

INSTITUTO FEDERAL DE EDUCAÇÃO CIÊNCIA E TECNOLOGIA
DE MINAS GERAIS – *CAMPUS* BETIM
BACHARELADO EM ENGENHARIA MECÂNICA

**PREVISÃO DE FALHAS EM ROLAMENTOS UTILIZANDO APRENDIZADO DE
MÁQUINA**

Betim

2021

JOÃO VITOR DE LIMA SANTOS

**PREVISÃO DE FALHAS EM ROLAMENTOS UTILIZANDO APRENDIZADO DE
MÁQUINA**

Trabalho de conclusão de curso apresentado ao Curso Bacharelado em Engenharia Mecânica do Instituto Federal de Minas Gerais – *Campus* Betim para obtenção do grau de bacharel em Engenharia Mecânica.

Orientador: Prof. Dr. Leandro Freitas de Abreu

Betim

2021

S237p

Santos, João Vitor de Lima.

Previsão de falhas em rolamentos utilizando aprendizado de máquina. [manuscrito] / João Vitor de Lima Santos – Betim: Instituto Federal de Minas Gerais, 2021.

87 f.; il., color.

Orientador: Prof. Leandro Freitas de Abreu.

Monografia (Graduação) – Instituto Federal de Minas Gerais – *Campus* Betim, Bacharelado em Engenharia Mecânica.

1. Rolamentos. 2. Vibração – Análise. 3. Aprendizado do computador.
I. Almeida, Leandro Freias de. II. Instituto Federal de Minas Gerais – *Campus* Betim. III. Título

CDU 658.58

Catálogo: Luciana Batista Neves - CRB-6/2000

João Vitor de Lima Santos

PREVISÃO DE FALHAS EM ROLAMENTOS UTILIZANDO APRENDIZADO DE MÁQUINA

Trabalho de Conclusão de Curso
apresentado ao curso de Engenharia
Mecânica do Instituto Federal de Minas
Gerais Campus Betim como requisito
parcial à obtenção do grau de Bacharel em
Engenharia Mecânica.

Betim, 07 de julho de 2021.

BANCA EXAMINADORA



Prof. Dr. Leandro Freitas
DAUTI – IFMG Campus Betim (orientador)



Prof. Me. Virgil Del Duca Almeida
DAUTI – IFMG Campus Betim



Eng. Saulo Gomes Galvão
Líder Aviação

AGRADECIMENTOS

Primeiramente agradeço a Deus por ter me dado a capacidade e a oportunidade de acreditar em mim. Agradeço à minha família pelo suporte e colaboração que permitiram a trajetória da minha jornada. Ao meu orientador agradeço pelas conversas inspiradoras que possibilitaram a conclusão deste trabalho e a todas as pessoas que de alguma forma estiveram envolvidas na concretização do mesmo.

“Você não é só uma gota no oceano, você é o próprio oceano dentro de uma gota. A inspiração que você procura já está dentro de você. Fique em silêncio e escute. A tarefa não é buscar o amor, mas apenas procurar e desfazer todas as barreiras dentro de si mesmo que você construiu contra ele.”

Jalaladim Rumi

RESUMO

O presente trabalho tem como objetivo utilizar o aprendizado de máquina, em uma aplicação mais voltada para a área da mecânica, prever falhas/defeitos em rolamentos mecânicos, que são bastante utilizados em todo tipo de sistema ou maquinário na indústria. Alguns caminhos e etapas foram extremamente necessários para se alcançar os resultados esperados, após a definição do estudo de caso, informações provenientes de um experimento real, foi encontrado no repositório da *NASA*, sendo preciso organizar e preparar os mesmos. Para o processamento dos dados a Transformada de *Wavelet* foi a escolhida, em termos de descrever melhor os eventos, considerando também o tempo na análise das frequências do sinal, o que se tornou imprescindível calcular alguns parâmetros estatísticos dos coeficientes que a *Wavelet* retornou. Sendo assim dois algoritmos de aprendizado de máquina foram selecionados, Máquina de Vetores de Suporte e a Rede Neural Artificial, para entender os parâmetros estatísticos dos coeficientes e realizar as previsões dos estados condicionais dos rolamentos. Por fim o desempenho da SVM foi muito satisfatório em comparação com a RNA.

Palavras-chave: Aprendizado de Máquina. Rolamentos Mecânicos. Previsão de Falhas. Transformada de Wavelet. Análise de Vibração.

ABSTRACT

The present work aims to use machine learning, in an application more focused on mechanics, to predict failures/defects in mechanical bearings, which are widely used in all types of systems or machinery in the industry. Some paths and steps were extremely necessary to achieve the expected results, after defining the case study, information from a real experiment was found in the NASA repository, being necessary to organize and prepare them. For data processing, the Wavelet Transform was chosen, in terms of better describing the events, also considering the time in the analysis of the signal frequencies, which made it essential to calculate some statistical parameters of the coefficients that the Wavelet returned. Thus, two machine learning algorithms were selected, Support Vector Machine and Artificial Neural Network, to understand the statistical parameters of the coefficients and to make the predictions of the conditional states of the bearings. Finally, the performance of SVM was very satisfactory compared to ANN.

Keywords: Machine Learning. Mechanical Bearings. Failure Prediction. Wavelet Transform. Vibration Analysis.

LISTA DE ILUSTRAÇÕES

Figura 1 – Percentual de falhas em componentes de elementos rotativos	18
Figura 2 – Metodologia do experimento	20
Figura 3 - Constituição de um rolamento de esfera.....	22
Figura 4 – Rolamento de Esfera	24
Figura 5 – Rolamento Reto ou Cilíndrico	25
Figura 6 – Rolamento de Rolo Cônico	25
Figura 7 – Rolamento de Rolo Esférico	26
Figura 8 – Rolamento de Rolo de Agulha	26
Figura 9 – Rolamento Axial de Esfera	27
Figura 10 – Sistema Massa-Mola	27
Figura 11 – Acelerômetro de Piezoelétrico	30
Figura 12 – Esquemático de um Acelerômetro de Piezoelétrico.....	30
Figura 13 – Transformada de <i>Fourier</i>	33
Figura 14 – Transformada de Janelada de <i>Fourier</i>	36
Figura 15 – Janela Típica da SFTF.....	36
Figura 16 – Senoide e Sua Transformada <i>Wavelet db10</i>	38
Figura 17 – Sinal Senoidal com Fator de Escala	38
Figura 18 – Sinal Com Fatores de Escala Distintos	39
Figura 19 – Transformada <i>Wavelet</i> Considerando Todo o Sinal	40
Figura 20 – Transformada <i>Wavelet</i> Discreta	40
Figura 21 – Transformada <i>Wavelet Haar</i>	41
Figura 22 – Transformada <i>Wavelet Daubechies</i>	42
Figura 23 – Transformada <i>Wavelet Biorthogonal</i>	42
Figura 24 – Transformada <i>Wavelet Coiflets</i>	43
Figura 25 – Transformada <i>Wavelet Symlets</i>	44
Figura 26 – Transformada <i>Wavelet Morlet</i>	44

Figura 27 – Transformada <i>Wavelet</i> Chapéu Mexicano	45
Figura 28 – Transformada <i>Wavelet</i> Meyer	45
Figura 29 – Hiperplanos de Separação de uma SVM.....	50
Figura 30 – Distâncias dos Hiperplanos	50
Figura 31 – Estrutura de Dados Não linear	51
Figura 32 – Transformação Do Espaço	52
Figura 33 – Principais Funções de Kernel.....	53
Figura 34 – Regressão por Vetores de Suporte	54
Figura 35 – Comparação entre dois parâmetros C	55
Figura 36 – Comparação entre a variação dos parâmetros γ	55
Figura 37 – Funcionamento de uma Rede Neural Biológica.....	57
Figura 38 – Rede Neural Biológica	57
Figura 39– Funcionamento de uma Rede Neural Artificial	58
Figura 40 – Arquitetura de uma Rede Neural Artificial.....	58
Figura 41 – Função Linear	59
Figura 42 – Função Degrau	59
Figura 43 – Função Degrau Bipolar	60
Figura 44 – Função Rampa.....	60
Figura 45 – Função Sigmóide.....	61
Figura 46 – Função Tangente Hiperbólica	61
Figura 47 – Função ReLu	62
Figura 48 – Esquemático da Metodologia.....	63
Figura 49 – Desempenho de Diversas Famílias <i>Wavelet</i> (Cinza Claro – Rolamentos Normais e Cinza Escuro – Rolamentos Defeituosos).....	65
Figura 50 – Série Temporal do Rolamento 1 Em Estado Normal (a) e Estado De Degradação (b) No Segundo Experimento	66
Figura 51 – Coeficientes de <i>Wavelet</i> do Rolamento 1 Em Estado Normal (a) e Estado De Degradação (b) No Segundo Experimento	67

Figura 52 – Monitoramento Do Valor RMS De Uma Máquina.....	68
Figura 53 – Valores RMS dos Coeficientes <i>Wavelet</i> do Rolamento 1 no Segundo Experimento	68
Figura 54 – Valores De Pico dos Coeficientes <i>Wavelet</i> do Rolamento 1 no Segundo Experimento	69
Figura 55 – Fator de Crista dos Coeficientes <i>Wavelet</i> do Rolamento 1 no Segundo Experimento	69
Figura 56 – Curtose dos Coeficientes <i>Wavelet</i> do Rolamento 1 no Segundo Experimento	70
Figura 57 – Dados de Entrada do Rolamento 1 no Segundo Experimento	71
Figura 58 – Dados de Entrada Considerando as Últimas 24 horas como Informações de um Rolmanto com Defeito.....	71
Figura 59 – Arquitetura da Rede Neural Artificial.....	74
Figura 60 – Gráfico das Previsões da RNA e SVM com 24 horas de antecedência	76
Figura 61 – Gráfico das Previsões da RNA e SVM com 48 horas de antecedência	77
Figura 62 – Gráfico das Previsões da RNA e SVM com 72 horas de antecedência	79

LISTA DE TABELAS

Tabela 1 - Explicação da realização do estudo de caso.....	21
Tabela 2 – Parâmetros da SVM.....	73
Tabela 3 – Acurácia da RNA e SVM com 24 horas de antecedência	75
Tabela 4 – Matriz de Confusão da RNA e SVM com 24 horas de antecedência.....	75
Tabela 5 – Acurácia da RNA e SVM com 48 horas de antecedência	76
Tabela 6 – Matriz de Confusão da RNA e SVM com 48 horas de antecedência.....	77
Tabela 7 – Acurácia da RNA e SVM com 72 horas de antecedência	78
Tabela 8 – Matriz de Confusão da RNA e SVM com 72 horas de antecedência.....	78
Tabela 9 – Comparação do Desempenho entre a RNA e SVM.....	80

LISTA DE ABREVIATURAS E SIGLAS

AC – Corrente Alternada

CWT – Transformada *Wavelet* Contínua

db - Daubechies

DFT – Transformada Discreta de *Fourier*

DWT – Transformada *Wavelet* Discreta

FFT – Transformada Rápida de *Fourier*

FT – Transformada de *Fourier*

Hz - *Hertz*

IDFT – Transformada Discreta de *Fourier* Inversa

IMS - Centro de Sistemas de Manutenção Inteligente

I/UCR – Centro de Pesquisa Cooperativa Indústria-Universidade

NASA – Administração Nacional da Aeronáutica e Espaço

NSF – Fundação Nacional da Ciência

RBF – Base Radial Gaussiana

RMS – Raiz Quadrática Média

RNA – Rede Neural Artificial

STFT – Transformada Janelada de *Fourier*

SVM – Máquina de Vetores de Suporte

SVR – Regressão por Vetores de Suporte

SUMÁRIO

1. Introdução.....	17
1.1. Justificativa.....	17
1.2. Tema.....	18
1.2.1. Especificação do tema.....	18
1.2.2. Limitação geográfica e temporal.....	18
1.3. Problema de Pesquisa.....	19
2. Objetivos.....	19
2.1. Objetivo Geral.....	19
2.2. Objetivo Específico.....	19
3. Referencial Teórico.....	19
3.1. Estudo de Caso.....	19
3.2. Componentes do Estudo de Caso.....	22
3.2.1. Definição do componente Rolamento.....	22
3.2.1.1. Tipos de Rolamentos.....	23
3.2.1.2. Rolamento de Esfera.....	24
3.2.1.3. Rolamento de Rolo Reto ou Cilíndrico.....	24
3.2.1.4. Rolamento de Rolo Cônico.....	25
3.2.1.5. Rolamento de Rolo Esférico.....	26
3.2.1.6. Rolamento de Rolo de Agulha.....	26
3.2.1.7. Rolamento Axial de Esfera.....	27
3.2.2. Definição do componente Acelerômetro.....	27
3.3. Definição de Processamento de Sinais.....	30
3.3.1. Técnicas de Processamento de Sinais.....	32
3.3.1.1. Transformada de <i>Fourier</i>	32
3.3.1.1.1. Transformada Contínua de <i>Fourier</i> (FT).....	33
3.3.1.1.2. Transformada Discreta de <i>Fourier</i> (DFT).....	33

3.3.1.1.3. Transformada Rápida de <i>Fourier</i> (FFT).....	34
3.3.1.1.4. Transformada Janelada de <i>Fourier</i> (STFT).....	35
3.3.1.2. Transformada <i>Wavelet</i>	37
3.3.1.2.1. Escala.....	38
3.3.1.2.2. Transformada <i>Wavelet</i> Contínua (CWT).....	39
3.3.1.2.3. Transformada <i>Wavelet</i> Discreta (DWT).....	40
3.3.1.2.4. Famílias <i>Wavelet</i>	41
3.3.1.2.4.1. <i>Wavelet Haar</i>	41
3.3.1.2.4.2. <i>Wavelet Daubechies</i>	41
3.3.1.2.4.3. <i>Wavelet Biorthogonal</i>	42
3.3.1.2.4.4. <i>Wavelet Coifltes</i>	43
3.3.1.2.4.5. <i>Wavelet Symlets</i>	44
3.3.1.2.4.6. <i>Wavelet Morlet</i>	44
3.3.1.2.4.7. <i>Wavelet Chapéu Mexicano</i>	45
3.3.1.2.4.8. <i>Wavelet Meyer</i>	45
3.3.2. Parâmetros Estatísticos do Sinal no Domínio do Tempo	46
3.4. Aprendizado de Máquina.....	46
3.4.1. Tipos de Aprendizado.....	48
3.4.2. Tipos de Algoritmos de Aprendizado de Máquina.....	49
3.4.3. Máquina de Vetores de Suporte - Support Vector Machines (SVM).....	49
3.4.3.1. SVM Linear	49
3.4.3.2. SVM Não Linear	51
3.4.3.3. Função Kernel	52
3.4.3.4. Regressão por Vetores de Suporte – SVR.....	53
3.4.3.5. Parâmetros da SVM.....	54
3.4.4. Redes Neurais Artificiais (RNAs)	56
3.4.4.1. Função de Ativação	59

3.4.4.1.1. Função Linear	59
3.4.4.1.2. Função Degrau.....	59
3.4.4.1.3. Função Degrau Bipolar.....	60
3.4.4.1.4. Função Rampa	60
3.4.4.1.5. Função Sigmóide	61
3.4.4.1.6. Função Tangente Hiperbólica.....	61
3.4.4.1.7. Função Unidade Linear Retificada (ReLU)	62
4. Metodologia.....	62
4.1. Aquisição de Dados	64
4.2. Processamento dos Dados	65
4.3. Parâmetros Estatísticos	67
4.4. Definição da Saída – Label.....	70
4.5. Equilíbrio dos Dados	72
4.6. Dados de Treinamento e Validação.....	72
4.7. Algoritmos de Aprendizagem de Máquina.....	73
4.7.1. Máquina de Vetores de Suporte.....	73
4.7.2. Rede Neural Artificial	73
5. Resultados.....	74
5.1. Previsões com 24 horas de Antecedência.....	74
5.2. Previsões com 48 horas de Antecedência.....	76
5.3. Previsões com 72 horas de Antecedência.....	78
5.4. Discussão dos Resultados.....	79
6. Conclusão	80
REFERÊNCIAS.....	82
APÊNDICE A – Algoritmos de Aprendizado de Máquina.....	87

1. Introdução

Nos últimos anos houve uma crescente busca por técnicas e métodos que visam auxiliar os profissionais dando uma vantagem de estarem à frente dos acontecimentos futuros, principalmente quando estamos falando de altas produtividades em setores industriais e processos que ficam prejudicados quando sua parada é inesperada. Com isso o foco em manutenções preditivas ou condicionais trouxe um olhar mais atrativo para os responsáveis por manterem o perfeito funcionamento dos maquinários envolvidos nos processos de fabricação, geração ou criação produtiva, em técnicas de prognósticos de avaliação do estado condicional.

Por meio da utilização de prognósticos, o objetivo consiste em obter informações que permitam determinar o tempo de vida remanescente de um componente ou equipamento (Tobon-Meija *et al*, 2012). Usualmente são empregados alguns prognósticos para realizar tal análise condicional: Prognóstico baseado em modelo que busca definições analíticas para representar o sistema, Prognóstico baseado na experiência que utiliza de informações colhidas ao longo de um determinado período para configurar modelos de confiabilidade e o Prognóstico baseado em dados que usam informações vindas de sensores para criação de modelos confiáveis para entender o comportamento de degradação (Tobon-Meija *et al*, 2012).

No presente trabalho será aplicado o prognóstico baseado em dados com a utilização de informações de acelerômetros, com o emprego de técnicas de aprendizado de máquina, usando SVM (Máquina de Vetores de Suporte) e Redes Neurais Artificiais para determinar o estado condicional dos rolamentos, prevendo se a falha já se encontra presente ou não dentro do tempo determinado para se antecipar ao final da degradação do componente.

1.1. Justificativa

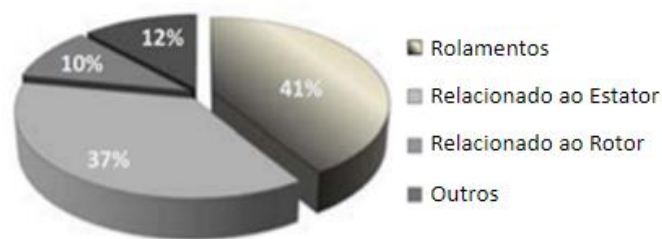
Mediante as necessidades de se prever os eventos para conseguir antecipar-se as falhas dos componentes mecânicos dos equipamentos e/ou máquinas industriais, como por exemplo, saber quando um determinado componente irá apresentar falhas, afetando o funcionamento ideal do sistema como um todo, sendo possível realizar toda uma programação de manutenção desde o correto requerimento de peças necessárias no almoxarifado até a organização da disponibilização do pessoal de manutenção para realizar tal tarefa. Sendo assim se torna vantajoso o estudo do desenvolvimento prematuro das características de uma

falha, desde como interpretar e analisar os dados até qual o melhor modelo preditivo implementar para realizar as previsões.

Os rolamentos foram escolhidos como componentes a serem avaliados e aplicados modelos de aprendizado de máquina para a previsão do estado condicional, pois estes são os equipamentos mecânicos mais comuns e regularmente vistos nos meios industriais, estando contidos em grande parte dos processos ou sistemas rotativos que realizam atividades produtivas.

Em máquinas, um dos componentes que apresenta o maior índice de falhas necessitando de manutenções são os rolamentos, Figura 1 (Tobon-Meija *et al*, 2010). Sendo assim, empregar metodologia preditiva nesses elementos e entender como um defeito se desenvolve pode aumentar a produtividade, confiabilidade, organização do setor de manutenção, segurança e reduzir muito o impacto do custo com as paradas indevidas.

Figura 1 – Percentual de falhas em componentes de elementos rotativos



Fonte: TOBON-MEIJA *et al*, 2010.

1.2. Tema

Aprendizado de máquina na engenharia mecânica.

1.2.1. Especificação do tema

Previsão de falhas em rolamentos utilizando aprendizado de máquina.

1.2.2. Limitação geográfica e temporal

O trabalho será desenvolvido durante o mês de agosto de 2020 até julho de 2021, utilizando o ambiente virtual na nuvem do *Google* para realizar as análises de dados e

criação de modelos por meio do banco de dados disponível no repositório da *National Aeronautics and Space Administration- NASA*.

1.3. Problema de Pesquisa

A grande questão que será investigada neste trabalho, será identificar qual o melhor modelo e as configurações finais dos algoritmos de aprendizado de máquina para entregar as melhores previsões do estado condicional dos rolamentos mecânicos.

2. Objetivos

2.1. Objetivo Geral

Aplicar as metodologias de aprendizado de máquina em um banco de dados que contém informações de sinais de aceleração de rolamentos, inicialmente em estado normal até o estado de degradação, com a finalidade de prever falhas futuras em componentes mecânicos de grande importância e aplicabilidade em diversos processos e setores industriais.

2.2. Objetivo Específico

- Analisar um banco de dados que possui sinais de aceleração de rolamentos.
- Processar os sinais de aceleração utilizando a Transformada de *Wavelet*.
- Calcular os parâmetros estatísticos do sinal processado.
- Utilizar SVM e Redes Neurais Artificiais para compreender os parâmetros estatísticos do sinal processado para prever o estado condicional dos rolamentos.
- Validar e comparar os melhores modelos preditivos.

3. Referencial Teórico

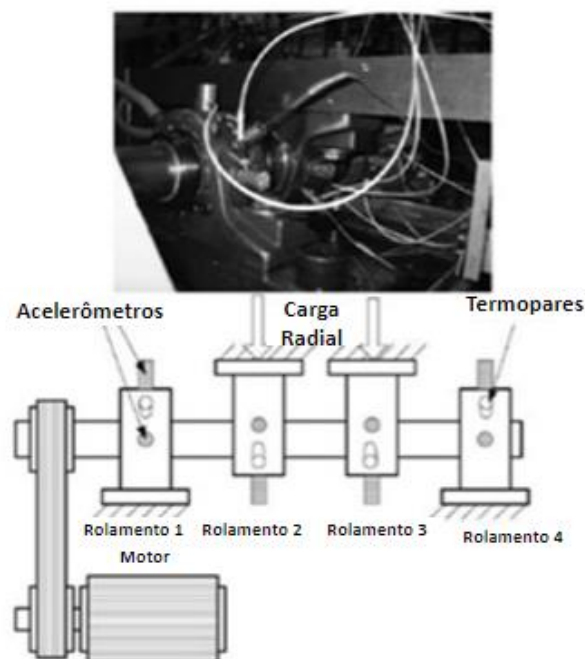
3.1. Estudo de Caso

O presente trabalho trata-se de um estudo de caso na análise de dados proveniente de acelerômetros utilizados para monitorar a condição interna de rolamentos mecânicos. Os

testes foram realizados em laboratório das empresas, *National Science Foundation's-NSF*, *Industry/University Cooperative Research Center-I/UCR* e *Center for Intelligent Maintenance Systems- IMS*, disponibilizado e extraído do repositório de dados da *National Aeronautics and Space Administration- NASA*, com sede em Washington, D.C nos Estados Unidos da América.

Sendo assim torna-se necessário mencionar quais foram os componentes utilizados no referido teste. O experimento consiste em quatro rolamentos lubrificados à força, conectados em um eixo, na qual este se encontra acoplado a um motor de corrente alternada (AC), por meio de correia, com velocidade constante de 2000 Rotações por Minuto (RPM), uma carga radial de 6000 libras aplicada ao eixo e rolamentos, conforme representado a metodologia do teste disposto na Figura 2 (Qiu *et al.*, 2006).

Figura 2 – Metodologia do experimento



Fonte: QIU, 2006, p.1083.

Acelerômetros de quartzo PCB 353B33 de alta sensibilidade foram instalados no alojamento dos mancais, para o primeiro conjunto de dados dois acelerômetros foram considerados para cada rolamento, somente um acelerômetro foi considerado no segundo e no terceiro experimento. É possível ressaltar que todas as falhas aconteceram depois de exceder o tempo de vida projetado para cada rolamento, ou seja, mais de 100 milhões de revoluções e conseqüentemente os testes foram feitos até a falha dos rolamentos. Durante a coleta de dados a taxa de amostragem foi de 20 kHz, cada conjunto de dados possui 20480 amostras,

armazenadas em arquivos TXT, com codificação ASCII, dispostos em intervalos de 10 minutos (Qiu *et al.*, 2006).

O primeiro experimento foi realizado em 22/10/2003 às 12h06 até o dia 25/11/2003 às 23h39 resultando no defeito interno do rolamento 3 e defeito no rolamento 4. O segundo experimento foi realizado em 12/02/2004 às 10h32 até 19/02/2004 às 06h22 produzindo defeito externo no rolamento 1. Por fim o terceiro experimento foi realizado em 04/03/2004 às 09h27 até o dia 04/04/2004 às 19h01 gerando defeito externo no rolamento 3, conforme disposto na Tabela 1.

Tabela 1 - Explicação da realização do estudo de caso

Experimento N° 1	
Duração:	22/10/2003 às 12:06:24 até 25/11/2003 às 23:39:56.
N° de arquivos:	2156.
N° de canais:	8.
Arranjo do canal:	Rolamento 1 - Canal 1 e 2, Rolamento 2 - Canal 3 e 4, Rolamento 3 - Canal 5 e 6, Rolamento 4 - Canal 7 e 8.
Intervalo de gravação dos arquivos:	10 minutos.
Formato dos arquivos:	ASCII.
Descrição:	No final do experimento de teste até a falha, o defeito da pista interna ocorreu no rolamento 3 e o defeito do elemento de rolo no rolamento 4.

Experimento N° 2	
Duração:	12/02/2004 às 10:32:39 até 19/02/2004 às 06:22:39
N° de arquivos:	984.
N° de canais:	4.
Arranjo do canal:	Rolamento 1 - Canal 1, Rolamento 2 - Canal 2, Rolamento 3 - Canal 3, Rolamento 4 - Canal 4.
Intervalo de gravação dos arquivos:	10 minutos.
Formato dos arquivos:	ASCII.
Descrição:	No final do experimento de teste até a falha, a falha da pista externa ocorreu no rolamento 1.

Experimento N° 3	
Duração:	04/03/2004 às 09:27:46 até 04/04/2004 às 19:01:57.
N° de arquivos:	4.448.
N° de canais:	4.
Arranjo do canal:	Rolamento 1 - Canal 1, Rolamento 2 - Canal 2, Rolamento 3 - Canal 3, Rolamento 4 - Canal 4.
Intervalo de gravação dos arquivos:	10 minutos.
Formato dos arquivos:	ASCII.
Descrição:	No final do experimento de teste até a falha, a falha da pista externa ocorreu no rolamento 3.

Fonte: Adaptado de <https://ti.arc.nasa.gov/tech/dash/groups/pcoe/prognostic-data-repository/>

3.2. Componentes do Estudo de Caso

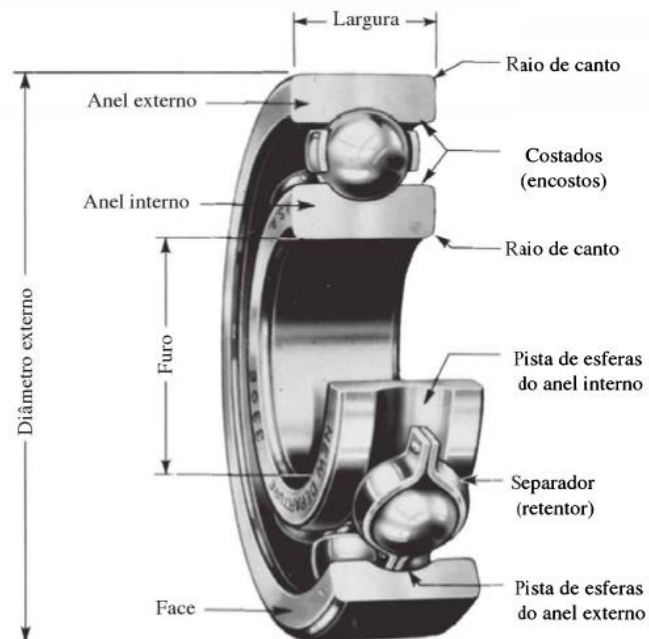
Repisa-se, o experimento analisado no presente trabalho é composto por um motor elétrico AC, correia, eixo, quatro rolamentos e dois acelerômetros em cada rolamento. Sendo assim será abordado especificamente o funcionamento do rolamento e do acelerômetro por se tratarem de partes vitais para o perfeito entendimento deste estudo.

3.2.1. Definição do componente Rolamento

Os rolamentos são componentes mecânicos comuns em diversas aplicações pela sua importância em sistemas e equipamentos.

Os rolamentos têm como objetivo receber e suportar cargas axiais, radiais ou combinação entre radial e axial para realizar a transmissão por meio dos elementos rolantes que fazem parte da sua constituição. As principais partes de um rolamento são: anel externo, anel interno, pista externa, pista interna, separador e o elemento rolante (Sassi *et al.*, 2007; Rao *et al.*, 2012). Podemos visualizar a constituição de um rolamento pela Figura 3.

Figura 3 - Constituição de um rolamento de esfera



Fonte: BUDYNASS, 2011, p.577.

Segundo Shigley podemos entender o termo rolamento como:

Usa-se o termo mancal de rolamento ou simplesmente rolamento para descrever um tipo de mancal em que a carga principal é transferida por meio de elementos em contato por rolamento em vez de deslizamento. Num mancal de rolamento o atrito estático é aproximadamente o dobro do atrito dinâmico, mas ainda é desprezível em relação ao atrito estático de um mancal de deslizamento. A carga, a velocidade e a viscosidade do lubrificante afetam sensivelmente as características de atrito de um mancal de rolamento. (SHIGLEY, 1984, p. 369).

De acordo com Budynass o termo rolamento é definido da seguinte forma:

Os termos mancal de contato com rolamento, mancais antifricção e mancais de rolamento são utilizados para descrever aquela classe de mancal na qual a carga principal é transferida por elementos em contato rolante em lugar de contato de deslizamento. Em um mancal de rolamento, a fricção de partida é cerca de duas vezes a de funcionamento, porém ainda assim insignificante em comparação com a fricção de partida de um mancal de manga. Carga, velocidade e a viscosidade de operação do lubrificante afetam as características friccionais de um mancal de rolamento (BUDYNASS, 2011, p. 576).

Sendo assim, podemos afirmar que o funcionamento do rolamento é bem simples, geralmente a pista de esferas do anel externo é fixa e a pista de esferas do anel interno é livre para realizar movimentos rotacionais, sendo que os elementos rolantes comumente empregados são as esferas que permanecem entre as duas pistas e na medida que ocorre movimento no anel interno os elementos rolantes também se deslocam, reduzindo a resistência ao deslocamento. O coeficiente de atrito dos elementos rolantes é projetado de tal forma para se torne mais vantajoso utilizá-los do que colocar um anel em contato com outro anel para transmitir movimento. Resumidamente a função de um rolamento é suportar cargas aplicadas e reduzir o atrito rotacional entre superfícies (Sassi et al., 2007; Rao et al., 2012).

3.2.1.1. Tipos de Rolamentos

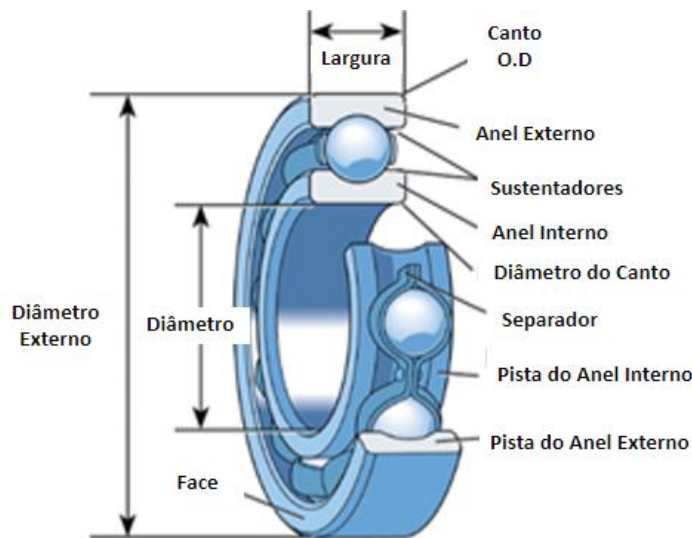
Para o presente trabalho, abordaremos alguns tipos mais frequentes de rolamentos em termos de suas aplicações, com intuito de sensibilizar o leitor das diversas aplicações possíveis de acordo com a carga na qual os mesmos serão submetidos.

Sendo que, para cada aplicação o tipo de esforço irá determinar qual rolamento usar, para situações na qual o esforço principal será axial podemos usar o rolamento do tipo axial puro, se o esforço for radial devemos utilizar um rolamento radial ou caso ocorra esforço combinado será importante selecionar um rolamento que irá suportar cargas combinadas que compreendem a axial e radial simultaneamente (Melconian, 2009).

3.2.1.2. Rolamento de Esfera

Os rolamentos de esfera são extremamente utilizados e comuns em aplicações tendo a capacidade de lidar tanto com cargas axiais como radiais, normalmente o anel externo é fixo no mancal ou carcaça do rolamento, sendo o anel interno responsável por receber o movimento do eixo e transmitir para as esferas presas em uma pista com curvatura ideal para acomodar os elementos entre o anel externo e interno, com a finalidade de reduzir o atrito no sistema, Figura 4 (Gonzalez, 2015).

Figura 4 – Rolamento de Esfera

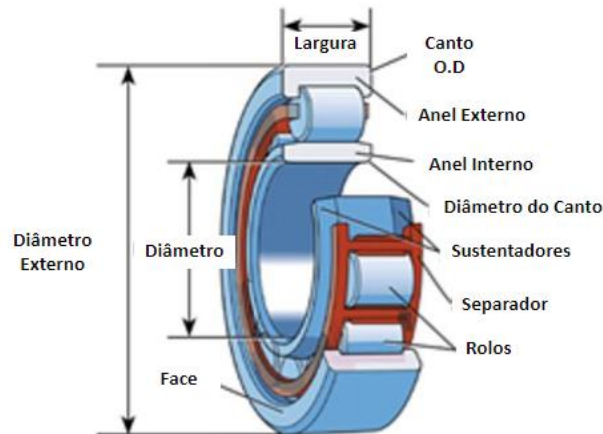


Fonte: GONZALEZ, 2015.

3.2.1.3. Rolamento de Rolo Reto ou Cilíndrico

Referente ao rolamento de rolo reto ou cilíndrico, neste tipo de configuração o elemento rolante tem a geometria de um cilindro, entre a pista externa e interna estes são presos em uma pista cilíndrica para acomodar os elementos e suas características. Por ter uma área maior a capacidade de suportar cargas também aumenta, pois ocorre uma melhor distribuição, tendo a habilidade de sofrer requisição de cargas radiais em altas velocidades, Figura 5 (Gonzalez, 2015).

Figura 5 – Rolamento Reto ou Cilíndrico

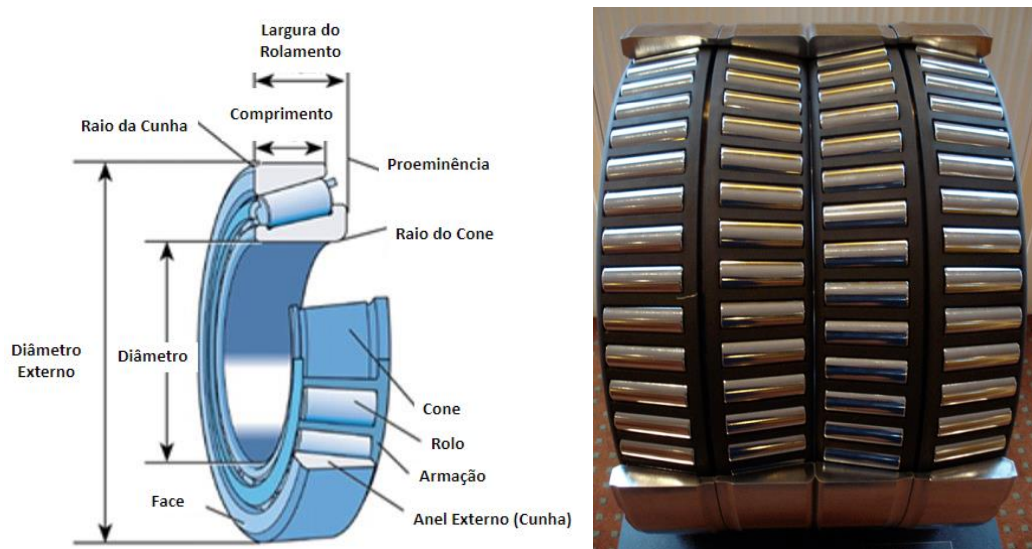


Fonte: GONZALEZ, 2015.

3.2.1.4. Rolamento de Rolo Cônico

Quanto ao rolamento de rolo cônico, para o mesmo suportar cargas combinadas, ou seja, axial e radial estes rolamentos têm os anéis e os elementos rolantes em formato de cone, o ângulo do rolo e rolamento irá determinar a carga que será suportada por este componente, Figura 6 (Gonzalez, 2015).

Figura 6 – Rolamento de Rolo Cônico

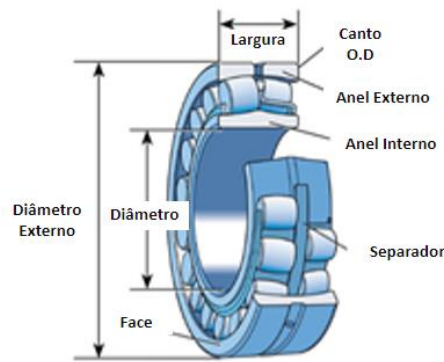


Fonte: GONZALEZ, 2015.

3.2.1.5. Rolamento de Rolo Esférico

Referente ao rolamento de rolo esférico, sua constituição contém duas divisões na pista e conseqüentemente duas fileiras de elementos rolantes em formato cilíndrico, uma pista assume configuração no anel interno e a outra em uma superfície esférica contínua no anel externo, sendo possível aplicações em que ocorre desalinhamento. Sua montagem é feita em direções opostas de uma pista para a outra, com finalidade de aceitar cargas em qualquer direção, Figura 7 (Gonzalez, 2015).

Figura 7 – Rolamento de Rolo Esférico

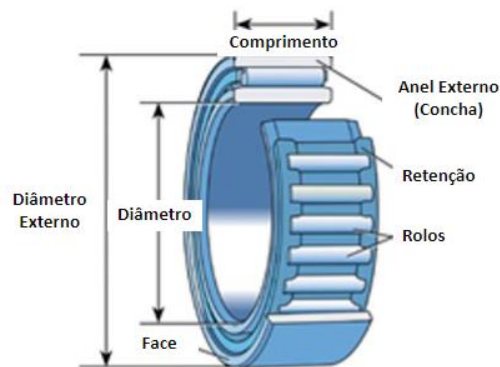


Fonte: GONZALEZ, 2015.

3.2.1.6. Rolamento de Rolo de Agulha

No rolamento de rolo de agulha, os seus elementos rolantes tem configuração cilíndrica com diâmetros pequenos e corpo alongado, sendo requisitados em operações na qual o espaço radial é módico, Figura 8 (Gonzalez, 2015).

Figura 8 – Rolamento de Rolo de Agulha

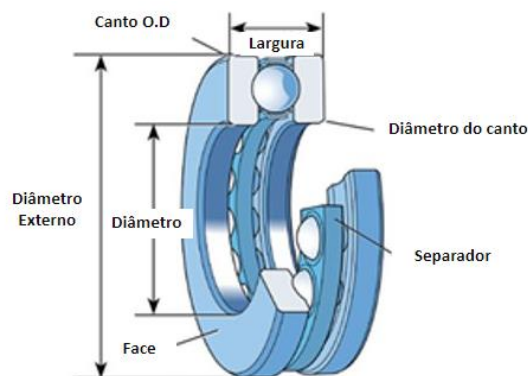


Fonte: GONZALEZ, 2015.

3.2.1.7. Rolamento Axial de Esfera

Os rolamentos axiais de esfera são projetados para receber cargas axiais e suportar altas requisições, não podendo receber componentes radiais devido ao fato de não serem projetados para tal tarefa, empregados em transmissão de movimento como, por exemplo, em engrenagens. Em sua configuração uma placa central que armazena os elementos rolantes que deslizam entre os anéis como resposta à força centrífuga, Figura 9 (Gonzalez, 2015).

Figura 9 – Rolamento Axial de Esfera

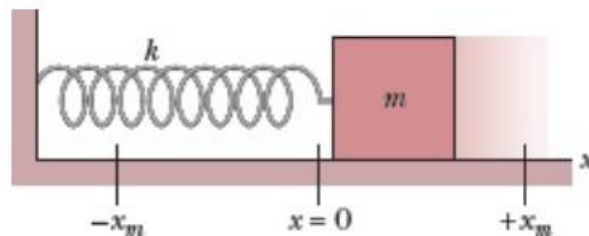


Fonte: GONZALEZ, 2015.

3.2.2. Definição do componente Acelerômetro

Os acelerômetros são instrumentos que tem a finalidade de medir a aceleração do corpo em que se deseja tal aferição. O princípio básico de funcionamento pode ser entendido por um sistema massa-mola, conforme demonstrado na Figura 10.

Figura 10 – Sistema Massa-Mola



Fonte: HALLIDAY, 2012, p. 92.

Em sua grande maioria os acelerômetros têm o seu funcionamento relacionado com a 2ª Lei de Newton, adotando uma visão simplista podemos compreender como as medições são feitas (Kardec, 2009).

No momento em que ocorre o deslocamento do bloco a mola se alonga produzindo assim uma força F , devido ao deslocamento no eixo x e a constante de rigidez da mola k , com isso a aceleração pode ser determinada dividindo o produto do deslocamento e da rigidez pela massa do bloco.

Usando a segunda Lei de Newton podemos determinar a aceleração de um corpo ou partícula dentro de um sistema massa-mola (Halliday, 2012), conforme a equação abaixo descrita:

$$a = \frac{k}{m} \Delta x \quad (3.1)$$

Quando o bloco é deslocado ocorre a produção de aceleração, conseqüentemente por se tratar de uma mola o bloco ficará oscilando durante um tempo até que não exista mais força atuando no mesmo, sendo possível determinar a frequência dessa oscilação.

O período oscilatório que surge no sistema massa-mola é produzido por se tratar de um oscilador harmônico simples linear, dependente da massa e constante de rigidez da mola (Halliday, 2012), conforme a equação abaixo descrita:

$$T = 2\pi \sqrt{\frac{m}{k}} \quad (3.2)$$

O inverso do período é a frequência, ou seja, determinando o período oscilatório do sistema é possível calcular a frequência do mesmo, mediante equação:

$$f = \frac{1}{T} \quad (3.3)$$

Por meio de um simples sistema oscilatório podemos compreender como se caracteriza o princípio de um sistema ou equipamento que realiza medições de vibração, utilizando como parâmetro a aceleração em que o corpo está sendo submetido, de maneira conceitual a vibração pode ser entendida como diversas frequências reproduzidas por um corpo em movimento oscilatório em relação a um sistema ou ponto de referência.

Vibração é caracterizada por um movimento oscilatório causado por forças que retiram o equilíbrio do sistema sendo de origem rotativa e movimentos alternados contidos em máquinas ou equipamentos, se existe um movimento oscilatório com períodos definidos mediante deslocamento em relação ao tempo, o corpo entra em vibração (Saliba, 2013).

Ainda assim a definição de vibração pode ser refletida:

Vibração ou Oscilação é qualquer movimento que se repete, regular ou irregularmente, depois de um intervalo de tempo. Assim, para o perfeito entendimento deste tipo de movimento, torna-se necessário o estudo do movimento de oscilações de um corpo em torno de uma posição de equilíbrio, bem como das forças e/ou momentos a ele associadas. Em engenharia estes movimentos ocorrem em elementos de máquinas e nas estruturas em geral, quando submetidas a ações dinâmicas. (SOEIRO, 2008, p. 6).

Para análise da saúde dos equipamentos usualmente a vibração se torna um elemento muito importante, neste processo de entender se o estado de funcionamento pode ser considerado normal ou anormal o que posteriormente poderá causar avarias e resultar na parada de máquinas refletindo na produção ao qual será interrompida.

Muitos dispositivos industriais e do nosso cotidiano geram vibração, a capacidade de identificar as bandas ou espectros de frequências de vibração nos permite associar com as condições e integridade dos dispositivos quanto ao seu funcionamento, saber como diagnosticar algum equipamento simplesmente analisando a frequência de vibração fornece o poder de na maioria das vezes realizar manutenções por condição ou denominada também de manutenção preditiva que visa antecipar algum desgaste ou eventuais falhas (Sequeira, 2013).

No momento em que a amplitude da vibração aumenta, de forma anormal em relação ao espectro de frequências habituais pode-se entender que uma provável deterioração daquele componente ou equipamento está ocorrendo, sendo um indicador de falha/defeito. Com técnicas de processamento de sinais se torna mais fácil extrair informações profundas para relacionar o sinal com a condição atual do equipamento ou máquina.

Segundo Moubray (1999) quando se desenvolve mudanças na amplitude e na frequência de funcionamento normal dos equipamentos, podemos entender que o padrão espectral começou a mudar e geralmente a integridade está sendo perdida tendo grandes probabilidades de concretizar em uma falha.

Em aplicações industriais usualmente o acelerômetro de piezoelétrico é comumente utilizado em relação aos demais disponíveis no mercado (Matsuo, 2017), Figura 11.

Figura 11 – Acelerômetro de Piezoelétrico

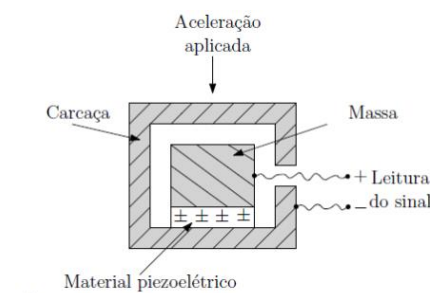


Fonte: (MATSUO, 2017)

Seu funcionamento consiste em uma massa instalada em cima de um material piezoelétrico que pode ser constituído de quartzo, sal de Rochelle, cerâmicos como o titanato de bário, materiais poliméricos e etc, que sofre um aumento ou diminuição da compressão mediante a magnitude da aceleração imposta no sólido, transmitindo esta informação que é convertida em forma de sinal elétrico, captado e variado proporcionalmente para a medição de vibração (López, 2014).

Conforme Campos (2010) o funcionamento de um acelerômetro piezoelétrico consiste em uma transformação de um sinal físico, que neste caso poderia ser um impulso mecânico ou vibração em manifestações de variações de sinal elétrico, Figura 12.

Figura 12 – Esquemático de um Acelerômetro de Piezoelétrico



Fonte: (LÓPEZ, 2014)

3.3. Definição de Processamento de Sinais

Todo conteúdo informacional precisa ser lapidado, limpo, filtrado e analisado. Sendo de notável importância conhecer as maneiras de manipular a informação com objetivo final único e exclusivo de se conseguir entender o processo de funcionamento das coisas, ao

processar um bloco ou banco de dados estamos buscando facilitar um pouco nossa perspectiva imatura de tal manifestação.

A definição de processamento de sinais pode ser conceituada por Diniz:

O processamento digital de sinais é a disciplina que estuda as regras que governam os sinais que são funções de variáveis discretas, assim como os sistemas usados para processá-los. Ela também lida com os aspectos envolvidos no processamento de sinais que são funções de variáveis contínuas utilizando técnicas digitais. O processamento digital de sinais permeia a vida moderna. Encontra aplicação nos *CD players*, na tomografia computadorizada, no processamento geológico, nos telefones celulares, nos brinquedos eletrônicos e em muitos outros dispositivos. (DINIZ, 2014, p. 1).

Portanto, o sinal pode ser definido como qualquer tipo de informação no mundo contínuo lido por equipamentos ou sensores específicos dependendo da aplicação, que possui variáveis que são independentes no seu domínio como o tempo, espaço, etc. Com intuito de representar o sinal, é necessário realizar a sua discretização, que pode ser entendido como a leitura do domínio em intervalos de tempo, ou seja, períodos de amostragem constante o que consiste em efetuar a medição da própria função.

Ainda assim sinal é qualquer grandeza física que varia no tempo transmitindo algum tipo de conhecimento daquele evento ou informação que pode descrever seu comportamento, traduzidos de forma matemática por funções de uma ou múltiplas variáveis (Sampaio, 2012).

Alternativamente, no processamento digital de sinais, processamos sequências de números usando algum tipo de *hardware* digital. Normalmente, chamamos essas sequências de números de sinais digitais ou sinais no tempo discreto. O poder do processamento digital de sinais decorre do fato de que uma vez que uma sequência de números esteja disponível para o hardware digital apropriado, podemos efetuar qualquer forma de processamento numérico sobre eles. (DINIZ, 2014, p. 1).

Por meio do sinal podemos efetivamente realizar a análise do mesmo que consiste em fazer um levantamento ou um estudo aprofundado com a finalidade de se compreender as características e leis que o regem, transformando o comportamento complexo deste em funções matemáticas mais simples. Uma forma de entender isto seria realizar, por exemplo, uma análise ondulatória de um sinal, visando a apuração de quais frequências podem estar contidas nele como forma de caracterizar esta informação.

Como menciona Haykin (2001) o objetivo de analisar e processar um sinal almeja exclusivamente extrair discernimentos e conclusões, que apoie no entendimento de tal maneira que o nível de compreensão dos mecanismos básicos do fenômeno nos dê a

possibilidade de diagnosticar o seu estado por meio da análise condicional e providenciar um tratamento adequado para antecipar um possível estado anormal.

Os sinais são geralmente não-estacionários, significa de maneira simplista que a sua variação ou seu modelo comportamental está constantemente mudando em relação ao domínio. Aplicações como imagens, sons, séries temporais, sinais de sensores, telecomunicações e etc, são áreas na qual o processamento de sinais tem crescido mediante interesse de interpretar as informações contidas e muitas vezes ocultas dentro de um sinal bruto na qual nenhum tratamento para o refinamento e extração de recursos foi feito (Sampaio, 2012).

3.3.1. Técnicas de Processamento de Sinais

Empregar técnicas de processamento facilita muito a análise dados buscando realizar tarefas de mineração, com intuito de colocar na luz aquilo que estava oculto seguindo a ideia de que no mundo real a obtenção de dados não esta isenta de ruídos ou outros agentes que camuflam as informações mais importantes de determinada atividade.

Mediante a obtenção de sinais de qualquer natureza é muito comum que estes dados sejam provenientes de sensores na qual representam alguma grandeza física em função do tempo, ou seja, a medição é realizada em determinados períodos de tempo, o que torna o conteúdo da informação contido no sinal de difícil compreensão por que as características que muitas vezes possibilitam o entendimento visível e claro precisam ser extraídas mediante técnicas matemáticas que modificam o domínio do sinal transformando para um outro domínio (Ricciotti, 2006).

Segundo Holtzmann (2019) é usualmente empregado artifícios que possibilitam modificar o sinal dependente do tempo para outra dimensão como, por exemplo, a frequência, dentre estas técnicas podemos citar: Transformada de Fourier (Contínua e Discreta), Transformada Rápida de Fourier, Transformada Janelada de Fourier, Transformada Wavelet, dentre outras.

3.3.1.1. Transformada de *Fourier*

Técnica que consiste em decompor uma série temporal em um sinal exclusivamente descrito em termos de frequência. Possibilitando descrever funções não usuais

que possuem um comportamento frequente e constante em somas de funções simples de senos e cossenos.

Podemos definir a transformada de *Fourier* de acordo com Diniz:

“[...] a transformada de Fourier de sinais no tempo discretos, que é uma generalização do conceito de resposta de frequência. Ela equivale à decomposição de um sinal no tempo discreto como uma soma infinita de senoides complexas no tempo discreto.” (DINIZ, 2014, p.123).

Mudamos a perspectiva de avaliação ao transmutar o sinal no tempo para o domínio da frequência, Figura 13.

Figura 13 – Transformada de *Fourier*



Fonte: (MISITI *et al.*, 1996)

3.3.1.1.1. Transformada Contínua de *Fourier* (FT)

Quando se trata de aplicar a FT como técnica de processamento de sinal é importante levar em consideração que a informação do tempo será perdida nesta modificação sendo que o trivial é apenas extrair as bandas de frequências, e pode ser definida como:

$$X(e^{j\omega}) = \sum_{n=-\infty}^{\infty} x(n)e^{-j\omega n} \quad (3.4)$$

Na qual o nosso objetivo é $X(e^{j\omega})$ como saída, sendo $x(n)$ o sinal de entrada contínuo e periodicidade de 2π (Diniz, 2014).

3.3.1.1.2. Transformada Discreta de *Fourier* (DFT)

A Transforma discreta de *Fourier* pode ser definida da seguinte forma:

$$W_n = e^{-j2\pi/N} \quad (3.5)$$

$$X(k) = \sum_{n=0}^{N-1} x(n)W_N^{kn}, \text{ para } 0 \leq k \leq N-1 \quad (3.6)$$

Onde $x(n)$ pode ser entendido como sequências periódicas de acordo com o período N , a frequência representada por k e $X(k)$ sendo o nosso objetivo, ou sinal de saída aplicando a DFT (Diniz, 2014).

3.3.1.1.3. Transformada Rápida de *Fourier* (FFT)

Por mais que a Transformada de *Fourier* seja extremamente eficiente em analisar sinais e modificar seu domínio para uma variável que permite maiores discernimentos, para um conjunto de dados grande, realizar a DFT se torna uma tarefa complicada pelo simples fato de que na medida em que o comprimento do conjunto de dados do sinal de entrada aumenta a complexidade de realizar o cálculo também aumenta.

Diniz descreve essa complexidade como:

“A fim de calcular a DFT e a IDFT de uma sequência de comprimento N , precisam-se realizar em torno de N^2 multiplicações complexas, isto é, a complexidade da DFT cresce com o quadrado do comprimento do sinal.” (DINIZ, 2014, p.190).

Em 1965 um algoritmo chamado Transforma Rápida de *Fourier* (em inglês, *Fast Fourier Transform* - FFT) foi desenvolvido por Cooley e Tukey resultando em uma popularização entre os acadêmicos, pois a metodologia da Transformada de *Fourier* se tornou viável para grandes conjuntos de dados com lógicas de programação extremamente rápidos para executar os cálculos, sendo um algoritmo que realiza a apuração do espectro de frequência de um conjunto de dados com N amostras no domínio do tempo consistindo em potência inteira de base dois (Misiti *et al.*, 1996).

A simplicidade da implementação de algo até então naquela época complexo para os olhos humanos pode ter extraordinários resultados e eficiência por meio da FFT, realizando transformadas e também as inversas de maneira rápida abrindo uma vasta aplicação para diversos campos aproveitarem a ideia desta teoria e verificar os seus corolários.

Como Diniz (2014, p. 190) explicita:

Por exemplo, mesmo para sinais de comprimento igual a 1024 amostras, que nem poderiam ser considerados tão longos, a redução na complexidade é da ordem de 100 vezes, isto é, duas ordens de grandeza. É desnecessário dizer que o advento desse algoritmo abriu um leque inesgotável de aplicações para a DFT, indo desde a análise de sinais até a filtragem linear rápida. Hoje, há um número enorme de algoritmos rápidos para cálculo da DFT, coletivamente conhecidos com algoritmos de FFT.

3.3.1.1.4. Transformada Janelada de *Fourier* (STFT)

Para muitas aplicações descrever um sinal em um somatório de senos e cossenos permite obter resoluções apropriadas para diversos problemas no cotidiano dando uma explicação do fenômeno de tal forma que outras técnicas não entregariam no nível de explicação daquilo que se analisa, tornando o desempenho das Transformadas de *Fourier* uma metodologia essencial no processamento de sinais.

Como explicado por Portnoff (1980), embora *Fourier* represente a frequência em sua transformada, contudo analisando o espectro da frequência não se consegue explicar ou até mesmo entregar quando tal informação aconteceu no espaço, desconsiderando a relação temporal com o conteúdo frequencial extraído.

Diniz (2014, p. 669) demonstra por meio de suas palavras o motivo pelo qual a Transformada de *Fourier* não se torna adequada, em situações que o tempo precisa ser analisado:

A transformada de Fourier calcula o conteúdo frequencial de um sinal levando em consideração sua duração completa, de $t = -\infty$ a $t = \infty$. Contudo, às vezes é interessante calcular o conteúdo frequencial de um sinal apenas numa certa região temporal. Por exemplo, se alguém fala “O bebê quer a bola”, pode-se estar interessado em analisar somente o artigo “a”, e não a frase como um todo. Claramente, a transformada de Fourier não é apropriada para uma análise desse tipo, já que sempre levará em conta a duração completa do sinal. Portanto, é desejável dispor de uma ferramenta para analisar o conteúdo frequencial local de um sinal. A transformada de Fourier de curta duração (STFT, do inglês short-time Fourier Transform), uma generalização da transformada de Fourier, é uma ferramenta com essa característica.

Portanto a Transformada de *Fourier* Janelada nos permite analisar um sinal de entrada habilitando a localizar os recursos que no caso seriam as frequências levando em consideração o tempo em que tal evento ocorreu, analisando janelas pequenas do sinal em relação a todo o conjunto de dados, ou seja, analisando cada intervalo janelado separadamente para se entender o todo, Figura 14. No lado esquerdo da Fig. 14 temos um sinal no domínio do tempo, sendo escolhido a janela/intervalo do sinal, semelhante a representação no lado direito é possível realizar a análise deste em um plano bidimensional na qual o tempo e a frequência são levados em consideração.

Figura 14 – Transformada Janelada de *Fourier*

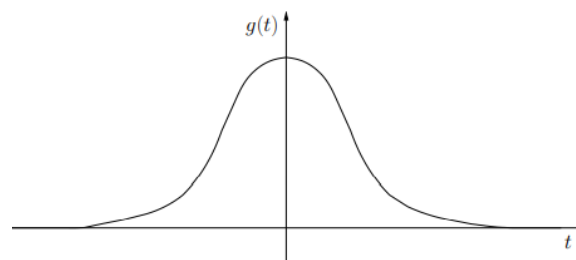
Fonte: (MISITI *et al.*, 1996)

A STFT pode ser definida como:

$$X_f(\Omega_0, b) = \int_{-\infty}^{\infty} x(t)g(t-b)e^{-j\Omega_0 t} dt \quad (3.7)$$

Podemos verificar uma janela da função $g(t)$ muito comum centrada em $t = 0$, com intuito de afastar os valores da $x(t)$ nas regiões de $t = b$ antes do cálculo da transformada de *Fourier* (Diniz 2014), Figura 15. Podemos entender pela Fig. 15 que ao definir o intervalo do sinal no tempo, as características das frequências atuam dentro de um limite de precisão, uma abordagem que não é flexível para todo o sinal.

Figura 15 – Janela Típica da SFTF



Fonte: (DINIZ, 2014, p.670)

Por mais que este método torne possível considerar o tempo de cada ocorrência mediante ao seu sinal de frequência, existe uma limitação referente à precisão da localização ao aplicar esta técnica.

Conforme Diniz (2014, p. 671):

“Para qualquer $g(t)$ que decaia mais rapidamente que $\frac{1}{\sqrt{t}}$ para $t \rightarrow \pm\infty$, tem-se”:

$$\sigma_{\Omega}^2 \sigma_b^2 \geq \frac{1}{4} \quad (3.8)$$

Isto significa que pelo Princípio da Incerteza que não é possível ter a localização exata do tempo e da frequência simultaneamente.

“Esse resultado implica que há uma resolução máxima que pode ser conjuntamente atingida na frequência e no tempo por qualquer transformada linear [...]” (DINIZ, 2014, p.671).

De acordo com o tipo de sinal e aplicação não é vantajoso utilizar a SFTF, se tornando necessário outras transformações para não se perder a precisão que é de extrema importância para o entendimento da natureza comportamental do evento.

Ainda assim Graps (1905) afirma que para sinais não-estacionários é ideal aplicar janelas com durações distintas, para que este parâmetro possa se adequar aos inúmeros eventos do sinal, sendo portanto as Transformadas *Wavelets* apropriadas para tal tarefa.

3.3.1.2. Transformada *Wavelet*

A grande diferença entre *Fourier* e *Wavelet* consiste no fato de que, esta utiliza de inúmeras ondas de curta duração para representar todo o sinal sendo possível obter informações de quando o conteúdo espectral ocorreu, ao contrário do primeiro que utiliza todo o sinal praticamente em ondas de longa duração para representar o sinal perdendo a variável tempo no procedimento.

Em 1909 Alfred Haar propôs pela primeira vez a Transformada de *Wavelet* na qual defendeu sua tese de PhD (Haar, 1909).

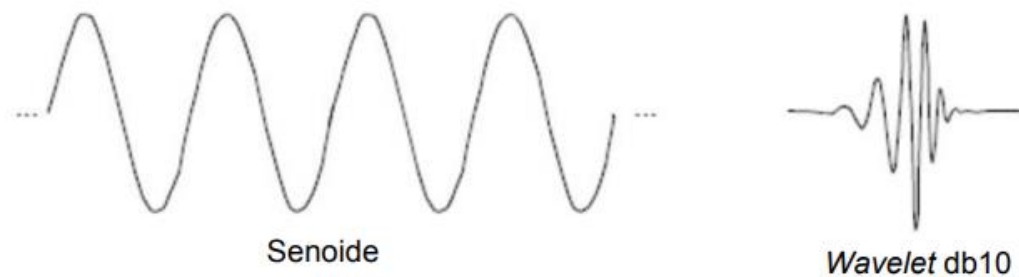
Estudos posteriores foram produzidos por Paul Levy, Jean Morlet, Alex Grossman, Stephane Mallat e Yves Meyer no final do século XX. Em 1990 Ingrid Daubechies utilizou a metodologia de análise multiresolução com *wavelet* para criar uma família de *wavelets* de base ortogonal na qual carregam seu nome em homenagem a esse feito (Daubechies, 1992).

Segundo Nielsen (1998) as funções *Wavelets* têm sido comumente utilizadas por muitos estudiosos em matemática e ciências, por ter como principal característica a possibilidade de decompor uma função hierarquicamente tendo a capacidade de analisar diversas regiões em seu domínio do tempo e também da frequência.

Os recursos extraídos podem descrever desde o detalhe mais genérico ao mais específico, a função de entrada pode ser uma imagem, uma curva, uma superfície ou sinal proveniente de sensores em diversas áreas de aplicação, a decomposição *Wavelet* oferece a metodologia para mostrar os níveis de um à função em seus mínimos detalhes e características (Stollnitz, DeRose & Salesin, 1995).

Uma senoide e sua transformada *Wavelet* db10 pode ser exemplificada, a conversão comumente é assimétrica com duração total do sinal, Figura 16.

Figura 16 – Senoide e Sua Transformada *Wavelet* db10

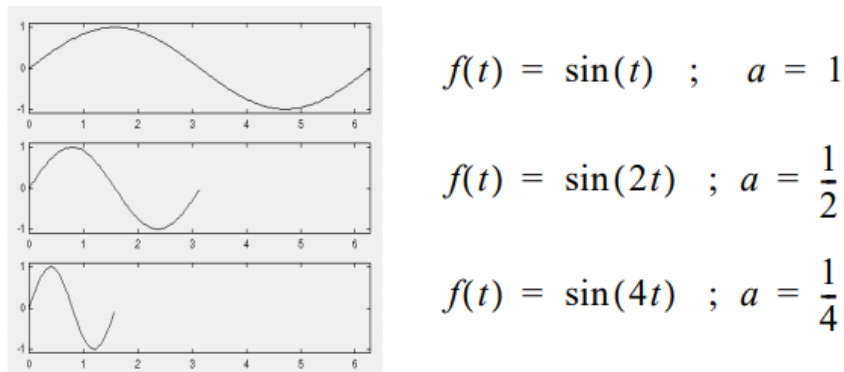


Fonte: (MISITI *et al.*, 1996)

3.3.1.2.1. Escala

Um dos parâmetros importantes para decompor uma função e realizar a transformada *Wavelet* é definir a sua escala. Entende-se por escala a virtude de comprimir ou alongar o sinal original, Figura 17.

Figura 17 – Sinal Senoidal com Fator de Escala



Fonte: (MISITI *et al.*, 1996)

De acordo com a figura 17 quanto menor a escala mais comprimida será a função de onda senoidal e conseqüentemente quanto maior o fator de escala o sinal ficará alongado, com esta mesma ideia a escala da *Wavelet* funciona. No nível de correlação podemos mencionar que a escala é diretamente dependente da frequência do sinal. Para uma baixa escala a *Wavelet* será comprimida, os detalhes do sinal mudam rapidamente e a frequência

será alta. Optando por uma escala alta a *Wavelet* será esticada, os detalhes mudaram de forma lenta e a frequência será baixa, Figura 18 (Misiti *et al.*, 1996).

Figura 18 – Sinal Com Fatores de Escala Distintos



Fonte: (MISITI *et al.*, 1996)

3.3.1.2.2. Transformada *Wavelet* Contínua (CWT)

Todo sinal proveniente de medições geralmente tende a ser discreto, e por assim dizer os dados são contabilizados em instantes ou intervalos de tempo.

A Transformada *Wavelet* Contínua também considera a discretização do sinal bruto, o termo contínuo desta técnica é referente ao fato de que para a transformação o conjunto de dados pode ser entregue em seu tamanho original ou até alguma escala estipulada pelo usuário, a capacidade de conseguir decompor um sinal em qualquer escala (Misiti *et al.*, 1996).

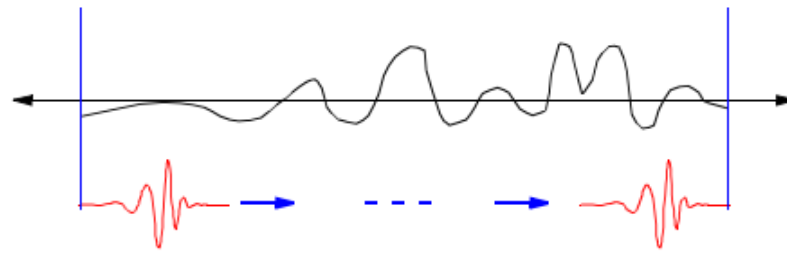
Pode ser definido por:

$$X_w(a, b) = \int_{-\infty}^{\infty} x(t)\psi_{a,b}^*(t)dt \quad (3.9)$$

O termo " * " denota conjugação complexa, podemos considerar que as *wavelets* serão supostas reais e, portanto, não levando em conta a característica complexa, $\psi(t) = \psi^*(t)$. O termo $x(t)$ é o sinal de entrada, $\psi_{a,b}(t)$ "a" e "b" sendo a escala e posição para aplicação da *wavelet* no tempo (Diniz, 2014).

Podemos entender que a decomposição considera todo o sinal para extrair os recursos da transformadas que são denominados de coeficientes, Figura 19.

Figura 19 – Transformada *Wavelet* Considerando Todo o Sinal



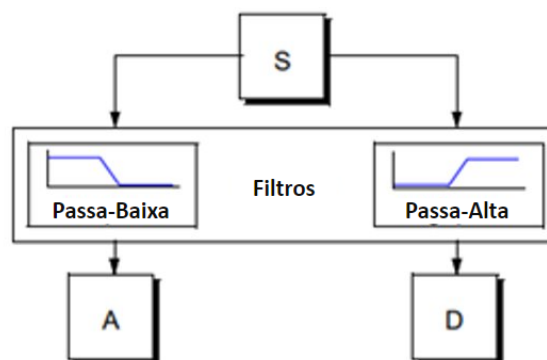
Fonte: (MISITI *et al.*, 1996)

3.3.1.2.3. Transformada *Wavelet* Discreta (DWT)

Basicamente quando utilizamos a Transformada *Wavelet* Contínua, temos um trabalho muito grande para determinar o tipo de escala mais adequado para todo o conjunto de dados, do outro lado temos a técnica Transformada *Wavelet* Discreta que ameniza o esforço de encontrar escalas para representar o sinal e visa diminuir a quantidade de dados que precisarão ser analisados.

A ideia por trás da Transformada *Wavelet* Discreta é considerar o sinal de entrada na qual os recursos deverão ser extraídos, utilizando uma abordagem que consiste na utilização de um filtro passa-bandas. Pode ser entendido como a remoção da componente de baixa frequência e da componente de alta frequência do sinal original. O sinal de entrada S passará por um filtro, como saída temos os coeficientes de aproximação pela letra A , armazenando a característica de baixa frequência e a letra D temos os coeficientes de Detalhes que são responsáveis por conter a alta frequência do sinal original (Misiti *et al.*, 1996). A Figura 20 ilustra a explicação.

Figura 20 – Transformada *Wavelet* Discreta



Fonte: (MISITI *et al.*, 1996)

O desenvolvimento matemático pode ser entendido como:

$$c_{m,n} = \int_{-\infty}^{\infty} \psi_{m,n}^*(t)x(t)dt \quad (3.10)$$

Entendemos como $x(t)$ o sinal original de entrada, a expressão $\psi_{m,n}^*(t)$ é a *wavelet* discreta interagindo no tempo e como saída temos os coeficientes de aproximação e detalhe representado por $c_{m,n}$ (Diniz, 2014).

3.3.1.2.4. Famílias *Wavelet*

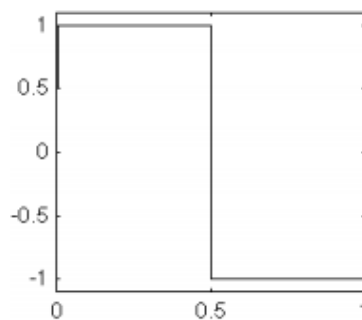
A fim de mencionar os diferentes tipos de *Wavelets*, neste trabalho será abordado as famílias mais usuais: Haar, Daubechies, Biorthogonal, Coifltes, Symlets, Morlet, Chapéu Mexicano e Meyer.

3.3.1.2.4.1. *Wavelet* Haar

O método Haar é a primeira técnica quando se menciona utilização da transformada de *Wavelet*, semelhante a um sinal digital e descontínuo, Figura 21.

Podendo ser aplicada em sinais que possuem mudança de direção forte em seus movimentos, comparado as outras famílias é praticamente uma Daubechies de nível 1 (Oliveira, 2007).

Figura 21 – Transformada *Wavelet* Haar



Fonte: (MISITI *et al.*, 1996)

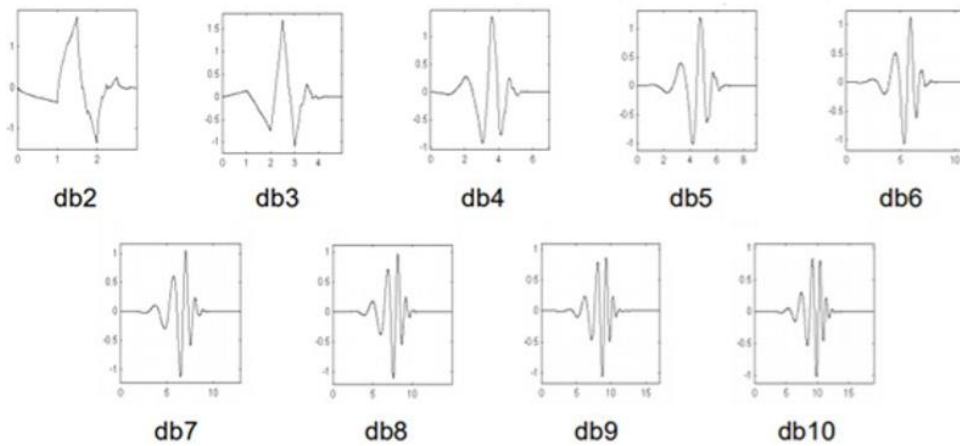
3.3.1.2.4.2. *Wavelet* Daubechies

No estudo das *Wavelets* temos Ingrid Daubechies que criou a série que carrega o seu sobrenome como forma de reconhecimento pelo estudo. O formato seria “dbN”, sendo “db” o nome desta família e o termo “N” é a ordem, por exemplo a db1 seria a ordem 1 que

representa a transformada de Daubechies, caracterizada por diversos momentos de intensidade zero (Oliveira, 2007).

Todas as ordens podem ser analisadas pela Figura 22.

Figura 22 – Transformada *Wavelet* Daubechies

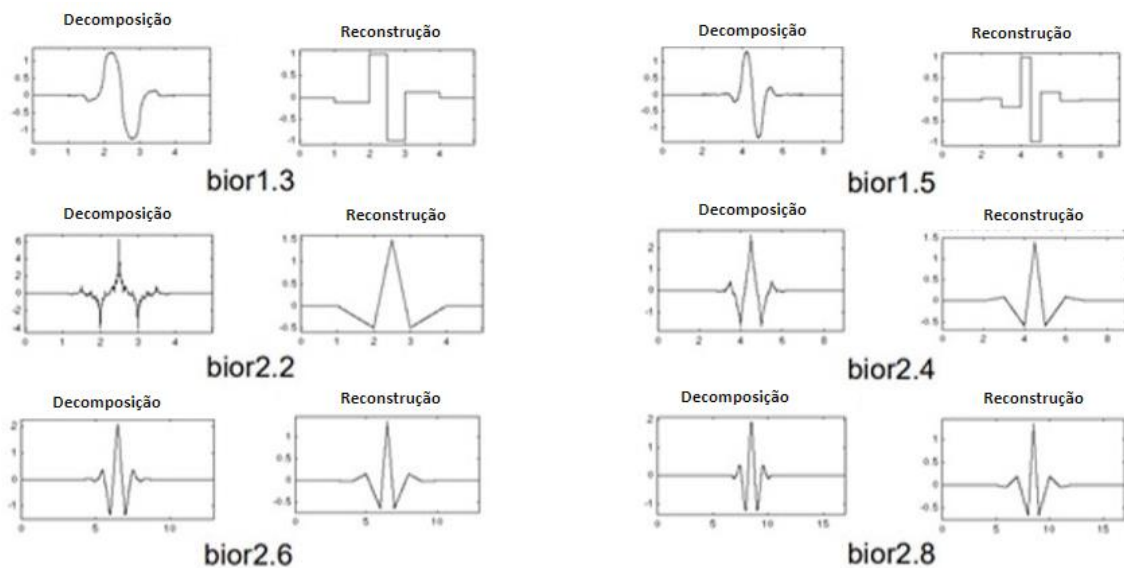


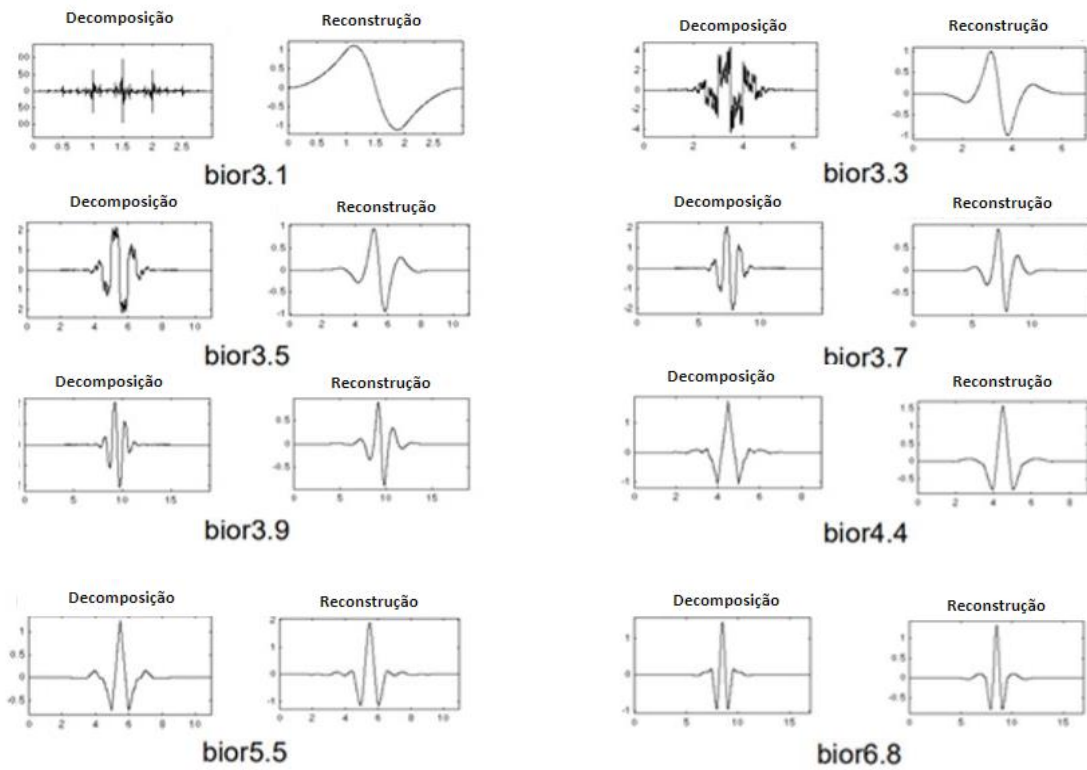
Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.3. *Wavelet* Biorthogonal

Empregada em aplicações que se faz necessário à decomposição do sinal e também a sua reconstrução de forma conjunta, como por exemplo, em imagens (Misiti *et al.*, 1996). As variações podem ser analisadas na Figura 23.

Figura 23 – Transformada *Wavelet* Biorthogonal



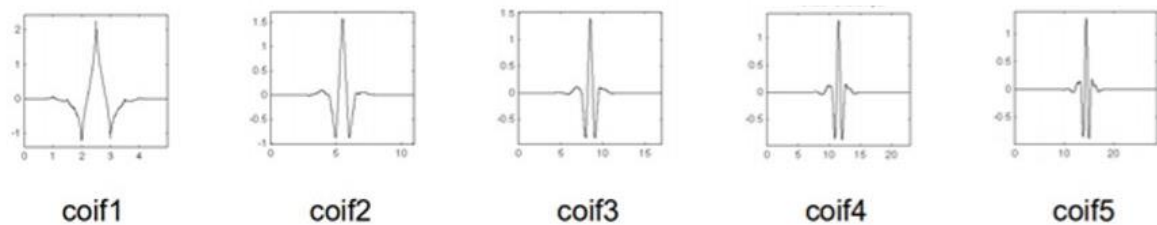


Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.4. Wavelet Coifltes

A mesma criadora das *Wavelets* Daubechies concretizou também outras famílias a pedido de Coifman, com intuito de aumentar a simetria sem descompensar e prejudicar a simplicidade da transformação (Oliveira, 2007). Representada pela Figura 24.

Figura 24 – Transformada *Wavelet* Coiflets

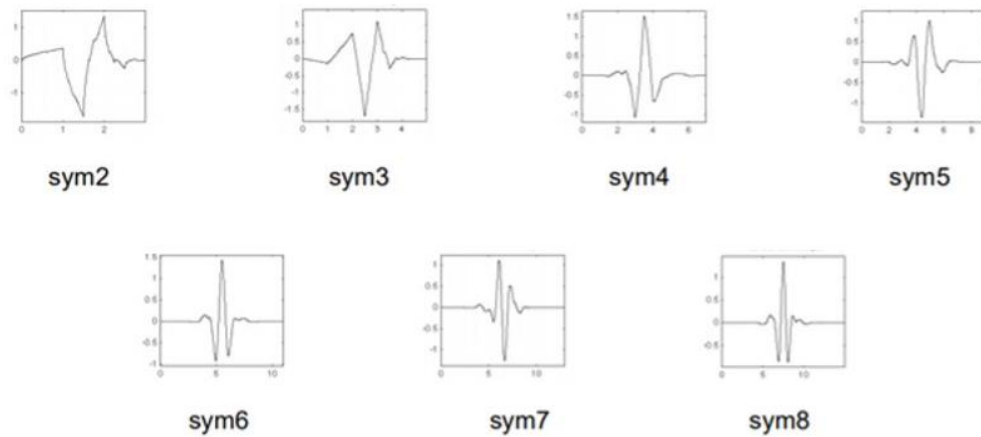


Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.5. Wavelet Symlets

São transformações semelhantes às *Wavelets* Daubechies, pois foram feitas também pela idealizadora destas com objetivo de modificar a sua primeira invenção (Oliveira, 2007). Podemos entender como ondas praticamente simétricas, Figura 25.

Figura 25 – Transformada *Wavelet* Symlets

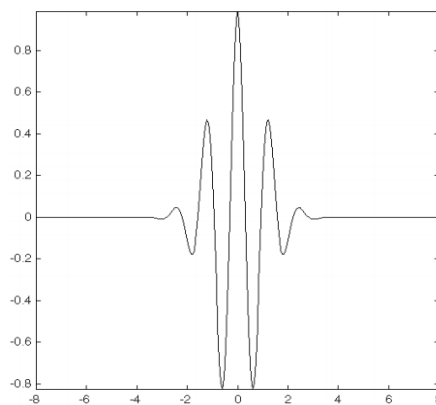


Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.6. Wavelet Morlet

Tipicamente não apresenta função de escala, porém é explícita. Por meio dos estudos de Morlet foram criadas primeiramente em aplicações de sinais sísmicos e também para a área da geofísica (Oliveira, 2007). Sua representação pode ser demonstrada pela Figura 26.

Figura 26 – Transformada *Wavelet* Morlet

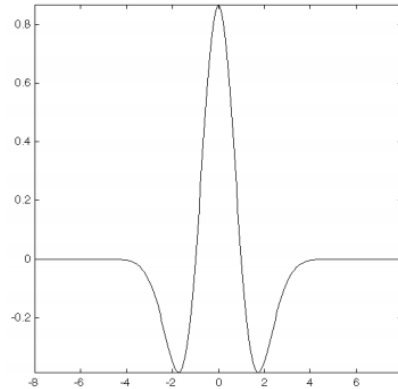


Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.7. *Wavelet* Chapéu Mexicano

Tipicamente usada em Transformações contínuas pode ser entendida como a segunda derivada de uma função normalizada gaussiana (Misiti *et al.*, 1996). Representada pela Figura 27.

Figura 27 – Transformada *Wavelet* Chapéu Mexicano



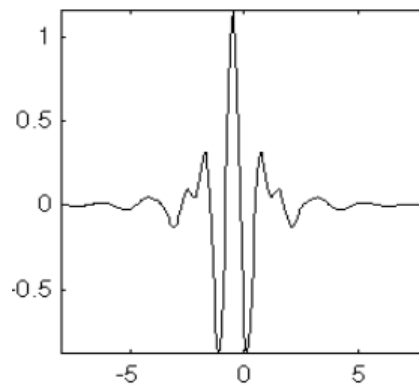
Fonte: (MISITI *et al.*, 1996)

3.3.1.2.4.8. *Wavelet* Meyer

Proposta por Meyer esta *Wavelet* possui definição no domínio da frequência, no entanto não possui definições claras no domínio do tempo o que torna sua aplicação vantajosa quando não se conhece o comportamento do sinal no tempo (Vermehren & Oliveira, 2015).

Podemos compreender essa transformada pela Figura 28.

Figura 28 – Transformada *Wavelet* Meyer



Fonte: (MISITI *et al.*, 1996)

3.3.2. Parâmetros Estatísticos do Sinal no Domínio do Tempo

Podemos utilizar algumas técnicas para extração de recursos informacionais do sinal no domínio do tempo, podemos citar: Valor da raiz quadrada média (do inglês root mean square – RMS), Valor de pico, Fator de crista e Curtose.

O RMS é utilizado como um indicador em termos de energia das vibrações do mecanismo analisado, sendo correlacionado diretamente para averiguar deterioração condicional do equipamento ou sistema, o termo x_i é o conjunto de dado de entrada de acordo com o quantitativo total N (Almeida, 2007).

$$RMS = \sqrt{\frac{1}{N} \sum_{i=1}^N x_i^2} \quad (3.11)$$

O valor de pico é a máxima amplitude contida no conjunto de dados, sua variação brusca pode ser o indicativo de eventuais falhas ou defeitos futuros (Almeida, 2007).

$$Pico = \frac{1}{2}(\max(x(t)) - \min(x(t))) \quad (3.12)$$

Fator de crista é a razão entre o Valor de pico e o RMS, sendo sua variação também está correlacionada a possíveis prognósticos de defeitos (Almeida, 2007).

$$Fator\ de\ Crista = \frac{Pico}{RMS} \quad (3.13)$$

Curtose é o parâmetro definido como o quarto momento estatístico de uma função de distribuição de probabilidade, sua variação está relacionada aos progressos de defeitos ou deteriorações (Almeida, 2007).

$$Curtose = \frac{\frac{1}{N} \sum_{i=1}^N (x(i) - \bar{x})^4}{RMS^4} \quad (3.14)$$

3.4. Aprendizado de Máquina

Com a modernização da tecnologia e suas aplicações avançadas simplificando operações complexas, se tornou útil nas tarefas do cotidiano aliar máquina ao homem. De certa forma na medida que o progresso aumenta também podemos considerar que a informação aumenta, conseqüentemente o volume de dados para armazenar e analisar se torna um desafio quando o fator humano está em jogo.

A dificuldade de processar um grande conjunto de dados trouxe como benefício o estudo e criação de métodos e técnicas de inteligência artificial, um dos ramos comumente proferidos nestes últimos anos sem dúvida é o Aprendizado de Máquina, cujo termo em inglês é Machine Learning, na qual pode ser entendido como a junção de conhecimentos de estatística provenientes da matemática, programação de computadores e algum problema que se queira resolver ou encontrar meios de entender as suas causas desde que se tenha dados suficientes para serem avaliados.

Conforme Mitchell (1997, p. 2) o conceito de aprendizado de máquina pode ser entendido como:

“Diz-se que um programa de computador aprende com a experiência E com respeito a alguma classe de tarefas T e medida de desempenho P , se o seu desempenho nas tarefas em T , conforme medido por P , melhora com a experiência E .”

Segundo Mendel & McLaren (1994) a definição de aprendizado pode ser entendida como, o processo em que o sistema inteligente é estimulado por variáveis ou fatores externos na qual está inserido com intuito de se adaptar e aprender com ele.

Os diversos campos para aplicação do aprendizado de máquina:

“Aprendizado de Máquina é inerentemente um campo multidisciplinar. Baseia-se nos resultados da inteligência artificial, probabilidade e estatística, teoria da complexidade computacional, teoria do controle, teoria da informação, filosofia, psicologia, neurobiologia e outros campos.” (MITCHELL, 1997, p. 2).

Como o interesse vem crescendo por parte de profissionais de diversas áreas as aplicações mais usuais para automatização de processos se concentram em reconhecimento de padrões provenientes de algum evento, detecção de falhas em equipamentos industriais, visão computacional, diagnósticos de exames médicos, veículos autônomos, robôs que executam atividades que antes eram exclusivas de seres humanos como jogar xadrez e até mesmo futebol, processamento de linguagem natural que consiste em criar códigos de programação com intuito de buscar entender e reproduzir a forma humana de se comunicar no entendimento do significado das expressões, análise de processos jurídicos para aumentar a velocidade da tomada de decisões (Rudin & Wagstall, 2014).

De forma simplista o Aprendizado de Máquina tem como objetivo a construção de programas computacionais que executam etapas baseados em lógicas condicionais e algébricas, como a finalidade de aprender constantemente com o histórico de informação ou banco de dados para extrair experiência se espelhando cada vez mais ao aprendizado humano.

Um dos exemplos de utilização do aprendizado de máquina na área das ciências naturais é a classificação de fitoplânctons que vivem nas águas marinhas, saber identificar corretamente os diversos tipos dessa espécie que habitam nos oceanos é um indicador da qualidade da água do mar, o que torna essa tarefa de classificação das possíveis espécies de extrema importância, quando falamos em facilitar este trabalho para os profissionais que atuam nesta área (Py *et al.*, 2016).

Ainda assim uma das áreas que também vem buscando aplicar os conceitos do aprendizado de máquina esta na saúde, criação de ferramentas desenvolvidas para apoiar os profissionais no entendimento, diagnóstico de enfermidades, avaliação prognóstica visando antecipar as mesmas e os melhores caminhos para realização de tratamentos e metodologias terapêuticas (Azevedo-Marques *et al.*, 2017).

3.4.1. Tipos de Aprendizado

O primeiro método consiste no aprendizado supervisionado, na qual podemos considerar que no conjunto de dados temos todas as variáveis dependentes e independentes, de modo que sabemos todas as respostas. Praticamente como se tivesse um professor conduzindo o aprendizado e mostrando a solução correta para aqueles dados de entrada, facilitando o entendimento ao estudar os dados contidos.

De acordo com Haykin (2007) de forma simples o aprendizado supervisionado consiste em ter a figura de um professor, tutor ou mestre que indique o caminho das respostas para a máquina e faça o aprendizado de maneira guiada, facilitando todo o processo, sempre comparando as respostas previstas com as reais e calculando os erros para um melhor aprendizado, ou seja, a experiência do tutor é transferida para a máquina.

Para o aprendizado não-supervisionado podemos considerar situações que não temos noção de qual seria a saída correta daquelas entradas, podemos pensar que temos um conjunto de dados, mas não existe um padrão determinado para direcionar o programa, consiste em entregar as variáveis independentes e deixar o computador encontrar os padrões existentes. Sendo um dos métodos complicados de avaliar o aprendizado, por que não temos meios objetivos de validar se os padrões gerados são bons de fato.

De maneira inversa ao supervisionado no aprendizado não-supervisionado o responsável por transferir seu conhecimento para o computador não existe, ou seja, somente dados de entrada são entregues para a máquina que tem a função de criar categorias de

padrões para classificar aquilo que está sendo visto sem saber a saída real ou otimizada de cada evento (Russel; Norvig, 2002).

Fazer a máquina aprender por reforço significa colocar pesos em suas ações mediante as decisões tomadas, girando em torno da recompensa ou a punição, tendo em vista que o futuro é incerto o que implica em cenários complexos. Por meio do processo, o objetivo é fazer com que a decisão que gere recompensa futura seja escolhida mediante ao histórico de acontecimentos estudado pela máquina naquela situação nova e evite ações que gerem erros ou punições (Sutton & Barton, 1998).

3.4.2. Tipos de Algoritmos de Aprendizado de Máquina

Podemos dividir os algoritmos de Aprendizagem de Máquina em duas categorias: Regressão e Classificação.

Os modelos de Regressão são métodos usuais em aplicações que precisam ser determinadas saídas quantitativas, um valor contínuo e específico deverá ser requisitado. Modelos de Classificação envolvem problemas em que iremos distinguir entre classes e simplesmente classificar as saídas como o próprio nome menciona de acordo com as características apresentadas nos dados de entrada (Duda; Hart; Stork, 2001).

3.4.3. Máquina de Vetores de Suporte - Support Vector Machines (SVM)

Máquina de Vetores de Suporte foi elaborado e criado por Vapnik, é um algoritmo usado em aprendizado supervisionado que se enquadra dentro de aplicações que envolvem classificação e também regressão (Vapnik, 1995).

Conforme Vidhya (2013) o objetivo dessa metodologia é criar um hiperplano - também denominado de Reta - que tenha a capacidade de separar os dados em classes distintas, ao criar limite de decisão para definir as classes do conjunto de dados, para facilitar a habilidade de generalização.

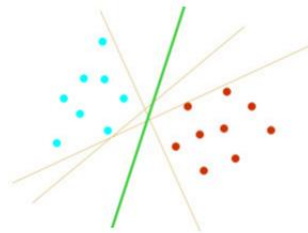
3.4.3.1. SVM Linear

Quando os dados apresentam uma estrutura que permite a separação por uma reta linear podemos usar como o próprio nome indica a SVM Linear, a partir dessa concepção muitas resoluções poderiam ser implementadas para realizar a separação e com isso a grande

indagação gira em torno de qual das prováveis soluções usar, ou seja, qual das retas de separação dos dados seria a melhor opção, Figura 29.

Como menciona Gunn (1998) o objetivo da classificação é separar os dados em classes da melhor maneira possível, contudo existem diversos classificadores lineares, ou seja, retas de separação, mas apenas um hiperplano maximiza a distância dos dados até a reta de separação sendo este o escolhido.

Figura 29 – Hiperplanos de Separação de uma SVM



Fonte: (GUNN, 1998)

A equação que separa os dados em hiperplanos é definida por:

$$(\mathbf{w} \cdot \mathbf{x}) + b = 0 \quad (3.15)$$

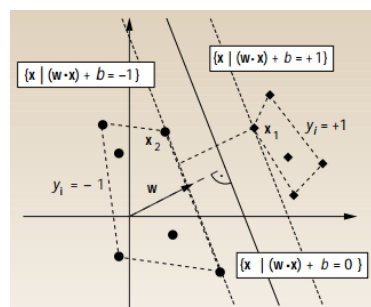
Onde $\mathbf{w} \cdot \mathbf{x}$ é o produto escalar entre os vetores \mathbf{w} e \mathbf{x} , em que \mathbf{x} é o conjunto de dados que contém as classes, \mathbf{w} é o vetor de pesos e b como bias (Gunn, 1998).

A equação acima pode ser descrita:

$$\begin{cases} \mathbf{w} \cdot \mathbf{x}_i + b \geq +1 & \text{se } y_i = +1 \\ \mathbf{w} \cdot \mathbf{x}_i + b \leq -1 & \text{se } y_i = -1 \end{cases} \quad (3.16)$$

A saída será positiva se o produto escalar com o somatório do bias for positivo maior do que 1, de maneira oposta a saída será negativa se o produto escalar com o somatório do bias for menor do que -1 (Burges, 1998).

Figura 30 – Distâncias dos Hiperplanos



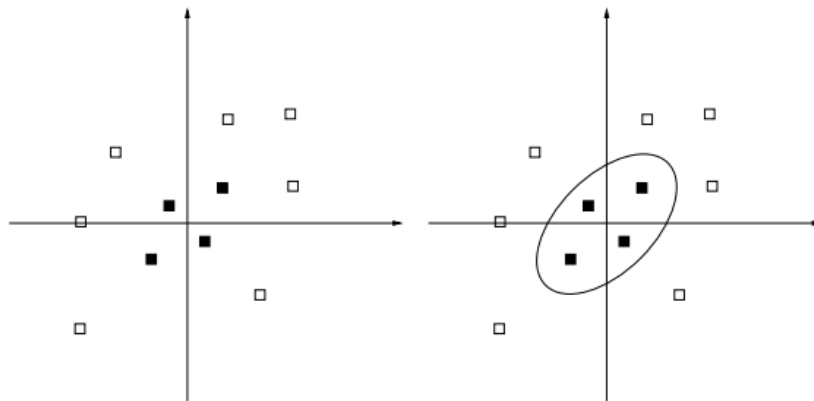
Fonte: (HEARST, 1998)

Pela Figura 30 temos um exemplo em que o conjunto de dados consiste em duas classes uma representadas por losangos e a outra por círculos, o hiperplano é a reta central em negrito que separa os dados, sendo otimizada mediante a menor distância encontrada pelo modelo entre as retas pontilhadas, que são traçadas referente ao X_1 e X_2 que recebem a denominação de vetores de suporte pela sua proximidade com o hiperplano (Hearst, 1998).

3.4.3.2. SVM Não Linear

Em conjuntos de dados que possuem alguma complexidade, ajustar a classificação por um hiperplano pode ser uma tarefa difícil, tendo em vista que a estruturação dos dados não é linear e, portanto, não permite que uma reta simples possa fazer a separação.

Figura 31 – Estrutura de Dados Não linear



Fonte: (PASSERINI, 2004)

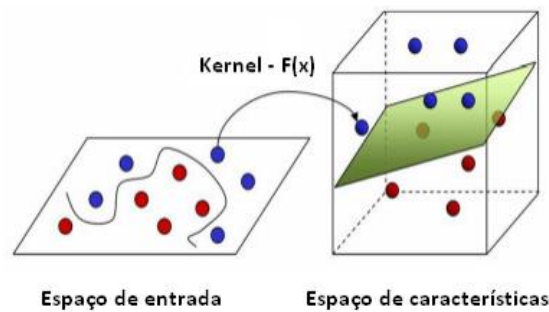
Na Figura 31 os dados representados por quadrados em negrito e quadrados sem o negrito não podem ser classificados por uma tradicional SVM Linear, a geometria que melhor descreveria essas classes e, portanto, a representação de como ocorre a distinção seria uma circunferência (Passerini, 2004).

Por meio desse exemplo temos a compreensão de que muitas vezes no mundo real os seus fenômenos não podem ser separados simplesmente de forma linear e com isso a necessidade de implementar outros métodos mais convenientes.

As SVMs não lineares utilizam o recurso de Kernel para transformar o espaço original dos dados em um espaço de maior dimensão por que a probabilidade de solucionar de forma linear também aumenta, o que significa que o problema que antes não poderia ser resolvido por técnicas lineares passa a ser possível tal utilização realizando a mudança do espaço, Figura 32.

Como menciona Gonçalves (2010) mudar o espaço dos dados permite uma resolução mais simples para a separação dos mesmos, pois diversos problemas quando podem ser expandidos no seu espaço dimensional as possibilidades de uma solução vir por meio da forma linear aumenta consideravelmente.

Figura 32 – Transformação Do Espaço



Fonte: (RABELO, 2008)

3.4.3.3. Função Kernel

A função Kernel ou produto interno é de extrema importância para situações na qual a não linearidade se apresenta fazendo com que os limites de decisão de um modelo se tornem árduo e até em alguns casos impossíveis de ser determinado por formas simples.

De forma similar (Leightley *et al.*, 2013) defini Kernel como um recurso da SVM para reduzir os riscos e as complexidades do problema a ser modelado, transformando o espaço na qual os dados estão inseridos.

A função Kernel pode ser definida como:

$$f(x) = \text{sign} \sum_{\text{vetores de suporte}} y_i \alpha_i k(x_i, x) - b \quad (3.17)$$

Inúmeras opções de funções Kernel $K(x, x_i)$ podem ser implementadas para diversos tipos de transformações de espaço nos dados em que se pretendem classificar, as funções comumente utilizadas são: Polinômios de diversos graus, Base Radial Gaussiana (RBF) e Sigmoide (Santos, 2002).

De acordo com Santos (2002) para utilizar a função de Kernel Polinomial com o grau estipulado pelo usuário devemos modificar $K(x, x_i)$, da seguinte forma:

$$K(x, x_i) = [(x, x_i) + 1]^d \quad (3.18)$$

Como dito por Santos (2002) ao utilizar uma função de base radial gaussiana devemos alterar $K(x, x_i)$:

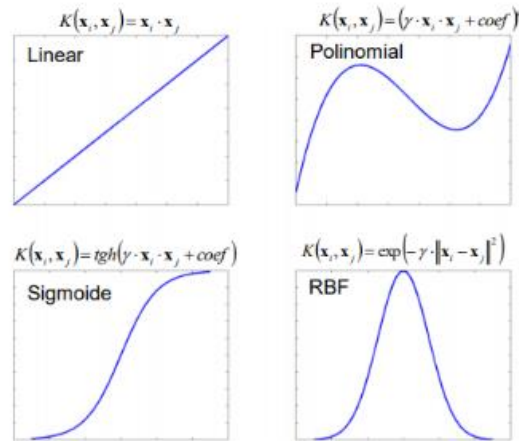
$$K(x, x_i) = e^{-\|x-x_i\|^2/2\sigma^2} \quad (3.19)$$

O Kernel RBF é a primeira opção em diversas aplicações, pois tem melhores resultados comparado ao Kernel Linear e Polinomial, porém sua modelagem e ajuste de parâmetros é uma tarefa árdua quando se pretende otimizar uma aplicação com este tipo específico de função (Novakovic; Veljovic, 2011).

Conforme Santos (2002) a função Sigmoide também denominada de Rede Neural de duas camadas, pode ser utilizada mediante a mudança de $K(x, x_i)$ por:

$$K(x, x_i) = S[v(x, x_i) + c] \quad (3.20)$$

Figura 33 – Principais Funções de Kernel



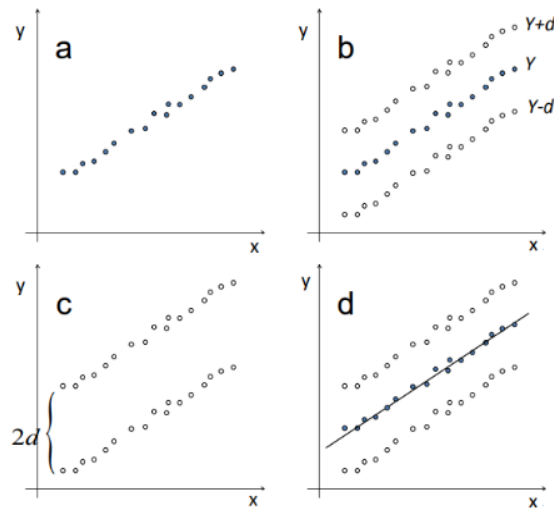
Fonte: (FILGUEIRAS, 2014)

3.4.3.4. Regressão por Vetores de Suporte – SVR

As aplicações da SVM norteiam muito para problemas que requerem somente a habilidade de classificar, ou seja, separar os dados, porém em situações que necessitamos de uma previsão de valores contínuos ou saídas quantitativas a SVR pode ser utilizada.

Da acordo com Filgueiras (2014) para os dados de entradas em x e seus valores correspondentes em y , Figura 34.a, um valor contínuo d é somado e também subtraído de cada correspondente de saída dos valores em x , Figura 34.b, sendo criado duas classes uma positiva (acima dos dados originais) e outra negativa (abaixo dos dados originais) que são denominadas de margens, Figura 34.c, com isso um hiperplano otimizado considerando os vetores das duas classes é traçado no dados originais, Figura 34.d, convertendo um problema que antes era de regressão para uma classificação binária.

Figura 34 – Regressão por Vetores de Suporte



Fonte: (FILGUEIRAS, 2014)

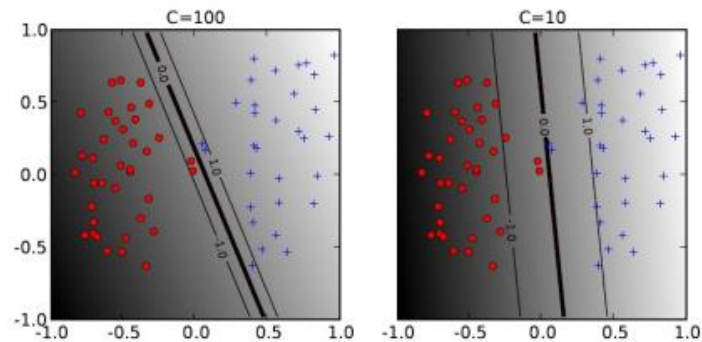
Duas formas de aplicar o SVR pode ser por meio do algoritmo ϵ -SVR e ν -SVR, no primeiro o valor de ajuste da margem é feito pelo termo ϵ , no segundo o termo ν se refere ao número de vetores de suporte escolhido para a criação do hiperplano, se for utilizado o ν -SVR em qualquer aplicação basta apenas definir ν que o parâmetro ϵ será configurado pelo próprio programa de maneira espontânea (Filgueiras, 2014).

3.4.3.5. Parâmetros da SVM

Um dos parâmetros de uma SVM é o $C(Cost)$ que está relacionado entre encontrar um caminho do meio, em relação aos erros e a criação do hiperplano de separação dos dados. Por se um valor dependente dos dados e da aplicação, sua determinação é uma tarefa árdua que requer diversos testes.

Segundo Ben-Hur e Weston (2008) quando o valor C aumenta o custo do erro aumenta e conseqüentemente o modelo tem uma margem muito reduzida e com pouca flexibilidade nas classificações, na medida que diminuimos o valor de C estamos diminuindo o custo do erro o que aumenta a margem de classificação, ou seja, temos um modelo mais flexível e menos complexo, Figura 35.

Figura 35 – Comparação entre dois parâmetros C

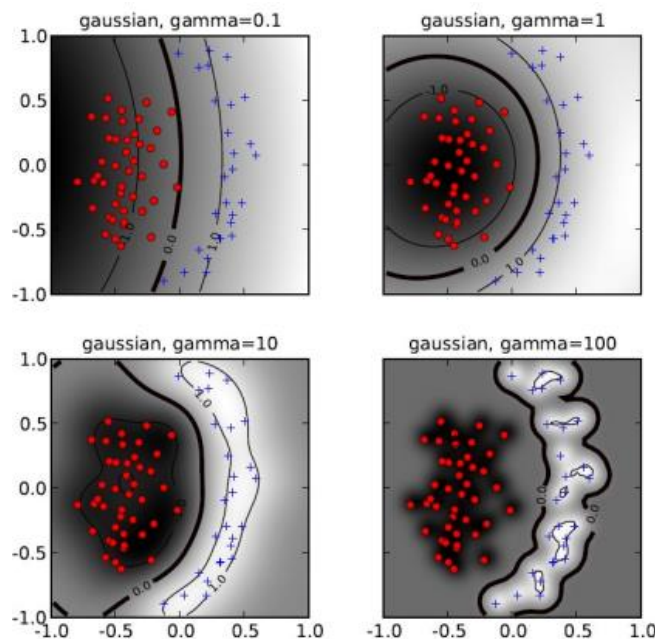


Fonte: (BEN-HUR; WESTON, 2008)

O parâmetro γ (Gamma) interfere na criação do hiperplano referente a sua complexidade, o que diretamente está relacionado com a sua flexibilidade ao fazer as classificações dos dados solicitados pelo usuário.

Simplesmente variando o fator γ e deixando C constante, para valores pequenos de Gamma o hiperplano é semelhante a uma linha ou reta linear que separa os dados, na medida em que aumentamos Gamma estamos configurando um modelo mais complexo nas separações se distanciando de um modelo especificamente linear (Ben-Hur; Weston, 20008). Conforme a Figura 36 nossa compreensão deste parâmetro pode ser manifesto.

Figura 36 – Comparação entre a variação dos parâmetros γ



Fonte: (BEN-HUR; WESTON, 2008)

3.4.4. Redes Neurais Artificiais (RNAs)

Os primeiros estudos divulgados a respeito das redes neurais artificiais compreendem o ano de 1943 por McCulloch e Pitts, o objetivo era criar uma estrutura computacional de lógicas matemáticas que se assemelhasse ao funcionamento das redes neurais biológicas, uma cadeia complexa de aprendizado baseado em programação inspirada no cérebro humano.

Em 1943 o primeiro artigo e contato da comunidade acadêmica de forma formal com o tema redes neurais artificiais foi introduzido, pois os autores evidenciaram um modelo de inteligência artificial com a metodologia do “tudo ou nada” (McCulloch *et al.*, 1943).

Conforme menciona Furtado (2019, p.1) o conceito de uma RNA pode ser entendido como:

Uma Rede Neural Artificial consiste em uma estrutura conexionista, na qual o processamento é distribuído por um grande número de pequenas unidades densamente interligadas. Este paradigma procura entender e emular as propriedades decorrentes do alto grau de paralelismo e conectividade dos sistemas biológicos. Uma rede neural é composta por um elevado número de elementos processadores, os neurônios, amplamente interligados através de conexões com um determinado valor que estabelece o grau de conectividade entre eles, denominado peso da conexão ou sinapse.

O avanço das RNAs se deu mediante ao progresso nas áreas de entendimento do comportamento humano no que rege ao funcionamento cibernético dos organismos vivos e também as evoluções no campo computacional pela criação de novos algoritmos que possibilitaram a resolução de operações matemáticas complexas.

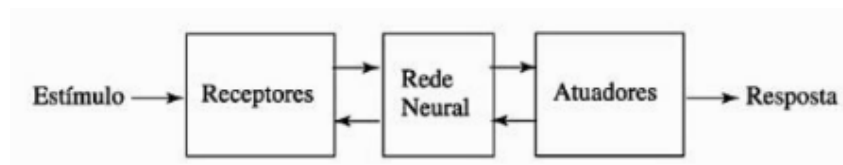
Somente em 1949 o autor Hebb propôs o conceito da forma como a uma rede neural biológica aprende, segundo o autor os nossos neurônios absorve o conteúdo do aprendizado por reforço mediante a toda cadeia de ligações de neurônios (Hebb, 1949). De forma comparativa o atraso do desenvolvimento de técnicas artificiais se deu por que pouco se conhecia sobre o funcionamento biológico das próprias redes neurais, sendo de extrema importância primeira o aprofundamento no ser humano para posteriormente buscar reproduzir nas máquinas formas de transmitir conhecimento.

As redes neurais naturais são constituídas por conjuntos de neurônios conectados entre si para o compartilhamento de informações provenientes de impulsos, estes são gerados por composições químicas em respostas a pensamentos, emoções, sentimentos, sensações

externas ou internas captadas pelo sistema nervoso sendo experimentadas pelo indivíduo e armazenadas em forma de conhecimento pelos neurônios.

De acordo com Haykin (2001) o cérebro humano pode ser entendido como um agrupamento consistindo de três estágios que constantemente está sendo alimentado e realimentado pelo próprio mecanismo e dinâmica biológica, os estímulos internos e externos são convertidos em informações por meio de sinais elétricos pelos receptores que comunicam para a rede neural que é o cérebro o grande processador humano, este gera informações para os atuadores que entregam como saída respostas orgânicas, Figura 37.

Figura 37 – Funcionamento de uma Rede Neural Biológica

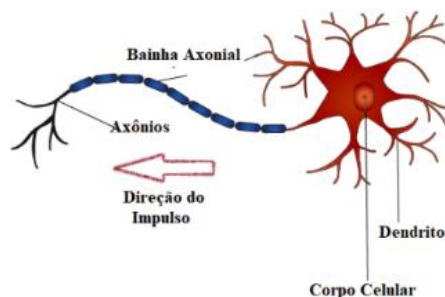


Fonte: (HAYKIN, 2008, p.32)

A composição de um neurônio pode ser explicada pelos dendritos que recebem os estímulos, o corpo celular que tem a função de processar esse estímulo vindo de um ou vários neurônios e transmitir a resposta por meio do axônio para conduzir esse sinal para diversas ramificações com intuito de atingir outras células e realizar a comunicação.

Ainda em Haykin (2001), a ligação da terminação final do axônio com os dendritos de outro neurônio é chamada de sinapse, mediante a sensibilidade de cada neurônio de acordo com a magnitude do impulso recebido, pode ser decidido por ele compartilhar o sinal ou inibir com base em seu histórico de conhecimento. Ao realizar as transmissões de impulsos cada neurônio tem sua própria função podendo amenizar ou aumentar a magnitude do impulso recebido, chamado de pesos adquirido por aprendizados anteriores, Figura 38.

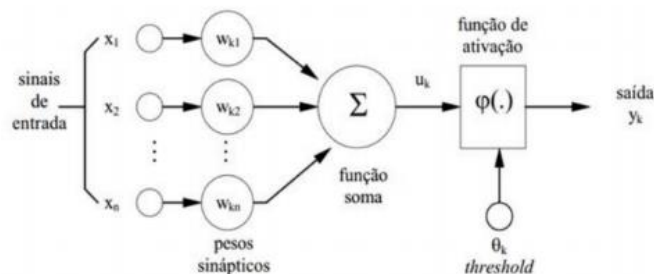
Figura 38 – Rede Neural Biológica



Fonte: (FURTADO, 2019, p.6)

De acordo com Furtado (2019) uma rede neural construída em um computador é inspirada nos neurônios biológicos, as entradas x_n são variáveis independentes de um determinado problema em que se tem a intenção de solucionar almejando um resultado final racional, usando a inteligência artificial, os sinais de entrada são multiplicados por pesos w_{kn} durante o aprendizado dando uma importância maior para os pesos maiores sendo que o peso é definido pela máquina em seus neurônios, em seguida é feita uma soma ponderada das entradas multiplicadas pelos pesos e o bias define se este valor do somatório u_k é relevante ou não, caso o somatório seja importante é então enviado para acionar uma função de ativação $\varphi(u)$ que irá gerar a saída no final desse processo y_k , Figura 39.

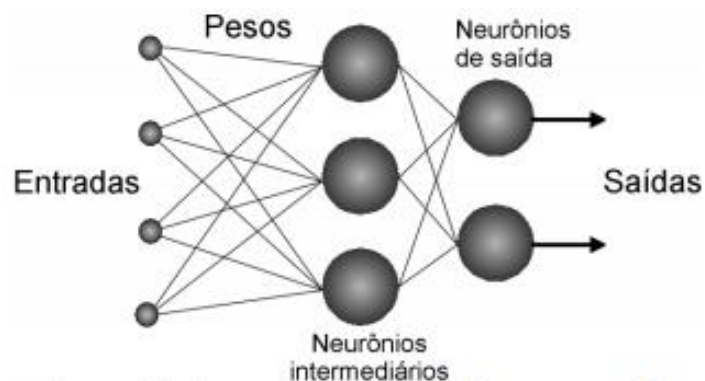
Figura 39– Funcionamento de uma Rede Neural Artificial



Fonte: (FURTADO, 2019, p.35)

A arquitetura de um RNA pode ser definida pela camada de entrada, camadas ocultas que contém os neurônios e a camada de saída, Figura 40.

Figura 40 – Arquitetura de uma Rede Neural Artificial



Fonte: (FURTADO, 2019, p.11)

3.4.4.1. Função de Ativação

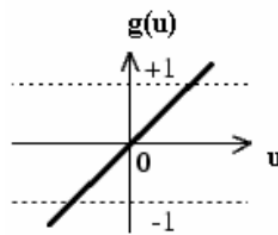
Tem por finalidade ao receber o somatório do bias entregar uma saída ou resposta para aquele tipo de sinal, podendo ser uma saída binária ou até mesmo valores quantitativos (valor contínuo).

3.4.4.1.1. Função Linear

Teremos uma saída com valores pertencentes aos reais, para qualquer valor u (Finocchio, 2014).

$$g(u) = u \quad (3.21)$$

Figura 41 – Função Linear



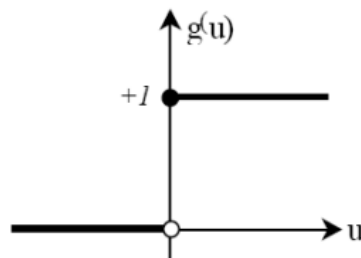
Fonte: (FINOCCHIO, 2014)

3.4.4.1.2. Função Degrau

Para valores negativos de u teremos somente a saída 0, valores positivos de u a saída terá somente o valor 1 (Finocchio, 2014).

$$g(u) = \begin{cases} 1 & \text{se } u \geq 0 \\ 0 & \text{se } u < 0 \end{cases} \quad (3.22)$$

Figura 42 – Função Degrau



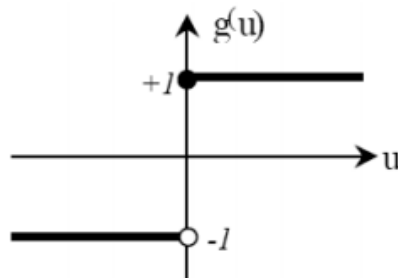
Fonte: (FINOCCHIO, 2014)

3.4.4.1.3. Função Degrau Bipolar

Se os valores de u forem positivos a saída irá produzir o valor 1, para qualquer valor de u negativo a saída será -1 (Finocchio, 2014).

$$g(u) = \begin{cases} 1 & \text{se } u \geq 0 \\ -1 & \text{se } u < 0 \end{cases} \quad (3.23)$$

Figura 43 – Função Degrau Bipolar



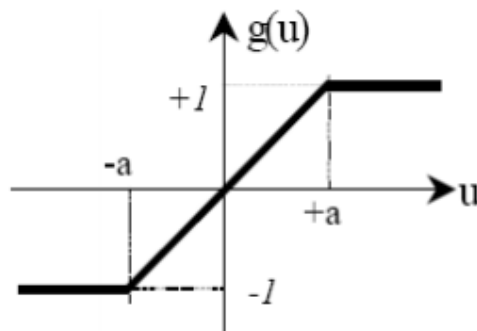
Fonte: (FINOCCHIO, 2014)

3.4.4.1.4. Função Rampa

A saída poderá alcançar qualquer valor entre -1 e 1 dentro de intervalo de $-a$ e a definidos com base na aplicação pelo usuário, para valores maiores que a temos o valor fixo de saída 1 e para valores menores do que $-a$ temos o valor fixo de -1 (Finocchio, 2014).

$$g(u) = \begin{cases} 1 & \text{se } u \geq a \\ u & \text{se } -a < u < a \\ -1 & \text{se } u \leq -a \end{cases} \quad (3.24)$$

Figura 44 – Função Rampa



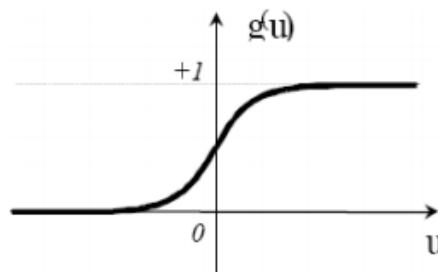
Fonte: (FINOCCHIO, 2014)

3.4.4.1.5. Função Sigmóide

A saída ficará entre 0 e 1 para qualquer valor de u , sendo o termo β o ganho da função sigmóide definida pelo usuário (Finocchio, 2014).

$$g(u) = \frac{1}{1 + e^{-\beta u}} \quad (3.25)$$

Figura 45 – Função Sigmóide



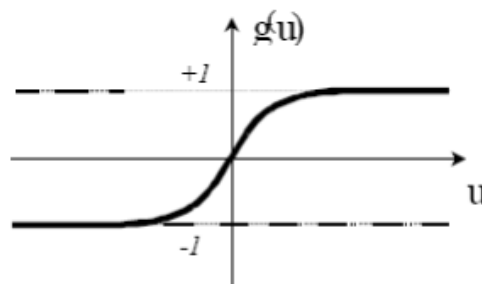
Fonte: (FINOCCHIO, 2014)

3.4.4.1.6. Função Tangente Hiperbólica

Qualquer valor de u a saída ficará dentro do intervalo de -1 a 1 (Finocchio, 2014).

$$g(u) = \tanh(u) = \frac{1 - e^{-u}}{1 + e^{-u}} \quad (3.26)$$

Figura 46 – Função Tangente Hiperbólica



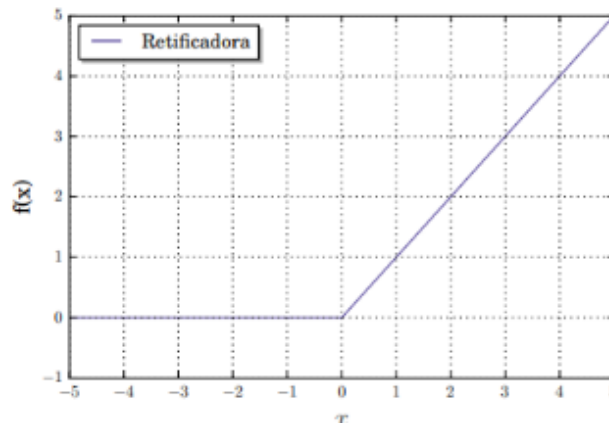
Fonte: (FINOCCHIO, 2014)

3.4.4.1.7. Função Unidade Linear Retificada (ReLU)

Produz saída nula para qualquer valor negativo e saída positiva para qualquer valor maior do que zero (Canalli, 2017).

$$f(x) = \max(0, x) \quad (3.27)$$

Figura 47 – Função ReLu



Fonte: (CANALLI, 2017)

4. Metodologia

No que diz respeito à metodologia, o presente trabalho consistirá em uma abordagem quantitativa. Tal estudo permitirá análises de dados estatísticos que poderão ser mensurados pelos resultados obtidos. “No método quantitativo, os pesquisadores valem-se de amostras amplas e de informações numéricas” (MARCONI E LAKATOS, 2006, p.269).

Quanto aos meios, à pesquisa será enquadrada como estudo de caso que se traduz pelo modo de profundidade e detalhamento, por uma descrição e análise intensiva, ainda assim pode ser definido como:

“Um estudo de caso é uma investigação empírica que investiga um fenômeno contemporâneo dentro do seu contexto da vida real, especialmente quando os limites entre o fenômeno e o contexto não estão claramente definidos.” (YIN, 2001, p.33).

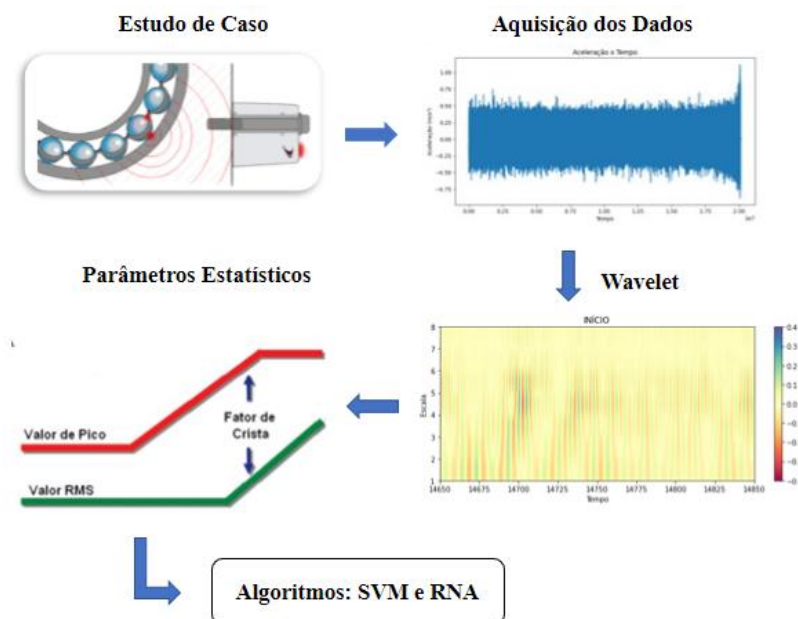
Primeiramente os dados foram disponibilizados em arquivos de texto com a identificação de cada um por meio da data e horário que foi realizado o teste no repositório da NASA, esses arquivos de textos serão armazenados após o *download* no *Google Drive* em formato *WinRar* pelo seu tamanho, sendo assim por meio do ambiente virtual *Colab* do

próprio *Google*, os dados poderão ser carregados utilizando a linguagem *Python* versão 3.8.3, etapa esta denominada de aquisição de dados.

Com os dados presentes no ambiente em nuvem a etapa de processamento será necessária, por serem dados de acelerômetros e classificados como impulsos mecânicos será utilizado a transformada de *Wavelet* ao invés da transformada de *Fourier* para analisar o conteúdo frequencial sem perder a localidade como explicado no referencial teórico, ou seja, quando essas características ocorrerem e por assim mencionar o retorno da *wavelet* será por meio de coeficientes que são entendidos como recursos característicos do sinal, estes passarão por extração de parâmetros estatísticos que consiste em obter o RMS, Pico, Fator de Pico e Curtose sendo relacionados com o estado condicional dos rolamentos de acordo com o tempo determinado para que as previsões possam ser feitas.

Por fim dois algoritmos de aprendizagem de máquina receberão os parâmetros estatísticos dos coeficientes de *wavelet* relacionados ao estado dos rolamentos, são estes uma SVM e uma Rede Neural Artificial. No processo de treinamento será considerado o experimento 1 e 3 para que a máquina possa aprender com os dados e os ajustes serem feitos com intuito final de avaliar o desempenho das previsões utilizando o experimento 2, e ao final comparar os dois algoritmos pelos resultados, Figura 48.

Figura 48 – Esquemático da Metodologia



Fonte: Próprio autor

4.1. Aquisição de Dados

Para o presente trabalho se tornou necessário à utilização de algum tipo de linguagem de programação, devido ao fato de que todas as informações experimentais foram armazenadas em arquivos se tornando essencial realizar a leitura, processamento e análise dos dados, na qual foi feita a escolha pela linguagem de programação *Python*. Por ser de fácil aprendizado, muito intuitiva, linguagem interpretada e amplamente utilizada o que resulta em uma vasta biblioteca com funções, algoritmos e códigos exemplos para tornar a implementação mais fácil para seus usuários, além de ser considerada uma linguagem de código aberto o que garante o direito de alterar e disponibilizar para qualquer um independente da finalidade.

Embora existam diversos softwares (Pycharm, VS Code, Atom, Wing, Thonny, IDLE, Pydev, Jupyter, etc) para instalar no próprio diretório do computador utilizando a memória e o processador deste, o que muitas vezes traz como consequência uma grande necessidade da máquina alocar todos os seus recursos para executar os códigos gerando perda de velocidade e indisponibilidade total da mesma para continuar uma única tarefa, podendo gerar até mesmo transtorno ao usuário. Para este trabalho será utilizado o ambiente virtual *Google Colab* que permite a qualquer programador escrever os seus códigos na linguagem *Python* utilizando o navegador sem precisar instalar ou fazer qualquer configuração, tendo acesso à unidade central de processamento do *Google*, ou seja, é possível rodar os códigos em nuvem utilizando todos os recursos do *Google* sem precisar executar as tarefas e exigir do próprio computador pessoal.

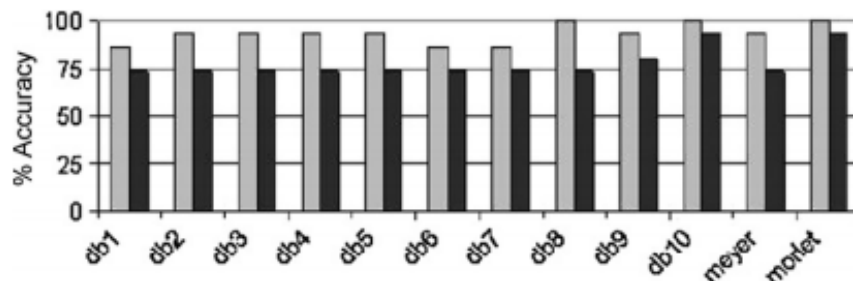
Conforme mencionado na Seção 3.1 os dados são informações provenientes de sensores acelerômetros instalados em rolamentos, obtidos de 3 experimentos independentes, disponível em um repositório aberto para qualquer um fazer o download e analisar o conteúdo ali disponibilizado. Após ter acesso aos dados, devido ao tamanho e quantidade de amostras, os arquivos foram dispostos em formato de texto, exigindo um pouco de trabalho até mesmo para conseguir colocá-los em um formato favorável à análise, para isto se tornou necessário compactar tudo em formato zip, inserir no *Google Drive* para que o *Google Colab* pudesse identificar a origem deste material, feito a compactação e o procedimento para abertura e leitura dos dados foi essencial utilizar um pouco do conhecimento para ler *byte a byte* e extrair somente os valores dos sensores acelerômetros e colocar em um *Data Frame* com formato em *XLSX (Excel)*.

4.2. Processamento dos Dados

Conforme foi exposto na Seção 3.3 a escolha da Transformada de *Wavelet* como técnica de processamento de sinal adveio devido às características dos dados. Sinais de aceleração são considerados como não estacionários, apresentando tendências, movimentações abruptas, início e fim de determinado evento, fazendo com que a sua aplicação se torne a mais adequada para este estudo de caso. Como vimos a transformada de Fourier, embora muito utilizada, não informa com precisão e maior resolução temporal características de um determinado evento, o que resulta em uma ineficiência quando o fator tempo também importa em uma análise.

A Transformada de *Wavelet* Contínua foi utilizada, pois os resultados obtidos na determinação dos coeficientes apresentam um maior detalhamento frequencial, mesmo com escalas reduzidas. Em um período curto de escalas é possível obter uma excelente resolução frequencial de eventos específicos, como por exemplo, frequências típicas de um componente com defeito ou falha. P. Konar e P. Chattopadhyay (2011) que analisaram um banco de dados contendo sinais de aceleração de um motor elétrico para identificar os estados de falhas, alcançaram bons resultados usando oito escalas e a *Wavelet Morlet*, sendo assim estes parâmetros foram utilizados para extrair os coeficientes do banco de dados deste estudo de caso. A Figura 49 demonstra a acurácia de diversas famílias *Wavelets* para identificar estados de falhas em um motor elétrico, sendo possível confirmar que a *Morlet* apresentou os melhores resultados.

Figura 49 – Desempenho de Diversas Famílias *Wavelet* (Cinza Claro – Rolamentos Normais e Cinza Escuro – Rolamentos Defeituosos)



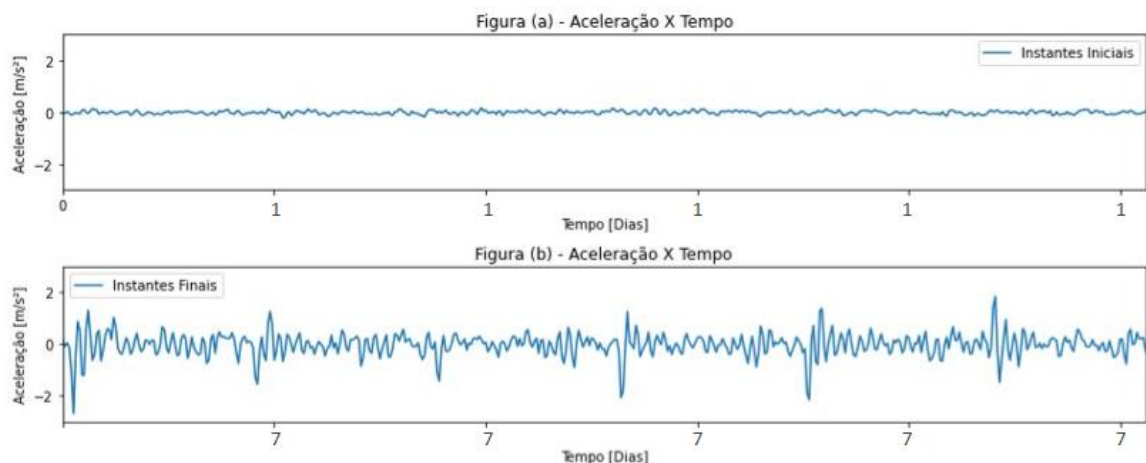
Fonte: (P. KONAR; P. CHATTOPADHWAY, 2011)

O processamento é uma das etapas mais importantes, pois sem esta técnica que pode ser considerada como um filtro as informações provenientes dos sensores acelerômetros

não fornecem detalhes referente ao espectro de frequências características que um defeito ou falha geralmente apresenta. A Figura 50 foi extraída do acelerômetro instalado no mancal do rolamento que apresentou falha no segundo experimento, na parte (a) da Figura 50 temos os instantes iniciais do experimento no primeiro dia, na qual o rolamento em operação ainda apresenta um estado condicional de funcionamento normal e na parte (b) os instantes finais no último dia, que apresentam o rolamento no estado condicional de falha ou degradação final.

A única diferença perceptível é a amplitude alterada, nos últimos minutos a aceleração aumenta muito, porém o início de uma falha ou defeito não é tão evidente, sendo necessário utilizar o processamento de sinais. No presente caso, será usada a Transformada *Wavelet*.

Figura 50 – Série Temporal do Rolamento 1 Em Estado Normal (a) e Estado De Degradação (b) No Segundo Experimento



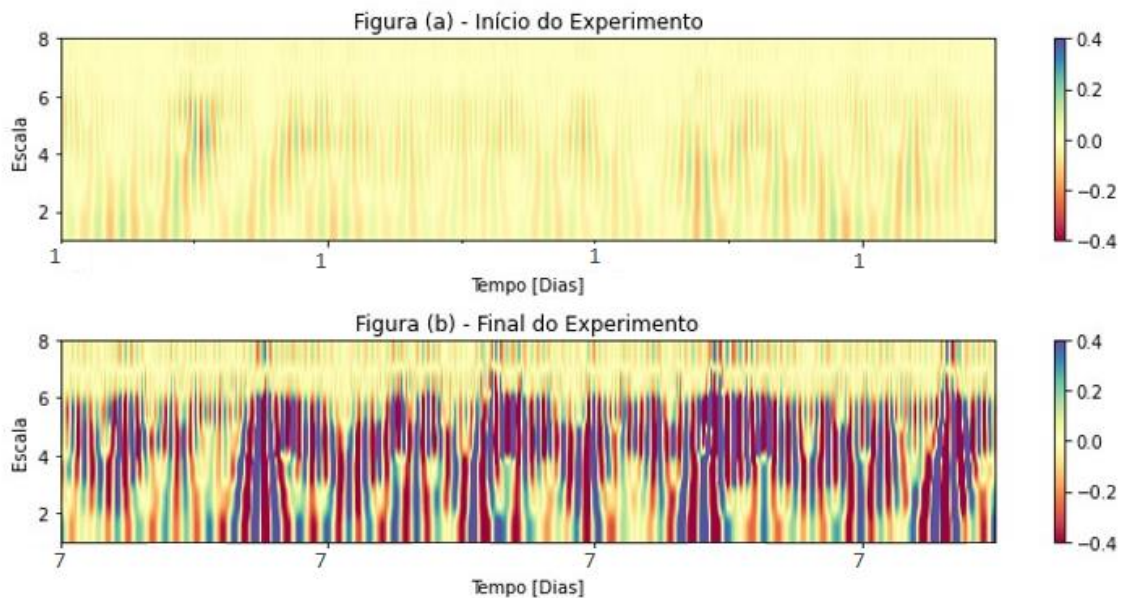
Fonte: Próprio autor

De acordo com a Figura 51 os coeficientes de Wavelet foram extraídos do segundo experimento, referente ao rolamento que apresentou defeito no final. No eixo vertical a escala esta representada conforme mencionada foi utilizado um intervalo curto, o que significa que os dados foram comprimidos fazendo com que qualquer variação fosse rapidamente captada, no eixo horizontal o tempo na qual cada amostra foi medida e a coloração representa os coeficientes *Wavelet* que fornecem informações dos coeficientes frequenciais (coloração – legenda de cores) de cada escala (eixo vertical) em cada intervalo de tempo (eixo horizontal). Para o início no primeiro dia, Figura 51 parte (a), do experimento é

bem nítido que os coeficientes ficaram em um intervalo específico que pode ser entendido como um rolamento em estado normal, ou seja, não apresenta defeito.

A Figura 51 parte (b) representa os Coeficientes de *Wavelet*, nos instantes finais no último dia, referente ao mesmo rolamento que apresentou defeito. Podemos perceber de forma comparativa que os valores dos coeficientes de *Wavelet* se elevaram muito de forma brusca, sendo possível caracterizar isto como um rolamento em estado defeituoso, permitindo uma detecção clara de falha.

Figura 51 – Coeficientes de *Wavelet* do Rolamento 1 Em Estado Normal (a) e Estado De Degradação (b) No Segundo Experimento

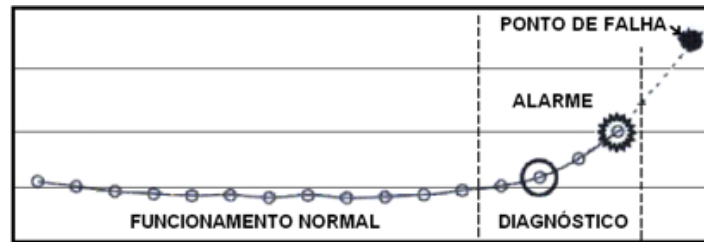


Fonte: Próprio autor

4.3. Parâmetros Estatísticos

Os parâmetros estatísticos são valores que representam um método aproximado do estado condicional (operação normal ou operação defeituosa) de um componente mediante sinais de vibração ou aceleração do mesmo, apenas avaliando a amplitude do sinal por meio de diferentes formas, são estas: RMS, Pico, Fator de Crista e Curtose. Mathias (2010) mencionou que quando a amplitude do valor de RMS de uma máquina se eleva acima do normal é um indicativo de alerta, sendo que futuramente uma falha ou defeito já está próximo de ocorrer, Figura 52.

Figura 52 – Monitoramento Do Valor RMS De Uma Máquina

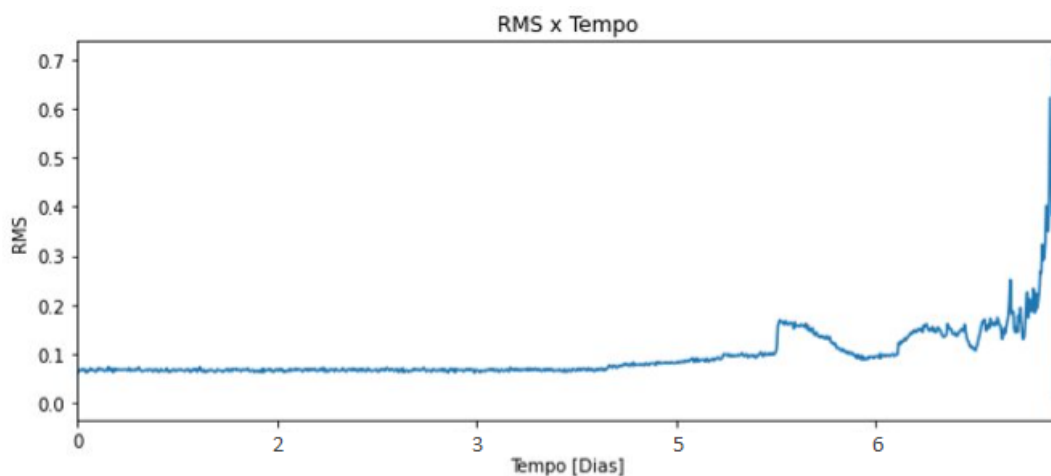


Fonte: (MATHIAS, 2010)

Os parâmetros Estatísticos podem ser calculados conforme as equações apresentadas na Secção 3.3.2, o RMS é determinado pela Equação 3.10, Pico pela Equação 3.11, Fator de Crista pela Equação 3.12 e a Curtose pela Equação 3.13.

De acordo com P. Konar e P. Chattopadhyay (2011) calcular os valores de RMS, Pico, Fator de Crista e Curtose dos coeficientes de *Wavelet* dos sinais de vibração de um motor elétrico em funcionamento normal e em estado condicional de falha, trouxeram ótimos resultados ao introduzir como dados de entrada nos algoritmos de aprendizado de máquina, sendo este método também utilizado no presente trabalho. Conforme a Figura 53, foi calculado o RMS dos coeficientes de *Wavelet* do segundo experimento, durante os 7 dias de experimento, referente ao rolamento que apresentou defeito, é perceptível o aumento na amplitude dos valores nos instantes finais quando o mesmo se aproxima do estado de degradação.

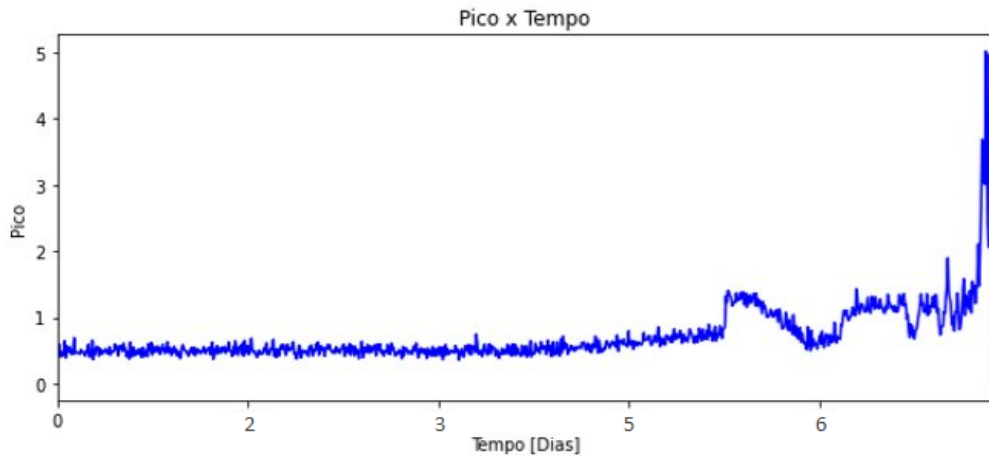
Figura 53 – Valores RMS dos Coeficientes *Wavelet* do Rolamento 1 no Segundo Experimento



Fonte: próprio autor

Assim como para o mesmo experimento, durante os 7 dias e o mesmo rolamento, o pico foi calculado e novamente nos instantes próximos do estado de degradação a amplitude aumenta consideravelmente, sendo um fator também muito sensível as variações internas do componente, Figura 54.

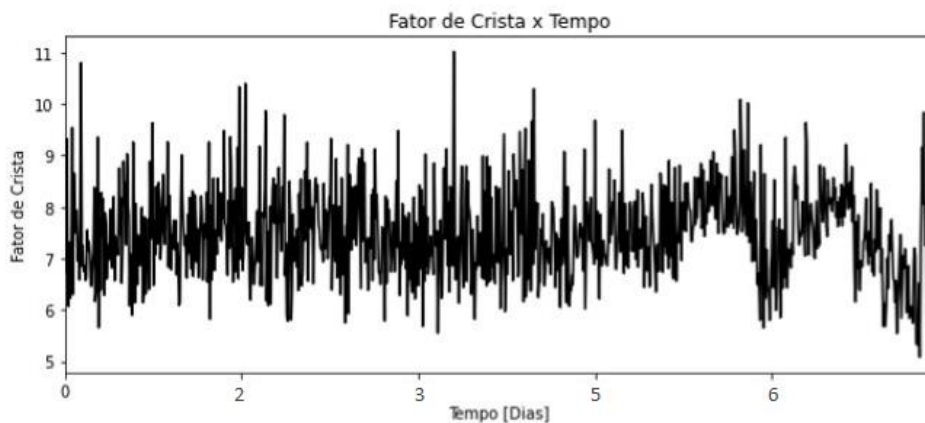
Figura 54 – Valores De Pico dos Coeficientes *Wavelet* do Rolamento 1 no Segundo Experimento



Fonte: próprio autor

O fator de crista, conforme observado na Figura 55, durante os 7 dias, um dos parâmetros que não apresentou uma tendência visível e perceptível quanto aos estágios iniciais e finais do experimento, deixando a cargo dos algoritmos interpretar e usar essas informações na classificação.

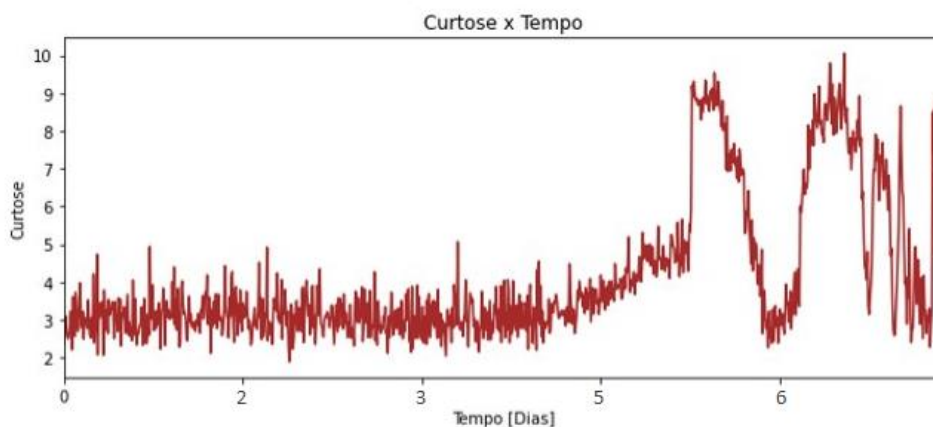
Figura 55 – Fator de Crista dos Coeficientes *Wavelet* do Rolamento 1 no Segundo Experimento



Fonte: próprio autor

Por fim, a curtose também foi calculada durante os 7 dias de experimento, sendo possível perceber um aumento na tendência e mudança do padrão ao se aproximar do estado de falha, indicando possivelmente o início do estágio de degradação do componente, Figura 56.

Figura 56 – Curtose dos Coeficientes *Wavelet* do Rolamento 1 no Segundo Experimento



Fonte: próprio autor

4.4. Definição da Saída – Label

Assim como foi exposto que a partir dos sinais de aceleração proveniente dos sensores, o processamento de sinais foi o caminho escolhido para detalhar as informações em termos de níveis ou espectro de frequências, ainda assim os parâmetros estatísticos foram utilizados para reduzir um pouco os dados sem perder a informação que a técnica de processamento forneceu. Sendo assim a RMS, Fator de Crista, Pico e Curtose dos coeficientes da Transformada de *Wavelet* foram designados para se tornarem as informações ou os dados de entrada que alimenta a entrada dos algoritmos de classificação com intuito de fazer com que estes possam aprender com o conteúdo dos dados apresentados.

Como saída foi definido utilizar o método de classificar o estado em binário, ou seja, zero significa o rolamento em estado normal sem avaria e o rolamento com defeito representado por um. Foram considerados três problemas diferentes de predição de falhas:

- I. Detecção de Iminência de Falha em menos de 24 horas.
- II. Detecção de Iminência de Falha em menos de 48 horas.
- III. Detecção de Iminência de Falha em menos de 72 horas.

O objetivo foi analisar qual a precisão dos algoritmos em prever os estados dos rolamentos em 1 dia, 2 dias e 3 dias antes da degradação final. A Figura 57 demonstra o rolamento que apresentou defeito no segundo experimento, como os dados de entrada foram organizados utilizando as funções necessárias na linguagem Python para armazenar as informações planejando com isso entregar para os algoritmos.

Figura 57 – Dados de Entrada do Rolamento 1 no Segundo Experimento

RMS	Pico	Fator de Crista	Curtose
0.060229	0.405111	6.736	2.378
0.064257	0.478369	7.444	2.828
0.064161	0.597974	9.319	3.094
0.067645	0.456128	6.743	2.696
0.066821	0.405375	6.067	2.541

Fonte: próprio autor

Podemos exemplificar pela Figura 58, como a saída foi definida no primeiro banco de dados utilizando as últimas 24 horas como caracterização de um rolamento com defeito, as primeiras linhas representam o início do experimento em que a coluna Saída ou label apresenta o valor zero (rolamento em estado normal), os três pontos são abreviações indicando que esta planilha contém mais informações e as últimas linhas representam o final do experimento na qual Saída apresenta o valor um (rolamento com defeito).

Figura 58 – Dados de Entrada Considerando as Últimas 24 horas como Informações de um Rolmanto com Defeito

RMS	Pico	Fator de Crista	Curtose	Saída
0.060229	0.405111	6.736	2.378	0
0.064257	0.478369	7.444	2.828	0
0.064161	0.597974	9.319	3.094	0
0.067645	0.456128	6.743	2.696	0
0.066821	0.405375	6.067	2.541	0
...
0.702032	4.975	7.085	6.856	1
0.433706	3.057	7.034	4.834	1
0.471959	3.191	6.762	4.135	1
0.000608	0.0033	5.447	2.279	1
0.000469	0.0028	6.008	3.090	1

Fonte: próprio autor

4.5. Equilíbrio dos Dados

Uma das grandes preocupações que giram em torno dos dados e do treinamento para as RNAs, é a questão do equilíbrio das informações a serem analisadas. Por exemplo no segundo experimento ao processar, calcular os parâmetros estatísticos e definir as últimas 24 horas como sinais de um rolamento com defeito, a proporção de sinais contendo um rolamento em estado normal se tornou muito superior, o que poderia resultar em uma tendência do algoritmo generalizar qualquer sinal como sendo de um rolamento em estado normal, mesmo que esteja com defeito. Em outras palavras a máquina se torna suscetível a “viciar” nos dados com proporção maior. No problema I (detecção em 24 horas) existem 840 amostras em estado normal e 144 em estado defeituoso, representando uma proporção aproximadamente de 85% de “zeros” em relação a “uns”.

Uma solução para balancear, ou equilibrar os dados é organizar de tal maneira que exista a mesma proporção entre sinais de um componente em estado normal e em estado defeituoso. A técnica adotada para equilibrar os dados dos três experimentos foi contabilizar o quantitativo de informações contendo saída 1 (rolamento com defeito) e saída 0 (rolamento em estado normal), criar um novo banco de dados com os sinais do rolamento com defeito e escolher de forma aleatória o mesmo quantitativo para os sinais do rolamento em estado normal. Este procedimento permite escolher sem viés sinais do rolamento em estado normal com o mesmo tamanho dos sinais do rolamento com defeito.

4.6. Dados de Treinamento e Validação

Finalizado toda a construção do banco de dados, foi preciso definir um método para realizar o procedimento de organização dos dados para treinar e validar os algoritmos com os melhores parâmetros. No primeiro experimento foram dois rolamento que apresentaram defeito ao fim de 33 dias, no segundo experimento somente um rolamento apresentou defeito ao fim de 8 dias e por fim no terceiro experimento somente um rolamento apresentou defeito ao fim de 30 dias.

Para equilibrar também os algoritmos nesse processo de aprender com os dados e serem colocados em prova o primeiro experimento e o terceiro foram escolhidos para deixar os algoritmos aprenderem com os dados, por serem os experimentos que apresentam o maior quantitativo de informações, forçando a máquina a aprender com o maior número de

informações possível e o segundo experimento para validar os algoritmos com os melhores parâmetros.

Sendo assim foi utilizado uma função que mistura com intuito de evitar que os mesmos possam criar algum tipo de viés com a ordem das informações que foram apresentadas.

4.7. Algoritmos de Aprendizagem de Máquina

As previsões de falhas nos rolamentos foram executadas por dois algoritmos bem conhecidos no campo da ciência de dados, a Máquina de Vetores de Suporte (SVM) e a Rede Neural Artificial (RNA).

4.7.1. Máquina de Vetores de Suporte

Conforme apresentado na Seção 3 que rege o trabalho bibliograficamente, os parâmetros que apresentaram os melhores resultados para a SVM foram:

Tabela 2 – Parâmetros da SVM

Parâmetros da SVM	
Kernel	Função de Base Radial (RBF)
Custo (C)	1×10^5
Gamma (γ)	1×10^{-6}

Fonte: próprio autor

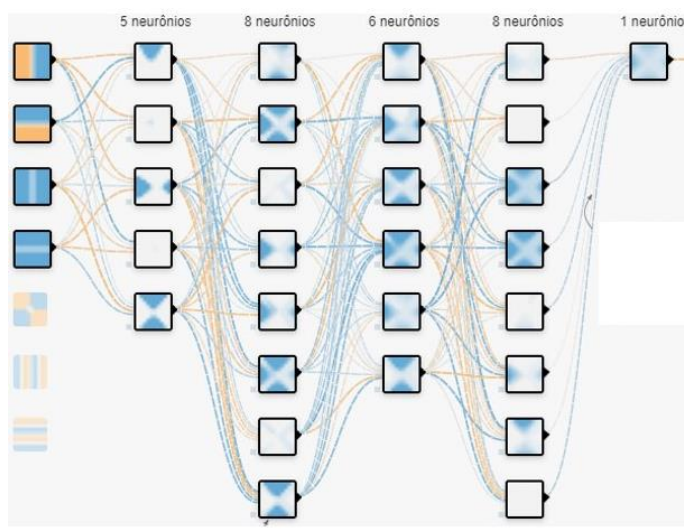
4.7.2. Rede Neural Artificial

Primeiramente a arquitetura selecionada foi de 4 variáveis de entrada, 5 camadas ocultas com configuração de neurônios 5868 e 1 neurônio de saída na última camada, conforme a Figura 59. A função escolhida na entrada por ser a mais utilizada pelos programadores e cientistas de dados foi a Relu e na saída a função Sigmoid para entregar estados em níveis binários. Para evitar sobreajuste foram usadas duas estratégias, o Dropout foi requerido para zerar metade dos neurônios em cada camada para reduzir a probabilidade de viciar nos dados dentre as 500 épocas escolhidas (SRIVASTAVA et al., 2014), e o

regularizador L2 foi utilizado também com intuito de reduzir o erro generalização, o que pode ser prejudicial no desempenho de novos conjuntos de dados.

O otimizador escolhido foi o Adam que tem como objetivo reduzir o erro entre os resultados previstos em comparação com os resultados reais/desejados (KINGMA; BA, 2014). A função para calcular a perda de entropia ou o erro entre os resultados previstos e os reais foi a Binary Cross Entropy (Entropia Binária Cruzada), o Batch Size utilizado foi com o valor de 100 o que refere a quantidade de treinamento que será executada em cada época, de forma a sinalizar para a rede neural que os dados de um rolamento com defeito devem ser priorizados em termos de aprendizado foi necessário utilizar Peso nas Amostras (*Sample Weight*) com valor maior do que 5 o que significa dar atenção aos dados com valor 1, por fim foi utilizado 7 camadas com neurônios diversificados em cada uma e escolhido o que gerou o melhor resultado.

Figura 59 – Arquitetura da Rede Neural Artificial



Fonte: próprio autor

5. Resultados

Por meio da metodologia os resultados podem ser analisados e discutidos, tendo em vista que o comparativo entre os dois algoritmos de aprendizado de máquina será exposto, assim como o apontamento daquele que apresentou o melhor desempenho neste trabalho.

5.1. Previsões com 24 horas de Antecedência

Podemos comparar a acurácia obtida pelos dois algoritmos no treinamento e na validação mediante a Tabela 3.

Tabela 3 – Acurácia da RNA e SVM com 24 horas de antecedência

	Acurácia no Treinamento (%)	Acurácia na Validação (%)
RNA	56,69	85,36
SVM	95,02	90,45

Fonte: próprio autor

Apenas analisando de forma mais superficial a Tabela 3, percebe-se que a SVM obteve um melhor entendimento dos dados durante o treinamento e os dois algoritmos apresentaram um excelente desempenho na validação, porém é necessário aprofundar um pouco mais na análise para entender como ocorreu as previsões em relação aos valores reais. Para isto utilizamos uma técnica muito importante na ciência de dados que é a matriz de confusão.

Tabela 4 – Matriz de Confusão da RNA e SVM com 24 horas de antecedência

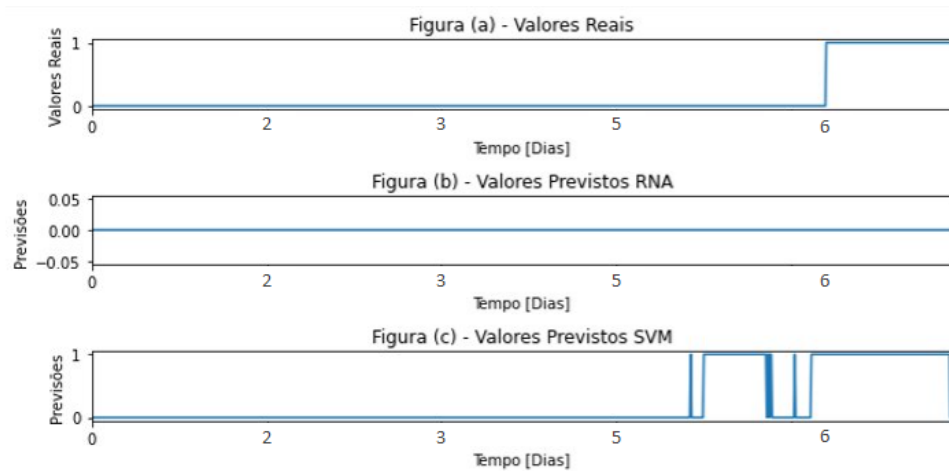
	RNA					
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras	0 (VP)	144 (FP)
Rolamento com Defeito (1)	0	0	0	144	0 (FN)	840 (VN)
Rolamento em Estado Normal (0)	85	100	92	840		

	SVM					
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras	142 (VP)	2 (FP)
Rolamento com Defeito (1)	61	99	75	144	92 (FN)	748 (VN)
Rolamento em Estado Normal (0)	100	89	94	840		

Fonte: próprio autor

Conforme a Tabela 4, a avaliação dos algoritmos durante a validação torna-se mais criteriosa. Como o objetivo é de prever falhas deve-se levar em consideração a análise do *Recall*, pois este penaliza o Falso Negativo (FN) e não tolera este tipo de erro, contudo o *F1 - score* também pode ser utilizado quando não se sabe ao certo se o *Recall* ou a *Precisão* (penaliza o Falso Positivo – FP) é a opção mais importante de acordo com a aplicação do modelo (*F1 -score* calcula a média harmônica entre a *Precisão* e o *Recall*). Dos 144 sinais de amostras de um rolamento com defeito a RNA não acertou nenhum, obtendo um *Recall* inferior a SVM que acertou 99% destes dados. Dos 840 sinais de amostras do rolamento em estado normal a RNA obteve uma *Recall* superior, acertou 100%, enquanto a SVM acertou 89% destes dados. Ainda assim para prever rolamento em estado de falha a SVM apresentou um *F1-score* de 75% sendo superior a RNA, comprovando ainda mais a sua eficiência.

Figura 60 – Gráfico das Previsões da RNA e SVM com 24 horas de antecedência



Fonte: próprio autor

A Figura 60 retrata o desempenho dos algoritmos durante a validação, na Figura 60 (a) temos os valores reais para compararmos com as previsões. Embora a acurácia da RNA tenha sido promissora podemos perceber na Figura 60 (b) que o algoritmo não apresentou um desempenho satisfatório, pois o mesmo apenas entregou durante toda a etapa de validação que o rolamento estava em estado normal, o que não aconteceu se analisarmos os valores reais, ou seja, este adquiriu um viés e apenas classificou todos os sinais que recebeu em nível zero. A Figura 60 (c) representa o desempenho da SVM que se mostrou muito melhor, pois realmente nas últimas 24 horas o algoritmo classificou todos os sinais como de um rolamento com defeito semelhante aos valores reais, sendo que este ainda entendeu que o rolamento já estava apresentando defeito antes mesmo das últimas 24 horas, devido ao fato de que os sinais já estavam sendo classificados como de um componente em falha.

5.2. Previsões com 48 horas de Antecedência

A acurácia obtida pelos dois algoritmos no treinamento e na validação, pode ser avaliada mediante a Tabela 5.

Tabela 5 – Acurácia da RNA e SVM com 48 horas de antecedência

	Acurácia no Treinamento (%)	Acurácia na Validação (%)
RNA	59,48	72,25
SVM	87,09	96,04

Fonte: próprio autor

Pelas informações contida na Tabela 5, percebemos que a RNA obteve uma acurácia melhor durante a validação enquanto a SVM apresentou um bom desempenho no treinamento e também na validação demonstrando uma maior congruência, sendo necessário avaliar por outras métricas como os dois algoritmos realmente se comportaram.

Tabela 6 – Matriz de Confusão da RNA e SVM com 48 horas de antecedência

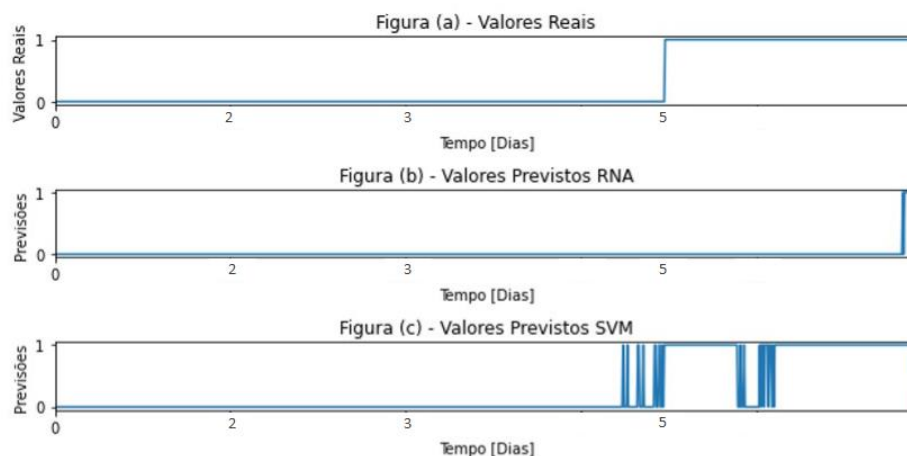
	RNA					
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras		
Rolamento com Defeito (1)	100	5	9	288	14 (VP)	274 (FP)
Rolamento em Estado Normal (0)	72	100	84	696	0 (FN)	696 (VN)

	SVM					
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras		
Rolamento com Defeito (1)	96	90	93	288	258 (VP)	30 (FP)
Rolamento em Estado Normal (0)	96	99	97	696	10 (FN)	686 (VN)

Fonte: próprio autor

Pela matriz de confusão na Tabela 6, novamente devemos ter um olhar mais intolerante em relação ao Falso Negativo (FN) e portanto analisar o *Recall*. É possível perceber que novamente o *Recall* da RNA em classificar um rolamento em estado normal foi um pouco melhor, acertou 100%, do que a SVM que acertou 99%. Para classificar este mesmo componente em estado defeituoso o *Recall* da SVM foi superior, acertou 90% destes dados, enquanto a RNA obteve apenas 5% de acerto das amostras em estado de falha. Avaliando também uma segunda métrica, o *F1 - score* da SVM se confirma quanto a sua eficiência, sendo superior a RNA tanto em preve rolamento em estado de falha quanto em estado de funcionamento normal.

Figura 61 – Gráfico das Previsões da RNA e SVM com 48 horas de antecedência



Fonte: próprio autor

A Figura 61 demonstra o desempenho dos algoritmos na validação para previsão de falhas com 48 horas de antecedência. O desempenho da RNA, Figura 61 (b), novamente não foi satisfatório, pois durante grande parte da validação os valores previstos foram de um rolamento em estado normal, sendo que apenas nos últimos instantes o algoritmo conseguiu entregar previsões de um componente com estado defeituoso, porém muito tarde em relação ao proposto que seria ter uma antecedência de 48 horas, sendo possível validar isto pela Figura 61 (a) que apresenta os valores reais. A Figura 61 (c) representa o desempenho da SVM que entregou previsões extremamente satisfatórias, conseguindo entender o comportamento do defeito com horas antes da degradação total, se assemelhante bastante ao gráfico dos valores reais, podendo até mesmo mencionar que estes resultados foram alcançados com grande precisão sendo praticamente um espelho destes.

5.3. Previsões com 72 horas de Antecedência

Pela Tabela 7, a acurácia da RNA e da SVM durante o treinamento e validação para realizar previsões com 72 horas de antecedência pode ser analisada.

Tabela 7 – Acurácia da RNA e SVM com 72 horas de antecedência

	Acurácia no Treinamento (%)	Acurácia na Validação (%)
RNA	49,68	62,5
SVM	77,51	84,04

Fonte: próprio autor

Durante o treinamento e validação conforme a Tabela 7, a SVM apresentou acurácia melhor durante o treinamento e a validação novamente, sendo perceptível notar que até este momento a RNA não obteve bons resultados durante as etapas de aprendizagem e conseqüentemente sua validação será coerente com o processo de entendimento do comportamento dos dados.

Tabela 8 – Matriz de Confusão da RNA e SVM com 72 horas de antecedência

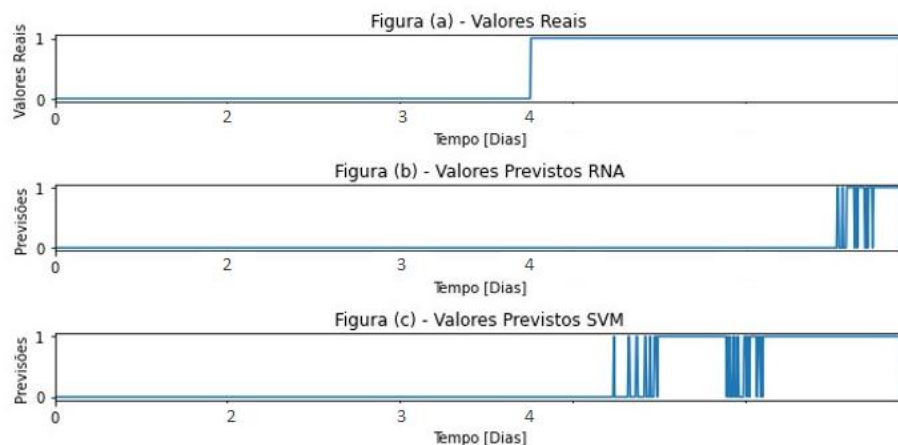
	RNA				
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras	
Rolamento com Defeito (1)	100	18	31	432	80 (VP) 352 (FP)
Rolamento em Estado Normal (0)	61	100	76	552	0 (FN) 552 (VN)

	SVM				
	Precisão (%)	Recall (%)	F1 - Score (%)	Total de Amostras	
Rolamento com Defeito (1)	100	66	79	432	284 (VP) 148 (FP)
Rolamento em Estado Normal (0)	79	100	88	552	0 (FN) 552 (VN)

Fonte: próprio autor

Conforme Tabela 8, novamente a análise deve levar em conta o critério de penalizar o Falso Negativo, dos 552 dados de rolamento em estado normal os dois algoritmos alcançaram 100% de acerto em realizar esta previsão, enquanto que para os 432 dados do rolamento em estado defeituoso a RNA acertou 18% deles e a SVM obteve novamente um melhor desempenho acertando 66% destes. O $F1 - score$ foi superior na SVM, novamente comprovando seu desempenho superior ao verificar por outra métrica.

Figura 62 – Gráfico das Previsões da RNA e SVM com 72 horas de antecedência



Fonte: próprio autor

Os valores reais para a validação de previsão de falhas com 72 horas de antecedência pode ser avaliada pela Figura 62 (a), quanto mais antecedente for para prever uma falha mais difícil se torna para os algoritmos obterem bons desempenhos, pois a dificuldade aumenta. A RNA, Figura 62 (b), na medida que a solicitação para se prever as falhas mais cedo foi requerida, começou a detectar um pouco mais de sinais de um rolamento defeituoso em comparação com as duas últimas previsões de falhas (24 horas e 48 horas de antecedência), pois nos últimos instantes conseguiu entregar previsões corretas, porém tarde para o que proposto. Enquanto que a SVM, pela Figura 62 (c), demonstrou resultados semelhantes ao último experimento (48 horas de antecedência), obtendo um desempenho ainda muito melhor do que a RNA, sendo um indício de que o seu entendimento se limitou a previsão com 48 horas de antecedência.

5.4. Discussão dos Resultados

Mediante a proposta deste trabalho em realizar previsões com 24, 48 e 72 horas de antecedência da falha completa dos rolamentos, a grande tarefa não foi ajustar os melhores parâmetros da SVM, mas sim ajustar a RNA o que apresentou os piores resultados embora

durante toda a trajetória deste trabalho inúmeras técnicas foram propostas e implementadas com intuito de obter melhores resultados. Sendo possível analisar novamente o desempenho dos algoritmos a nível comparativo, percebendo que a SVM entregou os melhores resultados, Tabela 9.

Tabela 9 – Comparação do Desempenho entre a RNA e SVM

	RNA - 24h		RNA - 48h		RNA - 72h	
	Recall (%)	F1 - score (%)	Recall (%)	F1 - score (%)	Recall (%)	F1 - score (%)
Rolamento com Defeito (1)	0	0	5	9	18	31
Rolamento em Estado Normal (0)	100	92	100	84	100	76

	SVM - 24h		SVM - 48h		SVM - 72h	
	Recall (%)	F1 - score (%)	Recall (%)	F1 - score (%)	Recall (%)	F1 - score (%)
Rolamento com Defeito (1)	99	75	90	93	66	79
Rolamento em Estado Normal (0)	89	94	99	97	100	88

Fonte: próprio autor

Os sinais brutos dos coeficientes de *Wavelet* foram entregues para a RNA, técnicas para reduzir a quantidade de coeficientes mantendo o maior detalhamento possível utilizando funções específicas para redimensionar os dados, a saída também foi modificada com intuito de realizar previsões do tempo para o final do experimento, a modificação de várias escalas de *Wavelet* diferentes foram implementadas para comprimir e esticar os dados com objetivo de obter resultados diferentes, diversas funções de ativação foram experimentadas na RNA, porém tudo que foi feito não teve a capacidade de ajudar o algoritmo a alcançar seu melhor desempenho. Os resultados obtidos foram apenas o reflexo da sua ineficiência de entender as informações contidas nos dados, da forma que foi apresentada para a mesma.

6. Conclusão

Conforme exposto durante todo este trabalho que consistiu em analisar um banco de dados público com valores reais provenientes de um experimento, foi possível aplicar técnicas e métodos quantitativos com intuito de alcançar resultados satisfatórios visando prever os estados de rolamentos mecânicos. O que gerou uma dedicação extensa devido a aplicação de diversas metodologias visando conseguir os melhores resultados dos algoritmos, gerando até mesmo uma certa expectativa quanto ao desempenho da RNA o que se mostrou um pouco diferente daquilo que era esperado sendo que a SVM se adequou melhor a esta tarefa.

Neste estudo de caso os resultados foram muito satisfatórios utilizando a SVM, que entregou boas previsões com antecedência de até 48 horas antes da falha total, sem deixar de comprometer a qualidade das métricas gerando uma excelente precisão. A RNA não se adequou a este trabalho devido ao fato de termos poucos dados de rolamentos distintos e o mais importante dados extremamente limitados de rolamentos com defeito, o que fez com que o desempenho deste algoritmo tenha sido comprometido. O comportamento de falha ou defeito não foi totalmente entendido pela RNA, porque esta necessita de um volume grande de dados para alcançar a sua melhor performance, diferentemente da SVM que não apresenta limitação para se entender um problema tendo pouca quantidade de dados, sendo que é apenas alterado o espaço e criado o melhor hiperplano de separação, independente se existe muito ou poucos dados de um rolamento com defeito.

Por meio destes algoritmos o aprendizado de máquina pôde ser aplicado e até mesmo de maneira positiva sendo eficiente em um cenário real, na qual se torna eficaz ter um programa que tem a inteligência computacional aliada aos modelos estatísticos capaz de monitorar em tempo real os estados dos componentes mecânicos, julgando quando solicitado se uma próxima falha/defeito está se desenvolvendo ou não. Amparado por precisões baseadas em históricos passados que aprendem o comportamento dos eventos desde o início de sua formação até o final.

Para futuros trabalhos o desejo de melhorar a RNA se tornou ardente, pelas suas qualidades e vantagens expostas em diversos trabalhos estudados, sendo necessário para isto obter mais dados/informações de diversos rolamentos distintos em estado normal de operação e o mais importante a maior quantidade possível de dados de inúmeros rolamentos no estado específicos de defeito/falha, pois novamente a RNA é um algoritmo que necessita de um grande volume de informações para se entender um problema.

REFERÊNCIAS

- ALMEIDA, Fabiano Ribeiro do Vale. **Análises Estatísticas e Reconhecimento de Padrão Aplicados em Diagnósticos de Defeitos em Rolamentos Através da Análise de Vibração** / Fabiano Ribeiro do Vale Almeida – Itajubá, (MG): UNIFEI, 2007.
- AZEVEDO-MARQUES, P.M., Mecantini A., Salmeri M., Rangayyan R.M. (editors). **Medical Image Analysis and Informatics: Computer-Aided Diagnosis and Therapy**. CRC Press, 518 pages, 1st Edition, 2017.
- BEN-HUR, A.; WESTON, J. **A user's guide to support vector machines**. v. 609, November 2008.
- BUDYNAS, R. G.; NISBETT, J. K. **Elementos de Máquinas de Shigley: Projeto de Engenharia Mecânica**. 8ª ed. Porto Alegre: AMGH, 2011. 1084 p.
- BURGES, C. J. **A tutorial on support vector machines for pattern recognition**. *Data Mining and Knowledge Discovery*, pages 243–245, 1998.
- CAMPOS, A. N. **Avaliação Metrológica de um sistema de calibração de Acelerômetros por Choque Mecânico**. Tese (doutorado) - Instituto Tecnológico de Aeronáutica. São José dos Campos - SP, 2010. 104f.
- CANALLI, Ygor de Mello. **Funções de Ativação Hiperbólicas em redes neurais**/Ygor de Mello Canalli. – Rio de Janeiro: UFRJ/COPPE, 2017.
- DAUBECHIES, I., “**Ten Lectures on Wavelets**”. CBMS-NSF regional conference series in applied mathematics, Society for Industrial and Applied Mathematics, Philadelphia, 1992.
- DINIZ, Paulo S. R. **Processamento digital de sinais** [recurso eletrônico]: projeto e análise de sistemas / Paulo S. R. Diniz, Eduardo A. B. da Silva, Sergio L. Netto ; tradução: Luiz Wagner Pereira Biscainho. – 2. ed. – Dados eletrônicos. – Porto Alegre: Bookman, 2014.
- DUDA, R. O.; HART, P. E.; STORK, D. G.: **Pattern Classification**. 2º Edição. Wiley, 2001.
- FILGUEIRAS, Paulo Roberto, 1982 - **Regressão por vetores de suporte aplicado na determinação de propriedades físico-químicas de petróleo e biocombustível** / Paulo Roberto Filgueiras. Campinas, SP: [s.n]. 2014.
- FINOCCHIO, Marco Antônio Ferreira. **Noções De Redes Neurais Artificiais**. Julho, 2014. 29 p.
- FURTADO, Maria Inês Vasconcellos. **Redes neurais artificiais** [recurso eletrônico]: Uma abordagem para sala de aula / Maria Inês Vasconcellos Furtado. – Ponta Grossa (PR): Atena Editora, 2019.
- GONÇALVES, A. R. **Máquina de vetores suporte**. Disponível em: <http://www.dca.fee.unicamp.br/~andreric/arquivos/pdfs/svm.pdf>, 2010. Acesso em: 15 de set. de 2020.

- GONZALEZ, C. **“What’s the Difference Between Bearings?”**, 2015. Disponível em: <https://cdn.baseplatform.io/files/base/ebm/machinedesign/document/2019/03/machinedesign_3351_differencebetweenbearings.pdf>. Acesso em: 03/10/2020.
- GRAPS, A. **"An introduction to wavelets"** in IEEE Computational Science and Engineering, vol. 2, no. 2, pp. 50-61, Summer 1995.
- GUNN S. R. **Support Vector Machine for Classification and Regression**. Faculty of Engineering, Science and Mathematics School of Electronics and Computer science, 1998.
- HAAR, A., **Zur Theorie der orthogonalen Funktionensysteme**, Tese de doutorado, Georg-August-Universität Göttingen, 1909.
- HALLIDAY, D. RESNICK, R. WALKER, J. **Fundamentos de Física**. Volume 2: Gravitação, Ondas e Termodinâmica. Rio de Janeiro: LTC, 2012.
- HAYKIN, S. **Redes Neurais: Princípios e Prática**. Porto Alegre: Bookman, 2001.
- HAYKIN, S.; VEEN, B. V. **Sinais e Sistemas**. Porto Alegre: Bookman, 2001.
- HAYKIN, Simon. **Redes Neurais: Princípios e Práticas**. 2 ed, 900p., 2007.
- HEARST, M. A., B. Schölkopf, S. Dumais, E. Osuna, and J. Platt. **Trends and controversies - support vector machines**. IEEE Intelligent Systems, 13(4):18–28, 1998.
- HEBB, D. H. 1949. **The Organization of the Behavior**. New York :John Wiley & Sons, 1949.
- HOLZMANN, A. H. **Técnicas de processamento de sinais e telecomunicações** [recurso eletrônico]– Ponta Grossa, PR: Atena Editora, 2019.
- KARDEC, Alan; NASCIF, Júlio. **Manutenção: Função Estratégica**. 3.ed. Rio de Janeiro: Qualitymark: Petrobras, 2009.
- KINGMA, D. P.; BA, J. **Adam: A method for stochastic optimization**. CoRR, abs/1412.6980, 2014.
- LÓPEZ, R. Z. **Classificação automática de defeitos em máquinas rotativas**. 2014. 120 f. TCC (Graduação) - Curso de Engenharia Eletrônica e de Computação, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2014.
- MARCONI, Mariana de A; LAKATOS, Eva M. **Fundamentos de metodologia científica**. 5° ed. São Paulo: Editora Atlas, 2006. 315 p.
- MATHIAS, Mauro. **Ferramentas de Diagnostico de Máquinas**. Disponível em: <http://acervodigital.unesp.br/handle/123456789/46460>, 2010. Acesso em: 10 maio de 2021.
- MATSUO, T. K. **Método de análise de vibrações para identificação de falhas em máquinas rotativas de geração de energia elétrica**. 2017. 161 f. Dissertação (Mestrado) - Curso de Mecatrônica, Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina,

Florianópolis, 2017.

MCCULLOCH, W. S. e Pitts, W. H. 1943. **A Logical Calculus of the Ideas Immanent in Nervous Activity**. Bulletin of Mathematical Biophysics. 1943, Vol. 5, pp. 115 - 133.

MELCONIAN, Sarkis. **Elementos de Máquinas**. 9. Ed. Érica, 2009.

MENDEL, J. M. & R. W. McLaren. **Reinforcement-learning control and pattern recognition systems**. pp. 287–318, 1994.

MISITI, M., Misiti Y., Oppenheim, G., Poggi, J-M., **Wavelet Toolbox™ User's Guide, The MathWorks**, Inc. versão 1.0 on line. Disponível em: www.mathworks.com. Acesso em: 08 de outubro de 2020, The MathWorks, Inc, USA, 1996.

MITCHELL, T. M.. **Machine Learning**. McGraw-Hill, New York, 1997.

MOUBRAY, John. RCM – **Reliability Centered Maintenance**. Ed. SQL Brasil – 2.ed. 1999.

NIELSEN, O. M. **Wavelets in scientific computing**. PhD thesis, Technical University of Denmark, 1998.

NOVAKOVIC, J.; VELJOVIC, A. **C-support vector classification: Selection of kernel and parameters in medical diagnosis**. In: Intelligent Systems and Informatics (SISY), 2011 IEEE 9th International Symposium on. [S.l.: s.n.], 2011, p. 465–470.

OLIVEIRA, H. M.. **Análise de Sinais para Engenheiros: Uma Abordagem via Wavelets**. Recife: Brasport, 2007.

PASSERINI, A. **Kernel Methods, multiclass classification and applications to computational molecular biology**. PhD thesis, Università Degli Studi di Firenze, 2004.

KONAR, PRATYAY & CHATTOPADHYAY, PAAMITA. (2011). **Bearing fault detection of induction motor using wavelet and Support Vector Machines (SVMs)**. Appl. Soft Comput.. 11. 4203-4211. 10.1016/j.asoc.2011.03.014.

PORTNOFF, M. **Time Frequency representation of digital signal and systems based on short-time Fourier analysis** Acoustics, Speech, and Signal, IEEE Transactions on, Volume: 28 Issue: 1, Feb, 1980.

PY, O., Hong, H., and Zhongzhi, S. **“Plankton classification with deep convolutional neural networks”**, in Proceedings of the 2016 IEEE Information Technology, Networking, Electronic and Automation Control Conference, Chongqing: IEEE, 132–136, 2016.

QIU, H., Lee, J., Lin, J., and Yu, G., **“Wavelet filter-based weak signature detection method and its application on rolling element bearing prognostics”**, *Journal of Sound Vibration*, vol. 289, no. 4–5. pp. 1066–1090, 2006.

RAO, B. K. N., Pai, P. S., Nagabhushana, T. N., (2012). “**Failure Diagnosis and Prognosis of Rolling-Element Bearings using Artificial Neural Networks: A Critical Overview**”. *Journal of Physics: Conference Serie*, vol. 364, pp. 1-28.

REBELO, L. D. T., **Avaliação automática do resultado estético do tratamento conservador do cancro de mama**. Faculdade de Engenharia da Universidade do Porto, 2008.

RICCIOTTI, Antonio Carlos Duarte. **Utilização de Wavelets no processamento de sinais EMG**/Antonio Carlos Duarte Ricciotti. – 2006.

RUDIN, Cynthia & Wagstaff, Kiri. **Machine learning for science and society**. *Machine Learning*, 95. 1-9. 10.1007/s10994-013-5425-9, 2014.

RUSSEL, S. J.; NORVING, P.: **Artificial Intelligence: A Modern Approach**. 2º Edição. Prentice-Hall, 2002.

SALIBA, Tuffi Messias. **Manual prático de avaliação e controle de vibração : PPRA / Tuffi Messias Saliba ; colaboradoras Márcia Angelin Chaves Corrêa, Maria Beatriz de Freitas Lanza. — 6. ed. — São Paulo : LTr, 2019.**

SAMPAIO, Rubens. **Análise e Processamento de Sinais** - São Carlos, SP: SBMAC, 2012, 130 p. - 20.5 cm- (Notas em Matemática Aplicada; v. 21).

SANTOS, Eulanda Miranda. **Teoria e Aplicação de Support Vector Machines à Aprendizagem e Reconhecimento de Objetos baseados na Aparência**. 2002. 121 p. Dissertação (Mestrado em Informática) – Universidade Federal da Paraíba. Campina Grande, Paraíba, 2002.

SASSI, S. 7 . “**A Numerical Model to Predict Damaged Bearing**”, *Journal of Vibration and Control*, vol. 13, nº 11, pp. 1603–1628.

SEQUEIRA, Cláudia D. **Sensores para Medições de Vibrações Mecânica - Acelerômetros**. *Revista Manutenção* ed. 116. Brasil, 2013.

SOEIRO, N. S. **Curso de Fundamentos de Vibrações e Balanceamento de Rotores**. Notas de aula. Universidade Federal do Pará. Pará, PA, Brasil, 2008.

SRIVASTAVA, N. et al. **Dropout: A simple way to prevent neural networks from overfitting**. *Journal of Machine Learning Research*, v. 15, p. 1929–1958, 2014.

STOLLNITZ, E., DeRose, T. & Salesin, D. **Wavelets for computer graphics: A primer part 1**. *IEEE Computer graphics and applications*, 15(3), 76-84, 1995.

SUTTON, R. & A. Barto. **Reinforcement Learning: An Introduction**. Cambridge, MA: MIT Press, 1998.

TOBON-MEIJA, Diego, Kamal Medjaher, Noureddine Zerhouni, Gérard Tripot. **A mixture of gaussians hidden markov model for failure diagnostic and prognostic..** 6th Annual IEEE Conference on Automation Science and Engineering, CASE'10., Aug 2010, Toronto, Canada. pp.338-343.

TOBON-MEIJA, Diego & Medjaher, Kamal & Zerhouni, Noureddine & Tripot, Gerard. (2012). **A Data-Driven Failure Prognostics Method Based on Mixture of Gaussians Hidden Markov Models.** IEEE Transactions on Reliability - TR. 61.

VAPNIK, V. N. **The nature of statistical learning theory.** New York, NY, USA: Springer-Verlag New York, Inc., 1995.

VERMEHREN, V. & de Oliveira, Hélio. **Close expressions for Meyer Wavelet and Scale Function,** 2015.

VIDHYA, M. **Efficient classification of port scan attacks using support vector machine.** In: Green High Performance Computing (ICGHPC), 2013 IEEE International Conference on. [S.l.: s.n.], 2013. p. 1–5.

YIN, R. K. **Estudo de caso: planejamento e métodos.** 2.ed. Porto Alegre: Bookman, 2001.

APÊNDICE A – Algoritmos de Aprendizado de Máquina

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import statsmodels
import statsmodels.api as sm
from google.colab import drive
from scipy import fft, ifft
import pywt
import zipfile
from numpy import mean, sqrt, square, arange
from scipy.stats import kurtosis
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix, accuracy_score
from scipy import signal

drive.mount('/content/drive', force_remount=True)
```

```
X_treino = pd.concat([df1_final, df3_final])
Y_treino = X_treino['Tempo']
X_treino.drop(['Tempo'], axis=1, inplace=True)

X_teste = df2
Y_teste = df2['Tempo']
X_teste.drop(['Tempo'], axis=1, inplace=True)

X_train, aaa, Y_train, aaa = train_test_split(X_treino, Y_treino, test_size=0.000001, random_state=42)
print(X_treino.shape, Y_treino.shape)
print(X_teste.shape, Y_teste.shape)
```

```
import keras
from keras.models import Sequential
from keras.layers import Dense, Dropout
from sklearn import metrics
from keras.regularizers import l2
from sklearn.metrics import confusion_matrix
from sklearn.metrics import classification_report
```

```
lbd = 1e-4
p_dropout = 0.5

Classificador = Sequential()
Classificador.add(Dense(units = 5, activation = 'relu', input_dim = 4, kernel_regularizer=l2(lbd)))
Classificador.add(Dropout(p_dropout))
Classificador.add(Dense(units = 8, activation = 'relu', kernel_regularizer=l2(lbd)))
Classificador.add(Dropout(p_dropout))
Classificador.add(Dense(units = 6, activation = 'relu', kernel_regularizer=l2(lbd)))
Classificador.add(Dropout(p_dropout))
Classificador.add(Dense(units = 8, activation = 'relu', kernel_regularizer=l2(lbd)))
Classificador.add(Dropout(p_dropout))
Classificador.add(Dense(units = 1, activation = 'sigmoid', kernel_regularizer=l2(lbd)))

Classificador.compile(optimizer='adam', loss='binary_crossentropy', metrics=['accuracy'])
sample_weight = np.ones(shape=(len(Y_treino),))
sample_weight[Y_treino > .5] = 5
model = Classificador.fit(X_treino, Y_treino.values, batch_size = 100, epochs = 500, sample_weight=sample_weight)
```

```
from sklearn import svm
Classificador2 = svm.SVC(kernel='rbf', C=1e5, gamma=1e-6)
Classificador2.fit(X_treino, Y_treino)
```