesse testar hipóteses, escrever artigos e essa dissertação. Obrigada pela compreensão, apoio e incentivo! Jamais esquecerei quando o editor de texto do meu computador não conseguiu mais abrir o arquivo da dissertação da qualificação, eu estava em viagem para participar de um congresso no Peru, e você dedicou seu tempo para ajustar o meu trabalho.

Por fim, obrigada a todos os conhecidos e desconhecidos que contribuíram com as ideias que formam os alicerces desse estudo.

“Lute com determinação, abrace a vida com paixão, perca com classe e vença com ousadia, porque o mundo pertence a quem se atreve e a vida é muito bela para ser insignificante”.

(Charlie Chaplin)

RESUMO

Bombas centrífugas submersas têm sido cada vez mais utilizadas na indústria de petróleo. A partida desse equipamento o submete a condições extremas que podem reduzir a sua vida útil. Além disso, a vazão de óleo produzido durante esse processo é inferior quando comparada a produção em condição normal de operação. Durante a partida, duas variáveis são manipuladas gradativamente a fim de conduzir o equipamento ao patamar operacional desejado, a Frequência do Inversor e o Percentual de Abertura da Válvula *Choke*. Atualmente, o controle desse processo é manual e está sujeito à experiência e sensibilidade do operador. Utilizar um sistema inteligente, baseado em classificação, capaz de predizer o momento adequado de realizar as manipulações permitirá que o processo ocorra de forma mais rápida e padronizada, trazendo como benefícios o aumento da produção de óleo devido à operação com restrição por menos tempo e a operação do equipamento dentro da faixa adequada, minimizando o risco de falha. Ao analisar os dados históricos disponíveis, detectou-se que há desbalanceamento de classes e ruído de classificação, desafios bem conhecidos no campo da aprendizagem de máquina em tarefas de classificação. Assim, os algoritmos de classificação *k*-NN e *RUSBoost* foram avaliados, utilizando como entrada os dados históricos disponíveis, tanto em sua natureza original (registros independentes), quanto agrupados em séries temporais. No caso das séries temporais, a medida de distância *Dynamic Time Warping* (DTW) foi empregada na extração de características. Duas abordagens para seleção da série temporal de referência no cálculo da distância foram avaliadas: utilizar a série de menor duração e empregar a série indicada por Algoritmo Genético (AG). Os resultados indicam a viabilidade do uso do *RUSBoost* com séries temporais de entrada. O modelo gerado com série temporal de referência escolhida por AG apresentou o melhor desempenho em relação às abordagens estudadas, com AUC de 0,9175.

Palavras-chave: Partida de bomba centrífuga submersa. *RUSBoost*. *k*-NN. Ruído de classificação. Desbalanceamento de classes.

ABSTRACT

Electrical submersible pumps have been increasingly used in oil and gas industry. The startup cycle of this equipment causes it to be subjected to extreme conditions that can reduce its useful life. Additionally, in startup cycle, production rates are lower than under normal operational conditions. Two variables are gradually manipulated to drive the equipment to the desired operational level, the Frequency of Variable Speed Drive and the Opening of the Flow Choke Valve. Currently, this process is controlled manually, so its subject to operator experience and sensitivity. A classification-based system, capable of predicting the right moment to perform the manipulations in the variables will allow the process to be more quickly and standardized, bringing as benefits an increase of oil production rate (due to restricted operation for less time) and a safer operation, minimizing the risk of failure. When analyzing the available historical data, it was detected that there is class imbalance and classification noise, well known challenges in classification tasks. Thus, the *k*-NN and RUSBoost classification algorithms were evaluated using the historical data available, either in their original nature (independent records) or grouped in time series. In the case of time series, the Dynamic Time Warping (DTW) distance measure was used to extract features. Two approaches to selecting the reference time series in the distance calculation were evaluated: the shorter time serie and the one elected by Genetic Algorithm (GA). The results indicate the feasibility of using RUSBoost with time series of input, the best model was the one generated with time series pointed by GA and presented AUC 0.9175.

Keywords: Electrical submersible pump startup cycle. RUSBoost. *k*-NN. Noise labeling. Imbalanced class problem.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS, SIGLAS

AG - Algoritmo Genético

AUC - *Area Under Curve ROC*

BAB - Base Adaptadora de Bombeio

BCSS - Bomba Centrífuga Submersa Submarina

BED - *Boundary Elimination and Domination Algorithm*

DTW - *Dynamic Timing Warping*

ENN - *Edited Nearest Neighbor Rule*

FN - Falso Negativo

FP - Falso Positivo

KDD - *Knowledge Discovery in Databases*

k-NN - *k-Nearest Neighbors*

MoBo - Módulo de Bombeio

ROC - *Receiver Operating Characteristic*

RUS - *Random Under Sampling*

TFN - Taxa de Falsos Negativos

TFP - Taxa de Falsos Positivos

TVN - Taxa de Verdadeiros Negativos

TVP - Taxa de Verdadeiros Positivos

SVM - *Support Vector Machine*

VN - Verdadeiro Negativo

VP - Verdadeiro Positivo

VSD - *Variable Speed Drive*

SUMÁRIO

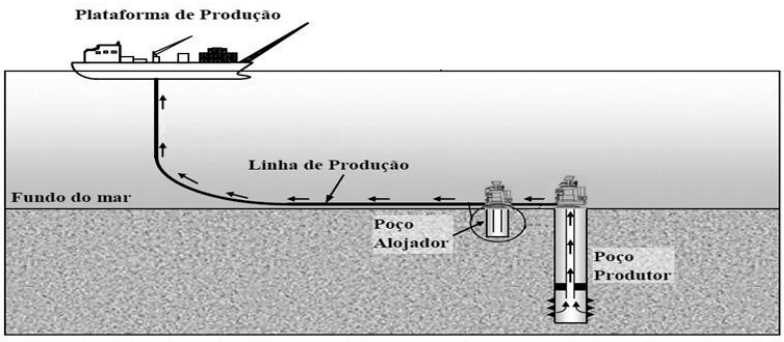
1 INTRODUÇÃO

O processo de produção de petróleo em campos marítimos compreende a atividade de extração do fluido da rocha reservatório, a sua elevação até as facilidades de produção da plataforma e o processamento primário, cujo principal objetivo é separar o óleo, o gás e a água.

Quando a pressão do reservatório é suficiente para elevar o fluído até as facilidades de produção, diz-se que a elevação é natural, e os poços que produzem dessa maneira são denominados surgentes. Quando a pressão é relativamente baixa, os fluidos não alcançam a superfície sem o uso de mecanismos que suplementem a energia natural através de elevação artificial (THOMAS, 2001).

A utilização do método de elevação artificial por bombeio centrífugo submerso submarino tem se expandido na indústria de petróleo. A forma mais comum de instalação da bomba centrífuga submersa submarina (BCSS) é diretamente no poço produtor. No entanto, esta apresenta como inconveniente a obrigatoriedade de retirada de toda a coluna de produção do poço quando da necessidade de intervenção, o que aumenta o tempo e o custo do procedimento. Uma alternativa a essa abordagem é o uso de um conjunto de bombeio instalado em um compartimento especial no leito do mar, denominado poço alojador. Nessa configuração, o fluxo do fluido ocorre de forma surgente do poço produtor até o equipamento, a partir de onde é bombeado para a superfície por meio da linha de produção. A Figura 1 ilustra o arranjo descrito.

Figura 1 - Sistema de produção com poço alojador



Fonte: Betônico (2013)

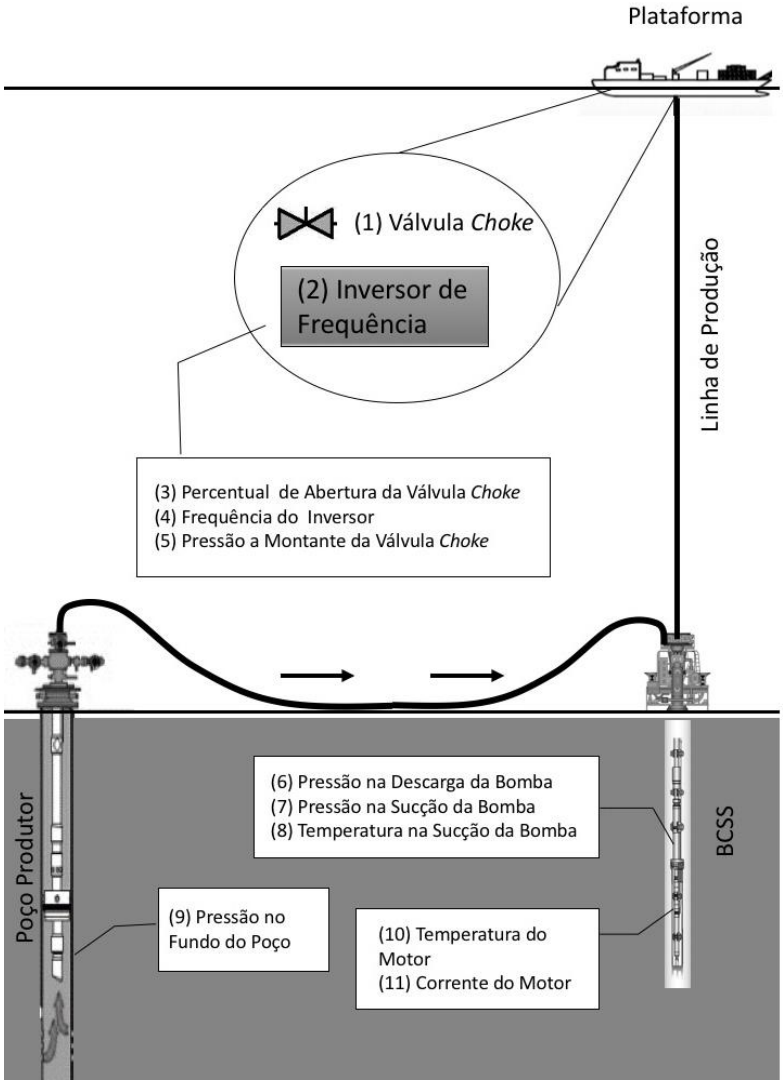
O conjunto de bombeio é projetado para operar de forma contínua, porém, ocorre deste ser desligado, seja por uma parada programada ou por eventos inesperados, como shutdown da plataforma ou do próprio equipamento. O processo de religar o equipamento é conhecido como partida e se estende do período em que este se encontra desligado até o momento em que as condições operacionais desejadas são alcançadas.

O processo de partida do equipamento instalado em poço alojador oferece uma série de desafios em virtude da baixa temperatura do fluido, risco de formação de emulsão e possibilidade de separação do gás. Operar sob essas circunstâncias representa uma ameaça à integridade do equipamento e requer que o processo de partida seja realizado de forma gradativa e controlada, tanto para atenuar condições que possam reduzir a sua vida útil, quanto evitar o descontrole no processo e, consequentemente, shutdown e repartida. Diversos sensores e atuadores são utilizados no monitoramento e controle da partida. A Figura 2 apresenta os principais entes envolvidos neste processo.

A Válvula Choke (Figura 2 - 1) e o Inversor de Frequência (Figura 2 - 2) atuam, respectivamente, no controle da vazão do fluído que chega à plataforma pela linha de produção e no controle da velocidade do rotor do motor. Para tal, manipula-se o Percentual de Abertura da Válvula Choke (Figura 2 - 3) e a Frequência do Inversor (Figura 2 - 4). Tanto o poço produtor quanto o conjunto de bombeio dispõem de sensores de temperatura e pressão (Figura 2 - 6, 7, 8, 9, 10). Adicionalmente, mede-se a Pressão a Montante da Válvula Choke (Figura 2 - 5) presente na plataforma e a Corrente do Motor (Figura 2 - 11).

A temperatura do fluido, medida pelo sensor de temperatura localizado na sucção da bomba, tem papel preponderante na execução do processo de partida, pois quando o desligamento do equipamento ocorre por um longo período de tempo, o fluido resfria e o óleo se torna mais viscoso. Partir o equipamento com um fluido de alta viscosidade representa um risco à integridade do conjunto de bombeio e, nesses casos, executa-se um procedimento específico, denominado partida a frio. A determinação do limite de temperatura leva em consideração as características reológicas do petróleo, e, portanto, varia entre campos produtores.

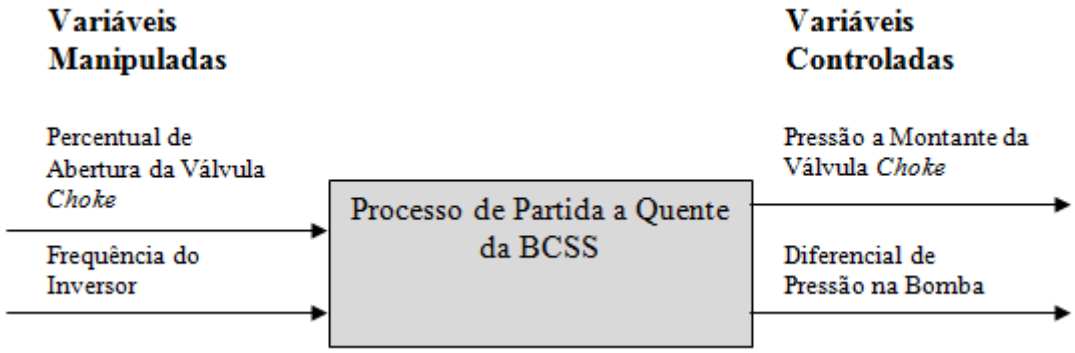
Figura 2 - Sensores e atuadores do processo de partida



Fonte: Autoria própria

Quando a temperatura é igual ou superior à temperatura limite, deve ser utilizado o procedimento de partida a quente, que será objeto de estudo desse trabalho. Neste, duas variáveis são gradativamente manipuladas manualmente: Frequência do Inversor e Percentual de Abertura da Válvula Choke. As variáveis Pressão a Montante da Válvula Choke e Diferencial de Pressão na Bomba (Pressão na Descarga da Bomba - Pressão Sucção da Bomba) são as variáveis controladas. A Figura 3 ilustra as variáveis envolvidas no controle do processo de partida a quente da BCSS.

Figura 3 - Controle do processo de partida a quente da BCSS

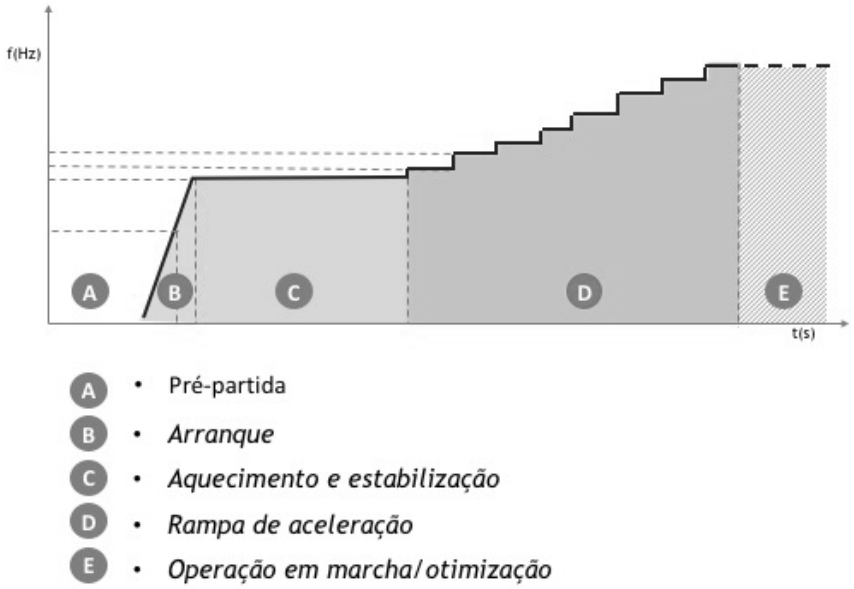


Fonte: Autoria própria

O procedimento de partida a quente é composto por cinco fases: (A) Pré-partida, (B) arranque, (C) aquecimento e estabilização, (D) rampa de aceleração e (E) operação em marcha/otimização. A fase de pré-partida consiste em verificar e adequar o sistema de modo que todas as pré-condições para o início da operação do equipamento, tais como a energização do Inversor de Frequência e o posicionamento adequado das válvulas dos equipamentos envolvidos, sejam satisfeitas. Na fase seguinte, de arranque, após uma etapa de magnetização inicial, o rotor do motor é acelerado até alcançar a velocidade mínima programada para o início da partida. Na fase de aquecimento e estabilização, a bomba opera, por um tempo mínimo definido (que pode variar de poço para poço) na velocidade mínima programada.

Caso a operação ocorra dentro da normalidade, passa-se a fase seguinte, a rampa de aceleração. Nesta, as variáveis Frequência do Inversor e Percentual de Abertura da Válvula Choke são gradativamente incrementadas, respectivamente, em 1Hz e 1%, até que o ponto de operação desejado seja atingido. Na última fase, operação em marcha/otimização, o objetivo é realizar pequenos ajustes (mais comuns no Percentual de Abertura da Válvula Choke) de modo a otimizar a operação do equipamento. Conforme descrito, o processo é complexo e precisa ser realizado com cautela. A Figura 4 mostra as fases citadas num gráfico que demonstra a relação Tempo de Partida (s) e a Frequência do Inversor (Hz).

Figura 4 - Fases da partida da BCSS



Fonte: Autoria própria

Conforme pode ser observado no gráfico da Figura 4, a fase Rampa de Aceleração ocupa a maior parte do tempo do processo de partida e é nesta que os incrementos nas variáveis manipuladas são realizados de forma mais sistemática. A Figura 5 detalha o fluxo do processo das fases A, B, C e D. Neste, é importante destacar:

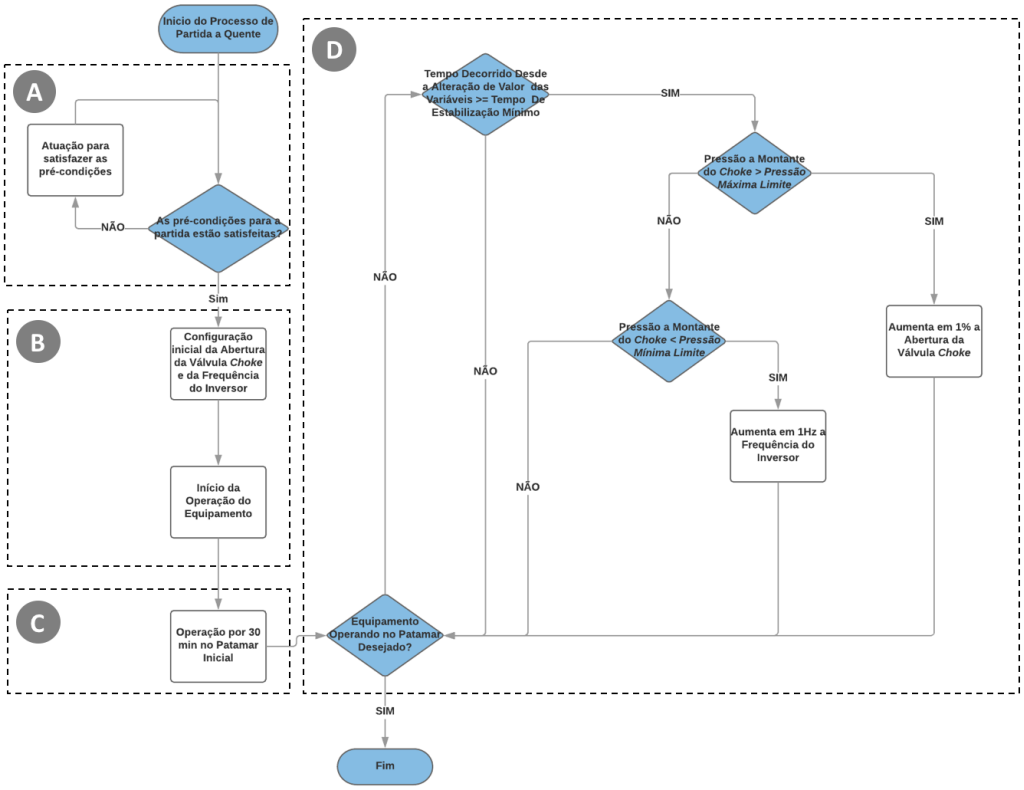
- A necessidade de estabilização do processo após incremento em qualquer que seja a variável manipulada, evidenciada na condicional “Tempo Decorrido Desde a Alteração de Valor das Variáveis >= Tempo de Estabilização Mínimo”;

- A importância do operador na tomada de decisão, pois toda verificação da condição das variáveis controladas é feita por este de forma manual.

Portanto, a experiência do operador é fator determinante para o sucesso da partida. Embora existam critérios objetivos, é a sua análise e ação que efetivamente conduzem o processo. Assim, pode-se hipotetizar que uma investigação no histórico de todas as variáveis, ao longo de diversas partidas completadas com êxito, relevará grande variância em termos de duração e eventuais divergências em termos de decisão a respeito de incrementar uma variável manipulada, dada uma mesma condição operacional. Nesse caso, embora todas as operações tenham conduzido a

partida de forma bem-sucedida, o incremento realizado de forma mais tardia constitui uma ineficiência do processo, que gera impacto financeiro devido à menor produção de óleo e pode comprometer a integridade do equipamento devido à exposição a condições adversas por mais tempo que o necessário.

Figura 5 - Processo de partida a quente, fases A, B, C e D



Fonte: Autoria própria

O aprendizado de máquina tem sido extensivamente empregado na indústria por “dar aos computadores a habilidade de aprender sem serem explicitamente programados” (SAMUEL, 1959). Na técnica de classificação, dados de entrada são divididos em classes (rótulos), e a partir destes, um modelo é gerado com o objetivo de predizer a classe de dados desconhecidos quando estes são apresentados. O emprego desta técnica no processo de partida a quente pode otimizar a sua execução através da definição do momento adequado de manipular as variáveis de interesse na fase Rampa de Aceleração (Figura 5 - D).

O processo de partida a quente não possui modelo matemático. Portanto, o aprendizado se dará com dados reais, o que é um desafio dada a forma como as partidas são conduzidas. Conforme hipótese, existem condições operacionais similares cuja decisão a respeito de incrementar a variável diverge. Em outras palavras, essas entradas apresentam rótulos de classificação (classes) diferentes. No campo do aprendizado de máquina, tal situação é chamada de ruído de classificação e é um fator complicador, pois o classificador não consegue generalizar o problema e gerar um modelo adequado.

Outro fator de risco é a desproporção de dados de cada classe. Conforme destacado anteriormente, é necessário que o processo estabilize antes de qualquer nova alteração nas variáveis manipuladas. Isso implica na existência de mais dados que apontam para a decisão de manter o set point do que efetuar incrementos na Frequência do Inversor ou no Percentual de Abertura da Válvula Choke. A esse problema, dar-se o nome de desbalanceamento de classes, e assim como a presença do ruído de classificação, acarreta em modelos pouco eficientes em predizer as classes de interesse, neste caso, que apontam para o incremento de alguma das duas variáveis manipuladas.

Para transpor as dificuldades, é necessário realizar um estudo minucioso das características dos dados disponíveis, bem como de técnicas adequadas para gerar modelos com cenário de desbalanceamento de classes e ruído de classificação. Além disso, deve-se investigar quais critérios são mais eficientes para julgar a capacidade de predição dos modelos gerados.

Ao revisar o estado da arte do processo de partida de BCSS, identificou-se que desde os estudos iniciais de Neely e Patternson (1984), cujos experimentos demonstraram que o tempo de vida do equipamento aumenta quando suas partidas ocorrem com tensão reduzida, e de Hyde e Brinner (1986), cujo trabalho gerou diversas recomendações a fim de evitar o estresse do equipamento durante esse processo, pouco se avançou no caminho de um sistema inteligente para apoiar a tomada de decisão na condução desse procedimento.

O simulador proposto por Batista (2009), que integrou modelos matemáticos de reservatório, de bomba centrífuga submersa, de escoamento de fluidos e de motor

elétrico para modelar o comportamento transiente de um poço com método de elevação por bombeio centrífugo submerso merece destaque. No entanto, cabe ressaltar que este trabalho não focou no processo de partida e considerou como cenário a produção de poços terrestres com a bomba centrífuga submersa instalada na coluna de produção.

Expandindo a pesquisa para o processo de partida do motor de indução, que compõe a BCSS, destaca-se o trabalho de Kashif e Saqib (2008), que propuseram um sistema de partida suave (soft starter) baseado em redes neurais artificiais e sistema de inferência fuzzy neuro adaptativo. Embora a BCSS utilize um Inversor de Frequência ao invés de um soft starter, é relevante destacar a emprego da inteligência artificial nessa área.

Assim, fica evidenciado que o que é proposto neste trabalho não foi abordado anteriormente.

1.1 JUSTIFICATIVA E IMPORTÂNCIA

O processo de partida do sistema de bombeio centrífugo submerso é crítico, pois submete o equipamento a condições adversas, tais como: baixa temperatura do fluido, fluido em emulsão e maior presença de gás. A operação nessas condições pode reduzir o tempo de vida útil do equipamento ou ocasionar falhas que impactam em sua eficiência.

A relação entre o tempo de vida e o número de tentativas de partida de BCSS tem sido objeto de diversos estudos. Tentativas de partidas excessivas podem reduzir significativamente a vida útil do equipamento ou expô-lo a modos de falha (VERGARA, 2015). A correta determinação do momento em que o incremento nas variáveis manipuladas deve ser aplicado é essencial para que o conjunto se mantenha dentro dos parâmetros operacionais, garantindo não só a integridade do equipamento, mas também a estabilidade do processo, evitando assim a ocorrência de shutdown e consequente repartida. No entanto, o intervalo entre incrementos não deve ser maior que o necessário para a estabilização do processo, pois as restrições

operacionais aplicadas à Frequência do Inversor e à abertura da Válvula Choke reduzem a vazão de fluido produzido.

É importante ressaltar que o processo de partida é acompanhado de forma individual, ou seja, em caso de shutdown da plataforma, todos os poços devem ser partidos de forma sequencial. Assim, a otimização desse processo assegurará maior eficiência operacional por meio da redução da duração das partidas e contribuirá para a integridade do equipamento, resultando em redução de custos com intervenção e diminuição de perdas de produção.

1.2 OBJETIVOS

Este trabalho tem como objetivo geral desenvolver um sistema inteligente de apoio a decisão para a fase Rampa de Aceleração do processo de partida a quente de poços produtores com método de elevação BCSS. Ter um sistema apoiando o processo garantirá maior previsibilidade e ajudará a reduzir a subjetividade na tomada de decisão. Assim, espera-se uma redução no tempo das partidas e, consequentemente, aumento na produção de óleo devido à operação com restrição por menos tempo. Do ponto de vista de integridade, o processo controlado de forma mais estável resultará em menos ocorrências de shutdown durante a partida e decorrentes repartidas. Além disso, assegurará a operação do equipamento dentro das faixas adequadas, minimizando o risco de falha.

Os objetivos específicos são:

- Avaliar os dados disponíveis com o objetivo de confirmar o desbalanceamento de classes e a presença de ruído de classificação;
- Compreender as implicações do processo de classificação com desbalanceamento de classes e ruído de classificação;
- Estudar alternativas de solução para classificação com desbalanceamento de classes e ruído de classificação;

- Comparar diferentes abordagens de classificação com o intuito de definir a mais adequada para o problema;
- Selecionar a (s) métrica (s) de avaliação de modelo de classificação mais adequada (s) ao problema.

1.3 ORGANIZAÇAO DO TRABALHO

Esta dissertação está dividida em cinco capítulos. Este, o capítulo 1, que contextualiza o problema a ser resolvido, expõe a justificativa e importância do trabalho, explicita os objetivos gerais e específicos e, descreve a sua organização. O capítulo 2, que apresenta o referencial teórico que embasa esta pesquisa, onde serão definidos os seguintes conceitos: Bombeio centrífugo submerso submarino, aprendizado de máquina, dynamic timing warping e algoritmo genético. O capítulo 3, que explica a metodologia empregada, destacando as fases: coleta de dados, estruturação dos dados, análise de dados, pré-processamento, extração de características e construção de modelos. Os capítulos 4 e 5, que mostram, respectivamente, os resultados e as conclusões obtidas com esse estudo.

2 REFERENCIAL TEÓRICO

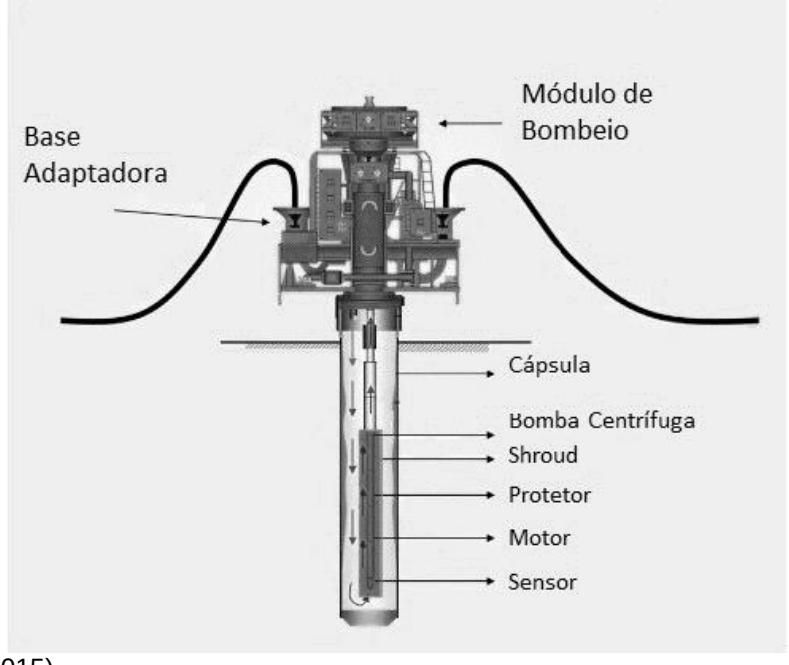
2.1 BOMBEIO CENTRÍFUGO SUBMERSO SUBMARINO

Neste método de elevação, a energia é transmitida para o conjunto de bombeio por meio de um cabo elétrico. A energia elétrica é transformada em energia mecânica através de um motor de subsuperfície, o qual está diretamente conectado a uma bomba centrífuga, que por sua vez transmite energia para o fluido sob a forma de pressão, elevando-o para a superfície.

O conjunto de bombeio instalado em poço alojador é composto por uma base adaptadora de bombeio (BAB), que permite o bypass do fluido caso ocorra algum problema com a bomba, e um módulo de bombeio (MoBo) recuperável, que abriga a bomba centrífuga submersa mencionada.

Os principais componentes do conjunto são ilustrados na Figura 6 e estão descritos a seguir.

Figura 6 - Conjunto de bombeio instalado em poço alojador



Fonte: Vergara (2015)

Bomba Centrífuga: Constituída por unidades distintas chamadas “estágios”. Cada estágio consiste de um impelidor e um difusor. O giro do impelidor cria um campo centrífugo responsável pelo aumento da velocidade e pressão do fluido. Ao escoar pelo difusor, o fluido encontra um aumento progressivo na área de escoamento que causará queda da velocidade e aumento de pressão (Teorema de Bernoulli). A forma e o tamanho do impelidor e do difusor determinam a vazão a ser bombeada. Cada estágio fornece um incremento de pressão ao fluído, dessa forma a quantidade de estágios da bomba é determinada pela pressão que necessita ser vencida para que o fluido chegue à superfície.

Protetor ou Selo: Equipamento instalado entre o motor e a admissão da bomba cujas principais funções são: conectar o eixo do motor com o eixo da bomba, suportar esforços axiais da bomba, evitar a entrada de fluido do poço para o interior do motor e prover o volume necessário para a expansão do óleo do motor devido ao seu aquecimento.

Separador de Gás: Devido à ineficiência da bomba centrífuga na presença de gás, faz-se necessário a presença de um separador. Esse equipamento está localizado na parte inferior da bomba.

Cabo Elétrico: Cabo trifásico com condutores de cobre e alumínio, cuja principal função é transmitir a energia da superfície para o motor elétrico de indução. A depender do modelo, também por esse cabo é transmitido o sinal dos sensores instalados na bomba para a superfície.

Sensores: Instrumentos instalados abaixo do motor que possuem como finalidade avaliar o comportamento do equipamento, como por exemplo, através de medições de pressão e temperatura.

Motor elétrico de Indução: Equipamento do tipo gaiola de esquilo de dois polos e três fases. Durante a operação do motor elétrico, calor é gerado devido à corrente elétrica em seus enrolamentos. Esta geração de calor eleva a sua temperatura. Assim, é necessário resfriar o motor elétrico, pois este possui uma temperatura máxima de operação acima da qual pode ocorrer uma falha nos materiais isolantes dos seus enrolamentos e conexões, impossibilitando o seu funcionamento. Na montagem do conjunto de fundo, o motor elétrico é conectado abaixo da bomba

centrífuga. Dessa maneira, todo o fluido produzido do reservatório que chega à entrada da bomba centrífuga passa antes em volta do motor elétrico. Uma das principais motivações para esta disposição dos equipamentos do conjunto de fundo é fazer com que os fluidos produzidos refrigerem o motor elétrico. O efeito do resfriamento ocorre por convecção forçada dos fluidos produzidos, mais frios, escoando em contato com a parede do motor, mais quente.

A utilização da Bomba Centrífuga Submersa Submarina está se expandindo na elevação artificial pela crescente flexibilidade e evolução dos equipamentos disponíveis para esse método (THOMAS, 2001). Acredita-se que hoje cerca de 10% do fornecimento mundial de petróleo seja produzido através do seu uso.

Conforme resumido por Takács (2009), a utilização dessa técnica nas seguintes condições abaixo deve ser analisada criteriosamente:

Gás livre presente na sucção da bomba submersível, pois deteriora a sua eficiência e pode inclusive impedir a produção de líquidos totalmente. O uso de separadores de gás ou manipuladores de gás é necessário se houver mais de 5% de gás livre na sucção da bomba.

Na presença de areia ou materiais abrasivos nos fluidos de produção, pois aumentam o desgaste do equipamento. Materiais especiais resistentes à abrasão estão disponíveis, mas aumentam os custos do equipamento;

Produção de óleos de alta viscosidade, pois aumenta os requisitos de energia e reduz o potencial de elevação.

2.3 APRENDIZADO DE MÁQUINA

Com a crescente complexidade dos problemas e do grande volume de dados gerados por diferentes setores, tornou-se clara a necessidade de ferramentas computacionais mais autônomas, capazes de criar por si próprias, a partir de uma experiência passada, uma hipótese ou função, capaz de resolver o problema que se

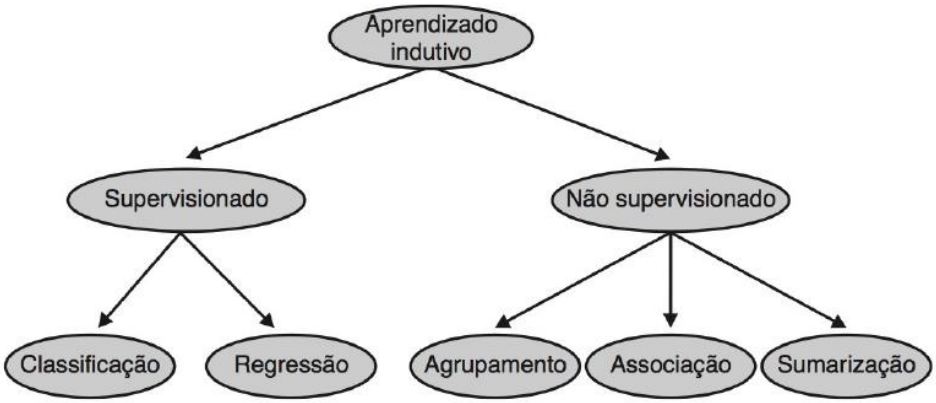
deseja tratar. A esse processo de indução de uma hipótese (ou aproximação de uma função), a partir de experiência passada, dar-se o nome de Aprendizado de Máquina (FACELI et al., 2011).

Os algoritmos de Aprendizado de Máquina aprendem a induzir uma função ou hipótese a partir de dados que representam instâncias do problema a ser resolvido. Cada dado (também chamado de objeto, exemplo ou registro) é formado por uma dupla de características (também conhecidas como atributos, campos ou variáveis), que descrevem seus principais aspectos. Assim, por meio de indução, conclusões genéricas são obtidas a partir de um conjunto particular de exemplos (FACELI et al., 2011).

Algoritmos de Aprendizado de Máquina tem sido amplamente utilizados em diversas tarefas, que podem ser classificadas como preditivas ou descritivas. Em tarefas de previsão, a dupla de características é composta por atributos de entrada (previsores) e um atributo de saída (atributo alvo), cujos valores podem ser estimados com base nos atributos de entrada. Assim, o objetivo é encontrar uma função (também chamada de modelo ou hipótese) a partir de um subconjunto dos dados, chamados de dados de treinamento, capaz de relacionar o conjunto de valores de entrada ao valor de sua saída. Em tarefas de descrição, a meta é explorar ou descrever um conjunto de dados. Não há no conjunto de dados um atributo de saída (FACELI et al., 2011).

A Figura 7 ilustra a hierarquia de aprendizado. No nível mais alto está o aprendizado indutivo. A seguir, tem-se o aprendizado supervisionado, associado a tarefas preditivas, e o aprendizado não supervisionado, relacionado a tarefas descritivas. Por sua vez, o aprendizado supervisionado está subdividido em tarefas de classificação (quando o valor predito é discreto) e tarefas de regressão (quando o valor predito é contínuo). Por fim, o aprendizado não supervisionado é dividido em tarefas de agrupamento, em que os dados são agrupados de acordo com a sua similaridade; associação, que consiste em encontrar padrões frequentes de associações entre os atributos de um conjunto de dados; e sumarização, cujo objetivo é encontrar uma descrição simples e compacta para o conjunto de dados (FACELI et al., 2011).

Figura 7 - Hierarquia de aprendizado



Fonte: Faceli et al. (2011)

Os termos Aprendizado de Máquina e Mineração de Dados são empregados como sinônimos em algumas situações. O que ocorre de fato é que a Mineração de Dados, que é uma etapa do processo de Descoberta de Conhecimento em Bases de Dados (*Knowledge Discovery in Databases* - KDD), depende fortemente de técnicas de Aprendizado de Máquina, reconhecimento de padrões e estatísticas para encontrar padrões de dados (FAYYAD et al., 1996).

2.3.1 Aprendizado supervisionado

O termo supervisionado está relacionado à simulação da presença de um "supervisor externo", que conhece a saída desejada para cada exemplo. Dessa forma, esse "supervisor" é capaz de avaliar a capacidade do modelo induzido de predizer a saída para novos exemplos. (FACELI et al., 2011).

Os algoritmos supervisionados utilizados nesse trabalho, bem como os principais aspectos relacionados são definidos a seguir.

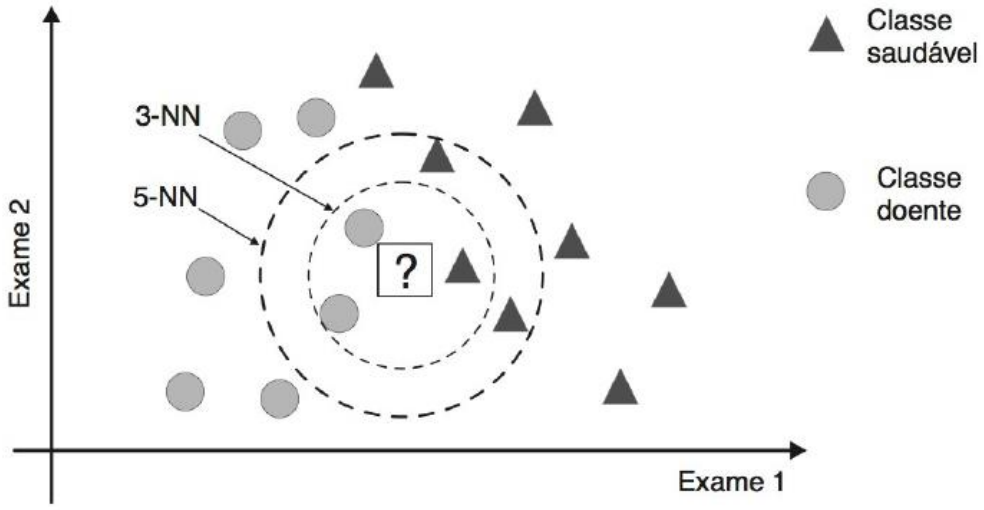
2.3.1.1 k-NN (K-Nearest Neighbors)

Algoritmo baseado em distância cuja hipótese consiste em considerar similares amostras que se encontram numa mesma região do espaço de entrada. Dessa forma, o classificador classifica um novo objeto com base nos exemplos de treinamento que estão próximos a ele (FACELI et al., 2011).

A quantidade de vizinhos que devem ser considerados na classificação é fornecida pelo usuário e é representada pelo parâmetro k, sendo k >= 1. Quando k é maior do que 1, para cada ponto de teste são obtidos k vizinhos, e cada vizinho vota em uma classe. O resultado do classificador é dado pela classe mais votada. Por essa razão, a fim de evitar empate, recomenda-se a escolha de um valor ímpar.

A Figura 8 ilustra, para um problema de classificação entre pessoas doentes e saudáveis, o impacto da escolha do parâmetro k no resultado do modelo. Neste exemplo, quando k = 3, o paciente foi classificado como doente. Quando o parâmetro é 5, a classificação aponta que o paciente é saudável. Conforme pode ser observado, a escolha de k não é trivial, pois impacta diretamente no resultado da classificação.

Figura 8 - Impacto do valor *k* no algoritmo *k*-NN



Fonte: Faceli et al. (2011)

Os principais aspectos positivos e negativos estão sumarizados no Quadro 1.

Quadro 1 - Aspectos positivos e negativos do k-NN

Aspectos Positivos	**Aspectos Negativos**
• Treinamento simples; • Naturalmente incremental, pois quando novos treinamentos estão disponíveis, basta adicioná-los a memória; • Aplicável em problemas complexos; • Permite modelar uma função objetiva complexa por meio de diversas funções objetivos locais para cada dado.	• Predição pode ser custosa quando há muitos dados no modelo de treinamento • Por ser baseado em distância, é impactado por atributos redundantes e irrelevantes; • Dificuldade de predição quando o conjunto de dados possui alta dimensionalidade, ou seja, muitas características.

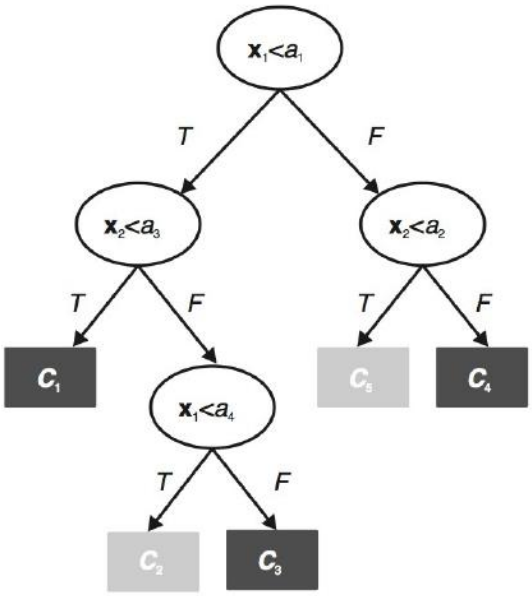
Fonte: Faceli et al. (2011)

2.3.1.2 Árvore de decisão

Árvores de decisão são modelos estatísticos de aprendizado supervisionado. Nestes, cada nó de decisão contém um teste para algum atributo de entrada e, cada ramo descendente corresponde a um possível valor deste atributo. O conjunto de ramos é distinto e cada folha está associada a uma classe. Assim, cada percurso da árvore, da raiz à folha, corresponde uma regra de classificação. No espaço definido pelos atributos, cada folha corresponde a um hiper-retângulo onde a interseção destes é vazia e a união é todo o espaço (GAMA, 2002).

Na Figura 9, x_1 e x_2 correspondem aos atributos de entrada. As condições avaliadas em cada nó estão expressas em seu interior. O nó raiz, por exemplo, corresponde a condição "$x_1 < a_1$". O resultado da avaliação de cada condição admite apenas um valor, T ou F. O caminho tomado a partir de um nó pode conduzir a um novo nó (condição) ou a uma folha (classe). Neste exemplo, as folhas C_1, C_2, C_3 e C_4 representam as classes do problema.

Figura 9 - Árvore de decisão



Fonte: Faceli et al. (2011)

2.3.1.3 Classificação com desbalanceamento de classe

De maneira geral, os algoritmos de aprendizado tradicionais utilizam, na fase de treinamento, dados com balanceamento de classe, ou seja, neste subconjunto de dados, os diferentes rótulos (classes) que constituem os valores possíveis do atributo de saída ocorrem de forma proporcional.

Cenários com conjuntos de treinamento complexos e altamente desbalanceados (desproporção entre as classes) têm apresentado dificuldade em diferenciar entre os grupos. A tendência é produzir modelos (ou regras) de classificação que favorecem a classe com maior probabilidade de ocorrência (majoritária), resultando em uma baixa taxa de reconhecimento para o grupo minoritário (CASTRO; BRAGA, 2011). Como resultado direto, têm-se modelos com alto grau de acurácia, mas que são ineficientes para identificar exemplos pertencentes a classe minoritária, que geralmente representa o foco de interesse.

Os principais aspectos relacionados ao impacto do desbalanceamento de classes são:

1. **A Teoria de Decisão Bayesiana fornece o modelo probabilístico fundamental para procedimentos de classificação de padrões** (BERGER, 1985) (BATHER, 2000). Os algoritmos tradicionais buscam minimizar a taxa de erro global de classificação. Nesse processo, assume-se que diferentes erros de classificação são igualmente importantes e que as distribuições das classes são relativamente equilibradas (MONARD; BATISTA, 2002) (HE; GARCIA, 2009).

2. **O nível de ruído associado ao dado.** Japkowicz e Stephen (2002) e Prati et al. (2004) conduziram estudos com dados sintéticos que mostram que para uma mesma razão de desbalanceamento, um aumento no nível de sobreposição das classes pode diminuir significativamente o número de classificações corretas para a classe minoritária. Utilizando dados reais, He e Shen (2007), Kubat et al. (1998) e Pearson et al. (2003) detectaram que conforme o nível de sobreposição apresentado pelas classes, regras de decisão obtidas pela simples minimização da taxa de erro global podem vir a perder sua capacidade de discriminação, classificando todos os exemplos como pertencentes à classe dominante.

3. **Falta de representatividade do grupo minoritário.** Estudos de Weiss (2004 e 2005) apontam que quando os exemplos da classe minoritária não são suficientes para representar as distribuições alvo (em termos de quantidade e disposição especial) no conjunto de treinamento, a capacidade de predição do modelo é comprometida, independente dos fatores de desbalanceamento e sobreposição do conjunto de dados.

Portanto, é importante salientar que o problema de classes desbalanceadas surge como uma propriedade inerente das soluções baseadas na taxa de erro global e que a intensidade do viés causado pelo grupo dominante está mais associada à complexidade dos dados (nível de sobreposição) do que a própria desproporção apresentada pelas classes. A partir dessas conclusões, é importante deixar claro que, para tarefas de classificação em que os dados possuem semelhança bem definidas e cujo agrupamento é bem separável no espaço de entrada, a influência do desbalanceamento deve ser mínima e, em geral, não deve prejudicar o reconhecimento da classe positiva (CASTRO; BRAGA, 2011).

Diversas pesquisas em aprendizado com classes desbalanceadas têm sido conduzidas com o objetivo de melhorar o número de acertos na predição de classes minoritárias. As abordagens utilizadas são divididas em duas categorias: pré-processamento de dados e adaptação em algoritmos de aprendizado. Na primeira categoria, reamostram-se os dados no espaço de entrada com o objetivo de modificar a distribuição das classes no conjunto de treinamento. Na segunda categoria, modificações são realizadas no algoritmo de classificação para considerar o desbalanceamento no espaço amostral de entrada.

A reamostragem no pré-processamento pode ocorrer por meio de:

- Sobreamostragem da classe minoritária, que ocorre quando novos registros da classe com menor número de representantes são gerados, seja pela repetição de exemplos já existentes no conjunto de entrada ou pela geração de dados sintéticos;

- Subamostragem da classe majoritária, que consiste em eliminar registros da classe com o maior número de ocorrências, seja de forma aleatória ou a partir de algum critério, como por exemplo, amostras redundantes, com ruído, ou próximas à fronteira de separação entre as classes. A eliminação aleatória pode acarretar na eliminação de exemplos representativos da classe majoritária, não sendo a mais recomendada.

- Qualquer combinação de ambas as técnicas.

No que se refere a abordagem baseada na adaptação de algoritmos de aprendizado, tem-se três classes de soluções:

- Baseada em reconhecimento, considera somente exemplos positivos durante o processo de aprendizado. Exemplos: *Auto-associator* (JAPKOWICZ, 2001), (MANEVITZ; YOUSEF, 2007), *One-class SVM* (SCHÖLKOPF et al., 2001), (RASKUTTI; KOWALCZYK, 2004), (MANEVITZ; YOUSEF, 2001), (BERGAMINI et al., 2009) e Detecção de novidades (MARKOU; SINGH 2003);

- Baseada em extensões do algoritmo de *Boosting*. O algoritmo consiste em agrupar em série classificadores com baixo poder de predição (*weak learners*) a fim de obter um modelo robusto. As extensões adicionadas atualizam iterativamente uma função de distribuição para o conjunto de treinamento de forma que maior/menor ponderação seja dada aos exemplos incorretamente/corretamente classificados. A maior parte dessas extensões é realizada através da incorporação de diferentes fatores (ou funções) de custo diretamente na função de distribuição, com o objetivo de distinguir a importância entre grupos e aumentar de forma mais significativa os pesos associados aos exemplos (erros/acertos) da classe minoritária. Essa estratégia, conhecida na literatura como *Cost-Sensitive Boosting*, permite o uso de amostras mais relevantes no treinamento das hipóteses, visando a obtenção de uma regra de decisão final que dá mais importância à classe de interesse (CASTRO; BRAGA, 2011). Como exemplos dessa classe tem-se: AdaCost (FAN et al., 1999), CSB1 e CSB2 (TING, 2000) e, AdaC1, AdaC2 e AdaC3 (SUN et al., 2007).

- Baseadas em modificações de funcionais riscos (função custo). A estratégia que tem sido mais usada é considerar a divisão do erro global entre as classes e incorporar funções de penalidade (ou fatores custo) distintas aos diferentes tipos de classificação. Essa técnica é comumente conhecida como Abordagem Sensível ao Custo e segue o princípio de minimização do custo esperado (risco global) da Teoria de Decisão Bayesiana (CASTRO; BRAGA, 2011).

2.3.1.4 *Ensemble* com Rusboost

Ensembles são algoritmos de aprendizagem que utilizam um conjunto de classificadores, ponderando o voto de suas previsões, para realizar a predição. Conforme Coelho (2006), o uso da abordagem *ensembles* tem sido bastante explorado por se tratar de uma técnica simples e capaz de aumentar a capacidade de generalização de soluções baseadas em aprendizado de máquina. No entanto,

para que um *ensemble* seja capaz de promover melhorias de desempenho, os seus componentes devem apresentar bons desempenhos individuais e, ao mesmo tempo, devem ter comportamentos diversos entre si.

O algoritmo *RUSBoost* (SEIFFERT et al., 2010) foi projetado para lidar com conjunto de dados desbalanceados. Trata-se de uma técnica híbrida que combina o algoritmo RUS (*Random Under Sampling*), cujo objetivo é promover a subamostragem aleatória da classe majoritária, a fim de, por *default*, igualar o seu número de amostras ao da classe minoritária, com o algoritmo AdaBoost, (FREUND; SCHAPIRE, 1996), um tipo de *ensemble* do tipo *boosting*, que combina classificadores, ponderando os seus resultados, para encontrar a melhor alternativa de classificação.

2.3.1.5 Detecção de ruído de classe

O efeito da presença de erros de rotulação (ruído) em conjuntos de dados de treinamento de classificadores bem como as técnicas para identificação e tratamento destes dados tem sido amplamente pesquisadas no campo do aprendizado de máquina. As duas abordagens mais comuns são focadas no desenvolvimento de algoritmos de classificação tolerantes a ruído e na limpeza de dados.

No que se refere a limpeza de dados, o uso de classificadores para identificação de dados rotulados incorretamente tem apresentado resultados expressivos. O trabalho de Frénay e Kabán (2014) apresenta extensa referência para os estudos realizados, que contemplam desde a aplicação de uma heurística simples, que considera a avaliação de um único classificador, até *ensembles* atuando por consenso ou maioria.

Conforme também pontuado por Frénay e Kabán (2014), muitos métodos baseados em *k*-NN foram propostos. Nesse caso, os vizinhos mais próximos identificam dados cuja remoção não faz com que outras instâncias sejam classificadas incorretamente.

Outras técnicas, como *Links* de Tomek (TOMEK, 1976), ENN (*Edited Nearest Neighbor rule*) (WILSON, 1972), C-Clear (MACHADO; LADEIRA, 2007), BED (*Boundry Elimination and Domination Algorithm*) (CASTRO et. al., 2009) também tem sido aplicadas para reduzir o nível de ruído presente nos dados. Adicionalmente, técnicas de agrupamento tem sido empregadas para detectar dados mal rotulados. O estudo de Bouveyron e Girard (2009) considera que dados cujos rótulos não são consistentes com o rótulo dos dados próximos provavelmente configuram erros de rotulação.

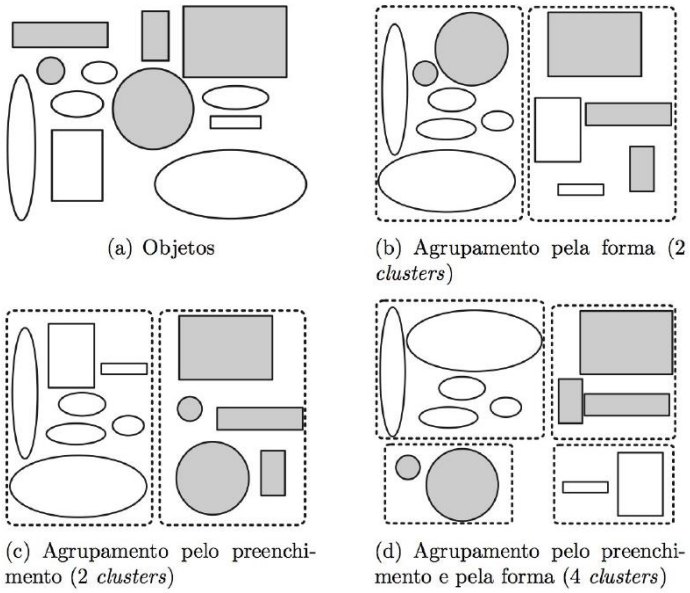
2.3.2 Aprendizado não supervisionado

2.3.2.1 Agrupamento

Agrupamento, também conhecida como clusterização, é uma técnica de aprendizado de máquina não supervisionado que consiste em agrupar dados baseado em sua similaridade, ou seja, os membros de um *cluster* são mais parecidos uns com os outros do que com membros de outros *clusters*.

A importância da escolha do critério de agrupamento é ilustrada na Figura 10. Em (a), tem-se todos os objetos a serem agrupados. Em (b), dois grupos (*clusters*) são formados ao utilizar como critério o agrupamento pela forma. Em (c), dois grupos, diferentes dos obtidos em (b), são formandos quando o critério utilizado é o preenchimento. Em (d), ao utilizar o critério preenchimento e forma, quatro grupos são formados.

Figura 10 - Objetos agrupados de diferentes maneiras



Fonte: Faceli et al. (2011)

Os principais modelos de agrupamento estão descritos a seguir:

Modelos de Conectividade: Baseiam-se na noção de que os dados mais próximos possuem mais semelhança entre si do que os mais distantes. O algoritmo de agrupamento hierárquico é um dos mais conhecidos e consiste em gerar, a partir de uma matriz de proximidade, uma sequência de partições aninhadas. O agrupamento hierárquico pode ser dividido em duas abordagens: a aglomerativa, que começa com *n clusters* com um único objeto e forma a sequência de partições agrupando os *clusters* sucessivamente, e a divisiva, que começa com um *cluster* com todos os objetos e forma a sequência dividindo os *clusters* sucessivamente. Um algoritmo hierárquico aglomerativo gera uma sequência de partições de *n* objetos em *k clusters* em que o nível 1 apresenta *n* clusters de um objeto e o nível *n* apresenta um *cluster* com todos os objetos. Assim, os dados são agrupados de forma que, se dois objetos são agrupados em algum nível, nos níveis mais altos eles continuam fazendo parte do mesmo grupo, construindo uma hierarquia de *clusters* (DUDA et al., 2001).

Modelos Centroide: A similaridade é definida pela proximidade do dado em relação a centroide do *cluster*. O *k-means* é um algoritmo popular que se enquadra nesta categoria. Neste, o conjunto de dados é particionado em *k clusters*, em que o valor

de *k* é fornecido pelo usuário. Esses *clusters* são formados de acordo com alguma medida de similaridade. O algoritmo *k-means* utiliza uma técnica de realocação iterativa, que pode convergir para um ótimo local. Existem várias versões do algoritmo, cada uma solucionando uma deficiência do algoritmo original. A versão tradicional do algoritmo encontra *clusters* compactos e de formato esférico. Mas existem versões, por exemplo, em que a distância de Mahalanobis pode ser utilizada para encontrar *clusters* híperelipsoidais (FACELI et al., 2011).

Modelos de Distribuição: Baseiam-se na noção de quão provável é que todos os dados no *cluster* pertençam à mesma distribuição. O algoritmo *Expectation-maximization* proposto por McLachlan e Krishnan (1997) é um exemplo desse modelo e usa distribuições normais multivariadas.

Modelos de Densidade: Assumem que os *clusters* são regiões com alta densidade, separadas por regiões com baixa densidade (ruídos). Um *cluster* definido como um componente denso conectado cresce em qualquer direção dada pela densidade (BERKHIN, 2002). Portanto, modelos desse tipo são capazes de obter *clusters* de formas arbitrárias. Exemplos populares de modelos de densidade são DBSCAN e OPTICS.

2.3.2.1.1 Silhueta

Conforme Kaufman e Rousseeuw (1990), o valor de silhueta para cada ponto é uma medida de quão semelhante esse ponto é em relação aos demais pontos em seu próprio *cluster*, quando comparado a pontos em outros *clusters*. O valor da silhueta para o *i-ésimo* ponto, *Si*, é definido (1),

$$\mathrm{Si} = \frac{\mathrm{bi} - \mathrm{ai}}{\max(\mathrm{ai}, \mathrm{bi})} \qquad (1)$$

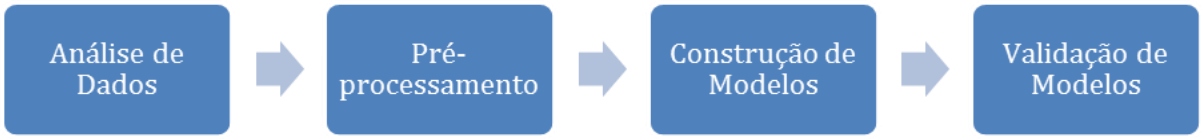
onde *ai* é a distância média do *i-ésimo* ponto para os outros pontos no mesmo conjunto como *i*, e *bi* é a distância média mínima do *i-ésimo* ponto para os pontos de um *cluster* diferente, minimizada sobre os *clusters*.

O valor da silhueta varia de -1 a +1. Um valor alto de silhueta, ou seja, mais próximo a 1, indica que o elemento *i* está bem adaptado ao seu próprio *cluster* e mal adaptado aos *clusters* vizinhos. Considera-se que a solução de agrupamento é apropriada se a maioria dos pontos tiver um valor de silhueta alto. O critério de avaliação de agrupamento de silhueta pode ser usado com qualquer algoritmo de distância.

2.3.3 Etapas do aprendizado de máquina

As principais etapas no processo de Aprendizado de Máquina são: Análise de Dados, pré-processamento, construção e validação de modelos. Na primeira etapa, busca-se conhecer as características do conjunto de dados disponível com o intuito de avaliar o esforço que deverá ser empreendido nas etapas seguintes, de modo que se tenham dados adequados para a realização do estudo. Durante o pré-processamento, são realizadas atividades de transformação e limpeza do dado, além da extração das características representativas para a modelagem do problema. Na etapa seguinte, construção de modelos, são escolhidos os hiperparâmetros que serão utilizados pelo algoritmo de aprendizagem e a forma de amostragem mais adequada para o treinamento e avaliação dos resultados. Por fim, valida-se o modelo gerado com o objetivo de avaliar a sua robustez na predição de dados novos. A Figura 11 ilustra as etapas citadas.

Figura 11 - Etapas do aprendizado de máquina



Fonte: Autoria própria

Na fase de pré-processamento, é importante uniformizar a escala do dado, em especial quando se usa algoritmos de distância. A *z-score* é uma transformação muito empregada e representa a quantidade de desvios padrão que o elemento possui em relação a média. Dessa forma, o conjunto de entrada transformado possui média 0 e variância unitária.

Ainda sobre o pré-processamento, é recorrente a existência, em especial quando do uso de dados reais, de dados atípicos, que destoam dos demais, os *outliers*. Utilizar esses dados na construção de um modelo pode acarretar em baixa performance. A técnica de remoção de *outlier* utilizada considera que qualquer dado que esteja 1,5 vezes a amplitude interquartil acima do quartil superior ou 1,5 vezes a amplitude interquartil abaixo do quartil inferior é espúrio e deve ser removido.

Quanto à construção de modelos, é fato que grande parte dos algoritmos de Aprendizado de Máquina possuem parâmetros (também chamados de hiperparâmetros), cujos valores devem ser especificados pelo usuário. Em geral, esses valores influenciam diretamente o desempenho de modelos induzidos, o que pode ser entendido como uma limitação das técnicas de Aprendizado de Máquina (ROSSI, 2009).

Escolher a parametrização adequada, de maneira manual, requer muito conhecimento tácito e tempo na condução de testes. Assim, é necessário empregar técnicas automatizadas para otimizar esse processo. A técnica *grid search* automatiza esta tarefa testando cada combinação de valores de parâmetro. Os valores possíveis são definidos para uma faixa específica, e do ponto inicial ao final, o dado varia a passos geométricos. Ao final, a melhor combinação de parâmetros é apresentada.

Calcular o desempenho preditivo do modelo nos mesmos objetos empregados para o seu treinamento produz estimativas otimistas, uma vez que todos os algoritmos de Aprendizado de Máquina tentam melhorar de alguma forma o seu desempenho nesses objetos na fase indutiva. O uso do mesmo conjunto de exemplos no treinamento e avaliação do preditor é conhecido como ressubstituição (FACELI et al., 2011).

Assim, deve-se utilizar métodos de amostragem alternativos para obter estimativas de desempenho mais confiáveis, definindo subconjuntos disjuntos de treinamento e testes. Os dados de treinamento são utilizados para indução e ajuste do modelo, e os dados de teste simulam a apresentação de novos exemplos ao preditor, os quais não foram vistos na indução. Um método bastante empregado quando da existência de poucos dados é a validação cruzada *k-fold cross-validation.* Neste, o conjunto de exemplos é dividido em *k* subconjuntos de tamanho aproximadamente igual. Então, os objetos de *k* - 1 partições são utilizados na fase de treinamento, e a partição restante é utilizada para o teste. Esse processo é iterativo e repetido *k* vezes. O desempenho final do preditor é a média de desempenho de cada subconjunto de teste (FACELI et al., 2011).

2.3.4 Avaliação de resultado

Tradicionalmente, a métrica usada na avaliação e seleção de modelos de classificação é a acurácia (ou taxa de erro) estimada em relação a um dado conjunto de teste. Essa metodologia é justificada pela formulação padrão do problema do aprendizado supervisionado que visa a minimização da probabilidade do erro global (CASTRO; BRAGA, 2011). Por definição, a acurácia é a relação entre a quantidade de classificações corretas e a quantidade total de classificações. No entanto, conforme pontuado nos trabalhos de Bradley (1997), Provost e Fawcett (1997, 1998), Maloof (2003), Cortes e Mohri (2004), Sun et al. (2007), o uso de tal métrica em cenários com desbalanceamento é inadequado, pois não distingue os erros (ou acertos) cometidos para cada classe. Para estes casos, a matriz de confusão é uma alternativa interessante, pois permite identificar quais classes o algoritmo de aprendizado possui maior dificuldade de predição.

O Quadro 2 mostra a matriz de confusão para um classificador binário. As linhas representam as classes verdadeiras, e as colunas, as classes preditas pelo classificador. Logo, cada elemento m_{ij} de uma matriz de confusão M_c apresenta o número de exemplos da classe *i* classificados como pertencentes à classe *j*. Para *k* classes, M_c tem a dimensão *k* x *k*. A diagonal apresenta os acertos do classificador,

enquanto os outros elementos correspondem aos erros cometidos em suas predições (FACELI et al., 2011).

Quadro 2 - Matriz de confusão para um classificador binário

	PREDIÇÃO POSITIVA	PREDIÇÃO NEGATIVA
CLASSE POSITIVA	Verdadeiro positivo (VP)	Falso negativo (FN)
CLASSE NEGATIVA	Falso positivo (FP)	Verdadeiro negativo (VN)

Fonte: Faceli et al. (2011)

A partir da matriz de confusão, uma série de outras medidas de desempenho pode ser derivadas. Entre elas, tem-se (MONARD; BARANAUSKAS, 2003):

- **Taxa de erro total (*err*):** dada pela soma dos valores da diagonal secundária da matriz, dividida pela soma dos valores de todos os elementos da matriz, conforme equação (2).

$$err = \frac{FP + FN}{n} \tag{2}$$

- **Taxa de acerto ou acurácia total (*ac*):** calculada pela soma dos valores da diagonal principal da matriz, dividida pela soma dos valores de todos os elementos da matriz, conforme equação (3).

$$ac = \frac{VP + VN}{n} \tag{3}$$

- **Taxa de falsos negativos (TFN):** corresponde a proporção de exemplos da classe positiva incorretamente classificados pelo preditor, conforme equação (4).

$$TFN = \frac{FN}{VP + FN} \tag{4}$$

- **Taxa de falsos positivos (TFP):** corresponde a proporção de exemplos da classe negativa incorretamente classificadas pelo preditor, conforme equação (5).

$$TFP = \frac{FP}{FP + VN} \tag{5}$$

- **Precisão (*prec*):** proporção de exemplos positivos classificados corretamente entre todos aqueles preditos como positivos, conforme equação (6).

$$prec = \frac{VP}{VP + FP} \tag{6}$$

- **Sensibilidade ou revocação, também chamada de taxa de verdadeiros positivos (*sens, rev* ou TVP):** corresponde à taxa de acerto na classe positiva, conforme equação (7).

$$sens = rev = TVP = \frac{VP}{VP + FN} \tag{7}$$

- **Especificidade, também chamada de taxa de verdadeiros negativos (*esp ou TVN*):** corresponde à taxa de acerto na classe negativa. Seu complemento corresponde à taxa TFP, conforme equação (8).

$$esp = TVN = \frac{VN}{VN + FP} \tag{8}$$

Além das taxas de erro/acerto para cada classe, outras métricas têm sido frequentemente adotadas com o objetivo de fornecer avaliações mais adequadas para aplicações desbalanceadas (SUN et al., 2007), (HE; GARCIA, 2009). Em geral, esses critérios focam na detecção da classe positiva, como a métrica $F_{measure}$ (9), ou consideram com mesma relevância a discriminação de ambas as classes, como a métrica G_{mean} (10).

$$F_{measure} = \frac{(1 + \beta).rev.prec}{\beta^2.rev + prec} \tag{9}$$

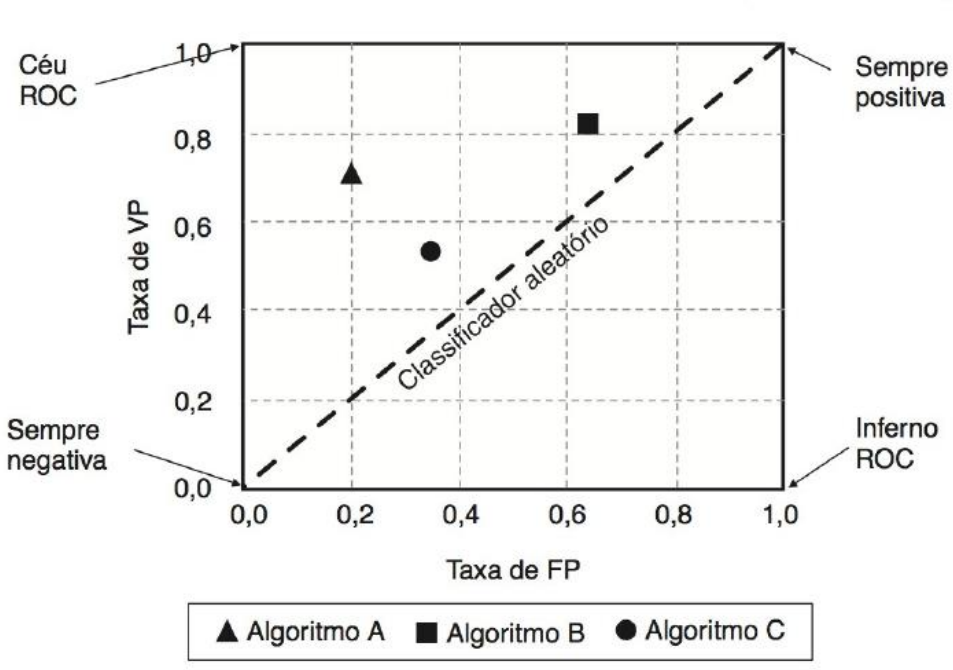
$$G_{mean} = \sqrt{rev.TVN} \tag{10}$$

No cálculo da métrica $F_{measure}$, o fator de ponderação β é utilizado para ajustar a importância relativa entre a precisão e revocação. Ao atribuir peso igual a 1 para esse fator, é dado o mesmo grau de importância a estas duas métricas.

Apesar das métricas extraídas da matriz de confusão serem mais eficientes que a acurácia na avaliação de classificadores em cenários desbalanceados, elas não permitem comparar seus desempenhos sobre uma faixa de valores de distribuições a priori ou custos de erros de classificação (CASTRO; BRAGA, 2011). Para contornar essa limitação, a curva ROC (*Receiver Operating Characteristic*) tem sido empregada na avaliação e comparação de modelos (SPACKMAN, 1989), (FAWCETT, 2004, 2005), (PRATI et al., 2008).

O gráfico ROC, ilustrado na Figura 12, possui como eixos X e Y, respectivamente, a taxa de falsos positivos (TFP) e taxa de verdadeiros positivos (TVP). Alguns pontos desse gráfico merecem destaque. O ponto (0,0) representa a estratégia de sempre classificar um exemplo como negativo. O ponto (1, 1) corresponde a classificação sempre como positivo. O ponto (0, 1) representa o modelo perfeito, em que todos os exemplos positivos e negativos são classificados corretamente. De forma inversa, o ponto (1, 0) sempre faz predições incorretas. A linha entre os pontos (0,0) e (1, 1) corresponde a um modelo de comportamento estocástico. Assim, pode-se afirmar que os pontos acima dessa linha possuem desempenho melhor que o aleatório. Portanto, os melhores modelos estão situados mais acima e a esquerda do gráfico.

Figura 12 - Gráfico ROC com três classificadores

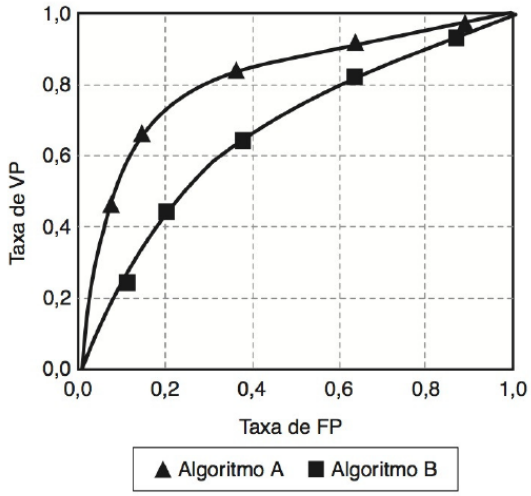


Fonte: Faceli et al. (2011)

Uma característica dos sistemas de classificação é que estes geram como saída valores contínuos, situados dentro de um intervalo. Durante a criação do modelo, define-se um valor que atuará como um limiar de decisão, ou seja, valores acima do limiar são atribuídos a uma das classes e abaixo a outra. Logo, é possível concluir que cada limiar pode gerar uma matriz de confusão distinta. A curva ROC é gerada variando o limiar de decisão e, para cada ponto deste, calcula-se a taxa de falsos positivos (TFP) e taxa de verdadeiros positivos (TVP).

A Figura 13 ilustra duas curvas ROC, correspondente ao desempenho dos algoritmos hipotéticos A e B. Para definir qual algoritmo possui melhor desempenho, é necessário avaliar se há ou não intersecção entre as suas curvas. Não ocorrendo, a curva que mais se aproxima do ponto (0, 1) é a de melhor desempenho. Havendo interseção, cada algoritmo possui uma região que é melhor que outro.

Figura 13 - Curva ROC de dois classificadores



Fonte: Faceli et al. (2011)

A área abaixo da curva ROC (*Area Under the ROC Curve* - AUC) fornece uma medida geral da capacidade de discriminação do classificador que é insensível aos custos de classificação e probabilidades a priori (HANLEY; MCNEIL, 1982). Sendo a AUC uma porção da área de um quadrado com lado igual a 1 (valor máximo possível para as medidas dos seus eixos), tem-se que o seu valor sempre estará entre 0 e 1. Classificadores perfeitos possuem área igual a 1 (HAND, 2009).

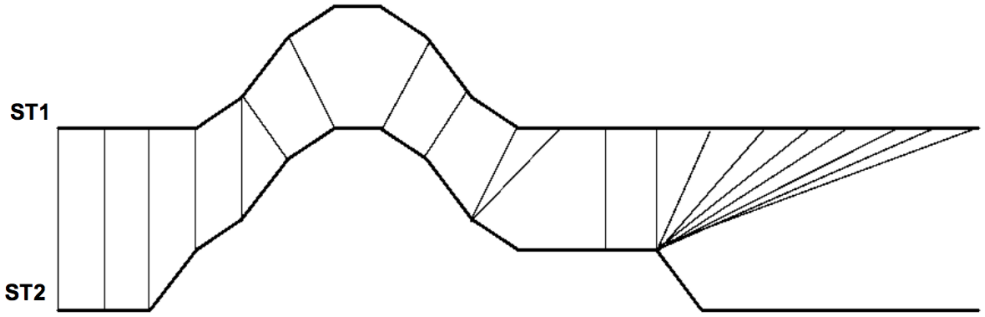
Classificadores perfeitamente aleatórios produzem a linha diagonal entre os pontos (0, 0) e (1, 1), com área 0,5 (FAWCETT, 2005). Não existe consenso quanto ao valor mínimo de AUC que um classificador deve apresentar para ser considerado bom ou excelente. De maneira geral, isso é dependente do problema a ser resolvido e a avaliação do quão bom é o resultado é feita comparando com referências anteriores.

2.4 *DYNAMIC TIME WARPING*

Na análise de séries temporais, a distorção do tempo dinâmico, mais conhecida como *Dynamic Time Warping* (DTW), é utilizada para medir a semelhança entre duas sequências temporais, que podem variar de velocidade. O objetivo dessa técnica é encontrar o melhor alinhamento entre as duas séries, utilizando para isso deformações temporais.

Supondo duas sequências numéricas (*a1, a2, ..., an*) e (*b1, b2, ..., bm*), que não necessariamente possuem o mesmo tamanho. O algoritmo inicia calculando as distâncias locais entre os elementos das duas sequências, utilizando, por exemplo, a distância euclidiana. Isso resulta em uma matriz de distâncias com *n* linhas e *m* colunas. A partir desta, aplica-se programação dinâmica para encontrar a menor dissimilaridade entre as séries. A matriz gerada ao final desse processo é utilizada para encontrar o menor caminho (caminho distorcido) que garanta o alinhamento entre as séries. Por fim, a distância entre as duas séries é calculada somando as distâncias de todos os elementos que compõem o caminho distorcido. A Figura 14 mostra a comparação entre duas séries temporais com DTW, cabe observar que a correspondência entre os pontos não é 1:1.

Figura 14 - Comparação de duas séries com o DTW mostrando a distorção



Fonte: Santos (2015)

2.5 ALGORITMO GENÉTICO

Algoritmo evolutivo baseado no mecanismo de seleção natural de Darwin e na genética Mendeliana. É amplamente utilizado em problemas de otimização combinatória, onde diferentes soluções são combinadas através de regras de probabilísticas para gerar outras que se aproximam do ótimo. (ALMEIDA; SALLES, 2016).

Conforme pontuado por Almeida e Salles (2016), é importante definir a seguinte terminologia empregada:

- Cromossomo ou individuo: vetores que representam as variáveis do problema;
- Gene: unidade básica do cromossomo, valor que descreve uma determinada variável;
- População: conjunto de indivíduos, inserido no espaço de busca do problema;
- Geração: número de iterações que o algoritmo genético executa;
- Operadores genéticos: operações executadas sobre os indivíduos com o objetivo de garantir a evolução da espécie e explorar novos espaços de busca;

- Função de aptidão, objetivo ou *fitness*: é a função que se deseja otimizar. Avalia, com base nas características do problema, a aptidão do indivíduo para resolvê-lo;

- Reprodução: consiste na cópia do indivíduo de uma geração para a outra;

- Recombinação ou *crossover*: troca aleatória de informações entre indivíduos;

- Mutação: troca aleatória de características de um gene em determinados indivíduos.

A partir de uma população inicial, o algoritmo genético evolui iterativamente, ao longo das gerações, selecionando os melhores indivíduos da geração atual (avaliados por meio da função objetivo), reproduzindo novos indivíduos e inserindo-os na geração seguinte. Ao final, sobrevivem os indivíduos mais aptos a resolver o problema.

É importante salientar que a seleção de indivíduos pode ocorrer de diversas maneiras, inclusive de forma combinada. Destacam-se duas abordagens:

- Seleção por roleta: Cada indivíduo ocupa uma posição em uma roleta, com tamanho proporcional a sua aptidão para solução do problema. Em seguida, faz-se simulações de giro da roleta por n vezes, sendo n o número de indivíduos que se manterão na etapa seguinte, a fim de selecioná-los. Os indivíduos mais aptos possuem maior probabilidade de seleção.

- Técnica elitista: Garante que um determinado número de indivíduos com melhor desempenho seja passado para a geração seguinte. É bastante usado em conjunto com o algoritmo de roleta.

3 METODOLOGIA

A proposta desse trabalho é otimizar o processo de partida do conjunto de bombeio centrifugo submerso a quente, fase Rampa de Aceleração. Para tal, diversos desafios necessitam ser transpostos, tais como: indisponibilidade de um modelo matemático para realização de simulações; poucos dados disponíveis, pois trata-se de um evento esporádico; desbalanceamento de classes, visto que a mudança de patamar operacional ocorre com menos frequência que a manutenção da condição; presença de ruído de classificação, tendo em vista que o processo é manual e está sujeito a experiência de quem o executa.

Portanto, faz-se necessário empregar uma metodologia que garanta uma coleta eficiente dos dados reais necessários ao estudo, que este seja estruturado de forma adequada para compreensão de suas características e uso em algoritmos de aprendizado de máquina e que os modelos possam ser adequadamente gerados e validados. Para tanto, a metodologia definida consiste das seguintes etapas: Coleta de Dados, Estruturação dos Dados, Análise de Dados, Pré-Processamento, Extração de Características e Construção de Modelos. Estas estão ilustradas na Figura 15.

Figura 15 - Etapas da metodologia

Fonte: Autoria própria

3.1 COLETA DE DADOS

O conjunto de bombeio centrífugo submerso submarino não possui modelo matemático disponível que permita gerar dados por meio de simulação. Por isso, este estudo utiliza dados reais. As variáveis de interesse correspondem aos sensores e atuadores listados a seguir:

- Corrente do Motor;
- Temperatura do Motor;
- Pressão na Descarga da Bomba;
- Pressão na Sucção da Bomba;
- Diferencial de Pressão na Bomba;
- Pressão no Fundo do Poço;
- Pressão a Montante do *Choke;*
- Frequência do Inversor;
- Percentual de Abertura da Válvula *Choke.*

Os dados dessas variáveis estão armazenados em um historiador, de modo que o histórico de partidas está preservado. No entanto, neste estão persistidos apenas o identificador da variável, o valor e, a data e hora da ocorrência. Não há nos registros qualquer identificação de eventos de interesse, como por exemplo, a ocorrência do procedimento de partida.

Assim, o primeiro esforço desse trabalho foi empreendido na identificação do início e do fim de eventos de partida. O valor das variáveis Frequência do Inversor e Corrente do Motor foram utilizados para identificar o início da operação. Em resumo, estes apresentam valor 0 quando o equipamento está desligado. A identificação do final da partida atende a um critério mais subjetivo, tendo em vista que o equipamento continua a operar e, eventualmente, a operação se mantém em

patamar diferente de quando o equipamento foi desligado. Por essa razão, uma comparação simples dos valores correntes das variáveis, Frequência do Inversor e Percentual de Abertura da Válvula C*hoke* com os valores do momento anterior ao desligamento do equipamento não pode ser realizada. Dessa maneira, foi arbitrado, com base na análise dos dados e em conversa com especialistas, que para os dados históricos analisados, a partida está encerrada quando não há alteração do valor dessas variáveis por 90 minutos.

Uma vez identificadas as partidas, aplicou-se um filtro com base no valor da temperatura do fluido, a fim de identificar apenas as que seguiram o procedimento a quente. Como as partidas não são um evento rotineiro, estas estão espaçadas ao longo de vários anos de operação e possuem poucas ocorrências. Diante desse cenário, selecionou-se para esse estudo de caso um poço com quantidade de partida suficiente para caracterização do domínio do problema.

Ao analisar os dados selecionados, verificou-se que o intervalo de tempo entre os registros não é constante e que não há, necessariamente, uma correspondência entre a data e hora da coleta das variáveis. Os quadros 3 e 4 apresentam, respectivamente, os resultados de uma consulta às variáveis Corrente do Motor e Frequência do Inversor para um mesmo período. A coluna Intervalo de Tempo entre Registros explicita a variação descrita acima. Também é possível observar na coluna Horário de Coleta do Dado a não correspondência entre os horários de coleta desses sensores. Também cabe observar uma diferença na quantidade de dados retornados.

Quadro 3 - Registros armazenados no historiador para a variável corrente do motor

Horário de Coleta do Dado	Corrente do Motor (A)	Intervalo de Tempo entre Registros (hh:mm:ss)
17:32:11	57	-
17:32:26	61	00:00:15
17:32:41	58	00:00:15
17:33:36	61	00:00:55
17:34:11	58	00:00:35
17:34:56	60	00:00:45
17:35:06	58	00:00:10
...	...	...
20:27:37	93	-
20:27:47	96	00:00:10
20:28:07	93	00:00:20
20:28:32	96	00:00:25
20:28:47	90	00:00:15
20:29:02	94	00:00:15

Fonte: Autoria própria

Quadro 4 - Registros armazenados no historiador para a variável Frequência do Inversor

Horário de Coleta do Dado	Valor (Hz)	Intervalo de Tempo entre Registros (hh:mm:ss)
17:36:36	43	-
17:36:41	44	00:00:05
17:58:11	46	00:21:30
17:58:21	46	00:00:10
18:29:07	47	00:30:46
18:29:17	48	00:00:10
18:33:37	50	00:04:20
20:28:47	50	01:55:10

Fonte: Autoria própria

Em levantamento realizado com especialistas foi identificado que o processo possui dinâmica lenta e que a tomada de decisão é realizada a cada minuto. Assim, como estratégia adotada para a uniformização da frequência de amostragem, foi realizada uma nova coleta de dados no historiador, utilizando como opção a interpolação linear a cada minuto. A data de início e data de fim foi igualada e fixada para todas as variáveis, de modo a obter correspondência de horário de coleta entre elas. A Figura 16 ilustra o processo de coleta de dados descrito.

Figura 16 - Etapas do processo de coleta de dados

Fonte: Autoria própria

3.2 ESTRUTURAÇAO DOS DADOS

Conforme citado brevemente na seção 3.1, os dados das variáveis de interesse estão armazenados em um sistema historiador. Internamente em sua organização, cada sensor possui um identificador único denominado *tag* e, associado a este, são persistidos os valores com as respectivas data e hora da coleta. O Quadro 5 ilustra a estrutura de armazenamento descrita.

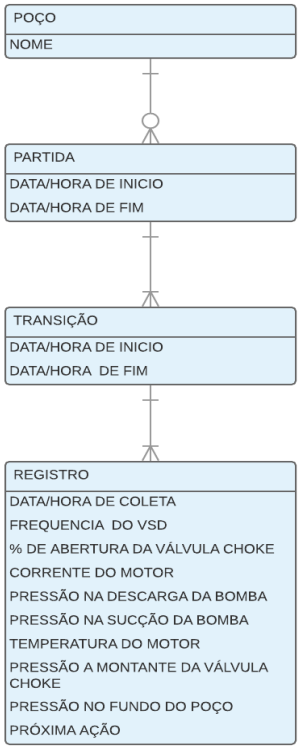
Quadro 5 - Estrutura de armazenamento do historiador

Tag	**Data/hora de Coleta do Dado**	**Valor**
Tag1	t1	v1
Tag1	t2	v2
Tag1	t3	v3
Tag2	t1	v4
...	...	...
Tag3	t1	v5
Tag3	t2	v6
...	...	...

Fonte: Autoria própria

Conforme pode ser observado, cada *tag* é independente, o que dificulta uma análise de dados consistente. No entanto, a interpolação realizada na etapa de coleta de dados gerou um indexador comum a todas os *tags*: a data e hora da coleta dos dados. Dessa forma, é possível agrupar os dados de modo tabular, gerando estruturas não apenas mais simples de compreender conceitualmente, como também mais adequadas para a exploração dos dados e uso pelos algoritmos de classificação. Conceitualmente, o problema foi modelado conforme Figura 17.

Figura 17 - Modelagem conceitual da partida



Fonte: Autoria própria

A entidade POÇO apenas identifica o poço ao qual a partida se refere. A entidade PARTIDA indica a data e hora de início e fim da partida. Cada partida é composta por transições (entidade TRANSIÇÃO), cujo objetivo é agrupar os registros compreendidos entre as mudanças no *set point* das variáveis Frequência do Inversor e Percentual de Abertura da Válvula *Choke.* A entidade REGISTRO reúne o valor de todas as variáveis indexadas pela data e hora da coleta. O atributo PRÓXIMA AÇÃO indica a ação que será tomada com relação a mudança do patamar operacional das

variáveis Frequência do Inversor e Percentual de Abertura da Válvula *Choke,* tendo como valores possíveis: M (Mantém Configuração Atual), F (Incrementa Frequência) ou C (Incrementa Percentual de Abertura da Válvula *Choke*).

3.3 ANÁLISE DE DADOS

Dada a interpolação por minuto realizada na seção 3.1, é possível afirmar que a quantidade de registros de uma partida corresponde a sua duração em minutos. Uma outra característica pertinente é que cada transição possui apenas um registro com rótulo Incrementa Frequência do Inversor (Próxima Ação = F) ou Incrementa Percentual de Abertura da Válvula *Choke* (Próxima Ação = C). Todos os demais são rotulados como Mantém Configuração Atual (Próxima Ação = M).

A composição da amostra de dados está detalhada na Tabela 1. Considerando a quantidade total de registros das oito partidas disponíveis, é possível observar uma desproporção entre os registros rotulados como Mantém Configuração Atual (Próxima Ação = M), Incrementa Frequência do Inversor (Próxima Ação = F) e Incrementa Percentual de Abertura da Válvula *Choke* (Próxima Ação = C) na razão 1:0,09:0,03, o que indica que o problema estudado possui desbalanceamento de classes.

Tabela 1 - Composição da amostra de dados por partida

Nome	Quantidade total de registros	Número de transições por alteração na Frequência do Inversor	Número de transições por alteração no Percentual de Abertura da Válvula *Choke*
Partida1	159	5	2
Partida2	90	12	10
Partida3	116	11	1
Partida4	107	13	5
Partida5	105	9	1
Partida6	157	11	4
Partida7	65	5	0
Partida8	62	4	3
TOTAL:	861	70	26

Fonte: Autoria própria

Também é relevante pontuar acentuada variância nos dados relativos a quantidade total de registros por partida e a quantidade de transições tanto por alteração na Frequência do Inversor quanto por alteração no Percentual de Abertura da Válvula *Choke.* A Tabela 2 expõe as estatísticas de duração das transições como um todo, bem como segregadas por alteração na Frequência do Inversor e por alteração no Percentual de Abertura da Válvula *Choke*. A análise cuidadosa de tais dados sugere:

- Necessidade de investigação quanto a presença de ruído de classificação, caracterizada pela manutenção do *set point* das variáveis Frequência do Inversor e Percentual de Abertura da Válvula *Choke* por mais tempo que o necessário;

- Limpeza dos dados a fim de descartar transições com duração inadequada para o processo de treinamento/validação do classificador;

- Remoção da classificação de registros com rótulo Incrementa Percentual de Abertura da Válvula *Choke* do escopo desse estudo, pois a quantidade de dados disponível não é suficiente para uma avaliação criteriosa.

Tabela 2 - Análise estatística da duração das transições

Medida Estatística	**Todas as Transições da Amostra (minutos)**	**Transições por Alteração na Frequência do Inversor (minutos)**	**Transições por Alteração no Percentual de Abertura da válvula *Choke* (minutos)**
Média	8,97	10,4	4,04
Mediana	7,00	8,00	3,00
Mínimo	1,00	1,00	1,00
Máximo	43,00	43,00	15,00
Desvio Padrão	7,72	7,97	4,01
Moda	1,00	7,00	1,00

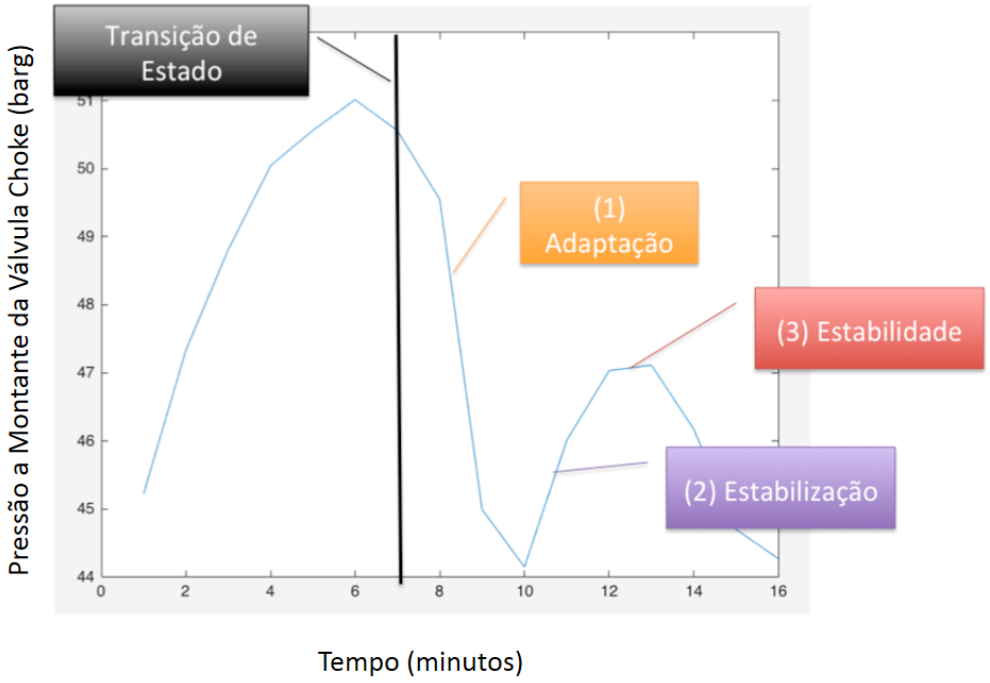
Fonte: Autoria própria

Para a investigação quanto a presença de ruído de classificação, é necessário observar o comportamento da variável controlada quando da alteração de *set point* da variável manipulada. A Figura 18 mostra o comportamento da variável Pressão a Montante da Válvula *Choke* após o incremento na variável manipulada Percentual

de Abertura da Válvula *Choke,* esse momento é assinalado na imagem pela reta rotulada como Transição de Estado.

Verificando a resposta do processo, é possível assinalar 03 fases distintas: (1) fase de adaptação, que ocorre de forma imediatamente posterior a alguma alteração nas variáveis manipuladas e representa a reação do processo à mudança realizada, (2) fase de estabilização, que marca a busca do processo pelo equilíbrio, e por fim, (3) a fase de estabilidade, em que o processo se encontra controlado e pronto para operar em um novo patamar operacional.

Figura 18 - Análise do comportamento da variável controlada Pressão a Montante da Válvula *Choke* após mudança no *set point* da variável manipulada Percentual de Abertura da Válvula *Choke*



Fonte: Autoria própria

Considerando as fases apresentadas, infere-se que a fase de estabilidade deve ser a mais breve possível e, portanto, identificá-la adequadamente e avaliar a sua duração permite detectar se o *set point* das variáveis Frequência do Inversor ou Percentual de Abertura da Válvula *Choke* foi mantido por mais tempo que o necessário. Para tal, foi empregada a técnica de agrupamento. Uma vez definido o uso desta, foi preciso estabelecer:

1. Variáveis de entrada consideradas como característica;
2. Número de *clusters;*
3. Algoritmo de agrupamento;
4. Algoritmo de distância;
5. Método aglomerativo;
6. Medida de similaridade para avaliar o resultado.

As variáveis de entrada Pressão a Montante da Válvula *Choke* e Diferencial de Pressão na Bomba (Pressão na Descarga - Pressão na Sucção) foram selecionadas, pois consta no manual do equipamento disponibilizado pelo fabricante que estas devem balizar as decisões relacionadas ao processo de partida. O número de *clusters* será igual a 3, dada a sua correspondência com a quantidade de fases da reposta do processo, vide Figura 18. Os demais parâmetros estão listados no Quadro 6.

Quadro 6 - Parâmetros empregados na tarefa de agrupamento

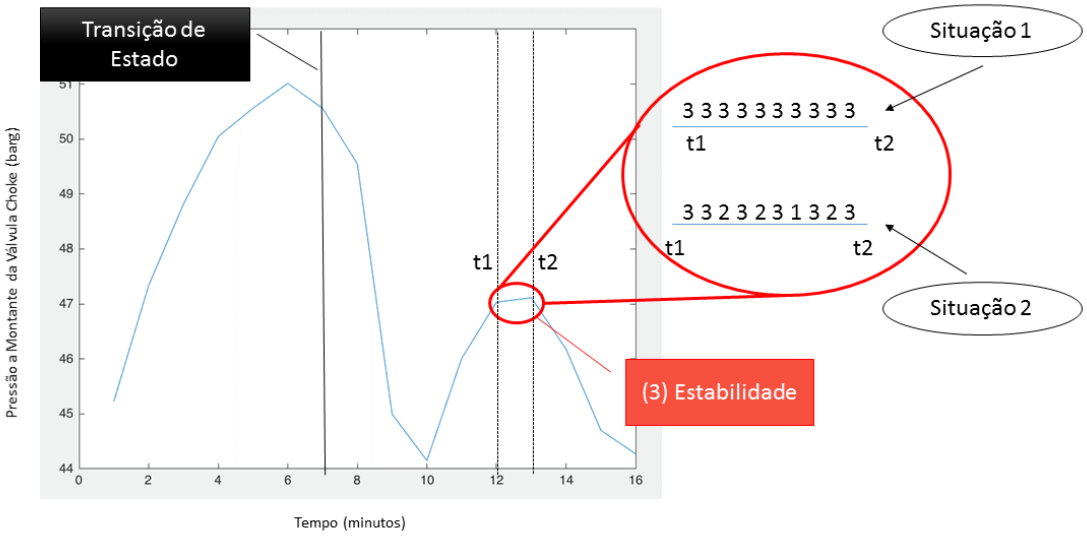
Algoritmo de agrupamento	Hierárquico aglomerativo
Algoritmo de distância	Distância euclidiana
Método aglomerativo	Ligação simples
Medida de similaridade para avaliar o resultado	Silhueta

Fonte: Autoria própria

Após aplicar o agrupamento, identificou-se o grupo de interesse verificando o *cluster* ao qual o último registro da transição, que indica a mudança de *set point* em alguma variável de interesse, pertencia. A seguir, buscou-se o primeiro registro pertencente a esse *cluster*. Na Figura 19, o instante *t1* corresponde ao primeiro registro do *cluster* e o instante de tempo *t2* corresponde ao final da transição. Assim todos os registros compreendidos nesse intervalo de tempo pertencem a fase de estabilidade.

A etapa seguinte consistiu em avaliar se todos os registros dessa fase pertenciam ao mesmo *cluster*. Por exemplo, a situação 1, assinalada na Figura 19, indica o cenário em que essa condição é satisfeita. Já na situação 2, verifica-se que não há homogeneidade nos *clusters* aos quais os registros do intervalo analisado pertencem. Em caso de satisfação da condição, foi avaliada a similaridade entre os registros utilizando a medida de similaridade silhueta. Nesse estudo, considerou-se que a transição poderia ter ocorrido a partir do primeiro registro do *cluster* que satisfaça a um valor mínimo de silhueta e, portanto, para estas amostras, o rótulo original dado pelo operador (Mantém a Configuração Atual) é um ruído de classificação.

Figura 19 - Detecção da fase de estabilidade utilizando agrupamento



Fonte: Autoria própria

Após a execução de diversos testes, identificou-se que o limite de 0,7 para silhueta é adequado ao problema. A Tabela 3 mostra a quantidade de transições com erro de rotulação por partida e a quantidade de registros incorretamente rotulados. Portanto, é possível concluir que de fato o conjunto de dados disponível possui registros com ruído de classificação.

Tabela 3 - Resultado da tarefa de agrupamento

Nome	Número de transições com ruído de classificação	Quantidade de registros rotulados incorretamente
Partida1	6	39
Partida2	4	24
Partida3	5	91
Partida4	2	14
Partida5	5	16
Partida6	4	15
Partida7	3	7
Partida8	0	0

Fonte: Autoria própria

3.4 PRÉ-PROCESSAMENTO

A primeira etapa da fase de pré-processamento objetivou remover transições com duração inferior a um limite estabelecido. Para seleção adequada desse limiar, recorreu-se ao manual do equipamento, cuja definição é que transições não devem ter duração inferior a 8 minutos, e as estatísticas apresentadas na seção 3.3, cuja constatação é que a moda de duração das transições por alteração na Frequência do Inversor é 7 minutos. Ao ponderar sobre esses valores, e tendo em vista que o objetivo do trabalho é otimizar o processo através da redução do tempo das partidas, considerou-se como limite 6 minutos. A Tabela 4 mostra a composição da amostra após essa etapa.

Tabela 4 - Composição da amostra por partida após eliminação de transições com duração inferior ao limite

Nome	Quantidade total de registros	Número de transições por alteração na Frequência do Inversor
Partida1	134	5
Partida2	31	4
Partida3	106	9
Partida4	76	9
Partida5	104	9
Partida6	145	10
Partida7	64	4
Partida8	13	2
TOTAL:	673	52

Fonte: Autoria própria

Uma vez que foram removidas as transições com duração inferior ao limite, os registros de todas as partidas foram unidos e agrupados por frequência. Assim, para cada frequência foi aplicado o método de identificação de *outlier* quartil. Os registros espúrios foram identificados, removidos e a composição da amostra após essa etapa segue na Tabela 5.

Tabela 5 - Composição da amostra por partida após eliminação de transições agrupadas por frequência com duração considerada *outlier*

Nome	Quantidade total de registros	Número de transições por alteração na Frequência do Inversor
Partida1	134	5
Partida2	31	4
Partida3	106	9
Partida4	76	9
Partida5	104	9
Partida6	123	9
Partida7	64	4
Partida8	13	2
TOTAL:	651	51

Fonte: Autoria própria

Conforme visto na Tabela 5, existe, entre as partidas, variância na quantidade de transições por alteração na Frequência do Inversor. Logo, percebe-se que não há uma distribuição uniforme. Assim, para evitar transições com duração muito acima da duração média, novamente aplicou-se o método de identificação de *outlier* quartil, porém sem realizar o agrupamento por frequência. Os registros espúrios foram identificados, removidos e a composição da amostra após essa etapa segue na Tabelas 6.

Tabela 6 - Composição da amostra por partida após eliminação de transições com duração *outlier*

Nome	Quantidade total de registros	Número de transições por alteração na Frequência do Inversor
Partida1	91	4
Partida2	31	4
Partida3	106	9
Partida4	76	9
Partida5	104	9
Partida6	123	9
Partida7	64	4
Partida8	13	2
TOTAL:	608	50

Fonte: Autoria própria

De forma análoga ao que foi feito para a duração das transições, o método de identificação de *outlier* quartil foi aplicado para os valores de cada variável. Quando detectada na transição, a presença de ao menos um registro espúrio, essa foi completamente removida. Isso foi necessário por se tratar de uma série temporal cuja frequência de amostragem foi unificada. Após remoção dos registros, a composição da amostra é apresentada na Tabela 7.

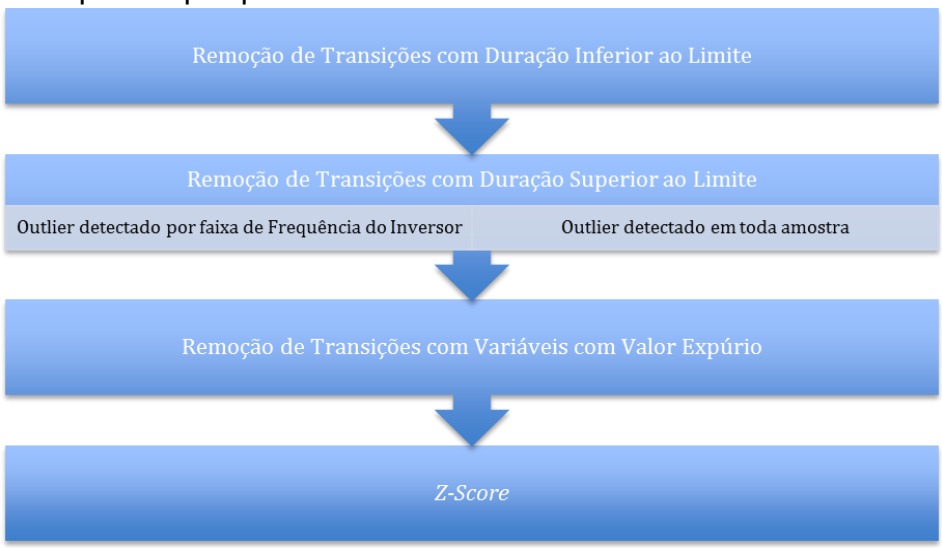
Tabela 7 - Composição da amostra por partida após eliminação de transições com *outliers* em dados coletados do sensor

Nome	Quantidade total de registros	Número de transições por alteração na Frequência do Inversor
Partida1	91	4
Partida2	31	4
Partida3	99	8
Partida4	65	8
Partida5	92	8
Partida6	123	9
Partida7	64	4
Partida8	13	2
TOTAL:	578	47

Fonte: Autoria própria

Por fim, como os valores coletados pelos sensores possuem ordem de grandezas distintas, foi aplicada a normalização *z-score*. As etapas de pré-processamento descritas estão sumarizadas na Figura 20.

Figura 20 - Etapas de pré-processamento



Fonte: Autoria própria

3.5 EXTRAÇAO DE CARACTERÍSTICAS

Com o objetivo de reduzir a quantidade de características para criação do modelo, as seguintes variáveis foram removidas do conjunto de dados: Pressão no Fundo do Poço, Pressão na Sucção da Bomba e Pressão na Descarga da Bomba. O motivo dessa decisão é que o sensor de fundo de poço é propenso a falhas e está indisponível em alguns equipamentos. Portanto, pensando na possível abrangência dessa solução, optou-se por sua exclusão. As demais variáveis estão representadas por meio da variável Diferencial de Pressão na Bomba (Pressão na Descarga da Bomba - Pressão na Sucção da Bomba).

A seguir, duas abordagens para extração de características foram analisadas a fim de identificar a mais adequada para representar o problema.

1. Considerou que cada registro é uma entrada para a construção do modelo e utilizou as variáveis a seguir como características: Frequência do Inversor, Percentual de Abertura da Válvula *Choke*, Diferencial de Pressão na Bomba, Pressão a Montante da Válvula *Choke*, Corrente do Motor e Temperatura do Motor;

2. Cogitou que a abordagem do item 1 não é eficiente, pois os registros possuem dependência entre si, já que se trata de série temporal multivariada. Assim, características da série temporal multivariada foram extraídas e utilizadas como entrada para a construção do modelo.

Para tornar a segunda abordagem possível, o primeiro passo foi gerar, para cada transição, séries temporais multivariadas derivadas. O processo é iterativo e obedece às seguintes especificações:

1. A primeira série derivada possui duração igual ao limite mínimo (6 minutos) ou será igual a transição caso esta tenha duração igual ou inferior ao limite mínimo mais incremento (2 minutos);

2. As séries derivadas são geradas acrescentando, via de regra, dois registros a série imediatamente anterior. A exceção pode ocorrer quando da geração da última série derivada, nas situações em que o acréscimo de dois registros resulta numa série derivada maior do que a original. Nesse caso, o acréscimo será de um registro. O número dois foi arbitrado como incremento padrão por se tratar da terça parte do limite mínimo e por ser representativo para indicar uma mudança na característica da série.

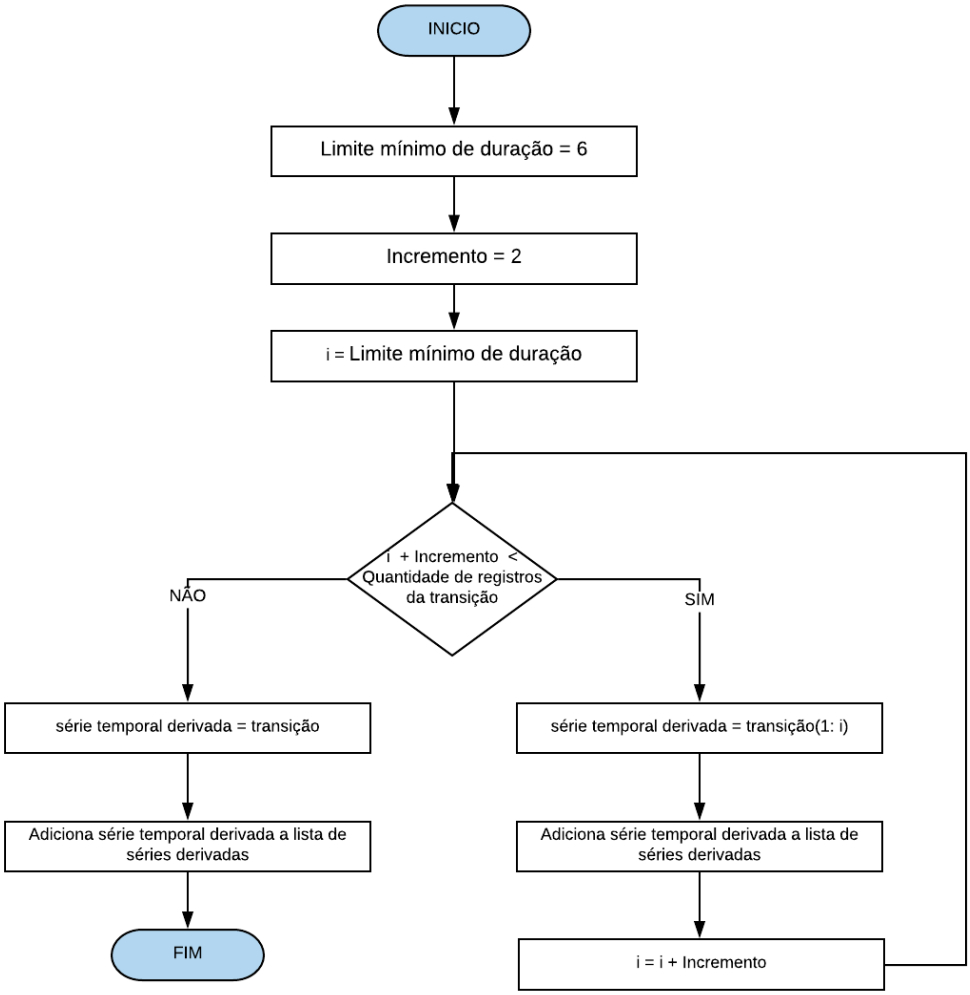
Para cada transição, as especificações acima são aplicadas, conforme demonstrado no fluxo descrito na Figura 21. As 47 transições por alteração na Frequência do Inversor geraram 165 séries temporais multivariadas derivadas. A Tabela 8 apresenta o número de séries temporais derivadas por partida.

Tabela 8 - Número de series temporais derivadas geradas a partir de transições por alteração na Frequência do Inversor

Nome	Número total de séries temporais multivariadas derivadas
Partida1	35
Partida2	5
Partida3	28
Partida4	11
Partida5	26
Partida6	37
Partida7	21
Partida8	2

Fonte: Autoria própria

Figura 21 - Fluxo de geração de séries temporais multivariadas derivadas



Fonte: Autoria própria

Uma vez que as séries temporais multivariadas foram obtidas, as seguintes medidas estatísticas foram calculadas para cada variável de cada série temporal: média, mediana, moda e desvio padrão. Os valores mínimo, máximo e a diferença entre os valores máximo e mínimo também foram usados como características.

Tomando como base:

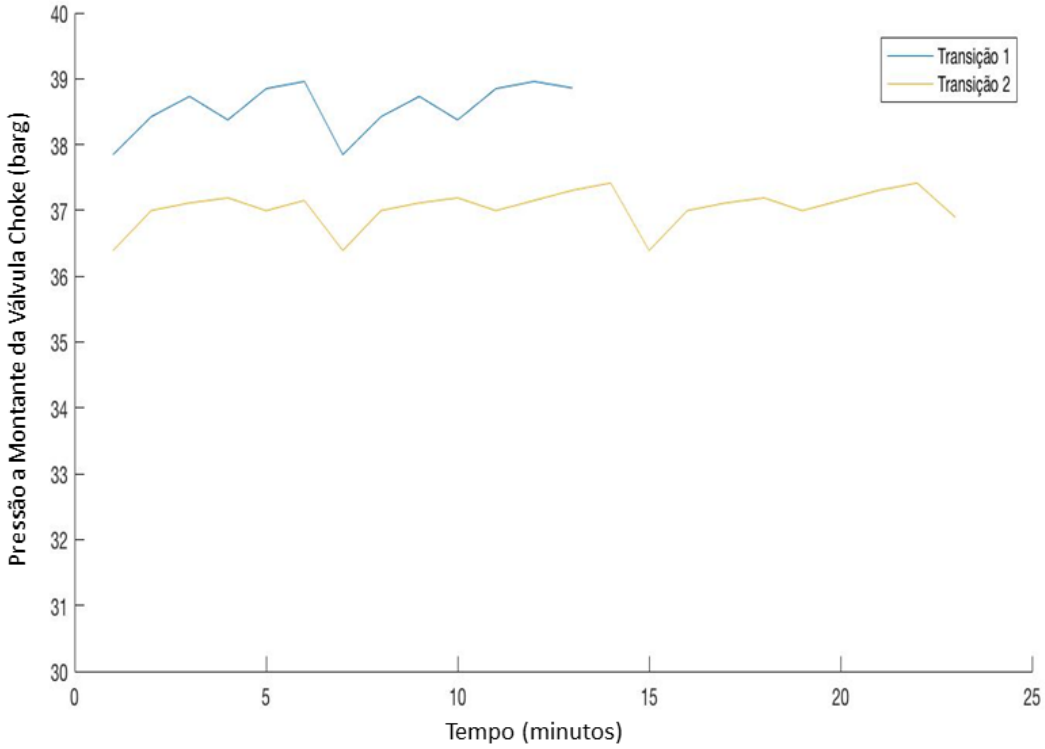
1. Que as transições possuem assinaturas semelhantes;

2. Que as amostras possuem ruído de classificação.

Conjecturou-se que comparar, para cada variável, a distância da série temporal em análise com uma de referência, utilizando o algoritmo *Dynamic Time Warping*

(DTW), permitiria identificar o momento adequado para manipular a variável de interesse Frequência do Inversor. A Figura 22 mostra o comportamento da variável Pressão a Montante da Válvula *Choke,* em duas transições, na mesma Frequência do Inversor, pertencentes a partidas distintas. É possível identificar que embora as transições 1 e 2 possuam durações distintas, tendo a segunda quase o dobro de registros da primeira, a forma das duas é bastante semelhante.

Figura 22 - Comportamento da variável Pressão a Montante da Válvula *Choke* em transições, na mesma frequência do inversor, pertencentes a partidas distintas



Fonte: Autoria própria

Para cada valor de Frequência do Inversor, duas alternativas foram comparadas para escolher a melhor série de referência: a série temporal multivariada com menor duração e a série temporal multivariada selecionada por algoritmo genético. O algoritmo genético utilizado para encontrar a série temporal de referência utilizou os parâmetros listados no Quadro 7.

Quadro 7 - Parâmetros do algoritmo genético utilizado para encontrar a melhor série temporal multivariada de referência

Número de Gerações	100
Tamanho da População	200
Função Objetivo	Maximiza a métrica AUC (*Area Under Curve ROC*) para um modelo de classificação criado com o algoritmo *RUSBoost*.
Indivíduo	Série temporal multivariada
Taxa de Mutação	0,1%
Taxa de *Crossover*	90%
Algoritmo de seleção de indivíduos	Elitismo (preservando 5% dos indivíduos melhores ranqueados e desprezando 5% dos piores indivíduos) combinado com roleta para os indivíduos restantes.

Fonte: Autoria própria

O objetivo de empregar algoritmo genético foi potencializar a construção de um bom modelo de classificação, por meio da escolha de uma série temporal multivariada de referência. Portanto, foi escolhido o algoritmo *RUSBoost* para a construção do modelo utilizado na função objetivo, por este ser, comprovadamente, eficiente em cenários com desbalanceamento de classe. Além disso, a métrica adotada para a maximização da função, a AUC, também é a mais adequada para problemas dessa natureza. Os parâmetros listados no Quadro 8 foram empregados na construção do modelo. Estas são as opções *default* ferramenta utilizada, o MATLAB®.

Quadro 8 - Parâmetros do algoritmo RUSBoost utilizado para encontrar a melhor série temporal multivariada de referência

Número de Ciclos de Aprendizado	30
Taxa de Aprendizado	0,1
Número Máximo de *Splits*:	20

Fonte: Autoria própria

As características extraídas das séries temporais multivariadas estão listadas no Quadro 9.

Quadro 9 - Características extraídas da série temporal multivariada

Frequência do Inversor
Percentual de Abertura da Válvula *Choke*
Diferencial de Pressão na Bomba Mínimo
Diferencial de Pressão na Bomba Máximo
Diferencial de Pressão na Bomba Médio
Desvio Padrão do Diferencial de Pressão na Bomba
Diferença entre o Diferencial de Pressão na Bomba Máximo e Máximo
DTW Diferencial de Pressão na Bomba
Pressão a Montante da Válvula *Choke* Mínima
Pressão a Montante da Válvula *Choke* Máxima
Pressão a Montante da Válvula *Choke* Média
Desvio Padrão da Pressão a Montante da Válvula *Choke*
Diferença entre a Pressão a Montante da Válvula *Choke* Máxima e Mínima
DTW Pressão a Montante da Válvula *Choke*
Corrente do Motor Mínima
Corrente do Motor Máxima
Corrente do Motor Média
Desvio Padrão da Corrente do Motor
Diferença entre Corrente do Motor Máxima e Mínima
DTW Corrente do Motor
Temperatura do Motor Mínima
Temperatura do Motor Máxima
Temperatura do Motor Média
Desvio Padrão da Temperatura do Motor
Diferença entre Temperatura do Motor Máxima e Mínima
DTW Temperatura do Motor

Fonte: Autoria própria

3.6 CONSTRUÇÃO DE MODELOS

Os algoritmos *k*-NN e *RUSBoost* foram escolhidos para comparação nesse estudo de caso por, respectivamente, sua simplicidade e sua robustez em cenários desbalanceados. Uma vez definidos os algoritmos, o passo seguinte foi escolher os hiperparâmetros adequados ao problema.

O algoritmo *k*-NN possui dois hiperparâmetros: o algoritmo de distância e o número de vizinhos. Para auxiliar na escolha dos valores adequados, foi empregada a técnica *Grid Search*. Os hiperparâmetros disponíveis para otimização bem como os valores analisados estão apresentados no Quadro 10. Tendo em vista que as duas

abordagens para extração de características em estudo são bastante distintas, considerou-se pertinente avaliar os hiperparâmetros mais adequados para cada situação.

Quadro 10 - Hiperparâmetro e respectivos valores analisados na técnica Grid Search

Hiperparâmetro	**Valores Analisados**
Algoritmos de distância	Manhattan, Chebychev, Euclidiana, Hamming, Jaccard, Mahalanobis, Minkowski e Spearman ("Pairwise distance between pairs of observations - MATLAB pdist", 2018)
Número de Vizinhos (k)	1 - 20

Fonte: Autoria própria

Com relação ao algoritmo *RUSBoost*, por uma questão de consistência com o modelo gerado na execução do algoritmo genético, optou-se por manter nessa etapa os mesmos valores de hiperparâmetros já especificados no Quadro 8.

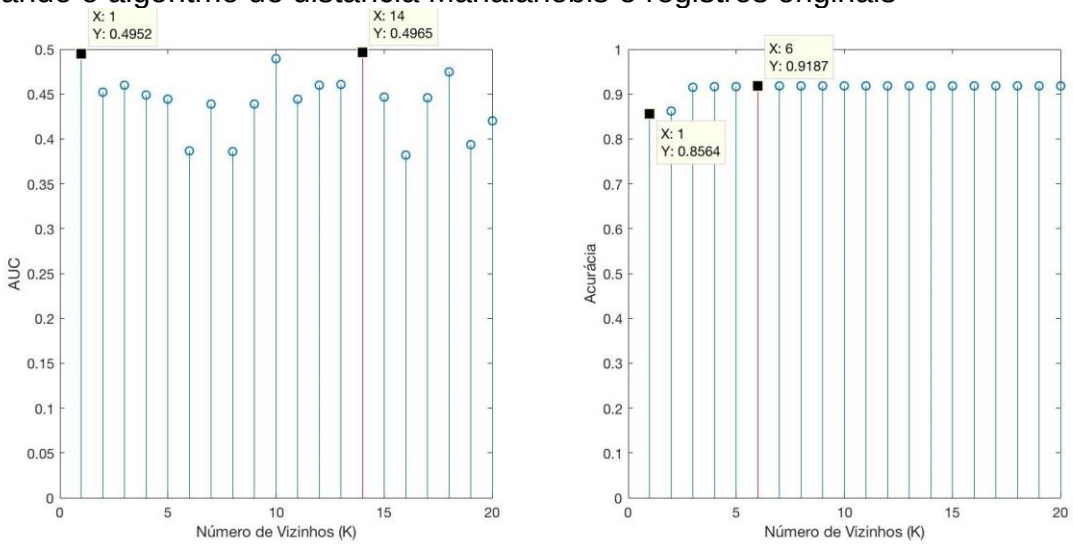
Como a estratégia para a otimização dos hiperparâmetros, o treinamento e o teste do modelo, optou-se, devido à pouca quantidade de dados disponível e a necessidade de evitar sobreajuste (*overfitting*), utilizar a abordagem *cross-validation* com 5 *folds*.

4 RESULTADOS E DISCUSSÕES

A execução do algoritmo *Grid Search* gerou diversos modelos variando os valores dos parâmetros *k* e distância, de acordo com o estabelecido no Quadro 10. Para cada classificador foram coletadas as métricas AUC, recomendada para uso em cenários desbalanceados, e acurácia, frequentemente utilizada no processo de otimização.

Para o cenário em que os modelos foram gerados utilizando como características de entrada os atributos Frequência do Inversor, Percentual de Abertura da Válvula *Choke*, Diferencial de Pressão na Bomba, Pressão a Montante da Válvula *Choke*, Corrente do Motor e Temperatura do Motor, o *Grid Search* selecionou o algoritmo de distância Mahalanobis como o mais adequado para o problema. A Figura 23 apresenta uma comparação das métricas AUC e acurácia para cada valor de *k* no intervalo definido, com este algoritmo de distância.

Figura 23 - Comparação da AUC e acurácia para os diferentes valores de *k*, utilizando o algoritmo de distância Mahalanobis e registros originais



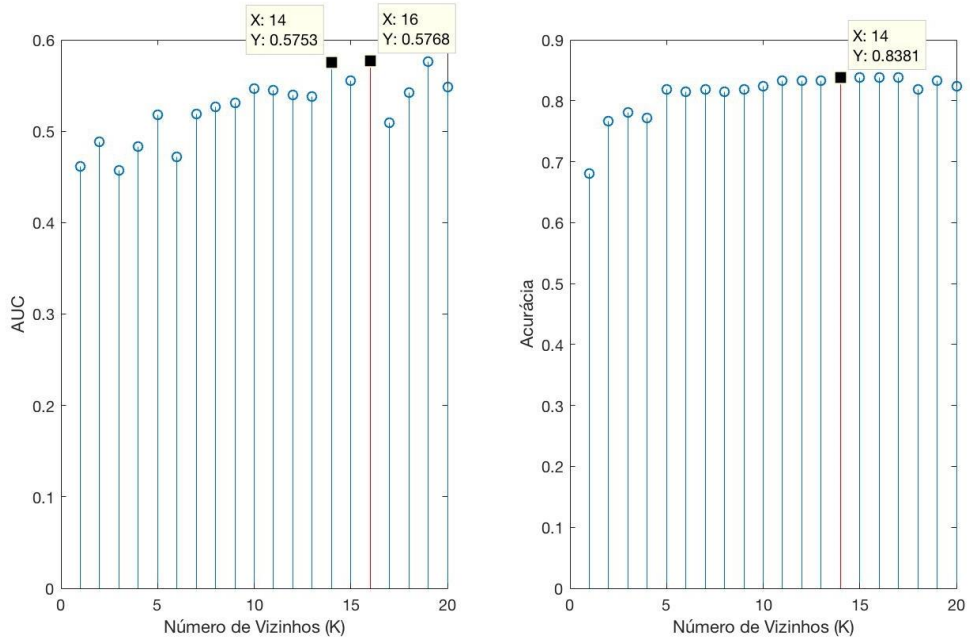
Fonte: Autoria própria

Conforme pode ser observado na Figura 23, o parâmetro *k* = 3 satisfaz algo grau de acurácia. No entanto, este apresenta valor de AUC inferior ao *k* = 1. Por se tratar de

um problema desbalanceado, a AUC é a métrica mais adequada. Portanto, é importante destacar que a escolha da métrica a ser maximizada pelo uso do hiperparâmetro é um elemento crucial.

Para o cenário em que os modelos foram gerados utilizando como atributos de entrada as características extraídas das séries temporais multivariadas (Quadro 9), o algoritmo de distância escolhido pelo *Grid Search* foi Jaccard. A Figura 24 apresenta uma comparação das métricas AUC e acurácia para cada valor de *k* no intervalo definido, utilizando este algoritmo de distância.

Figura 24 - Comparação da AUC e acurácia para os diferentes valores de *k*, utilizando o algoritmo de distância Jaccard e características extraídas das séries temporais multivariadas



Fonte: Autoria própria

Analisando os gráficos da Figura 24, também se percebe a diferença de indicativo do modelo com o melhor desempenho pelas métricas AUC e acurácia. Uma outra característica interessante é a existência de dois modelos com AUC próximas. Nestes casos, *trade offs* devem ser realizados e o modelo mais simples deve ter preferência. O Quadro 11 mostra os hiperparâmetros selecionados para o *k*-NN.

Quadro 11 - Hiperparâmetros selecionados para o *k*-NN

Tipo de Característica de Entrada do Modelo	Hiperparâmetros Selecionados
Registros originais	Número de Vizinhança: 1 Distância Utilizada: Mahalanobis
Extraídas das séries temporais multivariadas	Número de Vizinhança: 14 Distância Utilizada: Jaccard

Fonte: Autoria própria

A Tabela 9 apresenta o resultado dos modelos criados utilizando o algoritmo *k*-NN com hiperparâmetros otimizados por *Grid Search* e o *RUSBoost*. Conforme citado, para o *k*-NN foram gerados dois modelos distintos: o primeiro considerou como entrada os registros originais das variáveis e o segundo utilizou características extraídas das séries temporais multivariadas usando como referência a transição de menor duração por Frequência do Inversor. Para o *RUSBoost*, além de modelos análogos aos dois citados em termos de características de entrada, foi criado um modelo adicional com características extraídas das séries temporais com a transição de referência escolhida por algoritmo genético.

Tabela 9 - Comparação dos classificadores *k*-NN e *RUSBoost*

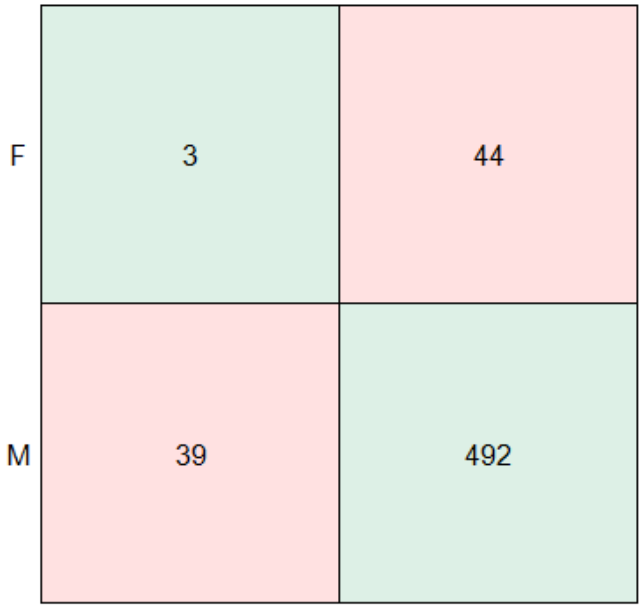
Métrica	***k*-NN**		***RUSBoost***		
	Registros originais	**Série temporal menor transição**	**Registros originais**	**Série temporal menor transição**	**Série temporal transição por AG**
Acurácia	0,8564	0,8381	0,4792	0,8061	0,8667
AUC	0,4952	0,5753	0,5117	0,7933	0,9175

Fonte: Autoria própria

Os Quadros 12 e 13 apresentam as matrizes de confusão para os dois modelos gerados com *k*-NN e evidenciam o baixo índice de acerto da classe F. Quando do uso dos registros originais, apenas 3 dos 47 registros foram preditos corretamente.

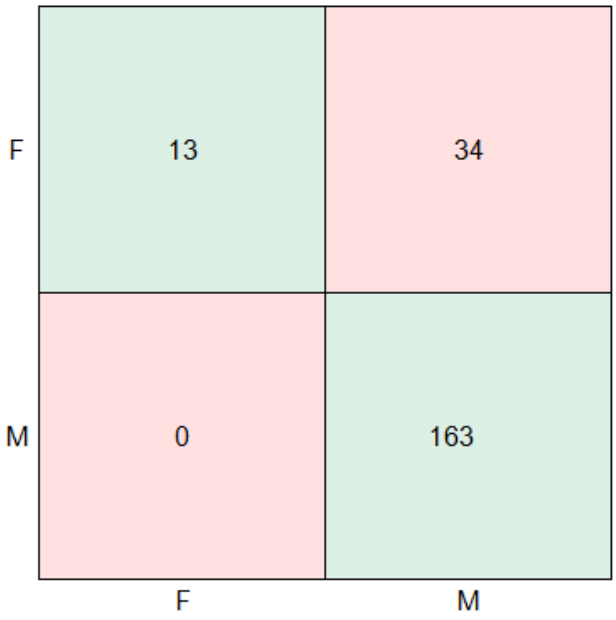
Esse número melhora quando do uso de séries temporais, subindo para 13 ocorrências, mas ainda assim é bastante baixo.

Quadro 12 - Matriz de confusão do modelo gerado com *k*-NN e uso dos registros como entrada para o algoritmo



Fonte: Autoria própria

Quadro 13 - Matriz de confusão do modelo gerado com *k*-NN e abordagem série temporal. Foi escolhida a menor transição de referência para cálculo do DTW

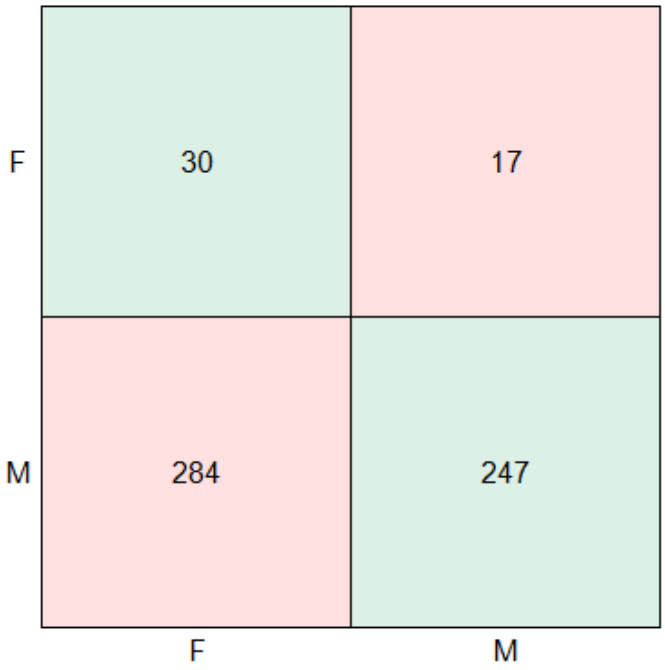


Fonte: Autoria própria

Conforme pode ser observado na Tabela 9 e nas matrizes de confusão apresentadas nos Quadros 12 e 13, os modelos gerados com *k*-NN, independente das características utilizadas, apresentaram alto grau de acurácia. No entanto, falham ao prever a ocorrência da classe minoritária F. Para o problema em estudo, este resultado demonstra que os classificadores gerados indicam, na maior parte do tempo, a manutenção do patamar operacional, ou seja, o seu emprego resultaria em uma partida com pouca alteração no *set point* da Frequência do Inversor (classe F), o que é o oposto do desejado.

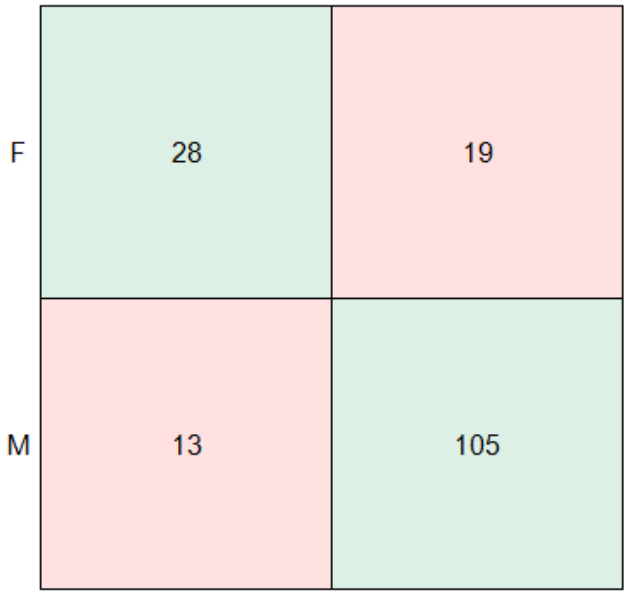
Os Quadros 14, 15 e 16 apresentam as matrizes de confusão dos modelos gerados com *RUSBoost*. Comparando-as, percebe-se o melhor desempenho do modelo que utilizou algoritmo genético para detecção da transição de referência, o pode ser evidenciado não apenas pelo maior número de acertos na predição da classe F (31 ocorrências contra 28) como também pelo número inferior de falsos positivos e falsos negativos.

Quadro 14 - Matriz de confusão do modelo gerado com *RUSBoost* e uso dos registros como entrada para o algoritmo



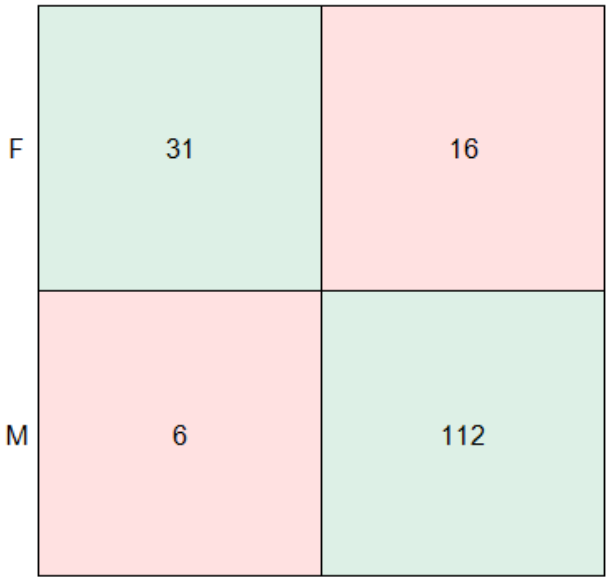
Fonte: Autoria própria

Quadro 15 - Matriz de confusão do modelo gerado com *RUSBoost* e abordagem série temporal. Menor transição como referência para cálculo do DTW



Fonte: Autoria própria

Quadro 16 - Matriz de confusão do modelo gerado com *RUSBoost* e abordagem série temporal. A transição de referência para cálculo do DTW foi escolhida por Algoritmo Genético



Fonte: Autoria própria

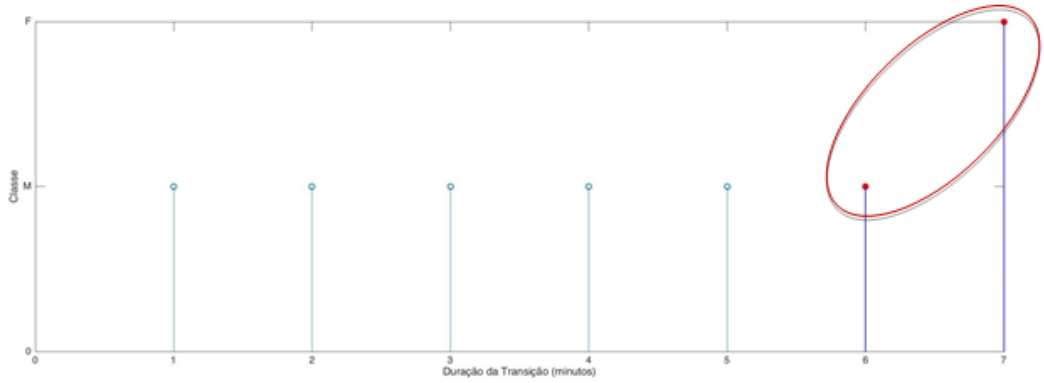
O modelo gerado com *RUSBoost* e registros originais como característica de entrada apresentou baixa acurácia, sendo considerado aleatório por essa métrica. Considerando o AUC, este foi um pouco melhor que o *k*-NN correspondente. Em

termos numéricos, previu corretamente 30 ocorrências da classe F (contra 3 do *k*-NN). Porém, o número de falsos positivos foi muito alto, 284 contra 39 do *k*-NN, demonstrando ser um modelo pouco robusto. Em termos práticos, o classificador gerado indica a mudança do patamar operacional com muita frequência, o que não é desejado, pois ocasionaria um descontrole do processo e exposição do equipamento à falha.

Quanto aos modelos *RUSBoost* com características extraídas das séries temporais multivariadas, ambos apresentaram valor de acurácia alto e AUC correspondentes a bons modelos. O que utilizou o algoritmo genético como mecanismo de escolha da transição de referência apresentou o melhor resultado, com acurácia de 0.8667 e AUC de 0,9175, valor bem acima de um classificador aleatório perfeito, que apresenta AUC = 0,5 e próximo ao modelo perfeito que, conforme Hand (2009), apresenta AUC = 1. O equilíbrio entre essas duas métricas de avaliação demonstra a robustez do modelo.

A existência de falsos positivos e negativos é esperada, pois conforme demonstrado nesse estudo, os dados apresentam ruído de classificação. Nesse contexto, em uma transição ineficiente, o modelo classifica em dado momento um registro como F (quando o valor real é M), e posteriormente, classifica um registro como M (quando o valor esperado é F). Na Figura 25, os instantes t = 6 e t = 7 apresentam erros de predição que ilustram essa avaliação. Tal comportamento não revela um problema do classificador, e sim reforça a hipótese da presença de ruído de classificação e que as partidas já realizadas poderiam ter tido menor duração.

Figura 25 - Avaliação do resultado da predição de uma transição ineficiente



Fonte: Autoria própria

De acordo com os resultados apresentados na Tabela 9 e nas matrizes de confusão apresentadas nos Quadros 15 e 16, os classificadores *RUSBoost* com características extraídas das séries temporais são capazes de indicar adequadamente tanto a manutenção do patamar operacional quanto uma mudança no *set point* da Frequência do Inversor, sendo adequados ao problema estudado. Além disso, situações análogas a pontuada na Figura 25, em que uma mudança no patamar operacional é antecipada de forma coerente, indicam que a utilização de um classificador desse tipo tem potencial para reduzir a duração das partidas da BCSS, ocasionando uma maior eficiência operacional.

5 CONCLUSÃO

Os resultados obtidos demonstram que é factível empregar um sistema inteligente para apoiar a tomada de decisão na fase Rampa de Aceleração do processo de partida a quente. Tal uso tem potencial para tornar o processo mais previsível e curto, aumentando a eficiência operacional e reduzindo o tempo de exposição do equipamento às condições adversas da partida.

Conforme hipotetizado, a dinâmica do sistema pede que os dados sejam tratados como séries temporais multivariadas e não registros individuais. A comparação entre os diversos modelos gerados apontou um desempenho muito baixo quando do uso dessa última abordagem. A escolha pelo uso do DTW e transições de referência se mostrou acertada.

Com relação ao ruído de classificação, o trabalho demonstrou que ele de fato está presente na base histórica e que este é um fator de risco na construção de um bom modelo de classificação. Embora o resultado apresentado pelo modelo *RUSBoost* com série temporal e transição escolhida por AG tenha sido ótimo, o número de erros, assinalados pelos indicadores falso positivo e falso negativo na matriz de confusão, podem ser reflexo de não ter sido aplicada nesse estudo uma técnica específica para remoção de ruído na fase de pré-processamento.

Dada a diferença de desempenho do *k*-NN e *RUSBoost*, ficou clara a importância de empregar um algoritmo de classificação que trate de forma específica a questão do desbalanceamento de classes. É importante frisar que o *RUSBoost* atua realizando um pré-processamento, balanceamento as amostras por meio de subamostragem da classe majoritária de forma aleatória. No entanto, existem outras abordagens disponíveis que não foram exploradas nesse trabalho.

Outro ponto de destaque é a comprovação de que a acurácia não é uma boa métrica para ser utilizada em cenários com desbalanceamento de classes e ruído de classificação. O uso do AUC, em conjunto com a análise da matriz de confusão, se mostrou uma estratégia mais robusta.

Por fim, cabem as seguintes indicações par trabalho futuro:

- Avaliar técnicas específicas para remoção de ruído de classe;
- Comparar abordagens para lidar com o problema de desbalanceamento de classes no que se refere a pré-processamento e ao uso de algoritmo com função de custo modificada;
- Analisar a pertinência de empregar técnicas de seleção de características para redução da dimensionalidade;
- Testar a solução proposta em outros conjuntos de bombeio submerso submarino;
- Prospectar estratégias para incorporar ao escopo da solução transições por mudança no Percentual de Abertura da Válvula *Choke*.

REFERÊNCIAS

ALMEIDA, Gustavo; SALLES, Jose. **Controle preditivo:** sintonia e aplicações na siderurgia. Curitiba: Appris, 2016.

BATHER, John. **Decision theory**. Tradução: Chichester: [S,l.]: Wiley, 2000.

BATISTA, Evellyne. **Desenvolvimento de uma ferramenta computacional para aplicação no método de elevação por bombeio centrífugo submerso.** 2009. Dissertação (Mestrado em Ciências) - Programa de Pós-Graduação em Engenharia Elétrica e de Computação, Universidade Federal do Rio Grande do Norte, Natal, 2009 Disponível em*:* <https://repositorio.ufrn.br/jspui/bitstream/123456789/15297/1/EvellyneSBpdf.pdf>. Acesso em: 05 jan. 2018.

BERGAMINI, C. et al. Combining different biometric traits with one-class classification. **Signal Processing**, v. 89, n. 11, p. 2117-2127, 2009.

BERGER, James. **Statistical decision theory and bayesian analysis**. New York: Springer New York, 1985.

BERKHIN, Pavel. In: KOGAN, J; NICHOLAS, C; TEBOULLE, M. **Grouping multidimensional data.** Berlin: Springer, 2002.

BETÔNICO, Gustavo. **Estudo da distribuição da temperatura em motores de bombas centrífugas submersas.** 2013. Dissertação (Mestrado em Ciências e Engenharia de Petróleo) - Programa de Pós-Graduação em Ciências de Engenharia de Petróleo, Universidade de Campinas, Campinas, 2013. Disponível em: <http://repositorio.unicamp.br/bitstream/REPOSIP/265663/1/Betonico_GustavodeCarvalho_M.pdf >. Acesso em: 05 jan. 2018.

BOUVEYRON, Charles; GIRARD, Stéphane. Robust supervised classification with mixture models: learning from data with uncertain labels. **Pattern Recognition**, v. 42, n. 11, p. 2649-2658, 2009.

BRADLEY, Andrew. The use of the area under the ROC curve in the evaluation of machine learning algorithms. **Pattern Recognition**, v. 30, n. 7, p. 1145-1159, 1997.

CASTRO, Cristiano; CARVALHO, Mateus; BRAGA, Antônio. An improved algorithm for SVMs classification of imbalanced data sets. **Engineering Applications of Neural Networks**, v. 43 of Communications in Computer and Information Science, p. 108-118, 2009.

CASTRO, Cristiano; BRAGA, Antônio. Aprendizado supervisionado com conjuntos de dados desbalanceados. **SBA: Controle & Automação Sociedade Brasileira de Automática**, v. 22, n. 5, p. 441-466, 2011.

COELHO, Guilherme. **Geração, seleção e combinação de componentes para ensembles de redes neurais aplicadas a problemas de classificação.** 2006. Dissertação (Mestrado em Engenharia Elétrica). Programa de Pós-graduação em

Engenharia Elétrica, Universidade de Campinas, Campinas, 2006. Disponível em: <ftp://ftp.dca.fee.unicamp.br/pub/docs/vonzuben/theses/tese_Coelho.pdf>. Acesso em: 05 jan. 2018.

CORTES, Corinna; MOHRI, Mehryar. **AUC optimization vs. error rate minimization.** Advances in Neural Information Processing Systems 16, MIT Press, Cambridge, MA. 2004.

DUDA, Richard; HART, Peter; STORK, David. **Pattern classification.** [S.l.]: Wiley-Interscience, 2001.

FACELI, Katti et al. **Inteligência artificial:** uma abordagem de aprendizado de máquina**.** Rio de Janeiro: LTC, 2011.

FAN, Wei et al. AdaCost: misclassification cost-sensitive boosting. In: IEEE INTERNATIONAL CONFERENCE ON MACHINE LEARNING. 1999. **Proceedings**... [S.l.: s.n.], 1999, p. 97-105.

FAWCETT, T. **Roc graphs:** notes and practical considerations for researchers. Technical report, HP Labora- tories, Palo Alto, USA, 2004.

FAWCETT, T. An introduction to roc analysis. **Pattern Recognition Letters**, v. 27, n. 8, p. 861-874, 2005.

FAYYAD, Usama; PIATETSKY-SHAPIRO, Gregory; SMYTH, Padhraic. **From data mining to knowledge discovery in databases.** American Association for Artificial Intelligence, 1996.

FRÉNAY, Benoît; KABÁN, Ata. **A comprehensive introduction to label noise.** In: EUROPEAN SYMPOSIUM ON ARTIFICIAL NEURAL NETWORKS. Computational Intelligence and Machine Learning. 2014. **Proceedings**... [S.l.: s.n.], 2014.

FREUND, Y; SCHAPIRE, R. Experiments with a new boosting algorithm. In: INTERNATIONAL CONFERENCE, 13., 1996. **Proceedings**... [S.l.]: Machine Learning, 1996.

GAMA, João. **Árvore de decisão.** Disponível em: <http://www.dcc.fc.up.pt/~ines/aulas/MIM/arvores_de_decisao.pdf>. Acesso em: 6 nov. 2017.

HAND, David. Measuring classifier performance: a coherent alternative to the area under the ROC curve. **Machine Learning**, v. 77, n. 1, p. 103-123, 2009.

HANLEY, J. A.; MCNEIL, B. J. The meaning and use of the area under a receiver operating characteristic (roc) curve. **Radiology**, v. 143, n. 1, p. 29-36, 1982.

HE, Haibo; GARCIA, Edwardo. Learning from imbalanced data. **IEEE Transactions on Knowledge and Data Engineering**, v. 21, n. 9, p. 1263-1284, 2009.

_____; SHEN, Xiaoping. A ranked subspace learning method for gene expression data classification. In: INTERNATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, 2007. **Proceedings**... [S.l.: s.n.], 2007,v. 1, p. 358-364.

HYDE, R.; BRINNER, T. Starting characteristics of electric submergible oil well pumps. **IEEE Transactions on Industry Applications**, v. IA-22, n. 1, p. 133-144, 1986.

JAPKOWICZ, Nathalie. Supervised versus unsupervised binary-learning by feedforward neural networks. **Machine Learning**, v. 42, n. 1-2, p. 97-122, 2001.

_____; SHAJU, Stephen. The class imbalance problem: a systematic study. **Intelligent Data Analysis**, v. 6, n. 5, p. 429-449, 2002.

KASHIF, S.; SAQIB, M. Soft starting of induction motors using neuro fuzzy and soft computing. 2008 In: INTERNATIONAL CONFERENCE ON ELECTRICAL ENGINEERING. 2., 2008. [**Proceedings...**]. [S.l. : s.n.], 2008.

KAUFMAN L.; ROUSSEEUW, P. J. **Finding groups in data:** an introduction to cluster analysis. Hoboken, NJ: John Wiley & Sons, 1990.

KUBAT, Miroslav; HOLTE, Robert; MATWIN, Stan. Machine learning for the detection of oil spills in satellite radar images. **Machine Learning,** v. 30, n. 2-3, p. 95-215, 1998.

MACHADO, Emerson; LADEIRA, Marcelo. Um estudo de limpeza em base de dados desbalanceada com sobreposição de classes. In: ENCONTRO NACIONAL DE INTELIGÊNCIA ARTIFICIAL. 6., 2007. Congresso da Sociedade Brasileira de Computação. 27., 2007. **Anais...** [S.l.]: SBC, 2007, p. 330-340.

MALOOF, Marcus. Learning when data sets are imbalanced and when costs are unequal and unknown. In: INTERNATIONAL CONF. MACHINE LEARNING. Workshop on Learning from Imbalanced Data Sets II. 2003. **Proceedings**... [S.l. : s.n.], 2003.

MANEVITZ, Larry; YOUSEF, Malik. One-class svms for document classification. **Journal of Machine Learning Research**, v. 2, p. 139-154, 2001.

MANEVITZ, Larry; YOUSEF, Malik. One-class document classification via Neural Networks. **Neurocomputing**, v. 70, n. 7-9, p. 1466-1481, 2007.

MARKOU, Markos; SINGH, Sameer. Novelty detection: a review - part 2. **Signal Processing**, v. 83, n. 12, p. 2499-2521, 2003.

MCLACHLAN, Geoffrey; KRISHNAN, Thriyambakam. **The EM algorithm and extensions.** [S.l.]: John Wiley & Sons, 1997.

MONARD, M. C.; BARANAUSKAS, J. A. Conceitos de aprendizado de máquina. In: REZENDE, S. O. **Sistemas inteligentes: fundamentos e aplicações**. Barueri: Manole, 2003. Cap.4, p.89-114.

MONARD, Maria; BATISTA, Gustavo. **Learning with skewed class distribution.** Advances in Logic, Artificial Intelligence and Robotics, p. 173-180, 2003.

NEELY, A.; PATTERSON, M. Soft start of submersible pumped oil wells. **Journal of Petroleum Technology**, v. 36, n. 4, p. 653-656, 1984.

PAIRWISE distance between pairs of observations - MATLAB pdist. Disponível em: <https://www.mathworks.com/help/stats/pdist.html>. Acesso em: 20 jan. 2018.

PEARSON, Ronald; GONYE, Gregory; SCHWABER, James. Imbalanced clustering for microarray time-series. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING. 20., 2003. **Proceedings**... [S.l. : s.n.], 2003.

PRATI, Ronaldo; BATISTA, Gustavo; MONARD, Maria. Class imbalances versus class overlapping: an analysis of a learning system behavior. **MICAI 2004: Advances in Artificial Intelligence**, v. 2972, p. 312-321, 2004.

_____; _____; _____. Evaluating classifiers using roc curves. **Latin America Transactions IEEE (Revista IEEE America Latina)**, v. 6 n. 2, p. 215-222, 2008

PROVOST, Foster; FAWCETT, Tom. Analysis and visualization of classifier performance: comparison under imprecise class and cost distributions. In: INTERNATIONAL CONFERENCE ON KNOWLEDGE DISCOVERY AND DATA MINING. 3., 1997. **Proceedings**... [S.l. : s.n.], 1997, p. 43-48.

_____; _____. Robust classification for imprecise environments. AAAI ’98/IAAI ’98: In: CONFERENCE ON ARTIFICIAL INTELLIGENCE. 15., 1998. Innovative applications of artificial intelligence. **Proceedings**... [S.l. : s.n.], 1998, p. 706-713.

RASKUTTI, Bhavani; KOWALCZYK, Adam. Extreme rebalancing for SVMs. ACM **SIGKDD Explorations Newsletter**, v. 6, n. 1, p. 60-69, 2004.

ROSSI, A. **Ajuste de parâmetros de técnicas de classificação por algoritmos bioinspirados.** 2009. Dissertação (Mestrado em Ciências de Computação e Matemática Computacional) - Programa de Pós-graduação do Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, São Carlos, 2009. Disponível em: <https://repositorio.ufrn.br/jspui/bitstream/123456789/.../Phelipe SenaOliveira_TESE.pdf>. Acesso em: 05 jan. 2018.

SAMUEL, Arthur. Some studies in machine learning using the game of checkers. **IBM Journal of Research and Development**, v. 44, n. 1-2, 1959.

SANTOS, Fábio José Justo dos. **Análise de séries temporais fuzzy para previsão e identificação de padrões comportamentais dinâmicos.** 2015. 132 f. Tese (Doutorado em Ciência da Computação) - Universidade de Federal de São Carlos, São Carlos, 2015.

SCHÖLKOPF, Bernhard et al. Estimating the support of a high-dimensional distribution. **Neural Computation**, v. 13, n. 7, p. 1443-1471, 2001.

SEIFFERT, Chris et al. RUSBoost: a hybrid approach to alleviating class imbalance. **IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans**, v. 40, n. 1, p. 185-197, 2010.

SPACKMAN, K. A. Signal detection theory: valuable tools for evaluating inductive learning. In: INTERNATIONAL WORKSHOP ON MACHINE LEARNING. 6., 1989. **Proceedings...** [S.l.: s.n.], 1989, p. 160-163.

SUN, Yanmin et al. Cost-sensitive boosting for classification of imbalanced data. **Pattern Recognition**, v. 40, n. 12, p. 3358-3378, 2007.

TAKÁCS, Gábor. **Electrical submersible pumps manual.** Amsterdam: Gulf Professional, 2009.

THOMAS, José. **Fundamentos de engenharia de petróleo.** 2. ed. Rio de Janeiro: Interciência, 2001.

TING, Kai. A comparative study of cost-sensitive boosting algorithms. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING. 17. 2000. **Proceedings...** [S.l.: s.n.], 2000, p. 983-990.

TOMEK, Ivan. Two modifications of CNN. **IEEE Transactions on Systems, Man, and Cybernetics**, v. SMC-6, n. 11, p. 769-772, 1976.

VERGARA, Luis et al. **Jubarte:** an approach to the operation of deepwater ESPs | Schlumberger. Disponível em: <http://www.slb.com/resources/technical_papers/artificial_lift/esp-workshop-2015-jubarte-deepwater-esp.aspx>. Acesso em: 6 nov. 2017.

WEISS, Gary. **Mining with rare cases.** Data Mining and Knowledge Discovery Handbook, [S.l.: s.n.], 2005. p. 65-776.

WEISS, Gary. Mining with rarity. **ACM SIGKDD Explorations Newsletter**, v. 6, n. 1, p. 7-19, 2004.

WILSON, Dennis. Asymptotic properties of nearest neighbor rules using edited data. **IEEE Transactions on Systems, Man, and Cybernetics,** v. SMC-2, n. 3, p. 408-421, 1972.