INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO
MESTRADO EM ENGENHARIA INFORMÁTICA

Deteção automática dos locais com maior índice de sinistralidade

TADEU TEIXEIRA GUIMARÃES JÚNIOR
Outubro de 2022

POLITÉCNICO
DO PORTO

Instituto Superior de Engenharia do Porto

Deteção automática dos locais com maior índice de sinistralidade

Tadeu Teixeira Guimarães Saraiva Júnior

Dissertação para obtenção do Grau de Mestre em Engenharia Informática, Área de Especialização em Engenharia de Software

Orientadora: Ana Maria Dias Madureira Pereira

Coorientador: Diogo Manuel de Matos Braga

Júri:

Presidente:

Vogais:

Porto, Outubro de 2022

Dedicatória

À Polícia Rodoviária Federal, instituição que prima pela excelência em suas ações, formando policiais que são mais que bons profissionais, são bons cidadãos e modificadores do ambiente em que atuam.

Obrigado por esta experiência!

Resumo

A Polícia Rodoviária Federal (PRF) é o órgão responsável pela fiscalização de trânsito no âmbito das rodovias federais brasileiras. Uma das suas missões é garantir a segurança viária e proteger a vida dos usuários das rodovias. Um dos seus grandes desafios é identificar causas sistemáticas de acidentes, sejam estruturais ou sociais, e atuar para reduzir o número de ocorrências, por meio de políticas e patrulhamento ostensivo.

O objetivo desse trabalho é realizar um estudo da utilização de técnicas de *Machine Learning* (ML) para a identificação dos locais com maior probabilidade de ocorrência de Acidentes de Trânsito (AT), a partir dos dados históricos de acidentes da PRF. Para isso, é analisada a capacidade de diversos algoritmos de ML em predizer o local de ocorrência de um novo AT. Além disso, é verificado se a utilização de outras variáveis que não são consideradas no método atual da instituição, tais como a condição climática, o tipo de acidente e a geolocalização do local da ocorrência, influenciam na assertividade dos algoritmos.

Palavras-chave: acidente de trânsito, acidente rodoviário, rodovias federais, inteligência artificial, Aprendizagem Automática, aprendizado de máquina, polícia rodoviária federal, PRF

Abstract

The Polícia Rodoviária Federal (PRF – Federal Highway Police) is responsible for supervising traffic on Brazilian federal highways. One of its missions is to ensure safety and protect the lives of road users. One of its great challenges is to identify systematic causes of accidents, whether structural or social, and to act to reduce the number of occurrences, through policies and ostensible patrolling.

The objective of this work is to carry out a study of the use of Machine Learning (ML) techniques to identify the places with the highest probability of occurrence of Traffic Accidents (TA), from the historical accidents data of PRF. For this, the ability of several ML algorithms to predict the place of occurrence of a new TA is analyzed. In addition, it is verified whether the use of other variables that are not considered in the institution's current method, such as the weather condition, the type of accident and the geolocation of the place of occurrence, influence the assertiveness of the algorithms.

Keywords: traffic accident, road accident, highway accident, federal roads, brazilian highway, federal highway, artificial intelligence, machine Learning, federal highway police, deep learning, PRF

Agradecimentos

À professora Ana Maria Madureira por ser minha salvadora em um momento de angústia, pela oportunidade de aprender mais uma área de conhecimento, pela sua orientação e clareza em me guiar por essa dissertação, e ao coorientador Diogo Manuel de Matos pelas diversas dicas na abordagem ao problema.

Ao professor Nuno Silva, que me abriu as portas à essa grande empreitada, me acolhendo com muito respeito e carinho, me guiando durante todo o processo. Ao professor Paulo Maio, que me ensinou a ser um desenvolvedor, e depois me ensinou a ver com os olhos de um engenheiro de software. E a todos os professores, no nome da professora Isabel Azevedo, que me desenvolveram as ferramentas necessárias para uma nova formação.

À minha esposa, sempre companheira em minhas aventuras e maluquices. O caminho foi difícil, mas a vista é maravilhosa!

Amo-te!

Aos meus filhos, Isabela e Guilherme. Que eu possa ser digno de ser-lhes um exemplo. Vocês são o sentido da minha vida!

À minha mãe, por toda a educação e suporte, sempre me provendo com o máximo que dispunha. E aos meus irmãos, pelas alegrias compartilhadas.

Ao meu amigo José Ferreira, parceiro de tantas desventuras e noites adentro de estudo. Valeu pelo suporte!

Você não é um *trolha*!

Índice

1 Introdução 1

1.1 Contexto 1

1.2 Problema 3

1.3 Objetivo 4

1.4 Estrutura do Documento 5

1.5 Sumário 5

2 Estado da arte 7

2.1 Acidentes de Trânsito em Rodovias Federais 7

2.2 Inteligência Artificial 12

2.3 Machine Learning 13

2.3.1 Classificação das Técnicas de Aprendizagem Automática 13

2.3.2 Tipos de Problemas 14

2.3.3 Algoritmos de AA 15

2.4 Deep Learning 19

2.4.1 Artificial Neural Networks 20

2.4.2 Convolutional Neural Networks 21

2.5 Utilização de ML na Análise de Acidente de Trânsito 22

2.6 Tecnologias e Frameworks 22

2.6.1 Python 23

2.6.2 R 23

2.6.3 Keras 24

2.6.4 TensorFlow 24

2.6.5 PyTorch 25

2.6.6 scikit-learn 26

2.6.7 Escolha das tecnologias 26

2.7 Sumário 27

3 Análise de Valor 29

3.1 New Concept Development 29

3.1.1 Identificação da Oportunidade 31

3.1.2 Análise da Oportunidade 31

3.1.3 Geração e Enriquecimento de Ideias 33

3.2 Proposta de Valor 33

3.3 Sumário 34

4 Metodologia 35

4.1 Hipótese e questões relevantes 35

4.2 Métricas de Avaliação 36

4.2.1 Mean Absolute Error 36
4.2.2 Mean Squared Error 37
4.2.3 Root Mean Squared Error 37
4.2.4 R^2 score 37
4.2.5 Accuracy 38
4.2.6 Confusion Matrix 38
4.2.7 Precision, Recall e F1-score 39
4.2.8 Curva ROC 40
4.3 Métodos de Otimização 40
4.3.1 Método Holdout 40
4.3.2 Cross-Validation 41
4.4 Metodologia de Avaliação 42
4.5 Sumário 42
5 Estudo Computacional 43
5.1 Tratamento e engenharia de dados 43
5.1.1 Coleta e descrição dos dados 43
5.1.2 Limpeza e reorganização dos dados 45
5.1.3 Engenharia de dados 47
5.1.4 Análise Exploratória dos dados 49
5.1.5 One-Hot-Encoding 51
5.1.6 Lidando com variáveis cíclicas 52
5.2 Geolocalização 53
5.3 Aplicação dos Algoritmos 54
5.4 Divisão dos dados 56
5.4.1 Definição dos valores das métricas 56
5.4.2 Normalização 57
5.5 Ambiente computacional 58
5.6 Regressão Linear 58
5.7 Machine Learning 60
5.7.1 Regressão 60
5.7.2 Classificação 64
5.8 ANN 66
5.9 Avaliação 67
5.10 Comparação com estudos semelhantes 68
5.11 Sumário 69
6 Conclusão 71
6.1 Contribuições deste trabalho 72
6.2 Limitações deste trabalho 72
6.3 Propostas de trabalhos futuros 73

Lista de Figuras

Figura 1 – As 10 principais causas de morte no Brasil para ambos os sexos e todas as idades em 2019 (Organização das Nações Unidas, 2020). 2
Figura 5 - Distribuição da Disposição a Pagar a mais em taxas para tentar evitar um novo acidente (Silveira, 2009) 9
Figura 7 – Árvore de decisão de uma *Decision Trees* (Raschka, 2019) 16
Figura 8 – Gráfico da função logística (Geron, 2019) 19
Figura 9 – As relações entre a rede, as camadas, a função de perda e os optimizadores de uma ANN. (Chollet, 2017) 20
Figura 10 – Camadas de uma CNN conectadas ao seus respetivos subquadrantes (Geron, 2019) 21
Figura 11 – Representação dos tensores de níveis 0 e 1 na primeira linha, e de níveis 2 e 3 na segunda linha (Raschka, 2019). 25
Figura 2 – Processo de Inovação (Martikainen, 2017) 30
Figura 3 – O NCD é um modelo relacional, não linear (Martikainen, 2017) 30
Figura 4 – *Value Proposition Canvas*. Adaptado de (Daniel Pereira, 2019). 34
Figura 12 – *Confusion Matrix* de classificação binária (Raschka, 2019). 38
Figura 13 – Curva ROC (Geron, 2019) 40
Figura 14 – Conceito de *Holdout* (Raschka, 2019) 41
Figura 15 – Representação da *Cross-Validation* (Raschka, 2019). 42
Figura 16 – Amostra dos dados originais sem tratamento 45
Figura 17 – Distribuição dos acidentes entre os estados. 49
Figura 18 – Principais causas de acidentes. 49
Figura 19 – Distribuição de acidentes entre os dias da semana. 50
Figura 20 – Distribuição de acidentes entre as horas do dia. 50
Figura 21 – Correlação entre as *features* e o alvo (km). 51
Figura 22 – Correlação entre as *features* e o alvo (km), mas sem a barra do “km” 52
Figura 23 – Plotagem dos pontos de latitude e longitude dos dados nacionais 53
Figura 24 - Plotagem dos pontos de latitude e longitude do Distrito Federal 53
Figura 25 - Plotagem dos pontos de latitude e longitude do Distrito Federal sem o ponto sobre o Pacífico. 54
Figura 26 – Distribuição de acidentes hora a hora na BR 116 em SC no ano de 2017. 60

Lista de Tabelas

Tabela 1 – Análise SWOT (GÜREL, 2017).....32
Tabela 2 – Dicionário de Dados.....44
Tabela 3 – Resultados da regressão linear.....58
Tabela 4 – Resultados da regressão em ML nos dados completos (dados nacionais).....61
Tabela 5 – Resultados da regressão em ML nos dados da regional do DF 61
Tabela 6 – Resultados da regressão em ML nos dados da regional de SP.....62
Tabela 7 – Resultados da regressão em ML nos dados da regional de SC.....63
Tabela 8 – Resultados da regressão em ML nos dados da UOPs 01 das regionais DF, SP e SC .64
Tabela 9 – Resultados da classificação em ML nos dados da regional do DF 65
Tabela 10 – Resultados da classificação em ML nos dados da regional de SP.....65
Tabela 11 – Resultados da classificação em ML nos dados da regional de SC.....65
Tabela 12 – Resultados da aplicação da ANN e comparação com os melhores resultados de ML 66
Tabela 13 – Comparação dos resultados entre ANN e os algoritmos de ML.....66

Acrónimos e Símbolos

Lista de Acrónimos

AA	Aprendizagem Automática
AI	Inteligência Artificial (*Artificial Intelligence*)
ANN	*Artificial Neural Networks*
AT	Acidentes de trânsito
AUC	*Area Under the Curve*
BR-###	Rodovia federal número ###
CE	Ceará
CNN	*Convolutional Neural Networks*
CNT	Confederação Nacional do Transporte
CV	*Cross-Validation*
DAST	Década de Ação pela Segurança no Trânsito
DEL	Delegacia
DENATRAN	Departamento Nacional de Trânsito
DF	Distrito Federal
DL	*Deep Learning*
DT	*Decision Trees*
EDK	Estimativa de Densidade Kernel
FFE	*Fuzzy Front End*
GO	Goiás
KNN	*k-Nearest Neighbors*
LSTM	*Convolutional Long Short-Term Memory Neural Network*
MAE	*Mean Absolute Error*
MG	Minas Gerais

ML	*Machine Learning* (Aprendizagem Automática)
MSE	*Mean Squared Error*
NCD	*New Concept Development*
NPD	*New Product Development*
OMS	Organização Mundial da Saúde
ONU	Organização da Nações Unidas
PA	Pará
PAGD	Plano de Ação Global para a Década
PRF	Polícia Rodoviária Federal
RF	*Random Forest*
RMSE	*Root Mean Squared Error*
ROC	*Receiver Operating Characteristic*
SAD	Sistema de Apoio a Decisão
SC	Santa Catariana
SIG	Sistema de Informações Geográfica
SP	São Paulo
SVM	*Support Vector Machine*
SWOT	*Strengths, Weaknesses, Opportunities and Threats*
UOP	Unidade Operacional
UOP	Unidade Operacional
UPS	Unidade Padrão de Severidade
VPM	Variação Percentual Mensal

1 Introdução

O objetivo deste capítulo é introduzir o contexto, descrever o problema e definir os objetivos para a utilização de técnicas de Inteligência Artificial (*Artificial Intelligence* – AI) na identificação de locais com maior probabilidade de ocorrência de Acidentes de Trânsito (AT) nas rodovias federais brasileiras. A quantidade e gravidade dos AT impactam significativamente o país, com consequências econômicas e de saúde pública. Nesse contexto, a Polícia Rodoviária Federal (PRF) tem a função precípua de implementar as políticas públicas que visam reduzir essa calamidade.

Esse documento descreve ainda o estado da arte das áreas relacionadas, a análise de valor, o tratamento e engenharia dos dados, o estudo da utilização dos algoritmos e a conclusão do trabalho realizado.

1.1 Contexto

Segundo a Organização das Nações Unidas (ONU), os acidentes de trânsito são a maior causa de morte entre pessoas de 5 a 29 anos no mundo (Organização das Nações Unidas, 2021). No Brasil, as mortes por AT representaram a oitava maior causa de mortes no ano de 2019, com aproximadamente 16 mortes a cada 100 mil habitantes (Organização das Nações Unidas, 2020). Apesar desse número representar uma redução em relação ao ano de 2000 – quando as mortes foram de 20 a cada 100 mil pessoas (Organização das Nações Unidas, 2020) – ele ainda representa um valor expressivo de vidas e recursos.

Estima-se que a perda econômica provocada pelos AT seja superior a 50 bilhões de reais (8,3 mil milhões de euros) por ano (Carvalho, 2020). Desse valor, R$ 40 bilhões representam o custo de acidentes nas rodovias de área rurais e R$ 10 bilhões o custo de acidentes nas áreas urbanas. Tais cifras consideram os valores relativos as despesas pré e pós hospitalares e os valores relativos à perda de produção das vítimas, responsável pela maior do montante.

No âmbito das rodovias federais (i.e., rodovias e estradas que ligam os estados brasileiros, sob a responsabilidade do governo federal), foram registados 64.452 sinistros em 2021, com 5.391 mortes (Confederação Nacional do Transporte, 2022). Segundo a Confederação Nacional do Transporte (CNT), o custo estimado dos AT do ano de 2021 é de R$ 12,2 bilhões (2,03 mil milhões de euros) (Confederação Nacional do Transporte, 2022).

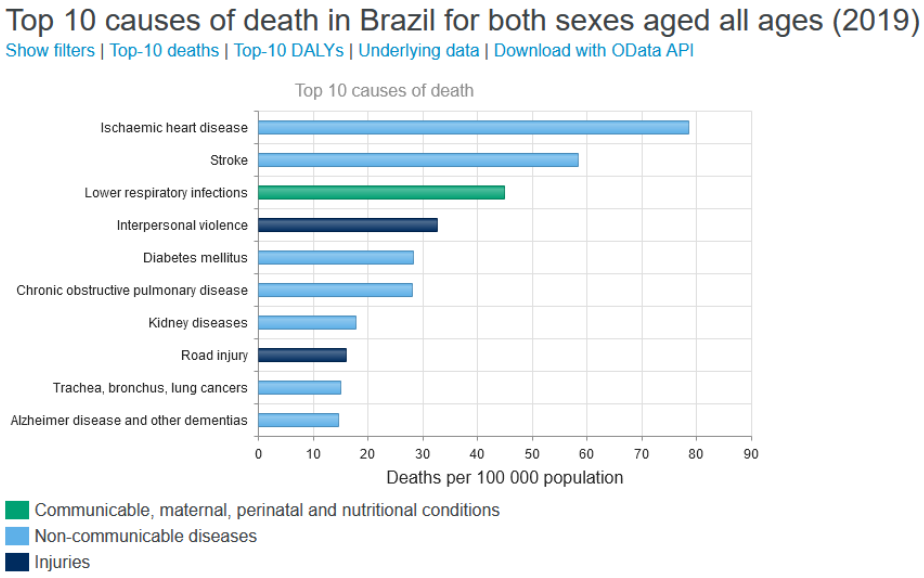

Figura 1 – As 10 principais causas de morte no Brasil para ambos os sexos e todas as idades em 2019 (Organização das Nações Unidas, 2020).

A Polícia Rodoviária Federal é o órgão de segurança pública responsável pela fiscalização de trânsito e aplicação das políticas públicas de redução de AT no âmbito das rodovias federais. Uma das suas missões é garantir a segurança e proteger a vida dos usuários das BRs[1], trabalhando para reduzir a quantidade de acidentes, salvando vidas e protegendo bens materiais.

Um dos seus grandes desafios é identificar os locais com maiores ocorrências de AT, objetivando planejar ações para reduzir essas ocorrências. Dentre as ações realizadas pela instituição, estão o patrulhamento ostensivo dos trechos rodoviários, a aplicação de políticas públicas, a execução de campanhas educativas e a proposição de mudanças na estrutura viária das rodovias. A alocação dos recursos disponíveis, principalmente humanos, é determinante para o sucesso dessas ações.

A instituição conta com um efetivo de mais de 10.500 policiais (Portal da Transparência, 2022), que são distribuídos em 364 Unidades Operacionais (UOPs[2]). A malha viária de

[1] Acrônimo de rodovia federal. Um exemplo é a BR-116, que liga a cidade de Fortaleza na região Nordeste, à cidade de Jaguarão na região Sul, com 4.660 quilômetros de comprimento e passando por 10 estados brasileiros.
[2] Postos de trabalho onde os policiais são escalados para o serviço.

responsabilidade da PRF é de 75,5 mil quilômetros de rodovias e estradas federais, que ligam os diversos estados do país (Ministério da Infraestrutura, 2022).

Devido a diversos fatores que reduzem o efetivo policial disponível para a área operacional, como alocação de uma parte dos policiais na área administrativa, férias e licenças, além da divisão do efetivo operacional em escalas de serviço, uma UOP tem em média três policiais trabalhando diariamente (dia e noite).

A doutrina policial da instituição rege que uma viatura operacional ostensiva (caracterizada com seus sinais identificadores) deve conter ao menos dois policiais quando em atendimento a uma ocorrência. Por isso a grande maioria das UOPs espalhadas pelo país contam apenas com uma viatura operacional disponível para o patrulhamento diário do trecho rodoviário sob sua circunscrição. Isso torna o posicionamento das viaturas operacionais uma ação primordial na missão de reduzir a quantidade de ocorrências de AT.

1.2 Problema

A identificação do local ideal para se posicionar uma viatura policial é primordial para a instituição. Atualmente essa identificação é feita por análise estatística e experiência profissional. A análise estatística é baseada nos dados históricos disponíveis nos sistemas da instituição, tratados com filtros que levam à elaboração de gráficos e mapas de calor. Já a experiência profissional se alicerça na experiência e subjetividade do chefe responsável pela unidade operacional, geralmente um policial com grande conhecimento do trecho rodoviário e suas peculiaridades.

Atualmente existem alguns estudos que visam auxiliar na melhoria da identificação dos locais com maiores ocorrências de AT. Em um dos mais recentes, Oliveira e Gomes (2021) utilizam um método estatístico ponderado para identificar os locais com as maiores concentrações de sinistros. Segundo os autores, esse método permitiria um melhor planejamento por parte dos gestores públicos.

Conforme o seu Plano Diretor de Tecnologia da Informação e Comunicação 2020|2023, nas ações A02, A12 e A58 (Polícia Rodoviária Federal, 2021), a PRF definiu como um dos seus objetivos a melhoria de seu método de identificação de locais de AT, com o desenvolvimento de um novo sistema informático para esse fim. A instituição acredita que a utilização de técnicas de Inteligência Artificial pode aprimorar sua capacidade de predizer os locais de ocorrências de AT. Porém não há nenhum estudo que confirme que, de facto, a utilização das técnicas de ML aplicados aos seus dados históricos é superior ao método que atualmente utiliza.

1.3 Objetivo

O objetivo dessa dissertação de mestrado é realizar um estudo sobre a utilização de diversas técnicas e algoritmos de Aprendizagem Automática (*AA*) na identificação dos locais com maior probabilidade de ocorrências de AT, e analisar suas capacidades preditivas.

Os estudos atuais que visam identificar os locais críticos de acidentes são baseados em técnicas estatísticas. Alguns utilizam métodos de mapas de calor e outros utilizam métodos de acumulação ponderada, baseada na gravidade do acidente. O sistema atual da PRF utiliza o método de mapas de calor, identificados a partir do local do acidente (marco quilométrico da ocorrência), dia e hora como parâmetros para o cálculo. Mas esse método não considera todas as variáveis envolvidas em um AT, podendo não identificar todas as possíveis correlações entre elas.

A utilização de técnicas de ML poderia considerar outras características envolvidas (e. g. a condição climática, se a ocorrência foi em um dia de feriado, o tipo de traçado da via, o tipo de acidente e uma localização mais precisa[3], com latitude e longitude), suas diferentes correlações e magnitudes. Isso possibilitaria uma nova perspetiva sobre os sinistros, com a indicação de quais elementos exercem maior influência em uma ocorrência.

Os objetivos específicos deste trabalho são:

- Aplicação de diversos algoritmos de ML nos dados históricos de AT da PRF.
- Comparação e escolha de um algoritmo baseada em sua capacidade preditiva.
- Análise crítica da escolha e utilização de técnicas de ML em detrimento das técnicas utilizadas atualmente pela instituição.
- Comparação dos resultados encontrados com trabalhos anteriores realizados na área.

Esses objetivos serão desenvolvidos através: da revisão do estado da arte sobre os algoritmos e técnicas utilizados na área de ML; do estudo sobre os acidentes de trânsitos nas rodovias federais brasileiras; do estudo dos trabalhos internacionais sobre a utilização de ML na predição de locais de AT; do tratamento dos dados históricos de AT a ser utilização nas análises; da definição dos indicadores de desempenho necessários para avaliar algoritmos; da identificação do modelo de ML mais adequado; da avaliação dos resultados obtidos; da comparação com os resultados de outros trabalhos desenvolvidos; da avaliação dos objetivos propostos e conclusão do trabalho.

[3] em 2017 a PRF iniciou o registo da geolocalização da ocorrência em seu Boletim de Acidente de Trânsito (BAT)

1.4 Estrutura do Documento

Esse documento está estruturado da seguinte forma:

- Estado da arte: contém a revisão do estado da arte dos elementos utilizados no estudo, incluindo ferramentas e *frameworks*, além de definição de conceitos utilizados durante esse trabalho.
- Análise de valor: análise do estudo apresentado por essa dissertação, objetivando medir o valor que um trabalho dessa natureza acrescenta à tentativa de resolução do problema apresentado, se indo de encontro à suas necessidades e demandas. Proposição de uma hipótese a ser verificada.
- Metodologia: definição um plano de avaliação para a hipótese a ser testada, definindo as medidas e métricas necessárias para a validação do resultado obtido pela análise realizada.
- Estudo computacional: aplicação de técnicas para preparar os dados para serem utilizados pelos diversos algoritmos selecionados. Aplicação das técnicas de ML, realizando uma análise sobre seu progresso e resultado. Avaliação dos resultados obtidos e comparação com os esperados, e suas implicações em relação aos objetivos propostos.
- Conclusão: análise e avaliação crítica do resultado obtido, verificando se ele solucionou o problema identificado e se responde satisfatoriamente as questões propostas. Descrição das contribuições do trabalho realizado e suas limitações, indicando ainda possíveis melhorias e possibilidades de trabalhos futuros.

Para o resto desse documento o termo “rodovias” significará rodovias e estradas federais do Brasil.

1.5 Sumário

Neste capítulo foram introduzidos o contexto, a motivação, os objetivos e a estrutura deste documento. A PRF precisa melhorar seu sistema de identificação de locais de acidentes, e ela acredita que a utilização de Aprendizagem Automática pode ser a melhor opção para isso. O objetivo deste trabalho é estudar a aplicação de algoritmos de ML e DL nos dados históricos de AT da PRF e avaliar se eles têm uma predição satisfatória. Este documento é composto pelos capítulos: Introdução, Estado da Arte, Análise de Valor, Metodologia, Estudo Computacional e Conclusão.

2 Estado da arte

Neste capítulo é realizada a revisão do conhecimento produzido nas áreas relacionadas com o trabalho: acidentes em rodovias federais brasileiras, *Machine Learning* (ML) e tecnologias correlatas e o uso de ML na identificação de locais de AT.

2.1 Acidentes de Trânsito em Rodovias Federais

Essa seção trata de diversos estudos sobre acidentes de trânsito (AT) em rodovias, suas causas e prevenções.

Um acidente de trânsito é um evento não intencional que ocorre em uma via pública envolvendo um ou mais veículos, podendo envolver também pedestres. Pode resultar em danos físicos ou materiais, incluindo a morte de algum envolvido. É um evento aleatório, que acontece pelo inter-relacionamento de diversos fatores presentes (Polícia Rodoviária Federal, 2015).

Segundo a Confederação Nacional do Transporte (CNT), em seu painel anual de acidente rodoviários, a principal causa de AT em 2021 foi “reação tardia ou ineficiência do condutor”, seguida por “velocidade incompatível” (Confederação Nacional do Transporte, 2022). Existem diversos estudos que tentam relacionar as causas de AT com outros elementos envolvidos, procurando identificar elementos que possam ser utilizados na sua prevenção.

Lima (2021) identifica que o excesso de velocidade é um dos principais fatores envolvidos em AT, mas que a utilização de câmeras de controle de velocidade está associada a uma redução do número e severidade das ocorrências já no primeiro trimestre após a sua instalação. Ele sugere que as rodovias com pista simples se beneficiam mais das câmeras de velocidade do que as com pistas múltiplas, ao dissuadir os condutores a não realizarem manobras de ultrapassagens irregulares. Sugere ainda que as câmeras são mais efetivas em vias rurais, porque elas carecem de outras formas de fiscalização e são onde os motoristas desenvolvem velocidades mais elevadas.

Em Lima (2021), é indicada uma redução de 4,7% no número de acidentes nos primeiros três meses após a instalação das câmeras, com uma redução de 7,8% após esse período. Nos AT com vítimas, a redução da ocorrência é de 5% após os primeiros três meses de instalação. Nos AT com vítimas fatais, há uma redução de 1,44% nas ocorrências de após o segundo semestre de monitoramento.

Apesar dos estudos indicarem que as principais causas de AT advêm de atitudes humanas, o estudo "Acidentes rodoviários e a infraestrutura" (Confederação Nacional do Transporte, 2018) procura demostrar que as deficiências existentes na infraestrutura das rodovias contribuem para que os acidentes ocorram ao influenciar o comportamento e a tomada de decisões dos condutores. Ele foi realizado com os dados de AT em rodovias federais de 2007 a 2017. Neste estudo, a CNT demonstra que a principal causa de acidente registada nos boletins de ocorrência do período foi "falta de atenção do condutor", representando 44,6% do total de AT com vítimas. Mas ela pondera que esse dado poderia ser refutado ou relativizado, pois alguns indicadores demonstram que as condições da infraestrutura da rodovia podem ser as verdadeiras indutoras das causas de ocorrências de acidentes que são atribuídas ao comportamento dos motoristas. Alguns desses indicadores são (Confederação Nacional do Transporte, 2022):

> • "Problemas de Sinalização nos locais de acidentes com indicação de causa a falta de atenção de condutores - Nos trechos com sinalização classificada como ótima, a gravidade é bem menor, sendo registados 5,3 óbitos a cada 100 acidentes. Comparativamente, o grau de severidade nos trechos avaliados como possuindo sinalização péssima (10,7 mortes por 100 acidentes) é 2 vezes maior do que nos trechos em que a sinalização é considerada ótima;
> • Presença de Placas de Limite de Velocidade nos locais de acidentes com indicação de causa a falta de atenção de condutores - Analisando os trechos segundo a presença ou ausência de placas de limite de velocidade, é percetível que, onde elas estão ausentes, o índice de acidentes chega a 18,3 óbitos por acidente, sendo 2,6 vezes maior do que nos trechos onde há placas regulamentadoras de velocidade (7,0)."

Andrade e Antunes (2019) estudaram o impacto da Década de Ação pela Segurança no Trânsito (DAST) nos acidentes em rodovias. Esse foi o período de 2011 a 2020 em que a ONU definiu objetivos de redução dos óbitos e lesões em acidentes no trânsito mundial, por meio de um Plano de Ação Global para a Década (PAGD). O plano continha as ações que seriam realizadas pelos países, coordenadas em diferentes níveis. No Brasil, as ações do PAGD foram implementadas através do Pacto Nacional pela Redução de Acidentes de Trânsito e também pelo Projeto Vida no Trânsito.

Para medir o impacto do PAGD, os autores utilizaram técnicas de séries temporais interrompidas aplicadas nos dados de acidentes rodoviários com vítimas entre os anos de 2007 e 2017. Eles constataram que a tendência de crescimento mensal no número de mortos e feridos que ocorria até o ano de 2011 foi interrompida. O aumento mensal de mortos e feridos em acidentes era de 0,71% VPM (variação percentual mensal), e após o PAGD houve o início de uma tendência inversa, com um decréscimo na taxa de -1,24% VPM. Eles concluem que, mesmo com o aumento da população e da frota de veículos, e ainda que o número de AT tenha permanecido elevado, a tendência de redução do número de mortos e feridos em

rodovias está em conformidade com o objetivo da DAST, indicando o impacto favorável e benéfico da iniciativa.

Silveira (2009), realizou um estudo muito interessante para tentar identificar um valor de Disposição a Pagar para se evitar um novo acidente. Por meio de uma pesquisa do tipo *survey*, ele estimou uma função de demanda para a redução do risco de AT através da disposição a pagar dos usuários das rodovias para se evitar o risco de mais um acidente. De forma mais prática, isso significa o quanto um condutor está disposto a pagar a mais em imposto de trânsito a fim de tentar evitar mais um acidente. Impressionantes 75% dos pesquisados não estão dispostos a pagar a mais em taxas para evitar o risco de mais um AT, sob as justificativas de "excessiva carga tributária cobrada no Brasil e o pensamento de que o dinheiro arrecadado seria utilizado na corrupção".

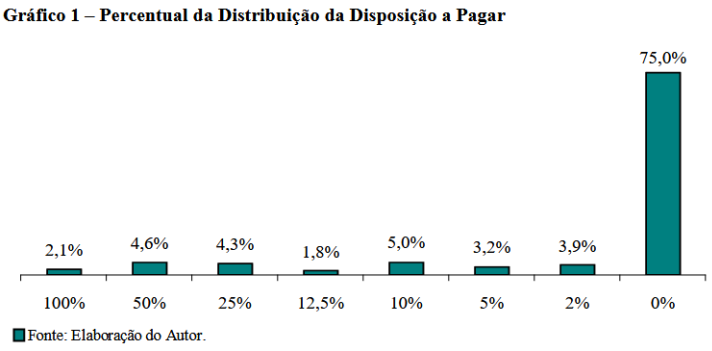

Figura 2 - Distribuição da Disposição a Pagar a mais em taxas para tentar evitar um novo acidente (Silveira, 2009)

Em um trabalho que busca identificar os fatores associados à letalidade dos AT ocorridos em rodovias no ano de 2016, Barroso Júnior, Bertho e Veiga (2019) utilizaram um modelo binomial de regressão logística que considera as informações sobre o contexto dos eventos ocorridos. A partir dos dados de AT da PRF, eles selecionaram 13 variáveis e modificaram-nas de forma a se adequarem melhor ao modelo empregado, ficando com 11 variáveis explicativas. Essas variáveis explicativas foram inseridas uma a uma no modelo e, as que não tinha uma significância de ao menos 5% eram retiradas do modelo, repetindo isso até que todas fossem testadas.

Os resultados encontrados pelos autores corroboram com resultados já consolidados na literatura em diversos aspetos. Alguns deles são: é maior a taxa de letalidade em AT entre os homens em relação às mulheres; as probabilidades de morte são mais elevadas para os idosos; e é maior o risco de morte para os pedestres, seguido dos ciclistas e dos ocupantes de motocicleta, em comparação aos ocupantes de automóveis. Outros pontos identificados foram que as chances de um acidente ser letal são maiores aos finais de semana e nos períodos da noite e madrugada, e que há uma maior letalidade nos acidentes de trânsito ocorridos em áreas rurais em relação aos ocorridos em áreas urbanas. Esses resultados vão de encontro aos dados apresentados pela Confederação Nacional do Transporte (2022) em seu painel de acidentes rodoviários. Os autores concluem que:

> "… que tanto as características de contexto dos acidentes de trânsito quanto as características de suas vítimas possuem grande influência na letalidade destes eventos. Tal resultado enriquece o debate sobre as mortes por esta causa, tantas vezes marcado exclusivamente pela culpabilização das vítimas e pouco por condições exógenas aos usuários das rodovias. É de suma importância conscientizar pedestres, condutores e passageiros sobre comportamentos perigosos no trânsito, tais como a associação entre álcool e direção, o não uso de dispositivos de segurança (como cinto de segurança, para passageiros de automóveis e caminhões, e capacetes, para motociclistas e ciclistas) e o uso de celular ao volante, fator que a cada ano ocasiona mais acidentes com vítimas fatais."

Teodoro, Alcantara e Barbosa (2014) decorrem que existem várias experiências brasileiras no tratamento dos dados de AT, mas que o grande problema está na coleta e qualidade desses dados, pois muitos apresentam inconsistências, reduzindo a possibilidade de utilizá-los. Para eles a identificação de locais críticos de AT é um dos modos de avaliação da segurança viária mais utilizados, permitindo criar um *ranking* com trechos e interseções que mais precisam de atenção, mas que a maioria dessas metodologias são trabalhosas e demandam tempo para compatibilizar os dados, apesar da baixa complexidade. O que eles querem responder é: "É possível trabalhar de uma forma mais simplificada sem perder a qualidade da informação?" Para responder à pergunta, os autores desenvolveram um trabalho que é comparar os resultados obtidos entre a metodologia para identificação de pontos críticos apresentada no manual PARE (Ministério dos Transportes, 2002) e uma metodologia por eles proposta, que utiliza o Estimador de Intensidade de Kernel. Os métodos foram aplicados em um estudo de caso de acidentes ocorridos na Av. Presidente Antônio Carlos, em Belo Horizonte, utilizando as informações do banco de dados de AT ocorridos em 2011, fornecido pela BHTrans (Empresa de Transportes e Trânsito de Belo Horizonte).

O método de identificação de locais críticos de AT do manual PARE é baseada na aplicação de uma técnica de severidade de acidentes, que leva em consideração o número de acidentes e a severidade deles por meio da associação de pesos a cada tipo de acidente, calculando-se assim um índice intitulado como Unidade Padrão de Severidade (UPS). Os pesos utilizados para cada tipo de acidente estão definidos no próprio manual PARE (Ministério dos Transportes, 2002) e são:

- Acidentes somente com danos materiais: 1
- Acidentes com feridos: 4
- Acidentes com feridos envolvendo pedestres: 6
- Acidentes com vítimas fatais: 13

O método proposto por Teodoro, Alcantara e Barbosa (2014) utiliza um Sistema de Informações Geográfica (SIG) e o Estimador de Intensidade de Kernel, uma técnica estatística espacial que demonstra as concentrações de um determinado evento. Eles interpolaram a localização com maior intensidade de ocorrências sobre bases cartográficas, ajustando-as com uma função bidimensional, compondo uma superfície que é proporcional à intensidade de amostras por unidade de área.

Como resultado, todos os trechos considerados críticos pelo método PARE ficaram evidentes também no método proposto pelos autores. Quando considerados trechos maiores da

rodovia, o método proposto tende a apresentar o local crítico com mais precisão, pois mostra onde ocorreu o pico do somatório das UPS, diferentemente do método PARE que divide a média de UPS ao longo da extensão do trecho. Isso demostra uma boa correspondência entre os modelos, o que já era esperado, pois ambos trabalham com o somatório das médias de severidade obtidas por ocorrência. Mas os autores salientam que a sua metodologia requer menos passos e tem a possibilidade de ser automatizada, tonando-se de uma opção interessante para a gestão da informação de AT pelos tomadores de decisão.

Silva Júnior (2019) cita que um dos métodos mais utilizados para a análise de dados em estudos de distribuição espacial é o mapa de calor. Esse método desenha uma mancha colorida no mapa de forma a representar a concentração espacial das ocorrências observadas, onde o vermelho significa o local com maior ocorrência. Ele comenta que sua interpretação é subjetiva, apesar da boa receção e utilização, e se propõem a desenvolver um software para gerar mapas dos pontos críticos de AT utilizando o geoprocessamento dos dados de AT, permitindo a exploração visual e análise de suas características.

Também utilizando o método de Estimativa de Densidade Kernel (EDK) para a identificação dos pontos críticos de AT, ele coloca uma função Kernel simétrica em cada ponto observado, gerando uma superfície de intensidade suavizada, onde a intensidade é o somatório de todas as superfícies de ocorrências sobrepostas. Segundo o autor, “uma das vantagens do método EDK comparado com outros métodos de clusterização é que ele leva em consideração a autocorrelação espacial das colisões. Além disso, este método é simples e menos computacionalmente intensivo, sendo uma das razões de sua popularidade.” O estudo realizado também utiliza ponderação para o cálculo da Unidade Padrão de Severidade (UPS), com os mesmos valores dos pesos utilizados para cada tipo de acidente mencionados anteriormente (Teodoro, Alcantara e Barbosa, 2014).

Como resultado é salientado neste trabalho que a EDK permite potencializar a utilização das coordenadas geográficas dos dados de AT, incorporando nos mapas uma zona de risco através de uma superfície contínua de densidade, sendo mais realista que os métodos que utilizam intervalos espaciais arbitrários, como os que utilizam os marcos quilométricos das rodovias. Estes criam segmentos espaciais artificiais para serem utilizados nos estudos, enviesando os resultados obtidos.

Oliveira e Gomes (2021) também utilizaram o método do manual de procedimentos PARE (Ministério dos Transportes, 2002) para a identificação de locais de acumulação de AT em rodovias. Eles citam que os autores Teodoro, Alcantara e Barbosa (2014) realizaram um estudo comparativo entre esse método e um método que utiliza uma técnica de estatística espacial com a utilização do Estimador de Intensidade de Kernel, concluindo que os resultados de ambos são convergentes. Dessa forma, Oliveira e Gomes (2021) acreditam que, apesar do método do programa PARE ser um método numérico absoluto, ele produz resultados confiáveis no que se refere à identificação de locais de acumulação de AT.

Utilizando os mesmos pesos para os tipos de acidentes dos trabalhos anteriores, e os dados de AT em rodovias da PRF de 2017 e 2018, eles apontam que houve uma redução de 0,63% na

média nacional da Unidade Padrão de Severidade (UPS) de 2018 em relação à 2017, quando o valor da UPS foi 30,32. Argumentam ainda que, a partir da verificação das UPS, é possível identificar os locais de acumulação de AT nas rodovias a serem usados na formulação, monitoramento e aplicação das políticas públicas de redução de AT.

Martins (2020) argumenta que seu trabalho se diferencia dos demais por utilizar um modelo multicritério e multiperíodo, dando nova abordagem ao problema. Seu objetivo é criar um Sistema de Apoio a Decisão (SAD) que auxilie na identificação dos trechos mais críticos de AT. Segundo ele, "problemas de decisão multicritério consistem em problemas nos quais um decisor precisa avaliar múltiplos objetivos de forma integrada; onde os objetivos são representados por variáveis que podem estar em unidades de medida diferentes". Para isso é necessário ter um conjunto de alternativas (pelo menos duas) para que o decisor possa comparar e escolher. O estudo faz uso da abordagem de sobreclassificação, utilizando assim uma racionalidade não-compensatória, onde não há perdas entre os critérios escolhidos, para ordenar os trechos de acordo com a criticidade de ocorrência de AT de cada um quando comparados aos outros. É utilizado o *Multicriteria Multi-Period Outranking Method*, que consiste em um método de tomada de decisão multicritério que usa índices de concordância e discordância e credibilidade, para uma configuração multiperíodo. Portanto, esse método considera os diversos cenários ao longo dos períodos atendendo as relações de preferências e ordens parciais. Os dados utilizados são os dados de AT da PRF no estado do Pernambuco de 2014 a 2018.

Como resultado ele argumenta-se que o SAD cumpriu com o objetivo de fornecer uma ferramenta de apoio a tomada de decisão sobre quais trechos críticos de AT se considerar, fornecendo um ordenamento da criticidade entre eles. E que a abordagem multicritério se mostrou apropriada ao problema, dado que a criticidade dos locais não deve ser analisada através de uma única dimensão. Ele considera ainda que a abordagem multiperíodo se mostra bastante útil por agregar mais informações aos resultados obtidos, deixando-os mais abrangentes.

Até à data de conclusão dessa dissertação não foram encontrados estudos brasileiros que utilizem técnicas de Aprendizagem Automática aplicados à dados de acidentes de trânsito.

2.2 Inteligência Artificial

A Inteligência Artificial (*Artificial Intelligence* - AI) é uma área antiga e ampla que pode ser entendida como "qualquer tentativa de automatizar tarefas intelectuais normalmente realizadas por humanos". Isso varia desde o básico, como uma planilha do Excel, até algo mais avançado, como um robô humanoide que pode andar e falar.

Durante muitos anos, muitos especialistas acreditavam que uma inteligência artificial do nível da humana seria alcançada pela programação manual de um número suficientemente grande de regras que manipulassem o conhecimento. Essa abordagem era conhecida como AI simbólica, e foi o paradigma predominante dos anos 50 aos anos 80. Apesar da comprovada

capacidade de resolver problemas lógicos, bem definidos, essa abordagem se mostrou incapaz de resolver problemas mais complexos e difusos, como a classificação de imagens, reconhecimento de fala e tradução de idiomas. Então uma nova abordagem tomou o lugar da AI simbólica: *machine learning* (Chollet, 2017).

2.3 Machine Learning

O campo da Aprendizagem Automática (*Machine learning* - ML) está preocupado com a questão de como construir programas de computador que melhorem automaticamente com sua própria experiência (Mitchell, 1997). É um subcampo da AI que visa o desenvolvimento automático de programas, chamados modelos, puramente pela sua exposição ao treinamento de dados. Esse processo de converter dados em um programa é chamado de aprendizado. Um sistema de ML é treinado, ao invés de programado. Ele é exposto ao uma grande quantidade de exemplos de uma tarefa, e descobre a estrutura estatística que permite descrever as regras de automação daquela tarefa. Apesar de ser bem antigo, só se tornou o campo predominante da inteligência artificial nos anos 2000 (Chollet, 2017).

ML utiliza algoritmos avançados para detetar padrões e automatizar processos sem necessidade de interação humana, ou com a mínima interação possível. São uma série de técnicas que tem como objetivo aprender a partir de dados históricos, usando a força computacional para melhor prever padrões, comportamentos, além de classificar e agrupar dados (Marsland, 2014).

Através do ML, os computados podem processar, aprender e desenhar relevantes perceções do lado fora dos muros impenetráveis do *big data*. Do supercomputador que suporta o motor de pesquisas do Google aos telemóveis que estão nos nossos bolsos, nós dependemos do poder do ML que está ao nosso redor, mesmo sem sabermos (Raschka, 2019).

Os algoritmos de ML podem ser classificados em quatro categorias: os que realizam uma aprendizagem supervisionada, os que realizam uma aprendizagem não supervisionada, os que realizam uma aprendizagem semi-supervisionada e os que realizam aprendizagem baseada em reforço.

2.3.1 Classificação das Técnicas de Aprendizagem Automática

As técnicas de AA podem ser classificadas em quatro tipos: aprendizagem supervisionada, aprendizagem não supervisionada, aprendizagem semi-supervisionada e aprendizagem por reforço.

2.3.1.1 Aprendizagem Supervisionada

O objetivo da aprendizagem supervisionada é aprender um modelo a partir de dados que são rotulados, permitindo realizar predições para dados ainda desconhecidos. O termo "supervisionado" se refere a um conjunto de dados onde seu resultado, sua saída, já é

conhecida, já está rotulada. Nesse processo, um conjunto de dados já rotulados é passado a um algoritmo de ML que é treinado, criando um modelo que passa a prever o rótulo de dados novos, ainda não rotulados. Uma tarefa supervisionada onde o resultado é um valor contínuo, como os quilômetros de uma rodovia, é uma tarefa de regressão. Uma tarefa onde o resultado são classes com rótulos discretos, é uma tarefa de classificação (Raschka, 2019).

2.3.1.2 Aprendizagem Não Supervisionada – Dados não rotulados

Na aprendizagem não supervisionada, o algoritmo analisa as diversas *features* dos dados e aprende a estrutura dos dados, e quando novos dados são introduzidos, ela utiliza o conhecimento adquirido anteriormente para classificar cada novo exemplo (Morgan, 2017). É utilizada para agrupamentos e reduções, sem a intervenção humana. Objetivo é ajustar e distribuir os dados a partir do treinamento adquirido com um conjunto de dados de treino, que contém diversas características, onde o modelo aprende as propriedades e a estrutura dessas características, identificando as correlações entre elas.

2.3.1.3 Aprendizagem Semi-Supervisionada

A aprendizagem semi-supervisionada é uma técnica que combina o poder da aprendizagem supervisionada e o da não supervisionada. O conjunto de dados contém elementos rotulados e não rotulados, geralmente com uma quantidade maior de elementos não rotulados, que são rotulados a partir do padrão aprendido anteriormente (Geron, 2019).

2.3.1.4 Aprendizagem por Reforço

A aprendizado por reforço está preocupada com o problema de encontrar ações adequadas a serem tomadas em uma determinada situação, objetivando maximizar uma recompensa. Nele o algoritmo não recebe nenhum exemplo de saída, como acontece no aprendizado supervisionado, mas deve descobri-la em um processo de tentativa e erro. Normalmente, há uma sequência de estados e ações em que o algoritmo de aprendizado está interagindo com seu ambiente. Em muitos casos, a ação atual não afeta apenas a recompensa imediata, mas também afeta a recompensa em todas as etapas de tempo subsequentes. Nele as ações do algoritmo são baseadas no resultado das ações anteriormente observadas, sendo elas positivas ou não. Com base no acerto anterior, a máquina tenta acertar novamente utilizando uma política aprendida, e melhorando essa política a cada novo acerto (Bishop, 2006).

Não há como classificar cada algoritmo de ML em somente uma categoria. Um algoritmo pode ser considerado de uma categoria em um projeto e de outra em outro projeto, dependendo da forma como foi utilizado.

2.3.2 Tipos de Problemas

Os problemas de AA podem ser do tipo de tarefas de regressão ou tarefas de classificação.

2.3.2.1 Regressão

A regressão é o tipo de tarefa em que o computador é solicitado a prever um valor numérico de um intervalo contínuo como saída. Para resolver essa tarefa, o algoritmo é solicitado a

resolver uma função do tipo $f : \mathbb{R}^n \rightarrow \mathbb{R}$. Um exemplo de tarefa de regressão é a predição do preço futuro de um imóvel (Goodfellow, Bengio e Courville, 2016). Alguns algoritmos podem ser utilizados tanto para regressão quanto para classificação.

2.3.2.2 Classificação

Nesse tipo de tarefa, o computador é solicitado a especificar em qual das k categorias um dado de entrada pertence. Para resolver isso, o algoritmo é solicitado a resolver uma função do tipo $f : \mathbb{R}^n \rightarrow \{1, \ldots, k\}$. Quando $y = f(x)$, o modelo atribui um elemento x do vetor de entrada à uma categoria identificada pelo código numérico y. Há outras variações de tarefas de classificação, como quando f atribui como saída uma probabilidade de distribuição sobre um conjunto de classes. Um exemplo de tarefa de classificação é o reconhecimento de um objeto, quando a entrada é uma imagem e a saída é um código numérico que identifica o objeto na imagem (Goodfellow, Bengio e Courville, 2016).

2.3.3 Algoritmos de AA

Neste seção são listados alguns algoritmos de Aprendizagem Automática que foram utilizados neste trabalho.

2.3.3.1 Decision Trees

Decision Trees (DT) é um algoritmo que aprende utilizando uma hierarquia de perguntas do tipo se/senão, levando a uma decisão. Para isso ele aprende uma sequência de perguntas que leve a uma resposta verdadeira o mais rápido possível. Essa perguntas são chamadas de testes. Se uma *feature* do conjunto de dados é do tipo binário, representada por sim ou não, os testes são facilmente verificados. Mas normalmente os dados não estão no formato binário, mas em formato contínuo, com representações numéricas. Nesses casos, os testes realizados pelo algoritmo são no tipo "a *features i* é maior que o valor *v*?" (Müller e Guido, 2016).

Cada DT é um gráfico em árvore que é utilizado para a tomada decisões. Em cada ramo do gráfico, uma característica *i* do conjunto de dados é examinada. Se o valor da característica é menor que um limite, o ramo esquerdo do nó é seguido, se não, o algoritmo irá pelo ramo direito. Quando o ramo chega a uma folha, uma decisão é tomada com base na classe ao qual aquela folha pertence (Burkov, 2019). Uma exemplificação das decisões tomadas por uma DT é visualizada na Figura 3, onde cada decisão de um nó leva a outro nó, até a decisão final. É um algoritmo de ML muito versátil, que pode ser utilizado tanto para tarefas de regressão quanto para tarefas de classificação.

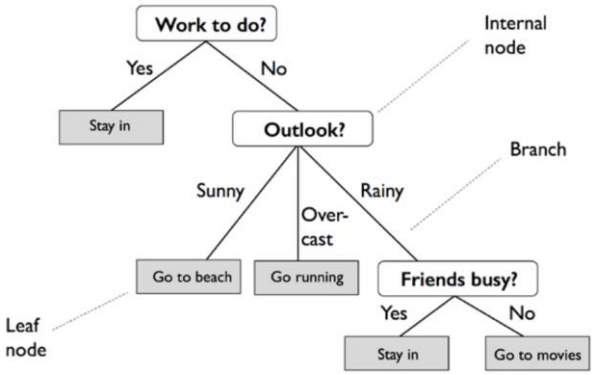

Figura 3 – Árvore de decisão de uma *Decision Trees* (Raschka, 2019).

2.3.3.2 Random Forest

Random Forest (RF) é um tipo de algoritmo que é um conjunto agregado de diversas *Decision Trees*, treinadas pelo método de *bagging*[4] com o híper parâmetro *max_samples* definido como o tamanho do conjunto de dados de treino. O algoritmo RF procura construir as árvores de decisão de forma mais randómica, ao invés de procurar a melhor característica ao fazer a divisão dos caminhos de um nó. Ele procura a melhor característica em um subconjunto de características. Isso leva a uma grande diversidade de árvores, gerando um modelo com melhor performance (Geron, 2019).

Uma das qualidades do algoritmo é que ele permite medir a relevância de cada característica na predição, ao observar o quanto cada nó da árvore reduziu a impureza na média dentre todas as árvores para aquela característica. É como se fosse uma média ponderada, onde o peso de cada nó é igual ao número de amostras do treinamento que estão associadas aquele nó (Geron, 2019).

O algoritmo pode ser aplicado a uma grande gama de problemas. Foi o algoritmo de ML mais popular do sítio de competição de ML Kaggle (Kaggle, 2022) entre os anos de 2010 e 2014 (Chollet, 2017).

2.3.3.3 Support Vector Machine

O *Support Vector Machine* (SVM) é um algoritmo poderoso e versátil, capaz de realizar classificações lineares e não lineares, regressões e até deteção de valores discrepantes. É um dos modelos mais populares em ML e atende bem a classificação de conjunto de dados complexos, que não sejam muito extensos (Geron, 2019).

O algoritmo SVM tenta resolver o problema de classificação procurando uma fronteira entre dois agrupamentos de pontos que pertencem a categorias diferentes. Essa fronteira pode ser interpretada como uma linha ou superfície que separa os dados de treino em dois espaços correspondentes a duas categorias. Para classificar um novo ponto basta verificar em qual

[4] *Bagging* é método de predição que utiliza um conjunto agregado do mesmo algoritmo que é treinado em diversos subconjuntos randômicos do conjunto de dados de treino, com o objetivo de melhorar a performance do algoritmo original

lado da fronteira ele se enquadra melhor. O algoritmo encontra essa fronteira em dois passos: primeiro os dados são mapeados em uma representação multidimensional onde essa fronteira pode ser expressa por meio de um híper plano; depois a fronteira é computada tentando maximizar a distância entre o híper plano e os pontos mais próximos de cada classe. Isso permite que a fronteira possa ser generalizada em amostras fora dos dados de treino (Chollet, 2017).

2.3.3.4 Gradient Boosting

Gradient Boosting é outro método agregado que combina um conjunto de múltiplas DT para criar um modelo mais poderoso, e pode ser utilizado tanto para regressão como para classificação. Ao contrário da abordagem da RF, ele constrói árvores de decisão de uma forma serial, tentando corrigir os erros da árvore anterior (Müller e Guido, 2016).

O algoritmo utiliza diversas árvores rasas, com nível de 1 a 5 em profundidade, o que torna o modelo menor em termos de uso de memória, com uma rápida predição. Cada árvore só pode prover boa previsão em uma parte dos dados, então o algoritmo acrescenta mais e mais árvores, como forma de melhorar a performance.

Gradient Boosting é frequentemente vencedor em competições de ML, e muito empregado na indústria. Ele geralmente é mais sensível a configurações de parâmetros do que uma RF, mas pode prover melhor assertividade se os parâmetros forem bem configurados (Müller e Guido, 2016).

Uma otimização do *Gradient Boosting* foi implementada na biblioteca *XGBoost*, que significa *Extreme Gradient Boosting*. Ela é extremamente rápida e portátil, com grande escalabilidade, e é uma ferramenta utilizada por diversos vendedores de competições de ML (Geron, 2019).

2.3.3.5 Ridle Regression

Regularização é uma abordagem utilizada para minimizar o problema de *overfitting*[5] adicionando mais informações ao modelo, com o objetivo de diminuir os valores dos parâmetros, induzindo uma penalidade contra a complexidade (Raschka, 2019). Um dos algoritmos que utiliza a regularização é o *Ridle Regression.*

Ridle Regression (também conhecido como *Tikhonov regularization*) é um tipo de regressão linear em que os coeficientes são escolhidos não somente para se ajustar bem aos dados de treino, mas também para se ajustar a restrições adicionais. É desejável que os coeficientes tenham o menor valor possível, para que cada *feature* tenha a menor influência possível no resultado, mas ainda mantendo uma boa predição. O objetivo é restringir o máximo possível o modelo para evitar *overfitting* (Müller e Guido, 2016).

[5] *Overfitting* é quando um modelo se ajusta bem aos dados de treino, mas não se ajusta bem aos dados de teste ou novos dados desconhecidos

2.3.3.6 Lasso Regression

Um outro tipo de regressão linear que utiliza regularização é o Lasso (*Least Absolute Shrinkage and Selection Operator Regression)*. Assim como o *Ridle*, o Lasso restringi os coeficientes para um valor próximo de zero, mas de uma forma diferente, no Lasso alguns coeficientes são reduzidos a exatamente zero. Isso significa que algumas características são completamente ignoradas pelo modelo. Isso pode ser visto como uma forma automatizada de seleção de características. Com algumas características sendo exatamente zero, o modelo se torna mais fácil de interpretar, revelando suas características mais importantes (Müller e Guido, 2016).

2.3.3.7 Elastic Net

Elastic Net é um algoritmo no meio termo entre o *Ridle* e o Lasso. Os termos da sua regularização é uma mistura dos termos da regularização do *Ridle* e do Lasso, e é possível controlar o nível dessa mistura por meio do híper parâmetro *r*. Se *r* é igual a 0, o algoritmo *Elastic Net* é equivalente ao *Ridle*, se é igual a 1 é equivalente ao Lasso (Geron, 2019).

2.3.3.8 k-Nearest Neighbors

O *k-Nearest Neighbors* (KNN) é um dos algoritmos mais simples. A construção do seu modelo consiste apenas no armazenamento dos seus dados de treino. Para realizar uma predição de um novo ponto, ele localiza o ponto mais próximo dentro dos dados de treino, o "vizinho mais próximo". Em sua versão mais simples, o KNN apenas considera exatamente o vizinho mais próximo, que é o ponto dos dados de treino mais próximo do ponto do qual se quer a previsão, retornando esse ponto dos dados de treino como previsão.

Mas ao invés de considerar o vizinho mais próximo, pode-se escolher um número *k* arbitrário de vizinhos. Então o algoritmo utiliza um sistema de votos para designar um rótulo. Para cada ponto testado, ele conta quantos vizinhos pertencem à classe 0 e quanto vizinhos pertencem à classe 1. Então o algoritmo designa uma classe ao mais frequente, ou seja, a maior classe dentre os *k* vizinhos mais próximos (Müller e Guido, 2016).

O algoritmo pode ser utilizado para tarefas de regressão como para tarefas de classificação.

2.3.3.9 Logistic Regression

Uma grande parte da análise de dados se resume a uma pergunta simples: isto é "A" ou "B"? Isto é "positivo" ou "negativo"? Essa pessoa é um cliente em potencial ou não é um cliente em potencial? Em AA essas questões são abordadas por meio de equações logísticas, principalmente através do uso da função *sigmoid*. A função *sigmoid* produz uma curva em formato de S, que pode converter qualquer número ou mapa em um valor entre 0 e 1, mas fazendo isso sem alcançar esses limites exatos.

Uma utilização comum da função *sigmoid* é encontrada no algoritmo *Logistic regression*. Ele adota essa função para analisar os dados e predizer classes discretas. Apesar do algoritmo compartilhar semelhanças visuais com a regressão linear, é tecnicamente um algoritmo de classificação. Enquanto a regressão linear trabalha com formas e equações numéricas para

prever as relações entre as variáveis, o algoritmo *Logistic Regression* predizer classes discretas (Theobald, 2017).

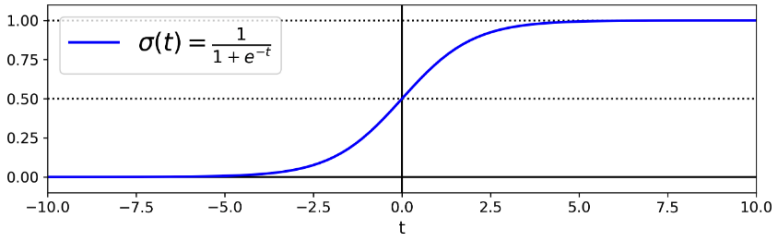

Figura 4 – Gráfico da função logística (Geron, 2019)

2.3.3.10 AdaBoost

Uma forma de um novo preditor corrigir os erros do preditor anterior é prestar mais atenção nas instâncias de treino em que o anterior cometeu *underfitting*[6]. Isso resulta em uma predição foca cada vez mais nos casos mais difíceis. Essa é a técnica utilizada pelo algoritmo *AdaBoost*.

Por exemplo, para construir um classificador *AdaBoost*, o primeiro classificador é treinado e utilizado para realizar uma predição nos dados de treino. O peso relativo das classificações erradas então é aumentado. Então um segundo classificador é treinado utilizando os pesos atualizados, realizando assim sua predição, aumentando novamente o peso das predições erradas. E assim o ciclo continua. Quando todos os preditores são treinados, o algoritmo realiza o agrupamento das predições utilizando o método *bagging*, excetuando os preditores que têm pesos diferentes da precisão geral do conjunto de treinamento ponderado (Geron, 2019).

2.4 Deep Learning

Deep Learning (DL) é um dos ramos de *Machine Learning*, onde os modelos são grandes cadeias de transformações geométricas, aplicadas uma após a outra. Essas operações são estruturadas em módulos chamados camadas: modelos de DL são tipicamente uma pilha de camadas, ou mais genericamente, são gráficos de camadas. Essas camadas são parametrizadas por pesos, que são os parâmetros de aprendizado durante o treinamento dos dados. O conhecimento de um modelo é armazenado nesses pesos, e o processo de aprendizado consistem em encontrar “bons valores” para esses pesos, valores esses que devem minimizar uma função de perda. Como a cadeia de transformações geométricas é

[6] *Underfitting* é quando um modelo não consegue aprender as relações mais importantes entre as *features*, ficando abaixo do esperado.

diferenciável, a atualização desses pesos utilizados para minimizar a função de perdas é feita de uma forma eficiente, via *gradient descent*[7] (Chollet, 2017).

2.4.1 Artificial Neural Networks

As *Artificial Neural Networks* (ANN) são o ponto de partida do *Deep Learning*. Elas são versáteis, poderosas e escaláveis, tornando-as ideais para abordar grandes e complexas tarefas de *Machine Learning*, como classificar imagens, suportar serviços de reconhecimento de voz e ainda recomendar vídeos para milhares de espectadores todos os dias (Geron, 2019).

O bloco de montar que é o núcleo de uma ANN é a camada, um módulo de processamento de informações que pode ser descrito como um filtro de dados. Alguns dados entram na camada e saem de uma forma mais útil. As camadas extraem representações dos dados que recebem, representações que tem mais significado para o problema abordado. A maioria dos modelos de DL consistem em cadeias de camadas simples de processamento que juntas implementam uma forma progressiva de reduzir os dados. Esses modelos são como uma peneira de dados, compostos de uma sucessão de filtros de dados cada vez mais refinados (Chollet, 2017).

Uma ANN é composta dos seguintes elementos: as camadas, combinadas em uma rede (ou modelo); os dados de entrada e os seus alvos correspondentes; uma função de perda, que defini o sinal de retorno para o aprendizado; e um optimizador, que determina como o aprendizado prossegue.

Na Figura 5 pode-se observar as iterações de uma ANN: a rede, composta pelas camadas que se encadeiam, mapeando os dados de entrada para uma predição. A função de perda então compara as predições com os alvos, produzindo um valor de erro: a medida do quanto as predições da rede batem com o esperado. Então o optimizador utiliza esse erro para atualizar os pesos da rede (Chollet, 2017).

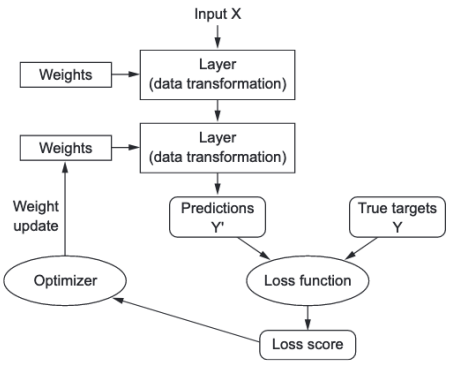

Figura 5 – As relações entre a rede, as camadas, a função de perda e os optimizadores de uma ANN. (Chollet, 2017)

[7] *Gradient descent*, de forma simplificada, é o método que permite ao algoritmo calcular tanto a direção quanto o montante que ele deve utilizar para atualizar os pesos e reduzir o erro (Trask, 2019).

2.4.2 Convolutional Neural Networks

As *Convolutional Neural Networks* (CNN) são um tipo de rede neuronal especializada em processar dados com uma topologia do tipo quadrante. Alguns exemplos são dados do tipo séries temporais, que podem ser visualizados como quadrantes de uma dimensão com intervalos regulares de tempo, ou dados do tipo imagem, que são quadrantes de *pixels* de duas dimensões. As CNN têm tido enorme sucesso em aplicações práticas. O nome "convolucional" indica que as redes aplicam uma operação matemática chamada convolução[8]. CNNs são redes neuronais que utilizam, em ao menos uma camada, a operação de convolução no lugar da multiplicação normal de matrizes (Goodfellow, Bengio e Courville, 2016).

O bloco de construção mais importante de uma CNN é a camada convolucional. A primeira camada convolucional não é conectada à cada *pixel* da imagem processada, mas apenas aos *pixels* do seu respetivo subquandrante, como na Figura 6. Da mesma forma, a segunda camada é conectada aos respetivos subquadrantes da primeira camada. Essa arquitetura permite a rede se concentrar em pequenas características de baixo nível na primeira camada, para depois montá-las em grandes características de alto nível na próxima camada, e assim por diante. Essa estrutura hierarquizada é comum em imagens do mundo real, motivo pelo qual as CNNs são tão eficazes em reconhecimentos de imagens.

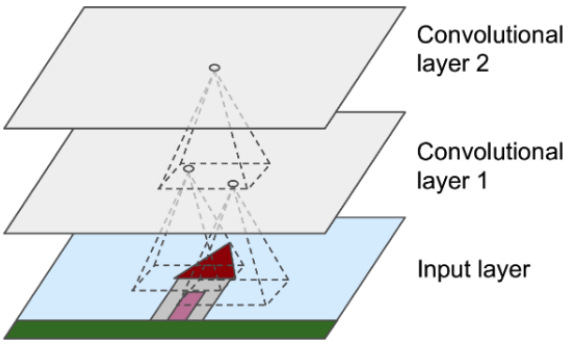

Figura 6 – Camadas de uma CNN conectadas ao seus respetivos subquadrantes (Geron, 2019).

As CNNs têm sido utilizadas para o reconhecimento de imagens desde os anos 1980. Nos últimos anos, graças ao crescimento do poder computacional e ao grande volume de dados disponíveis para treinamento, elas conseguiram alcançar uma performance significativa no campo das análises visuais complexas. Elas suportam os serviços de busca por imagens, carros autônomos, sistemas automáticos de classificação de vídeos, dentre outros, além de tarefas de reconhecimento de voz e processamento de linguagem natural (Geron, 2019).

[8] Convolução é uma operação matemática que sobrepõe uma função sobre outra e calcula a integral da multiplicação entre elas. As camadas convolucionais atualmente utilizam correlação cruzada, que é muito similar à convolução (Geron, 2019).

2.5 Utilização de ML na Análise de Acidente de Trânsito

Na literatura internacional, existem diversos exemplos da aplicação de técnicas de ML em estudos sobre acidentes de trânsito (AT). A grande maioria estuda as relações entre as causas e as ocorrências de AT, outros estudam a relação entre as ocorrências de sua severidade, mas são poucos os que tentam identificar precisamente o local de ocorrência de um novo AT. Alguns deles serão descritos nessa seção. As fontes de dados desses estudos não serão mencionadas pois não são relevantes para a análise dos modelos, algoritmos e resultados encontrados.

Utilizando dados de AT ocorridos na região de Setúbal, Portugal, Santos et al. (2021) utilizaram técnicas não supervisionadas de ML para classificar e identificar relações entre os diversos fatores relacionados aos acidentes, revelando por exemplo, que AT com motocicletas ou envolvendo pedestres tem mais chance de resultarem em feridos. Outro objetivo do estudo foi utilizar ML para predizer locais de AT, com dados que continham informações como local, data, hora, tipo de acidente, condição climática, dentre outras. Para realizar a análise, os dados de ocorrência foram completados com dados de não ocorrências de AT, gerados aleatoriamente e compreendendo todos os elementos das variáveis envolvidas, totalizando três vezes o tamanho dos dados que continham o rótulo pretendido, que é a ocorrência de um AT. Foram utilizados os algoritmos de *Logistic Regression, Random Forest e Decision Trees*, tendo o RF o melhor resultado, com uma assertividade de 73% e índice de falso negativo de 3%. Este é o estudo que mais se assemelha à proposta dessa dissertação.

Wenqi, Dongyu e Menghua, 2017) realizaram um estudo para provar que, para a predição da ocorrência de um AT, um modelo baseado em CNN é mais preciso e eficiente que um baseado em redes de regressão. O estudo considerou diversos fatores de acidente, muitos deles relacionados a velocidade, volume de veículos e ocupação da via, além de iluminação e condição climática, e demonstrou que o modelo baseado em CNN apresentou uma assertividade de 78,5%, 7,7% superior ao modelo de redes de regressão.

Um estudo de Yuan, Zhou e Yang (2018) utilizou uma *Convolutional Long Short-Term Memory Neural Network* (LSTM) para analizar um conjunto de dados muito heterogêneos, composto por volume de tráfego, volume de chuva, temperatura e imagens de satélite, dentre outros, objetivando prediser os locais de ocorrência de novos AT. Os resultados mostram uma boa capacidade de predição do modelo, incluía um estudo de caso que realizava a predição de AT para o dia 23 de dezembro de 2013 no estado de Iowa, EUA. Na vida real, nesse dia houve uma tempestade de neve no estado, e o modelo encontrou diversos dos 49 acidentes ocorridos naquele dia.

2.6 Tecnologias e Frameworks

Como o principal objetivo deste trabalho é realizar um estudo da aplicação das técnicas de ML nos dados históricos de AT da PRF, é importante conhecer as tecnologias que estão

disponíveis atualmente, que são consideradas estado da arte e que permitam explorar ao máximo as possibilidades de resposta às necessidades do cliente. Além do conhecimento dos algoritmos de ML e DL, é necessário o conhecimento das tecnologias relacionadas. Esta seção descreve as alternativas e diferentes ferramentas, linguagens e *frameworks* que estão disponíveis no mercado para o uso no contexto desse estudo.

2.6.1 Python

Python é uma linguagem de programação moderna e poderosa, e seu uso e distribuição são livres, incluindo a possibilidade de modificações na sua estrutura e código fonte. Uma de suas características é que ela não é fortemente tipada, permitindo o uso de variáveis sem a sua declaração, fazendo com que a determinação do tipo seja realizada implicitamente pelo interpretador. Ela utiliza a indentação como controle de estrutura do código, não sendo necessário o uso de um delimitador ao final do bloco de código.

Python combina o poder de uma linguagem de programação de uso geral com a facilidade de uso de linguagens de domínio específico, como MATLAB e R. Ela tem bibliotecas para o carregamento de dados, visualização, estatística, processamento de linguagem natural, de imagens, dentre outras. Essa grande variedade provém aos cientistas um leque de funcionalidades, sejam gerais ou específicas. Uma das grandes vantagens de *Python* é a possibilidade de interagir diretamente com o código, usando o terminal ou outras ferramentas como o *Jupyter Notebook*. Como uma linguagem de uso geral, ela permite a criação de complexas interfaces de usuário e de *Web Services*, além da integração com outros sistemas (Müller e Guido, 2016).

Apesar da performance das linguagens interpretadas, como *Python*, para tarefas computacionais intensas ela está no mesmo nível de linguagens de programação de baixo nível, com o uso de bibliotecas que implementam rápidas operações vetorizadas em *arrays* multidimensionais, pois essas bibliotecas foram desenvolvidas em linguagens de baixo nível, como Fortran e C (Raschka, 2019).

Python é considerada uma das melhores opções para cálculos matemáticos, pois sua sintaxe é parecida com a forma matemática de expressar as ideias, além de ter uma grande utilização em projetos web. É também uma das linguagens mais utilizadas no meio científico, por ser fácil de usar e aprender, sendo a linguagem de programação mais utilizada em ML e a linguagem padrão para a utilização de diversos frameworks da área, como o *Keras*, o *PyTorch* e outras bibliotecas (Morgan, 2017).

2.6.2 R

R é um sistema computacional com funcionalidades de linguagem de programação, manipulação de gráficos de alto nível e com capacidade de *debugging*. A linguagem de programação em si é um dialeto da linguagem S e é de grande utilização no meio estatístico. A

sintaxe da linguagem R tem similaridades com a da linguagem C, porém a sua semântica vem de linguagens funcionais, como Lisp e APL (R Core Team, 2022). É possível a utilização de expressões e funções como parâmetros de outras funções.

Muitos dizem que R foi especificamente desenhada para estatísticos, principalmente por sua fácil e poderosa capacidade de visualização dos dados. R é relativamente fácil de aprender, especialmente se sua utilização for para a análise de dados (Morgan, 2017).

2.6.3 Keras

Keras é uma API de D*eep Learning* para *Python*, construída sobre o *TensorFlow* (TensorFlow, 2022), que disponibiliza uma forma conveniente de definir e treinar qualquer tipo de modelo de DL. Ela foi inicialmente desenvolvida para pesquisa, com o objetivo de permitir que o DL fosse experimentado de forma rápida. Através do *TensorFlow*, *Keras* pode rodar em diferentes tipos de *hardware*, GPU, TPU ou somente em CPU, e pode facilmente ter sua escalabilidade distribuída para centenas de máquinas.

Keras (Keras, 2022) é conhecida por priorizar a experiência do desenvolvedor. É uma API para seres humanos, não máquinas. Ela segue as boas práticas ao reduzir a carga cognitiva, ao oferecer fluxos de trabalho simples e consistentes. Isso minimiza o número de ações requeridas para executar a maioria das tarefas, providenciando um retorno simples em caso de erro. Por esse motivo *Keras* é fácil de aprender para um principiante, e altamente produtiva para um profissional (Chollet, 2017).

Atualmente existem duas técnicas que todo pesquisador deve estar familiarizado para ter sucesso na aplicação de ML: algoritmos que utilizam *gradient Boosting,* para a resolução de problemas de baixa complexidade, e D*eep Learning* para o restante. Em termos técnicos, isso significa conhecer XGBoost e *Keras*, as duas bibliotecas que dominam as competições de ML no sítio Kaggle (Chollet, 2017).

2.6.4 TensorFlow

TensorFlow (TensorFlow, 2022) é uma complexa biblioteca utilizada para a computação numérica distribuída. Ela possibilita treinar e rodar grandes redes neuronais de forma eficiente distribuindo a computação através de centenas de servidores de múltiplos GPUs, se necessário. Foi construída para ter uma ótima escalabilidade na área de ML, mas pode ser utilizada para qualquer tipo de computação pesada. Foi desenvolvida pela Google e suporta diversos serviços de larga escala da empresa. Se tornou *open source* em 2015, e é atualmente a biblioteca mais popular de DL. Incontáveis projetos utilizam *TensorFlow* para todos os tipos de tarefas de ML, como classificação de imagens, processamento de linguagem natural, sistemas de recomendações, previsões de séries temporais, dentre tantos outros (Geron, 2019).

Felizmente para os usuários de Python, atualmente TensorFlow é a API de ML completa, atraindo muitos praticantes de AA e DL. Além de possuir uma API oficial em C++, diversas novas ferramentas baseadas em TensorFlow foram lançadas, como TensorFlow.js e TensorFlow Lite, que se concentram na execução e implantação de modelos de Aprendizagem Automática em navegadores Web e em dispositivos móveis, além da Internet das Coisas (IoT). As APIs em outras linguagens, como em *Java*, *Haskell*, *Node.js* e *Go*, ainda não são estáveis, mas a comunidade de código aberto e os desenvolvedores do TensorFlow as estão aprimorando constantemente (Raschka, 2019).

O TensorFlow é construído em torno de um gráfico de computação composto por um conjunto de nós. Cada nó representa uma operação que pode ter zero ou mais entradas ou saídas. Um tensor é criado como um identificador simbólico para se referir à entrada e saída dessas operações. Matematicamente, tensores podem ser entendidos como uma generalização de escalares, vetores, matrizes e assim por diante. Mais concretamente, um escalar pode ser definido como um tensor de nível 0 (*rank* 0), um vetor pode ser definido como um tensor de nível 1 (*rank* 1), uma matriz pode ser definida como um tensor de nível 2 (*rank* 2), assim como matrizes empilhadas de terceira dimensão podem ser definidas como tensores de nível 3 (*rank* 3), como é possível ver na Figura 7. Mas é importante notar que no TensorFlow, os valores são armazenados em matrizes NumPy e os tensores fornecem referências a essas matrizes (Raschka, 2019).

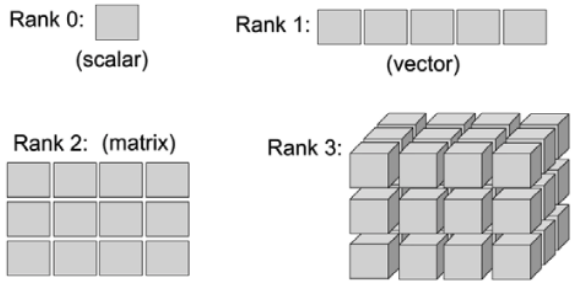

Figura 7 – Representação dos tensores de níveis 0 e 1 na primeira linha, e de níveis 2 e 3 na segunda linha (Raschka, 2019).

Na primeira versão do *TensorFlow*, a computação baseava na construção de gráficos diretos e estáticos para representar o fluxo de dados. Como a computação de gráficos estáticos provou-se um ponto de fragilidade para muitos usuários, na versão 2.0 a biblioteca implementou também a computação dinâmica de gráficos, o que a torna mais flexível, permitindo a construção e treinamento de redes neuronais de forma mais simples (Raschka, 2019).

2.6.5 PyTorch

É uma biblioteca *open source* de ML em *Python* desenvolvida pelo *Facebook* (Facebook, 2022). Um de suas principais características é ser uma plataforma para modelos computacionais gráficos dinâmicos, permitindo que o usuário realize modificações durante o seu *runtime.*

Além de incluir diversas funcionalidades e funções, ela ainda pode ser executada com a ajuda de placas gráficas (GPUs).

PyTorch é um pacote *Python* que provém duas funcionalidades de alto nível: computação por meio de tensores com forte aceleração via GPU e redes neurais profundas construídas em um sistema *autograd* baseado em fita. Atualmente *PyTorch* é utilizado tanto como um substituto para o *NumPy* para usar o poder das GPUs quanto como uma plataforma de pesquisa em DL que provém máxima flexibilidade e velocidade (GitHub, 2022).

Recentemente seu desenvolvimento foi transferido do *Facebook* para a *Linux Foundation* (PyTorch, 2022).

2.6.6 scikit-learn

scikit-learn (scikit-learn, 2022) é um projeto *open source* que contém muitos algoritmos estados da arte de ML, com uma extensa documentação de cada um. Por ser *open source*, pode ser utilizado e distribuído gratuitamente, e seu código pode ser inspecionado por todos. Ele é constantemente aprimorado e possuí uma grande comunidade de usuários. *scikit-learn* é uma ferramenta muito popular e é a biblioteca *Python* mais proeminente na área de ML. É largamente utilizado na indústria e academia, com uma grande quantidade de tutoriais e código de exemplo disponíveis, e funciona muito bem com uma grande gama de ferramentas científicas em *Python* (Müller e Guido, 2016).

2.6.7 Escolha das tecnologias

Para a execução desse estudo, a escolha das tecnologias foi tomada com base na experiência e nos relatos dos autores dos livros utilizados como referência.

Por falta de estudo formal e desconhecimento prático na aplicação das técnicas de ML e DL, ou de conhecimento nas linguagens R e Python, foi escolhida a linguagem *Python* para este trabalho, por ser considerada uma linguagem de programação de fácil aprendizado, além de ser considerada a melhor linguagem para a área de Aprendizagem Automática. Essa escolha vai de encontro à atual tendência de expansão da linguagem no universo de ML (Stack Overflow, 2021).

Dentre os frameworks, a escolha foi pelo conjunto Keras mais TensorFlow, devido a sua reconhecida facilidade de aprendizado para um iniciante, além de ter como uma das suas funcionalidades a possibilidade futura de implementação do estudo realizado como um W*eb Service* e ter a maior comunidade de usuários da área atualmente, o que facilita a resolução de problemas encontrados.

Para aplicação dos algoritmos de ML será utilizado o *scikit-learn*, escolha unânime de todos os autores pesquisados.

2.7 Sumário

Neste capítulo foi descrito o estado da arte dos campos envolvidos nesta dissertação: acidentes de trânsito em rodovias federais brasileiras, Aprendizagem Automática e aplicação de AA em dados de AT. Foram relatados os estudos brasileiros mais recentes sobre AT em rodovias, foram descritos os conceitos relacionados à ML e DL, assim como todos os algoritmos e técnicas que foram utilizadas neste trabalho e relatados os estudos internacionais sobre a aplicação de AA na identificação de locais de AT. Foram descritas as tecnologias e *frameworks* disponíveis, bem como as que foram escolhidas para esse trabalho, devidamente justificadas.

3 Análise de Valor

A geração de valor é objetivo de qualquer empresa. Ao agregar valor aos seus produtos, ela satisfaz as necessidades de seus clientes e recebe em retorno a recompensa desejada. Umas das formas de agregar valor é por meio da inovação de produtos existentes ou pela criação de novos. Mas o processo de inovação, que é a introdução de uma ideia nova, algo novo ou uma nova forma de fazer algo já existente, é complexo e arriscado. Muitas vezes seu resultado não atende à uma necessidade do mercado, trazendo prejuízos indesejados.

Para minimizar isso, uma forma de medir o possível sucesso de uma nova ideia é fazer uma análise de valor. Essa análise permite descobrir como agregar valor a um produto, novo ou já existente, ao menor custo e com a melhor qualidade possíveis. E isso deve realizado de uma forma sistemática, organizada, continua e focada no valor a ser agregado.

Neste capítulo é realizada a análise de valor da solução proposta, a fim de avaliar o trabalho a ser desenvolvido para a solução do problema. Vários conceitos são descritos e aplicados aos diversos aspetos dessa dissertação.

3.1 New Concept Development

Uma ideia ou um conceito se torna uma inovação quando atende a necessidade ou visão de um consumidor, que visualiza um valor agregado que compensa o seu custo de aquisição. Uma das formas de se desenvolver e analisar uma ideia ou um conceito é por meio do modelo *New Concept Development* (NCD).

De uma forma genérica, o processo de inovação pode ser desenvolvido em um modelo composto de três etapas: o *Fuzzy Front End* (FFE), o *New Product Development* (NPD) e o processo de comercialização.

O FFE é a etapa onde o processo de criação de uma nova ideia ou conceito inicia. É caracterizado por experimental, criativo, não estruturado e muitas vezes caótico. Ao contrário

disso, o NPD é um o processo mais disciplinado, organizado e com objetivos claros. Devido a sua natureza caótica, o FFE não pode ter uma abordagem sequencial como o NPD.

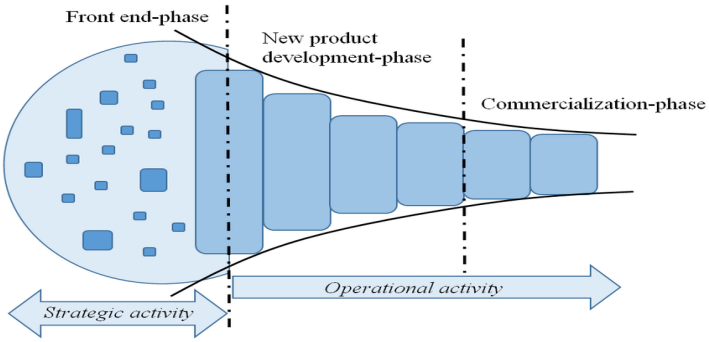

Figura 8 – Processo de Inovação (Martikainen, 2017)

Como uma forma de definir métodos, ferramentas, técnicas e uma estrutura para melhorar o gerenciamento do FFE, foi criado o *New Concept Development* (NCD), um modelo relacional composto por três partes. A primeira parte é o motor, que representa a cultura, liderança e estratégia da organização. Esse motor controla os cinco elementos da área interna do NCD. A segunda parte é a área interna, que contém os cinco elementos principais do modelo: identificação da oportunidade, análise da oportunidade, geração e enriquecimento da ideia, seleção da ideia e definição do conceito. A terceira parte do NCD é composta por fatores de influência, que são formados pela capacidade da organização, pelo mundo exterior (consumidores, competidores, leis, …) e pelo nível tecnológico existente, tanto interno quanto externo. Todos esses elementos podem afetar o processo de inovação. O modelo NCD é representado pela **Erro! Fonte de referência não encontrada.**.

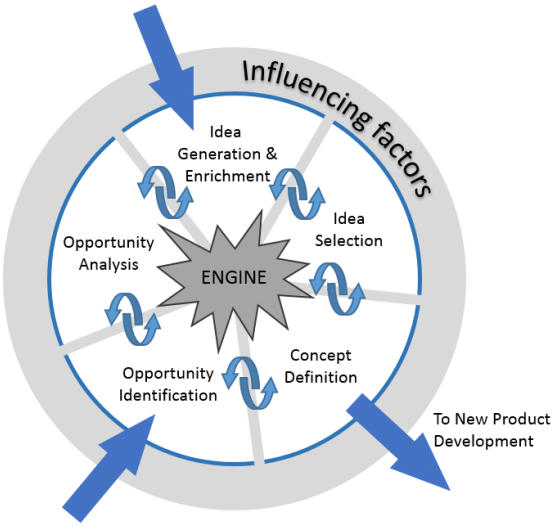

Figura 9 – O NCD é um modelo relacional, não linear (Martikainen, 2017)

Objetivando amadurecer a análise do problema objeto deste estudo, os cinco elementos do NCD foram aplicados no contexto desse trabalho.

3.1.1 Identificação da Oportunidade

Esse elemento define os métodos para identificar as oportunidades do que se busca alcançar, por meio da análise de tendências tecnológicas, análise de tendência do cliente, análise de inteligência competitiva, pesquisa de mercado ou planeamento de cenários, dentre outros.

A malha viária brasileira é muito grande, e com isso os possíveis locais de ocorrências de AT são muitos. A identificação aproximada do local de um novo acidente é primordial, seja para a implementação de políticas públicas, seja para a execução de fiscalização policial ostensiva direta.

Mesmo que a eficiência do modelo de identificação não possa ser diretamente medida, pois a medida do alcance da correta predição de um local de acidente é a fiscalização daquele local e a não ocorrência de um novo AT, a constante busca na melhoria do modelo compensam os esforços envolvidos. Esse ciclo de busca pela proteção de vidas leva à muitas oportunidades, e uma delas é a necessidade constante da melhoria dos modelos de identificação de locais de AT.

Além dessa oportunidade, para a PRF há a necessidade da criação de um sistema que faça isso de forma automatizada. Isso levaria a uma melhor gestão por parte dos tomadores de decisão, sendo ainda possível uma integração com outros sistemas e serviços, servindo-os com as informações de que necessitam. Um exemplo de sistema que poderia consumir essas informações seria um que faria a criação automática do Cartão de Rondas[9] ostensivas. Esse documento indica os locais onde as viaturas policias devem estar presentes e em quais horários, sendo o maior motivador da necessidade de melhoria do modelo de identificação de locais de AT da instituição.

3.1.2 Análise da Oportunidade

Uma vez identificada a oportunidade, é necessária a avaliação das vantagens e desvantagens do que se pode alcançar quando se persegue essa oportunidade. A comparação desses aspetos é necessária para a avaliação da demanda de recursos e trabalho necessários à execução do estudo proposto, permitindo assim a mensuração do sacrifício necessário para o aproveitamento da oportunidade.

Uma das metodologias mais utilizadas na avaliação de uma oportunidade é a matriz de análise SWOT (*Strengths, Weaknesses, Opportunities and Threats*). Devido a sua simplicidade e eficiência, que coloca lado a lado as forças, fraquezas, oportunidade e ameaças do objeto em análise, ela permite uma avaliação mais fácil e visual. Aplicando a SWOT ao contexto desse trabalho, se identifica os seguintes elementos (GÜREL, 2017):

[9] Ronda é o ato de percorrer a rodovia realizando o patrulhamento e a fiscalização de trânsito

- Forças (*Strengths*): conhecimento do autor sobre o domínio relacionado ao objeto em estudo, acidentes de trânsito, e sobre o modelo atual de identificação de locais de AT da instituição.
- Fraquezas (*Weaknesses*): falta de conhecimento prático da aplicação de técnicas de Aprendizagem Automática por parte do autor, assim como a falta de estudo formal da área de ML.
- Oportunidades (*Opportunities*): utilização de técnicas estado da arte do campo de Aprendizagem Automática no tratamento de dados históricos de AT, que já se mostram promissoras em outros campos da ciência. Possibilidade futura do desenvolvimento de um sistema automático de identificação de locais de AT baseado no resultado desse trabalho, mais eficiente, fácil de ser utilizado e que consuma menos tempo dos tomadores de decisão. Além disso, o novo sistema poderia ser integrado a outros serviços, automatizando ainda mais processos.
- Ameaças (*Threats*): conclusão de que um novo modelo de identificação de locais de AT não acrescenta nada para além dos modelos já utilizados, somente novos custos. Ou que, mesmo que o estudo seja satisfatório, a sua utilização ou o desenvolvimento de um novo sistema não compensam os custos envolvidos.

Tabela 1 – Análise SWOT (GÜREL, 2017).

Forças	**Fraquezas**
• Conhecimento do autor sobre o domínio relacionado à acidentes de trânsito em rodovias brasileiras. • Conhecimento do modelo atual de identificação de locais de AT da instituição.	• Falta de conhecimento prático da aplicação de técnicas de Aprendizagem Automática por parte do autor • Falta de estudo formal da área de ML.
Oportunidades	**Ameaças**
• Uso de tecnologias “estado da arte”, com maior potencial futuro. • Possibilidade futura de desenvolvimento de um sistema automático de identificação de locais de AT, mais eficiente e fácil de ser utilizado. • Além disso, o novo sistema poderia ser integrado a outros serviços, automatizando ainda mais processos.	• Conclusão de que um novo modelo de identificação de locais de AT não acrescenta nada para além dos modelos já utilizados. • Conclusão de que, mesmo com um estudo satisfatório, a sua utilização ou o desenvolvimento de um novo sistema não compensam os custos envolvidos.

Comparando as vantagens e desvantagens evidenciadas na **Erro! Fonte de referência não encontrada.**, fica claro que os benefícios da implementação de um novo modelo de identificação de locais de AT baseado em uma metodologia de ML compensam os custos envolvidos, tornado a oportunidade apresentada válida, principalmente se isso aumentar as chances de salvar vidas humanas.

3.1.3 Geração e Enriquecimento de Ideias

Esse elemento do modelo NCD se relaciona com a proposta, desenvolvimento e amadurecimento da ideia concreta. É natural que a ideia passe por diversas mudanças ao ser examinada e desenvolvida junto aos outros elementos do modelo. Essas diversas iterações de análises do modelo fazem com que a ideia se torne mais robusta, madura. Essa etapa pode ser um processo formal, incluindo sessões de brainstorming e bancos de ideias, de modo a incentivar a organização a gerar ideias novas ou modificadas para a oportunidade identificada.

No caso concreto deste trabalho, após a análise da literatura associada ao problema e dos objetivos propostos, algumas ideias surgiram para contribuir com o estudo proposto:

- Início do estudo com análises estatísticas convencionais.
- Teste de pelo menos três algoritmos de ML, justificando a escolha do mais adequado.
- Teste de redes neuronais, com o objetivo de superar os algoritmos de ML.

3.2 Proposta de Valor

O termo valor pode ser utilizado de diversas formas e, dependendo do contexto, pode ter diferentes significados. Mas de uma forma genérica, valor pode ser entendido como o quanto determinada coisa vale, ou o quanto uma pessoa está disposta a pagar por ela. Pode ainda ser definido pela quantidade monetária necessária para se obter um produto ou serviço, ou a medida da importância financeira de determinado bem (Merriam-Webster, 2022)

Proposta de valor é uma ferramenta organizacional usada para convencer um cliente a comprar o seu produto ou serviço, por meio da demonstração dos benefícios propostos e da ajuda oferecida aos seus problemas. Ou ainda, o que o cliente pode esperar como solução para as suas necessidades ao consumir aquele produto. É uma forma de demonstrar ao cliente o ganho que ele terá com aquele produto e o valor que isso está agregando as suas necessidades.

Uma das formas de se criar uma proposta de valor a ser apresentada ao cliente é por meio do preenchimento da ferramenta *Value Proposition Canvas,* um quadro onde a empresa procura fazer uma adequação daquilo que é oferecido por ela, que fica à esquerda, junto às necessidades do cliente, que ficam à direita. A aplicação do *Value Proposition Canvas* à essa dissertação gerou o quadro abaixo.

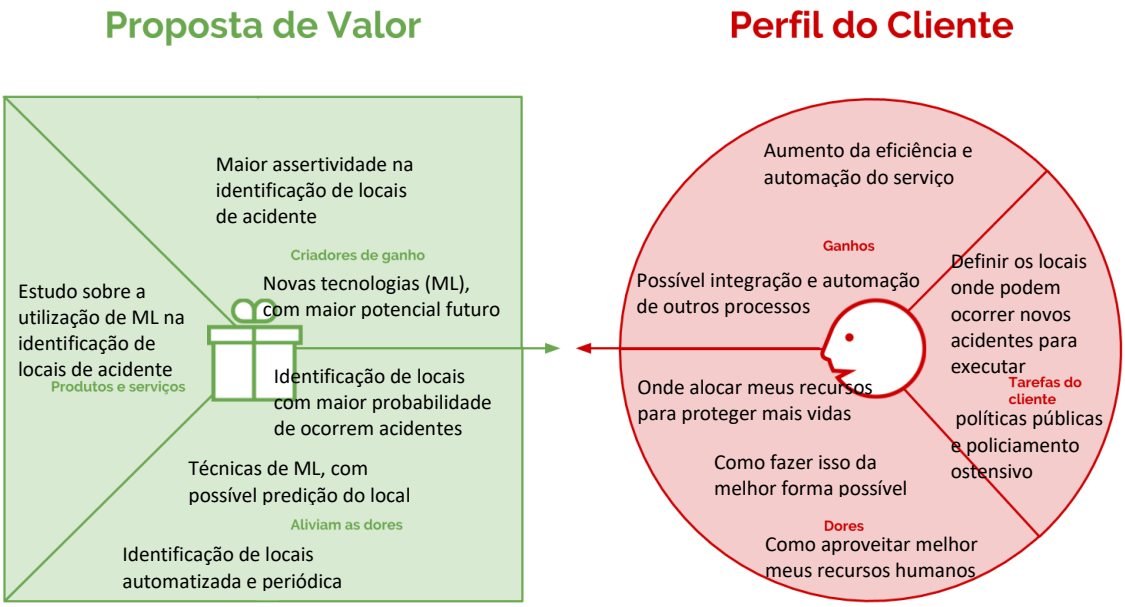

Figura 10 – *Value Proposition Canvas*. Adaptado de (Daniel Pereira, 2019).

A partir do *Canvas* acima, podemos compor a proposta de valor desse trabalho como um estudo da aplicação de técnicas de *Machine Learning* (ML) para a identificação de locais com maior probabilidade de ocorrerem acidentes em rodovias federais, objetivando melhorar o atual modelo de identificação do cliente. Ela atende as suas necessidades ao entregar uma solução ao seu problema, que é realizar a identificação dos locais com ferramentas convencionais e que requerem manipulação manual dos operadores.

Além disso, o estudo pode contribuir para a decisão sobre o desenvolvimento futuro de um novo sistema de identificação de locais de AT, possibilitando a integração a outros serviços do cliente, automatizando outros processos, como o de criação do cartão de rondas ostensivas nas rodovias.

3.3 Sumário

Neste capítulo foi realizada a análise de valor da solução proposta. Vários conceitos relacionados à área foram descritos. Estes conceitos foram aplicados ao contexto deste estudo, identificando o valor da solução proposta para resolução do problema relatado.

4 Metodologia

Aplicar com sucesso as técnicas de Aprendizagem Automática requer mais do que conhecer quais algoritmos existem e como eles funcionam. Um bom profissional de ML deve saber como escolher um determinado algoritmo para uma aplicação particular, como monitorar e como responder aos resultados obtidos a partir dos experimentos, para poder melhorar o desempenho do sistema. Determinar o objetivo e as métricas de erro que serão utilizadas é o primeiro passo de um projeto, porque a métrica irá guiar todas as ações futuras. É necessário ter uma ideia de qual nível de performance se quer alcançar (Goodfellow, Bengio e Courville, 2016). Para avaliar o alcance dos objetivos propostos por essa dissertação, algumas métricas devem ser utilizadas para mensurar a aplicação dos modelos de AA no estudo dos dados históricos de AT.

Este capítulo descreve a hipótese a ser confirmada com o alcance dos objetivos propostos, as métricas de performance para os algoritmos empregados e a metodologia de avaliação que foi utilizada para nesse estudo.

4.1 Hipótese e questões relevantes

Segundo Barros (2008), hipótese é uma resposta possível formulada diante de um problema identificado, que se destina a ser posteriormente verificada. É uma asserção provisória que pode ou não ser verdadeira, e que deve ser submetida a cuidadosos procedimentos de verificação e demonstração.

Como o objetivo desse trabalho é analisar se os algoritmos de Aprendizagem Automática empregados sobre os dados históricos de acidentes da PRF têm boas capacidades preditivas, mostrando outra perspetiva para além do método utilizado pela instituição, a hipótese de investigação é que as técnicas de ML e DL superem consideravelmente as técnicas de estatística convencionais, como a regressão linear, possibilitando predições confiáveis, dentro das métricas estabelecidas. Se uma técnica tiver uma boa predição, ela poderia ser utilizada como o novo método de identificação de locais de AT da PRF.

Além disso, espera-se que algumas questões possam ser respondidas:

- Os dados de geolocalização são relevantes e colaboram com as técnicas utilizadas no estudo?
- Informações como a condição meteorológica ou a ocorrência do acidente em um dia de feriado são relevantes para as predições?
- Os dados utilizados são suficientes para uma aplicação satisfatória dos algoritmos?

4.2 Métricas de Avaliação

Uma consideração importante em um projeto de ML é qual métrica de performance utilizar. Diversas métricas de performance diferentes podem ser utilizadas para medir a eficácia de uma aplicação de ML. Essas métricas geralmente são diferentes da função de custo utilizada para treinar um modelo. Elas são comumente utilizadas para medir a acurácia ou, equivalentemente, a taxa de erro de um sistema.

Mas algumas aplicações requerem uma métrica mais avançada. Muitas vezes em um projeto é mais custoso cometer um tipo de erro que cometer outros. Por exemplo, um sistema de deteção de *spam* em *emails* pode cometer dois tipos de erros: incorretamente classificar uma mensagem legítima como *spam* ou incorretamente classificar um *spam* como um *email* legítimo, permitindo que ele apareça na caixa de entrada. É muito pior bloquear uma mensagem legítima, que vai ser movida para outra caixa que não a caixa de entrada, do que permitir que uma mensagem questionável passe pelo filtro. Nesses casos, ao invés de medir a taxa de erro do classificador de *emails*, pode ser mais interessante medir algum tipo de custo total, onde o custo de bloquear uma mensagem legítima é maior que o custo de permitir mensagens que são *spams* (Goodfellow, Bengio e Courville, 2016).

Nesta seção, são descritas as métricas mais utilizadas em ML, tanto em tarefas de regressão quanto em tarefas de classificação.

4.2.1 Mean Absolute Error

O *Mean Absolute Error* (MAE) é uma métrica que mede a magnitude média dos erros em um conjunto de predições, sem considerar sua direção. É a média, dentro de um conjunto de medições, da diferença absoluta entre o valor da predição e o valor real observado, onde cada diferença individual tem um peso igual. No cálculo do MAE, o algoritmo compara os valores das predições do modelo com os valores corretos do conjunto de dados, medindo o acerto. Um modelo por ser considerado assertivo quando a taxa de erro entre os dados de treino e os dados de teste é baixa. Isso significa que o modelo aprendeu os padrões e tendências subjacentes do conjunto de dados (Theobald, 2017). Sua fórmula é:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|x_i - y_i| \quad (1)$$

, onde y é valor da predição, x é o valor da medição real e n é o número de predições.

4.2.2 Mean Squared Error

O *Mean Squared Error* (MSE) é a média do quadrado da distância entre o valor da predição e o valor real observado. Como ela utiliza o quadrado da unidade do dado observado, ao invés da sua unidade natural, sua interpretação é menos intuitiva. Elevar a distância do valor da predição menos o real aumenta a impacto dos erros muito grandes, penalizando mais os erros grande menos os erros pequenos. Essa propriedade é essencial quando se deseja que um modelo tenha erros menores (Frost, 2022b). Sua fórmula é:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \quad (2)$$

, onde y é valor da predição, x é o valor da medição real e n é o número de predições.

4.2.3 Root Mean Squared Error

O *Root Mean Squared Error* (RMSE) é calculado computando a raiz quadrado do MSE. É uma forma de retornar a unidade do erro MSE para a unidade natural do dado observado, tornando sua interpretação mais intuitiva. Sua fórmula é:

$$RMSE = \sqrt{MSE} \quad (3)$$

, onde y é valor da predição, x é o valor da medição real e n é o número de predições.

4.2.4 R² score

O *R² score* (coeficiente R², também conhecido como Coeficiente de Determinação) é uma forma de medir a predição de um modelo de regressão linear, e seu valor varia entre 0 e 1. O valor 1 corresponde a uma predição perfeita, e um valor 0 corresponde a um modelo que constantemente apenas prediz como resultado a média dos dados do conjunto de treino (Müller e Guido, 2016).

Ela indica a percentagem da variância na variável dependente que as variáveis independentes explicam coletivamente. O coeficiente R² mede a força da relação entre o modelo e a variável dependente, representando o quanto o modelo se ajusta ao conjunto de dados (Frost, 2022a).

4.2.5 *Accuracy*

Accuracy é definida como a proporção de classificações corretas de um modelo. É o número de classificações corretas, tanto as predições que foram corretamente classificadas como positivas quanto as que foram corretamente classificadas como negativas, dividido pelo número total de predições. A *accuracy* é uma métrica útil quando os erros de predições de todas as classes são igualmente importantes (Burkov, 2019). Sua fórmula é:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (4)$$

, onde TP são os positivos, TN são as negativas, FP são os falsos positivos e FN são os falsos negativos.

4.2.6 *Confusion Matrix*

A *Confusion Matrix* é uma tabela que resumi o quanto um modelo de classificação foi bem-sucedido em predizer cada amostra como pertencente as diversas classes. Um eixo da *confusion matrix* é o eixo que contém as predições do modelo, o outro eixo contém as classes corretas. Em uma classificação binária, a tabela teria duas linhas e duas colunas, onde as linhas correspondem às classes verdadeiras e as colunas correspondem às predições realizadas. Em uma classificação multiclasse, ela teria o número de linhas e colunas igual ao número de classes diferentes. Uma *confusion Matrix* ajuda a encontrar erros nos padrões do modelo (Burkov, 2019).

As entradas na diagonal principal da matriz correspondem as classificações corretas, enquanto as outras entradas revelam quantas amostras de uma classe foram incorretamente classificadas como de outra classe. A *confusion matrix* de um modelo com uma classificação perfeita não teria falsos positivos e negativos, e todas as entradas da matriz, exceto a diagonal principal, seriam 0.

No entanto, inspecionar uma *confusion matrix* completa é um pouco complicado e, embora seja possível obter muitos *insights* ao se analisar todos os seus aspetos, esse processo é muito manual e qualitativo. Existem outras métricas que podem resumir as informações da *confusion matrix* (Müller e Guido, 2016).

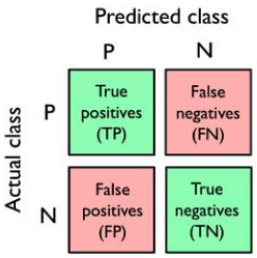

Figura 11 – *Confusion Matrix* de classificação binária (Raschka, 2019).

4.2.7 Precision, Recall e F1-score

Existem algumas aplicações de ML em que se deseja treinar um classificador binário com a intenção de detetar um evento raro. Por exemplo, um modelo médico que detete uma doença rara. Supondo que apenar uma pessoa em um milhão tenha a doença, é fácil chegar a uma *accuracy* de 99,9999%, bastando apenas criar o modelo de uma forma em que ele sempre reporte a ausência da doença. Nessas situações percebe-se que a métrica de *accuracy* pode não ser a melhor forma de medir a performance de um modelo (Goodfellow, Bengio e Courville, 2016). Nesses casos existem outras métricas que podem melhor avalizar essa performance.

Precision mede quantas predições classificadas como positivas são realmente positivas:

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

, onde TP são os positivos e FP são os falsos positivos.

Precision é usada como métrica quando o objetivo é limitar o número de falsos positivos. Como exemplo, imagine um modelo para predizer quando uma nova droga vai ser efetiva no tratamento de uma doença em testes clínicos. Testes clínicos são muito caros, então a companhia farmacêutica somente vai executar os testes se tiver certeza que a droga irá realmente funcionar. Para isso é importante que o modelo não produza muitos falsos positivos – em outras palavras, que ele tenha uma alta precisão (Müller e Guido, 2016).

Recall mede o quanto das amostras positivas são capturadas pelas predições positivas:

$$Precision = \frac{TP}{TP + FN} \tag{6}$$

, onde TP são os positivos e FN são os falsos negativos.

Recall é usada como métrica quando é necessário identificar todas as amostras positivas, isto é, quando é importante evitar falsos negativos. O diagnostico de câncer é um bom exemplo: é importante encontrar todas as pessoas que estão doentes, mesmo que isso signifique que algumas pessoas saudáveis possam ser classificas como doentes.

Existe um balanço entre otimizar *recall* e otimizar *precision*. Pode-se obter um *recall* perfeito ao se classificar todas as amostras da predição como a classe positiva, assim não haveria nenhuma classe negativa nem falso negativa. Entretanto, predizer todas as classes como positivas resultaria em muitos falsos positivos, e então a precisão (*precision*) seria muito baixa. Por outro lado, se houver um modelo faça a predição de apenas uma única amostra dos dados positivo e o restante como negativo, então a precisão será perfeita (assumindo que essa amostra seja de fato positiva), mas o *recall* será ruim (Müller e Guido, 2016).

4.2.8 Curva ROC

A curva ROC (*receiver operating characteristic*) é uma métrica comum utilizada para medir a performance de modelos de classificação. Ela utiliza a combinação da taxa de positivos corretos com a taxa de falsos positivos para construir uma imagem do resumo da performance da classificação (Burkov, 2019).

A curva ROC plota no gráfico a taxa de positivos corretos (*true positive rate* – TPR, ou *recall*) contra a taxa de falsos positivos (*false positive rate* – FPR). A FPR é a taxas de instâncias negativas que foram incorretamente classificadas como positivas. A FPR é igual a 1 menos a taxa de negativos corretos (*true negative rate* – TNR), que é a taxa de instâncias negativas classificadas como negativas. A TNR também é chamada de *specificity* (Geron, 2019).

Na análise da curva ROC existe um balanço: quanto maior for o *recall* (TPR), mais o classificador produz falsos negativos (FPR). A linha pontilhada da Figura 12 representa a curva ROC de um classificador puramente randómico; um bom classificador terá sua linha o mais distante possível da linha pontilhada (em direção ao topo esquerdo).

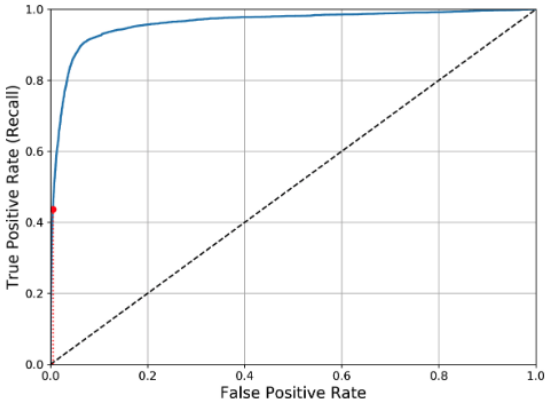

Figura 12 – Curva ROC (Geron, 2019).

Uma forma de comparar classificadores é medir a área abaixo da curva (*area under the curve* – AUC). Um classificador perfeito terá a ROC AUC igual a 1, onde um classificador puramente randómico terá a ROC AUC igual a 0,5 (Geron, 2019).

4.3 Métodos de Otimização

4.3.1 Método Holdout

O método *Holdout* (reservar) consiste em dividir o conjunto inicial de dados em um conjunto de dados de treino, que será utilizado para o treinamento do algoritmo, e um conjunto de dados de teste, que será utilizado para estimar a performance da generalização do modelo. O objetivo dessa divisão é permitir que seja possível ajustar e comparar as diferentes configurações dos parâmetros do algoritmo, a fim de melhorar a performance da predição do

modelo em dados desconhecidos. Esse processo é conhecido como *model selection.* Entretanto, se o mesmo conjunto de dados de teste for utilizado muitas vezes nesse processo, ele se tornará uma parte dos dados de treino, e o modelo terá mais tendência a cometer *overfitting*. Apesar desse erro, muitos ainda utilizam os dados de teste para o *model selection*, o que não é uma boa prática de ML (Raschka, 2019).

Uma melhor forma de realizar o método *Holdout* consiste em separar o conjunto inicial de dados em três partes: uma para treino, uma para a validação do modelo e uma para o seu teste. Os dados de treino são utilizados para ajustar o modelo e os dados de validação são utilizados para melhorar sua performance no *model selection*. Uma das vantagens de se ter um conjunto de treino que não é conhecido pelo modelo quando do processo de treinamento e *model selection* é que é possível se obter uma estimação da habilidade de generalizar novos dados menos enviesada. A Figura 13 ilustra o conceito de *Holdout*, onde os dados de validação são utilizados para avaliar a performance do modelo, após o treinamento dos dados de treino com diferentes híper parâmetros. Só após o fim desse processo de treinamento a performance do modelo é estimada com os dados de teste (Raschka, 2019).

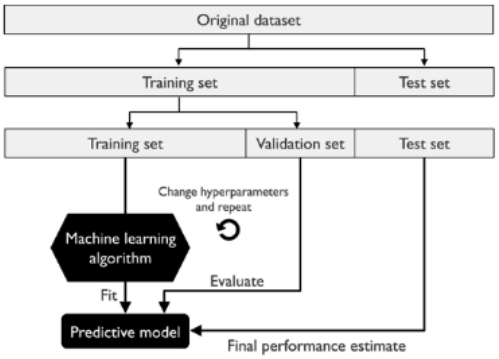

Figura 13 – Conceito de *Holdout* (Raschka, 2019).

4.3.2 Cross-Validation

No processo de *Cross-Validation* (CV) os dados que serão utilizados no processo de *model selection* são divididos em *k* partes sem repetições, onde *k-1* partes serão utilizadas para o treinamento do modelo e a parte restante será utilizada para a validação da performance. Esse processo é repetido *k* vezes até se obter *k* modelos e *k* estimativas de performance (Figura 14). Então a performance do modelo é calculada pela média da performance das diferentes partes de teste, para se obter uma estimativa de performance que é menos sensível à subdivisão dos dados de treinamento em comparação com o método *Holdout* (Raschka, 2019).

Após a identificação dos hiper parâmetros mais satisfatórios, o modelo é treinado no conjunto completo de dados de treinamento para se obter uma estimativa de desempenho final usando os conjuntos independentes de dados de teste. A lógica por trás do ajuste do modelo

em todo o conjunto de dados de treinamento é fornecer mais exemplos de treinamento para o algoritmo, resultando em um modelo mais preciso e robusto.

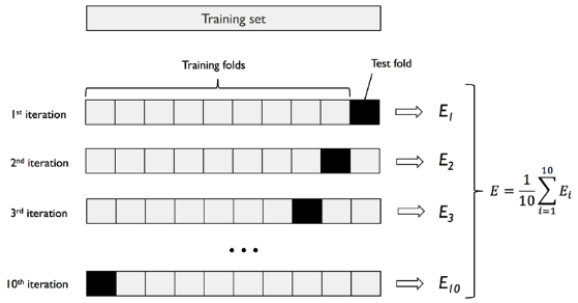

Figura 14 – Representação da *Cross-Validation* (Raschka, 2019).

4.4 Metodologia de Avaliação

Como o objetivo do trabalho é superar os métodos estatísticos convencionais, a avalização será com base nas métricas de erro e coeficientes estabelecidos. A cada algoritmo aplicado, as métricas dos resultados obtidos serão comparadas com as métricas anteriores estabelecidas, revelando se o algoritmo foi mais eficiente ou não.

Para a avaliação dos algoritmos de regressão será utilizada o *Root Mean Square Error* (RMSE) como função de custo e o coeficiente de determinação (R^2 score) como métrica de avaliação.

O RMSE foi escolhido em detrimento da métrica *Mean Absolute Error* (MAE) porque penaliza mais as predições mais distantes do valor real. A raiz quadrada é aplicada para retornar o valor do erro para um número mais fácil de visualizar por humanos, além de retornar à unidade do erro ao seu natural. Assim, um erro RMSE de valor 2 km significa que a predição foi até 2 quilómetros para mais ou para menos do local real do acidente. O coeficiente de determinação, R^2 score, foi escolhido por representar o quanto o modelo treinado explica cada predição realizada.

Para a avaliação dos algoritmos do tipo classificação será utilizado como função de custo o erro MSE, para um comparativo com os algoritmos de regressão. Como na avaliação do modelo não há a necessidade de se evitar classificações falsas positivas, nem há problemas de haver classificações como falso negativas, já que a predição de um acidente não necessariamente significa que ele vai ocorrer, será utilizada a métrica *accuracy* para medir o nível de acerto do modelo.

4.5 Sumário

Neste capítulo foram descritas as métricas de performance dos algoritmos de AA, os métodos de otimização e a metodologia de avaliação. As métricas utilizadas neste estudo serão o erro RMSE, o R^2 *score* e a *accuracy.*

5 Estudo Computacional

Este capítulo descreve as etapas da aplicação das técnicas de AA na predição de locais de AT. São descritos os procedimentos de coleta, tratamento e engenharia dos dados utilizados além do estudo computacional da aplicação dos algoritmos de ML e DL nos dados históricos de AT.

5.1 Tratamento e engenharia de dados

A qualidade dos dados e a quantidade de informações úteis que eles contêm são fatores determinantes para o quanto um algoritmo de ML pode aprender. Para isso, é absolutamente necessário assegurar que os dados sejam examinados e pré processados antes de serem inseridos em um algoritmo de aprendizado (Raschka, 2019).

O valor de um conjunto de dados está intimamente ligado a possibilidade de extrair deles conhecimento de alto nível, que sirvam de apoio à tomada de decisão ou possam ser utilizados para exploração e melhor compreensão do fenómeno que os gerou (Goldschmidt, Passos e Bezerra, 2015).

5.1.1 Coleta e descrição dos dados

Os dados utilizados neste estudo são os dados históricos de ocorrências de acidentes de trânsito (AT) ocorridos em rodovias federais brasileiras. Eles forma compilados pela Polícia Rodoviária Federal (PRF) e estão disponíveis no sítio (Polícia Rodoviária Federal, 2022a). No estudo são utilizados os dados da seção “Agrupados por ocorrência”, que estão em uma formatação mais adequada. O período utilizado será de 2017 a junho de 2022. O ano de 2017 foi escolhido porque a partir dele os boletins de acidentes da PRF passaram a conter os dados de geolocalização (latitude e longitude) do local do acidente.

A descrição dos dados pode ser encontrada no Dicionário de Dados, no sítio (Polícia Rodoviária Federal, 2022b), transcrito na Tabela 2.

Tabela 2 – Dicionário de Dados

NOME DA VARIÁVEL	DESCRIÇÃO
Id	Variável com valores numéricos, representando o identificador do acidente.
data_inversa	Data da ocorrência no formato dd/mm/aaaa.
dia_semana	Dia da semana da ocorrência. Ex.: Segunda, Terça, etc.
Horário	Horário da ocorrência no formato hh:mm:ss.
uf	Unidade da Federação. Ex.: MG, PE, DF, etc.
br	Variável com valores numéricos, representando o identificador da BR do acidente.
km	Identificação do quilômetro onde ocorreu o acidente, com valor mínimo de 0,1 km e com a casa decimal separada por ponto.
municipio	Nome do município de ocorrência do acidente
causa_acidente	Identificação da causa principal do acidente. Neste conjunto de dados são excluídos os acidentes com a variável causa principal igual a "Não"
tipo_acidente	Identificação do tipo de acidente. Ex.: Colisão frontal, Saída de pista, etc. Neste conjunto de dados são excluídos os tipos de acidentes com ordem maior ou igual a dois. A ordem do acidente demonstra a sequência cronológica dos tipos presentes na mesma ocorrência.
classificação_acidente	Classificação quanto à gravidade do acidente: Sem Vítimas, Com Vítimas Feridas, Com Vítimas Fatais e Ignorado.
fase_dia	Fase do dia no momento do acidente. Ex. Amanhecer, pleno dia, etc.
sentido_via	Sentido da via considerando o ponto de colisão: Crescente e decrescente.
condição_meteorologica	Condição meteorológica no momento do acidente: Céu claro, chuva, vento, etc.
tipo_pista	Tipo da pista considerando a quantidade de faixas: Dupla, simples ou múltipla.
tracado_via	Descrição do traçado da via.
uso_solo	Descrição sobre as características do local do acidente: Urbano: Sim; Rural: Não.
latitude	Latitude do local do acidente em formato geodésico decimal.
longitude	Longitude do local do acidente em formato geodésico decimal.
pessoas	Total de pessoas envolvidas na ocorrência.
mortos	Total de pessoas mortas envolvidas na ocorrência.
feridos_leves	Total de pessoas com ferimentos leves envolvidas na ocorrência.
feridos_graves	Total de pessoas com ferimentos graves envolvidas na ocorrência.
feridos	Total de pessoas feridas envolvidas na ocorrência (é a soma dos feridos leves com os graves).
ilesos	Total de pessoas ilesas envolvidas na ocorrência.
ignorados	Total de pessoas envolvidas na ocorrência e que não se soube o estado físico.
veiculos	Total de veículos envolvidos na ocorrência.

Uma amostra dos dados originais sem tratamento pode ser visualizada na Figura 15.

id	data_inversa	dia_semana	horario	uf	br	km	municipio	causa_acidente	tipo_acidente	classificacao_acidente	fase_dia	sentido_via	condicao_metereologica
8	01/01/2017	domingo	00:00:00	PR	376	112	PARANAVAI	Fenômenos da Natureza	Queda de ocupante d	Com Vítimas Feridas	Plena Noite	Crescente	Chuva
9	01/01/2017	domingo	00:01:00	SC	101	234	PALHOCA	Falta de Atenção à Condi	Colisão com objeto e	Sem Vítimas	Plena Noite	Crescente	Chuva
11	01/01/2017	domingo	00:00:00	PR	153	56,9	SANTO ANTONIO D	Animais na Pista	Capotamento	Com Vítimas Feridas	Plena Noite	Decrescente	Garoa/Chuvisco
12	01/01/2017	domingo	00:00:00	GO	153	435	ANAPOLIS	Avarias e/ou desgaste e	Tombamento	Com Vítimas Feridas	Plena Noite	Decrescente	Céu Claro
13	01/01/2017	domingo	00:00:00	SC	280	77,3	CORUPA	Ingestão de Álcool	Saída de leito carroçá	Com Vítimas Feridas	Plena Noite	Decrescente	Chuva
14	01/01/2017	domingo	00:40:00	GO	60	188	GUAPO	Falta de Atenção à Condi	Colisão traseira	Com Vítimas Feridas	Plena Noite	Decrescente	Nublado
15	01/01/2017	domingo	00:01:00	PB	104	3,4	NOVA FLORESTA	Ingestão de Álcool	Tombamento	Com Vítimas Fatais	Plena Noite	Decrescente	Céu Claro
16	01/01/2017	domingo	00:30:00	TO	153	141,7	ARAGUAINA	Ingestão de Álcool	Colisão traseira	Com Vítimas Feridas	Plena Noite	Decrescente	Ignorado
17	01/01/2017	domingo	01:45:00	RS	116	34,9	VACARIA	Defeito Mecânico no Vei	Colisão traseira	Com Vítimas Feridas	Plena Noite	Decrescente	Céu Claro

tipo_pista	tracado_via	uso_solo	pessoas	mortos	feridos_leves	feridos_graves	ilesos	ignorados	feridos	veiculos	latitude	longitude	regional	delegacia	uop
Simples	Reta	Não	1	0	0	1	0	0	1	1	-23,09880731	-52,38789369	SR-PR	DEL7/7	UOP05/PR
Dupla	Curva	Não	1	0	0	0	1	0	0	1	-27,8101	-48,6357	SR-SC	DEL8/1	UOP02/SC
Simples	Reta	Não	2	0	2	0	0	0	2	1	-23,36951985	309,9351311	SR-PR	DEL7/7	UOP07/PR
Dupla	Reta	Sim	1	0	0	1	0	0	1	1	-16,27473677	-48,96908998	SR-GO	DEL1/2	UOP01/GO
Simples	Não Informado	Não	1	0	1	0	0	0	1	1	-26,44675249	-49,20166969	SR-SC	DEL8/6	UOP03/SC
Dupla	Reta	Sim	2	0	1	0	1	0	1	2	-16,82489647	-49,53520775	SR-GO	DEL1/1	UOP02/GO
Simples	Curva	Sim	1	1	0	0	0	0	0	1	-6,4622	-36,1899	SR-PB	DEL14/2	UOP04/PB
Múltipla	Reta	Sim	4	0	2	0	1	1	2	2	-7,18668515	-48,22970957	SR-TO	UOP04/TO	
Simples	Reta	Não	6	0	4	0	2	0	4	2	-28,5071196	-50,941176	SR-RS	DEL9/5	UOP03/RS

Figura 15 – Amostra dos dados originais sem tratamento

5.1.2 Limpeza e reorganização dos dados

A maioria dos algoritmos de *machine learning* não funcionam com *features* com valores vazios. Além disso é necessário examinar os dados para confirmar que eles têm a mesma estrutura em todas as características. Para isso, os dados foram primeiramente tratados com o programa MS Excel para a resolução de problemas de fácil identificação.

Em uma primeira análise, foi identificado que nos ficheiros originais .csv retirados do sítio da PRF, nos anos de 2021 e 2022, o registo numérico da casa decimal foi modificado em relação aos registado nos anos anteriores. Nos ficheiros dos anos anteriores, o registo da casa decimal foi feito com o uso de vírgula, conforme padrão numérico brasileiro/português. Mas nos anos de 2021 e 2022 esse registo passou a ser feito com o uso de ponto. Para manter um padrão, o registo numérico das casas decimais dos anos de 2021 e 2022 foram modificados para o uso de vírgula como separador decimal.

Também foram encontrados diversos registos em que os campos *br* e *km* continham o valor *NA*. Como essas informações são determinantes para o estudo, esses registos foram completamente apagados, pois não é possível utilizar nenhuma técnica de tratamento de dados para sua recuperação ou estimação. Os registos excluídos foram movidos para um ficheiro chamado *registos excluidos*, que está no repositório git dessa dissertação[10]. Foram excluídas 146 entradas do ano de 2017, 114 de 2018, 95 de 2019, 158 de 2020, 166 de 2021 e 106 de 2022.

No campo *condicao_metereologica*, os registos que continham o valor *ignorado* foram modificados para *Céu Claro*, pois esse é o valor mais próximo de um valor padrão e representa a condição com menos influência para a ocorrência de um acidente. Foram modificados 1502 registos no ano de 2017, 1095 de 2018, 992 de 2019, 776 de 2020, 779 de 2021 e 397 de 2022.

[10] O sítio do repositório dessa dissertação é https://bitbucket.org/1190129/dissertacao/src/master .

Ainda no campo *condicao_metereologica*, os registos que continham o valor *Neve* foram modificados para *Chuva*. Como as chances de nevar no Brasil são muito baixas, mesmo que esse valor represente um registo correto da condição no momento do acidente, as suas ocorrências – 1 em 2017, 1 em 2019, 1 em 2020 e 3 em 2021 – representam um *outlier* grande e desnecessário ao estudo.

No campo *tracado_via* os registos que continham o valor *Não Informado* foram modificados para *Reta*, pois é o mais próximo de um valor padrão e representa o tipo de traçado predominante na malha viária. Foram modificados 8108 registos no ano de 2017, 6771 de 2018, 7172 de 2019, 7613 de 2020, 8245 de 2021 e 4251 de 2022.

Nos dados dos anos de 2017, 2018, 2019 e 2020, no campo *uop*, os valores que estavam vazios foram preenchidos com os valores do campo *delegacia* do mesmo registo. Uma Delegacia, nesse contexto, é um conjunto de uma ou mais unidades operacionais (UOP). Nesses anos, alguns estados não possuíam nenhuma delegacia definida na estrutura organizacional da PRF, somente UOPs. Isso foi modificado a partir de 2021.

Nos dados do ano de 2021, o título das colunas *uop* e *delegacia* estavam invertidos. Eles foram corrigidos.

Nos dados do ano de 2020, 56041 registos estavam com os valores dos campos *delegacia* e *uop* invertidos. Eles foram corrigidos.

Nos dados dos anos de 2017, 2018, 2019 e 2020, a designação /*estado* dos valores do campo *uop* foram retirados, para atender a nova nomenclatura vigente do ano de 2021 em diante. O estado da federação pode ser identificado no campo *regional*. O mesmo foi feito no campo *delegacia* dos mesmos anos.

Nos dados dos anos de 2021 e 2022, no campo *delegacia* os valores *DEL02PE* foram modificados para *DEL02*. Nos dados dos anos de 2017, 2018, 2019 e 2020, no mesmo campo, os valores *DEL 10/10* foram modificados para *DEL10/10*.

Todos os valores dos campos *delegacia* e *uop* dos anos de 2017, 2018, 2019 e 2020 foram modificados para atender os novos padrões identificados nos anos de 2021 e 2022.

Como sugestão à PRF na melhoria da coleta de informações de AT, no campo *condicao_metereologica* a possibilidade do valor *Sol* deveria ser retirada, pois o valor *Céu Claro* representa uma condição meteorológica mais adequada e abrangente, tanto para o dia como para noite.

No campo *tracado_via* não deveria ser possível registar o valor *Não Informado*, pois essa imprecisão dificulta uma análise estatística dos dados.

5.1.3 Engenharia de dados

Feature engineering é o processo de utilizar o conhecimento sobre os dados e o seu domínio para fazer com que o algoritmo possa trabalhar melhor. Isso é realizado modificando os dados no que for necessário, antes que eles sejam inseridos no modelo. Além da aplicação do algoritmo, o processo de engenharia de dados é a etapa mais importante dentro do fluxo de trabalho de um projeto de ML, porque muitos algoritmos não são ricos o suficiente para aprender quais são as *features* mais relevantes sozinho. A forma de apresentar os dados ao algoritmo é crucial ao sucesso do modelo (Chollet, 2017).

O objetivo da análise de dados é construir características informativas: aquelas que permitirão ao algoritmo construir um modelo que realize boas predições nos dados de treino. Uma característica altamente informativa é uma característica com um alto poder de predição (Burkov, 2019).

Um modelo só é capaz de aprender se os dados de treinamento contêm *features* relevantes e em quantidades suficientes, e poucas irrelevantes. Uma parte crucial de um projeto de AA é desenvolver um bom conjunto de dados que possam ser utilizados no seu treinamento (Geron, 2019). Esse processo de desenvolvimento é composto por:

- *Feature selection*: seleção das características mais úteis para o treinamento.
- *Feature extraction*: combinação de características para a criação de uma mais útil
- Criação de novas características a partir de novos dados.

5.1.3.1 Feature Selection

Para o estudo proposto, foi realizada uma análise do domínio dos dados de AT disponíveis. Identificou-se que muitas das *features* presentes nos dados originais não existiam até o momento do sinistro, que são geradas após a ocorrência em si. São informações que caracterizam o sinistro ocorrido, mas que só passam a existir como consequência da ocorrência de um AT. Muitas delas não caracterizam as causas do sinistro, mas são consequência direta dele.

Isso é relevante porque se essas características só passam a existir após a ocorrência de um AT, e isso significa que elas desconhecidas no momento da solicitação de uma predição ao modelo. O problema é que não é possível treinar um algoritmo com determinadas *features* e depois solicitar uma predição sem informar um valor para cada *feature* presente no treinamento. O algoritmo simplesmente não funciona. Isso significa que boa parte das *features* presentes nos dados desse estudo não podem ser utilizadas na aplicação dos algoritmos de ML e DL.

Além disso, muitas dessas características não têm ligação com as causas do AT. Um exemplo é o campo “tipo_acidente”. O tipo de um acidente é uma consequência direta do acidente. É uma informação que só existe após o acidente e não tendo influência nenhuma nas circunstâncias que levaram a sua ocorrência. O que não é o caso do campo “km”, que tem relação direta com a ocorrência de um AT.

Como o objetivo do estudo é encontrar a localização do próximo AT, é importante que os dados analisados contenham somente *features* que seja relevante para determinação desse local. Nesse contexto, as *features* que são consequência do AT serão retirados do estudo. Da mesma forma, as características que podem ser inferidas a partir de outras ou as características que não são relevantes para a identificação do local de um novo AT serão retirados do estudo, para diminuir o número de *features* sem poder de predição.

Os campos *classificacao_acidente*, *pessoas*, *mortos*, *feridos_leves*, *feridos_graves*, *ilesos*, *ignorados* e *feridos* foram retirados pois são consequências do AT, não sendo dados relevantes para a identificação de um novo acidente. E por mais que eles pudessem ter correlação com o local da ocorrência de um AT, eles não podem ser utilizados para o treinamento de um algoritmo porque não são conhecidos no momento de realizar uma predição.

O campo *veiculos* representa o número de veículos envolvidos no acidente, e apesar de ser útil para algumas análises estatísticas, não é relevante para a identificação do local de um novo sinistro. Sendo assim, foi retirado. O campo *id* foi retirado pois é um campo utilizado para base de dados da PRF.

O campo *municipio* foi retirado pois não há a necessidade de um filtro nessa granularidade. Para esse estudo, a característica *regional* exprime melhor uma subdivisão do território brasileiro. As regionais são subdivisões organizacionais da PRF. Cada estado brasileiro possui sua respetiva regional. Mas uma regional não compreende necessariamente o mesmo espaço territorial do estado. Muitas regionais são responsáveis por trechos de rodovias que se estendem territorialmente para dentro de outros estados. Um exemplo é a regional do Distrito Federal (DF), que tem sob sua circunscrição rodovias que adentram os estados do Goiás, Minas Gerais e até uma pequena parte da Bahia.

Das 29 características originais dos dados históricos de AT, sem contar o *km* que é o alvo do estudo, após a retirada das que não são relevantes para a identificação do alvo, restaram 11 *features*: *data_inversa*, *dia_semana*, *horario*, *uf*, *br*, *condicao_metereologica*, *latitude*, *longitude*, *regional*, *delegacia* e *uop*.

5.1.3.2 Feature extraction

Para ajudar na análise dos algoritmos, ainda no programa MS Excel, foi adicionado o campo *data_hora*, que é a concatenação dos campos *data_invertida* e *horario*. O objetivo é ter uma *feature* do tipo *datetime*.

A partir do conhecimento do domínio relativo aos AT, partindo do pressuposto que os dias que são feriados podem ser relevantes na predição de um AT, foi criado o campo *feriado*, informando se no dia do acidente era feriado nacional ou não. A lista dos feriados foi extraída das portarias do governo brasileiro, e os seus sítios estão presentes no Anexo 1.

Foi criado o campo *chuva*, que simplifica as informações do campo *condicao_metereologica* de forma mais relevante para a análise dos algoritmos. Os registos *Chuva*, *Granizo* e *Garoa*/*Chuvisco* serão considerados como *chuva*, informando que no momento do acidente

choveu e que a rodovia estava molhada, informação que pode ser relevante na ocorrência de um sinistro. O campo *condicao_metereologica* foi retirado.

5.1.4 Análise Exploratória dos dados

Foi realizada uma pequena análise exploratória dos dados para a identificação das correlações existentes entre as *features* dos dados. A análise completa pode ser visualizada no ficheiro *1_Explora_dados.ipynb*.

Na Figura 16, verifica-se a distribuição de AT entre os estados brasileiros. Notoriamente, estados com mais malha viária tem uma maior quantidade de sinistros, como Minas Gerais (MG) e Santa Catariana (SC).

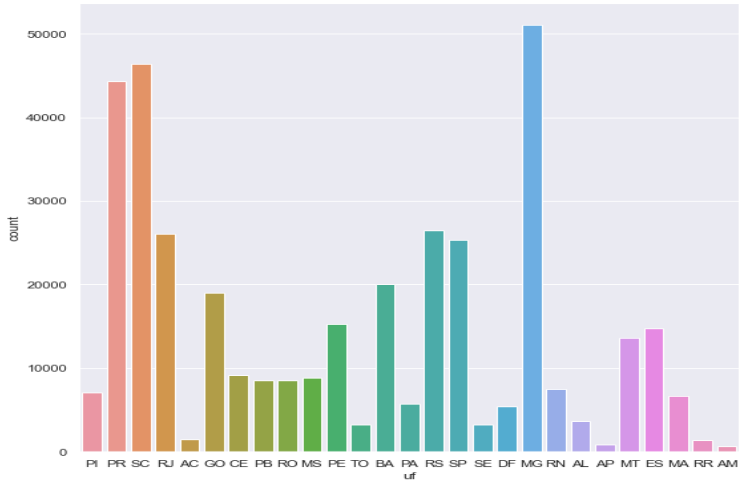

Figura 16 – Distribuição dos acidentes entre os estados.

As principais causas de AT podem ser sintetizadas na Figura 17. Como alguns estudos evidenciados anteriormente comprovam, as principais causas estão ligadas a comportamentos humanos.

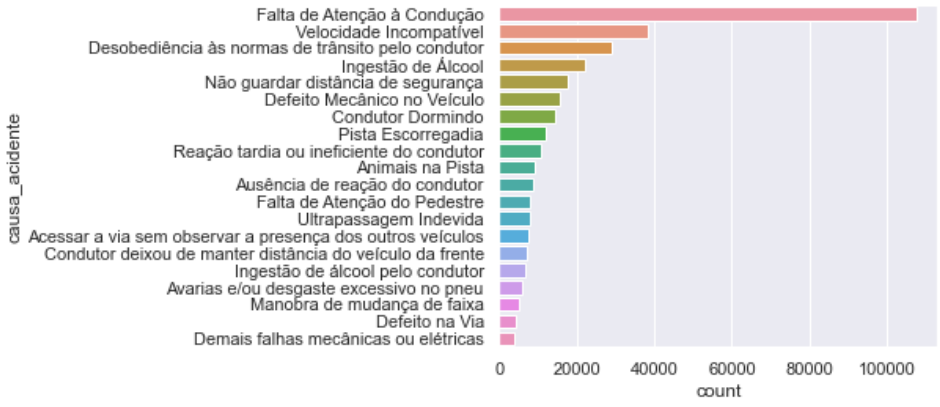

Figura 17 – Principais causas de acidentes.

Na distribuição dos AT entre os dias da semana, Figura 18, verifica-se um aumento nas ocorrências na sexta-feira e final de semana. Além disso pode-se identificar um comportamento cíclico nas ocorrências durante os dias da semana.

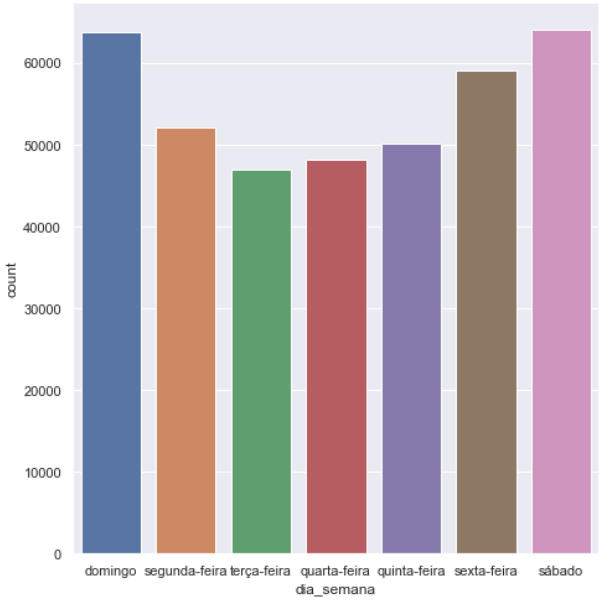

Figura 18 – Distribuição de acidentes entre os dias da semana.

Verifica-se também uma concentração de AT nos horários de pico de veículos, as 7:00 e 18:00 horas. No caso da distribuição dos sinistros durante as horas do dia, também é possível verificar um comportamento cíclico, com momentos de alto e baixo volume (Figura 19).

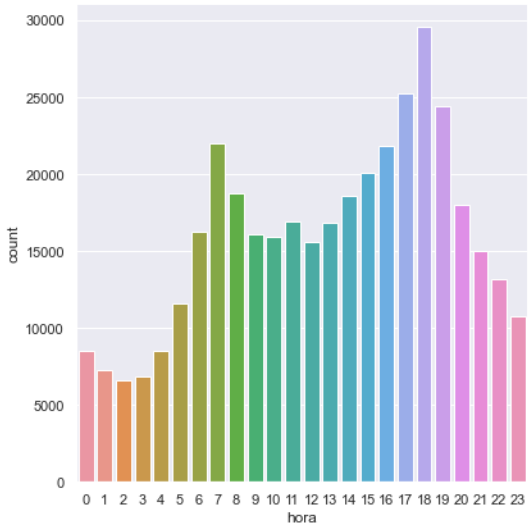

Figura 19 – Distribuição de acidentes entre as horas do dia.

Procurando identificar corelações entre as *features,* foi aplicada a função `corr()`, que computa o *standard correlation coefficien*t (também chamado de *Pearson's r*) entre os pares de características. O coeficiente varia de -1 a 1. Quanto mais próximo de 1, maior é a

correlação positiva. Quanto mais próximo de -1, maior é a correlação negativa. Coeficientes próximos de zero significam que não há correlação linear (Geron, 2019).

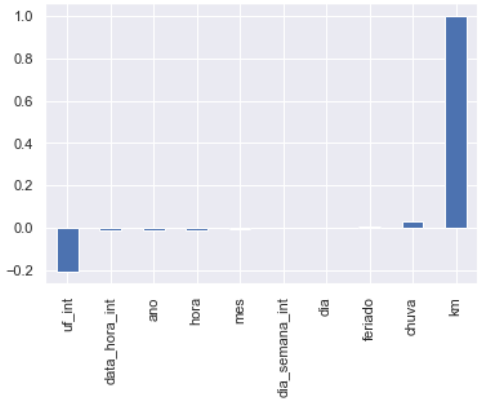

Figura 20 – Correlação entre as *features* e o alvo (km).

Avaliando a Figura 20, percebe-se a fraca correlação entre as *features* disponíveis para o treinamento dos algoritmos e o alvo pretendido, o *km*. A característica que tem mais correlação, ainda que baixa, com a determinação do quilómetro de um AT é a unidade da federação (estado). Essa falta de correlação pode ser um fator preocupante na aplicação dos modelos de ML e DL deste estudo.

5.1.5 One-Hot-Encoding

Poucos algoritmos aceitam dados de entrada contendo *features* que sejam do tipo categórica. Uma forma de representar variáveis categóricas é utilizando a técnica *One-Hot-Encoding* (OHE). A ideia é criar uma característica binária para cada categoria da variável: é atribuído o valor 1 quando a designação da característica for a mesma da categoria da variável; se a designação da característica for diferente da categoria, é atribuído o valor 0 (Geron, 2019).

No contexto deste estudo, essa técnica será aplicada às *features uf* e *dia_semana*, que no conjunto original dos dados são representadas por variáveis do tipo *string*.

Essa técnica é utilizada para evitar um erro comum quando se trabalha com variáveis categóricas. Não se deve atribuir um valor qualquer para cada categoria em uma variável desse tipo somente para possibilitar o treinamento do algoritmo. Atribuir valores as categorias, como AC = 1, AL = 2, AM = 3, aumenta a dimensionalidade do algoritmo, pois ele vai identificar que há uma ordem entre os valores nessa categoria e que essa ordem específica é importante para a tomada de decisão. Se a ordem de uma característica não é importante, atribuir valores numéricos ordenados vai confundir o algoritmo, que vai tentar identificar uma regularidade onde não há, o que pode levá-lo ao *overfitting* (Burkov, 2019).

5.1.6 Lidando com variáveis cíclicas

A Figura 20 pode ser simplificada ao retirar as *features uf_int* e *km*, para melhorar a visualização:

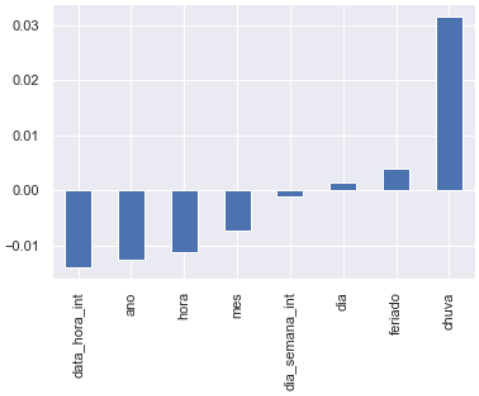

Figura 21 – Correlação entre as *features* e o alvo (km), mas sem a barra do "km".

Pode-se visualizar que as correlações não mudaram, apenas as barras ficaram maiores. Mas desta figura podem ser feitas algumas considerações.

A característica *data_hora_int* é a data e hora da ocorrência de um AT de forma arredondada, sem os minutos, transformada em número inteiro. Observa-se que ela tem menos correlação que a característica *chuva*. A característica *data_hora_int* foi criada para ser possível o treinamento dos algoritmos, pois seria impossível a utilização da característica original *data_hora*, que é do tipo categórica – o tipo *datetime64[ns]* é identificado como uma *string* pela maioria dos algoritmos. Se ela não fosse transformada em inteiro, seriam milhares de categorias, praticamente inviabilizando o treinamento.

Uma outra forma da *feature data_hora* ser utilizada no estudo seria realizando sua decomposição nas *features ano*, *mes*, *dia* e *hora*, como foi feito para a construção da Figura 21.

Mas a *feature hora*, que é a hora do dia que ocorreu o sinistro, tem uma característica interessante, como pode-se observar na Figura 19: ela é cíclica. Assim como as *features dia_semana*, *mes* e *dia*, a *feature hora* poderia ser representada de outra forma no conjunto de dados.

Uma forma de lidar com números inteiros que representam uma característica cíclica seria representá-los como as coordenadas *x* e *y* de um círculo. Isso significa mapear a variável cíclica em um círculo, para que o valor mais baixo da variável apareça após o seu valor mais alto. Para isso, basta calcular os componentes *x* e *y* do valor da variável usando as funções trigonométricas *seno* e *cosseno*. Assim, ao invés da *feature hora* se estender de 0 a 23, haverá duas novas *features*, *hora_sin* e *hora_cos* que vão de 0 a 1, e as duas combinadas representam a característica cíclica desejada. Desta forma, 23:59 será representada próximo de 00:00, como deveria ser (Kaleko, 2017).

5.2 Geolocalização

Antes de iniciar a o estudo computacional, será verificada a qualidade das informações dos campos relacionados à geolocalização, *latitude* e *longitude*.

Desde o ano de 2017, a PRF incorporou no seu boletim de acidente os campos de latitude e longitude, visando acrescentar mais qualidade as informações dos AT e com isso possibilitar análises estatísticas de maior qualidade.

Para facilitar, foi criada uma função que plota os pontos de geolocalização em um mapa para uma análise visual. Mas já primeira execução da função um erro foi retornado. Havia valores de longitude que não estavam entre os valores permitidos do padrão geodésico decimal, entre -360 e 720. Da mesma forma ocorreu com a latitude, que deveria estar entre -90 e 90.

Excluindo da função os valores fora dos intervalos permitidos do padrão geodésico decimal, foi gerado o seguinte mapa da Figura 22.

Figura 22 – Plotagem dos pontos de latitude e longitude dos dados nacionais

Como se observa, muitos registos estão com seus pontos de geolocalização fora do território brasileiro.

Isso também acontece quando geramos um mapa de um estado qualquer, neste exemplo, do Distrito Federal (Figura 23):

Figura 23 - Plotagem dos pontos de latitude e longitude do Distrito Federal

Mesmo excluindo o ponto que está sobre o Oceano Pacífico, o mapa continua com pontos que não correspondem ao espaço territorial das rodovias da regional do DF. Muitos deles estão desconexos das rodovias (Figura 24):

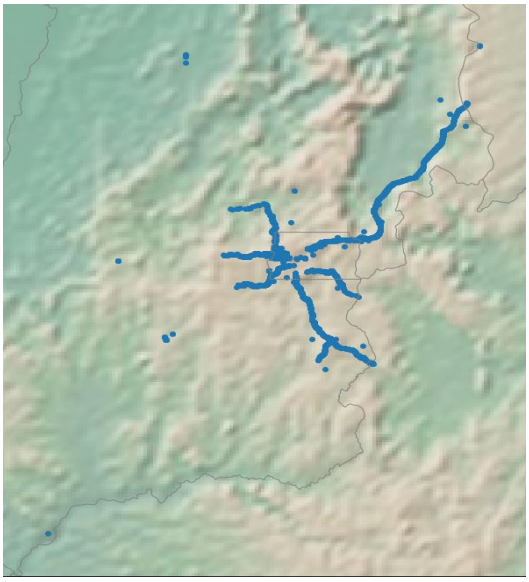

Figura 24 - Plotagem dos pontos de latitude e longitude do Distrito Federal sem o ponto sobre o Pacífico.

Como percebe-se, muitos registos estão com suas informações de latitude e longitude não correspondentes à localização da ocorrência do AT. E isso gera uma certa desconfiança em relação à qualidade dos dados de geolocalização.

Mas o que é relevante sobre o uso dos dados de geolocalização na previsão de novos AT por meio da utilização de técnicas de ML é que não é possível realizar o treinamento de um algoritmo incluindo uma *feature*, nesse caso a *latitude* e *longitude*, e depois tentar realizar uma predição sem a utilização dessa *feature*. Como não é possível saber a geolocalização de um acidente antes da sua ocorrência, a solicitação de predição sem informar a latitude e longitude em um modelo que foi treinado com esses campos não seria possível, simplesmente o algoritmo retornaria um erro.

Por esse motivo os dados de geolocalização não serão utilizados nesta dissertação. Outras técnicas de identificação de locais de ocorrência de AT, como as técnicas de mapa de calor por exemplo, poderiam beneficiar do acesso às informações de latitude e longitude em suas análises.

5.3 Aplicação dos Algoritmos

O objetivo do treinamento de um modelo é predizer o local da possível ocorrência de um acidente quando informados os dados de entrada necessários. Para a PRF, o objetivo dessa identificação é responder a seguinte situação real: em qual local da BR X_1, pertencente à UOP

X_2, situada na regional X_3, deve ser posicionada uma viatura operacional a fim de evitar a ocorrência de um acidente no dia X_4, às X_5 horas, com as seguintes condições: previsão climática X_6, etc.? Responder essa pergunta seria a principal funcionalidade de um sistema que poderia ser desenvolvido pela PRF.

Um principiante na área de ML pode ter a impressão de que o ingrediente mais importante para ser um especialista em Aprendizagem Automática é conhecer uma ampla variedade de algoritmos e ser bom em matemática. Na prática, geralmente pode-se fazer muito melhor com a aplicação correta de um algoritmo genérico do que com a aplicação descuidada de um algoritmo obscuro. A aplicação correta de um algoritmo depende do domínio de uma metodologia bastante simples, que pode ser resumida no seguinte processo (Goodfellow, Bengio e Courville, 2016):

- Determine as metas — defina qual métrica de erro utilizar e o seu valor alvo. A definição da métrica de erro deve ser orientada pelo problema ao qual o aplicativo pretende resolver.
- Estabeleça um fluxo de trabalho de ponta a ponta o mais rápido possível, incluindo a estimativa das métricas de desempenho apropriadas.
- Instrumente bem o sistema para determinar gargalos no desempenho. Diagnosticar quais componentes estão tendo um desempenho pior do que o esperado e se é devido a *overfitting*, *underfitting*, defeito nos dados ou no software.
- Faça alterações incrementais repetidamente, como coletar novos dados, ajustar híper parâmetros ou alterar algoritmos, com base em descobertas específicas da instrumentação.

A abordagem utilizada para a aplicação dos algoritmos nesse trabalho foi: primeiro a definição de um *baseline* para métricas selecionadas a partir da utilização da técnica de regressão linear sobre os conjuntos de dados. Dentre os conjuntos de dados, um será o conjunto de todos os dados disponíveis, identificados como dados nacionais, e os outros serão conjuntos compostos por de dados de três regionais escolhidas por representarem os grupos da distribuição do número de ocorrências de sinistros entre os estados do país, conforme a Figura 16: a regional do DF representando as regionais com poucos acidentes no nível nacional, SP representando as regionais com uma distribuição central de ocorrências e SC representando as regionais com mais acidentes no país.

A subdivisão por regionais foi escolhida porque, para a PRF, elas representam melhor os estados do país, conforme descrito na seção 5.1.3.1. Cada regional está subdividida em delegacias e cada uma delas é composta por unidades operacionais (UOP). Como a responsabilidade de definir onde estacionar a viatura é do chefe da UOP, se houver a necessidade de realizar filtros subdividindo ainda mais os dados, os filtros seguirão essa grandeza: *regional -> delegacia -> uop -> br*.

Após a definição da *baseline* inicial, a próxima etapa será utilizar técnicas de ML de regressão para a superação a meta anteriormente, isso tanto nos dados nacionais quando nos dados das

regionais escolhidas. Em seguida, serão utilizadas técnicas de ML de classificação, na tentativa de superação do *baseline* anterior. Por fim, serão utilizadas técnicas de ANN nos dados utilizados anteriormente, sempre objetivando superar o *baseline* anterior.

Após a aplicação dos algoritmos, será realizada uma avaliação dos resultados obtidos, com uma análise crítica da utilização das técnicas empregadas.

5.4 Divisão dos dados

Seguindo as boas práticas de aplicação de técnicas de ML e DL, é utilizada a técnica de *Holdout* na separação dos dados. Na análise da regressão linear, regressão em ML e classificação em ML, os dados são divididos em dois grupos: um grupo de dados para o treinamento do algoritmo (denominado de TRAIN_RAW), composto pelos dados dos anos de 2017, 2018, 2019, 2020 e 2021, e um grupo de teste de predição (denominado de TEST_RAW), composto pelos dados do primeiro semestre de 2022. Nessa divisão, os dados de treino correspondem à 92,0% dos dados totais, e os dados de teste à 8,0%.

No caso da aplicação das técnicas de ANN e do algoritmo *Gradient Boosting Regressor* de ML, os dados são divididos em três grupos: um grupo de dados de treino (TRAIN_RAW), composto pelos dados dos anos de 2017, 2018, 2019 e 2020, um grupo de dados de validação (EVAL_RAW), composto pelos dados do ano de 2021 e um grupo de teste de predição (TEST_RAW), composto pelos dados do primeiro semestre de 2022. Nessa divisão, os dados de treino correspondem à 75,3% dos dados totais, os de validação à 16,7% e os dados de teste à 8,0%. A escolha da divisão em três grupo nas técnicas de DL – treino, avaliação e teste – é para evitar que ocorram vazamentos de informações dos dados de testes para os dados de treinamento.

5.4.1 Definição dos valores das métricas

Para desenvolver um bom modelo de ML, deve-se preparar os dados para que possam ser processados pelo algoritmo, selecionar um protocolo de avaliação, definir uma *baseline* simples a ser superada, treinar um primeiro modelo que tenha poder de generalização e que possa realizar *overift* e, em seguida, ajustar o modelo até obter-se o melhor desempenho de generalização possível (Chollet, 2017).

Um modelo é considerado satisfatório quando as medidas dos seus erros alcançam os valores alvos definidos no início do projeto. Mas como determinar um nível razoável de desempenho alvo das métricas de avaliação? Normalmente, no ambiente acadêmico, tem-se alguma estimativa da taxa de erro que é atingível com base em resultados de *benchmark* publicados anteriormente. No contexto do mundo real, deve se ter uma ideia da taxa de erro necessária para que um aplicativo seja seguro, econômico ou atraente para os consumidores. Depois de determinar a taxa de erro desejada, as decisões do projeto devem ser guiadas por atingir essa taxa de erro (Goodfellow, Bengio e Courville, 2016).

Analisando o domínio do objeto desse estudo, não seria difícil definir o valor alvo das métricas escolhidas. A distância ideal máxima que permitiria a um condutor em uma rodovia de alta velocidade enxergar um objeto estático em seu caminho é de 0,5 quilómetros. Essa é a distância máxima em que um condutor visualizaria uma viatura da PRF no acostamento, com tempo hábil para aumentar sua atenção e diminuir sua velocidade sem causar nenhuma mudança no fluxo normal dos veículos.

No Brasil, as rodovias têm placas com marcos de medida de distância de 1 em 1 quilómetro. No boletim de acidente de trânsito da PRF, porém, no campo de registo do local da ocorrência é permitido colocar valores que vão até a casa dos metros, por exemplo *km* 11,506. Mas nesse número, somente o valor do quilómetro é realmente conhecido pelo policial, o restante é uma estimativa com base no seu conhecimento prático. Para o registo do local, o policial considera a placa do marco quilométrico como início e estiva o restante do valor. Isso faz com que o último algarismo significativo real da *feature km* seja o algarismo da unidade, ou seja, o quilómetro. Desta forma, não é possível definir o valor do erro RMSE em 0,5 *km*, já que o registo dos valores da *feature km* nos dados disponíveis para esse estudo não apresentavam essa precisão real.

Então, definiu-se o valor de um quilómetro como sendo o valor alvo ideal do erro RMSE. Esse valor pode ser relativizado, e dentro do domínio do problema, poderia ser aceito um valor máximo de erro de até cinco quilómetros. Cinco quilómetros é o tamanho máximo aceitável de um trecho de rodovia onde uma viatura policial faria uma ronda ostensiva objetivando chamar a atenção de todos os condutores de forma efetiva.

Em relação às métricas R^2 *score*, no caso de tarefas de regressão, e *accuracy*, no caso de classificação, definiu-se que o valor alvo ideal é acima de 0,7, sendo o valor 0,5 o mínimo aceitável, visto a grande aleatoriedade presente nos dados.

5.4.2 Normalização

Seria muito problemático inserir em um modelo de ML valores que têm um intervalo numérico muito grande. O modelo até poderia se adaptar automaticamente a esses dados tão heterogéneos, mas certamente isso tornaria o aprendizado mais difícil. A melhor prática para lidar com esse tipo de dados é a normalização: de cada *feature* dos dados de entrada é subtraída a média dos seus valores e seguida pela divisão pelo seu desvio padrão, para que a *feature* fique centrada em torno do ponto 0 e tenha um desvio padrão unitário. É importante notar que os valores usados na normalização dos dados de teste devem ter sido computados usando os dados de treino. Nunca se deve utilizar valores calculados no conjunto de dados de teste, nem para uma coisa simples como a normalização (Chollet, 2017). Nesse trabalho todos os dados foram normalizados utilizando a função `StandardScaler()`, que foi ajustada somente no conjunto de dados de treino.

5.5 Ambiente computacional

Os estudos foram realizados em um computador com processador AMD 5950X, com 32 Gb de memória RAM, placa de vídeo nVidia GTX 3060 e SSD Samsung 980 Pro. As versões das ferramentas utilizados foram: python 3.10.6, pandas 1.4.4, numpy 1.21.0, matplotlib 3.5.3, openpyx 3.0.10, basemap 1.3.4, seaborn 0.11.2, sklearn 1.1.2, tensorflow 2.9.1, keras 2.9.0, xgboost 1.6.2.

5.6 Regressão Linear

O objetivo da utilização da técnica de Regressão Linear é prever o local de ocorrência de um AT, ou seja, prever a *feature km.* A previsão do conjunto de dados de teste servirá para a definição de um *baseline* inicial, que será a meta a ser superada pelas outras técnicas. Os dados utilizados foram os dados completos, denominados dados nacionais, e os dados filtrados por *regional*, representando os estados brasileiros definidos anteriormente.

Todas as análises de regressão foram avaliadas por meio da métrica *root mean squared error* (RMSE), que é a raiz do erro quadrático médio. O erro quadrático médio foi escolhido porque penaliza as predições que estão muito distantes do valor real. Quanto mais distante do valor real o a predição está, mais ela é penalizada, devido a sua potência quadrada. E a raiz quadrada será utilizada para trazer o erro para próximo de um número mais interpretável por humanos. Por exemplo, uma predição cujo erro RMSE é de 25,5 significa que o ponto real está em média a 25,5 quilómetros, para mais ou para menos, do ponto predito. Todas as predições estão na unidade *km*.

Toda a análise pode ser visualizada no ficheiro *3_Regressao_linear.ipynb*, no repositório da dissertação. Os resultados encontram-se na Tabela 3:

Tabela 3 – Resultados da regressão linear

Erro: RMSE (*km*)	**Regressão Linear**	**R^2 Score**
Nacional	225,38	0,00
SR-DF	40,21	-4,41
SR-DF BR 020	61,41	8,90
SR-DF BR 040	28,25	-0,02
SR-SP	135,75	0,00
SR-SP BR 101	25,48	-0,02
SR-SP BR 116	128,50	0,00
SR-SC	135,10	-1,85
SR-SC BR 101	101,60	1,53
SR-SC BR 116	92,31	0,00

Observa-se que os valores dos erros encontrados foram altos, longe dos valores aceitáveis. E a causa mais provável é a significativa diferença entre os tamanhos das diversas rodovias do país. São mais de 75.000 quilómetros de rodovias distribuídos entre os 27 estados. Estados territorialmente grandes tem grandes trechos de rodovias, ao passo que estados pequenos tem trechos menores.

Isso é relevante porque a contagem dos quilómetros de um BR reinicia quando ela passa para outro estado. Por exemplo, a BR 040 inicia na cidade de Brasília, Distrito Federal (DF), no marco quilométrico 0 e vai até a fronteira com o estado do Goiás (GO), no marco 8,4. Quando a BR entra no GO, a contagem reinicia no marco quilométrico 0 e vai até o marco 157,3, na fronteira com Minas Gerais (MG). Quando entra em MG a contagem reinicia novamente.

Um outro exemplo é a BR 230. Ela tem um trecho de 1592 *km* no estado do Pará (PA), do km 0 ao km 1591,8. A mesma BR tem um trecho de 116 *km* no estado do Ceará (CE), do km 0 ao km 116,0. Essa descontinuidade da marcação dos quilómetros entre os estados leva a *outliers* relevantes no estudo. Um acidente no km 1500 da BR 230 no PA irá influenciar de forma negativa a análise dos dados do estado do CE, se for considerada a BR 230 de forma integral.

Objetivando contornar esse problema, foi definido que as análises dos dados seriam realizadas por regionais das PRF, que representam seus respetivos estados da federação. Foram escolhidas as regionais do Distrito Federal (DF), que representa os estados pequenos e com menos ocorrências de AT, dentro da distribuição nacional, a regional de São Paulo (SP) representado os estados com o número médio de ocorrências, e a regional de Santa Catarina (SC), que representa os estados com mais ocorrências. As análises para os dados nacionais foram realizadas apenas para comparação.

Entretanto, analisando os dados entres as regionais escolhidas, os erros encontrados permaneceram altos, conforme a Tabela 3. E isso ocorre porque, mesmo dentro dos estados, a distribuição da BRs também é discrepante. O estado de SC tem a BR 282 que contém 684 quilómetros, do km 0 ao 684,2, e a BR 480 com 133 *km*, do 0 ao 133,0.

Ainda com o objetivo de diminuir o erro e procurando agrupar dados com características semelhantes, foi definido que as análises também seriam realizadas por BR. Para isso foram escolhidas as BRs 020 e 040 no DF e 101 e 116 em SP e SC, que representam as rodovias com maior volume de ocorrência nesses estados.

Contudo, mesmo em uma análise por BR, os erros encontrados forma altos e aquém dos valores aceitáveis, conforme Tabela 3. Isso demonstra a grande aleatoriedade do acontecimento estudado, gerando baixas correlações entre as características. A distribuição das ocorrências hora a hora na BR 101 em SC no ano de 2017, por exemplo, representam bem essa aleatoriedade, conforme a Figura 25. Observa-se pontos de acumulação de sinistros, mas visualmente é impossível definir um padrão de ocorrências.

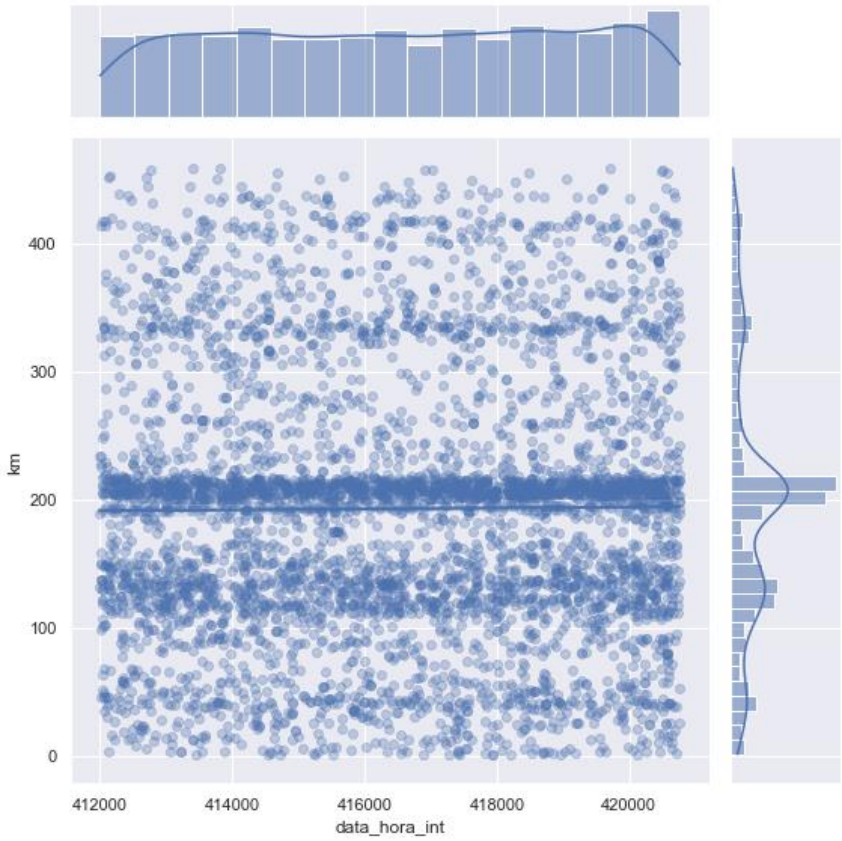

Figura 25 – Distribuição de acidentes hora a hora na BR 116 em SC no ano de 2017.

Mesmo altos, esses valores formaram o *baseline* para a aplicação dos algoritmos de ML, iniciada na próxima seção.

5.7 Machine Learning

Esta etapa do estudo também tem como objetivo prever o *km* de um AT, mas procurando superar os valores dos erros encontrados anteriormente, utilizando técnicas de *Machine Learning* (ML).

5.7.1 Regressão

Iniciando a tarefa de regressão de ML, foram aplicados os seguintes algoritmos: *Decision Tree Regressor, Random Forest Regressor, Linear SVR, SVR, Gradient Boosting Regressor, XGB Regressor, K-Neighbors Regressor, Ridge, SGD Regressor, Bayesian Ridge, Lasso, Elastic Net.*

Os resultados da aplicação destes algoritmos nos dados completos (nacional) podem ser visualizados na Tabela 4.

Tabela 4 – Resultados da regressão em ML nos dados completos (dados nacionais)

	Nacional		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**
XGB Regressor	153,78	0,53	3,8
Gradient Boosting Regressor	155,20	0,53	548,6
Random Forest Regressor	159,08	0,50	376,5
Random Forest Regressor com cross-validation (CV)	159,49	0,50	1691,1
Decision Tree Regressor	161,71	0,48	1,4
Decision Tree Regressor com CV	161,73	0,48	16,7
SVR com kernel="poly"	182,93	0,34	48457,2
Ridle	190,97	0,28	0,1
Elastic Net	191,24	0,28	0,7
Lasso	191,50	0,28	0,4
Linear SVR com CV	196,17	0,24	23,2
Linear SVR	196,19	0,24	2,3

Observa-se uma melhora interessante no valor do erro nos dados nacionais, passando de 225,38 da regressão estatística para 153,78 utilizando o algoritmo *XGB Regressor*, uma queda de 32%, com uma melhora considerável no R^2 *score*, marcando 0,53, superior ao valor mínimo aceitável. Mas mesmo com uma queda considerável, o erro RMSE nos dados nacionais encontra-se muito distante do alvo aceitável.

Procurando reduzir o tamanho dos *outliers* que estão presentes nos dados nacionais e o tempo de aplicação dos algoritmos num conjunto tão grande de dados, as próximas análises serão realizadas somente nos dados das regionais escolhidas. A aplicação dos mesmos algoritmos dos dados nacionais nos dados da regional do Distrito Federal (DF), incluindo nos dados das BRs 040 e 020 especificamente, são visualizados abaixo.

Tabela 5 – Resultados da regressão em ML nos dados da regional do DF

	SR-DF			SR-DF BR 020			SR-DF BR 040		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**
Decision Tree Regressor CV	28,17	0,51	0,5	44,44	0,48	0,9	27,55	0,03	0,8
Gradient Boosting Regressor	28,61	0,49	5,4	45,44	0,45	0,9	27,28	0,05	1,3
Decision Tree Regressor	30,81	0,41	0,4	44,14	0,48	0,3	27,28	0,05	0,3
Random Forest Regressor CV	31,18	0,40	53,6	48,11	0,39	20,2	30,49	-0,19	31,6
Random Forest Regressor	31,18	0,40	5,4	48,11	0,39	1,9	30,50	-0,19	4,1
XGB Regressor	32,94	0,33	0,2	55,00	0,20	0,1	32,69	-0,36	0,1
Elastic Net	37,95	0,11	0,1	46,41	0,43	0,3	27,55	0,03	0,3
Lasso	37,97	0,11	0,4	44,89	0,47	0,3	27,39	0,04	0,3
Ridle	38,04	0,10	0,4	44,94	0,46	0,3	27,84	0,01	0,3
SVR com kernel="rbf"	38,22	0,10	2,4	57,70	0,12	0,2	35,42	-0,60	0,4
K-neighbors Regressor	38,37	0,09	25,9	48,71	0,37	2,1	28,90	-0,06	3,3
SVR com kernel="poly"	39,09	0,06	21,9	65,54	-0,14	1,5	35,26	-0,59	2,6
Linear SVR	39,15	0,05	0,4	47,73	0,40	0,3	29,48	-107,00	0,3
Linear SVR com CV	39,16	0,05	0,5	47,70	0,40	0,3	29,46	-106,00	0,4
SVR com kernel="linear"	39,29	0,05	11,6	47,14	0,41	0,6	31,44	-0,26	2,4

Observa-se uma melhora do erro RMSE com a aplicação da regressão de ML nos dados da regional do DF, caindo de 40,21 para 28,17 com o algoritmo *Decision Tree Regressor*, queda de 30%, com o R^2 *score* em 0,51. No caso dos dados da BR 020, relativos à regional do DF, o erro caiu de 61,41 para 44,14 com o algoritmo *Decision Tree Regressor*, queda de 28%, com o R^2 *score* em 0.48. Para a BR 040 o erro caiu de 28,25 para 27,28 com os algoritmos *Decision Tree Regressor* e *Gradient Boosting Regressor*, queda de somente 3%, com o R^2 *score* em 0,03.

Nos dados da regional de São Paulo (SP), incluindo a aplicação nos dados das BR 101 e 116 especificamente, os resultados podem ser vistos abaixo.

Tabela 6 – Resultados da regressão em ML nos dados da regional de SP

	SR-SP			SR-SP BR 101			SR-SP BR 116		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**
Decision Tree Regressor	109,99	0,34	0,4	25,29	0,00	0,5	125,19	0,05	0,3
Decision Tree Regressor CV	109,99	0,34	0,6	25,31	0,00	0,7	125,40	0,05	0,5
Gradient Boosting Regressor	110,17	0,34	0,7	25,31	0,00	0,4	125,13	0,05	3,2
Random Forest Regressor	114,15	0,29	17,6	26,27	-0,08	0,9	129,97	-0,02	14,5
Random Forest Regressor CV	114,15	0,29	167,2	26,27	-0,08	9,2	129,97	-0,02	145,1
XGB Regressor	115,45	0,28	0,2	26,24	-0,08	0,1	133,32	-0,08	0,2
Elastic Net	118,86	0,23	0,2	25,86	-0,05	0,3	125,25	0,05	0,2
Ridle	118,89	0,23	0,3	25,95	-0,06	0,3	125,31	0,05	0,3
Lasso	118,89	0,23	0,1	25,28	0,00	0,3	125,28	0,05	0,4
Linear SVR	119,70	0,22	0,4	30,82	-0,49	0,3	125,15	0,05	0,4
Linear SVR com CV	119,76	0,22	0,5	30,81	-0,49	1,2	125,17	0,05	0,5
SVR com kernel="linear"	119,77	0,22	47,4	25,87	-0,05	0,7	125,12	0,05	16,1
K-neighbors Regressor	122,69	0,18	103,2	26,06	-0,06	0,2	127,77	0,01	50,9
SVR com kernel="poly"	123,75	0,17	52,7	41,78	-1,74	0,4	136,14	-0,12	29,7
SVR com kernel="rbf"	124,16	0,16	25,2	31,45	-0,55	0,3	133,76	-0,09	11,3

No caso da regional de SP, percebe-se que o erro RMSE caiu de 135,75 para 109,99 com a utilização do algoritmo *Decision Tree Regressor*, queda de 19%, com o R^2 *score* em 0,34. Para a BR 101 o erro caiu de 25,48 para 25,28 com o algoritmo *Lasso*, queda de somente 1%, com o R^2 *score* em 0,0. Na BR 116 o erro caiu de 128,50 para 125,12 com o algoritmo SVR, queda de 3%, com o R^2 *score* em 0,05.

O resultado da aplicação dos algoritmos nos dados da regional de Santa Catarina (SC), incluindo nos dados das BRs 101 e 116 especificamente, são visualizados abaixo.

Tabela 7 – Resultados da regressão em ML nos dados da regional de SC

	SR-SC			SR-SC BR 101			SR-SC BR 116		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**
Decision Tree Regressor CV	120,77	0,20	1,1	101,55	0,00	0,6	90,72	0,03	1,0
Decision Tree Regressor	120,77	0,20	0,8	101,55	0,00	0,4	90,72	0,03	0,2
Random Forest Regressor CV	123,91	0,16	455,9	103,55	-0,04	179,4	93,17	-0,02	19,3
Random Forest Regressor	123,91	0,16	44,8	103,55	-0,04	17,7	93,17	-0,02	1,9
XGB Regressor	124,80	0,15	0,3	105,10	-0,07	0,2	100,29	-0,18	0,1
Ridle	132,84	0,03	0,4	101,58	0,00	0,3	89,77	0,05	0,6
Elastic Net	132,84	0,03	0,3	101,58	0,00	0,7	89,81	0,05	0,3
Lasso	132,94	0,31	0,4	101,69	0,00	0,4	90,60	0,04	0,6
Gradient Boosting Regressor	133,25	0,03	4,5	101,55	0,00	0,6	91,55	0,02	0,6
K-neighbors Regressor	135,09	0,00	333,2	103,78	-0,04	72,7	94,80	-0,06	1,2
SVR com kernel="linear"	135,61	-0,01	274,1	103,12	-0,03	36,9	90,93	0,03	0,3
Linear SVR com CV	135,74	-0,01	0,9	102,68	-0,02	0,5	90,07	0,05	0,7
Linear SVR	135,76	-0,01	0,9	102,66	-0,02	0,4	90,05	0,05	0,3
SVR com kernel="rbf"	138,41	-0,05	216,5	104,93	-0,07	19,7	108,46	-0,38	0,1
SVR com kernel="poly"	139,94	-0,07	250,4	104,03	-0,05	55,1	123,47	-0,79	0,5

Observa-se uma melhora nos dados da regional de SC, com o erro RMSE caindo de 135,10 para 120,77 com a utilização do algoritmo *Decision Tree Regressor*, queda de 24%, com o R^2 *score* em 0,20. No caso dos dados da BR 101, o erro caiu de 101,60 para 101,55 com os algoritmos *Decision Tree Regressor* e *Gradient Boosting Regressor*, queda de menos de 1%, com o R^2 *score* marcando 0,0. Para a BR 116 o erro caiu de 92,31 para 89,77 com o algoritmo *Ridle*, queda de 3%, com o R^2 *score* em 0,05.

Os resultados da aplicação dos algoritmos de regressão de ML mostram uma melhora geral no valor do erro RMSE em relação à regressão estatística, mas nada muito significativo. Foram poucos os casos em que o R^2 *score* ficou acima do mínimo aceitável. Apesar da melhora, os valores das métricas estão distantes dos objetivos estabelecidos.

Como uma última tentativa de diminuir a influência dos *outliers* entre as diversas rodovias o universo dos dados foi reduzido, na tentativa de encontrar grupos que tenham uma maior correlação entre seus dados. Como uma rodovia dentro de uma regional pode ter diversas unidades operacionais (UOPs), a redução do universo dos dados a esse espaço, uma UOP, pode revelar uma melhor correlação entre as características.

Para isso foram escolhidas como amostras a primeira UOP da primeira delegacia de cada uma das regionais anteriores. Essas UOPs são as que tem o volume de AT, pois estão próximas das capitais dos respetivos estados. Os resultados encontram-se abaixo.

Tabela 8 – Resultados da regressão em ML nos dados da UOPs 01 das regionais DF, SP e SC

	SR-DF BR 040 DEL01 UOP01			SR-SP BR 116 DEL01 UOP01			SR-SC BR 101 DEL01 UOP01		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**	**Erro**	**R^2 Score**	**Tempo (s)**
Decision Tree Regressor	6,75	0,10	0,3	35,35	0,01	0,3	7,46	0,01	0,4
Elastic Net	6,79	0,09	0,3	35,36	0,01	0,4	7,49	0,00	0,3
Ridle	6,81	0,09	0,3	35,37	0,01	0,3	7,51	0,00	0,3
Decision Tree Regressor CV	6,75	0,10	0,7	35,58	0,00	1,0	7,46	0,01	1,0
Gradient Boosting Regressor	6,76	0,10	1,4	35,64	-0,01	1,1	7,63	-0,04	2,5
Lasso	6,78	0,09	0,3	35,70	-0,01	0,3	7,51	0,00	0,3
Linear SVR	11,79	-1,74	0,4	35,74	-0,01	0,3	17,45	-4,42	0,3
Linear SVR com CV	11,79	-1,74	0,7	35,74	-0,01	0,9	17,45	-4,42	1,2
K-neighbors Regressor	6,96	0,05	2,5	36,03	-0,03	4,5	7,59	-0,03	5,7
SVR com kernel="rbf"	23,26	-9,65	0,3	36,11	-0,03	0,1	11,15	-1,21	0,3
Random Forest Regressor	7,15	-0,02	4,7	36,13	-0,04	3,9	7,65	-0,04	4,3
Random Forest Regressor CV	7,21	-0,02	27,5	36,13	-0,04	39,5	7,65	-0,04	43,9
SVR com kernel="linear"	21,88	-8,43	0,3	36,70	-0,07	10,6	11,15	-1,21	0,3
XGB Regressor	8,02	-0,26	0,1	38,20	-0,16	0,1	8,25	-0,21	0,1
SVR com kernel="poly"	23,07	-9,48	0,3	39,50	-0,24	1,1	11,15	-1,21	0,3

Os resultados obtidos na aplicação dos algoritmos nos dados das UOPs contêm valores de erro RMSE inferiores do que os encontrados nas análises das regionais e BRs, mas ao custo de uma diminuição significativa no R^2 *score*. Isso indica que a diminuição do erro é devido a natural redução da dimensão do intervalo contínuo presente em cada *feature* nos dados das regionais e BRs, bem maiores que o intervalo contínuo presente nos dados das UOPs. Não foi uma melhoria da predição de cada algoritmo, mas apenas uma diminuição no intervalo numérico disponível para o cálculo do erro. O baixo R^2 *score* indica que os modelos das UOPs explicam menos as predições realizadas.

5.7.2 Classificação

Ainda na área de *Machine Learning* (ML), foram aplicados alguns algoritmos de classificação com o objetivo diminuir o erro RMSE estabelecido na análise anterior. Os algoritmos utilizados foram: *Decision Tree Classifier, Random Forest Classifier, Bagging Classifier, AdaBoost Classifier, Logistic Regression, Linear SVC* e *SVC*.

Os algoritmos de classificação não são os mais indicados para os dados desse estudo, visto que os dados são contínuos, mas eles foram empregados numa tentativa de superar os erros em uma abordagem diferente. Para possibilitar o processamento pelos algoritmos de classificação, os dados foram modificados, com os valores da *feature* alvo “km” transformados em *strings*, para que os algoritmos reconhecessem cada *string* como uma classe. Os algoritmos de classificação foram aplicados nos mesmos conjuntos de dados da seção anterior.

O objetivo ainda é prever o *km*, mas utilizando a classificação. Como o resultado da classificação da *feature km* é uma *string* que foi formada a partir de um número inteiro, essa

string será novamente transformada em inteiro, permitindo assim o cálculo do erro RMSE para a comparação com os resultados anteriores.

Os resultados obtidos revelam um pequeno aumento geral no erro RMSE. No caso da *accuracy*, métrica de avaliação escolhida para os algoritmos de classificação, os valores ficaram bem abaixo do mínimo aceitável, de 0,50. As análises de todos os conjuntos de dados estão no repositório da dissertação. Abaixo está a tabela relativa à análise da regional do DF.

Tabela 9 – Resultados da classificação em ML nos dados da regional do DF

	SR-DF			SR-DF BR 020			SR-DF BR 040			SR-DF BR 040 DEL01 UOP01		
Erro: RMSE (*km*)	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**
Decision Tree Classifier	35,96	0,08	0,6	64,92	0,03	0,4	31,30	0,10	0,6	9,39	0,11	0,7
Linear SVC	45,41	0,06	1,2	64,19	0,04	0,3	30,44	0,08	0,2	7,97	0,09	0,1
Bagging Classifier	45,42	0,07	0,4	70,21	0,08	3,3	30,61	0,07	2,8	8,56	0,09	2,6
SVC	45,65	0,06	10,7	63,60	0,05	0,9	31,38	0,09	1,2	8,80	0,10	0,7
Random Forest Classifier	45,71	0,07	0,3	61,96	0,04	0,6	31,29	0,10	0,6	8,84	0,11	0,6
Logistic Regression	48,58	0,06	39,3	61,47	0,04	5,9	53,80	0,05	7,1	8,51	0,08	1,9
AdaBoost Classifier	66,56	0,02	19,3	72,34	0,00	1,2	37,55	0,09	1,5	9,43	0,11	0,5

Observa-se que todos os valores dos erros encontrados foram piores com os algoritmos de classificação em relação aos de regressão. O mesmo se verifica nas regionais de SP e SC.

Tabela 10 – Resultados da classificação em ML nos dados da regional de SP

	SR-SP			SR-SP BR 101			SR-SP BR 116			SR-SP BR 116 DEL01 UOP01		
Erro: RMSE (*km*)	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**
Decision Tree Classifier	115,05	0,02	0,7	25,42	0,08	0,3	142,12	0,01	0,8	35,98	0,04	0,3
Random Forest Classifier	115,29	0,02	1,9	25,30	0,06	0,5	136,77	0,01	1,7	36,04	0,04	0,6
Bagging Classifier	116,60	0,02	5,6	25,49	0,10	2,5	133,39	0,01	5,5	36,39	0,03	2,6
SVC	143,92	0,02	70,7	25,25	0,06	0,9	150,80	0,01	36,4	36,04	0,04	1,5
Linear SVC	145,77	0,01	6,9	25,49	0,10	0,4	151,56	0,01	3,7	36,73	0,03	0,1
AdaBoost Classifier	148,20	0,01	22,8	25,53	0,05	0,1	171,77	0,01	15,6	36,99	0,04	0,7
Logistic Regression	152,10	0,02	125,7	51,22	0,06	0,3	162,19	0,01	56,7	41,83	0,00	2,2

Tabela 11 – Resultados da classificação em ML nos dados da regional de SC

	SR-SC			SR-SC BR 101			SR-SC BR 116			SR-SC BR 101 DEL01 UOP01		
Erro: RMSE (*km*)	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**	**Erro**	**Accuracy**	**Tempo (s)**
Linear SVC	153,59	0,02	23,1	103,38	0,02	4,6	121,58	0,02	0,3	9,53	0,09	0,1
Bagging Classifier	164,54	0,02	8,4	103,01	0,02	6,2	129,88	0,03	4,7	9,47	0,09	2,9
SVC	164,81	0,02	228,7	102,43	0,02	59,6	152,73	0,01	0,7	9,70	0,08	2,7
Random Forest Classifier	168,66	0,02	3,3	102,31	0,02	1,9	152,92	0,01	0,9	9,58	0,09	0,7
Decision Tree Classifier	169,18	0,02	0,1	102,22	0,02	0,1	138,37	0,01	0,1	9,52	0,07	0,4
AdaBoost Classifier	189,42	0,01	51,4	147,17	0,01	20,1	157,69	0,02	1,4	13,62	0,05	0,7
Logistic Regression	202,34	0,01	654,6	110,73	0,01	77,8	121,05	0,02	10,4	9,47	0,09	1,6

5.8 ANN

Após a aplicação dos algoritmos de regressão e classificação de ML, esta etapa do estudo objetiva a aplicação de redes neuronais. O objetivo é melhorar as métricas anteriores utilizando redes ANN. Os conjuntos de dados serão os mesmo da seção anterior: os dados das regionais do DF, SP e SC, e os dados das rodovias escolhidas. Os resultados da aplicação de ANN encontram-se na Tabela 12.

Tabela 12 – Resultados da aplicação da ANN e comparação com os melhores resultados de ML

	ANN		
Erro: RMSE (*km*)	**Erro**	**R^2 Score**	**Tempo (s)**
DF	29,06	0,48	30,1
DF BR 020	45,64	0,45	4,2
DF BR 040	27,13	0,06	5,6
DF BR 040 DEL01 UOP01	6,77	0,10	1,1
SP	110,69	0,33	112,1
SP BR 101	25,72	-0,04	2,2
SP BR 116	124,99	0,05	32,5
SP BR 116 DEL01 UOP01	35,42	-0,02	15,2
SC	128,87	0,09	138,3
SC BR 101	101,49	0,00	30,4
SC BR 116	89,51	0,06	4,2
SC BR 101 DEL01 UOP01	7,48	0,00	20,3

Observa-se que os resultados da utilização de redes ANN foram bem próximos dos resultados da utilização dos melhores algoritmos de ML para cada conjunto de dados. Em alguns conjuntos de dados o resultado foi até melhor que com os algoritmos de ML. A comparação dos resultados de ANN com os melhores resultados anteriores pode ser analisada na Tabela 13.

Tabela 13 – Comparação dos resultados entre ANN e os algoritmos de ML

	ANN		ML	
Erro: rmse (km)	**Erro**	**R^2 Score**	**Erro**	**R^2 Score**
DF	29,06	0,48	28,17	0,51
DF BR 020	45,64	0,45	44,14	0,48
DF BR 040	27,13	0,06	27,28	0,05
DF BR 040 DEL01 UOP01	6,77	0,10	6,75	0,10
SP	110,69	0,33	109,99	0,34
SP BR 101	25,72	-0,04	25,28	0,00
SP BR 116	124,99	0,05	125,12	0,05
SP BR 116 DEL01 UOP01	35,42	-0,02	35,35	0,01
SC	128,87	0,09	120,77	0,31
SC BR 101	101,49	0,00	101,55	0,00
SC BR 116	89,51	0,06	89,77	0,05
SC BR 101 DEL01 UOP01	7,48	0,00	7,46	0,01

Assim como aconteceu com os algoritmos de ML, a utilização de redes ANN não foi nada satisfatória. Os erros RMSE foram acima do valor limite aceitável de 5 quilómetros e os R^2 *scores* ficaram abaixo do mínimo definido, de 0,50, excetuando no conjunto de dados da regional do DF que foi de 0,51.

5.9 Avaliação

Um acidente de trânsito é um evento com uma aleatoriedade muito grande. Os fatores envolvidos em suas causas são diversos e a previsão do local de uma nova ocorrência é muito difícil. A utilização de técnicas de Aprendizagem Automática na identificação desse locais comprova isso. Apesar das técnicas de ML e DL empregadas se mostrarem melhores que a aplicação de técnicas estatísticas convencionais, os resultados não são significativos, como o desejado. Os erros encontrados foram muito acima do esperado.

É importante considerar que, para a maioria das aplicações, é impossível alcançar o erro zero absoluto. Isso ocorre porque as *features* dos dados de entrada podem não conter informações completas sobre a variável alvo ou porque o sistema pode ser intrinsecamente estocástico. Outro limitador pode ser ter uma quantidade finita de dados de treinamento. (Goodfellow, Bengio e Courville, 2016).

A quantidade de dados de treinamento pode ser limitada por vários motivos. Quando o objetivo é criar o melhor produto ou serviço possível do mundo real, normalmente pode-se coletar mais dados, mas deve-se determinar o valor de reduzir ainda mais esse erro, e ponderar isso em relação ao custo de coletar mais dados. A coleta de dados pode exigir tempo, dinheiro ou sofrimento humano (por exemplo, se o processo de coleta de dados envolver a realização de exames médicos invasivos). Quando o objetivo é responder a uma pergunta científica sobre qual algoritmo tem melhor desempenho em um conjunto fixo de dados, a quantidade de dados determina o conjunto de treinamento, e pode ser impossível coletar mais dados (Goodfellow, Bengio e Courville, 2016).

Tal limitação ficou evidente no contexto deste estudo. Quantitativamente o conjunto de dados é pequeno e é impossível se coletar mais dados, além de indesejável devido a natureza da ocorrência. Em termos qualitativos isso foi determinante. A qualidade das informações persentes nas *features* dos dados é basicamente irrelevante para a determinação do local de uma nova ocorrência de AT. Do conjunto de dados originais da PRF, que contém 29 *features*, apenas 8 delas – *data_hora*, *dia_semana*, *uf*, *br*, *condicao_metereologica*, *regional*, *delegacia* e *uop* – puderam ser utilizadas para o emprego dos algoritmos. E quase todas elas tinham baixíssima correlação com a variável alvo, que era o “km”, conforme Figura 20.

É preciso estar ciente da hipótese que se está fazendo: de que os alvos podem ser preditos com base nas entradas. Ou seja, a hipótese de que os dados disponíveis são suficientemente informativos para aprender a relação entre entradas e alvos. Até que se tenha um modelo de trabalho, estas são apenas hipóteses, esperando para serem validadas ou invalidadas. Nem

todos os problemas podem ser resolvidos com Aprendizagem Automática; só porque se reuniu exemplos de entradas *X* e alvos *y* não significa que *X* contém informações suficientes para prever *y* (Chollet, 2017).

A hipótese definida no início do trabalho foi que as técnicas de ML e DL superariam consideravelmente as técnicas de estatística convencionais, possibilitando predições confiáveis. Claramente essa hipótese não foi confirmada. Os algoritmos de ML e DL superaram as técnicas de estatística convencionais, mas não de forma considerável a ponto de possibilitar predições úteis dos locais de ocorrência de AT. Os erros encontrados foram muito acima do limite aceitável.

Observe que nem sempre é possível obter poder estatístico. Se não é possível superar uma *baseline* simples depois de tentar várias arquiteturas razoáveis, pode ser que a resposta para a pergunta que se está fazendo não esteja presente nos dados de entrada. *Deep learning* é o ajuste de uma curva à um problema, não mágica (Chollet, 2017)!

5.10 Comparação com estudos semelhantes

A comparação desse trabalho com outros que objetivaram a identificação do local de ocorrência de um novo acidente não pode realizada de uma forma direta. A metodologia empregada é bem diversa entre os trabalhos.

No trabalho realizado por Santos et al. (2021), os autores completaram a quantidade de dados de ocorrências de AT que possuíam com o triplo de dados onde não ocorriam a ocorrência dos sinistros. Eles utilizaram os algoritmos de *Logistic Regression, Random Forest e Decision Trees* em um conjunto de dados que possuía 13 *features*, dentre elas: condição da via, direção do vento e histórico de velocidade média na via. Eles obtiveram uma *accuracy* de 73% na predição com classificação.

No estudo de Wenqi, Dongyu e Menghua (2017), eles aplicaram uma CNN a um conjunto de dados que continham *features* como velocidade média, volume de veículos, ocupação da via, iluminação e condição climática, e obtiveram uma *accuracy* de 78,5%. No estudo de Yuan, Zhou e Yang (2018), eles utilizaram uma *Convolutional Long Short-Term Memory Neural Network* (LSTM) para analisar um conjunto de dados que tinham como *features* volume de tráfego, volume de chuva, temperatura e imagens de satélite, dentre outros, obtendo bons resultados na predição do local de um novo AT.

O que esses estudos comprovam é a importância da qualidade e relevância das *features* para um trabalho de Aprendizagem Automática. Eles corroboram o diagnóstico de que o conjunto de dados utilizado no estudo dessa dissertação não possuía informações necessária para a predição dos algoritmos empregados de forma satisfatória.

5.11 Sumário

Neste capítulo foram descritas as etapas da aplicação das técnicas de AA. Os dados históricos de AT da PRF foram tratados seguindo as melhores práticas. Depois da engenharia de dados, eles foram divididos em três conjuntos, segundo a técnica *holdout*. Com os dados prontos, foi aplicada a técnica de regressão linear objetivando definir uma *baseline* para ser utilizada como ponto de partida para a aplicação dos algoritmos de ML e DL. Após o emprego dos algoritmos, os resultados foram relatados, e foi realizada uma avalização dos mesmos. A avaliação demonstrou que a aplicação das técnicas de AA não se mostrou satisfatória. Em seguida foi realizada uma comparação dos resultados com os resultados de outros trabalhos da área.

6 Conclusão

O presente documento descreve o estudo da aplicação de técnicas de ML e DL utilizadas realizar a identificação e predição do local de ocorrência de um novo AT. Ele é composto dos seguintes capítulos: contextualização do problema e definição dos objetivos, revisão da literatura e estado da arte, análise de valor, metodologia, estudo computacional e conclusão.

A análise dos fatores que influenciam as ocorrências de AT é primordial para a PRF. A identificação do local que deve receber a maior atenção e trabalho a fim de mitigar a ocorrência novos de sinistros é fundamental, e os esforços para melhorar esse método de identificação são constantes. A PRF acredita que o emprego de técnicas modernas, como a utilização de técnicas de Aprendizagem Automática, são o próximo passo nesse caminho.

O objetivo deste trabalho foi avaliar aplicação e a capacidade preditiva dos algoritmos de AA na identificação dos locais com maior probabilidade de ocorrências de AT, utilizando para isso os dados históricos de acidentes da PRF.

Os resultados encontrados após o estudo computacional demonstraram que a utilização de técnicas de ML e DL na identificação dos locais de novos AT não foi satisfatória. A capacidade de predição dos algoritmos empregados não foi o esperado, com os erros medidos muito acima dos limites estabelecidos. A avaliação dos resultados obtidos sugere que as causas prováveis são a grande aleatoriedade das ocorrências presentes nos dados, que dificulta a determinação de padrões por parte dos algoritmos, e a baixa qualidade das informações presentes nos dados, que não contêm as correlações necessárias à identificação do alvo, demonstrando que eles não têm a relevância necessária para a identificação dos locais de acidentes.

A hipótese de investigação proposta, que as técnicas de ML e DL superariam consideravelmente as técnicas de estatística convencionais a ponto de possibilitar predições confiáveis dos locais de AT, não foi confirmada. Os resultados do emprego dos algoritmos de AA foram superiores aos resultados do emprego da regressão linear, mas a capacidade de predição foi bem abaixo do esperado, invalidando a hipótese.

Em relação as perguntas complementares formuladas, as respostas são: os dados de geolocalização não são relevantes e não puderam ser utilizados na aplicação dos algoritmos de AA; as informações de condição meteorológica e ocorrência do acidente em um dia de feriado não são relevantes para as predições, pois tinham baixíssima correlação com o alvo do estudo; e, por fim, os dados utilizados não são suficientemente relevantes para uma aplicação satisfatória dos algoritmos utilizados.

O resultado deste estudo revela à PRF que a mudança do seu método de identificação atual para um que utilize a Aprendizagem Automática pode não ser a melhor opção. Talvez um aprimoramento do seu método atual, que utiliza mapas de calor, utilizando as informações de geolocalização presentes nos dados seja a melhor alternativa. Apesar dos registos de latitude e longitude nos dados conterem locais equivocados, eles podem ser aperfeiçoados com o tempo, auxiliando a identificação dos locais de AT pelo método de mapas de calor.

6.1 Contribuições deste trabalho

O resultado deste estudo contribui para a PRF ao revelá-la que a utilização de AA não contribui na melhoria do seu método de identificação de locais de AT, contrariando o que a instituição acreditava.

Este estudo, apesar de insatisfatório, contribui com o campo da Aprendizagem Automática ao reforçar que há um limite em sua aplicação. Esta contribuição é tanto no sentido de comprovar que a qualidade das informações presente nas *features* dos dados é determinante para sucesso do emprego de um algoritmo, quanto no sentido de comprovar que existem tipos de problemas que talvez não possam ser resolvidos por meio da utilização de AA.

6.2 Limitações deste trabalho

O resultado da execução deste estudo foi influenciado por dois fatores limitantes. Primeiro a qualidade dos dados disponíveis. Além dos registos presentes nos dados serem muito inconsistentes, necessitando um grande tratamento inicial para deixá-los aptos a serem utilizados, a informação presente não continha a relevância necessária para um resultado satisfatório do trabalho.

Segundo foi a ausência inicial de conhecimento formal e prático do assunto por parte do autor. O domínio de um campo da ciência tão grande como a AA requer anos de estudos e experiência. A falta de um estudo académico formal e sistemático, e a sua devida prática, se mostrou um fator relevante, apesar de não determinante, no resultado do trabalho.

6.3 Propostas de trabalhos futuros

Uma proposta de trabalho futuro seria a aplicação das técnicas *Convolutional Neural Networks* (CNN), *Recurrent Neural Network* (RNN) e *Long Short-Term Memory* (LSTM) para a predição dos locais de AT, com a devida modificação dos dados que permita essa análise.

Outra seria a utilização de técnicas de aprendizagem não supervisionada, como as de clusterização, objetivando aproveitar a vantagem das informações de geolocalização presentes nos dados, possibilitando a identificação dos locais com maiores ocorrências de acidentes. Essa aplicação não estava prevista no escopo deste trabalho.

Referências

Andrade, F.R. de and Antunes, J.L.F. (2019) “Tendência do número de vítimas em acidentes de trânsito nas rodovias federais brasileiras antes e depois da Década de Ação pela Segurança no Trânsito,” *Cadernos de Saúde Pública*, 35(8). Available at: https://doi.org/10.1590/0102-311X00250218.

Barros, J. (2008) “As Hipóteses nas Ciências Humanas,” *Sísifo: Revista de Ciências da Educação, ISSN 1649-4990, Nº. 7, 2008 (Ejemplar dedicado a: Educação, Trabalho e Identidades Profissionais), pags. 151-162* [Preprint].

Barroso Júnior, G.T., Bertho, A.C.S. and Veiga, A. de C. (2019) “A letalidade dos acidentes de trânsito nas rodovias federais brasileiras em 2016,” *Revista Brasileira de Estudos de População*, 36. Available at: https://doi.org/10.20947/S0102-3098A0074.

Bishop, C.M. (2006) *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag.

Burkov, A. (2019) *The Hundred-page Machine Learning Book*. Andriy Burkov.

Carvalho, C.H.R. (2020) “Custos dos acidentes de trânsito no Brasil : estimativa simplificada com base na atualização das pesquisas do Ipea sobre custos de acidentes nos aglomerados urbanos e rodovias,” *http://www.ipea.gov.br* [Preprint]. Available at: http://repositorio.ipea.gov.br/handle/11058/10075 (Accessed: September 23, 2022).

Chollet, F. (2017) *Deep Learning with Python*. 1st edn. USA: Manning Publications Co.

Confederação Nacional do Transporte (2018) *Acidentes Rodoviários e a Infraestrutura - íntegra*. Available at: https://cnt.org.br/acidentes-rodoviarios-infraestrutura (Accessed: September 23, 2022).

Confederação Nacional do Transporte (2022) *Painel CNT de Acidentes Rodoviários*. Available at: https://www.cnt.org.br/painel-acidente (Accessed: September 23, 2022).

Facebook (2022) *Facebook – entre ou cadastre-se*. Available at: https://www.facebook.com/ (Accessed: October 10, 2022).

Frost, J. (2022a) *How To Interpret R-squared in Regression Analysis - Statistics By Jim*. Available at: https://statisticsbyjim.com/regression/interpret-r-squared-regression/ (Accessed: October 2, 2022).

Frost, J. (2022b) *Mean Squared Error (MSE) - Statistics By Jim*. Available at: https://statisticsbyjim.com/regression/mean-squared-error-mse/ (Accessed: October 2, 2022).

Geron, A. (2019) *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow*. Sebastopol: O’Reilly Media, Incorporated.

GitHub (2022) *GitHub - pytorch/pytorch: Tensors and Dynamic neural networks in Python with strong GPU acceleration*. Available at: https://github.com/pytorch/pytorch#more-about-pytorch (Accessed: September 30, 2022).

Goldschmidt, R., Passos, E. and Bezerra, E. (2015) *Data Mining: Conceitos, tecnicas, algoritmos, orientacoes e aplicacoes*. Elsevier Editora Ltda. Available at: https://www.researchgate.net/publication/309354464_Data_Mining_Conceitos_tecnicas_algoritmos_orientacoes_e_aplicacoes (Accessed: October 3, 2022).

Goodfellow, I., Bengio, Y. and Courville, A. (2016) *Deep Learning*. MIT Press.

GÜREL, E. (2017) "SWOT ANALYSIS: A THEORETICAL REVIEW," *Journal of International Social Research*, 10(51), pp. 994–1006. Available at: https://doi.org/10.17719/JISR.2017.1832.

Kaggle (2022) *Kaggle: Your Machine Learning and Data Science Community*. Available at: https://www.kaggle.com/ (Accessed: September 25, 2022).

Kaleko, D. (2017) *From Neutrinos to Data Science*. Available at: https://blog.davidkaleko.com/feature-engineering-cyclical-features.html (Accessed: September 28, 2022).

Keras (2022) *Keras: the Python deep learning API*. Available at: https://keras.io/ (Accessed: October 10, 2022).

Lima, R.O. (2021) "Ensaios sobre segurança no trânsito." Available at: https://repositorio.ufpb.br/jspui/handle/123456789/21469 (Accessed: September 23, 2022).

Marsland, S. (2014) *Machine Learning: An Algorithmic Perspective, Second Edition*. 2nd edn. Chapman & Hall/CRC.

Martikainen, A. (2017) *Front End of Innovation in Industrial Organization*. Available at: https://doi.org/10.13140/RG.2.2.19644.36481.

Martins, M.A. (2020) "Modelo de decisão para redução de risco de acidentes de trânsito nas rodovias federais de Pernambuco : desenvolvimento de um Sistema de Apoio à Decisão." Available at: https://repositorio.ufpe.br/handle/123456789/37662 (Accessed: September 23, 2022).

Merriam-Webster (2022) *Value Definition & Meaning*. Available at: https://www.merriam-webster.com/dictionary/value (Accessed: September 21, 2022).

Ministério da Infraestrutura (2022) *Rodovias Federais - Informações Gerais - Sistema Federal de Viação — Português (Brasil)*. Available at: https://www.gov.br/infraestrutura/pt-br/assuntos/transporte-terrestre_antigo/rodovias-federais/rodovias-federais-informacoes-gerais-sistema-federal-de-viacao (Accessed: September 23, 2022).

Ministério dos Transportes (2002) *PROCEDIMENTOS PARA O TRATAMENTO DE LOCAIS CRÍTICOS DE ACIDENTES DE TRÂNSITO*.

Mitchell, T.M. (1997) *Machine learning*. McGraw-hill New York.

Morgan, P. (2017) *Data Analysis from Scratch with Python: Step-By-Step Guide*.

Müller, A.C. and Guido, S. (2016) *Introduction to Machine Learning with Python: A Guide for Data Scientists*. O'Reilly Media, Incorporated.

Oliveira, J.C.M. de and Gomes, E.M.D.C. (2021) "A UTILIZAÇÃO DE LOCAIS DE ACUMULAÇÃO DE ACIDENTES DE TRÂNSITO COMO NORTEADORES DE POLÍTICAS DE REDUÇÃO A ACIDENTES NAS RODOVIAS FEDERAIS BRASILEIRAS," *REVISTA BRASILEIRA DE ESTUDOS DE SEGURANÇA PÚBLICA - REBESP*, 14(2), pp. 81–96. Available at: https://doi.org/10.29377/REBESP.V14I2.563.

Organização das Nações Unidas (2020) *Global health estimates: Leading causes of death*. Available at: https://www.who.int/data/gho/data/themes/mortality-and-global-health-estimates/ghe-leading-causes-of-death (Accessed: September 23, 2022).

Organização das Nações Unidas (2021) *Acidentes de trânsito são a maior causa de morte de pessoas de 5 a 29 anos | ONU News*. Available at: https://news.un.org/pt/story/2021/11/1771092 (Accessed: September 23, 2022).

Pereira, D. (2019) *Canvas da Proposta de Valor - O Analista de Modelos de Negócios*. Available at: https://analistamodelosdenegocios.com.br/canvas-da-proposta-de-valor/ (Accessed: September 21, 2022).

Polícia Rodoviária Federal (2015) *RESPOSTA - PEDIDO - MPO 015 - Atendimento de Acidentes PDF | PDF | Tráfego | Transporte*. Available at: https://pt.scribd.com/document/372534748/RESPOSTA-PEDIDO-MPO-015-Atendimento-de-Acidentes-pdf (Accessed: September 23, 2022).

Polícia Rodoviária Federal (2021) *Plano Diretor de Tecnologia da Informação e Comunicação - PDTIC 2020-2023 — Português (Brasil)*. Available at: https://www.gov.br/prf/pt-br/acesso-a-informacao/acoes-e-programas/pdtic-2020-2023 (Accessed: September 23, 2022).

Polícia Rodoviária Federal (2022a) *Acidentes — Português (Brasil)*. Available at: https://www.gov.br/prf/pt-br/acesso-a-informacao/dados-abertos/dados-abertos-acidentes (Accessed: September 19, 2022).

Polícia Rodoviária Federal (2022b) *Dicionário de dados — Português (Brasil)*. Available at: https://www.gov.br/prf/pt-br/acesso-a-informacao/dados-abertos/dicionario-de-dados (Accessed: September 19, 2022).

Portal da Transparência (2022) *Departamento de Polícia Rodoviária Federal - DPRF/MJ - Portal da transparência*. Available at: https://www.portaltransparencia.gov.br/orgaos/30802-departamento-de-policia-rodoviaria-federal-mj (Accessed: September 23, 2022).

PyTorch (2022) *PyTorch strengthens its governance by joining the Linux Foundation | PyTorch*. Available at: https://pytorch.org/blog/PyTorchfoundation/ (Accessed: September 30, 2022).

R Core Team (2022) *R Language Definition*. Available at: https://cran.r-project.org/doc/manuals/r-release/R-lang.html#Introduction (Accessed: September 30, 2022).

Raschka, S. (2019) *Python Machine Learning*. Packt Publishing.

Santos, D. *et al.* (2021) "Machine Learning Approaches to Traffic Accident Analysis and Hotspot Prediction," *Computers 2021, Vol. 10, Page 157*, 10(12), p. 157. Available at: https://doi.org/10.3390/COMPUTERS10120157.

scikit-learn (2022) *scikit-learn: machine learning in Python — scikit-learn 1.1.2 documentation*. Available at: https://scikit-learn.org/stable/ (Accessed: October 10, 2022).

Silva Júnior, C.M. da (2019) "SIGA Seguro: uma aplicação para análise de trechos críticos de acidentes de trânsito nas rodovias federais." Available at: https://bdm.unb.br/handle/10483/26499 (Accessed: September 23, 2022).

Silveira, M.T. da (2009) "Avaliação econômica dos riscos de acidentes de trânsito em rodovias federais brasileiras." Available at: https://repositorio.unb.br/handle/10482/6294 (Accessed: September 23, 2022).

Stack Overflow (2021) *Stack Overflow Developer Survey 2021*. Available at: https://insights.stackoverflow.com/survey/2021#most-loved-dreaded-and-wanted-language-want (Accessed: September 23, 2022).

TensorFlow (2022) *TensorFlow*. Available at: https://www.tensorflow.org/?hl=pt-br (Accessed: October 10, 2022).

Teodoro, A.B., Alcantara, F.A. and Barbosa, H.M. (2014) "COMPARAÇÃO ENTRE DOIS MÉTODOS PARA IDENTIFICAÇÃO DE LOCAIS CRÍTICOS DE ACIDENTES DE TRÂNSITO," *ANPET* [Preprint].

Theobald, O. (2017) *Machine Learning for Absolute Beginners: A Plain English Introduction*. Scatterplot Press. (Machine Learning from Scratch Series).

Trask, A. (2019) *Grokking Deep Learning*. 1st edn. USA: Manning Publications Co.

Wenqi, L., Dongyu, L. and Menghua, Y. (2017) "A model of traffic accident prediction based on convolutional neural network," *2017 2nd IEEE International Conference on Intelligent Transportation Engineering, ICITE 2017*, pp. 198–202. Available at: https://doi.org/10.1109/ICITE.2017.8056908.

Yuan, Z., Zhou, X. and Yang, T. (2018) "Hetero-ConvLSTM: A deep learning approach to traffic accident prediction on heterogeneous spatio-temporal data," *Proceedings of the ACM SIGKDD*

International Conference on Knowledge Discovery and Data Mining, 18, pp. 984–992. Available at: https://doi.org/10.1145/3219819.3219922.

Anexo 1

Sítios que contém as portarias que definem os feriados nacionais no Brasil:

https://legis.sigepe.planejamento.gov.br/legis/detalhar/12946

https://legis.sigepe.planejamento.gov.br/legis/detalhar/14417

https://legis.sigepe.planejamento.gov.br/legis/detalhar/15091

https://legis.sigepe.planejamento.gov.br/legis/detalhar/19288

https://legis.sigepe.planejamento.gov.br/legis/detalhar/23077

https://legis.sigepe.planejamento.gov.br/legis/detalhar/23369