

UNIVERSIDADE DE TAUBATÉ
CARLOS HENRIQUE DOMICIANO

**“REDES NEURAS ARTIFICIAIS APLICADA À TRANSMISSÃO AUTOMÁTICA
ADAPTATIVA”.**

Taubaté
2023

CARLOS HENRIQUE DOMICIANO

**“REDES NEURAS ARTIFICIAIS APLICADA À TRANSMISSÃO
AUTOMÁTICA ADAPTATIVA”.**

Dissertação apresentada para obtenção do título de Mestre pelo curso de Engenharia Mecânica do Departamento de Engenharia Mecânica da Universidade de Taubaté.

Área de Concentração: Automação

Orientador: Prof. Dr. Luis Fernando de Almeida

**Taubaté
2023**

Grupo Especial de Tratamento da Informação - GETI
Sistema Integrado de Bibliotecas – SIBi
Universidade de Taubaté - Unitau

D669r Domiciano, Carlos Henrique
Redes neurais artificiais aplicada à transmissão automática adaptativa /
Carlos Henrique Domiciano. -- 2024.
88 f. : il.

Dissertação (mestrado) – Universidade de Taubaté, Pró-reitoria de
Pesquisa e Pós-graduação, Taubaté, 2024.

Orientação: Prof. Dr. Luis Fernando de Almeida, Departamento de
Engenharia Mecânica.

1. Redes neurais artificiais. 2. Transmissão automática. 3. Mineração
de dados. 4. Machine Learning. I. Universidade de Taubaté. Departamento
de Engenharia Mecânica. Mestrado em Engenharia Mecânica. II. Título.

CDD – 658.5

CARLOS HENRIQUE DOMICIANO
REDES NEURAIAS ARTIFICIAIS APLICADA À TRANSMISSÃO
AUTOMÁTICA ADAPTATIVA

Dissertação apresentada para obtenção do título de Mestre pelo curso de Engenharia Mecânica do Departamento de Engenharia Mecânica da Universidade de Taubaté.

Área de Concentração: Automação

Orientador: Prof. Dr. Luis Fernando de Almeida

Data: 01/03/2024

Resultado: Aprovado

BANCA EXAMINADORA

Prof. Dr. Luis Fernando de Almeida

Universidade de Taubaté

Assinatura:



Prof. Dr. Arcione Ferreira Viagi

Universidade de Taubaté

Assinatura: _____

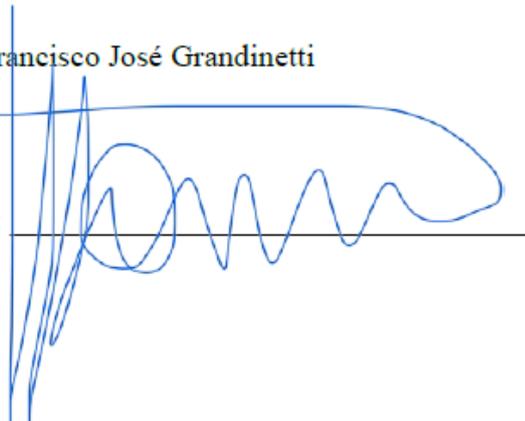


Prof. Dr. Francisco José Grandinetti

Universidade Estadual

Paulista

Assinatura: _____



DEDICATÓRIA

Dedico primeiramente a Deus, a minha família que tanto admiro. Enfim a todos que contribuíram para realização desse trabalho, ofereço o resultado do esforço realizado ao longo deste percurso.

AGRADECIMENTOS

Primeiramente, agradeço ao professor Luis Fernando de Almeida pela excepcional orientação ao longo do desenvolvimento do trabalho.

Agradeço aos professores Evandro Luis Nohara e Francisco José Grandinetti pelo excelente suporte concedido oferecidos para a conclusão deste trabalho.

Aos meus pais pelo apoio incondicional oferecido.

À minha esposa Renata Fernandes de Souza e ao meu filho Daniel Fernandes Domiciano pelo apoio e compreensão ao longo do desenvolvimento deste trabalho.

À UNITAU – Departamento de Engenharia Mecânica, pela oportunidade de aprimorar meus conhecimentos neste tema que acredito ser de extrema importância e relevância para a sociedade.

EPÍGRAFE

*“Crê em ti mesmo, age e verá os resultados.
Quando te esforças, a vida também se esforça
para te ajudar.”*

(Chico Xavier)

RESUMO

Os padrões de mudança fixo, apresentado pelas transmissões automáticas convencionais pode gerar um certo aborrecimento, principalmente nos aclives e declives, expõe certas limitações, desconforto e influenciando no aumento do consumo de combustível. A proposta desta pesquisa é apresentar um estudo sobre o aprendizado de máquina através das Redes Neurais Artificiais, proporcionando o aperfeiçoamento do Sistema de Transmissão Automática, para melhoria da dirigibilidade do veículo e economia de combustível. Este estudo baseia-se na compreensão técnica de *Machine Learning*, sendo este um domínio bastante amplo e complexo, que figura entre as áreas mais importantes da Inteligência Artificial. São aplicados os dados característicos do motor, oriundos da unidade de controle, a qual são utilizados para o aprendizado de máquina, representados através de um modelo gerado a partir das Redes Neurais Artificiais, que podem ser utilizadas em um algoritmo de classificação, o qual irá prever o modo de condução do motorista, identificar o estilo de estrada transitada ou o tipo de carga aplicada ao veículo. Esse modelo deve evoluir através do avanço do aprendizado e esse aprendizado deve ser medido, para verificar se o modelo é capaz de entender o quanto a máquina aprendeu. A aquisição de dados foi realizada através do aplicativo TORQUE, junto a um módulo de extensão denominado *Realtime Charts* que através de Mini scanner OBD2 com comunicação Bluetooth, permite acessar as diferentes aplicações do veículo. Esses dados são aplicados no software *PyCharm* que através da biblioteca gratuita *Scikit-learn*, de código aberto em *Python*, oferece algumas opções de recursos capazes de realizar uma modelagem estatística, diagnóstico e mineração de dados, ofertando suporte ao aprendizado supervisionado e não supervisionado. Entre outras conclusões, os modelos criados com os dados para aprimorar a dirigibilidade e a economia através de um controle adaptativo da transmissão automática, alcançou um prognóstico desejável no treinamento e nos testes, apresentando valores que atingem até 97% de acurácia durante a classificação correta das instâncias, para a etapa de treinamento e 86% de acurácia para a etapa de testes, indicando o quanto o modelo aprendeu, validando os acertos do algoritmo.

Palavras-chave: Redes Neurais Artificiais. Transmissão Automática. Mineração de dados. *Machine Learning*.

ABSTRACT

The fixed shift patterns presented by conventional automatic transmissions may generate a certain amount of annoyance, especially on incline and descent, presenting certain limitations, discomfort and influencing an increase in fuel consumption. The intention of this research is to present a study on machine learning through Artificial Neural Networks, providing the improvement of the Automatic Transmission System, to upgrade vehicle drivability and fuel economy. This study is based on the technical understanding of Machine Learning, which is a very wide and complex domain, who is among the most important areas of Artificial Intelligence. The characteristic data of the engine are applied, coming from the control unit, which are used for machine learning, represented through a model generated from Artificial Neural Networks, who can be used in a classification algorithm, that one will predict the driver's driving style, identify the style of road used or the type of load applied to the vehicle. This model must develop through the advancement of learning and this learning must be measured to verify whether the model can understand how much the machine has learned. Data acquisition was carried out using the TORQUE application, in conjunction with an extension module called Realtime Charts which, using a Mini OBD2 scanner with Bluetooth communication, allows access to the vehicle's different applications. This data is applied to the PyCharm software, which, through the free Scikit-learn library, open source in Python, offers some resource options capable of performing statistical modeling, diagnosis and data mining, offering support for supervised and unsupervised learning. Among other conclusions, the models created with the data to improve drivability and economy through adaptive control of the automatic transmission, achieved a desirable prognosis in training and testing, presenting values that reach up to 97% accuracy during the correct classification of instances, for the training stage and 86% accuracy for the testing stage, indicating how much the model learned, validating the algorithm's successes.

Keywords: Artificial Neural Networks. Automatic transmission. Data mining. Machine Learning.

LISTA DE ABREVIATURAS

ABS	<i>Antilock Braking System.</i>
AED	A análise exploratória de dados
BSD	<i>Berkeley Software Distribution</i>
CAN	<i>Controller Area Network</i>
CV	Coefficiente de variação
DLC	<i>Diagnostic Link Connector</i>
DTC	<i>Decision Tree Classifier</i>
ECU	<i>Engine Control Unit</i>
GNU	<i>General Public License</i>
IA	Inteligência Artificial
IDE	<i>integrated development environment</i>
KDD	<i>Knowledge-Discovery in Databases</i>
KNN	<i>k-Nearest Neighbors</i>
ML	<i>Machine Learning</i>
MLP	<i>Multilayer Perceptron</i>
OBD2	<i>On Board Diagnostics</i>
PSO	<i>Particle Swarming Optimization</i>
RNA	Redes Neurais Artificiais
ROC	<i>Receiver Operating Characteristic</i>
SVM	<i>Support Vector Machine</i>
TCU	<i>Transmission Control Unit</i>
TI	Tecnologia da Informação

LISTA DE FIGURAS

Figura 2.1 - Conversor de torque.....	23
Figura 2.2 - Conjunto Planetário Simples.	24
Figura 2.3 - Relação de transmissão de um conjunto de planetária	24
Figura 2.4 - Corpo de válvulas	25
Figura 2.5 - Eletroválvulas	25
Figura 2.6 - Padrões de mudança.....	26
Figura 2.7 - Modelo de controle baseado em RNA.....	27
Figura 2.8 - Esquema de ligação do sensor MAP/ACT.	28
Figura 2.9 - Sensor de aceleração Hall.....	29
Figura 2.10 - Eletroválvula.....	30
Figura 2.11 - Padrões de passagem ascendentes e descendentes.	31
Figura 2.15 - Aprendizado de Máquina.....	34
Figura 2.16 - Tipos de métodos de aprendizado indutivo	35
Figura 2.17 - Treinamento e teste do modelo de ML	36
Figura 2.18 - Cálculo da similaridade entre padrões no KNN usando a distância euclidiana.....	39
Figura 2.19 - Árvore de decisão	39
Figura 2.20 - <i>Support Vector Machine</i>	40
Figura 2.21 - Rede Neural Artificial.....	41
Figura 2.22 - Curva ROC	42
Figura 2.23 - Etapas do processo KDD.	45
Figura 2.24 - Registro de agrupamento.	48
Figura 2.25 - Divisão do conjunto de treinamento em folds para um <i>K Fold-Cross-Validation</i> com $K = 10$	49
Figura 2.26 - Modelo de um neurônio não linear	51
Figura 2.27 - Função limite	52
Figura 2.28 - Função linear por partes.....	53
Figura 2.29 - Função Sigmóide	54
Figura 2.30 - <i>Rede feedforward</i> com uma única camada de neurônios.	55
Figura 2.30 - <i>Rede feedforward</i> com uma camada oculta e uma camada de saída.....	56
Figura 2.31 - Rede MLP com duas camadas ocultas.....	57
Figura 2.33 - Fluxo básico de direção do sinal.....	57

Figura 2.34 - Apresentação dos modelos com <i>underfitting</i> , ajuste genérico e <i>overfitting</i> .	60
Figura 3.1 - <i>Scanner bluetooth</i> com conector OBD2.	61
Figura 3.2 - Aplicativo Torque.	62
Figura 3.3 - Dados obtidos por meio do aplicativo Torque.	63
Figura 3.4 - Conversor .CSV para .XLSX.	66
Figura 3.5 - Regras de passagens adaptativas.	67
Figura 3.6 - Ambiente de desenvolvimento integrado do <i>PyCharm</i> .	68
Figura 3.7 - Alteração dos padrões de passagem.	70
Figura 3.7 - Modelo utilizado – Honda Fit 2014.	71
Figura 3.8 - Trajeto urbano aplicado	72
Figura 3.8 - Trajeto rodoviário aplicado.	73
Figura 4.1 - Verificação dos dados.	74
Figura 4.2 - Representação do modelo de rede com 175 neurônios na camada oculta.	76
Figura 4.3 - Representação do modelo de rede com 125 neurônios na camada oculta.	77
Figura 4.4 - Indicação da média e do desvio padrão da acurácia.	78
Figura 4.5 - Indicação da acurácia.	78
Figura 4.6 - Indicação da precisão.	78
Figura 4.7 - Indicação da precisão.	79
Figura 4.8 - Indicação de especificidade	80
Figura 4.9 - Indicação da média harmônica.	81

LISTA DE QUADROS

Quadro 3.1 - Relação de atributos adquiridos via scanner <i>bluetooth</i>	64
Quadro 4.1 - Matriz de confusão para treino e testes	76

LISTA DE TABELAS

Tabela 2.1 - Elementos da Matriz de Confusão.....	41
Tabela 3.1 - Prioridades de utilização das regras de funcionamento.....	70
Tabela 3.2 - Ficha técnica Honda Fit.....	71
Tabela 4.1 - Dados de treino e teste.....	74
Tabela 4.2 - Propriedades de controle de treinamento do modelo de ML.....	75
Tabela 4.3 - Resultado de acurácia encontrados.....	77
Tabela 4.4 - Resultado de precisão encontrados.....	78
Tabela 4.5 - Resultado de sensibilidade encontrados.....	79
Tabela 4.6 - Resultado de especificidade encontrados.....	80
Tabela 4.7 - Resultado do F1 Score encontrados	80

SUMÁRIO

1 INTRODUÇÃO	17
1.1 ESTADO DA ARTE	18
1.2 OBJETIVOS	20
1.2.1 Objetivos Gerais	20
1.2.2 Objetivos Específicos	20
1.3 ORGANIZAÇÃO DO TRABALHO	21
2 FUNDAMENTAÇÃO TEÓRICA	22
2.1 CÂMBIO AUTOMÁTICO	22
2.1.1 Estrutura básica de uma transmissão automática	23
2.2 SISTEMAS DE CONTROLE	26
2.2.1 Estratégias de controles tradicionais	26
2.2.2 Estrutura do sistema de controle	27
2.2.3 Métodos de controle	30
2.2.4 Redes de comunicação nos veículos automotivos e processos de aquisição de dados	32
2.3 SISTEMAS INTELIGENTES	33
2.3.1 Inteligência Artificial	33
2.3.2 Aprendizado de máquina	33
2.3.3 Algoritmos de aprendizado de máquina	37
2.3.4 Parâmetros de avaliação de desempenho dos algoritmos	41
2.4 DESCOBERTA DO CONHECIMENTO	43
2.4.1 Definição	43
2.4.2 Compreensão do desenvolvimento do KDD	43
2.4.3 O processo KDD	44
2.4.4 Análise exploratória dos dados	45
2.4.5 Mineração de dados	46
2.4.6 Tarefas de mineração de dados	47
2.4.7 Interpretação e Avaliação	48
2.5 REDES NEURAIS ARTIFICIAIS	50
2.5.1 Introdução a Rede Neural Artificial	50
2.5.2 Neurônio	50
2.5.3 Função de ativação	52

2.5.3.1 Função limite	52
4.5.3.2 Função linear por partes	53
4.5.3.3 Função Sigmoide	53
4.5.3.4 Função ReLU.....	54
2.5.4 Arquitetura de rede	55
2.5.5 Multilayer Perceptron	56
2.5.6 Algoritmo de etropropagação.....	58
2.5.7 Hiperparâmetros de rede	59
3 MATERIAIS E MÉTODOS	61
3.1 VERIFICAÇÃO DO CONJUNTO DE DADOS	61
3.1.1 Dispositivo de coleta de dados	61
3.1.2 Acesso aos dados veicular	62
3.1.3 Obtenção dos dados	62
3.1.4 Alinhamento dos dados	65
3.1.5 Transição dos dados	65
3.1.6 Definição dos atributos classe.....	66
3.1.7 Execução do Algoritmo	67
3.1.8 Rede Neural Artificial Utilizada.....	69
3.1.9 Aplicação do método de controle	69
3.1.9 O automóvel utilizado	71
3.1.10 O trajeto.....	72
4 TESTES E RESULTADOS	73
4.1 MÉTODOS DE AVALIAÇÃO E DESEMPENHO	73
4.1.1 Separação dos dados.....	74
4.1.2 Modelos.....	75
4.1.3 Matriz de confusão	76
4.1.4 Métricas de avaliação	77
4.1.4.1 Acurácia.....	77
4.1.4.2 Precisão.....	78
4.1.4.3 Sensibilidade	79
4.1.4.4 Especificidade	79
4.1.4.5 F1 Score.....	80
5 CONCLUSÃO.....	81
REFERÊNCIAS.....	85

1 INTRODUÇÃO

A procura por conforto no trânsito, associado a possibilidade de veículos com custo mais acessível, tem motivado a busca por carros com transmissão automática. No início as propostas empregadas na transmissão automática tinham inúmeras limitações de comando, influenciando o consumo de combustível e as emissões que são diretamente ligadas à tecnologia dos veículos.

Esses projetos funcionavam com dados insuficientes para definir qual relação de transmissão aplicar, sendo a seleção da marcha determinada com foco nos padrões de mudança de velocidade do veículo e posição de borboleta do motor, isto é, pedal do acelerador, utilizando padrões de mudanças inconvenientes, especialmente em aclave e nos declives. Ao desacelerar em um declive, não retinham a marcha para produzir freio-motor. Em um aclave, passavam à marcha seguinte toda vez que se aliviava o pé do acelerador (FHB PERFORMANCE, 2023). Nessas condições, os motoristas precisavam utilizar a alavanca de câmbio para selecionar a posição ‘2’, por exemplo, mantendo as altas rotações e reforçando as implicações do freio – motor. Outras variáveis podem ser utilizadas para aprimorar as trocas de marcha, como a aplicação dos acelerômetros, para ajudar a *Transmission Control Unit* (TCU) a interpretar se o veículo está em um aclave ou declive, somando essas informações aos dados do pedal do acelerador e a velocidade do veículo, todavia como saber se o veículo está carregado ou não? a capacidade de acelerar e conservar a velocidade varia em função dessa variável associada ao peso.

A mudança indevida de marcha, gerado pelos padrões inconveniente, podem submeter o câmbio a um desgaste prematuro, provocando superaquecimento interno e redução da vida útil de todos os dispositivos que compõem o sistema do câmbio automático.

Ao longo dos anos houve um amplo avanço na evolução dos sistemas de transmissão, que ocorreu de forma conjunta com o emprego de algoritmos mais complexos na unidade de controle da transmissão, na evolução do hardware e no progresso dos materiais utilizados nas engrenagens e dos engrenamentos. Na década de 2000 os fabricantes começaram a adotar as transmissões “inteligentes”, que usavam algoritmos mais complexos na central eletrônica para melhorar dirigibilidade, consumo e desempenho (FHB PERFORMANCE, 2023), e o justo motivo para a constante evolução está na moderação do consumo de energia.

O avanço da *Tecnologia da Informação* (TI), definida como um conjunto de atividades e soluções processado por recursos computacionais (PUCRS Online, 2023), permitiu tratar as informações adquiridas pela *Unidade de Controle do Motor* (ECU) e classifica – lá, para

auxiliar na adaptação dos sistemas de controle. Os assim chamados “sistemas inteligentes” são aqueles que fornecem respostas que solucionam problemas, tais respostas apropriadas às situações específicas deste problema (SIMÕES et al., 2017).

Uma das áreas mais importantes da Inteligência Artificial (IA) é o *Machine Learning* (ML), que é bem amplo, e em função disso aplica muitos conceitos e técnicas qualificadas a criar um “modelo” a partir do aprendizado de máquina, gerando uma memória do que foi aprendido.

Com o propósito de otimizar as mudanças de velocidades, proporcionando uma condução mais tranquila, adequadas a vontade do motorista e as condições da estrada, é possível aplicar uma *Rede Neural Artificial* (RNA), um tipo de processamento de ML, com seu prévio “treinamento” para condução em decidas, estradas planas e aclives, aplicando estratégias de controle de mudanças adaptáveis. Os padrões fixos passaram a ser substituídos por padrões que entende a vontade do motorista e o perfil de estrada, além da carga aplicada ao veículo.

Uma RNA é um processador paralelo distribuído de forma massiva que possui a capacidade natural de armazenar conhecimento por experiência adquirida e torna – lá disponível ao uso (CAMPOS et al., 2004). Muitos modelos de RNA foram propostos para classificação de padrões. A rede *Feed Forward* multicamada com o algoritmo de aprendizado Retropropagação é um dos métodos populares.

1.1 ESTADO DA ARTE

Essa seção foi dividida em três partes. A primeira apresenta uma discussão sobre o controle e a eficiência das transmissões automáticas convencionais. Na segunda parte, é exposto modelos de algoritmos capazes de otimizar o controle das transmissões. Na terceira parte apresenta argumentações focadas no desempenho e validação dos modelos computacionais.

A redução do consumo de combustível é um objetivo dos fabricantes de automóveis. Além disso às reivindicações do consumidor (baseado em razões econômicas e ambientais) para sistemas de transmissões mais eficientes, junto dos vários regulamentos existentes, obrigam a indústria automotiva a avançar na economia de combustível (SANTICIOLLI et al., 2015).

Geralmente, as transmissões automáticas projetadas no início dos anos 2000, apresentavam sistemas de controle mais obsoletos. Um mapa fixo de mudança de marcha padronizada que normalmente indica a relação de engrenagem que deve ser engatada em função de duas entradas: velocidade do veículo e abertura da válvula borboleta. Este mapa está instalado na TCU (para transmissões automatizadas) ou no dispositivo indicador de mudança

de marcha (para transmissões manuais) em onde são obtidas as entradas e a execução ou indicação de mudança de marcha (SANTICIOLLI et al., 2015). Naturalmente, como este mapa é imutável, não há flexibilidade para adaptar a estratégia de mudança de marcha a diferentes condições de condução (HA et al., 2013). A calibração da programação de mudança de marcha é uma parte importante da indústria automobilística para melhorar a experiência do cliente (HE et al., 2021). A alteração da programação de mudança de marcha tem grande influência tanto sobre a economia de combustível quanto a dirigibilidade (KIM et al., 2017).

Normalmente, os motoristas se comportam de maneira diferente nas operações nos pedais do acelerador devido às suas preferências, mostrando diferentes estilos de condução, que podem ser aproximadamente classificados em agressivo, moderado e leve (HE et al., 2021). Através desse comportamento pode ser aplicado algoritmos para resolução de diversos problemas de otimização na área da engenharia. Dentre os processos de otimização aplicados, pode-se citar o *Particle Swarm Optimization* (PSO) que conduz a calibração da programação de mudanças de velocidades, personaliza modelos de direção empregados para equipar os modelos de acionamento da transmissão com preferências de condução da transmissão manual (HE et al., 2021).

Um outro modelo computacional capaz de identificar o comportamento do motorista é RNA. Essas redes são compostas por unidades conhecidas como “neurônios”, e o mais simples desses neurônios sendo o classificador *Perceptron* (AMARAL, 2021). Com o objetivo de processar informações de acordo com um prévio treinamento, a seleção de variáveis de entrada e saída é etapa essencial para desenvolvimento de modelos que aproximam bem o comportamento do sistema (COLOMBO, 2020). Isso explica o fato de que em diferentes estudos são oferecidos diferentes classificadores como os mais adequados. (NEDEVAL et al., 2020).

A identificação de Sistemas pode ser compreendida como um ciclo iterativo, em que o processo avalia as etapas anteriores e posteriores, num processo de melhoria continuadas, etapas envolvidas na obtenção de um modelo, até que os modelos desenvolvidos apresentem boa aproximação dos dados experimentais (COLOMBO, 2020).

O ajuste ideal representa um modelo computacional perfeitamente ajustado ao *dataset*, capaz de fazer previsões generalistas a partir de dados inéditos com margem de erro dentro do esperado. É possível detectar *under* e *overfitting* através de técnicas de validação, e então alterar os hiperparâmetros para adequar o ajuste da rede (AMARAL, 2021). Os parâmetros de desempenho são derivados da matriz de confusão para cada classificação (NEDEVAL et al.,

2020). De uma matriz de confusão binária, derivam-se as métricas de desempenho para a avaliação de modelos (AMARAL, 2021).

Os modelos computacionais podem ser aplicados para aprimorar a dirigibilidade através calibração dos mapas de mudança de marcha. A calibração automatizada pode ser realizada e refinada com veículos reais. Além disso, a função de calibração online pode ser integrada no sistema assistente para aperfeiçoar o desempenho das relações de trocas de marcha (HE et al., 2021), atendendo as necessidades do consumidor, baseado na redução do consumo de combustível.

1.2 OBJETIVOS

1.2.1 Objetivo Geral

Este trabalho propõe um estudo sobre a otimização da aplicação do sistema de transmissão automática, gerando economia de combustível e melhorando a dirigibilidade do veículo. Isso pode acontecer por meio da adaptação dos pontos de passagem das velocidades do módulo eletrônico de controle da transmissão em função da vontade do condutor, do estilo de condução, do perfil da estrada, da carga do veículo e das reações repentinas do condutor, através da aplicação de RNA, que podem efetuar a classificação do melhor método de controle da caixa de transmissão automática.

1.2.2 Objetivos Específicos

Para alcançar o objetivo principal deste trabalho, os objetivos específicos abaixo foram delimitados:

- Apresentar um modelo que demonstre como a TCU altera o seu funcionamento a partir de uma classificação gerada por uma RNA. Assim, essa classificação pode ser usada na adaptação ao modo de dirigir do motorista do modo econômico ao modo esporte e no perfil de carga como estradas, serras e centros urbanos.
- Analisar os modelos de RNA do tipo *Multilayer Perceptron* (MLP) com algoritmo de aprendizado de retropropagação, pela interação de um número de neurônios artificiais de processamento, que recebem sinais externos e produzem um sinal de saída.

- Realizar a aquisição de dados aplicado nesse estudo, através de um *scanner bluetooth* automotivo instalado junto ao conector *on board diagnostics* (OBD2), que significa “diagnóstico de bordo”, que permite acessar a central eletrônica do carro, realizar a leitura e transmitir os diversos tipos de dados. Através de um *smartphone* conectado, via *bluetooth*, a central eletrônica, o aplicativo torque que utiliza o *plugin realtime charts*, escolhidos para esse trabalho obtém os dados empregados.
- Utilizar o método de extração de conhecimento, denominado *Knowledge-Discovery in Databases* (KDD), para criar um conjunto de dados que envolve as variáveis necessárias, garantir a qualidade dos dados envolvidos no processo, colocar os dados de forma apropriada para a técnica de mineração a ser utilizada e executar o algoritmo de aprendizagem para classificação da melhor estratégia de controle do câmbio automático.
- Interpretar os resultados, com possível retorno aos passos anteriores, para fortalecer e agrupar o conhecimento e divulgar os resultados considerando as diferentes características de estrada e tipos de condução.

1.3 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está estruturado da seguinte forma:

- No capítulo 1, apresenta-se a introdução que elucida de forma avançada a proposta desse estudo a otimização das mudanças de marcha de um câmbio automático. No estado da arte compreende pesquisa aplicadas ao tema desse estudo como: eficiência das transmissões automáticas convencionais, modelos de algoritmos capazes de otimizar o controle das transmissões e desempenho e validação dos modelos computacionais. Por fim, aplica-se a identificação do objetivo geral e dos objetivos específicos.
- No capítulo 2, tem-se a presença do referencial teórico que discute teorias, conceitos e pesquisas que auxiliam no desenvolvimento desse trabalho. Demonstrou-se neste capítulo teorias articuladas sobre a mecânica que compõem o câmbio automático, os sistemas inteligentes e as RNA.
- No capítulo 3, tem-se a escolha do método para elaboração do trabalho, descreve-se como a pesquisa será aplicada com a preocupação em apresentar os detalhes

para que outras pesquisas possam ser elaboradas a partir desse estudo. Foi demonstrado os dispositivos aplicados a aquisição de dados e como foi realizado o trabalho junto aos dados para a realização da pesquisa, foi explorado a aplicação dos algoritmos escolhidos e os softwares utilizados.

- No capítulo 4, são examinados e tratados os resultados obtidos nas simulações realizadas, entregando respostas a proposta do projeto de pesquisa.
- O capítulo 5, apresenta uma conclusão do estudo realizado, ofertando uma seleção dos principais resultados obtidos, agrupando os seus benefícios com foco na área da pesquisa escolhida. Foi enumerado outras pesquisas que podem ser realizadas no futuro com base nos resultados dessa pesquisa desenvolvida.

2 FUNDAMENTAÇÃO TEÓRICA

Este capítulo tem como objetivo discutir a teoria e conceitos existentes que se tornam base para este estudo aplicado, apresentando a estrutura de funcionamento de um câmbio automático com seus principais componentes mecânicos. É discutido os fundamentos aplicados ao sistema de controle tradicional de troca de marcha, e exposto estratégias de controle adaptativo. O Aprendizado de Máquina é formalizado e analisado através da identificação de algoritmos e dos parâmetros de avaliação e desempenho. O sistema de descoberta de conhecimento é oficializado pela compreensão das suas fases de desenvolvimento e a arquiteturas de Redes Neurais Artificiais aplicada ao aprendizado é formalizada e analisada.

2.1 CÂMBIO AUTOMÁTICO.

A transmissão de um automóvel tem como função fornecer as forças de tração e impulsão necessária para induzir o movimento. Os valores característicos de potência e torque não são oferecidos uniformemente; os valores máximos estão disponíveis em faixas específicas. As relações de transmissão dos elementos da transmissão adaptam o torque disponível à forma de tração requerida no momento (BOSCH, 2005).

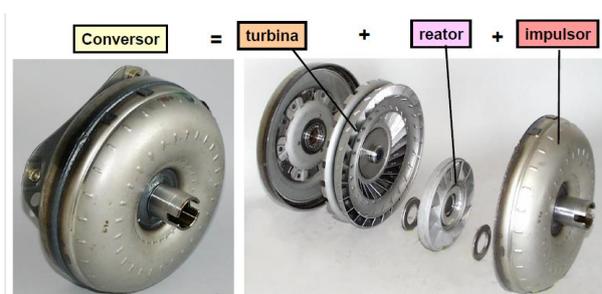
A transmissão automática possibilita que o próprio carro realize a troca de marcha, assim a TCU “percebe” quando a marcha deve ser trocada e efetua a mudança. Através desse sistema, o carro não precisa aplicar uma embreagem em seu sistema mecânico.

2.1.1 Estrutura básica de uma transmissão automática

A transmissão automática pode ser dividida em módulo de entrada de potência, módulo do conversor de torque, o módulo de conjuntos de engrenagens planetárias e o módulo de saída de energia. A energia gerada pelo motor (ZOU et al.,2022).

O conversor hidrodinâmico de torque é constituído por uma bomba que é o dispositivo propulsor, uma turbina que é o elemento movido e um reator que é o componente auxiliar do conversor de torque, conforme mostrado na Figura 2.1

Figura 2.1 – Conversor de torque.

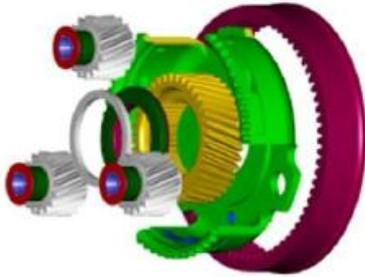


Fonte: DIAS (2011)

O conversor de torque é cheio de óleo e propaga o torque do motor pela energia cinética do óleo em circulação. O conversor anula a diferença de rotação entre o motor e a transmissão e por esse motivo é muito adequado para o arranque. No impulsor, a energia mecânica é convertida em energia cinética e à frente na turbina, convertida novamente em energia mecânica. O reator entre o impulsor e a turbina direciona o óleo para recirculação na bomba. Isso aumenta o torque concedido inicialmente pelo motor ao impulsor.

O conjunto planetário simples, observado na Figura 2.2, consiste de engrenagem planetária, coroa e ponte com engrenagens satélites. Cada elemento pode atuar como engrenagem motriz ou movida, ou também ser mantido estacionário. A concepção coaxial dos três elementos permite uma combinação ideal com acoplamentos e freios de fricção para acoplar ou fixar individualmente cada elemento. A mudança da relação de transmissão pode ser feita sem interrupção da tração. (BOSCH, 2005).

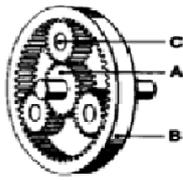
Figura 2.2 – Conjunto Planetário Simples.



Fonte: DIAS (2011)

Como várias engrenagens se agrupam simultaneamente sob carga, as transmissões são simplificadas. Elas não detêm forças de rolamentos livres, concedem níveis elevados de torque, fracionamento ou combinação de potência e exibe um nível de eficiência muito bom.

Figura 2.3 – Relação de transmissão de um conjunto de planetária



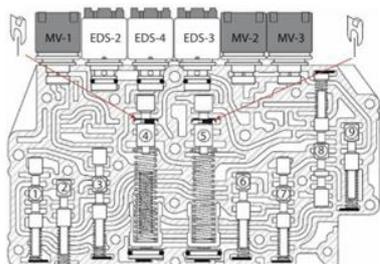
Entrada	Saída	Fixo	Relação de transmissão	Observações
A	C	B	$i=1+ZB/ZA$	$2.5 \leq i \leq 5$
B	C	A	$i=1+ZA/ZB$	$1.5 \leq i \leq 1.67$
C	A	B	$i=1/(1+ZB/ZA)$	$0.2 \leq i \leq 0.4$ marcha sobre multiplicada
C	B	A	$i=1/(1+ZA/ZB)$	$0.6 \leq i \leq 0.8$ marcha sobre multiplicada
A	B	C	$i=ZB/ZA$	Ponto morto com mudança no sentido de rotação - $0.4 \leq i \leq -1.5$
B	A	C	$i=ZA/ZB$	Ponto morto com mudança no sentido de rotação - $0.25 \leq i \leq -0.67$

Fonte: adaptado de BOSCH (2005).

As transmissões automáticas, trocam de marchas sob cargas, ou seja, mesmo durante a troca de marchas a propulsão do veículo é mantida, sem nenhuma interferência do motorista. O sistema de controle hidráulico transforma a carga do motor e a velocidade do veículo, em pressões hidráulicas aptas para serem aplicadas com harmonia na tarefa da troca de marchas.

O sistema hidráulico é composto por um cárter, uma bomba de óleo, uma válvula governadora, e um corpo de válvulas. A bomba de óleo é acionada pela bomba impulsora do conversor de torque. Ela movimenta sempre em sincronia com o motor. A válvula governadora é operada pelo pinhão motriz, e transforma a rotação do pinhão motriz em sinais hidráulicos, que são transmitidos ao corpo de válvulas. O corpo de válvulas é como um labirinto, visto na Figura 2.4, contendo muitos caminhos pelos quais passa o fluido hidráulico da transmissão. Nestas passagens estão contidas várias válvulas que se movimentam, abrindo e fechando estas passagens, continuando ou interrompendo os sinais de mudança hidráulica aos elementos de acionamento da unidade de engrenagens planetárias.

Figura 2.4 – Corpo de válvulas



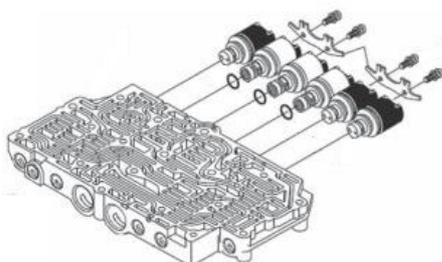
Fonte: adaptado de SENAI (2016).

O controle da transmissão automática regularmente é realizado por sistemas hidráulicos comandados eletronicamente. A operação hidráulica se baseia no acionamento da embreagem, enquanto a eletrônica se resume na escolha correta das marchas e na pressão apropriada em função do torque a ser transmitido.

Em sua atribuição como elementos adjuntos, sensores e atuadores constituem a interface entre o veículo e a ECU como a unidade de controle. Usualmente, um circuito-adaptador é aplicado para preparar os sinais do sensor na forma padronizada exigidos pela ECU, além disso, o funcionamento do sistema de controle pode sofrer influência de outras unidades de controle eletrônica ou através de equipamentos operados por driver.

Os sensores identificam a rotação do eixo de saída da transmissão, condições de cargas e rotação do motor, posição da alavanca seletora e do comutador “*Kick down*”. O sinal de controle organiza estas informações conforme um programa preestabelecido e define as variáveis que serão destinadas a caixa de mudanças. Conversores eletro-hidráulicos representam a interface entre a eletrônica e a hidráulica, como exibida na Figura 2.5. Válvulas solenoides simples fazem com que os acoplamentos funcionem. Reguladores de pressão analógicos ou digitais garantem a regulação precisa da pressão nos dispositivos de fricção.

Figura 2.5 – Eletroválvulas.



Fonte: adaptado de SENAI (2016).

Conforme a carga do motor e a rotação do eixo de saída da caixa de velocidade, o sistema eletrônico seleciona a marcha a ser utilizada através das válvulas solenoides. O condutor pode optar por um dos programas disponíveis para troca de marcha. Adicionalmente a alavanca seletora permite ao motorista interferir manualmente na caixa de mudanças.

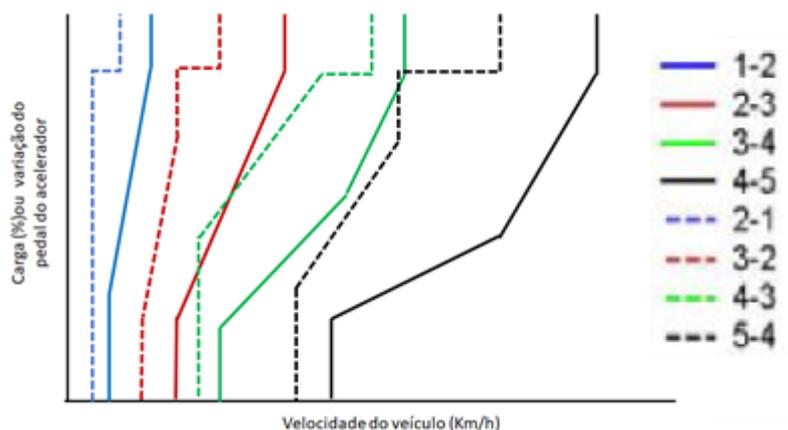
Integrando aos parâmetros do sistema algumas variáveis, como ângulo de inclinação da carroceria, pressão de admissão do ar para o motor e potência nas rodas os controles “inteligentes” para a troca de marchas aprimoram a condução do veículo.

2.2 SISTEMA DE CONTROLE.

2.2.1 Estratégias de controles tradicionais

A TCU é projetada para atuar no controle das engrenagens planetárias com um conversor de torque. As convencionais, selecionam manualmente os padrões de mudança de marcha aplicada a relação de transmissão. Padrões de mudanças consiste em curvas características, que descrevem quando uma mudança de marcha deve ocorrer de acordo com a velocidade e posição do pedal do acelerador, Figura 2.6.

Figura 2.6 – Padrões de mudança.



O motorista pode optar por diversos programas de trocas por exemplo, maior economia ou maior performance (BOSCH, 2005). Esses padrões de mudança não possuem uma boa adaptação a relação de transmissão para as diversas condições de condução e eram selecionados manualmente.

Somente a velocidade de saída da transmissão e a posição do pedal do acelerador são considerados para determinar a seleção de marcha adequada. Isso não é suficiente para envolver

por completo o sistema controle, como a alteração da carga do veículo ou o modo de condução do motorista. Um caso típico do afinamento da transmissão automática ocorre quando, ao conduzir em "inclinações íngremes", o acionamento do motorista reduz a abertura do acelerador e hipoteticamente aciona os freios. A intenção é dirigir com a menor relação de transmissão possível e maior torque de frenagem do motor de combustão, o sistema de controle de transmissão faz exatamente o oposto. Um outro exemplo acontece quando, diante de um aclive, altera à marcha seguinte toda vez que se diminuía o acionamento do acelerador. Essa fraqueza ocorre em sistemas de controle de transmissão automática tradicionais, que possuem padrões de mudança fixos.

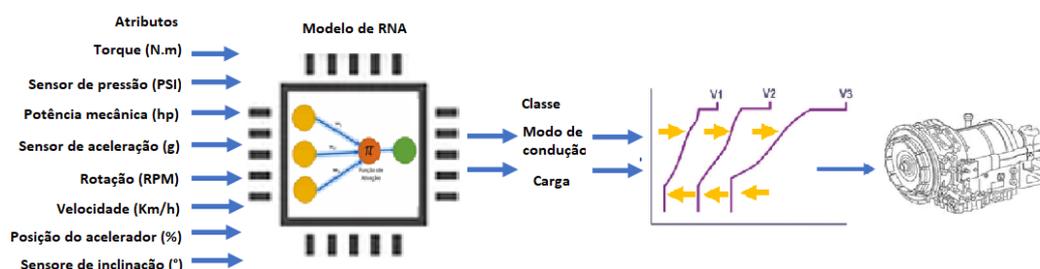
2.2.2 Estrutura do sistema de controle

Um módulo de controle eletrônico, deve possuir informações necessárias a respeito do local onde veículo está trafegando, isso significa a aplicação de sensores. Carros modernos são equipados com muitos sensores diferentes para sistemas de controles diferentes, como por exemplo Piloto Automático, Sistema de Gerenciamento do Motor e *Antilock braking system* (ABS).

Para aperfeiçoar a resposta esperada pelo motorista, algumas soluções tem sido observadas e implementadas, onde essas estratégias convencionais de controle se apoiam em modelos matemáticos que busca descrever o comportamento humano. Dessa forma é necessário configurar um modelo que demonstre o esforço que corresponde a essa funcionalidade direcionada.

Um módulo de controle inteligente deve superar esta metodologia e introduzir métodos que possam substituir a implementação por descrição matemática, especialmente se não houver nenhum contexto ou descrição prática óbvia disponível. A Figura 2.7, mostra a estrutura geral do sistema de controle da TCU baseado em RNA.

Figura 2.7 – Modelo de controle baseado em RNA.



O sistema desempenha o papel de um observador que seleciona a estratégia de mudança adaptada por meio de um padrão. Todas as necessidades de troca de marcha resultam da ultrapassagem da linha de mudança que são deslocadas ou suspensas para se adaptarem à situação de condução.

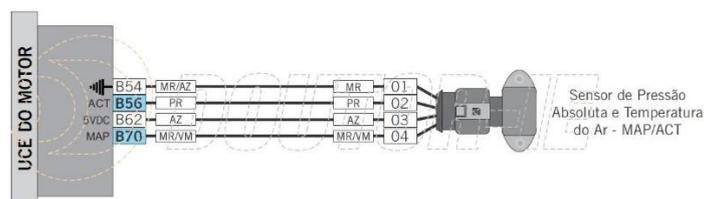
O controle adaptativo baseado em RNA necessita de oito sinais de informações diferentes e estima dois sinais distintos. O primeiro sinal identifica a descoberta do modo de condução, enquanto o segundo descobre o tipo de carga. A escolha da estratégia adotada pela TCU se aplica nas mais adequadas técnicas de condução.

A RNA utiliza a forma como o torque do motor está sendo produzido em uma determinada velocidade e rotação, com a devida posição do pedal do acelerador, e quanto a aceleração e a velocidade do motor variam. Sinais dos acelerômetros do controle de estabilidade ajudam com informações destinadas a aclives e declives. O trabalho realizado pela transmissão pode ser identificado pela potência nas rodas enquanto a pressão de ar de admissão é aplicada no cálculo de massa de ar admitida que implica no comportamento do motor durante seu funcionamento.

O torque do motor é calculado pela ECU a partir do deslocamento do cilindro, para determinação do deslocamento do motor, e da pressão média do cilindro. O sinal enviado pelo Sensor de Pressão Absoluta do Coletor de Admissão – MAP (*Manifold Absolute Pressure*), visto na Figura 2.8, indica a pressão absoluta medida na parte interna do coletor de admissão, ou seja, ele é um sensor que mede pressão (BLOG DOUTOR-IE, 2023).

Os padrões de valores de pressão absoluta do coletor de admissão variam de acordo com o tipo de motor. Nos motores aspirados, a pressão nunca será maior que a pressão atmosférica (pressão atmosférica: aproximadamente 1 bar ou 750 mmHg), pelo contrário, os valores serão sempre iguais ou menores pois a entrada de ar no motor é feita de forma natural. Os valores menores que 1 bar se dão pela depressão no coletor de admissão causada pelo movimento de declive do pistão na fase de admissão em conjunto com a restrição na entrada de ar causada pela borboleta de aceleração (BLOG DOUTOR-IE, 2023).

Figura 2.8 – Esquema de ligação do sensor MAP/ACT.



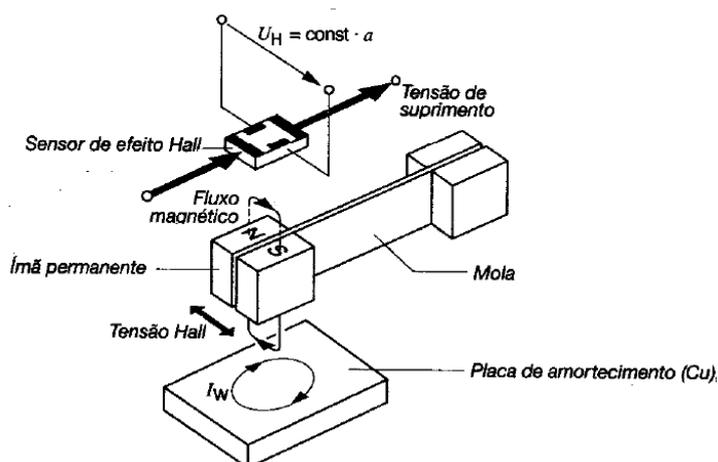
Fonte: BLOG DOUTOR-IE (2023).

O sensor de posição do acelerador, também conhecido como posicionador da borboleta, é um potenciômetro que mede o deslocamento angular através de uma alteração resistiva proporcional ao deslocamento, a posição do sensor é utilizada para o reconhecimento das posições de pé levantado, plena carga e para os momentos de transição.

Os sensores de rotação são dispositivos responsáveis pela análise de rotação de entrada e saída da transmissão. Essa classe de informação é importante para o controle, pois indica ao módulo por meio das diferenças de rotações, o modo de trabalho de cada marcha. Esses sensores são instalados na carcaça do câmbio e realizam a leitura de sinais por meio de elementos montados dentro da transmissão, nas rodas dentadas ou nos tambores. Os sensores de rotação são indutivos em função disso, geram um sinal na forma de onda senoidal que quando chega ao módulo são convertidos em sinais digitais, de onda quadrada.

Os sensores de aceleração, Figura 2.9, são dispostos em carros com ABS e sistemas de controle de estabilidade eletrônica, os sensores de velocidade instalados na roda são incorporados por um sensor de aceleração *Hall*, com o objetivo de avaliar as taxas de aceleração lateral e longitudinal. Os níveis de erros no sistema massa/mola são utilizados nessa aplicação e são registrados através do uso de um ímã e um sensor de efeito *Hall*. O sinal gerado pelo sensor possui um amortecimento eletrodinâmico e funciona em uma faixa estreita de frequência (vários Hz).

Figura 2.9 – Sensor de aceleração Hall.



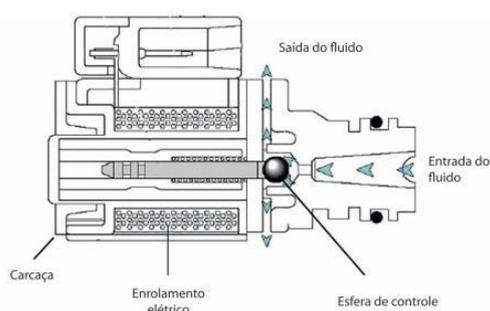
Fonte: adaptado de BOSCH (2005).

Um módulo de controle inteligente realiza função de ação na mudança de marcha adaptada através dos dois sinais de controle estimados, sendo eles modo de condução e carga. A operação é gerenciada através do comando eletrohidráulico, que abrange a TCU e o corpo

mecatrônico, o conjunto possui eletroválvulas “*on-off*”, conhecidas como válvulas de sequências, que atuam na mudança de marchas.

As eletroválvulas, Figura 2.10 são constituídas de esferas ou pistão e podem ser do tipo normalmente aberta ou normalmente fechada. Para realizar uma troca de marcha, a unidade de comando alimenta com uma tensão as válvulas solenoides correspondentes à marcha que vai ser engatada. Através da combinação das válvulas abertas e fechadas ocorre a passagem de óleo sob pressão para o circuito hidráulico da marcha desejada.

Figura 2.10 – Eletroválvula.



Fonte: adaptado de SENAI (2016)

2.2.3 Método de controle

No controle do câmbio automático, a escolha de passagem de uma relação para outra é tomada a partir de um conjunto de curvas, denominados padrões de mudanças que estabelece as "regras de passagem".

Um conjunto de regras de passagem, proporciona todas as relações disponíveis e define os pontos de passagem e as relações a passar. Como princípio na concepção de um conjunto das regras de passagem, tem-se a escolha da relação com que se obtém o menor consumo de combustível em relação à resistência a aceleração e a seleção da relação que libera aplicar à potência máxima possível.

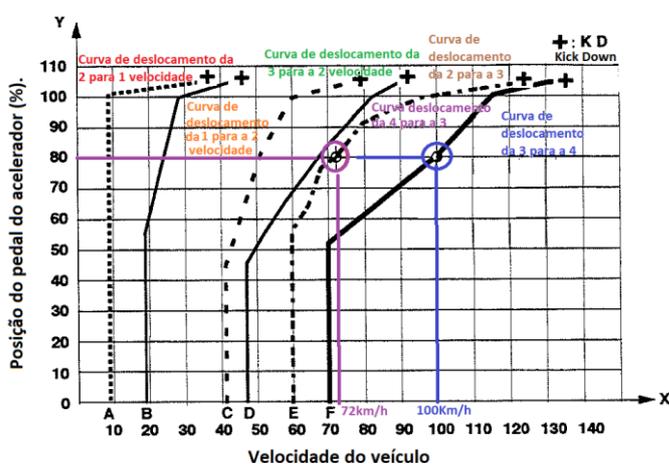
A troca de relação é conseguida a velocidade crescente, uma vez que o ponto de funcionamento corte a curva (velocidade crescente ou decrescente). Os padrões de passagem são diferentes em aclave e em declive de relação, cria-se uma histerese para os padrões, como objetivo de evitar mudanças repetidas.

A apresentação das curvas aplicadas as regras de passagem 1/2, 2/3 e 3/4 e demais, será realizada quando a curva de progressão do ponto de atuação as cruzar vinda da esquerda

(velocidade do veículo crescente) e se o veículo ainda não estiver em segunda marcha para a regra 1/2, em terceira marcha para a regra 2/3, ou em quarta marcha para a regra 3/4.

A apresentação das curvas aplicadas as regras de passagem 4/3, 3/2 e 2/1, será efetuada quando a curva de progressão do ponto de funcionamento as cruzar vinda da direita (velocidade do veículo decrescente) e se o veículo ainda não estiver em terceira marcha para a regra 4/3, em segunda marcha para a regra 3/2, ou em primeira marcha para a regra 2/1. As regras aplicadas a passagem de marcha são demonstradas na Figura 2.11.

Figura 2.11 – Padrões de passagem ascendentes e descendentes.



As regras de passagens auto adaptativas permitem a TCU adaptar - se automaticamente aos pontos de mudança das velocidades em função do desejo do condutor e do estilo de condução do motorista, esportivo ou tendo em vista a economia de combustível. Isto possibilita a TCU adotar uma condução esportiva por exemplo, selecionando um conjunto de regras específicas. O carácter esportivo pode ser aplicado a longo prazo (sustentação da condução no tempo) ou a curto prazo (ultrapassagem de um veículo por exemplo).

O perfil da estrada, representada pelo grau de declive e da carga do veículo que indica a resistência a aceleração, possibilita, ao veículo quando se encontra em um aclive, utilizar um conjunto de regras específicas adaptadas ao percurso de serra. Assim é possível evitar principalmente os fenômenos de bombagem. Da mesma maneira, quando o veículo se encontra em declive, é escolhido um conjunto de regras específicas com o objetivo de manter ou obter a frenagem do motor.

A determinação do parâmetro do condutor, apresenta a regra mais adequada ao comportamento do motorista. A escolha é realizada a parti de valores médios adquiridos nos últimos minutos de funcionamento do veículo ou a partir de valores momentâneos em função

do aumento repentino de esportividade. As principais informações utilizadas são a posição da borboleta, potência de saída da transmissão e velocidade do veículo.

2.2.4 Redes de comunicação nos veículos automotivos e processo de aquisição de dados

Em sua totalidade os veículos produzidos e comercializados atualmente possuem sistemas eletrônicos de controle independentes, cada qual cumpre uma determinada tarefa de controle no veículo, essas tarefas são realizadas pelos microcontroladores.

Em contrapartida, o real domínio dos diversos dados eletrônicos disponíveis em um automóvel é mais facilmente conseguido com a utilização de sistemas eletrônicos interligados, cada qual responsável por uma parte do veículo, mas compartilhando informações entre si (GUIMARÃES, 2007).

Para que a comunicação entre os módulos eletrônicos efetue – se, é utilizado um meio de transmissão e recepção de dados denominados Protocolos de Rede, que são um conjunto de normas que permitem que qualquer máquina possa se comunicar com outra também já conectada na rede (WEBLINK,2023).

As informações coletadas durante a tarefa de aquisição de dados aplicados no modelo da RNA, pertence a sensores conectados a ECU por uma rede, dentre ele tem-se o sensor de velocidade, sensor de rotação, sensor de aceleração, sensor de inclinação, etc.

Esses dados trafegam entre os módulos eletrônicos do veículo através da rede CAN e estão disponíveis para diagnóstico e leitura. A rede CAN foi desenvolvida pela Bosch em 1986, ela é a principal rede de comunicação dos veículos, pois possibilita a troca de informação em tempo real. Para garantir os requisitos de operação as taxas de transmissão ficam entre 125Kbits/s e 1Mbits/s (BOSCH, 2005).

A rede CAN tem como particularidade tornar padronizada e assegurar a comunicação entre os microcontroladores, mesmo com a inexistência de um elemento eletrônico “centralizador” (*host*).

A aquisição de dados utiliza o sistema *On Board Diagnostico* (OBD2) que significa “diagnóstico de bordo” e é uma norma para diagnóstico de veículo, o sistema OBD2 basicamente verifica os códigos de falha armazenados no módulo eletrônicos e disponibiliza através de uma porta de diagnóstico por um protocolo padrão. (GUIMARÃES, 2007).

No Brasil, a padronização OBD-2 foi adotada, em 2010. Sua implementação, assim como no restante do mundo, trouxe consigo, além de diversas regras, a interface de comunicação de 16 pinos *Diagnostic Link Connector* (DLC). Essa interface permite uma

comunicação direta com a ECU e outros módulos para monitoração e controle, com algumas restrições, de todas as funções e parâmetros comandados pela central veicular (BARRETO, 2018).

Em 2011 foi definido que todos os automóveis manufaturados vendidos no Brasil deveriam utilizar as especificações OBD-2 para monitoramento. Foi estabelecido ainda que todas as funções do motor deveriam ser monitoradas e reguladas com o objetivo de diminuir a emissão de poluentes. A especificação possibilita a sistemas internos e externos (via DLC) solicitar informações e ter retorno sobre o respectivo sensor ou funcionalidade (BARRETO, 2018).

2.3 SISTEMA INTELIGENTE

2.3.1 Inteligência Artificial

Segundo (SIMÕES et al., 2017), IA é uma disciplina que estuda como as pessoas resolvem problemas e como as máquinas podem emular este comportamento humano de “solução de problemas”. Em outras palavras, como fazer com que as máquinas sejam mais atribuídas de características da inteligência humana.

As máquinas não estão somente fazendo trabalhos manuais, mas também trabalhos racionais, tarefas que requerem o uso do que se considera inteligência (LUDEMIR, 2021).

De maneira simplificada, pode – se afirmar que a área da IA procura utilizar os computadores para simular o processo do pensamento humano durante a resolução de problemas (CAMPOS et al., 2004).

A IA se aplica a vários problemas complexos e consegue alcançar soluções com sucesso, através de suas técnicas. Problemas aplicados a IA têm um ponto em comum, não há para esses problemas soluções algorítmicas, ou seja, não se consegue descrever uma maneira de solucionar – lá, mas existe muitas informações (dados) aplicados a esses problemas, o que permite treinar um algoritmo e aplicar o aprendizado de máquina.

2.3.2 Aprendizado de Máquina

Na computação tradicional os programadores criam os programas, códigos que são executados pelos microcontroladores ou computadores.

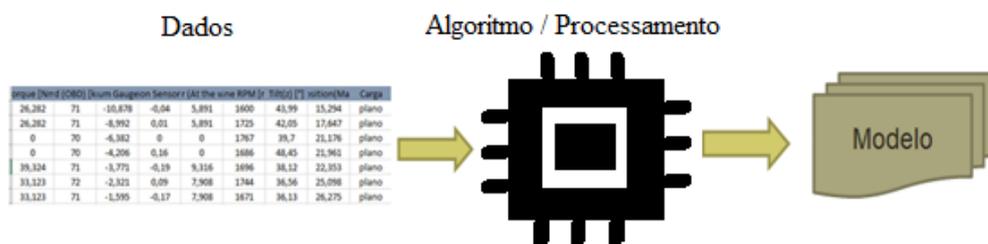
O ML é uma área de IA cujo objetivo é o desenvolvimento de técnicas computacionais sobre o aprendizado bem como a construção de sistemas capazes de adquirir conhecimento de forma automática, com o início na análise de dados e seus produtos, ou sobre a estrutura e arranjo de um conjunto de dados.

Então se considera inteligência artificial um tipo de inteligência produzida pelo homem para beneficiar as máquinas de algum tipo de habilidade que simula a inteligência natural do homem (DA SILVA et al., 2010). ML é basicamente a aptidão aplicada pelos computadores de aprenderem a realizar uma tarefa ao invés de serem programadas para tal.

O método mais aplicado no aprendizado de máquina se dá com a utilização de um conjunto de dados, que geram um conjunto de respostas a partir desses dados, conforme demonstrado na Figura 2.15. O ML irá desenvolver um modelo computacional ou um modelo matemático que faça com que as repostas sejam alcançadas.

A partir do modelo criado, toda vez que um conjunto de dados exatamente igual são aplicados, um resultado exatamente igual será devolvido. O modelo irá avaliar cada dado e o computador que utiliza esse modelo irá fazer a previsão.

Figura 2.15 – Aprendizado de Máquina



Existem três tipos principais de Aprendizado de Máquina, ilustrado pela Figura 2.16: supervisionado, não supervisionado e por reforço (LUDEMIR, 2021).

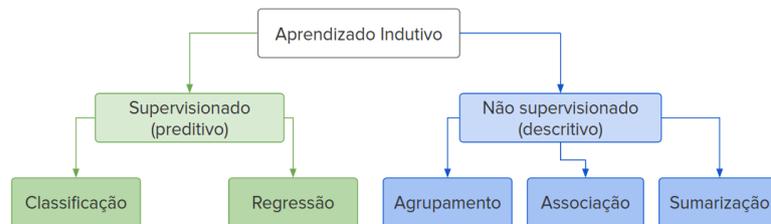
O Aprendizado supervisionado normalmente é preditivo, traz um objetivo estabelecido e pode ser dividido entre problemas de regressão e de classificação. No aprendizado supervisionado, para cada exemplo apresentado ao algoritmo de aprendizado é necessário apresentar a resposta desejada (LUDEMIR, 2021). Existe uma identificação informando a que classe o exemplo pertence, no caso de um problema de classificação de modo de condução e carga do veículo, pode-se distinguir o desejo do motorista em ECO, NORMAL ou *SPORT* e o tipo de carga em C1, C2 e C3.

O objetivo do aprendizado não supervisionado é aprender estruturas desconhecidas que estão presentes nos dados (HOSSEINI – ASL et al., 2016). O algoritmo agrupa os exemplos

pelas similaridades dos seus atributos. O algoritmo analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser capazes de se associar, formando agrupamentos ou Clusters (LUDEMIR, 2021). Após a determinação dos agrupamentos, normalmente, é necessária uma análise para determinar o que cada agrupamento significa no contexto do problema que está sendo analisado (MONARD et al., 2003).

No Aprendizado por reforço ocorre quando as saídas não estão bem definidas e as respostas só podem ser aferidas após algumas execuções. No Aprendizado por Reforço, o algoritmo não recebe a resposta correta, mas recebe um sinal de reforço, de recompensa ou punição. O algoritmo faz uma hipótese baseado nos exemplos e determina se essa hipótese foi boa ou ruim (LUDEMIR, 2021).

Figura 2.16 – Tipos de métodos de aprendizado indutivo



Fonte: BARRETO (2018)

Os trabalhos de aprendizado supervisionado são caracterizados por uso do registro dos históricos de dados com resultados que representam algum valor com base em um atributo que permite dividi-los em subconjuntos. O resultado é importante para fazer uma caracterização dos objetos na classe de dados conhecidos como atributo alvo ou classe. Pode-se aplicar esses dados na identificação dos objetos mais similares ou dos objetos distintos. Essa tarefa é conhecida como predição, que é definida como ato de afirmar com convicção aquilo que poderá acontecer num momento futuro; previsão (DICIO, 2023).

O ML supervisionado soluciona problemas de Regressão e Classificação. Quando o resultado, ou classe, é numérico e todos os atributos são numéricos, a regressão linear é uma técnica natural a ser considerada. Este é um método básico em estatística. A ideia é expressar a classe como uma combinação linear dos atributos, com pesos pré-determinados (WITTEN et al., 2016).

A Regressão é uma maneira de classificar matematicamente quais variáveis realmente causam impacto sobre as outras. Responde às perguntas: Quais fatores são mais importantes?

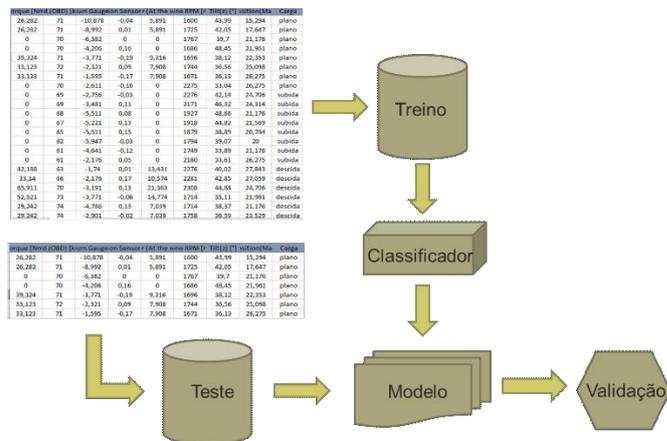
O que se pode ignorar? Como esses fatores interagem entre si? E, talvez o mais importante, até que ponto se está certo sobre todos esses fatores?

Os problemas de classificação, associam a descrição de um objeto a uma classe discreta, como por exemplo definir o estilo da condução do motorista como NORMAL, são utilizados para isso as características motor e da transmissão para realizar essa descrição. Normalmente a classe em uma relação está representada em um atributo especial. A classificação utiliza um conjunto de valores, objetos dentro de um conjunto de dados e divide em subconjuntos de acordo com as suas características semelhantes.

As etapas de construção do modelo preditivo dividi – se em treinamento onde o algoritmo aprenderá o modelo que utilizando os dados rotulados e teste para a verificar se o modelo é adequado para a tarefa de classificação. Um outro fator importante para a medição de desempenho do modelo é a existência de vários algoritmos e as diferentes parametrizações que permite a comparação entre eles. É importante conhecer que não existe um melhor modelo dentro do ML com um melhor ajuste, para um determinado algoritmo. Cada problema aplica um algoritmo diferente com parâmetros diferentes que vai ter um desempenho melhor.

Dentro do conjunto de dados, separou-se uma quantidade maior de dados para o treinamento e os dados de teste aplica os restantes dos dados que não foram usados durante o treinamento. O conjunto de dados pode ser dividido em 80% para treinamento e 30% para teste ou aplica – se também 70% dos dados para treinamento e 30% para teste. A Figura 2.17 ilustra esse processo de treino e teste.

Figura 2.17 – Treinamento e teste do modelo de ML



No ML não supervisionado ou descritivo, agrupam – se objetos de acordo com as suas características criando associações entre eles. Tem-se um conjunto de dados, onde não existe uma relação, com isso ML para separar os dados em grupos, irá realizar uma classificação por

similaridade entre os atributos. Se não há um atributo classe, o algoritmo tentará encontrar padrões nos conjuntos de dados. O ML não supervisionado é dividido em agrupamento, associação e sumarização.

No agrupamento não existe classe, não existe algo que se deseja prever, seja categórico ou numérico, o agrupamento baseia – se na similaridade das características dos objetos, com o objetivo de descobrir grupos homogêneos, agrupar os dados a partir de características ou semelhanças matemáticas.

Na associação, aplica-se a base de dados transacionais, nessas bases tem-se um conjunto de itens que são relacionados a uma operação e não necessariamente são os mesmos objetos, é comum ter essas bases de dados quando se utiliza dados empresariais, podendo ser aplicados na detecção do comportamento de compra dos clientes, nas campanhas de promoção e marketing e até na gestão de estoque.

Sumarização é a redução da dimensionalidade, necessária tanto na classificação quanto no agrupamento. Quando os dados são grandes e se gostaria de ter um resumo ou abstração dos dados em vez de todos os detalhes se utiliza a Sumarização (MURTY; DEVI, 2010).

O uso de ML para solucionar problemas nem sempre é fácil e precisa de alguns pré-requisitos. Precisa de um bom conjunto de exemplos. Muitas vezes a base de exemplos precisa ser construída e atualizada constantemente. Como os dados nem sempre são bons, faz-se necessário o uso de técnicas que melhorem a qualidade dos dados. Nem todo algoritmo de ML resolve todo tipo de problema, então é preciso fazer a seleção dos conjuntos de algoritmos apropriadas para o problema que se precisa resolver. Uma vez escolhidos os algoritmos, precisa-se definir os parâmetros dos algoritmos (por exemplo, o número de camadas de uma Rede Neural). Depois do treinamento precisa-se saber se o algoritmo está resolvendo o problema e com que precisão o problema está sendo resolvido. Por fim, o sistema precisa ser atualizado, porque mudanças nos dados podem fazer com que os sistemas deixem de funcionar (LUDEMIR, 2021).

2.3.3 Algoritmos de aprendizado de máquina

Para utilização de um algoritmo de aprendizado supervisionado é necessária uma base de treinamento e existem alguns conceitos básicos associados à base de treinamento que serão listados a seguir (NETO, 2021):

- Exemplo: também chamado de registro ou padrão é uma dupla de valores de atributos. Como exemplo pode-se citar um cliente, um paciente, um carro e etc.

- Atributo: descreve uma característica ou um aspecto de um exemplo. Ele pode ser classificado como nominal quando assume valores categóricos como, por exemplo: sexo e estado civil, ou pode ser classificado como contínuo quando assume valores numéricos como, por exemplo: peso, altura e idade.
- Classe: atributo especial também denominado rótulo. Ele é utilizado no aprendizado supervisionado para emular a figura do “professor” externo, que informa o valor que deseja ser aprendido para todas as variáveis de entrada correspondente aquele registro.
- Conjunto de exemplos: Um conjunto de exemplos é composto por exemplos contendo valores de atributos bem como a classe associada. Existem alguns nomes específicos para conjuntos de exemplos dependendo de sua finalidade: treinamento, validação ou teste

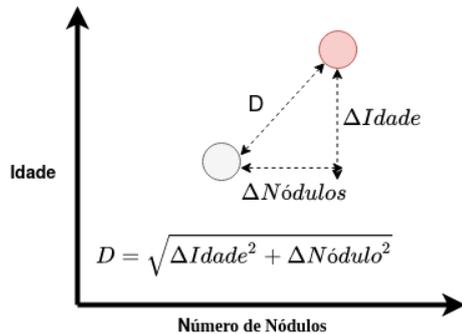
O aprendizado de máquina depende de diferentes algoritmos para resolver problemas de dados. Os cientistas de dados gostam de salientar que não existe um tipo de algoritmo que sirva para todos e que seja melhor para resolver um problema. O tipo de algoritmo empregado depende do tipo de problema que se deseja resolver, do número de variáveis, do tipo de modelo que melhor se adequa a ele e assim por diante (MAHESH, 2020).

Na aprendizagem não supervisionada, não há respostas corretas e não há treinamento. Os algoritmos são deixados por conta própria para descobrir e apresentar uma estrutura interessante dos dados. Os algoritmos de aprendizagem não supervisionados aprendem poucos recursos dos dados. Quando novos dados são introduzidos, ele utiliza os recursos aprendidos anteriormente para reconhecer a classe dos dados. É usado principalmente para *clustering* e redução de recursos (MAHESH, 2020).

Os algoritmos de aprendizado de máquina supervisionado são aqueles algoritmos que necessitam de assistência externa. O conjunto de dados de entrada é dividido em conjunto de dados de treinamento e teste (MAHESH, 2020).

Baseado por analogia o *k-Nearest Neighbors* (KNN), é um algoritmo apoiado em instâncias, ou seja, o modelo é caracterizado pelos exemplos de treinamento. O valor K constitui a quantidade de exemplos similares do conjunto de treinamento válido para classificação. Dado um valor inexplorado, um classificador KNN busca os K exemplos análogos ao padrão desconhecido para efetivar a classificação. O exemplo desconhecido é relacionado à classe que contém um maior número de registros correspondente. A correspondência entre exemplos é medida em termos de métricas de similaridade. Entre essas métricas tem-se a distância euclidiana.

Figura 2.18 – Cálculo da similaridade entre padrões no KNN usando a distância euclidiana.



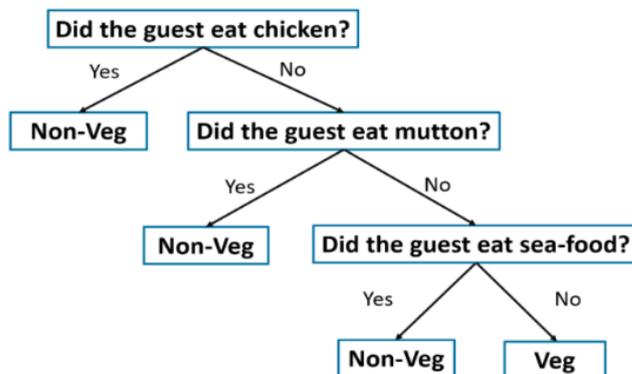
Fonte: NETO (2021).

De acordo com Beyer (1999) apud Neto, (2021, p.29) o KNN apresenta como vantagem sua simplicidade e facilidade de implementação. No entanto, ele é extremamente sensível a tributos irrelevantes e redundantes, uma vez que todos os atributos contribuem de forma igualitária para a classificação através do cálculo da distância.

Árvores de decisão, são regras determinadas para atributos que são estruturados como uma árvore. Esse tipo de algoritmo é conhecido como modelo em caixa branca porque é possível verificar a criação da árvore e a estrutura das regras que aplica a função do conectivo lógico *AND*.

Os nós no gráfico representam um evento ou escolha e as bordas do gráfico representam as regras ou condições de decisão, conforme demonstrado na Figura 2.19. Cada árvore consiste em nós e ramos. Cada nó representa atributos em um grupo que será classificado e cada ramificação representa um valor que o nó pode assumir (MAHESH, 2020).

Figura 2.19 – Árvore de decisão



Fonte: MAHESH (2020).

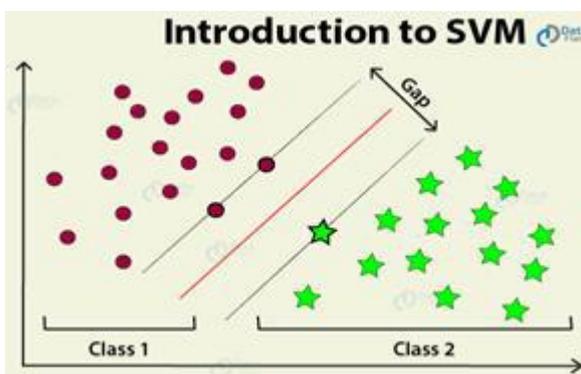
O *Naïve Bayes* é uma das técnicas mais simples de classificação, baseada no Teorema de Bayes, que é a independência condicional das classes. Um classificador Naïve Bayes assume que a presença de um atributo específico não está relacionada à presença de qualquer outro atributo. É usado principalmente para fins de agrupamento e classificação, e utiliza a probabilidade condicional.

$$P(A/B) = \frac{P(B/A) \times P(A)}{P(B)} \quad (1)$$

Onde: $P(A/B)$ e $P(B/A)$ são as probabilidades a posterior, $P(A)$ e $P(B)$ são as probabilidades a prior.

Support Vector Machine (SVM) são modelos de aprendizado supervisionado com algoritmos de aprendizado associados que analisam dados usados para classificação e análise de regressão. Além de realizar classificação linear, os SVM podem realizar com eficiência uma classificação não linear usando o que é chamado de truque do kernel, mapeando implicitamente suas entradas em espaços de recursos de alta dimensão. Basicamente, traça margens entre as classes. As margens são traçadas de forma que a distância entre a margem e as classes seja máxima e, portanto, minimizando o erro de classificação (MAHESH, 2020).

Figura 2.20 – *Support Vector Machine*

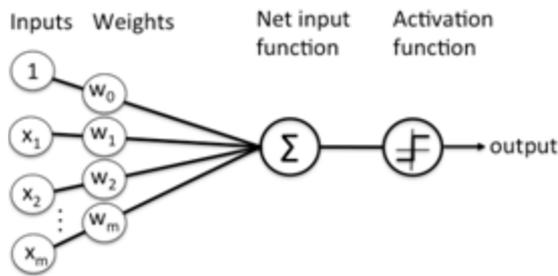


Fonte: MAHESH (2020).

Rede neural é um mecanismo de inteligência artificial que informa aos computadores o método de organizar dados de um jeito orientado pelo cérebro humano. É um tipo de processo de aprendizado de máquina, chamado aprendizado profundo, que emprega nós ou neurônios interconectados através de um arranjo em camadas, semelhante ao cérebro humano.

Uma rede neural é uma das vertentes da inteligência artificial, do qual o objetivo é esboçar os mecanismos analíticos executados pelo cérebro humano. As funções que uma rede neural pode resolver são a classificação, a previsão e o reconhecimento.

Figura 2.21 – Rede Neural Artificial



Fonte: MAHESH (2020).

2.3.4 Parâmetros de avaliação de desempenho dos algoritmos.

A avaliação de desempenho dos algoritmos que é frequentemente utilizada em ML, permite medir se a diferença entre quaisquer algoritmos diferentes é significativa. É importante medir e avaliar os resultados para termos certeza que o modelo criado através do ML é conveniente para ser aplicado na vida real. Para calcular o controle de abstração é necessário aplicar métricas de avaliação de desempenho.

As métricas de avaliação de classificação são utilizadas quando a variável alvo possui valor categórico. De uma forma geral, as métricas de avaliação de desempenho para problemas de classificação podem ser descritas a partir de uma matriz de confusão (NETO, 2021).

Uma matriz de confusão, contém informações sobre classificações reais e previstas feitas por um sistema de classificação. O desempenho de tais sistemas é geralmente avaliado usando os dados na matriz (GOMEDE, 2020). A Tabela 2.2 a seguir mostra a matriz de confusão para um classificador de duas classes.

Tabela 2.1 – Elementos da Matriz de Confusão

Classe Verdadeira	Classe Prevista	
	Positivo	Negativo
Positivo	TP	FP
Negativo	FN	TN

As entradas na matriz de confusão têm o seguinte significado no contexto dessa pesquisa. Vários termos padronizados foram definidos para a matriz de 2 classes, binária (GOMEDE, 2020):

- TP é o número de previsões corretas de que uma instância é negativa,
- FP é o número de previsões incorretas de que uma instância é positiva,
- FN é o número de previsões incorretas que uma instância é negativa e

- TN é o número de previsões corretas de que uma instância é positiva.

Outras métricas de avaliação de desempenho podem ser derivadas a partir da matriz de confusão. Entre elas pode-se destacar: acurácia, precisão, sensibilidade, especificidade e a área sobre a curva ROC - *Receiver Operating Characteristic* (NETO,2021).

A acurácia verifica o total de acertos, ela é a métrica mais simples. É indicada pelo uso da equação 2 uma vez que, quanto maior, melhor o classificador. A precisão considera a proporção de casos positivos previstos, ela indica o quanto o modelo acerta na identificação da tarefa e pode ser representada na equação 3. A sensibilidade é uma métrica de desempenho utilizada na avaliação da eficiência e eficácia de um modelo de classificação. Ela mede a proporção de casos positivos reais, sendo representada pela equação 4. A especificidade é a razão de casos negativos que foram incorretamente classificados como positivos, calculados usando a equação 5.

$$\text{Acurácia} = \frac{TP+TN}{TP+FN+FP+FN} \quad (2)$$

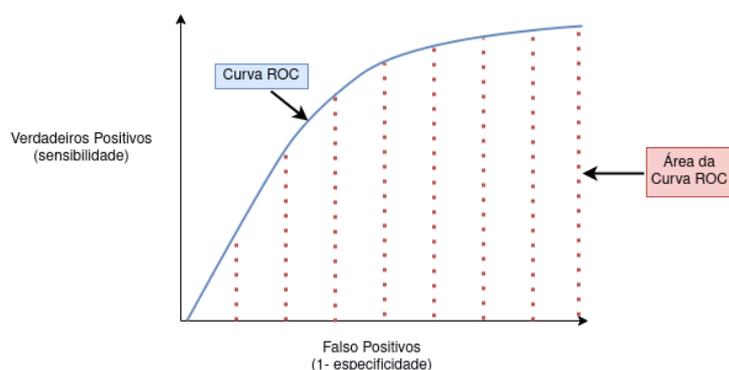
$$\text{Precisão} = \frac{TP}{TP+TN} \quad (3)$$

$$\text{Sensibilidade (Recall)} = \frac{TP}{TP+FP} \quad (4)$$

$$\text{Especificidade} = \frac{TN}{FN+TN} \quad (5)$$

O gráfico ROC é uma métrica que realiza a análise por meio de um método gráfico simples e robusto, o qual permite estudar a variação da sensibilidade e especificidade do modelo para diferentes valores de ponto de corte Townsend, Papes e Soberón (2008) apud Neto, (2021, p.73). O gráfico ROC avalia a relação das taxas de falsos positivos e verdadeiros positivos, sendo usados para avaliar o desempenho de classificação binária, como observado na figura 2.22, pode variar entre 0 e 1 e quanto mais próximo de 1 melhor a performance do classificador.

Figura 2.22 – Curva ROC



Fonte: NETO (2021)

2.4 DESCOBERTA DO CONHECIMENTO

2.4.1 Definição

O termo Descoberta do Conhecimento em Banco de Dados – DCBD (*Knowledge Discovery in Databases - KDD*) foi formalizado em 1989 Parsaye, (1989) apud Schneider (2003, p.10) em referência ao processo de procurar conhecimento em dados. Este processo tem como objetivo extrair as relações implícitas e previamente desconhecidas entre os dados, na busca de informações potencialmente úteis.

Segundo Bruha, (2006) apud Camilo (2009, p.6) o DCBC pode ser vista como uma atividade multidisciplinar, pois explora a pesquisa de diversos ramos da inteligência artificial, como aprendizado de máquina, reconhecimento de padrões, sistemas inteligentes, aquisição de conhecimento, bem como disciplinas matemáticas, como estatística, teoria da informação e processamento de incerteza

2.4.2 Compreensão do desenvolvimento do KDD

Seja em qualquer campo, o estudo clássico de análise de dados, baseia – se em um ou mais pesquisadores, intimamente familiarizado com os dados trabalhados e funcionando como uma interface entre os dados, os usuários e os produtos.

Para inúmeras aplicações, esta forma de pesquisa manual de dados é demorada, onerosa e abstrata. À medida que a quantidade de dados cresce, esse tipo de trabalho se torna custoso e completamente impraticável. A demanda de ampliação das capacidades de análise humana, para tratar com uma grande quantidade de número de dados armazenados é tanto econômica quanto científica. Os dados armazenados são aplicados para construir teorias, modelos de problemas e trabalho do universo em que se vive.

A quantidade de informações que se consegue coletar através do computador é maior do que a capacidade que se tem de tratá-la, então é natural utilizar técnicas computacionais para encontrar importantes padrões e tratar grandes estruturas com grandes volumes de dados. Assim sendo, o KDD é uma tentativa de solucionar um problema de informação digital para todos nós: a sobrecarga de dados.

2.4.3 O Processo KDD

KDD é o processo não trivial de identificação de padrões válidos, novos, potencialmente úteis e em última análise, compreensíveis em dados (FAYYAD et al., 1996). Os dados são uma coleção de valores que transmitem informações, descrevendo quantidade e qualidade, e padrão é um termo que descreve um subconjunto de dados ou um modelo apropriado ao subconjunto.

Portanto, extrair um padrão significa ajustar um modelo aos dados, realizando qualquer definição de alto nível de um conjunto de dados.

KDD é um processo de descoberta de conhecimento com várias etapas, sendo não trivial porque existe uma dificuldade na execução e manutenção dos processos de KDD, isto é, é não é um cálculo direto de quantidades predefinidas, como calcular a média, iterativo pois existe a possibilidade de repetição das etapas em qualquer momento do processo e interativo, pois o é possível retomar em qualquer uma das etapas deste processo. Assim o processo de KDD possibilita a identificação de padrões explícitos, apropriados, e potencialmente uteis. O processo KDD é cooperativo entre humanos e computadores. Humanos projetam a arquitetura do sistema, descrevem problemas e identificam os objetos a serem atingidos, enquanto, os computadores, processam os dados a procura de padrões, a fim de satisfazer os objetivos anteriormente definidos. (SCHNEIDER, 2003).

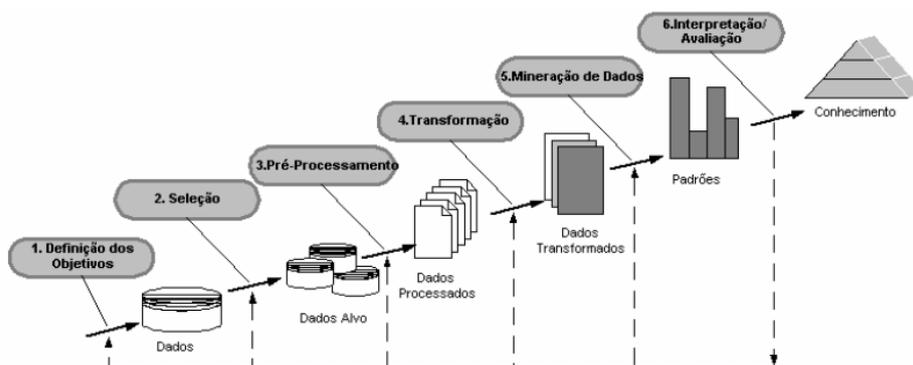
Usualmente, o processo KDD é dividido em 6 etapas (Figura 2.23):

- Definição dos Objetivos – O processo de KDD inicia com a ideia do domínio da aplicação e dos objetivos a serem alcançados, ou seja, nessa fase é definido o que se pretende conhecer e quais são os objetivos com o conhecimento obtido.
- Seleção – Esta etapa abrange a seleção de um conjunto ou subconjunto de dados que se torna parte da análise. A base de dados pode ser variada como: planilhas, bancos de dados, *datawarehouse* e traz dados com formatos diferentes (estruturados, semiestruturado e não estruturados).
- Pré-Processamento – Consiste em realizar a verificação da qualidade dados. Esta etapa consiste na limpeza de dados, ruídos e exceções serão removidos. A revisão, retirada de dados inconsistentes, incompletos ou não íntegros pertencem ao processo.
- Transformação – Significa em aplicar técnicas, visando a normalização, agregação, criação de novos atributos uteis nos dados para alcançar os objetivos

pretendidos. Esta transformação investiga atributos que retratem todas as características destes dados.

- Mineração de dados – Compreende – se nesta etapa a aplicação de algoritmos e técnicas para identificar padrões nos dados. Esta etapa pode ser vista como núcleo do processo de KDD.
- Interpretação e Avaliação – Consta em qualificar a atuação do modelo, ao final do processo de mineração de dados, ocorrendo a conservação do conhecimento descoberto. Os padrões enumerados devem ser entendidos e avaliados com base em análise de profissionais ou mesmo em comparação com dados coletados anteriormente.

Figura 2.23 – Etapas do processo KDD.



Fonte: SCHNEIDER (2003).

2.4.4 Análise exploratória de dados

A análise exploratória de dados (AED) é um termo bastante usado por profissionais de ciência da computação, ele tem como objetivo observar os dados previamente à aplicação de qualquer técnica estatística. Assim o analista alcança um entendimento básico de seus dados e das ligações existentes entre as variáveis empregadas.

Han e Kamber (2006) apud Camilo (2009, p.6) descrevem várias técnicas estatísticas de análise de dispersão (Quartiles, Variância) e de medida central (média, mediana, moda e faixa de valores) combinadas com gráficos (Histogramas, Frequência, Barra, BoxPlot, Dispersão) são usadas para a exploração dos dados.

Na AED é muito natural a análise descritiva, que de maneira detalhada possibilita ao cientista de dados adaptar-se aos dados, organizá-los e sintetizá-los de modo a obter as

informações necessárias do conjunto de dados para discutir as questões que o problema de *Data Science* está tentando solucionar.

A AED pode estar contida nas quatro primeiras etapas do KDD, e pode ser conceituada como a primeira e importante análise sobre os dados. Para efetuar a AED é decisivo conhecer tecnicamente o que seus dados representam e como eles são classificados.

2.4.5 Mineração de dados

Segundo Cabena et al. (1998) apud Costa (2012, p.3), Mineração de Dados é uma área interdisciplinar, mobilizando principalmente conhecimentos de análise estatística de dados, aprendizagem de máquina, reconhecimento de padrões e visualização de dados.

A mineração de dados envolve ajustar modelos ou determinar padrões a partir de dados observados. Os modelos ajustados desempenham o papel de inferidos conhecimentos: se os modelos refletem ou conhecimento interessante faz parte do conjunto geral, processo KDD iterativo onde subjetivo o julgamento humano é normalmente necessário.

A etapa de mineração de dados é considerada como o núcleo do KDD, e a etapa mais importante, na qual os dados que passaram pelo processo de pré-processamento e de transformação, serão analisados pelos algoritmos de mineração de dados, extraindo assim alguma informação relevante, podendo ser padrões entre atributos ou uma classificação que permite realizar uma análise.

Após realizada a etapa de mineração de dados os cientistas de dados, utilizarão essas informações ou padrões para realizar uma análise e buscar identificar um conhecimento a parti do trabalho realizado.

Durante o processo de mineração de dados, os resultados encontrados, podem não ser os esperados. Assim ocorre a possibilidade de executar o algoritmo novamente ou executar um outro algoritmo, avaliando os resultados novamente, até que se torne possível obter o resultado esperado. Assim é possível obter o “conhecimento” que é o objetivo esperado dentro do processo.

Para descobrir a relevância do conhecimento, é indispensável estabelecer metas bem definidas. No processo de descoberta de conhecimento as metas são definidas em função dos objetivos na utilização do sistema, podendo ser de dois tipos básicos: verificação ou descoberta (FAYYAD et al., 1996). Nos momentos que a meta é do tipo verificação, o sistema está restringido a verificar possibilidades definidas pelo usuário, ao passo que na descoberta o

sistema resolve novos padrões de forma autônoma. A meta do tipo descoberta, está ligada com as seguintes tarefas de mineração de dados: predição e descrição.

2.4.6 Tarefas de mineração de dados

As tarefas de mineração de dados são classificadas em função do tipo de aplicação, elas podem ser separadas em: Descritivas e Preditivas. As descritivas têm a função de extrair o conteúdo dos dados, ou seja, as propriedades que os dados possuem, permitindo um melhor conhecimento sobre os dados que serão trabalhados. Já as preditivas irão realizar previsões a partir dos dados obtidos, os dados são aplicados para realizar classificações. Através de um modelo pronto, gerado a partir de um algoritmo é possível utilizar os dados para realização de inferência, criar uma relação entre os atributos e os objetos contidos em conjunto de dados. Nessa etapa é possível gerar um novo conhecimento. Na fase descritiva busca-se entender os dados, enquanto na fase preditiva tem-se como objetivo a geração de conhecimento a partir dos dados.

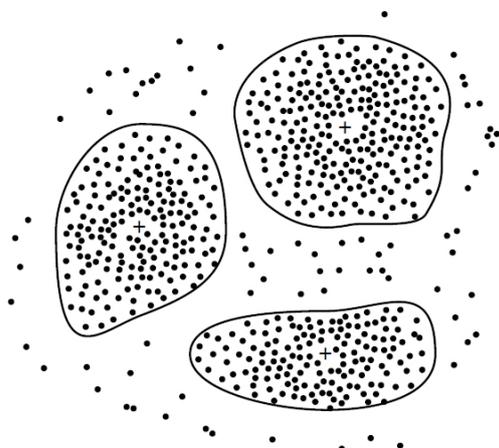
A principais tarefas são divididas em: análise descritiva de dados, predição (classificação e estimação), Análise de grupo, associação e detecção de anomalias.

A análise descritiva de dados, busca entender os dados aplicados para compreender as possibilidades de uso, permite também aprofundar no detalhamento desses dados. A análise descritiva geralmente apresenta uma possível leitura para os resultados encontrados, ela é também muito utilizada em conjunto com as técnicas de análise exploratória, para comprovar a influência de certas variáveis no resultado obtido (CAMILO, 2009).

Na tarefa de predição busca-se classificar ou estimar os dados a partir de outros dados conhecidos. Aplica-se os atributos de um objeto para separá-los em classes. No processo de predição, é fundamental que boa parte dos dados sejam rotulados manualmente, ou seja, a aprendizagem do modelo ocorrerá de forma supervisionada e dar-se-á utilizando um conjunto de treinamento com valores previamente conhecidos (COSTA,2012).

As tarefas de análise de grupo se assemelham com as tarefas de classificação, porém na análise de grupo não é possível identificar a que classe os dados pertencem, sendo assim, eles são agrupados em função das características dos atributos que eles possuem por similaridade, conforme Figura 2.24.

Figura 2.24 – Registro de agrupamento.



Fonte: CAMILO (2009).

A tarefa de associação procura entender as relações existentes entre os objetos que estão presentes nos atributos, ela prever a ocorrência de um item baseado na ocorrência de outros itens. Apresentam a forma: SE atributo X ENTÃO atributo Y (CAMILO, 2009). Os algoritmos clássicos de mineração de regras de associação derivam regras apenas conjuntivas, limitando-se a utilização do operador lógico AND (COSTA, 2012).

O objetivo da tarefa de detecção de anomalias é encontrar valores fora de parâmetros, tenta – se identificar objetos inconsistentes com o restante da base de dados que não estão de acordo com o comportamento normal, como por exemplo uma detecção de erro ou ruído.

2.4.6 Interpretação e Avaliação

Nesta etapa, serão examinados os resultados explicados na fase anterior, disposto conforme os objetivos prefixados, que devem possuir um entendimento claro dos objetivos, para que a tarefa de mineração da dados seja considerada um caso de sucesso.

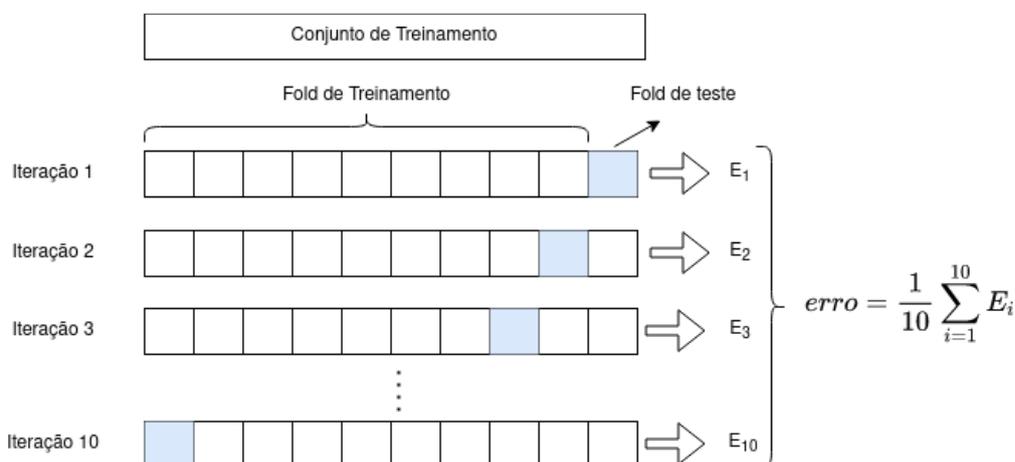
Segundo Kohavi (1995) apud Neto (2021, p.69), avaliar o desempenho de um modelo com o mesmo conjunto utilizado na sua construção, não fornece uma boa estimativa de como será a sua predição em dados nunca vistos, pois a estimativa será otimista. Uma forma de eliminar este problema é não utilizar todos os dados disponíveis para a construção do modelo. Esta é a ideia básica para uma classe de métodos de avaliação que serão descritos a seguir:

- *Hold-out*: Neste método, uma parte dos dados disponíveis é utilizada para avaliação do modelo, chamado conjunto de teste, e o restante é utilizado para construção do classificador, chamado conjunto de treinamento. Em geral, é

utilizado 1/3 dos dados disponíveis para o conjunto de teste e 2/3 para o conjunto de treinamento.

- *K-Fold-Cross-Validation*: Segundo Jain, Duin e Mao (1995) apud Neto (2021, p.70), os dados do conjunto de treinamento são divididos em k subconjuntos disjuntos chamados de *folds* e são construídos k modelos. Para cada modelo, um dos k subconjuntos é separado para teste e os outros k-1 conjuntos são utilizados para o treinamento, eliminando assim a dependência de como os dados foram divididos, pois todos os dados podem ser utilizados na construção e avaliação do modelo, o que fornece uma estimativa mais robusta de desempenho.
- *Leave-one-out*: pode ser definido como uma versão do método *K-Fold-Cross-Validation* levado ao extremo, onde o valor de k é igual ao número de padrões disponíveis. Desta forma, um elemento é utilizado como conjunto de teste em cada experimento. Em conformidade com Kearns e Ron (1999) apud Neto (2021, p.70) este método é indicado quando o conjunto de dados disponível é muito pequeno.
- *Bootstrap*: dado um conjunto de dados com N exemplos, é realizada uma amostra de mesmo tamanho N com replicação, ou seja, nesta nova amostra deve existir exemplos replicados. Este conjunto gerado por amostragem com replicação será o conjunto de treinamento. Os exemplos do conjunto original que não foram selecionados para compor o conjunto treinamento formam o conjunto de teste Kohavi (1995) apud Neto (2021, p.69).

Figura 2.25 – Divisão do conjunto de treinamento em *folds* para um *K Fold-Cross-Validation* com K = 10.



Fonte: NETO (2021).

2.5 REDES NEURAIS ARTIFICIAIS

2.5.1 Introdução a rede Neural.

As redes neurais foram desenvolvidas, originalmente, na década de 1940, pelo neurofisiologista McCulloch e pelo matemático Walter Pitts, da Universidade de Illinois, que dentro do espírito cibernético da época, publicaram um artigo intitulado “*A Logical Calculus of the Ideas Immanent in Nervous Activity*” (AGUIAR et al., 2007). Eles definiram desta forma o “neurônio formal” a partir dos resultados da neurobiologia e propuseram arquiteturas para a realização de funções lógicas (CAMPOS et al., 2004).

Uma rede neural artificial é um modelo computacional inspirado no funcionamento do cérebro humano Beale e Jackson (1990) apud Neto (2021, p.32).

O cérebro é um computador (sistema de processamento de informações) altamente complexo, não linear e paralelo. Tem a capacidade de organizar seus constituintes estruturais, conhecidos como neurônios, de modo a realizar certos cálculos (por exemplo, reconhecimento de padrões, percepção e controle motor) muito mais rápido do que o computador digital mais rápido que existe hoje (HAYKIN,2005).

De acordo com (HAYKIN,2005), uma rede neural é um processador distribuído massivamente paralelo, composto de unidades de processamento simples, que tem uma propensão natural para armazenar conhecimento experiencial e torná-lo disponível para uso. assemelha-se ao cérebro em dois aspectos:

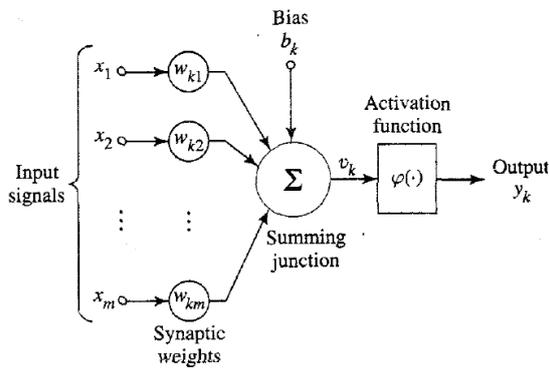
- O conhecimento é adquirido pela rede a partir de seu ambiente por meio de um processo de aprendizagem.
- As forças de conexão entre neurônios, conhecidas como pesos sinápticos, são usadas para armazenar o conhecimento adquirido.

O procedimento utilizado para realizar o processo de aprendizagem é denominado algoritmo de aprendizagem, cuja função é modificar os pesos sinápticos da rede de forma ordenada para atingir o objetivo de projeto desejado (HAYKIN,2005).

2.5.2 Neurônio

Um neurônio é uma unidade de processamento de informações fundamental para o funcionamento de uma rede neural. O diagrama de blocos da Figura 2.26 mostra o modelo de um neurônio, que forma a base para o projeto de redes neurais artificiais (HAYKIN,2005).

Figura 2.26 – Modelo de um neurônio não linear



Fonte: NETO (2021).

A ideia do neurônio artificial é simular o funcionamento do neurônio biológico. De uma forma extremamente simplificada, pode-se dizer que a comunicação entre os neurônios biológicos ocorre a partir de impulsos elétricos. Cada neurônio recebe uma determinada quantidade de estímulo de outros neurônios e a partir de uma determinada quantidade de estímulo recebido, o neurônio “dispara” um novo estímulo para outro neurônio. Desta forma, o neurônio artificial possui o seguinte objetivo: receber entradas de conjunto de unidades X, computar função sobre entradas e enviar resultado para conjunto de unidades Y (NETO,2021).

Um neurônio artificial possui duas propriedades: estado de ativação e função de ativação. Cada uma destas propriedades será descrita a seguir (NETO,2021):

- Estado de ativação: Este atributo define o valor recebido pelo neurônio. Ele envolve soma ponderada das conexões de entradas (w_{kj}) com o valor do neurônio correspondente.
- Função de ativação: Esse método é responsável por gerar o valor de saída do neurônio a partir do seu estado de ativação. Essa função seria equivalente ao “disparo” do neurônio biológico. Para neurônios de redes MLP, a função de ativação mais popular é a Sigmoide Logística.

Tipicamente, a faixa de amplitude normalizada para um neurônio é descrito como um intervalo de unidade fechada de 0 a 1 ou alternativamente de -1 a 1 (HAYKIN,2005).

O modelo do neurônio artificial da Figura 2.26 também inclui um *Bias*, denominado de b_k . O *Bias* b_k tem um efeito de incremento ou decremento o peso de entrada da função de ativação, podendo ser positivo ou negativo.

Matematicamente, pode-se descrever um neurônio pelas equações a seguir:

$$u_k = \sum_{j=1}^m w_{kj}x_j \quad (6)$$

$$y_k = \varphi(u_k + b_k) \quad (7)$$

Onde:

- x_1, x_2, \dots, x_m são sinais de entradas;
- $w_{k1}, w_{k2}, \dots, w_{km}$ são as sinapse de pesos do neurônio k ;
- u_k é a saída linear combinada gerado pelos sinais de entradas
- b_k é o *Bias*;
- $\varphi(-)$ é a função de ativação;
- y_k é o sinal de saída do neurônio.

2.5.3 Função de ativação

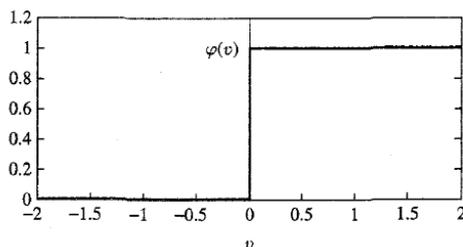
A função de ativação, representada por $\varphi(v)$, define a saída para o neurônio em termos de campo local induzido v . Existem três tipos básicos de função de ativação:

2.5.3.1 Função limite:

Este tipo de função de ativação é demonstrado na figura 2.27.

$$\varphi(v) = \begin{cases} 1 & \text{se } v \geq 0 \\ 0 & \text{se } v < 0 \end{cases} \quad (8)$$

Figura 2.27 – Função limite



Fonte: HAYKIN (2005).

A saída do neurônio K empregada como função de ativação é expressa como:

$$Y_K \begin{cases} 1 & \text{se } v_k \geq 0 \\ 0 & \text{se } v_k < 0 \end{cases} \quad (9)$$

Onde v_k é o campo local induzido do neurônio.

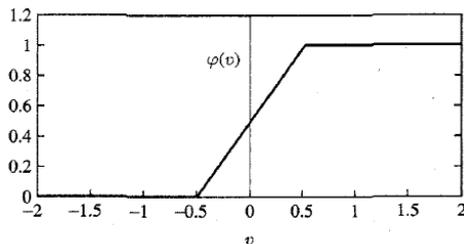
$$v_k = \sum_{j=1}^m w_{kj} x_j + b_k \quad (10)$$

Em engenharia, este formato de função limite é comumente conhecido como *Heaviside function*, ou seja, uma função especial e descontínua com valor zero quando o seu argumento é negativo e valor unitário quando o argumento é positivo.

2.5.3.2 Função Linear por partes:

A função Linear por partes é descrita na Figura 2.28.

Figura 2.28 – Função linear por partes



Fonte: HAYKIN (2005).

$$\varphi(v) = \begin{cases} 1, & v \geq 1/2 \\ v, & + 1/2 > v > -1/2 \\ 0, & v \leq -1/2 \end{cases} \quad (11)$$

O fator amplificador dentro da região linear assumido é unitário. Essa forma de função de ativação pode ser vista como um amplificador de aproximação não linear. As duas situações a seguir podem ser vistas de forma especial para uma função linear por partes:

- Um combinador linear surge se a região linear de operação está mantida sem operações dentro da saturação.
- A Função Linear por Partes transforma-se em uma função limite se o fator de amplificação para a região linear é feito infinitamente grande.

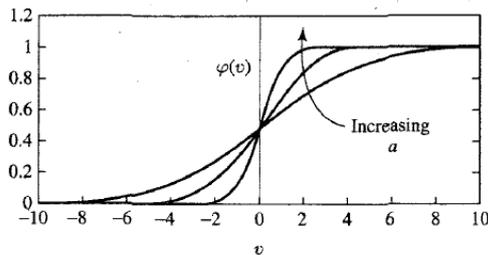
2.5.3.3 Função Sigmoide:

A função Sigmoide, cujo gráfico tem a forma de s é de longe a forma de ativação mais comum usado na construção das redes neurais artificiais. É definida como uma função estritamente crescente e exibe um balanço gracioso entre o comportamento linear e não linear. Um exemplo de função Sigmoide é a função logística, definida por (HAYKIN,2005):

$$\varphi(v) = \frac{1}{1+\exp(-av)} \quad (12)$$

Onde a é o parâmetro de declive para a função sigmoide. Pela variação do parâmetro a nós obtém-se a função Sigmoide para diferentes declives, como ilustrado na figura 2.29. Dentro de um limite, como parâmetro aproximado de declive infinito, a função Sigmoide torna-se uma simples a função de ativação limite.

Figura 2.29 – Função Sigmoide



Fonte: HAYKIN (2005).

As funções de ativação determinadas nas expressões 8 e 12 variam de 0 a 1. Esporadicamente, espera-se que a função de ativação varie de -1 a 1, nesse caso, a função de ativação é uma função ímpar do campo local induzido. Particularmente a função limite da equação 8 agora é definido como:

$$\varphi(v) = \begin{cases} 1, & \text{se } v > 0 \\ 0, & \text{se } v = 0 \\ -1, & \text{se } v < 0 \end{cases} \quad (13)$$

Normalmente declarada como função sinal. Para a forma correspondente de uma função Sigmoide, podemos usar a função tangente hiperbólica, definida por:

$$\varphi(v) = \tanh(v) \quad (14)$$

Essa expressão consente que uma função de ativação do tipo Sigmoide expresse valores negativos conforme estabelecido pela equação 14 e pode efetuar benefícios práticos sobre a função logística da equação 12.

2.5.3.4 Função ReLU:

De acordo com Glorot (2011) apud Araújo, (2019, p.20) a função Linear Retificada (Rectified Linear Unit, ReLU), foi introduzida pela primeira vez por Hahnloser et al. em 2000, em uma rede dinâmica. Em 2011, o uso da ReLU como uma não-linearidade foi mostrado para permitir o treinamento de redes neurais supervisionadas profundas sem a necessidade de pré-treinamento não supervisionado.

A ReLU é uma função de ativação definida apenas como a parte positiva de seu argumento, ou seja, sempre que houver valores negativos na sua entrada, esse valor será convertido para zero e àquele neurônio não será ativado, dessa forma, pode-se afirmar que apenas alguns neurônios são ativados, tornando a rede esparsa (ARAÚJO,2019).

Este tipo de função de ativação é demonstrado pela equação 15.

$$\varphi(v) = \max(0, v) \quad (15)$$

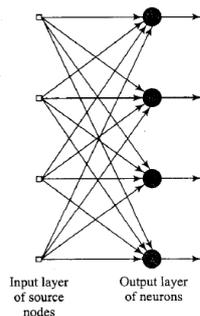
A ReLU, associadas às funções Sigmoide ou Tanh, permite um treinamento mais rápido e verdadeiro de arquiteturas mais acentuadas em conjuntos de dados grandes e relevantes, visto que, distinto das demais, a ReLU não aplica expoentes, ela efetua somente operações de comparação, adição e multiplicação.

2.5.4 Arquitetura de rede

A maneira pela qual os neurônios da rede neural são estruturados está intimamente ligado com os algoritmos de aprendizado usado no treinamento da rede. Nós pode-se então dizer que os algoritmos (regras) usados no projeto da rede são estruturados. Em geral, nós pode-se identificar três diferentes classes fundamentais para a arquitetura de rede (HAYKIN,2005):

Redes *feedforwards* de camada única – Dentro da rede neural em camadas os neurônios são organizados em camadas. Na forma simples para a rede em camadas, nós tem-se uma entrada para os nós de origem do projeto para uma camada de saída dos neurônios. Em outras palavras, esta rede é estritamente uma *feedforward* ou tipo acíclico. A figura 2.30, ilustra um caso de quatro nós como entradas e camadas de saída. Uma rede assim é conhecida como uma rede de camada simples, sendo designada “camada simples” referindo a camada de saída como nós de computação ou neurônios (HAYKIN,2005).

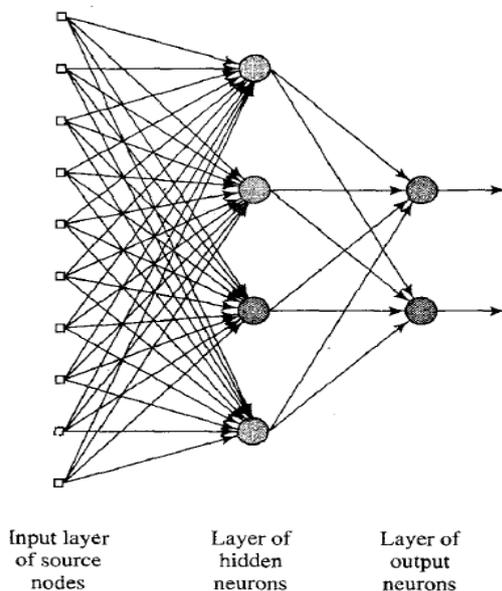
Figura 2.30 – Rede *feedforward* com uma única camada de neurônios.



Fonte: HAYKIN (2005).

Redes *feedforward* de múltiplas camadas – A segunda classe da rede neural *feedforward* distingue-se pela presença de uma ou mais camadas ocultas, cujos nós de computação são correspondentemente chamados de neurônios ocultos ou unidades ocultas. A função dos neurônios ocultos é intervir entre a entrada externa e a saída da rede de alguma maneira útil. Ao adicionar uma ou mais camadas ocultas, a rede permite extrair estatísticas de ordem superior. A habilidade dos neurônios ocultos para extrair estatística de ordem superior é particularmente válido quando o tamanho da camada de entrada é grande. A figura 2.31 ilustra a disposição de uma rede neural *feedforward* de múltiplas camadas com uma camada oculta simples (HAYKIN,2005).

Figura 2.31 – Rede *feedforward* com uma camada oculta e uma camada de saída.



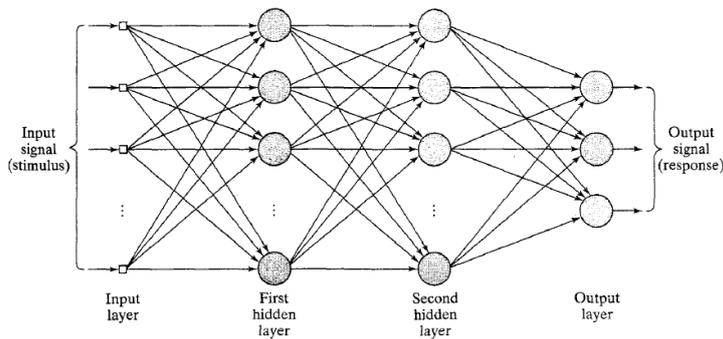
Fonte: HAYKIN (2005).

2.5.5 Multilayer Perceptron

Tipicamente, a rede consiste de um conjunto de unidades de sensores que constitui uma camada de entrada, uma ou mais camadas ocultas para a computação dos nós e a camada de saída. O sinal propaga pela rede dentro de uma direção, camada por camada. Essas redes são comumente conhecidas como *multilayer perceptrons* (MLP) (HAYKIN,2005).

A Figura 2.33 mostra um diagrama para representar uma rede *multilayer perceptron* com duas camadas ocultas e uma camada de saída. A rede mostrada está totalmente conectada. Isto significa que o neurônio dentro de qualquer camada da rede está conectado em todos os nós e neurônios das camadas anteriores

Figura 2.32 – Rede MLP com duas camadas ocultas



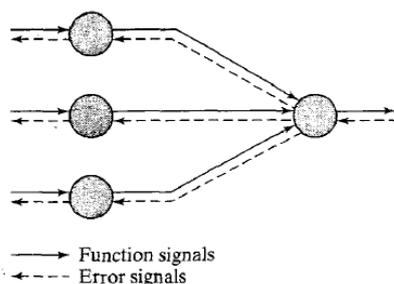
Fonte: HAYKIN (2005).

MLP tem sido aplicado com sucesso para resolver algumas dificuldades e diversos problemas, ao treiná-los de forma supervisionada com o algoritmo conhecido como algoritmo de retropropagação de erro ou *error back-propagation algorithm*. Este algoritmo é baseado na regra de aprendizagem de correção de erros ou *error correction learning rule* (HAYKIN,2005).

Basicamente, aprendizagem de correção de erros consiste de duas fases através da diferença de camada de redes: *a foward pass and a backward pass*. Na *forward pass*, um vetor de entrada é aplicado no nós sensoriais da rede e isto propaga um efeito através da rede camada por camada. Finalmente, um conjunto de saída é produzido como uma saída atual de resposta da rede. Durante o *backward pass*, o peso das sinapses é ajustado de acordo com a regra de correção de erro. Especificamente, a resposta da rede é subtraída de uma resposta desejada para geração do sinal de erro. Esse sinal de erro é então propagado para trás através da rede em direção oposta a direção das sinapses de correção, recebendo o nome de retropropagação de erro (HAYKIN,2005).

Um sinal de erro tem origem na saída do neurônio e propaga-se para trás (camada por camada) através da rede conforme Figura 2.33. Se refere a ele como um “sinal de erro” porque seu cálculo por cada neurônio da rede envolve uma função dependente de erro de uma forma ou de outra (HAYKIN,2005).

Figura 2.33 – Fluxo básico de direção do sinal



Fonte: HAYKIN (2005).

2.5.5 Algoritmo de retropropagação

A Figura 2.32 apresenta uma arquitetura para uma rede *Multilayer Perceptron*. O fluxo de sinal gráfico para a aprendizagem de retropropagação incorpora a fase de avanço e de recuo para a computação envolvida no processo de aprendizagem (HAYKIN,2005).

Como mencionado anteriormente a atualização sequencial dos pesos é o método preferido para a implementação não linear do algoritmo *back-propagation*. Nesse método de operação, os algoritmos circulam através de amostras de treinamentos $\{x(n), d(n)\}_{n=1}^N$, como segue (HAYKIN,2005):

- Inicialização. Assumindo que a informação prioritária está disponível, escolha o peso sináptico e os limites de uma distribuição uniforme cuja a média é zero e cuja variância é escolhida para fazer com que o desvio padrão dos campos locais induzidos dos neurônios fique na transição entre as partes linear e saturada da função de ativação Sigmoide.
- Apresentações de exemplos de treinamento. Apresente à rede uma amostra de exemplos de treinamento. Para cada exemplo do conjunto, ordenado de alguma forma, execute a sequência de cálculos para frente e para trás descrita nos pontos a seguir.
- Cálculo direto. Permite que um exemplo de treinamento na amostra seja apresentado por $(x(n), d(n))$, com o vetor de entrada $x(n)$ aplicado à camada de entrada dos nós sensoriais e o vetor de resposta desejado $d(n)$ apresentado a camada de saída de nós de computação. Calcula os campos locais induzidos e o sinal de função, avançando pela rede, camada por camada. O campo local induzido $v_j^l(n)$ para o neurônio j na camada l é:

$$v_j^l = \sum_{i=0}^{m_0} w_{ji}^{(l)}(n) y_i^{(l-1)}(n) \quad (16)$$

Onde:

$y_i^{(l-1)}(n)$ é o sinal de saída (função) do neurônio i da camada anterior l-1 e interação n $w_{ji}^{(l)}(n)$ é a sinapse de peso do neurônio j na camada l que é alimentada pelos neurônios i da camada l-1. Para $i=0$, nós tem-se $y_0^{(l-1)}(n) = 1$ e $w_{j0}^{(l)}(n) = b_j^{(l)}(n)$ é as bias aplicado no neurônio J na camada l. Assumindo o uso da função sigmoideal, o sinal de saída do neurônio j na camada l é:

$$y_j^{(l)} = \varphi_j(v_j(n)) \quad (17)$$

Se o neurônio J na primeira camada oculta for ($l=1$), define-se $y_j^{(0)}(n) = x_j(n)$. Onde, $x_j(n)$ é o elemento $j^{(th)}$ para o vetor $x(n)$. Se o neurônio J estiver na camada de saída, por exemplo $l=L$, quando L se refere com a profundidade da rede, defini-se $y_j^{(0)} = o_j(n)$.

Cálculo do sinal de erro e expresso pela equação (14)

$$e_j(n) = d_j(n) - o_j(n) \quad (18)$$

Onde $d_j(n)$ é o elemento $j^{(th)}$ para a resposta do vetor $d(n)$ desejado.

- Computação retroativa. calcular os δ s (ou seja, gradientes locais) da rede definida por:

$$\delta_j^{(l)}(n) = \begin{cases} e_j^{(l)}(n)\varphi_j'(v_j^{(l)}(n)) & 1 \\ \varphi_j'(v_j^{(l)}(n))\sum_k \delta_k^{(l+1)}(n)w_k^{(l+1)}(n) & 2 \end{cases} \quad (19)$$

1 - Para o neurônio j na camada de saída L.

2 - Para o neurônio j na camada de saída l.

onde em $\varphi_j'(\cdot)$ denota a diferenciação em relação ao argumento. Os Ajustes dos pesos sinápticos da rede na camada l de acordo com a regra delta generalizada é mostrada na equação (16).

$$w_{ji}^{(l)}(n+1) = w_{ji}^{(l)}(n) + \alpha [w_{ji}^{(l)}(n-1)] + \eta \delta_j^{(l)}(n) y_i^{(l-1)}(n) \quad (20)$$

Onde η é a taxa de aprendizagem e α é a constante momentânea.

- Interação. Itera os cálculos para frente e para trás nas etapas citadas acima, apresentando novos exemplos de época de treinamento à rede até que o critério de parada seja atendido.

2.5.6 Hiperparâmetros da Rede

Denominam-se “hiperparâmetros” as variáveis que determinam a estrutura da rede neural ou a forma como a mesma será treinada (RADHAKRISHNAN, 2017). Os hiperparâmetros são estipulados como valores iniciais e aleatórios da rede. Pode-se destacar duas categorias aplicadas aos hiperparâmetros a Estrutura de Rede e o Treinamento de Rede. Na Estrutura de Rede defini-se o número de camadas, número de neurônios em cada camada, método de inicialização dos pesos, vieses e a funções de ativação. No Treinamento da Rede pode-se ajustar a taxa de aprendizado, número de épocas, tamanho de lote e o coeficiente de regularização.

É por meio da definição de hiperparâmetros que se torna possível otimizar uma rede neural artificial dentro do contexto de aplicação. O objetivo é maximizar uma ou mais métricas de interesse, garantindo que a rede não apresente subajustes ou sobreajustes respectivamente, *under* ou *overfitting* (AMARAL, 2021).

O propósito de todo classificador é formar um modelo genérico, que pode acertar ou errar uma determinada instância no momento da execução do algoritmo, mas na maioria dos casos o modelo vai acertar, é isso que se deseja do modelo gerado pelo algoritmo. No decorrer do processo de formação do modelo, pode ocorrer o surgimento do *underfitting* e do *overfitting*.

O modelo de ML que apresenta o *underfitting* não alcança um bom ajuste aos dados de treinamento e, portanto, também não realiza uma generalização adequada dos novos dados. Esse tipo de modelo não aprende com o padrão dos dados. As possíveis causas do *underfitting* são:

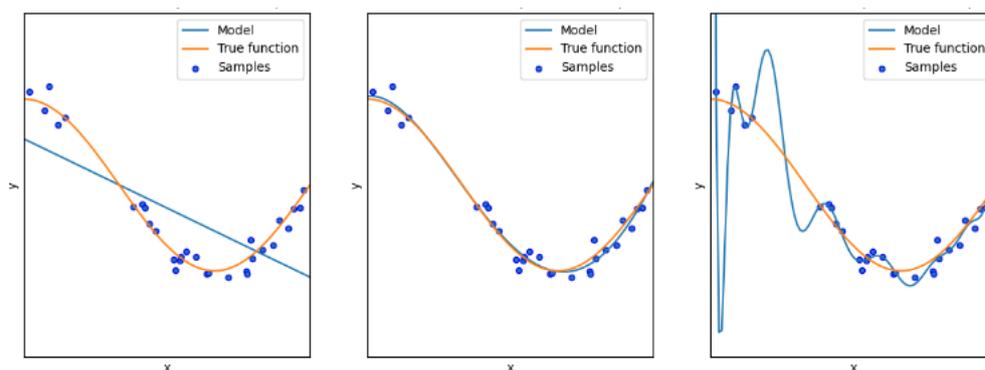
- Simplicidade do modelo;
- Conjunto de dados insuficientes;
- Seleção incorreta dos atributos;
- Falta de ajuste dos hiperparâmetros.

O *overfitting* funciona bem com os dados de treinamento, mas tem o desempenho ruim em dados de teste ou de execução do modelo.

Além da complexidade exagerada do modelo de treinamento e dados que possuem erros, inconsistências ou desvios em relação aos demais dados, tem-se alguns fatores que podem gerar o *overfitting*, como:

- O Tamanho deficiente do conjunto de dados;
- Seleção incorreta dos atributos;
- Falta de validação cruzada.

Figura 2.34 – Apresentação dos modelos com *underfitting*, ajuste genérico e *overfitting*.



Fonte: (PEDREGOSA et al., 2011).

3 MATERIAIS E MÉTODOS

Neste trabalho, foi realizado um estudo de caráter quantitativo através da aquisição de dados, com representação numérica, que facilita sua comprovação. De natureza aplicada o estudo de caso tenta dispor de meios capazes de proporcionar uma otimização nos sistemas de transmissão automática dos veículos motores, tendo como recurso de fundamentação teórica, as revistas acadêmicas e científicas acessíveis de forma on-line, com objetivo exploratório, reúne e compara meios de se executar um algoritmo aplicado ao aprendizado de máquina.

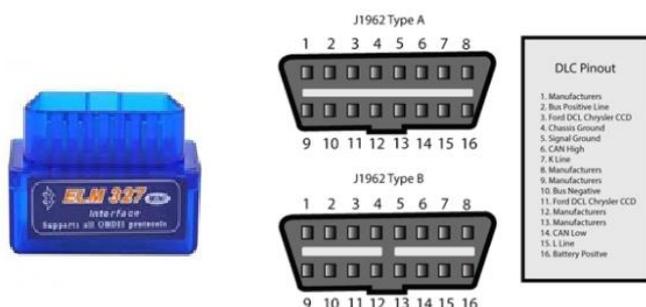
3.1 VERIFICAÇÃO DO CONJUNTO DE DADOS

3.1.1 Dispositivo de coleta de dados

As circunstâncias aplicadas no alcance de dados de veículos aplicados por pessoas comuns e em trajetos do dia a dia, é o primeiro fator a ser apontado. Por isso, foi essencial efetivar uma busca por dispositivos de hardware preparados para conectar com a interface DLC e também buscar por softwares capazes de remeter comandos OBD-II para um dispositivo de hardware, obter as respostas e armazená-las de forma apropriada.

Foi determinado que haveria a exigência de pesquisar sistemas em diversas plataformas com a aptidão citada e que a conexão padrão a ser solicitada nos dispositivos seria a Bluetooth, devido à origem da aplicação. Geralmente os dispositivos habilitados a se interligar com a interface DLC são conhecidos como *scanner bluetooth* e utilizam o microcontrolador PIC 18F25K80. Através desse *scanner bluetooth* automotivo, demonstrado na Figura 3.1 instalado junto a interface DLC, foi realizado a leitura e a transmissão dos dados aplicados nessa pesquisa.

Figura 3.1 – Scanner *bluetooth* com conector OBD2.



Fonte: Adaptado de CARRO & TÉCNICA TREINAMENTOS AUTOMOTIVOS (2023).

Esses dispositivos são, geralmente, constituídos por: um conector DLC macho; uma placa com microcontrolador; software pré-instalado para comunicação OBD-2; e transmissor *bluetooth*. Trata-se de um dispositivo que é capaz de enviar comandos a ECU, receber a resposta, traduzi-la para linguagem de alto nível (próxima da linguagem humana) e enviar essa resposta via *bluetooth* para dispositivo previamente pareado (BARRETO, 2018).

3.1.2 Acesso aos dados veicular

Hoje, existem alguns aplicativos de smartphone que realizam a leitura das informações veicular via sistema OBD2. Esses aplicativos trazem informações úteis sobre como o veículo está se comportando, tornando-se uma ferramenta útil no processo de aquisição de dados.

O Torque Pro, demonstrado na figura 3.2, foi o aplicativo escolhido para a realização desse trabalho. O aplicativo Torque é um instrumento de desempenho e diagnóstico OBD2 que pode ser executado no sistema Android e iOS que sustenta o acesso aos diferentes sensores de controle do motor, permitindo também visualizar e limpar códigos de falhas.

Por meio de um smartphone conectado, via *bluetooth*, a central eletrônica, o aplicativo Torque que utiliza o plugin *Realtime Charts*, recolheu os dados empregados nesse estudo.

Figura 3.2 – Aplicativo Torque.



Fonte: Imagem extraída do software Torques® em 29 out 2023.

3.1.3 Obtenção dos dados

A obtenção dos dados, que foi realizado inicialmente para adquirir informações e apresentar um modelo de definição de perfis de estilo de condução apresentado pelos condutores e carga aplicada ao veículo.

O aplicativo Torque realiza o armazenamento dos dados durante a realização do trajeto em formato de planilha. Os dados são apresentados de forma visual, de acordo com a figura 3.3. Esse arquivo, apresenta-se no .csv (*Comma-separated values*), possui em sua primeira linha os títulos para cada “coluna” (atributos) e nas linhas subsequentes as coletas individuais (instâncias), que são realizadas a cada 2 segundos. Cada coluna representa uma informação oriunda da ECU e possui uma relação específica com o funcionamento do automóvel.

Figura 3.3 – Dados obtidos por meio do aplicativo Torque.

Sample Number	Sample Time	Torque [Nm]	Speed (OBD) [km/h]	Turbo Boost & Vacuum Gauge [psi]	Relative Accelerator Pedal Position [%]	Accelerator Pedal Position [%]
0.0	15:50:34.0	0.0	0.0	-10.298	0.0	0.02
2.0	15:50:36.0	0.0	0.0	-10.298	0.0	0.02
4.0	15:50:38.0	0.0	0.0	-6.527	0.0	0.01
6.0	15:50:40.0	0.953	1.0	-6.817	0.0	0.03
8.0	15:50:42.0	6.77	4.0	-6.962	0.0	0.03
10.0	15:50:44.0	11.635	9.0	-6.672	0.0	-0.02
12.0	15:50:46.0	17.587	13.0	-6.962	0.0	-0.01
14.0	15:50:48.0	17.985	18.0	-4.206	0.0	-0.12
16.0	15:50:50.0	34.448	24.0	-4.351	0.0	0.12
18.0	15:50:52.0	26.285	27.0	-8.267	0.0	-0.03
20.0	15:50:54.0	26.285	28.0	-5.221	0.0	0.0
22.0	15:50:56.0	16.03	30.0	-3.771	0.0	-0.03
24.0	15:50:58.0	29.323	34.0	-4.206	0.0	-0.04
26.0	15:51:00.0	19.515	35.0	-7.107	0.0	0.04
28.0	15:51:02.0	17.914	37.0	-6.527	0.0	0.0
30.0	15:51:04.0	17.914	37.0	-8.122	0.0	-0.01
32.0	15:51:06.0	27.15	40.0	-2.176	0.0	-0.02
34.0	15:51:08.0	35.792	43.0	-1.885	0.0	0.12
36.0	15:51:10.0	22.923	46.0	-2.031	0.0	-0.05
38.0	15:51:12.0	18.504	48.0	-2.901	0.0	0.04
40.0	15:51:14.0	16.895	49.0	-5.947	0.0	0.08
42.0	15:51:16.0	0.0	0.0	-4.496	0.0	-0.11

Fonte: Imagem extraída do software Microsoft Excel® em 29 out 2023.

É importante destacar que o significado de cada rótulo (atributo) está relacionado com a sua consequência no comportamento do automóvel. Com a função de assessorar nesse aspecto, especialistas foram entrevistados para assessorar essa pesquisa. Os profissionais são instrutores de educação profissional atuantes na área de veículos automotivos da cidade de Resende, com prática no mercado, inclusive na chefia de equipes de manutenção de marcas tradicionais como Volkswagen, Chevrolet e Nissan em concessionárias autorizadas da cidade. Seus aportes foram imprescindíveis, pois permitiu ter visão do funcionamento do motor

baseado nos sensores e atuadores aplicados em um automóvel e também possibilitou uma melhor compreensão relacionada a atribuição dos sensores automotivos e sua interdependência.

A melhor compreensão sobre os atributos e suas propriedades, pode auxiliar na compreensão dos resultados em função do uso das redes neurais artificiais. Tabela 3.1 demonstra, os significados, os nomes, formatos, exemplos e unidades dos valores recebidos via OBD-II para cada atributo.

Quadro 3.1 –Relação de atributos adquiridos via *scanner bluetooth*.

Variáveis precedentes do veículo		
Atributo	Significado	Unidade
Torque	Representa a força que o motor do carro é capaz de gerar. Essa força é gerada pelo movimento dos pistões.	Nm
Sensor de Velocidade	O Sensor de Velocidade é um formador de pulsos, que provê um sinal elétrico correspondente a velocidade do automóvel.	Km/h
Sensor MAP (Sensor de Pressão Absoluta)	Ele é um importante componente da ECU dos veículos e, está localizado no cofre do motor e conectado ao coletor de admissão, por meio de uma mangueira.	PSI
Sensor de aceleração	Os acelerômetros atuam medindo os movimentos de pequenas estruturas internas provocados pela aceleração do sensor em uma direção específica. Eles medem em metros por segundo ao quadrado (m/s ²) ou em forças G (g).	g
Potência nas rodas	A potência da roda é a mensuração da potência com o sistema de transmissão conectado e é efetuado com o motor do carro e todos os sistemas funcionando.	hp
Sensor de rotação	RPM ou Revoluções por Minuto em carros é uma expressão usada para apresentar o número de rotações efetuadas pelo virabrequim do veículo por minuto. Um medidor de rotação determina quantas voltas ou rotações ocorrem dentro do motor por minuto.	rpm
Sensor de ângulo	Sensores de inclinação detectam o ângulo de orientação de um objeto. O Sensor pode ser utilizado na horizontal para detectar inclinação e na vertical para vibração.	°
Sensor de posição do acelerador	A posição do acelerador controla a quantidade de ar que circula para o coletor de admissão do motor; a quantidade de ar é diretamente proporcional a posição do acelerador.	%

A aquisição de dados para o treinamento da rede, compõem-se em três etapas distintas, onde forma feitas as aquisições de dados para cada etapa. Cada etapa representa um comportamento diferente do condutor, que identifica o modo da condução econômico, normal

e esportivo. A carga aplicada ao veículo é definida pelos aclives e declives encontrados ao longo do trajeto realizado pelo veículo.

A base de dados definitiva para treinamento e teste, possui um total de 2.475 instâncias. As instâncias apresentadas são resultantes das aquisições de dados realizado pelo aplicativo Torque junto a ECU, via *scanner bluetooth*, a cada 2 segundos. Essas instâncias foram adicionadas os atributos classes, para definição do modo de condução e da determinação da carga do veículo.

A base subsequente de dados, com seus 8 atributos é, portanto, uma representação inicial do uso do veículo em trajetos urbanos e trajetos rodoviários rotineiros e perante condições reais de trânsito. Seu desempenho tem foco nos dados fornecidos pelos sensores e um vínculo firme com o motor do veículo, visto que os atributos estão relacionados unicamente com seus componentes.

3.1.3 Alinhamento dos dados

Técnicas de limpeza de dados foram utilizadas para solucionar problemas presentes na base criada. Existiam no conjunto de dados, algumas instancias que não possuíam nenhum valor, nessa situação o problema foi resolvido realizando a eliminação da instância do conjunto de dados. Dados anormais ou ruidosos, também foram abordados da mesma forma, sendo excluídos.

Logo após, os dados foram categorizados entre trajeto urbano e trajeto rodoviário, foram aplicados métodos descritivos (agrupamento de dados) para identificação dos grupos.

Para a realização das tarefas citadas foi preciso antes converter os arquivos no formato .csv para o formato .xlsx que é o formato de um arquivo de planilha Excel, conforme apresentado na figura 3.4. Para isso foi utilizado o conversor online Convertio, que suporta diversos tipos de arquivo, inclusive os arquivos aplicados nesse estudo.

3.1.4 Transição dos dados

Aplicando o conjunto de dados devidamente limpo e relacionado, as instâncias foram fragmentadas e atributos classes foram atribuídos posteriormente a cada conjunto fragmentado, para definição do modo de condução e da determinação da carga do veículo.

Figura 3.4 – Conversor .CSV para .XLSX



Fonte: Imagem extraída do software Convertio® em 02 nov. 2023.

Dados que representam instâncias limites entre os diferentes modos de condução e as diferentes cargas suportadas pelo veículo também foram excluídas, exercendo a função de um filtro, com o objetivo de determinar de forma apropriada as instâncias que definem as diferentes classes que se pretende classificar.

Com a conclusão dessas tarefas, realiza-se o processo contrário de conversão dos arquivos, alterando os dados do formato .xlsx para o formato .csv, utilizando o conversor online Convertio novamente. O formato .csv é o formato adequado de dados para utilização do algoritmo de ML.

3.1.5 Definição dos atributos classe

As regras escolhidas para determinar o desejo do condutor formam ECO, NORMAL e ESPORTE, determinando a vontade do motorista em acelerar pela posição do pedal do acelerador.

Para determinação do parâmetro de carga do veículo, foram utilizadas as variações de carga do veículo nos momentos de aclave, trechos planos e declive, durante seu trajeto nas ruas e rodovias sendo denominados respectivamente como C1, C2 e C3.

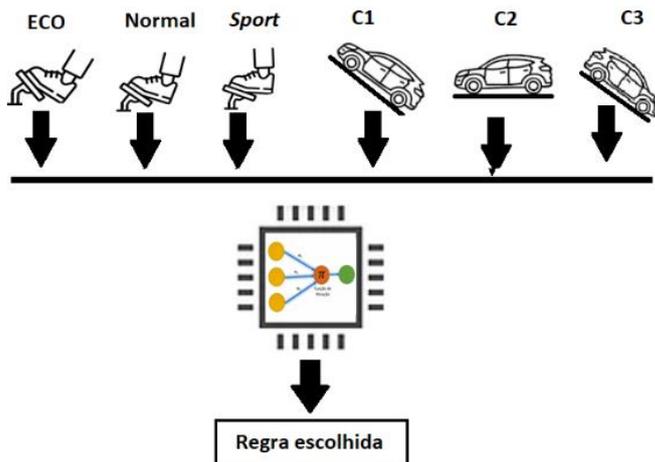
As regras de passagem de velocidade auto adaptativas foram definidas como:

- Regra ECO – Permite obter economia no consumo de combustível;
- Regra NORMAL – Apresenta um comportamento um pouco mais esportivo que a regra ECO;
- Regra *SPORT* – Disponibiliza um comportamento esportivo para o veículo;
- Regra C1 – Aplica-se essa regra a grandes aclives e muita carga.

- Regra C2 – Aplica-se essa regra a aclives pequenos ou perfil plano de estrada e pouca carga;
- Regra C3 – Aplica-se essa regra a identificação de todos os tipos de declives

A Figura 3.5 ilustra as regras de passagens adaptativas aplicadas.

Figura 3.5 – Regras de passagens adaptativas



3.1.6 Execução do algoritmo

Para a extração de informações oriundas dos dados, aplica-se a linguagem de programação Python através do software *PyCharm* e o módulo *Scikit-learn*.

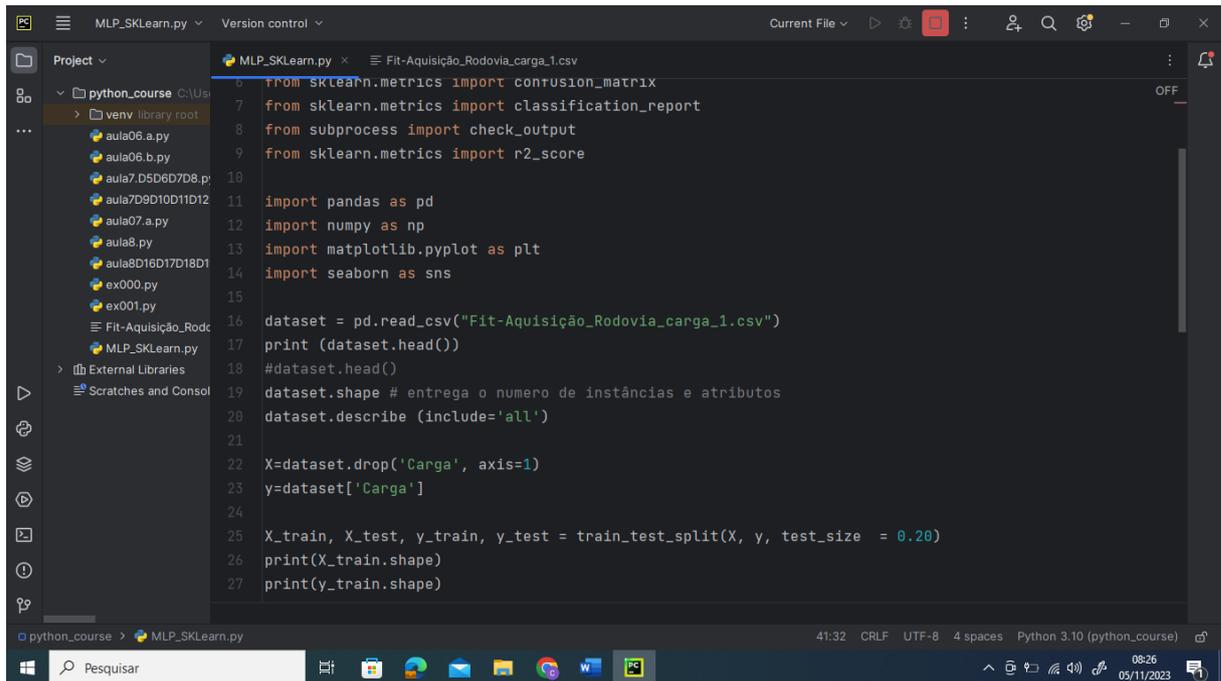
O *Python* é uma linguagem de programação amplamente usada em aplicações da Web, desenvolvimento de software, ciência de dados e *machine learning* (ML). Os desenvolvedores usam o *Python* porque é eficiente e fácil de aprender e pode ser executado em muitas plataformas diferentes. O *software Python* pode ser baixado gratuitamente, integra-se bem a todos os tipos de sistema e agiliza o desenvolvimento (AMAZON WEB SERVICE, 2023).

A linguagem Python tem vários casos de uso no desenvolvimento de aplicações, e pode ser aplicada em pesquisa que possui temas relacionados a ciência de dados e (ML).

Para a realização do nosso trabalho foi utilizado a IDE *integrated development environment PyCharm*. Ele tem uma edição gratuita da comunidade, que é adequada para pequenas aplicações Python, e uma edição profissional paga, que é adequada para criar aplicações Python de grande escala

A (IDE) é um software que oferece aos desenvolvedores as ferramentas necessárias para escrever, editar, testar e depurar código em um só lugar.

Figura 3.6 – Ambiente de desenvolvimento integrado do PyCharm.



Fonte: Imagem extraída do software PyCharm® em 05 nov. 2023.

O *Scikit-learn* é um módulo Python que integra uma ampla gama de algoritmos de aprendizado de máquina de última geração para problemas supervisionados e não supervisionados de média escala. Este pacote se concentra em levar o aprendizado de máquina a não especialistas usando uma linguagem de alto nível de uso geral. A ênfase é colocar facilidade de uso, desempenho, documentação e consistência de API. Possui dependências mínimas e é distribuído sob licença *Berkeley Software Distribution* (BSD) simplificada, licença de código aberto, incentivando seu uso tanto em ambientes acadêmicos, quanto uso comercial. (PEDREGOSA et al., 2011).

Geralmente o módulo *SciKit-Learn* não é usado sozinho. Juntamente com o módulo *Scikit_Learn*, tem-se a aplicação dos módulos *Numpy*, *Pandas* e *Matplotlib*. A Biblioteca *Numpy* é um módulo numérico do *Python*, usado para operações matemáticas. O módulo *Pandas* fornece determinadas estruturas Python para organização de dados. O módulo *Matplotlib* ajuda na visualização dos dados e resultados obtidos nas atividades de aprendizado de máquina.

Na tarefa de realizar o treinamento da rede, foi utilizado a IDE *PayCharm*, através do módulo *Scikit-learn*, os dados coletados através *scanner bluetooth*, devidamente limpos e analisados, foram inseridos no algoritmo. Dentro do algoritmo foi possível separar o conjunto de dados em dados de treino e teste na proporção de 80% para 20%, respectivamente. A escolha

dos hiperparâmetros foi realizada de forma empírica, alterando-se os valores manualmente, um de cada vez, e foi-se computando os melhores resultados, as funções de ativação escolhidas que reproduziram os melhores resultados foram a Sigmoide e a Tangente Hiperbólica. A pesquisa empírica apesar de não ser eficiente em tempo, possibilitou encontrar valores adequados ao problema. Realizou-se o emprego de um conjunto de instruções que definem um classificador *Perceptron* multicamadas. Através do código, verificou-se a acurácia utilizando a matriz de confusão.

Na tarefa de realizar a execução do modelo criado pela rede neural artificial, uma nova aquisição de dados através do *scanner bluetooth* foi executada, com o mesmo método, com as condições de contorno análoga a dos dados adquiridos para treinamento. Na etapa de execução os novos dados adquiridos para teste são submetidos ao modelo que avalia cada atributo, o computador que avalia esses atributos irá então fazer a previsão, ou seja, o prognóstico das condições em que o veículo estará submetido.

3.1.8 Rede Neural Artificial utilizada

A RNA apresentada consiste em uma rede *Feed Forward* com o algoritmo de aprendizado tipo *backpropagation*, aplicando em seu treinamento a troca de informações entre a ECU e a TCU, que estão conectados à rede denominada *Controller Area Network (CAN)*.

O treinamento da RNA aplica um conjunto de dados correspondente a uma amostra da aquisição de dados do motor através da ECU para os sinais de entrada resultando uma saída para o sistema, para essa finalidade é feito a mineração de dados, que é definida como o processo de descoberta de padrões nos dados (WITTEN et al., 2016). O processo de mineração de dados detecta os dados para que a rede alcance o desempenho esperado. Como consequência a RNA produzirá valores de saídas semelhantes ao conjunto de dados apresentados no treinamento para predizer a melhor estratégia de controle para mudança de marcha aplicada pela TCU.

3.1.9 Aplicação do método de controle

A partir da seleção realizada pela rede neural, propõem-se que a TCU irá aplicar os parâmetros do condutor e os parâmetros de carga do veículo, efetuando a escolha da regra de condução em função de um nível de prioridade aplicado, demonstrado na Tabela 3.2. Esta tabela

ilustra quais são as possibilidades existentes para aclive e declive, conforme a forma de conduta do motorista ou com a ação adequada/esperada.

Tabela 3.1 – Prioridades de utilização das regras de funcionamento.

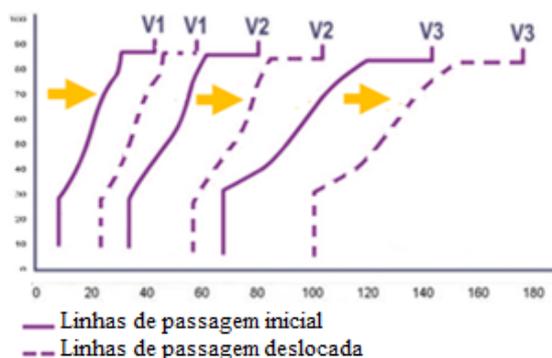
CONDIÇÕES DE CIRCULAÇÃO DO VEÍCULO			
Condições de utilização do condutor	Movimento em declive	Circulação em plano	Movimento em aclive forte
Regra ECO	C3	ECO	C1
Regra MÉDIA	C3	NORMAL	C1
Regra SPORT	SPORT	SPORT	C1

O movimento perfeito de variação da linha de mudança de marcha, nos padrões de passagem aplicadas em inclinação com aclives íngremes, provoca uma redução na marcha para evitar a falta de força motriz aplicada ao veículo. Em uma condução com aclives suaves é possível dirigir em marcha mais alta, mantendo a força motriz suficiente para que o veículo não altere a sua velocidade, nessas condições, procura-se entregar a TCU o melhor padrão de passagem de velocidade, evitando assim ações de controles inconveniente para os motoristas.

Recomenda-se que a linha de mudança de marcha, mova-se para manter a força motriz correta proporcionalmente à inclinação da estrada, como visto na Figura 3.7.

Figura 3.7 – Alteração dos padrões de passagem.

Alteração dos padrões para o modo esportivo ou grandes aclives



Alteração dos padrões para um modo mais moderado de direção e pequenos aclives



Em algumas condições, a mudança de velocidade às vezes não corresponde à intenção do motorista porque define-se vários estilos de direção que atendem, situações de condução distintas. Por exemplo, na condução com velocidades mais baixas, mesmo o motorista pretendendo economizar combustível, não se aplica uma regra de passagem econômica em aclives, é necessário um padrão de mudança que proporcione aos condutores mais força motriz para a aceleração.

3.1.10 O automóvel utilizado

Para realização deste experimento foi utilizado um automóvel Honda Fit, modelo 2014 (Figura 3.7) com motor 1.5, flex (Figura 3.7). O automóvel em foco foi selecionado por conveniência, em função da disponibilidade. Esse modelo foi comercializado para a população e possui aspectos comuns a muitos outros modelos dessa categoria, o que fortalece a possibilidade de utilização desse ensaio em outros veículos comumente utilizados. A ficha técnica do veículo é descrita na Tabela 3.1.

Figura 3.7 - Modelo utilizado – Honda Fit 2014.



Tabela 3.2 – Ficha técnica Honda Fit.

FICHA TÉCNICA HONDA FIT EX 1.5 (flex.) - AUTOMÁTICO		
Motorização	1.5	
Combustível	Álcool	Gasolina
Potência (cv)	116	115
Torque (kgf.m)	14,79	14,79
Câmbio	Automática com modo manual de 5 marchas	
Tração	Dianteira	
Altura (mm)	1.535	
Largura (mm)	1.695	
Comprimento (mm)	3.900	
Peso (Kg)	1.160	
Tanque (L)	47	
Entre eixos (mm)	2.500	
Ocupantes	5	

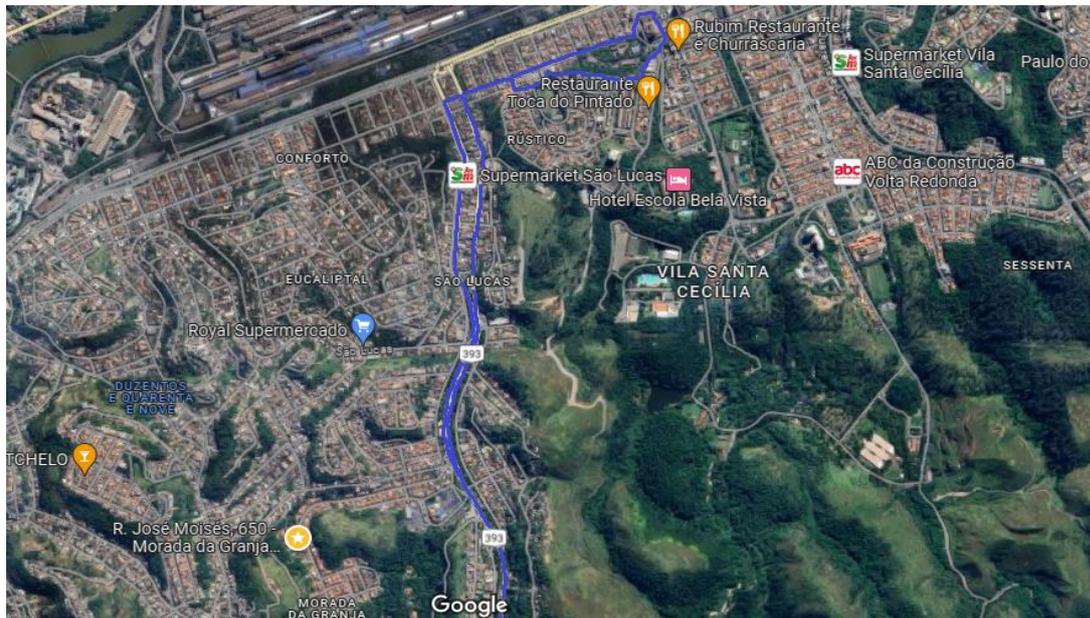
Fonte: Manual do proprietário Honda FIT 2013.

3.1.9 O Trajeto

A pesquisa foi realizada em um trajeto que contém trechos de rodovias e trechos urbanos, entre as cidades de Barra Mansa e Volta Redonda no estado do RJ. Devido a características de relevo dessas cidades, os dados colhidos entregam condições de condução em trechos planos e trechos de aclave e declives, atendendo dessa forma os requisitos necessários para determinação do modo de direção e da carga aplicada ao veículo.

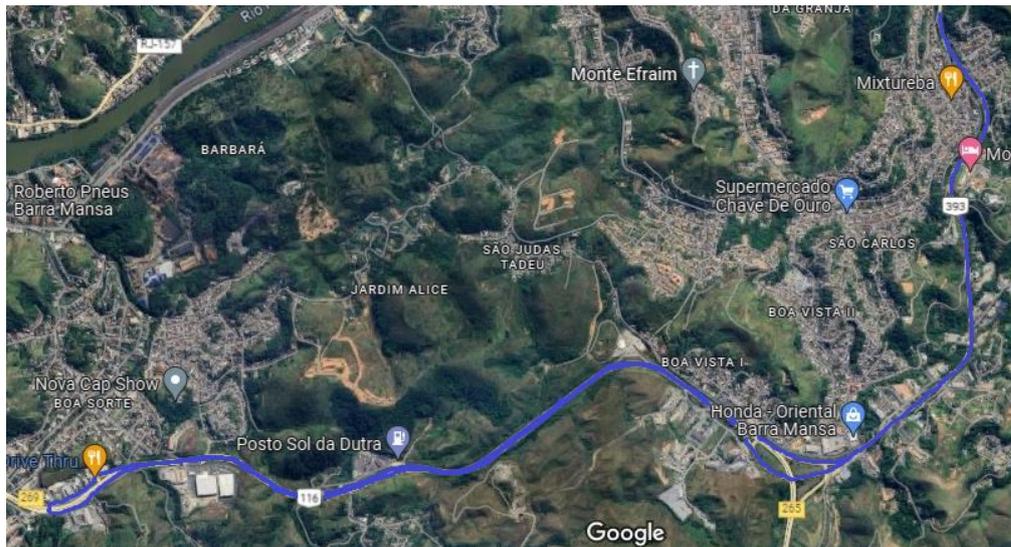
A distância total compreende cerca de 22Km de distância percorrida, o trecho urbano possui 6Km de extensão enquanto o trecho de rodovia possui 16Km, tendo como ponto de partida a garagem da Viação Cidade do aço e como ponto de chegada o Posto de Gasolina Metano, ambos na cidade de Barra Mansa. Este trajeto delonga em torno de 40 minutos, segundo a projeção de trajeto mais rápido feito pelo Google Maps. O trecho urbano do trajeto pode ser visto na figura A visão básica do trajeto pode ser encontrada na Figura 3.8

Figura 3.8 – Trajeto urbano aplicado.



Fonte: Imagem extraída do software GOOGLE MAPS® em 24 set 2023.

Figura 3.8 – Trajeto rodoviário aplicado



Fonte: Imagem extraída do software GOOGLE MAPS® em 24 set 2023

4 TESTES E RESULTADOS

Nesse capítulo as respostas as práticas associadas com a tarefa de classificação e observações são apresentadas, exploradas e debatida. A compreensão dos resultados é demonstrada através da associação entre as teorias aplicadas e as soluções criadas, como modelos, gráficos e as métricas de desempenho. A apresentação dos hiperparâmetros aplicados à estrutura do modelo proporciona a multiplicação e a comparação com outros procedimentos com o objetivo de solucionar uma variedade de problemas análogos. Os gráficos gerados através das métricas de desempenho permite uma melhor interpretação do trabalho.

4.1 MÉTODOS DE AVALIAÇÃO DE DESEMPENHO

Os modelos para a otimização do sistema de transmissão automática foram efetuados de acordo com os passos descritos na seção anterior. Entre treinamento, teste I e teste II, os resultados foram avaliados em função dos parâmetros de desempenho, com destaque para acurácia ao final de cada processo.

As definições de parametrização, foram realizadas segundo os melhores resultados apresentados pelos parâmetros de desempenho, sendo a acurácia a de mais importância e o desempenho desejável compatível dos erros mínimos entre treino, teste e execução. Ao fim de cada análise, fixou-se o valor dos parâmetros aplicados em cada modelo de ML, levando-os em consideração nos próximos testes.

A avaliação dos resultados para a validação, foi realizada através a matriz de confusão que permite determinar o desempenho do modelo criado e tornando – o prático, representando o quanto ele aprendeu, essa avaliação é importante para determinar o quanto o modelo encontrado é vantajoso.

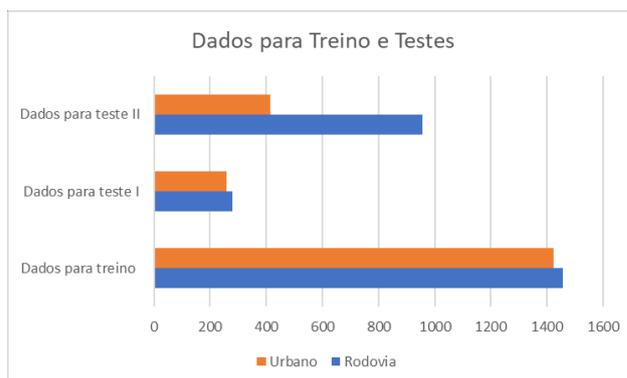
4.1.1 Separação dos dados

A técnica de separação de dados utilizada para o processo de treino e teste aplicada foi a *Hold-Out*. Essa técnica envolve separar os dados em duas partes e usar uma parte para treino que processa os dados e cria o modelo, o restante dos dados para teste é utilizado para a realização da previsão e verificação dos erros e acertos. A quantidade de dados separados em treino e teste são demonstrados na tabela 4.1 e figura 4.1.

Tabela 4.1 – Dados de treino e teste

Dados obtidos	Trajeto		Métricas		
	Rodovia	Urbano	Média	σ	CV (%)
Treino	1457	1422	1440	25	2
Teste I	280	260	270	14	5
Teste II	955	415	685	382	56

Figura 4.1 – Verificação dos dados.



As barras representam as quantidades de dados aplicados para treinamento e testes em rodovias e no trajeto urbano. Observa-se que no teste II existe uma alteração nos padrões de dados, indicado pelo desvio padrão (σ) e coeficiente de variação (CV), demonstrando a variabilidade do conjunto de dados utilizados nos testes. Os dados aplicados a etapa de criação do modelo, treinamento, e submissão desse modelo ao Teste I são mais homogêneos, como

indicado pelo CV e o desvio padrão. Houve uma preocupação em garantir a quantidade de dados adequados com o objetivo de criar um modelo de classificador genérico, evitando o surgimento do sub ajuste e do super ajuste, respectivamente *Underfitting* e *Overfitting*. O conjunto de dados completos aplicado ao Teste II, atesta que a amostra de dados dentro do trajeto em rodovia é maior que a do trajeto em perímetro urbano, uma vez que não houve a preocupação em determinar a quantidade de dados aplicados ao modelo criado.

4.1.2 Modelos

A proposta do trabalho se baseia na criação de quatro modelos. O primeiro e o segundo irão aprender a prever o estilo de condução do motorista, é constituído por 8 atributos que utilizam informações adquiridas da ECU como torque, velocidade, pressão, aceleração, potência mecânica, rotação, inclinação e posição do acelerador. As redes possuem propriedades apresentadas na tabela 4.2, que demonstram os melhores resultados e um atributo classe que vai predizer o modo de condução em ECO, para econômico, normal, para condições onde o motorista precisa de um pouco mais de aceleração como o trânsito diário e esporte quando se deseja uma maior performance do veículo.

O terceiro e o quarto modelos, também possui os 8 atributos análogos as redes anteriores e são capazes de descreverem o perfil de carga aplicado a condução. Esses modelos tem suas características apresentadas na tabela 4.2, para a determinação do modo de condução, porém seu atributo classe define se o veículo se encontra em um declive, aclive ou se está em um trecho plano.

Tabela 4.2 – Propriedades de controle de treinamento do modelo de ML

Características dos modelos à RNA	Modelos aplicados à RNA			
	Rodovia/Modo	Urbano/Modo	Rodovia/Carga	Urbano/Carga
Número de camadas ocultas	1	1	1	1
Número de neurônios	175	125	175	125
Função de ativação	Sigmoide	Tangente Hiperbólica	Sigmoide	Tangente Hiperbólica
Taxa de aprendizado	Constante	Constante	Constante	Constante
Número de épocas	6000	6000	6000	6000

No processo de treinamento e teste da rede, alterou-se o número de neurônios na camada oculta para cada modelo, com o propósito de apurar o número apropriado e oferecer o resultado ideal. Os modelos aplicados encontrados são apresentados na figura 4.2 e 4.3.

Figura 4.2 – Representação do modelo de rede com 175 neurônios na camada oculta.

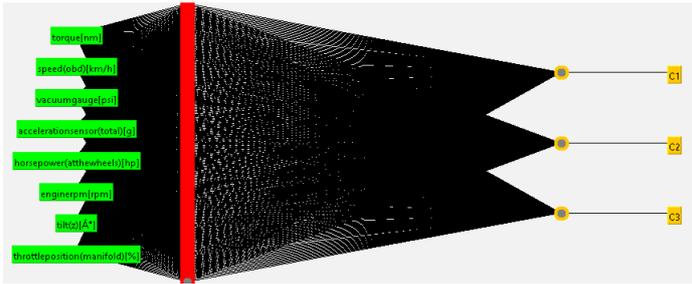
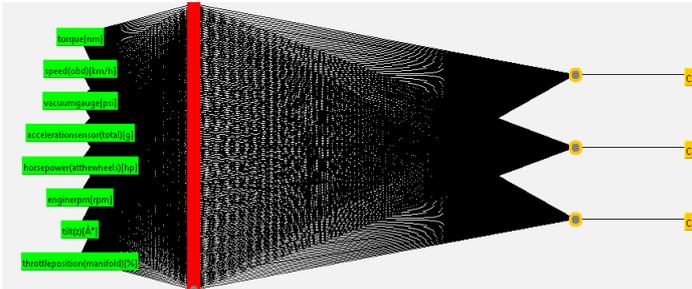


Figura 4.3 – Representação do modelo de rede com 125 neurônios na camada oculta.



4.1.3 Matriz de confusão

Os modelos analisam todos os atributos, fornecendo simultaneamente a classificação do modo de condução e da carga em cada proporção de aprendizado. Contou-se com a matriz de confusão para avaliar os resultados previstos pelos modelos aplicados conforme o quadro 4.1.

Quadro 4.1 – Matriz de confusão para treino e testes.

	Treino	Teste I	Teste II
Carga (Rodovia)	Confusion Matrix: [[229 12 2] [9 258 8] [1 11 137]] Accuracy: 0.9355322338830585	Confusion Matrix: [[29 2 0] [2 30 2] [0 0 14]] Accuracy: 0.9240506329113924	Confusion Matrix: [[48 10 1] [4 50 7] [5 9 39]] Accuracy: 0.791907514450867
Modo (Rodovia)	Confusion Matrix: [[327 3 0] [12 278 0] [0 0 170]] Accuracy: 0.9810126582278481	Confusion Matrix: [[51 4 0] [1 21 0] [0 0 24]] Accuracy: 0.9504950495049505	Confusion Matrix: [[283 33 0] [44 287 4] [0 2 129]] Accuracy: 0.8938618925831202
Carga(Urbano)	Confusion Matrix: [[161 4 0] [0 298 0] [0 2 103]] Accuracy: 0.9894366197183099	Confusion Matrix: [[17 1 0] [0 57 0] [0 0 11]] Accuracy: 0.9883720930232558	Confusion Matrix: [[15 0 0] [1 58 0] [0 0 10]] Accuracy: 0.9880952380952381
Modo (Urbano)	Confusion Matrix: [[421 0 0] [1 365 0] [2 0 67]] Accuracy: 0.9964953271028038	Confusion Matrix: [[82 0 0] [0 76 0] [1 0 15]] Accuracy: 0.9942528735632183	Confusion Matrix: [[126 27 1] [51 92 1] [0 0 24]] Accuracy: 0.7515527950310559

A partir da matriz de confusão completa, após a realização do treinamento e testes, apura-se o status do modelo através das avaliações de desempenho que forma uma medida mensurável, essa ação específica permite representar a dimensão do que ir-se-á conduzir, para entender o que está válido no trabalho e permite identificar o que está impreciso.

4.1.4 Métricas de avaliação

4.1.4.1 Acurácia

Para determinar os acertos obtidos pelo algoritmo realizam-se os cálculos de acurácia, com as informações obtidas da matriz de confusão. Os melhores modelos construídos obtiveram um resultado de 79%, 89%, 98% e 75% em condição de operação, indicando uma classificação correta média de 86% das instâncias para os dados aplicados em uma segunda aquisição de dados. O desvio padrão baixo apresentado pelos dados aplicados no teste II, indica que o modelo consegue operar com resultados homogêneos, apresentando pouca variação. O coeficiente de variabilidade médio apresentado após a execução do teste II, aponta uma pequena instabilidade do modelo. Esses valores são todos apresentados na tabela 4.4.

Tabela 4.3 – Resultado de acurácia encontrados.

Grupos	Tabela de acurácia					
	Treino	Teste I	Teste II	Média	σ	CV (%)
Carga (Rodovia)	0,94	0,92	0,79	0,88	0,08	9,02
Modo (Rodovia)	0,98	0,95	0,89	0,94	0,04	4,7
Carga (Urbano)	0,99	0,99	0,99	0,99	0,00	0,07
Modo (Urbano)	0,99	0,99	0,75	0,91	0,14	15,34

Figura 4.4 – Indicação da média e do desvio padrão da acurácia.

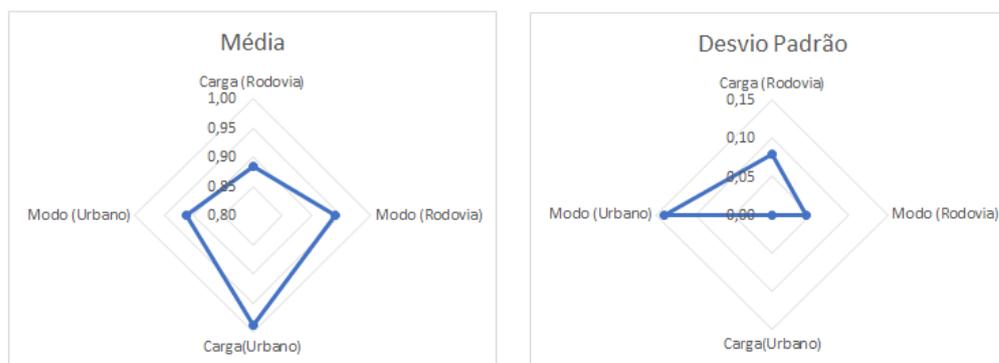
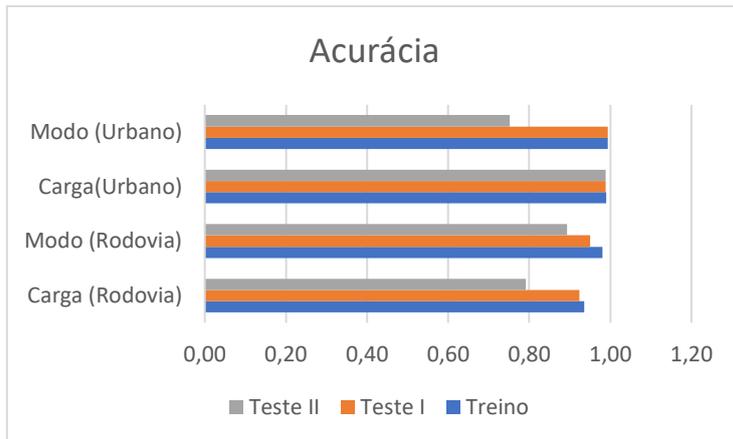


Figura 4.5 – Indicação da acurácia.



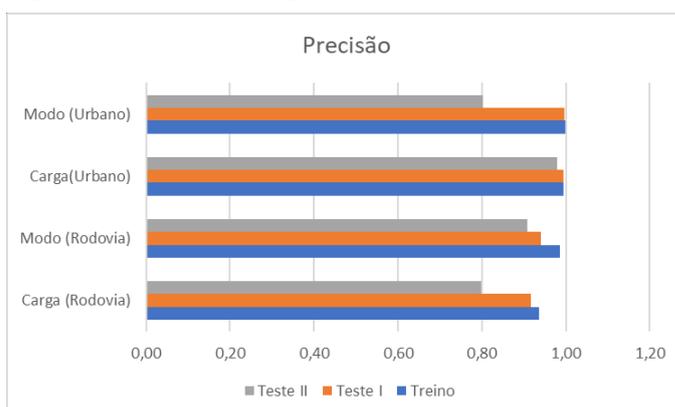
4.1.4.2 Precisão

No contexto da determinação do modo de condução e da indicação do tipo de carga aplicada ao veículo, realizam-se os cálculos de precisão. Os resultados são importantes porque eles apresentam a variação de acertos gerados pelo modelo de ML, das instâncias previstas como positivas, aproximadamente uma média de 94% são realmente positivas, conforme pode-se verificar na Tabela 4.5 e, também, na Figura 4.5. A precisão mede a proporção do trajeto em aclave, que foi previsto como carga C1 e que realmente aplicou a carga C1.

Tabela 4.4 – Resultado de precisão encontrados

Grupos	Tabela de Precisão					
	Treino	Teste I	Teste II	Média	σ	CV (%)
Carga (Rodovia)	0,94	0,92	0,80	0,88	0,07	8,39
Modo (Rodovia)	0,98	0,94	0,91	0,94	0,04	4,03
Carga (Urbano)	0,99	0,99	0,98	0,99	0,01	0,86
Modo (Urbano)	1,00	1,00	0,80	0,93	0,11	12,02

Figura 4.6 – Indicação da precisão.



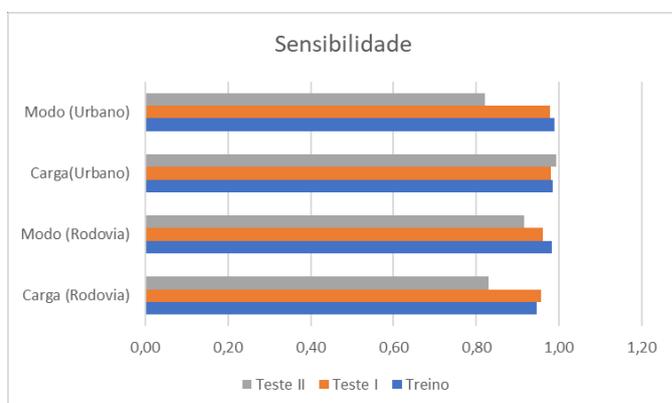
4.1.4.3 Sensibilidade

Conhecido na literatura como *Recall*, a sensibilidade verifica com êxito os resultados classificados como positivo, calcula-se a sensibilidade para verificar a habilidade de identificação e para entender melhor, a razão entre os modos de condução econômica classificado corretamente e o total de instâncias indicadas como econômica, como exemplo. O cálculo foi realizado para todas as matrizes de confusão aplicadas à essa pesquisa. Os modelos gerados indicaram corretamente uma média de 95% das instâncias boas. A sensibilidade é demonstrada pela tabela 4.6.

Tabela 4.5 – Resultado de sensibilidade encontrados.

Grupos	Tabela de Sensibilidade					
	Treino	Teste I	Teste II	Média	σ	CV (%)
Carga (Rodovia)	0,95	0,96	0,83	0,91	0,07	7,74
Modo (Rodovia)	0,98	0,96	0,92	0,95	0,03	3,6
Carga (Urbano)	0,99	0,98	0,99	0,99	0,01	0,67
Modo (Urbano)	0,99	0,98	0,82	0,93	0,09	10,07

Figura 4.7 – Indicação da sensibilidade



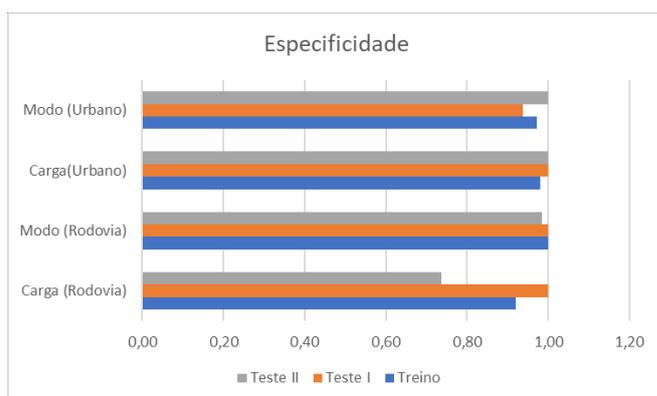
4.1.4.4 Especificidade

A especificidade pode ser descrita como aptidão do sistema em classificar corretamente a inexistência de uma condição para episódios que realmente não tem. Como exemplo, pode-se citar a possibilidade de um teste em modo normal ou plano não serem econômico em conduções identificadas como normal ou plano. Esse cálculo, assim como a sensibilidade, foi realizado para todas as matrizes de confusão. A especificidade é demonstrada pela tabela 4.7 aponta uma média de 96% de acerto

Tabela 4.6 – Resultado de especificidade encontrados.

Grupos	Tabela de Especificidade					
	Treino	Teste I	Teste II	Média	σ	CV (%)
Carga (Rodovia)	0,92	1,00	0,74	0,89	0,14	15,3
Modo (Rodovia)	1,00	1,00	0,98	0,99	0,01	0,89
Carga (Urbano)	0,98	1,00	1,00	0,99	0,01	1,11
Modo (Urbano)	0,97	0,94	1,00	0,97	0,03	3,23

Figura 4.8 – Indicação de especificidade



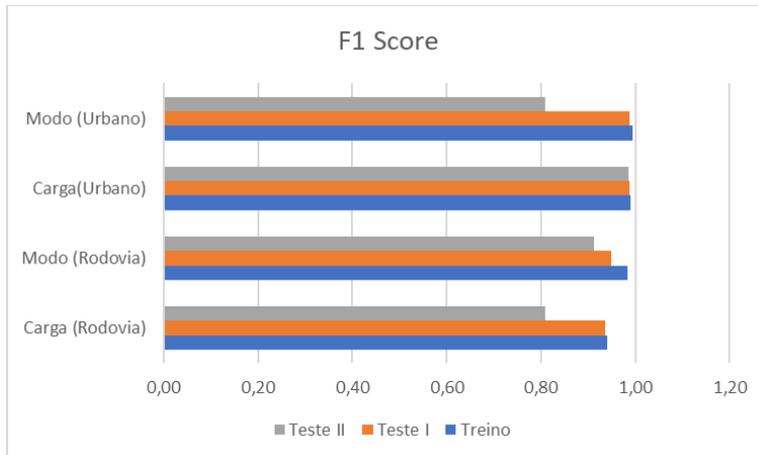
4.1.4.5 F1 Score

Calculou-se as métricas F1 Score em todas as matrizes de confusão utilizadas, para determinada a qualidade geral dos modelos criados, com isso verificam - se a efetividade dos modelos ao receberem um dado novo. Os resultados encontrados para esse métrica é demonstrado na tabela 4.7.

Tabela 4.7 – Resultado do F1 Score encontrados

Grupos	Tabela F1 Score					
	Treino	Teste I	Teste II	Média	σ	CV (%)
Carga (Rodovia)	0,94	0,94	0,81	0,90	0,07	8,31
Modo (Rodovia)	0,98	0,95	0,91	0,95	0,04	3,78
Carga (Urbano)	0,99	0,99	0,99	0,99	0	0,15
Modo (Urbano)	0,99	0,99	0,81	0,93	0,1	11,29

Figura 4.9 – Indicação da média harmônica.



Através dos resultados considerados nas Figuras 4.4, 4.5, 4.6, 4.7, 4.8 e 4.9 5 e nas tabelas 4.4, 4.5, 4.6, 4.7 e 4.8, foi possível verificar que os modelos utilizados são aptos de contribuir nas previsões das simulações e entregar resultados precisos a partir de um conjunto de dados desconhecido pelo modelo, obtidos com base no trajeto selecionado para realização dessa pesquisa.

5 CONCLUSÃO

Este trabalho propôs aplicar as RNA para a resolução de problemas concretos, idealizando e aplicando exímios classificadores, usados para prever o desejo do motorista e as condições de direção apresentadas pelas estradas, para promover uma alteração dos padrões de mudança de marcha, tornando esses padrões adaptáveis a cada condição diferente enfrentada pelo veículo.

Os dados necessários para realização deste trabalho, foram obtidos com êxito, através da utilização do scanner *bluetooth*, que possibilitou o acesso a ECU, em conjunto com o aplicativo Torque, onde os dados dos sensores foram lidos, transmitidos e registrados em tempo real, em trechos urbanos e rodovias, permitindo assim o seu emprego.

A aplicação do KDD para organização dos dados utilizados nessa pesquisa, preparando os dados de forma proveitosa e com aptidão para o uso foi realizado com sucesso, seguindo suas principais etapas, utilizando de ferramentas que promovem a remoção de valores inadequados e permitindo a inserção do atributo classe, que age na tomada de decisão a partir do algoritmo de mineração de dados MLP utilizado. Os projetos que aplicam as RNA

necessitam de uma excessiva compreensão sobre os dados de pesquisa, bem como sua organização e análise, obtidos a parti do algoritmo aplicado.

Essa pesquisa mostra a base de conceitos necessários para a efetuação da inteligência artificial através das RNA, com aplicação da rede MLP com o algoritmo *backpropagation*. A aplicação da rede MLP é relativamente fácil quando se aplica a biblioteca da linguagem Python *scikit-learn*, onde os testes realizados apontam para utilização de dois modelos computacionais. O primeiro possui 175 neurônios na camada oculta, aplicando uma função de ativação sigmoideal, o segundo apresenta 125 neurônios na camada oculta utilizando a Tangente Hiperbólica como função de ativação. Estes modelos alcançaram com êxito as aplicações práticas no sistema de controle da transmissão automática, como estudo de caso de classificação do modo de condução e do tipo de carga aplicado.

Um modo simples e notável de verificar o modelo de classificação gerado é através do emprego de métricas como a acurácia, que fornece um entendimento dos resultados obtidos pelos modelos. Contudo, não se pode fundamentar nossa avaliação somente na acurácia, embasar nossa análise em outras métricas pode ser importante. O F1 Score indica o desempenho do modelo ao se deparar com dados novos, onde essa métrica se torna muito importante na interpretação do problema, sendo apontado como um bom resultado.

Os parâmetros de desempenho apresentados por esse trabalho como a acurácia, precisão, especificidade, sensibilidade e F1 Score, mostram que os resultados são promissores. As métricas aplicadas ao modelo de classificação utilizadas exibem uma média geral acima de 80%, os desvios padrão médio apresentado nesse estudo, indicam que o modelo consegue operar com resultados próximos a média, demonstrando assim um bom desempenho geral para o modelo, sendo apontado como um resultado seguro.

Através dos resultados considerados nas figuras e tabelas apresentados, foi possível verificar que os modelos utilizados são aptos a contribuir nas predições das simulações e entregam resultados precisos a partir de um conjunto de dados desconhecidos pelo modelo, obtidos com base no trajeto selecionado para realização dessa pesquisa.

Esse trabalho utiliza-se do deslocamento da linha de mudança de marcha, em função das classificações realizadas pela inteligência artificial, para preservar a força motriz adequada em dimensão à inclinação da estrada, e do estilo de direção aplicado pelo condutor. O quadro no qual o veículo acelera isoladamente, como numa descida, é possível realizar as trocas de marchas mais cedo, praticasse também nessas condições o freio motor, permitindo reduzir a velocidade do veículo com a ajuda do motor, sem acionar o freio. Na condição de veículo com

carga em um aclave, admita-se reter as marchas para manter as rotações mais altas, impedindo trocas ascendentes ao reduzir o pé do acelerador para fazer curvas como exemplo.

As vantagens relacionadas ao controle da transmissão adaptável na prática, pode ser resumida na diminuição do consumo de combustível, no fornecimento de uma boa qualidade de troca de marcha, na adaptação do câmbio ao desejo do motorista e o tipo de estrada que o veículo está transitando. Nenhum hardware específico precisa ser utilizado, apenas a implementação por software é aplicada, as seleções de padrões de mudança ocorrem de forma automática e off-line sem a necessidade de utilização de botões que indiquem a vontade do motorista, como por exemplo o botão “ECO”, utilizado por alguns fabricantes.

Os resultados apresentados neste trabalho sugerem que a proposta possui um bom potencial para uma implementação em ambiente real. Hipoteticamente, na prática, o resultado do trabalho poderia melhorar o desempenho, se fosse implementado em um veículo. Para isso alguns aspectos deveriam ser contemplados:

A aplicação de microcomputadores ligados entre si e o conceito de computação ubíqua são aspectos necessários para realização desse trabalho, utilizando a arquitetura distribuídas onde os diferentes módulos de controle eletrônico são interligados de forma similar às redes de computadores existentes.

É necessário que os sinais dos sensores sejam digitalizados e transmitidos aos diferentes módulos que fazem uso dessas informações com alta velocidade e segurança, para que o acionamento do sistema ocorra de forma adequada.

Portanto, pode-se admitir a implementação desse estudo como proposta de trabalhos futuros, utilizando-se dos recursos da eletrônica embarcada a veículos que possui a capacidade de realizar tarefas semelhante à de um computador, que se utiliza de um programa fixo, conhecido como Firmware, empregados em sua memória, de interfaces de entradas e saídas aplicadas aos sensores e atuadores e de uma interface de comunicação com outros dispositivos. O desenvolvimento de procedimentos para calibração dos padrões de mudança de velocidade do sistema de transmissão, de forma complementar esse estudo, aplicando outras classes de sistema inteligente, que permitem simular o comportamento humano, atribuindo aos sistemas de controle características da inteligência humana, dentre eles tem se os controladores *Fuzzy*, pode ser reconhecido como um trabalho futuro.

A aplicação das RNA com prática nesse método, pode ser estendida a outros tipos de veículos que utiliza o sistema de transmissão automática, como ônibus e caminhões, a aplicação de diferentes algoritmos que podem ser analisados, testados e correlacionados com o resultado apresentado por este estudo de RNA, visto que este trabalho exibiu valores de parâmetros

apoiados na matriz de confusão e suas métricas como Acurácia e Recall. Por fim, entre outras oportunidades de pesquisa, pode-se aplicar RNA a outros mecanismos que compõem o veículo como por exemplo os freios, com o objetivo de se obter um melhor desempenho em todas as condições de funcionamento, classificando o tipo de superfície e determinando a força de frenagem adequada para evitar um possível escorregamento e executar uma frenagem mais rápida e segura.

Os resultados foram muitos positivos, no entanto esses resultados não se traduzem a todos os sistemas de controle de transmissão automática, mas sobre a pesquisa a mesma metodologia pode ser aplicada a outros modelos de veículos com o entendimento de que o mesmo desempenho pode ser alcançado. O funcionamento do sistema foi modelado ao comportamento de um único motorista especialista, que reflete na maneira de condução do profissional.

Este estudo alcançou com êxito um modelo computacional sujeito a aplicações práticas no sistema de controle da transmissão automática, como estudo de caso de classificação do modo de condução e do tipo de carga aplicados. Trabalhos alheios de relevância coeso a esse estudo podem ser aplicados.

REFERÊNCIAS BIBLIOGRÁFICAS

AGUIAR, H.; JUNIOR, O. **Inteligência Computacional aplicada à administração, economia e engenharia em Matlab®**. Editora Thomson Learning, 2007.

AMARAL, M. C. **Emprego e otimização de redes neurais artificiais na classificação de trincas detectadas por ultrassom em engates metálicos de vagões ferroviários**. Orientador: Marcelo Lucas Pereira Machado, 2021. Dissertação (mestrado) – Instituto Federal do Espírito Santo, Programa de Pós-graduação em Engenharia Metalúrgica e de Materiais, Vitória, 2021. Disponível em: <https://repositorio.ifes.edu.br/handle/123456789/1490>.

AMAZON WEB SERVICE. **O que é Python?** Disponível em: <https://aws.amazon.com/pt/what-is/python/>, acesso em 29 de outubro de 2023.

ARAÚJO, A. M. **Classificação e detecção de pessoas em ambientes não controlados utilizando redes neurais convolucionais**. Orientador: Prof. Dr. Leonardo Vidal Batista, 2019. Monografia (Bacharel) – Universidade Federal da Paraíba, Programa de Graduação em Ciências da Computação, João Pessoa, 2019. Disponível em: <https://repositorio.ufpb.br/jspui/bitstream/123456789/16098/1/AMA02102019.pdf>

BARRETO, C. A. S. **Uso de técnicas de aprendizado de máquina para identificação de perfis de uso de automóveis baseado em dados automotivos**. Orientador: Prof. Dr. João Carlos Xavier Júnior, 2018. Dissertação (mestrado) – Universidade Federal do Rio Grande do Norte, Programa de Pós-graduação em Engenharia de Software, Natal, 2018. Disponível em: <https://repositorio.ufrn.br/handle/123456789/26017>

BLOG DOUTOR-IE, 2023. Disponível em: <https://www.doutorie.com.br/blog/sensor-de-pressao-absoluta-map/>. Acesso em 23setembro 2023.

BOSCH, R.; **Manual de Tecnologia Automotiva**. Editora Edgard Blücher Ltda., 2005.

CAMILO, C. O. **Mineração de Dados: Conceitos, Tarefas, Métodos e Ferramentas**. Orientador: Prof. João Carlos da Silva, 2009. Relatório Técnico – Universidade Federal de Goiás, Programa de Pós-graduação em Ciência da Computação Disponível em: https://rozero.webcindario.com/disciplinas/fbmg/dm/RT-INF_001-09.pdf

CAMPOS, M. M.; SAITO, K. **Sistemas inteligentes em controle e automação de processo**, Editora Ciência Moderna Ltda., 2004.

COLOMBO, M. S. **Identificação de colunas de destilação por redes neurais artificiais**. Orientador: Prof. Dr. Oswaldo Curty da Motta Lima. Coorientador: Prof. Dr. Cid Marcos Gonçalves Andrade. Dissertação (Mestrado) - Universidade Estadual de Maringá, Centro de Tecnologia, Departamento de Engenharia Química, Programa de Pós-Graduação em Engenharia Química, 2020. Disponível em: <https://www.unicesumar.edu.br/anais-epcc-2021/content/uploads/sites/236/2021/11/698.pdf>

CONVERTIO, 2023. Disponível em: <https://convertio.co/pt/csv-xlsx/>, aceso em 31 outubro 2023.

COSTA, E. **Mineração de dados educacionais: conceitos, técnicas, ferramentas e aplicações.** Jornada de Atualização em Informática na Educação 1.1 (2012): P.1-29. Disponível em: <http://milanesa.ime.usp.br/rbie/index.php/pie/article/view/2341> .

DA SILVA, B.M., VANDERLINDE, M. **Inteligência artificial, aprendizado de máquina.** Universidade do Estado de Santa Catarina, 2010. Disponível em: https://www.ceavi.udesc.br/arquivos/id_submenu/387/brigiane_machado_da_silva_marcos_vanderlinde.pdf

DIAS, J. **Curso de Especialização em Engenharia Automotiva – Módulo de Transmissão.** Universidade Tecnológica Federal do Paraná, 2011. Disponível em: <https://pt.scribd.com/document/172920203/Apostila-UTF-PR-Curso-de-Especializacao-em-Engenharia-Automotiva-Modulo-Transmissoes-I>

FAYYAD, U.; PIATESTSKY – SHAPIRO, G.; SMITH, P. **From Data Mining to Knowledge Discovery in Databases.** AI Magazine, Vol. 17, N°. 3, p. 37 – 54, 1996.

FHB PERFORMANCE, 2023. Disponível em: <https://www.fhbperformance.com.br/como-o-cambio-automatico-se-adapta-a-cada-condicao/> , acesso em 30 agosto 2023.

GOMEDE, E. **Arquiteturas de redes neurais artificiais para aprendizado adaptativo em sistemas de e-learning.** Orientador: Leonardo de Souza Mendes, 2020. Tese (Doutorado) – Universidade Estadual de Campinas, Faculdade de Engenharia Elétrica e Computação. Campinas, SP 2020. Disponível em: <https://repositorio.unicamp.br/Busca/Download?codigoArquivo=470056>

GUIMARÃES A.; **Eletrônica Embarcada Automotiva.** Editora Érica, 2007.

HA, S. H.; JEON, H. T. **Development of Intelligent Gear-shifting Map Based on Radial Basis Function Neural Networks.** International Journal of Fuzzy Logic and Intelligent Systems, Vol. 13, N°. 2, p. 116-123, 2013.

HAYKIN, S. **Neural Networks a Comprehensive Foudantion.** Editora Pearson Education, Inc, 2005

HE, X.; LIN, M.; XU, L. **Integrating Gear Shifting Preference into Personalized Shift-Scheduling Calibration.** Appl. Sci. 2021. Disponível em: <https://doi.org/10.3390/app11072986>

HOSSEINI – ASL, E.; ZURADA, J.M.; NASRAOUI, O. **Deep Learning of Part-Based Representation of Data Using Sparse AutoencodersWith Nonnegativity Constraints.** IEEE Transactions on Neural Networks and Learning Systems 2016. Disponível em: <https://doi.org/10.1109/TNNLS.2015.2479223>

KIM, N.; VIJAYAGOPAL, R.; ROUSSEAU, A.; LEE, B.; KIM, S. **Gearshift Calibration for Automatic Transmission Using a Model-Based Optimization Algorithm.** IEEE Vehicle Power and Propulsion Conference (VPPC), Belfort, França, 11–14 dezembro 2017; Disponível em: <https://ieeexplore.ieee.org/document/8330948>

LUDEMIR, T. B. **Inteligência Artificial e Aprendizado de Máquina: estado atual e tendências.** Estudos Avançados, Vol 35, 2021. Disponível em: <https://www.scielo.br/j/ea/a/wXBdv8yHBV9xHz8qG5RCgZd>.

MAHESH, B. **Machine learning algorithms-a review**. International Journal of Science and Research (IJSR). Vol. 9, N°.1, Internet, 2020. Disponível em: <https://www.ijsr.net/archive/v9i1/ART20203995.pdf>

MONARD, M. C.; BARANAUSKAS, J. A. **Conceitos Sobre Aprendizado de Máquina. Sistemas Inteligentes Fundamentos e Aplicações**. 1 ed. Barueri-SP: Manole Ltda, 2003.

MURTY, M. N.; DEVI V. S. **Introduction to pattern recognition and machine learning**. Vol 5, Ed., Board for the IISc Lecture Notes Series, 2010.

NEDEVAL, V.; PEHLIVANOVA, T. **Students' Performance Analyses Using Machine Learning Algorithms in WEKA**. IOP Conference Series: Materials Science and Engineering, Vol 1031, November 2020. Disponível em: 10.1088/1757- 899X/1031/1/012061

NETO, R. F. O.; **Ciência dos Dados pelo Processo de KDD**. Livro eletrônico. 1 ed. Petrolina- PE, 2021.

PREDIÇÃO. In: DICIO, Dicionário Online de Português. Porto: 7Graus, 2023. Disponível em: <https://www.dicio.com.br/trabalho/> . Acesso em: 30/09/2023.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E.; **Scikit-learn: Machine Learning in Python**, Journal of Machine Learning Research, Vol 12, 2011. Disponível em : <https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf>

PUCRS Online, 2023. Disponível em: <https://online.pucrs.br/blog/public/tecnologia-da-informacao/>, acesso em 09 setembro 2023.

RADHAKRISHNAN, P. **What are hyperparameters, and how to tune the hyperparameters in a deep neural-network?** Disponível em: <https://towardsdatascience.com/what-are-hyperparameters-and-how-to-tune-the-hyperparameters-in-a-deep-neural-network-d0604917584a> . Acesso em 02 Dez. 2023.

SANTICIOLLI, F.M.; COSTA, E.S.; ECKERT, J.J.; DIONÍSIO, H.J.; ALKIMIN, L.C.; DEDINI, F.G, **Multiobjective gear shifting optimization considering a known driving cycle**, Acta Scientiarum. Technology, Vol 37, N° 3, 2015. Disponível em: <https://periodicos.uem.br/ojs/index.php/ActaSciTechnol/article/view/26466>

SCHNEIDER, L. F. **Aplicação do Processo de Descoberta de Conhecimento em Dados do Poder Judiciário do Estado do Rio Grande do Sul**. Orientador: Prof. Luís Otávio Alvares. Dissertação (Mestrado) - Universidade Federal do Rio Grande do Sul, Programa de Pós-Graduação em Computação, 2003.

SENAI. SERVIÇO NACIONAL DE APRENDIZAGEM INDUSTRIAL. **Transmissão automática com controle eletrônico**, Editora Senai – SP, 2016.

SIMÕES, M.G.; SHAW, I.S., **Controle e Modelagem Fuzzy**, 2º edição, Editora Edgard Blücher, 2017.

WEBLINK. Protocolos de rede. Disponível em: <https://www.weblink.com.br/blog/tecnologia/conheca-os-principaisprotocolos-de-internet/>, Acesso em 29 set. 2023.

WITTEN, I. H.; FRANK E.; HALL, M. A.; PAL, C. J. **Data mining: practical Machine Learning tools and techniques**, 4. Ed., Morgan Kaufmann, 2016.

ZOU, W.; WANG Y.; ZHONG, C.; SHENLONG, L.; 2022. **Research on shifting process control of automatic transmission**, Scientific Report, Vol 12, 2022. Disponível em: <https://rdcu.be/dyDFO>