PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS
NÚCLEO DE EDUCAÇÃO A DISTÂNCIA
Pós-graduação *Lato Sensu* em Ciência de Dados e Big Data

Raquel Cafaro Marinho

PREVISÃO DA OCORRÊNCIA DE FERIMENTOS EM VÍTIMAS DE ACIDENTES DE TRÂNSITO EM BELO HORIZONTE

Belo Horizonte
2023

Raquel Cafaro Marinho

PREVISÃO DA OCORRÊNCIA DE FERIMENTOS EM VÍTIMAS DE ACIDENTES DE TRÂNSITO EM BELO HORIZONTE

Trabalho de Conclusão de Curso apresentado ao Curso de Especialização em Ciência de Dados e Big Data como requisito parcial à obtenção do título de especialista.

Belo Horizonte
2023

SUMÁRIO

1. Introdução

1.1. Contextualização

Segundo estudo realizado pela Organização Pan-Americana da Saúde em 2018, estima-se que, anualmente, cerca de 1,3 milhão de pessoas morrem no mundo por acidentes de trânsito com um total de feridos podendo chegar a 50 milhões.

No ano de 2021 foram registrados 11.122 boletins de ocorrência de acidentes de trânsito com vítimas em Belo Horizonte, envolvendo um total de 23.736 pessoas. Além dos custos humanos, esses acidentes causam impacto direto no sistema de saúde, aumentando o número de atendimentos e internações, e na economia da cidade, por despesas médicas, perda de produtividade das vítimas e custos materiais.

Ao longo do tempo foram identificados comportamentos com potencial de aumentar a frequência e/ou a gravidade dos acidentes, como consumo de bebidas alcoólicas, não uso de cinto de segurança ou capacete e excesso de velocidade e foram tomadas medidas para diminuir a ocorrência desses comportamentos. Em 2008 foi sancionada a Lei 11.705, conhecida como Lei Seca, que reduzia a tolerância de 6 decigramas de álcool para 0,1mg de álcool por litro de sangue, sendo que hoje o máximo tolerado é 0,05mg/L. No mesmo ano também entrou em vigor a Resolução 277 do Conselho Nacional de Trânsito, conhecida como Lei da Cadeirinha, que dispõe sobre as regras para o transporte de crianças menores de 10 anos em veículos.

Também é conhecido que em determinados momentos é mais frequente a ocorrência de acidentes, como feriados, fins de semana e o período noturno. É papel de toda a sociedade cumprir as leis para a condução de veículos e estimular a direção consciente para diminuir a ocorrência dos acidentes e sua gravidade, minimizando assim os danos humanos e sociais.

1.2. O problema proposto

Conhecendo as características e condições que podem aumentar a ocorrência de ferimentos nas vítimas de acidentes de trânsito é possível criar ações públicas de conscientização e de primeiros socorros de forma mais assertiva a fim de minimizar a ocorrência de ferimentos e, consequentemente, de óbitos.

A BHTrans, Empresa de Transportes e Trânsito de Belo Horizonte, disponibiliza dados dos boletins de ocorrência de acidentes de trânsito com vítimas ocorridos em Belo Horizonte entre 2011 e 2021 com informações sobre o acidente, as vítimas, os veículos e o local da ocorrência. Em relação às vítimas, é disponibilizada a ocorrência de ferimentos ou não. Partindo das informações fornecidas pela BHTrans e demais fontes, é possível utilizar modelos de Aprendizado de Máquina (Machine Learning) para prever a ocorrência de ferimentos nas vítimas de acidentes de trânsito em Belo Horizonte baseado em informações sobre as condições do acidente, da vítima e do local de ocorrência. Devido à quantidade de informações, para a análise foram utilizadas as informações de 2021, sendo as mais recentes divulgadas.

1.3. Objetivos

O objetivo do presente trabalho é criar modelos de aprendizado de máquina para prever a ocorrência de ferimentos em vítimas envolvidas em acidentes de trânsito em Belo Horizonte baseado em informações sobre as vítimas, o local e as características do acidente.

2. Coleta de Dados

Os dados utilizados foram coletados no portal de dados abertos da prefeitura de Belo Horizonte que disponibiliza informações sobre os acidentes de trânsito com vítimas ocorridos entre 2011 e 2021. São fornecidos dados sobre as ocorrências, os veículos envolvidos, o local do acidente e as vítimas. Essas informações são divulgadas em quatro conjuntos de dados separados que podem ser relacionados pelo número do boletim de ocorrência:

- si-bol-2021: dataset com os acidentes com vítimas ocorridos em Belo Horizonte em 2021;
- si-veic-2021: dataset com todos os veículos envolvidos nos acidentes com vítima em Belo Horizonte em 2021;
- si-log-2021: dataset com todos os logradouros envolvidos nos acidentes com vítimas em Belo Horizonte em 2021;
- si_env-2021: dataset com todas as vítimas envolvidas nos acidentes com vítimas em Belo Horizonte em 2021.

Também é disponibilizado um dicionário de variáveis para cada dataset. Para este trabalho foram considerados os dados de 2021 por serem os mais recentes disponíveis.

Os dados foram coletados no formato CSV no dia 27 de julho de 2023 pelo link: https://dados.pbh.gov.br/organization/d2359389-8980-4503-a502-4b6e76b7658b?tags=Acidentes.

O dataset com informações sobre a ocorrência possui 23 colunas conforme a Tabela 1 abaixo.

Tabela 1 – Campos do dataset de ocorrências em 2021 (si-bol-2021)

Nome da coluna	**Descrição**	**Tipo da variável**
NUMERO_BOLETIM	Número do boletim de ocorrência.	Texto
DATA HORA_BOLETIM	Data e hora do acidente registradas no boletim de ocorrência.	Data/hora
DATA_INCLUSAO	Data e hora do registro do boletim de ocorrência.	Data/hora
TIPO_ACIDENTE	Código do tipo de acidente.	Texto
DESC_TIPO_ACIDENTE	Descrição do tipo de acidente.	Texto
COD_TEMPO	Código da condição climática no momento do acidente.	Numérico
DESC_TEMPO	Descrição da condição climática no momento do acidente.	Texto
COD_PAVIMENTO	Código do pavimento do logradouro.	Numérico
PAVIMENTO	Descrição do pavimento do logradouro.	Texto
COD_REGIONAL	Número da regional de Belo Horizonte em que ocorreu o acidente.	Numérico
DESC_REGIONAL	Descrição da regional de Belo Horizonte em que ocorreu o acidente.	Texto
ORIGEM_BOLETIM	Origem do boletim de ocorrência.	Texto
LOCAL_SINALIZADO	Indica se o local do acidente é sinalizado.	Texto
VELOCIDADE_PERMITIDA	Velocidade máxima permitida para o local do acidente.	Numérico
COORDENADA_X	Coordenada x do endereço referenciado no boletim de ocorrência.	Numérico

Nome da coluna	Descrição	Tipo da variável
COORDENADA_Y	Coordenada y do endereço referenciado no boletim de ocorrência.	Numérico
HORA_INFORMADA	Indicador de hora do acidente informada.	Texto
INDICADOR_FATALIDADE	Indicador de vítima fatal no acidente.	Texto
VALOR_UPS	Código da severidade do acidente.	Numérico
DESCRIÇÃO_UPS	Descrição da severidade do acidente.	Texto
DATA_ALTERACAO_SMSA	Data de alteração similar.	Data/hora
VALOR_UPS_ANTIGA	Código da severidade do acidente antigo.	Numérico
DESCRIÇÃO_UPS_ANTIGA	Descrição da severidade do acidente antigo.	Texto

O dataset com informações sobre os veículos envolvidos possui 11 colunas conforme a Tabela 2 abaixo.

Tabela 2 – Campos do dataset de veículos envolvidos nos acidentes em 2021 (si-veic-2021)

Nome da coluna	Descrição	Tipo da variável
Nº_boletim	Número do boletim de ocorrência.	Texto
data_hora_boletim	Data e hora do acidente registradas no boletim de ocorrência.	Data/hora
seq_veic	Número sequencial do veículo relacionado ao boletim de ocorrência.	Numérico
cod_categ	Código da categoria do veículo.	Numérico

Nome da coluna	Descrição	Tipo da variável
descricao_categoria	Descrição da categoria do veículo.	Texto
cod_especie	Código da espécie do veículo.	Numérico
descricao_especie	Descrição da espécie do veículo.	Texto
cod_situacao	Código da situação do veículo.	Numérico
desc_situacao	Descrição da situação do veículo.	Texto
tipo_socorro	Código do tipo de socorro prestado para o acidente.	Numérico
desc_tipo_socorro	Descrição do tipo de socorro prestado para o acidente.	Texto

O dataset com informações sobre o logradouro dos acidentes possui 16 colunas conforme a Tabela 3 abaixo.

Tabela 3 – Campos do dataset de logradouros dos acidentes em 2021 (si-log-2021)

Nome da coluna	Descrição	Tipo da variável
Nº_boletim	Número do boletim de ocorrência.	Texto
data_boletim	Data e hora do acidente registradas no boletim de ocorrência.	Data/hora
Nº_municipio	Número do município do acidente.	Numérico
nome_municipio	Nome do município do acidente.	Texto
seq_logradouros	Número sequencial do logradouro relacionado ao boletim de ocorrência.	Numérico
Nº_logradouro	Número do logradouro que compõe o endereço.	Numérico
tipo_logradouro	Tipo do logradouro.	Texto
nome_logradouro	Nome do logradouro.	Texto

Nome da coluna	Descrição	Tipo da variável
tipo_logradouro_anterior	Tipo do logradouro atribuído ao logradouro em momento anterior ao atual.	Texto
nome_logradoro_anterior	Nome do logradouro atribuído ao logradouro em momento anterior ao atual.	Texto
Nº_bairro	Código do bairro do acidente.	Numérico
nome_bairro	Nome do bairro do acidente.	Texto
tipo_bairro	Código do tipo do bairro do acidente.	Texto
descricao_tipo_bairro	Descrição do tipo de bairro do acidente.	Texto
Nº_imovel	Número do imóvel.	Numérico
Nº_imovel_proximo	Número do imóvel mais próximo.	Numérico

O dataset com informações sobre as vítimas dos acidentes possui 16 colunas conforme a Tabela 4 abaixo.

Tabela 4 – Campos do dataset de vítimas envolvidas nos acidentes em 2021 (si_env-2021)

Nome da coluna	Descrição	Tipo da variável
Nº_boletim	Número do boletim de ocorrência.	Texto
data_hora_boletim	Data e hora do acidente registradas no boletim de ocorrência.	Data/hora
Nº_envolvido	Número sequencial da vítima relacionada ao boletim de ocorrência.	Numérico
condutor	Indica se o envolvido é condutor do veículo.	Texto
cod_severidade	Código da severidade do acidente.	Numérico

Nome da coluna	Descrição	Tipo da variável
desc_severidade	Descrição da severidade do acidente.	Texto
sexo	Sexo do envolvido.	Texto
cinto_seguranca	Indica se o envolvido utilizava cinto de segurança.	Texto
Embreagues	Indica se o envolvido estava embriagado.	Texto
Idade	Idade do envolvido.	Numérico
nascimento	Data de nascimento do envolvido.	Data
categoria_habilitacao	Categoria de habilitação do envolvido, se condutor.	Texto
descricao_habilitacao	Descrição da categoria de habilitação do envolvido, se condutor.	Texto
declaracao_obito	Indica se houve óbito ou não.	Numérico
cod_severidade_antiga	Código da severidade antiga.	Numérico
especie_veiculo	Descrição da espécie do veículo.	Texto
pedestre	Indica se o envolvido é pedestre.	Texto
passageiro	Indica se o envolvido é passageiro.	Texto

Além dos dados relacionados aos acidentes, também foram utilizados datasets com a relação de bairros e regionais de Belo Horizonte (para enriquecimento dos dados ausentes das regionais no dataset das ocorrências) e com a relação de feriados e/ou recessos de 2021.

A relação de bairros e regionais de Belo Horizonte foi obtida pelo link: https://prefeitura.pbh.gov.br/sites/default/files/estrutura-de-governo/cultura/2019/COMUC/Rela%C3%A7%C3%A3o%20de%20bairro%2C%20regional%20e%20territ%C3%B3rios.pdf.

O dataset com a relação de feriados e/ou recessos de 2021 foi criado com as colunas “Data” e “Dia”, sendo a primeira coluna cada dia do ano e a segunda, o tipo de dia com as categorias “Dia útil”, “Fim de semana” e “Feriado/Recesso”. Além dos

feriados oficiais de Belo Horizonte, os períodos já conhecidos de recesso (Carnaval, Semana Santa, véspera de Natal e véspera de Ano Novo) também foram classificados como “Feriado/Recesso”.

3. Processamento/Tratamento de Dados

Todas as análises foram realizadas com o auxílio do Jupyter Notebook em Python 3 e para o tratamento e análise descritiva dos dados foram utilizadas as bibliotecas “pandas”, “NumPy” e “matplotlib”.

A leitura dos arquivos sobre os acidentes foi realizada com a função “read_csv()” da biblioteca “pandas” com separador “;” e encoding “cp1252”.

Os arquivos com a relação de bairros e regionais de Belo Horizonte e com a relação dos feriados e/ou recessos de 2021 foram lidos com a função “read_excel()” também da biblioteca “pandas”. O código utilizado é apresentado na Figura 1 abaixo.

```
# Lendo os dados dos acidentes
ocorrencias = pd.read_csv("si-bol-2021.csv",sep=";",encoding='cp1252')
veiculos = pd.read_csv("si-veic-2021.csv",sep=";",encoding='cp1252')
logradouro = pd.read_csv("si-log-2021.csv",sep=";",encoding='cp1252')
vitimas = pd.read_csv("si_env-2021.csv",sep=";",encoding='cp1252')
# Lendo a relação de bairros e regionais
regionais = pd.read_excel("Relação de bairro, regional e territórios.xlsx")
# Lendo a relação de feriados de 2021
feriados = pd.read_excel("Feriados.xlsx",sheet_name='Plan2')
```

Figura 1 – Código utilizado para a leitura dos arquivos

As figuras 2 a 7 apresentam as primeiras linhas e colunas dos arquivos após a leitura.

	NUMERO_BOLETIM	DATA HORA_BOLETIM	DATA_INCLUSAO	TIPO_ACIDENTE	DESC_TIPO_ACIDENTE	COD_TEMPO	DESC_TEMPO	COD_PAVIMENTO
0	2021-008886628-002	20/02/2021 10:01	20/02/2021 11:10	H01002	ABALROAMENTO COM VITIMA ...	1	BOM	1
1	2021-008888878-001	20/02/2021 10:25	20/02/2021 11:30	H09002	COLISAO DE VEICULOS COM VITIMA ...	1	BOM	1
2	2021-008891464-001	20/02/2021 11:22	20/02/2021 11:55	H04000	QUEDA DE PESSOA DE VEICULO ...	1	BOM	1
3	2021-008891884-001	19/02/2021 23:00	20/02/2021 11:59	H08002	CHOQUE MECANICO COM VITIMA ...	0	NAO INFORMADO	0
4	2021-008892064-001	20/02/2021 11:22	20/02/2021 12:02	H01002	ABALROAMENTO COM VITIMA ...	1	BOM	1

Figura 2 – Visualização do arquivo “si-bol-2021” atribuído ao dataframe “ocorrencias"

	N°_boletim	data_hora_boletim	seq_veic	cod_categ	descricao_categoria	cod_especie	descricao_especie	cod_situacao	desc_situacao	tipo_socorro
0	2021-014038208-001	21/03/2021 12:29	1	3	PARTICULAR	6	AUTOMOVEL	1	EM MOVIMENTO	6
1	2021-014038208-001	21/03/2021 12:29	2	3	PARTICULAR	4	MOTOCICLETA	1	EM MOVIMENTO	5
2	2021-014050858-001	21/03/2021 14:03	1	3	PARTICULAR	14	CARROCA	1	EM MOVIMENTO	5
3	2021-014050858-001	21/03/2021 14:03	2	3	PARTICULAR	6	AUTOMOVEL	1	EM MOVIMENTO	6
4	2021-014056225-001	21/03/2021 15:18	2	3	PARTICULAR	6	AUTOMOVEL	1	EM MOVIMENTO	6

Figura 3 – Visualização do arquivo “si-veic-2021” atribuído ao dataframe “veiculos"

	N°_boletim	data_boletim	N°_municipio	nome_municipio	seq_logradouros	N°_logradouro	tipo_logradouro	nome_logradouro	tipo_logradouro_anterior
0	2021-043343627-001	07/09/2021 23:16	1	BELO HORIZONTE	1	117712	RUA	JOAO ARANTES ...	RUA
1	2021-006866782-001	09/02/2021 20:26	1	BELO HORIZONTE	1	117712	RUA	JOAO ARANTES ...	RUA
2	2021-010570968-001	01/03/2021 18:13	1	BELO HORIZONTE	2	116776	RUA	CARLOS DRUMOND DE ANDRADE ...	NI
3	2021-034391637-001	17/07/2021 14:00	1	BELO HORIZONTE	2	116911	PCA	CARTUNISTA HENFIL ...	NI
4	2021-018223712-001	10/04/2021 10:50	1	BELO HORIZONTE	1	116952	RUA	DO PASSARO PRETO ...	RUA

Figura 4 – Visualização do arquivo “si-log-2021” atribuído ao dataframe “logradouro”

	N°_boletim	data_hora_boletim	N°_envolvido	condutor	cod_severidade	desc_severidade	sexo	cinto_seguranca	Embreagues	Idade	nascimento
0	2021-014038208-001	21/03/2021 12:29	1	S	3	SEM FERIMENTOS	M	SIM	NÃO	37	24/04/1983
1	2021-014038208-001	21/03/2021 12:29	2	S	1	NAO FATAL	M	SIM	NÃO	37	21/07/1983
2	2021-014050858-001	21/03/2021 14:03	1	S	1	NAO FATAL	M	SIM	NÃO	30	09/12/1990
3	2021-014050858-001	21/03/2021 14:03	2	N	1	NAO FATAL	M	SIM	NÃO	27	29/06/1993
4	2021-014050858-001	21/03/2021 14:03	6	S	3	SEM FERIMENTOS	M	NÃO	NÃO INFORMADO	0	00/00/0000

Figura 5 – Visualização do arquivo “si_env-2021” atribuído ao dataframe “vitimas”

	BAIRRO	REGIONAL
0	AARÃO REIS	NORTE
1	ACABA MUNDO	CENTRO-SUL
2	ACAIACA	NORDESTE
3	ADEMAR MALDONADO	BARREIRO
4	AEROPORTO	PAMPULHA

Figura 6 – Visualização do arquivo “Relação de bairro, regional e territórios” atribuído ao dataframe “regionais”

	Data	Dia
0	2021-01-01	Feriado/Recesso
1	2021-01-02	Fim de semana
2	2021-01-03	Fim de semana
3	2021-01-04	Dia útil
4	2021-01-05	Dia útil

Figura 7 – Visualização do arquivo “Feriados” atribuído ao dataframe “feriados”

Após a visualização dos arquivos, verificou-se que os bairros de Belo Horizonte possuíam acentos e caracteres especiais, ao contrário dos dados do dataset “logradouro”. Para possibilitar a correspondência entre as duas informações, foram retirados os acentos e caracteres especiais com a função “str.replace()”.

Todas as variáveis dos datasets dos acidentes foram analisadas individualmente para verificar a presença e a quantidade de dados ausentes, além das categorias de resposta e viabilidade de inclusão no trabalho.

O dataset de ocorrências conta com 11.122 linhas em que cada linha corresponde a um boletim de ocorrência e não há dados duplicados. As variáveis “TIPO_ACIDENTE”, “COD_TEMPO”, “COD_PAVIMENTO”, “COD_REGIONAL”, “VALOR_UPS” e “VALOR_UPS_ANTIGA” foram desconsideradas por serem apenas codificações de outras variáveis. As variáveis “DATA_INCLUSAO”,

“COORDENADA_X” e “COORDENADA_Y” também foram desconsideradas por não serem relevantes ao estudo.

Não foram encontrados valores ausentes como NA em nenhumas das colunas, porém as colunas “DESC_TEMPO” e “PAVIMENTO” apresentaram mais de 53% de respostas como “não informado”, a variável “REGIONAL” apresentou 5,5% de respostas em branco, a variável “LOCAL_SINALIZADO” apresentou todos os valores como “não” e a variável “HORA_INFORMADA” apresentou todos os valores como “sim”. As variáveis “DESCRIÇÃO_UPS” e “DESCRIÇÃO_UPS_ANTIGA” apresentaram todos os valores como “não informado” e a variável “DATA_ALTERACAO_SMSA” apresentou todas as respostas como “00/00/0000”. A variável “VELOCIDADE_PERMITIDA” apresentou 96,1% dos valores como 0, o que se entende ser “não informado”.

A variável “ORIGEM_BOLETIM” apresentou 95,4% das respostas como “polícia militar” e a variável “INDICADOR_FATALIDADE” apresentou 99% dos valores como “não”, indicando que a mortalidade nos acidentes foi baixa.

Devido à quantidade de dados ausentes e/ou falta de variabilidade, as variáveis “DESC_TEMPO”, “PAVIMENTO”, “ORIGEM_BOLETIM”, “LOCAL_SINALIZADO”, “VELOCIDADE_PERMITIDA”, “HORA_INFORMADA”, “DESCRIÇÃO_UPS”, “DESCRIÇÃO_UPS_ANTIGA”, “DATA_ALTERACAO_SMSA” foram excluídas. A variável “INDICADOR_FATALIDADE” também foi excluída já que será utilizada a informação de severidade das vítimas disponível no dataset dos envolvidos no acidente, assim como a variável “DATA HORA_BOLETIM”.

A variável “DESC_TIPO_ACIDENTE” não apresentou valores ausentes ou respostas quase constantes, mas algumas de suas categorias indicam o desfecho do acidente, como “ATROPELAMENTO DE PESSOA SEM VITIMA FATAL” e “ATROPELAMENTO DE PESSOA COM VITIMA FATAL”, não fazendo sentido incluí-la como variável explicativa nos modelos.

O dataset de veículos possui 20.506 linhas referentes a 11.121 boletins de ocorrência. As variáveis “cod_categ”, “cod_especie”, “cod_situacao”, “tipo_socorro” foram desconsideradas por serem codificações das outras variáveis. Nenhuma das variáveis apresentou dados ausentes como NA. A variável “descricao_categoria”

apresentou 86,9% das respostas como “PARTICULAR”, a variável “desc_situacao” apresentou 90,9% das respostas como “EM MOVIMENTO” e a variável “desc_tipo_socorro” apresentou 48,5% das respostas como “NÃO SE APLICA” e 7,4% como “NÃO INFORMADO”, portanto foram excluídas pelo excesso de dados ausentes ou falta de variabilidade. As variáveis “data_hora_boletim” e “descricao_especie” também foram excluídas pois serão utilizadas as informações do dataset de vítimas envolvidas.

O dataset de logradouros apresenta 14.437 linhas referentes a 11.116 boletins de ocorrência e não há dados ausentes como NA em nenhuma das variáveis. As variáveis “Nº_bairro” e “tipo_bairro” foram desconsideradas por serem apenas codificação de outras variáveis. As variáveis “Nº_municipio” e “nome_municipio” foram excluídas dado que todos os acidentes foram em Belo Horizonte e as variáveis “Nº_logradouro”, “nome_logradouro”, “Nº_imovel” e “Nº_imovel_proximo” foram excluídas pois o endereço completo não será utilizado nas análises. As variáveis “tipo_logradouro_anterior” e “nome_logradoro_anterior” também foram desconsideradas por não serem relevantes ao estudo. A variável “descricao_tipo_bairro” apresentou 94,5% das respostas como “BAIRRO”, sendo também desconsiderada pela falta de variabilidade. A variável “nome_bairro” não será utilizada diretamente devido ao grande número de respostas possíveis, mas será utilizada no enriquecimento dos dados ausentes na variável “REGIONAL” do dataset de ocorrências. A variável “data_hora_boletim” também foi excluída pois será utilizada a informação do dataset de vítimas envolvidas.

O dataset de vítimas envolvidas nos acidentes conta com 23.736 linhas referentes a 11.122 boletins de ocorrência. Apenas as variáveis “Pedestre” e “Passageiro” apresentaram dados ausentes como NA, sendo 2.260 (9,5%) e 2.165 (9,1%) respectivamente. As variáveis “cod_severidade”, “categoria_habilitacao” foram desconsideradas por serem apenas codificação de outras variáveis. A variável “descricao_habilitacao” foi excluída por possuir quase 33% de respostas em branco ou “NÃO INFORMADO” e se referir apenas às vítimas condutoras do veículo. As variáveis “declaracao_obito” e “cod_severidade_antiga” apresentaram todas as respostas como “0” (sem informação) e também foram excluídas. A variável “especie_veiculo” apresentou 6,1% de respostas em branco ou como “não informado”. A variável “sexo” apresentou 3,3% das respostas como “0” (não informado) e a

variável “Embreagues” apresentou 7,1% das respostas como “NÃO INFORMADO”. A variável resposta do estudo, “desc_severidade” apresentou 1,9% de respostas como “NÃO INFORMADO”.

A variável “Idade” não apresentou dados ausentes como NA, mas a idade mais frequente foi 0, levantando a suspeita de que dados ausentes possam ter sido codificados como 0. Analisando a variável “nascimento” verificou-se que há 2.415 vítimas com resposta “00/00/0000”, indicando falta de informação.

Após a exclusão das colunas que não serão utilizadas, as colunas resultantes foram:

- Dataset de ocorrências: número do boletim e regional;
- Dataset de veículos: número do boletim e número do veículo na ocorrência (“seq_veic”);
- Dataset de logradouros: número do boletim, número do logradouro na ocorrência ("seq_logradouros”), tipo do logradouro e bairro;
- Dataset de vítimas: número do boletim, data e hora da ocorrência, número do envolvido na ocorrência (“Nº_envolvido”), sexo, idade, data de nascimento, cinto de segurança, embriaguez, condutor, passageiro, pedestre, espécie do veículo e severidade.

As variáveis foram renomeadas para retirar espaços em branco ao início e ao final do nome, corrigir erros de ortografia e facilitar a utilização posterior.

Antes do tratamento dos dados ausentes, foram criadas algumas variáveis a partir da contagem do número registros por boletim de ocorrência (com o auxílio das funções “groupby()” e “size()”), como demonstrado nas figuras 8 a 10 a seguir:

- “Num envolvidos”: número de vítimas envolvidas em cada acidente.
- “Num veiculos”: número de veículos envolvidos em cada acidente.
- “Num logradouros”: número de logradouros envolvidos em cada acidente

```
# criação do dataframe com o número de envolvidos por acidente
envolv_aux = vitimas[['Boletim']].groupby(by=['Boletim'],group_keys=False,as_index=False).size()
```

```
print(vitimas['Boletim'].nunique())
print(envolv_aux.shape)
```

```
11122
(11122, 2)
```

```
envolv_aux.columns = ['Boletim','Num envolvidos']
```

```
# colocando o número de envolvidos na base de vitimas
vitimas = vitimas.merge(envolv_aux,left_on='Boletim',right_on='Boletim',how='left')
```

Figura 8 – Criação da coluna “Num envolvidos” e inclusão no dataset “vitimas”

```
# criação do dataframe com o número de veículos por acidente
numveic_aux = veiculos[['Boletim']].groupby(by=['Boletim'],group_keys=False,as_index=False).size()
```

```
print(veiculos['Boletim'].nunique())
print(numveic_aux.shape)
```

```
11121
(11121, 2)
```

```
numveic_aux.columns = ['Boletim','Num veiculos']
```

```
# colocando o número de veículos envolvidos na base de vitimas
vitimas = vitimas.merge(numveic_aux,left_on='Boletim',right_on='Boletim',how='left')
```

Figura 9 – Criação da coluna “Num veiculos” e inclusão no dataset “vitimas”

```
# criação do dataframe com o número de logradouros por acidente
numlog_aux = logradouro[['Boletim']].groupby(by=['Boletim'],group_keys=False,as_index=False).size()
```

```
print(logradouro['Boletim'].nunique())
print(numlog_aux.shape)
```

```
11116
(11116, 2)
```

```
numlog_aux.columns = ['Boletim','Num logradouros']
```

```
# colocando o número de logradouros envolvidos na base de vitimas
vitimas = vitimas.merge(numlog_aux,left_on='Boletim',right_on='Boletim',how='left')
```

Figura 10 - Criação da coluna “Num logradouros” e inclusão no dataset “vitimas”

Além destas, também foram criadas variáveis dummies para a ocorrência de logradouros e veículos dos tipos mais frequentes, apresentado nas figuras 11 e 12. Para os logradouros, foram utilizadas as funções “groupby()” e “size()” para calcular quantos logradouros de cada tipo havia em cada boletim de ocorrência do dataset “logradouro”. Então, o dataframe no formato long foi transformado para o formato wide (com a função “pivot()”) em que cada coluna corresponde à quantidade de determinado tipo de logradouro em cada ocorrência. Foi retirada a terceira dimensão do dataframe (que é gerada automaticamente pela função “pivot()”), foram retirados os espaços extras no nome das colunas e os valores automaticamente criados como NA foram modificados para 0. Foram considerados os logradouros mais frequentes (rua, avenida e rodovia) e os demais, excluídos. Foi criada também uma coluna “RUA AVE” com a soma das colunas “RUA” e “AVE” para representar os casos em que o acidente ocorreu em uma rua ou em uma avenida e posteriormente será avaliado se esta coluna apresenta relação mais forte com a variável resposta que as colunas individuais.

```
# criação do dataframe com o número de logradouros de cada tipo
log_aux = logradouro[['Boletim','Tipo logradouro']].   \
    groupby(by=['Boletim','Tipo logradouro'],group_keys=False,as_index=False).size()
# transformando a tabela do formato long para wide
log_aux = pd.pivot(log_aux,index=['Boletim'],columns=['Tipo logradouro'],values=['size'])
# retirando o multiindex
log_aux.columns = log_aux.columns.droplevel(0)
# retirando espaços em branco do nome das colunas
log_aux.columns = log_aux.columns.str.strip()
# substituindo os valores 'NaN' por 0
log_aux.fillna(0,inplace=True)
# colocando o nº do boletim como coluna e não como índice
log_aux.reset_index(inplace=True)
```

```
# criando nova coluna com rua e avenida
log_aux['RUA AVE'] = log_aux['RUA'] + log_aux['AVE']
```

```
log_aux.drop(columns=['ACS', 'ALA', 'BEC', 'EST', 'PCA', 'TRE', 'TRI', 'TUN', 'VDT', 'VIA'],inplace=True)
```

```
# colocando os tipos de logradouro na base de vitimas
vitimas = vitimas.merge(log_aux,left_on='Boletim',right_on='Boletim',how='left')
```

Figura 11 – Criação das variáveis dummies para os tipos de logradouro mais frequentes e inclusão no dataset “vitimas”

Para os veículos foi utilizada a coluna “espécie veículo” do dataset de vítimas e foram criadas colunas dummies para os veículos mais frequentes. Como a frequência de caminhonetes e camionetas é baixa e há semelhança entre os veículos,

as frequências destas foram somadas na coluna “caminhonete”. As colunas criadas foram “automóvel”, “motocicleta”, “ônibus”, “bicicleta”, “caminhonete” e “caminhão”.

```
# criando colunas para os tipos de veículos mais frequentes
vitimas['Especie veiculo'] = vitimas['Especie veiculo'].str.strip()

vitimas['AUTOMOVEL'] = [1 if x=='AUTOMOVEL' else 0 for x in vitimas['Especie veiculo']]
vitimas['MOTOCICLETA'] = [1 if x=='MOTOCICLETA' else 0 for x in vitimas['Especie veiculo']]
vitimas['BICICLETA'] = [1 if x=='BICICLETA' else 0 for x in vitimas['Especie veiculo']]
vitimas['CAMINHAO'] = [1 if x=='CAMINHAO' else 0 for x in vitimas['Especie veiculo']]
vitimas['ONIBUS'] = [1 if x=='ONIBUS' else 0 for x in vitimas['Especie veiculo']]
vitimas['CAMINHONETE'] = [1 if x in ['CAMINHONETE','CAMIONETA'] else 0 for x in vitimas['Especie veiculo']]
```

Figura 12 – Criação de variáveis dummies no dataset “vitimas” para os tipos de veículos mais frequentes

As variáveis com o número de envolvidos, veículos e logradouros e com a frequência dos tipos de logradouros foram incluídas no dataset “vitimas” através da função “merge()” com o argumento “how” igual a “left”.

As variáveis referentes aos tipos de logradouros foram transformadas em dummies recebendo valor 1 caso o número de logradouros de cada tipo fosse maior ou igual a 1 e 0, caso contrário, como demonstrado na figura 13.

```
# transformando variáveis de tipo de logradouro em 0 ou 1
for i in ['AVE', 'ROD', 'RUA', 'RUA AVE']:
    vitimas[i] = [0 if x==0 else 1 for x in vitimas[i]]
```

Figura 13 – Transformação das colunas com a frequência dos tipos em logradouro em dummies

Devido ao grande número de acidentes e vítimas e baixa porcentagem de dados ausentes nas variáveis restantes, optou-se por excluir os registros que não possuíam informações completas. Foram excluídos os registros de vítimas sem informação sobre sexo e severidade. Foi necessário retirar os espaços em branco ao início e ao final das respostas através da função “str.strip()”.

Verificou-se que, dos 830 registros sem informação sobre embriaguez da vítima, 721 eram de vítimas que eram condutoras do veículo. Como a relação entre

condutores embriagados e a frequência e gravidade de acidentes já é conhecida, optou-se por excluir os registros de vítimas sem informação sobre embriaguez.

Após as exclusões anteriores, restaram 114 registros com idade 0 e 791 registros com data de nascimento “00/00/0000”. Dos 114 registros com idade 0, 111 possuíam data de nascimento “00/00/0000” e, portanto, foram excluídos. Os 3 restantes possuíam data de nascimento compatível com a idade informada. Como os demais registros sem informação sobre a data de nascimento possuíam informação sobre a idade, foram mantidos.

Dos demais registros, haviam 570 dados ausentes para a variável “passageiro” e 634 para a variável “pedestre”. Partindo do pressuposto que cada vítima só pode estar em uma das categorias “condutor”, “passageiro” ou “pedestre”, verificou-se que, dos 570 envolvidos sem informação em “passageiro”, 510 eram condutores e 56 eram pedestres. Estes 566 envolvidos tiveram a resposta “não” atribuída à variável “passageiro”. Dos 634 envolvidos sem informação em “pedestre”, 510 eram condutores e 120 eram passageiros, portanto estes 630 tiveram a resposta “não” atribuída à variável “pedestre”. Ao final, apenas 4 registros se mantiveram sem informação sobre os envolvidos serem passageiros ou pedestres e foram excluídos.

Para enriquecer os dados ausentes da variável “regional” do dataset de ocorrências, foi utilizada a coluna “bairro” do dataset “logradouro”. Como cada acidente poderia envolver mais de 1 logradouro, inicialmente os registros foram agrupados por número de boletim de ocorrência e bairro para verificar se haviam acidentes com logradouros em bairros diferentes. Foram utilizadas as funções “groupby()” e “size()”. Como o dataset resultante do agrupamento, denominado “bairros1”, apresentou 11.116 linhas e não há boletins de ocorrência repetidos, conclui-se que não há acidentes envolvendo mais de 1 bairro.

Foi realizado um left join entre o dataset “ocorrências” e o novo dataset “bairros1” com o número do boletim de ocorrência e o bairro. Para tal, foi utilizada a função “merge()” em que a chave entre os datasets foi o número do boletim de ocorrência. Como esperado (já que o dataset “logradouro” possui informações para 11.116 boletins de ocorrência), não foi possível identificar o bairro de 6 ocorrências e estas foram excluídas por também não possuírem informação sobre a regional. Após, foi realizado um left join entre o dataset “ocorrencias” (já com os bairros) e a relação

de bairros e regionais de Belo Horizonte, utilizando o bairro como chave. Quatorze bairros não foram encontrados na relação, referentes a 22 ocorrências. Os bairros não encontrados foram pesquisados no site da prefeitura de Belo Horizonte e sites auxiliares e foi possível identificar a regional de 13 deles. Apenas 1 registro ficou sem informação sobre a regional e foi excluído. A coluna “regional” do dataset de ocorrências foi atualizada para manter as regionais originais do dataset, quando disponível, e completar as informações ausentes com as regionais encontradas. A variável “regional” foi incluída no dataset de vítimas através da função “merge()” e verificou-se 14 registros sem informação sobre a regional, que foram excluídos.

Utilizando a data e a hora da ocorrência, também foram criadas variáveis com o dia da semana em que o acidente ocorreu, o turno do dia e o tipo de dia (dia útil, fim de semana ou feriado/recesso). A data e a hora da ocorrência foram separadas e a variável “turno” foi definida como:

- Madrugada: 00h às 05h59
- Manhã: 06h às 11h59
- Tarde: 12h às 17h59
- Noite: 18h às 23h59

A data foi transformada do tipo string para o tipo data com a função “to_datetime()” da biblioteca “pandas” e o dia da semana foi obtido com o atributo “dt.dayofweek”. Como o dia da semana obtido é no formato numérico, os valores foram substituídos pelo nome do dia da semana.

Para obter o tipo de dia, foi realizado um left join com a data como chave entre o dataset de vítimas e o dataset “Feriados” que possui a classificação de todos os dias do ano em “dia útil”, “fim de semana” e “feriado/recesso”, categorizados conforme descrito na seção 2.

Após a inclusão de todas as variáveis no dataset “vitimas”, foi verificado que apenas a variável “número de veículos” possuía 1 dado ausente (pois o dataset de veículos possui 1 boletim de ocorrência a menos) e o registro foi excluído.

As variáveis resultantes no dataset final foram: “severidade”, “condutor”, “sexo”, “cinto de segurança”, “embriaguez”, “idade”, “pedestre”, “passageiro”, “nº envolvidos”, “nº veículos”, “nº logradouros”, “automóvel”, “bicicleta”, “caminhão”, “caminhonete”,

“motocicleta”, “ônibus”, “avenida”, “rodovia”, “rua”, “rua/avenida”, “regional”, “turno”, “dia da semana” e “tipo de dia”. O dataset possui 21.869 linhas e nenhuma possui dados ausentes.

A variável resposta “severidade” apresenta três categorias de resposta: “sem ferimentos” com 9.294 registros, “não fatal” com 12.468 registros e “fatal” com 107 registros. Como as categorias são muito desbalanceadas e a frequência de acidentes fatais é extremamente baixa (o que poderia impactar em um possível oversampling por ser necessário criar muitos registros sintéticos a partir de uma amostra muito pequena), optou-se por agrupar as categorias “fatal” e “não fatal” para fazer a previsão da ocorrência de ferimentos nas vítimas dos acidentes. As frequências absolutas das novas categorias são apresentadas na Figura 14.

```
vitimas['Severidade'].value_counts()
```

```
COM FERIMENTOS    12575
SEM FERIMENTOS     9294
```

Figura 14 – Frequências absolutas das categorias da variável “severidade”

Das variáveis explicativas, apenas 4 eram numéricas: idade, número de envolvidos, número de veículos e número de logradouros. Pela própria natureza das variáveis, o número de envolvidos, veículos e logradouros são bastante assimétricos, com a maior parte dos dados concentrados nos menores valores e possuindo vários valores com frequência muito baixa, como pode-se perceber nas figuras 15 e 17.

De modo a criar categorias com tamanhos razoáveis e como a partir de 4 envolvidos o número de registros é muito baixo, optou-se por agrupar o número de envolvidos em “1 envolvido”, “2 envolvidos”, “3 envolvidos” e “4 ou mais envolvidos”, com frequências absolutas conforme a Figura 16.

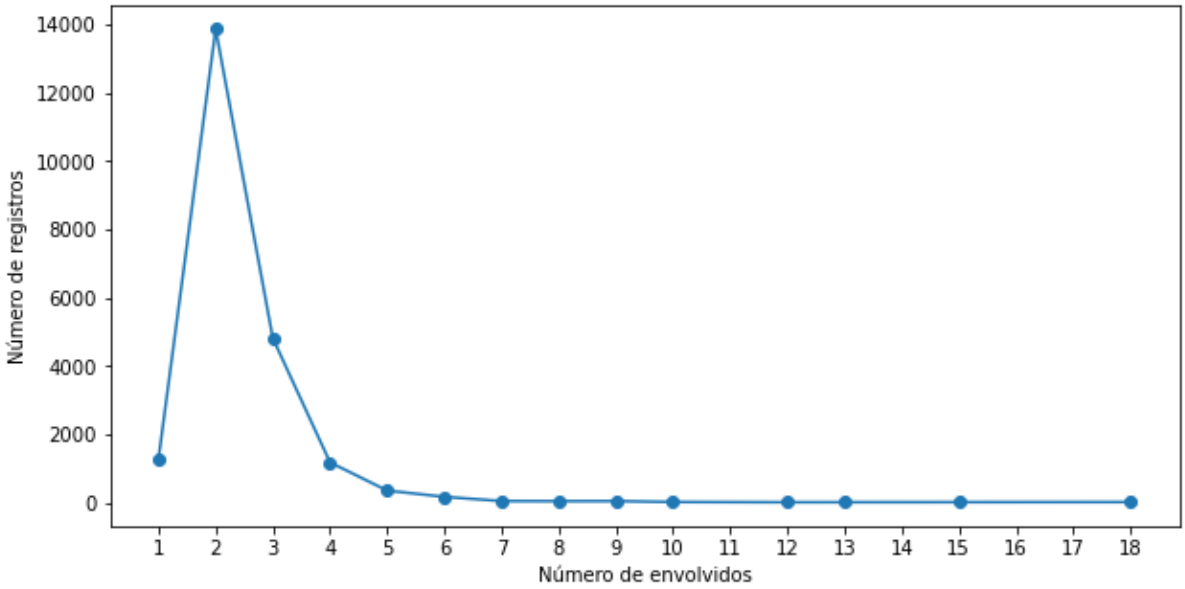

Figura 15 – Número de registros por número de envolvidos

```
vitimas['Num envolvidos cat'].value_counts(sort=False,dropna=False)
```

```
1 envolv            1277
2 envolv           13866
3 envolv            4811
4 envolv ou mais    1915
```

Figura 16 – Frequências absolutas das categorias da variável “Num envolvidos cat”

Em relação ao número de veículos, percebe-se que a partir de 3 veículos as frequências são extremamente baixas, portanto a variável foi agrupada em “1 veículo”, “2 veículos” e “3 veículos ou mais”. As frequências absolutas estão apresentadas na Figura 18.

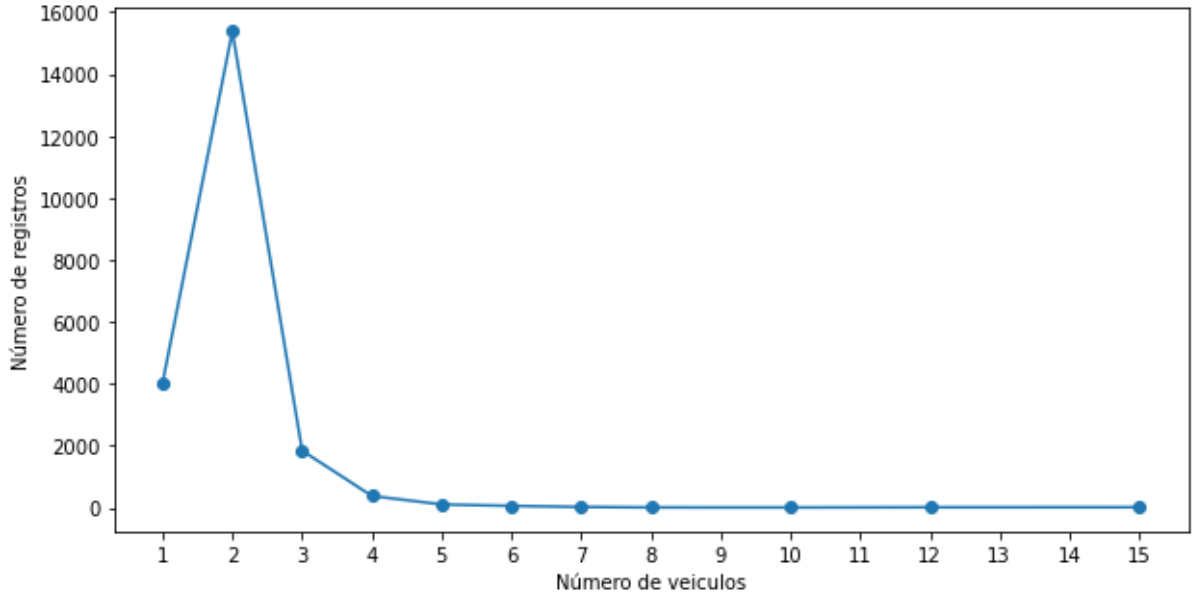

Figura 17 – Número de registros por número de veículos

```
vitimas['Num veiculos cat'].value_counts(sort=False,dropna=False)
```

```
1 veic             4036
2 veic            15378
3 veic ou mais     2455
```

Figura 18 – Frequências absolutas das categorias da variável “Num veiculos cat”

O número de logradouros foi no máximo 3, tendo ocorrido em apenas 4 registros. Neste caso a variável foi agrupada em “1 logradouro” e “2 ou mais logradouros”, conforme a Figura 19.

```
vitimas['Num logradouros cat'].value_counts(sort=False,dropna=False)
```

```
1 logradouro            15141
2 logradouros ou mais    6728
```

Figura 19 – Frequências absolutas das categorias da variável “Num logradouros cat”

Em relação à idade, os envolvidos foram categorizados em crianças, adolescentes, jovens adultos, adultos e idosos conforme a seguinte classificação:

- Criança: 0 a 11 anos
- Adolescente: 12 a 17 anos
- Jovens adultos: 18 a 29 anos
- Adultos: 30 a 59 anos
- Idosos: 60 anos ou mais

Embora as categorias de 0 a 11 anos e de 12 a 17 anos tenham agrupado poucos registros, como pode ser visto na Figura 20, optou-se por mantê-las separadas devido à grande diferença na vulnerabilidade dos envolvidos em cada grupo.

```
vitimas['Faixa etaria'].value_counts(sort=False)
```

```
0 a 11          197
12 a 17         278
18 a 29        7475
30 a 59       12230
60 ou mais     1689
```

Figura 20 – Frequências absolutas das categorias da variável “Faixa etaria”

4. Análise e Exploração dos Dados

O dataset final conta com 21.869 vítimas e 24 variáveis explicativas além da variável alvo, severidade. Das 21.869 vítimas, 12.575 (57,5%) tiveram ferimentos e 9.294 (42,5%) não. A Tabela 5 apresenta a análise descritiva das variáveis em geral e por severidade. A maior parte das vítimas eram do sexo masculino (76,8%), com idade entre 30 e 59 anos (55,9%). As faixas etárias com maiores diferenças entre os grupos foram de 18 a 29 anos, com maior proporção de vítimas com ferimentos, e de 30 a 59 anos, com maior proporção de vítimas sem ferimentos. Vítimas que eram as condutoras dos veículos apresentaram ferimentos com menor frequência que as não condutoras, ao contrário de pedestres e passageiros, que apresentaram ferimentos com maior frequência. Como esperado, houve uma proporção um pouco maior de ferimentos entre as vítimas sem cinto de segurança e embriagadas.

Em relação aos veículos, vítimas que estavam em automóveis sofreram ferimentos com menor frequência, ao contrário das vítimas que estavam em motocicletas.

Já sobre os logradouros, houve pouca diferença com a severidade das vítimas, mas vítimas de acidentes em rodovias apresentaram frequência um pouco maior de ferimentos. Como as variáveis “rua” e “avenida” individuais apresentaram maiores diferenças entre os grupos com e sem ferimentos, optou-se por mantê-las e excluir a variável “rua/avenida”.

A regional nordeste apresentou frequência um pouco maior de vítimas com ferimentos e a regional centro-sul, frequência maior de vítimas sem ferimentos.

Os dias que apresentaram frequência maior de vítimas com ferimentos foram sábado e domingo e os turnos, noite e madrugada. Vítimas de acidentes aos fins de semana ou feriados também apresentaram frequência um pouco maior de ferimentos.

Tabela 5 – Análise descritiva geral e por grupo das variáveis explicativas

	Geral	**Com ferimentos**	**Sem ferimentos**
Sexo			
Feminino	5.073 (23,2)	3.079 (24,5)	1.994 (21,5)
Masculino	16.796 (76,8)	9.496 (75,5)	7.300 (78,5)
Faixa etária			
0 a 11 anos	197 (0,9)	180 (1,4)	17 (0,2)
12 a 17 anos	278 (1,3)	255 (2,0)	23 (0,2)
18 a 29 anos	7.475 (34,2)	5.436 (43,2)	2.039 (21,9)
30 a 59 anos	12.230 (55,9)	6.006 (47,8)	6.224 (67,0)
60 anos ou mais	1.689 (7,7)	698 (5,6)	991 (10,7)
Condutor			
Sim	18.470 (84,5)	9.509 (75,6)	8.961 (96,4)
Não	3.399 (15,5)	3.066 (24,4)	333 (3,6)
Passageiro			
Sim	2.275 (10,4)	1.955 (15,5)	320 (3,4)
Não	19.594 (89,6)	10.620 (84,5)	8.974 (96,6)
Pedestre			
Sim	1.121 (5,1)	1.108 (8,8)	13 (0,1)
Não	20.748 (94,9)	11.467 (91,2)	9.281 (99,9)
Cinto de segurança			
Sim	20.230 (92,5)	10.992 (87,4)	9.238 (99,4)
Não	1.639 (7,5)	1.583 (12,6)	56 (0,6)
Embriaguez			
Sim	411 (1,9)	285 (2,3)	126 (1,4)
Não	21.458 (98,1)	12.290 (97,7)	9.168 (98,6)
Veículos			
Automóvel			
Sim	9.507 (43,5)	2.210 (17,6)	7.297 (78,5)
Não	12.362 (56,5)	10.365 (82,4)	1.997 (21,5)

	Geral	Com ferimentos	Sem ferimentos
Motocicleta			
Sim	8.890 (40,7)	8.248 (65,6)	642 (6,9)
Não	12.979 (59,3)	4.327 (34,4)	8.652 (93,1)
Caminhonete			
Sim	517 (2,4)	101 (0,8)	416 (4,5)
Não	21.352 (97,6)	12.474 (99,2)	8.878 (95,5)
Caminhão			
Sim	317 (1,4)	48 (0,4)	269 (2,9)
Não	21.552 (98,6)	12.527 (99,6)	9.025 (97,1)
Ônibus			
Sim	673 (3,1)	182 (1,4)	491 (5,3)
Não	21.196 (96,9)	12.393 (98,6)	8.803 (94,7)
Bicicleta			
Sim	399 (1,8)	390 (3,1)	9 (0,1)
Não	21.470 (98,2)	12.185 (96,9)	9.285 (99,9)
Logradouros			
Rua			
Sim	11.309 (51,7)	6.456 (51,3)	4.853 (52,2)
Não	10.560 (48,3)	6.119 (48,7)	4.441 (47,8)
Avenida			
Sim	10.338 (47,3)	5.909 (47,0)	4.429 (47,7)
Não	11.531 (52,7)	6.666 (53,0)	4.865 (52,3)
Rua ou avenida			
Sim	19.443 (88,9)	11.149 (88,7)	8.294 (89,2)
Não	2.426 (11,1)	1.426 (11,3)	1.000 (10,8)
Rodovia			
Sim	2.205 (10,1)	1.300 (10,3)	905 (9,7)
Não	19.664 (89,9)	11.275 (89,7)	8.389 (90,3)

	Geral	Com ferimentos	Sem ferimentos
Regional			
Barreiro	1.936 (8,9)	1.133 (9,0)	803 (8,6)
Centro-sul	3.427 (15,7)	1.913 (15,2)	1.514 (16,3)
Leste	1.709 (7,8)	1.000 (8,0)	709 (7,6)
Nordeste	2.383 (10,9)	1.422 (11,3)	961 (10,3)
Noroeste	2.504 (11,4)	1.433 (11,4)	1.071 (11,5)
Norte	1.920 (8,8)	1.129 (9,0)	791 (8,5)
Oeste	2.734 (12,5)	1.554 (12,4)	1.180 (12,7)
Pampulha	3.445 (15,8)	1.940 (15,4)	1.505 (16,2)
Venda Nova	1.811 (8,3)	1.051 (8,4)	760 (8,2)
Dia da semana			
Domingo	2.306 (10,5)	1.419 (11,3)	887 (9,5)
Segunda-feira	3.170 (14,5)	1.787 (14,2)	1.383 (14,9)
Terça-feira	3.141 (14,4)	1.787 (14,2)	1.354 (14,6)
Quarta-feira	3.221 (14,7)	1.810 (14,4)	1.411 (15,2)
Quinta-feira	3.171 (14,5)	1.776 (14,1)	1.395 (15,0)
Sexta-feira	3.851 (17,6)	2.209 (17,6)	1.642 (17,7)
Sábado	3.009 (13,8)	1.787 (14,2)	1.222 (13,1)
Turno			
Madrugada	1.161 (5,3)	800 (6,4)	361 (3,9)
Manhã	6.476 (29,6)	3.610 (28,7)	2.866 (30,8)
Tarde	8.220 (37,6)	4.607 (36,6)	3.613 (38,9)
Noite	6.012 (27,5)	3.558 (28,3)	2.454 (26,4)
Tipo de dia			
Dia útil	15.886 (72,6)	8.971 (71,3)	6.915 (74,4)
Fim de semana	5.011 (22,9)	3.015 (24,0)	1.996 (21,5)
Feriado/recesso	972 (4,4)	589 (4,7)	383 (4,1)
Número de envolvidos			
1 envolvido	1.277 (5,8)	1.277 (10,2)	0 (0,0)
2 envolvidos	13.866 (63,4)	7.875 (62,6)	5.991 (64,5)

	Geral	Com ferimentos	Sem ferimentos
3 envolvidos	4.811 (22,0)	2.516 (20,0)	2.295 (24,7)
4 envolvidos ou mais	1.915 (8,8)	907 (7,2)	1.008 (10,8)
Número de veículos			
1 veículo	4.036 (18,5)	3.077 (24,5)	959 (10,3)
2 veículos	15.378 (70,3)	8.509 (67,7)	6.869 (73,9)
3 veículos ou mais	2.455 (11,2)	989 (7,9)	1.466 (15,8)
Número de logradouros			
1 logradouro	15.141 (69,2)	8.837 (70,3)	6.304 (67,8)
2 logradouros ou mais	6.728 (30,8)	3.738 (29,7)	2.990 (32,2)

As figuras 21 a 34 apresentam a distribuição das variáveis de acordo com a severidade.

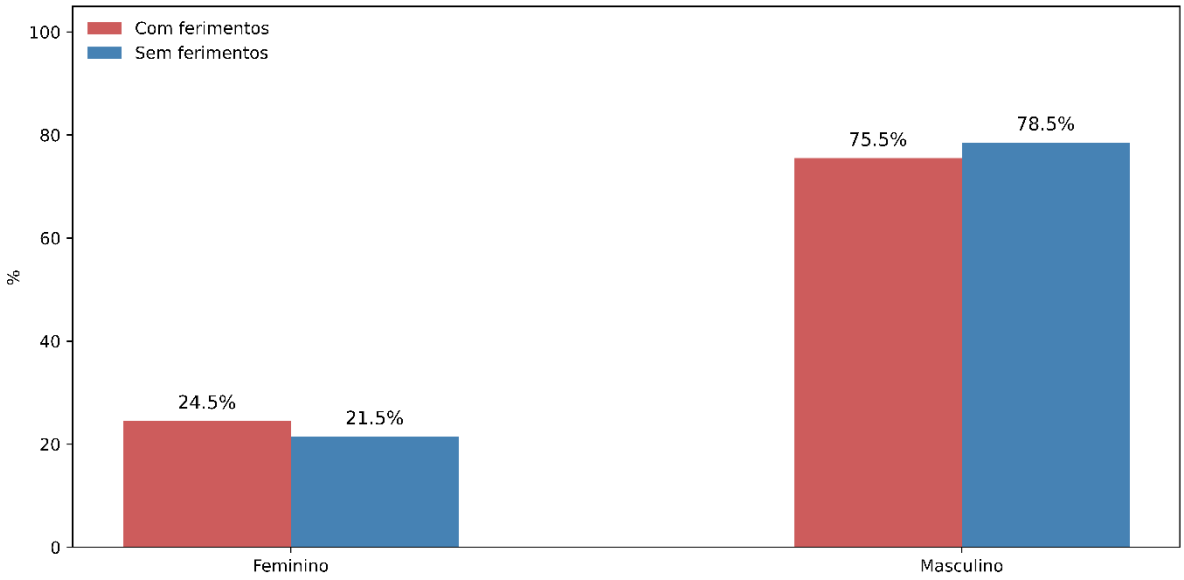

Figura 21 – Distribuição da variável “sexo” entre as vítimas com e sem ferimentos

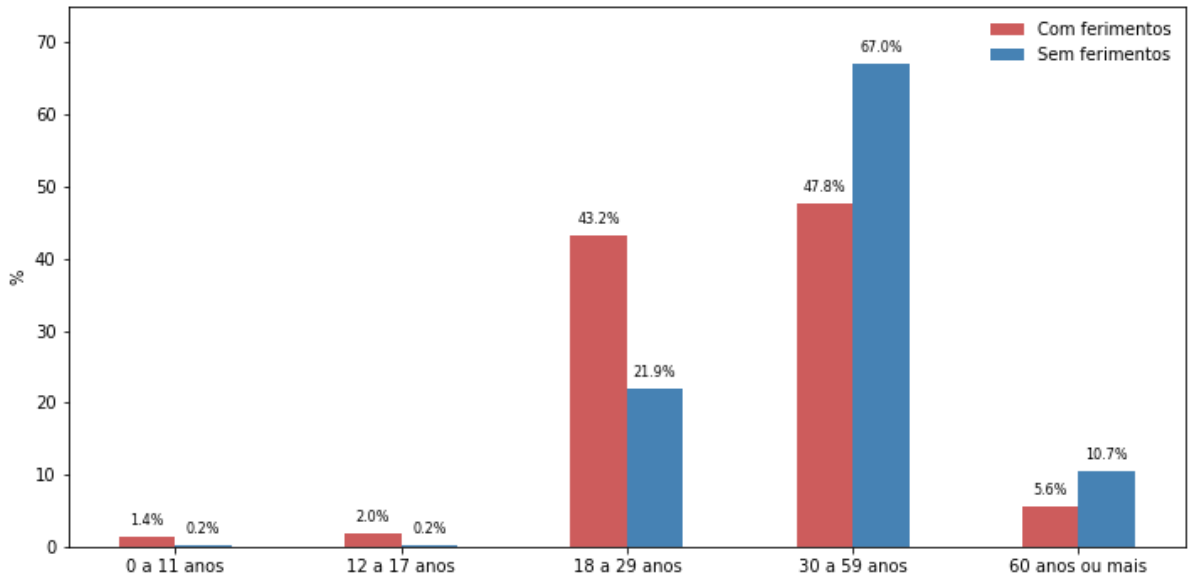

Figura 22 – Distribuição da variável “faixa etária” entre as vítimas com e sem ferimentos

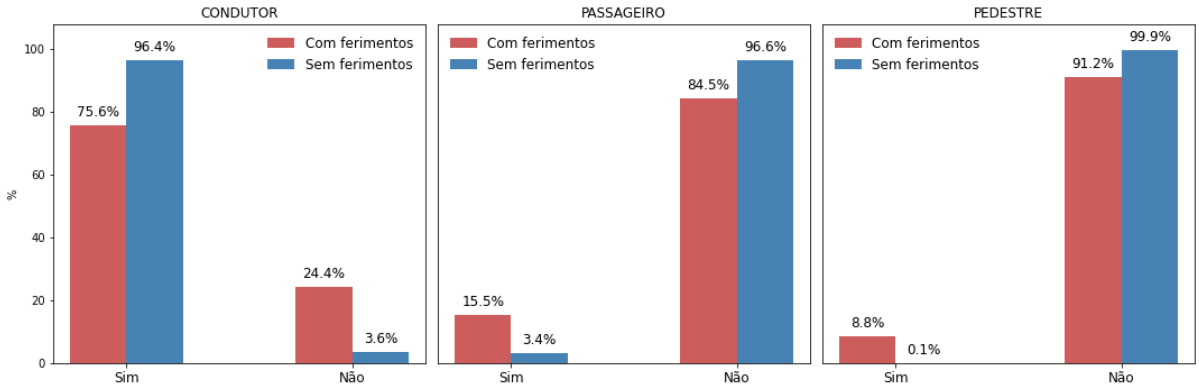

Figura 23 – Distribuição das variáveis “condutor”, “passageiro” e “pedestre” entre as vítimas com e sem ferimentos

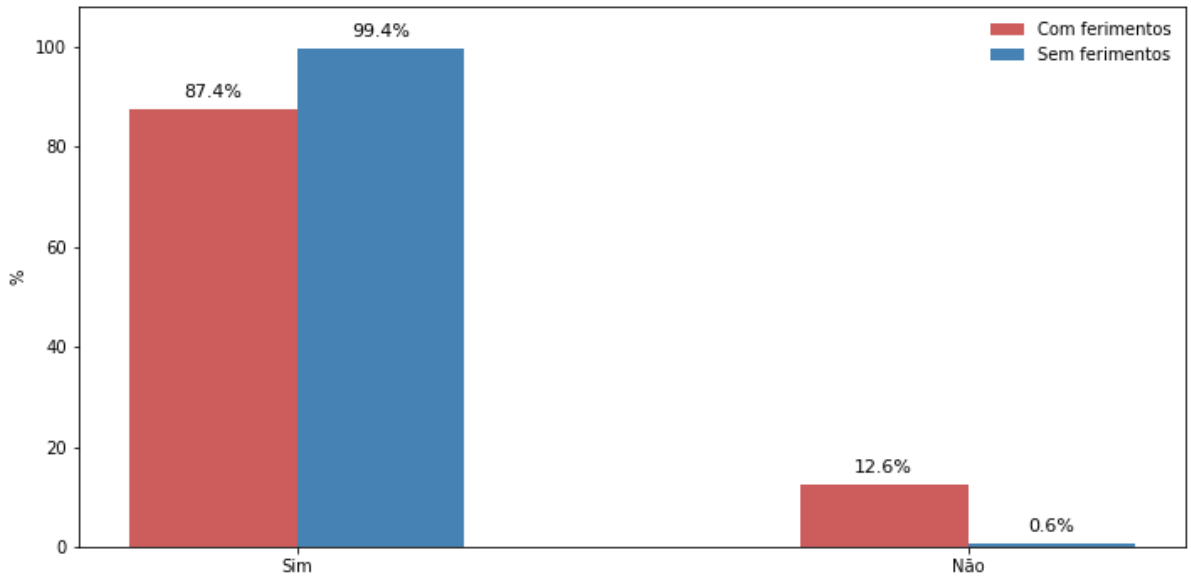

Figura 24 – Distribuição da variável “cinto de segurança” entre as vítimas com e sem ferimentos

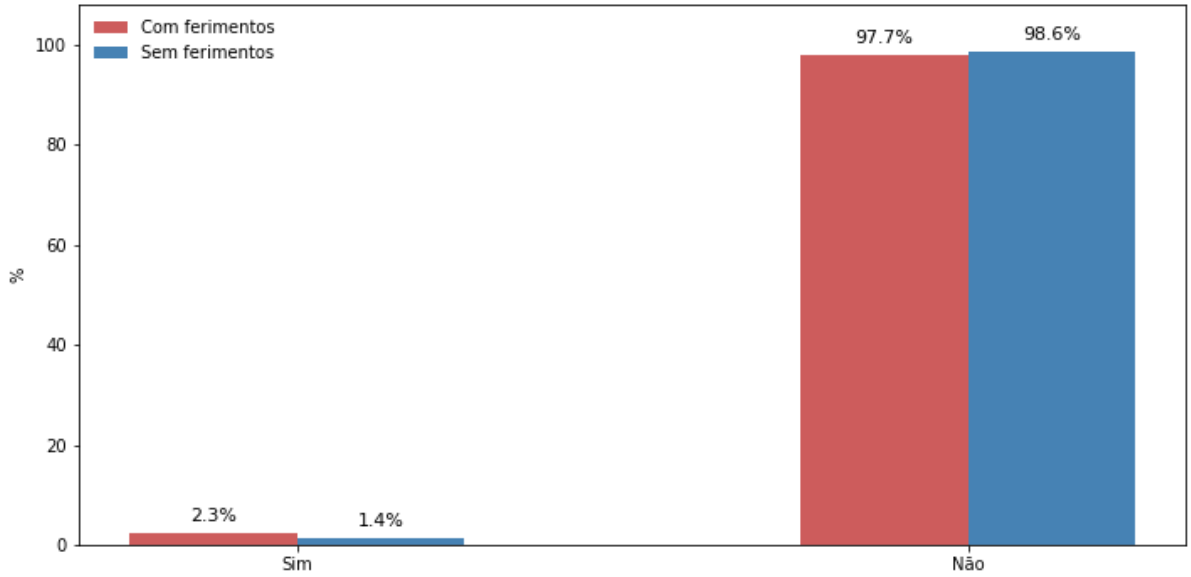

Figura 25 – Distribuição da variável “embriaguez” entre as vítimas com e sem ferimentos

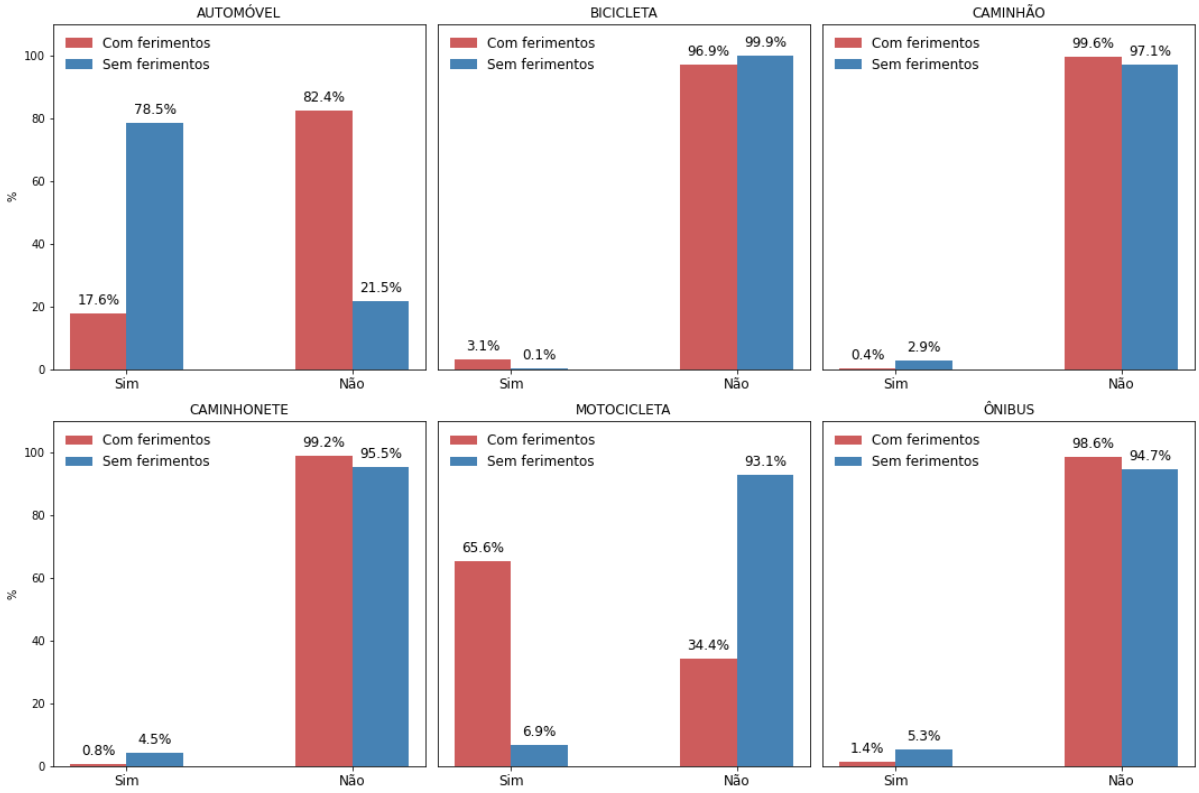

Figura 26 – Distribuição das variáveis “automóvel”, “bicicleta”, “caminhão”, “caminhonete”, “motocicleta” e “ônibus” entre as vítimas com e sem ferimentos

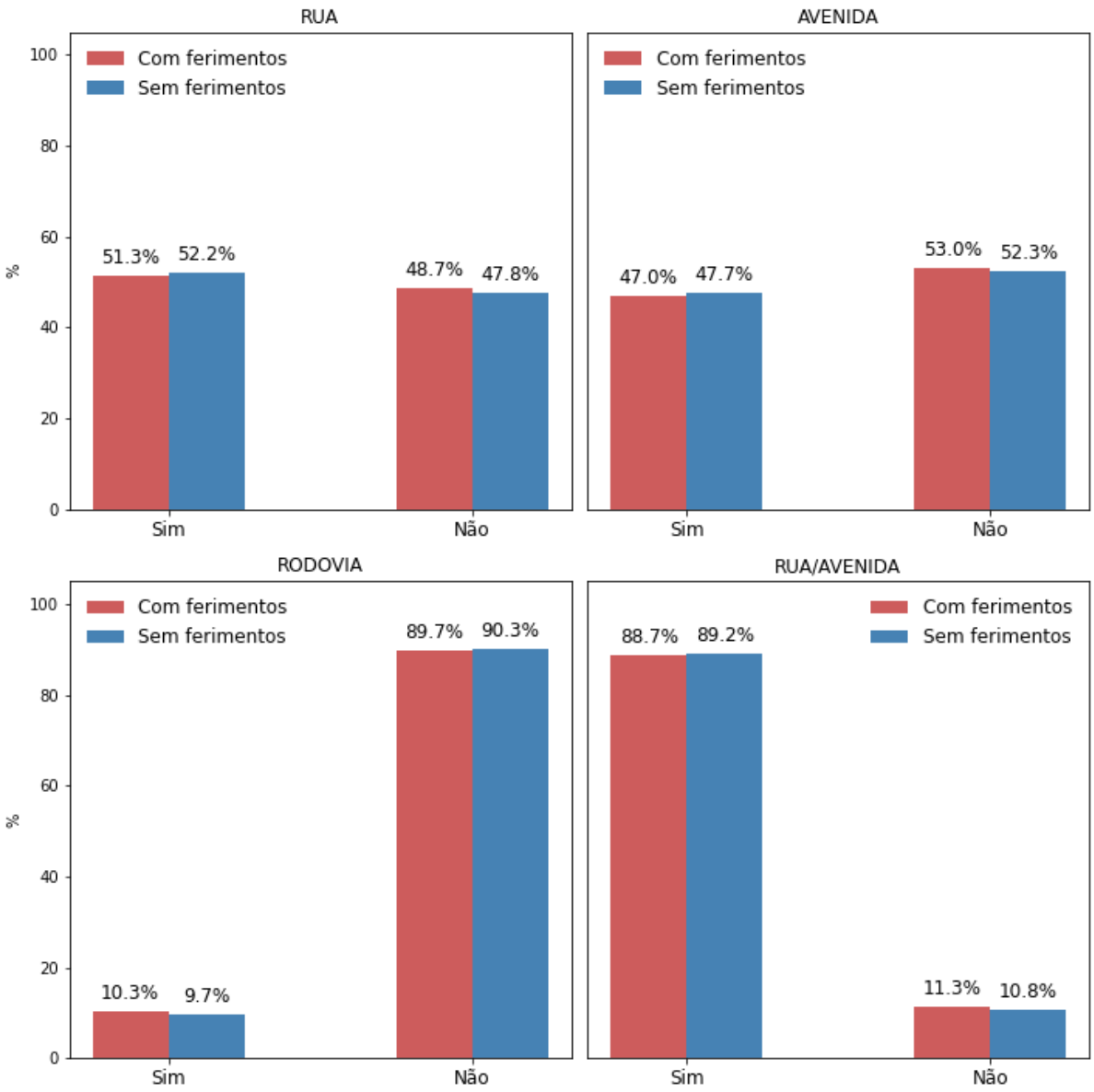

Figura 27 – Distribuição das variáveis “rua”, “avenida”, “rodovia” e “rua/avenida” entre as vítimas com e sem ferimentos

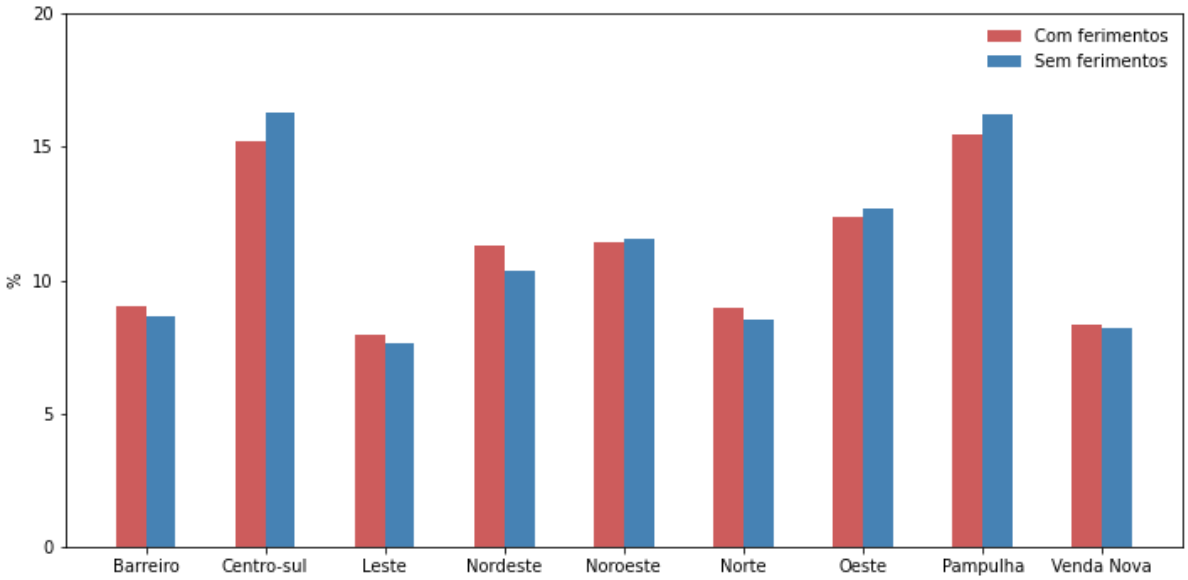

Figura 28 – Distribuição da variável “regional” entre as vítimas com e sem ferimentos

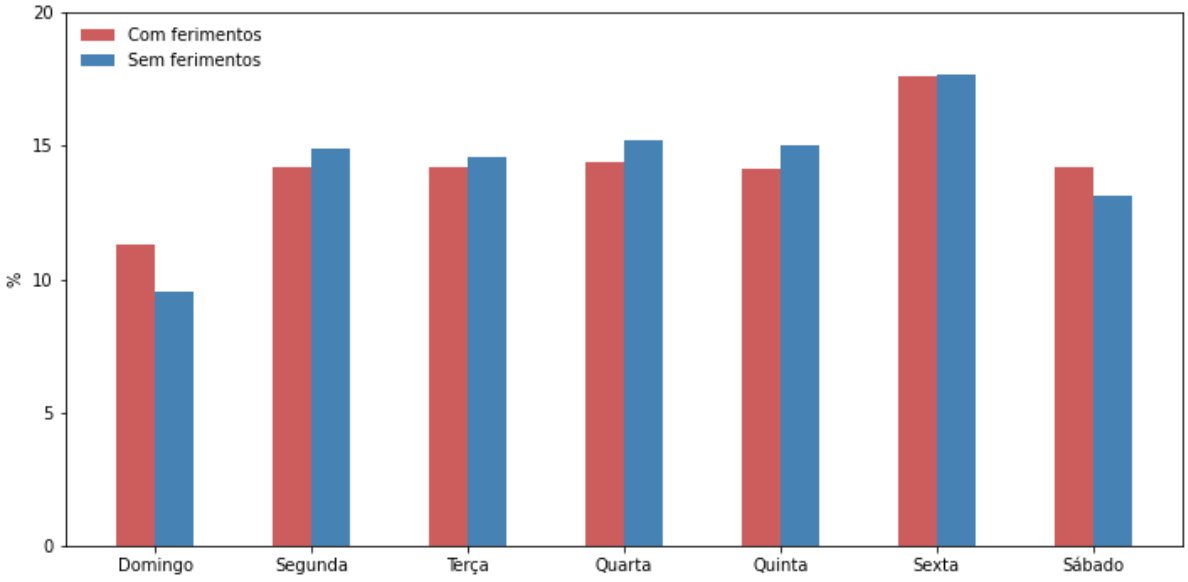

Figura 29 – Distribuição da variável “dia da semana” entre as vítimas com e sem ferimentos

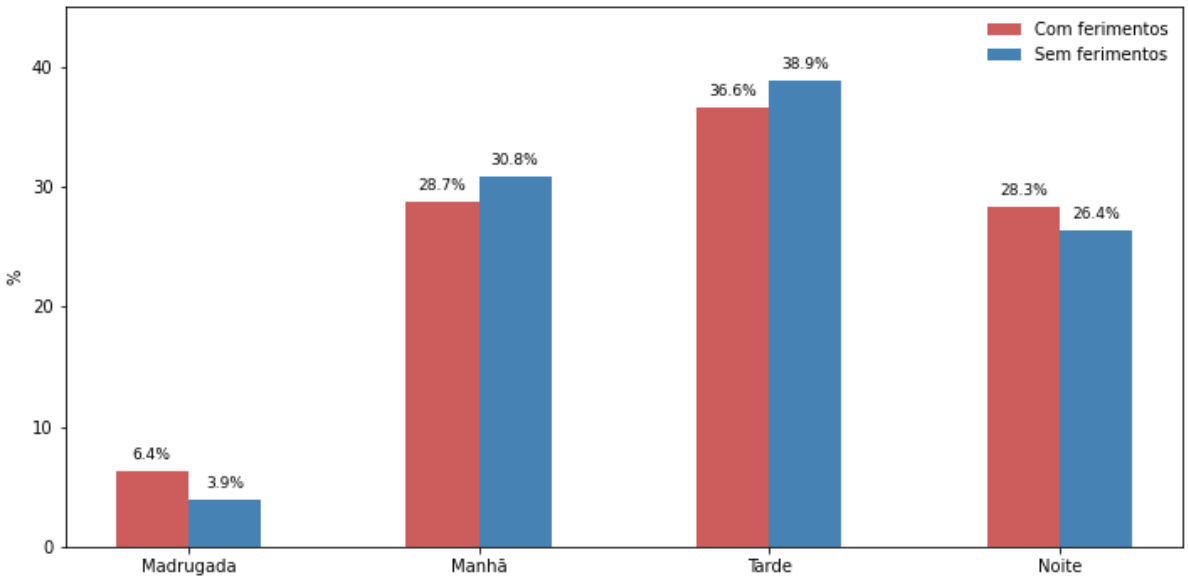

Figura 30 – Distribuição da variável “turno” entre as vítimas com e sem ferimentos

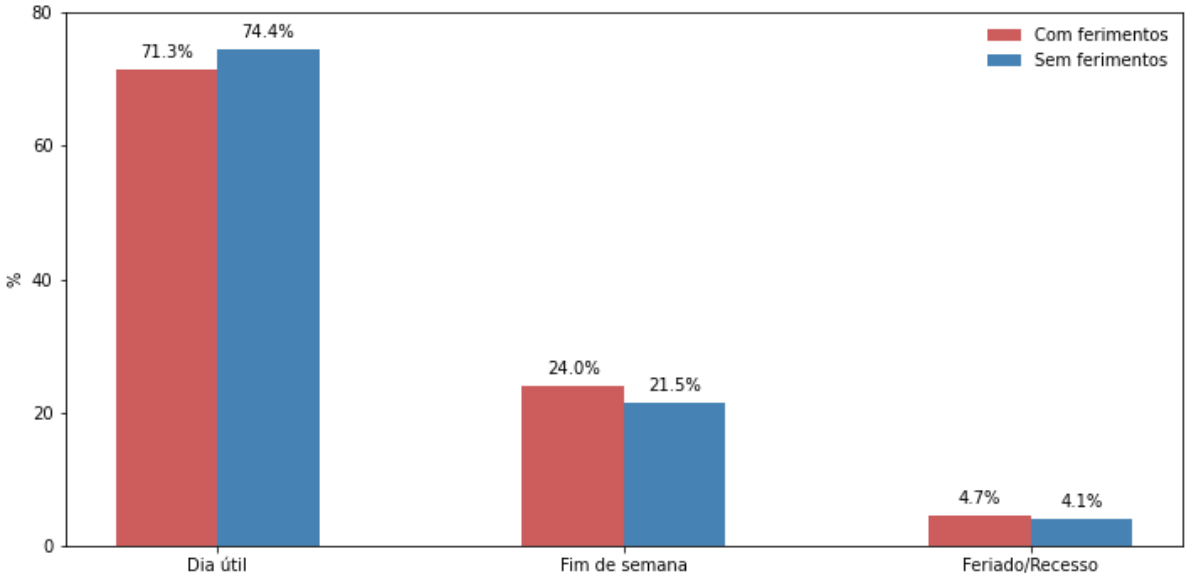

Figura 31 – Distribuição da variável “tipo de dia” entre as vítimas com e sem ferimentos

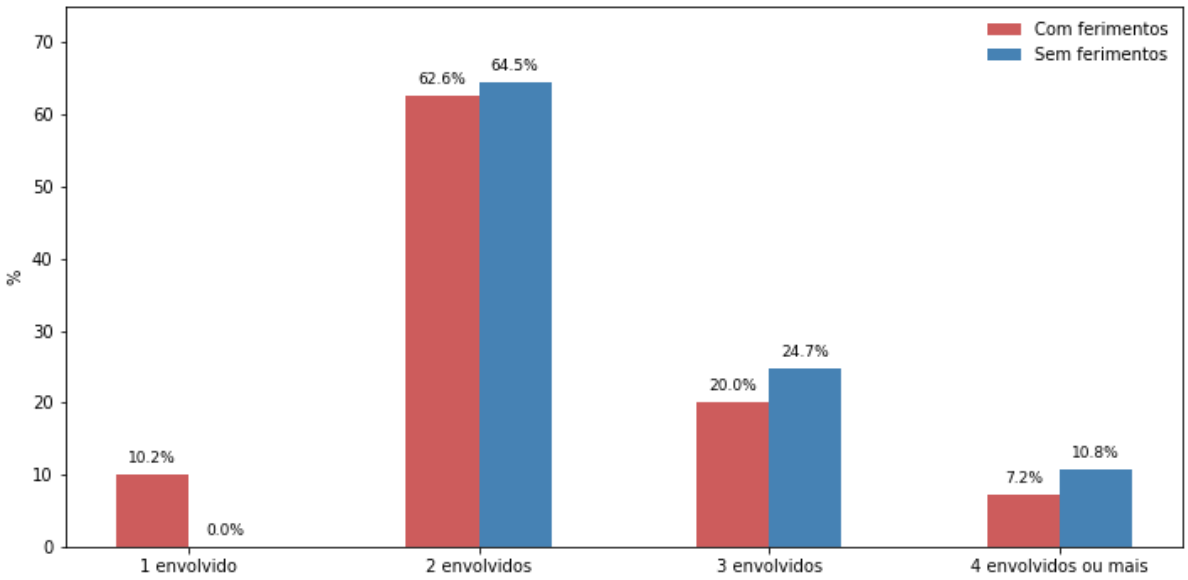

Figura 32 – Distribuição da variável “número de envolvidos – categórica” entre as vítimas com e sem ferimentos

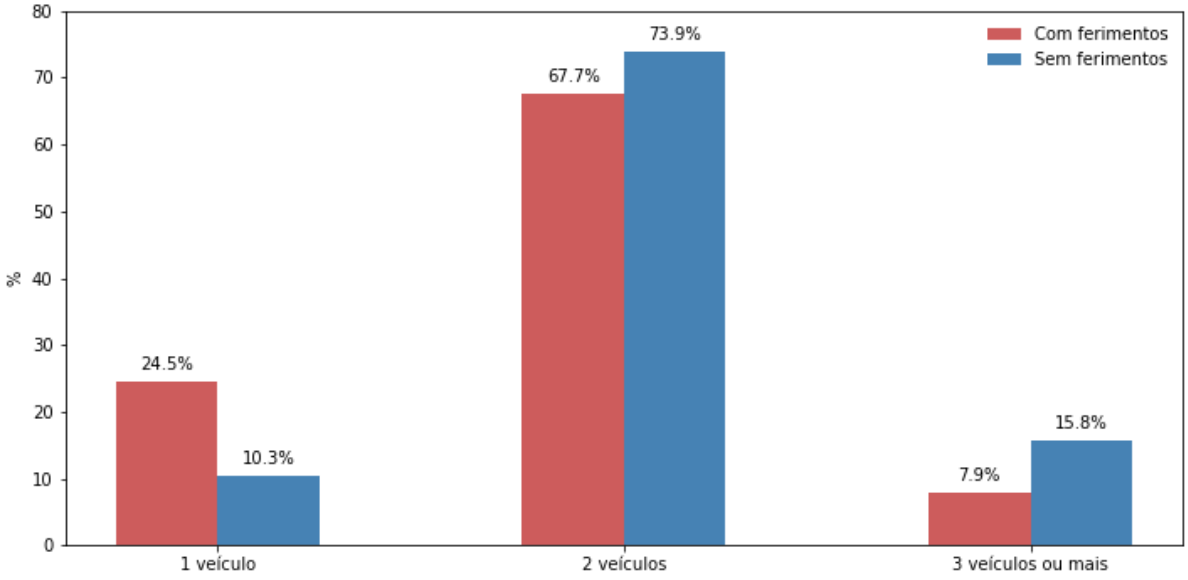

Figura 33 – Distribuição da variável “número de veículos – categórica” entre as vítimas com e sem ferimentos

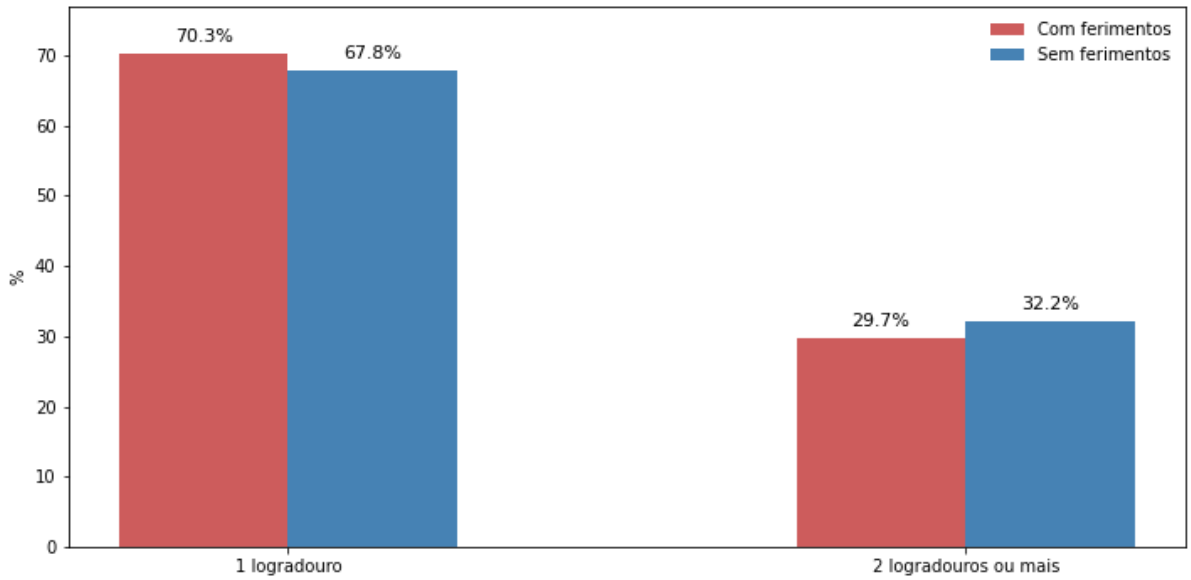

Figura 34 – Distribuição da variável “número de logradouros – categórica” entre as vítimas com e sem ferimentos

5. Criação de Modelos de Machine Learning

O dataset final para análise possui 21.869 registros e 24 colunas, incluindo a variável resposta. Antes da aplicação dos modelos de machine learning foi realizada a codificação e a seleção das variáveis. Para todos os passos foi utilizada a biblioteca “sklearn”, com os módulos “preprocessing”, “feature_selection”, “metrics”, “model_selection”, “linear_model”, “tree”, “ensemble” e “neighbors”. A função “get_dummies” da biblioteca “pandas” também foi utilizada.

5.1. Codificação das variáveis

Das 23 variáveis explicativas do modelo, 15 são binárias, 4 são categóricas nominais (regional, dia da semana, tipo de dia e turno) e 4 são categóricas ordinais (faixa etária, número de vítimas envolvidas, número de veículos envolvidos e número de logradouros envolvidos).

As variáveis binárias foram transformadas em 0 e 1, com o valor 1 indicando a presença da característica de interesse e o sexo feminino, no caso da variável “sexo”.

As variáveis ordinais foram transformadas para valores numéricos sequenciais a partir de 0 com o auxílio da função “OrdinalEncoder()” do módulo “preprocessing” da biblioteca “sklearn” mantendo a ordem natural das categorias.

As variáveis nominais foram transformadas em dummies com a função “get_dummies()” da biblioteca “pandas” mantendo variáveis para todas as categorias com o argumento “drop_first=False”.

Após as alterações, o dataset passou a conter 43 colunas, incluindo a variável resposta.

5.2. Separação da base de treinamento e de teste

Após a codificação das variáveis, o dataset foi dividido em duas partes: uma para treinamento, com 75% dos dados, e uma para teste, com 25% dos dados. Foi utilizada a função “train_test_split()” do módulo “model_selection” da biblioteca “sklearn” e os dados foram divididos de forma proporcional, mantendo a mesma proporção da variável resposta nos dois datasets.

A base de treinamento, então, foi composta por 16.401 registros e a base de teste, por 5.468 registros.

5.3. Seleção de variáveis

Os métodos de seleção de variáveis podem ser divididos em dois grupos principais: não supervisionados e supervisionados. Os métodos não supervisionados utilizam apenas as próprias variáveis explicativas para a definição do conjunto de variáveis a ser utilizado, como por exemplo a correlação entre elas. Os métodos supervisionados utilizam também a variável resposta para definir o conjunto de variáveis mais relevante para o problema e podem ter abordagens de filtro, de wrapper ou intrínsecas ao modelo utilizado. Os métodos baseados em filtro utilizam medidas e testes estatísticos entre a variável resposta e cada variável explicativa de forma individual e selecionam as variáveis de acordo com algum critério. Os métodos baseados em wrapper utilizam modelos de aprendizado de máquina e testam vários subconjuntos de variáveis para obter o com melhor desempenho. Os métodos intrínsecos aos modelos utilizados são aqueles em que o próprio modelo tem a capacidade de realizar a seleção de variáveis, como por exemplo a regressão Lasso.

Para a seleção de variáveis optou-se por utilizar um método de filtro. O método utilizado foi o teste Qui-quadrado entre cada variável explicativa e a variável resposta, considerando que todas são categóricas.

O teste Qui-quadrado pode ser utilizado para avaliar a independência entre duas variáveis categóricas comparando as frequências esperadas caso não houvesse

relação entre as variáveis e as frequências observadas. Os testes foram aplicados com a função “SelectKBest()” do módulo “feature_selection” da biblioteca “sklearn” com os argumentos “score_func=chi2” e “k=’all’”, conforme demonstrado na Figura 35. As variáveis com valor-p menor ou igual a 0,05 (considerando um nível de significância de 5%) foram selecionadas para o ajuste dos modelos.

```
# aplicando teste qui-quadrado entre cada variável explicativa e a variável resposta
chi2 = SelectKBest(score_func=chi2,k='all').fit(X_train,y_train)
```

```
chi2.feature_names_in_[chi2.pvalues_<=0.05]
```

```
array(['Condutor', 'Sexo', 'Cinto', 'Embriaguez', 'Pedestre',
       'Passageiro', 'AUTOMOVEL', 'MOTOCICLETA', 'BICICLETA', 'CAMINHAO',
       'ONIBUS', 'CAMINHONETE', 'Num envolvidos cat', 'Num veiculos cat',
       'Num logradouros cat', 'Faixa etaria', 'Turno_Madrugada',
       'Turno_Tarde', 'Turno_Noite', 'Dia Semana_Domingo',
       'Dia Semana_Sábado', 'Dia_Dia útil', 'Dia_Feriado/Recesso',
       'Dia_Fim de semana'], dtype=object)
```

```
# criando dataframe com variáveis selecionadas pelo valor-p do teste qui-quadrado
X_train_chi2 = X_train[chi2.feature_names_in_[chi2.pvalues_<=0.05]]
```

Figura 35 – Aplicação do teste Qui-quadrado e seleção das variáveis com valor-p menor ou igual a 0,05

As 24 variáveis selecionadas foram: “sexo”, “faixa etária”, “condutor”, “passageiro”, “pedestre”, “cinto de segurança”, “embriaguez”, “automóvel”, “motocicleta”, “bicicleta”, “caminhão”, “ônibus”, “caminhonete”, as variáveis dummies para os turnos “madrugada”, “tarde” e “noite”, as variáveis dummies para os dias da semana “sábado” e “domingo”, todas as variáveis dummies para o tipo de dia (“dia útil”, “fim de semana” e “feriado/recesso”), “número de envolvidos”, “número de veículos” e “número de logradouros”.

Com o objetivo de encontrar o conjunto de variáveis e modelo com melhor desempenho, optou-se por avaliar cada modelo de machine learning com dois conjuntos de variáveis: todas as variáveis disponíveis e as variáveis com valor-p menor ou igual a 0,05 no teste Qui-quadrado.

5.4. Métricas de avaliação

Como o objetivo dos modelos é classificar as vítimas corretamente nas categorias “com ferimentos” e “sem ferimentos”, foram utilizadas quatro métricas para a avaliação e comparação dos modelos utilizados:

- Acurácia média entre as classes: média entre a proporção de acerto em cada uma das duas classes da variável resposta.
- Recall (sensibilidade): proporção de acerto na classe positiva (com ferimentos).
- Especificidade: proporção de acerto na classe negativa (sem ferimentos).
- ROC AUC: área abaixo da curva ROC (sensibilidade vs. 1-especificidade).

Foram utilizadas as métricas acima para garantir uma acurácia adequada em ambas as classes, evitando a escolha de um modelo que possua bom desempenho em uma classe e desempenho ruim na outra (o que pode acontecer utilizando apenas a acurácia geral na avaliação dos modelos).

Foi utilizado o módulo “metrics” da biblioteca “sklearn” e a especificidade foi calculada a partir da acurácia média entre as classes e o recall.

5.5. Modelos de Machine Learning

Os algoritmos de aprendizado de máquina se dividem, principalmente, em três grupos: supervisionados, não supervisionados e por reforço. Os algoritmos supervisionados são aqueles em que os dados são rotulados e os algoritmos são treinados com os dados de entrada e saída. Em maioria, esses algoritmos são utilizados para problemas de regressão e classificação e alguns dos principais modelos são a regressão linear, a regressão logística, árvores de decisão, entre outros. Os algoritmos não supervisionados exploram dados não rotulados para encontrar relações e padrões ocultos. Um dos principais exemplos é a análise de agrupamento (clustering) que possui o objetivo de encontrar grupos de dados

semelhantes em uma amostra. Os algoritmos de aprendizado por reforço são treinados para utilizar a abordagem de “tentativa e erro”, recebendo recompensas em caso de acertos e penalizações em caso de erros.

Com o objetivo de encontrar o modelo com melhor desempenho para a classificação da ocorrência de ferimentos nas vítimas, foram avaliados quatro modelos supervisionados:

- Regressão logística
- Árvore de decisão
- Floresta aleatória
- K-nearest neighbors (KNN)

5.5.1. Regressão Logística

A regressão logística é um método estatístico que possui o objetivo de modelar a relação entre uma variável resposta binária (geralmente codificada como 0 e 1) e uma ou mais variáveis explicativas. O modelo estabelece a relação entre a variável resposta e uma combinação linear das variáveis explicativas através da função de ligação logit. O modelo retorna a probabilidade de uma instância pertencer à classe positiva da variável resposta (codificada como 1) e, em geral, caso a probabilidade seja maior ou igual a 0,5 o registro é classificado como 1 (categoria “positiva”) e caso a probabilidade seja menor que 0,5, classificado como 0 (categoria “negativa”). Em python, umas das formas de se aplicar a regressão logística é através da função “LogisticRegression()” do módulo “linear_model” da biblioteca “sklearn”.

5.5.2. Árvore de Decisão

Uma árvore de decisão é um algoritmo supervisionado não paramétrico que pode ser utilizado para regressão e classificação e utiliza a estratégia de dividir um conjunto de dados recursivamente a fim de resolver um problema de decisão. As árvores possuem nós (raiz, internos e folhas) e ramos, em que cada nó interno utiliza uma condição baseada em uma das variáveis para fazer uma partição nos dados.

No caso de problemas de classificação, dois dos critérios mais utilizados para a definição do atributo a ser utilizado em cada nó são o Índice de Gini, que mede o grau heterogeneidade dos dados (ou a “pureza” de um nó) e a entropia, uma medida da falta de homogeneidade dos dados de entrada em relação a sua classificação. Em cada nó, o objetivo é utilizar o atributo com menor Índice de Gini ou entropia (aumentando o ganho de informação).

Em python, uma possível implementação é a partir da função “DecisionTreeClassifier()” do módulo “tree” da biblioteca “sklearn”. É possível indicar o critério de seleção dos atributos com o argumento “criterion=’gini’” ou “criterion=’entropy’”.

5.5.3. Floresta Aleatória (Random Forest)

O algoritmo Random Forest é um método ensemble, ou seja, que combina o resultado de diferentes modelos para obter um único resultado. O algoritmo cria várias árvores de decisão de maneira aleatória e é realizada uma espécie de votação para definir o resultado final. Ao contrário das árvores de decisão simples, o Random Forest escolhe de maneira aleatória algumas variáveis e dados para inicializar as árvores. Assim como nas árvores de decisão, o Índice de Gini e a entropia podem ser utilizados como critério de qualidade de uma partição.

Em python, pode ser implementado com a função “RandomForestClassifier()” do módulo “ensemble” da biblioteca “sklearn”.

5.5.4. K-nearest neighbors (KNN)

No algoritmo KNN (“K-vizinhos mais próximos”), cada instância representa um ponto em um espaço definido pelas variáveis. De acordo com a métrica de distância definida, cada instância é classificada de acordo com a classificação mais frequente entre os k vizinhos mais próximos. O valor de k deve ser definido pelo usuário e pode ser utilizada validação cruzada para obter o valor que resulta no melhor desempenho da classificação.

Diversas medidas de distância podem ser utilizadas de acordo o tipo das variáveis. No caso do presente trabalho em que todas as variáveis são categóricas, foi utilizada a distância de Hamming.

Em python, a implementação se dá pela função “KNeighborsClassifier()” do módulo “neighbors” da biblioteca “sklearn”.

5.6. Aplicação dos modelos de Machine Learning

Para cada uma das duas possibilidades de conjunto de variáveis, foram aplicados os quatro modelos citados na seção 5.5. No caso das árvores de decisão e das florestas aleatórias, foram ajustados modelos tanto com o Índice de Gini quanto com a entropia como medida de impureza. Para a aplicação do algoritmo KNN, para cada conjunto de variáveis foi realizada uma busca em grade com validação cruzada (com 10 dobras) para identificar o valor de k que resultasse no melhor desempenho de classificação. Baseado na literatura, um ponto de partida para a investigação do melhor valor de k é a raiz quadrada do número de instâncias na base de dados. Como a base de treinamento possui 16.401 amostras, com raiz quadrada aproximada igual a 128, optou-se então por fazer a busca do valor de k nos valores ímpares entre 1 e 138 (a raiz quadrada do número de instâncias mais 10). Os valores de k com melhor desempenho foram 49 vizinhos para o conjunto de todas as variáveis e 25 vizinhos para o conjunto de variáveis selecionadas pelo teste Qui-quadrado. Todos os modelos foram treinados com validação cruzada com 10 dobras.

As figuras 36 a 39 apresentam os boxplots dos resultados das métricas avaliadas nas validações cruzadas em cada modelo utilizado. Os modelos de regressão logística foram denominados como “LR”, as árvores de decisão como “DT” com os critérios “gini” e “entropy” e as florestas aleatórias como “RF” também com os critérios “gini” e “entropy”. Para todos os modelos, “all” indica o conjunto de todas as variáveis e “X2” o conjunto de variáveis selecionadas pelo teste Qui-quadado.

É possível perceber que os modelos com menor desempenho em todas as métricas foram as árvores de decisão com todas as variáveis, mesmo com valores acima de 80%.

Os modelos com melhor desempenho na acurácia média foram o KNN e as florestas aleatórias com o conjunto de variáveis selecionadas pelo teste Qui-quadrado. Em relação ao recall, os modelos com melhor desempenho foram o KNN com o conjunto de variáveis selecionadas pelo teste Qui-quadrado e as florestas aleatórias com todas as variáveis.

Para a especificidade, a regressão logística com o conjunto de variáveis selecionadas pelo teste Qui-quadrado apresentou o maior valor, seguida pelo KNN com o mesmo conjunto de dados. Considerando a área abaixo da curva ROC (ROC AUC), as florestas aleatórias e o KNN com o conjunto de variáveis selecionadas pelo teste Qui-quadrado apresentaram os maiores valores.

Considerando as quatro métricas conjuntamente, o modelo que apresentou o melhor desempenho geral foi o KNN com o conjunto de variáveis selecionadas pelo teste Qui-quadrado, em que o valor de k que obteve melhor desempenho foi 25.

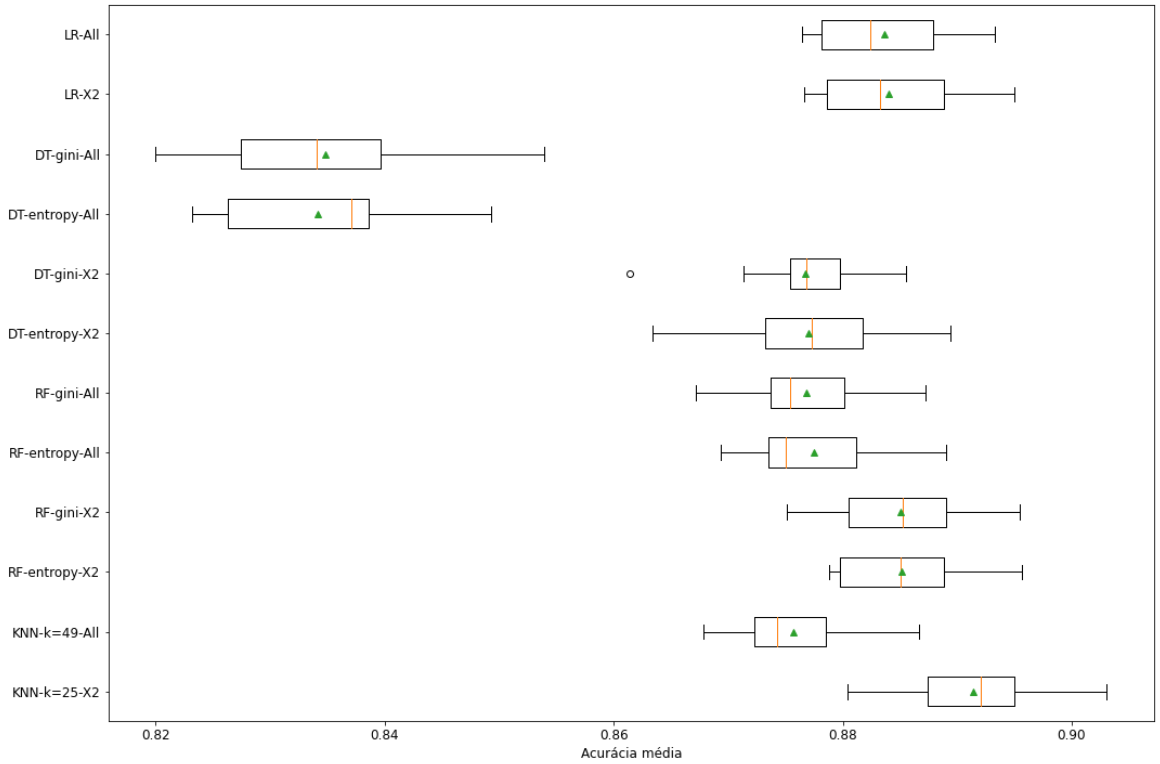

Figura 36 – Boxplots dos valores de acurácia média nas validações cruzadas para os modelos avaliados

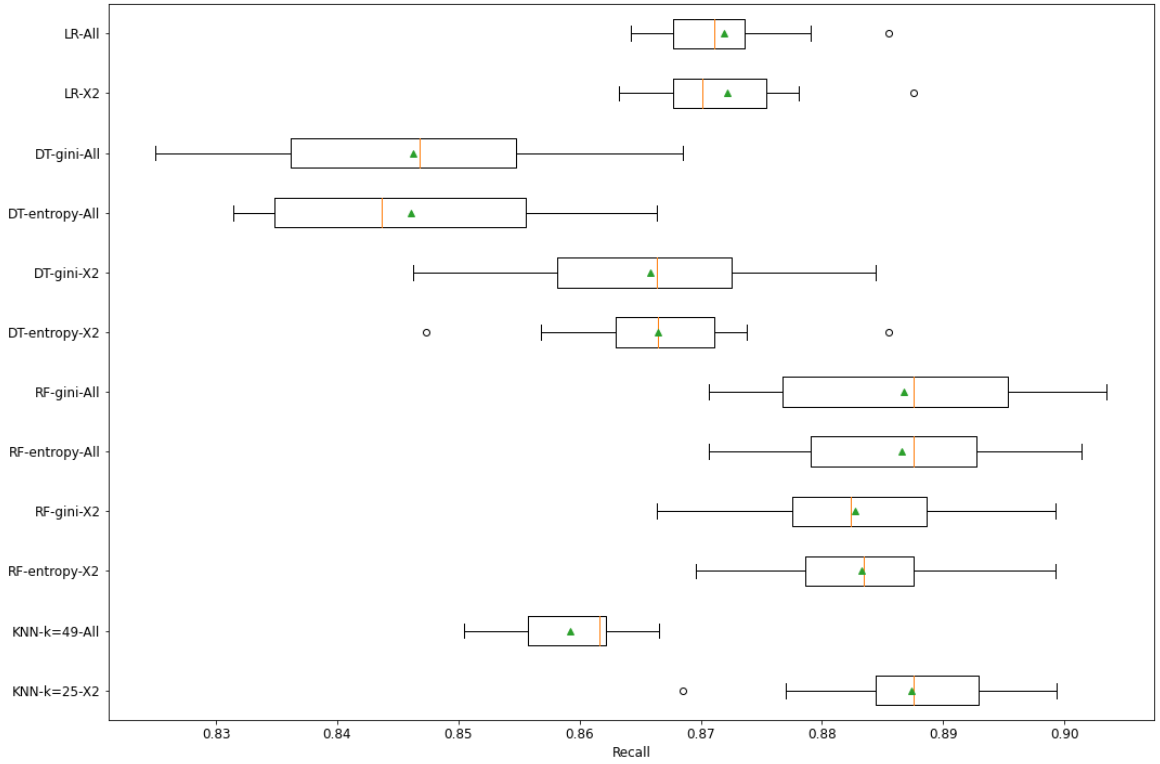

Figura 37 – Boxplots dos valores de recall nas validações cruzadas para os modelos avaliados

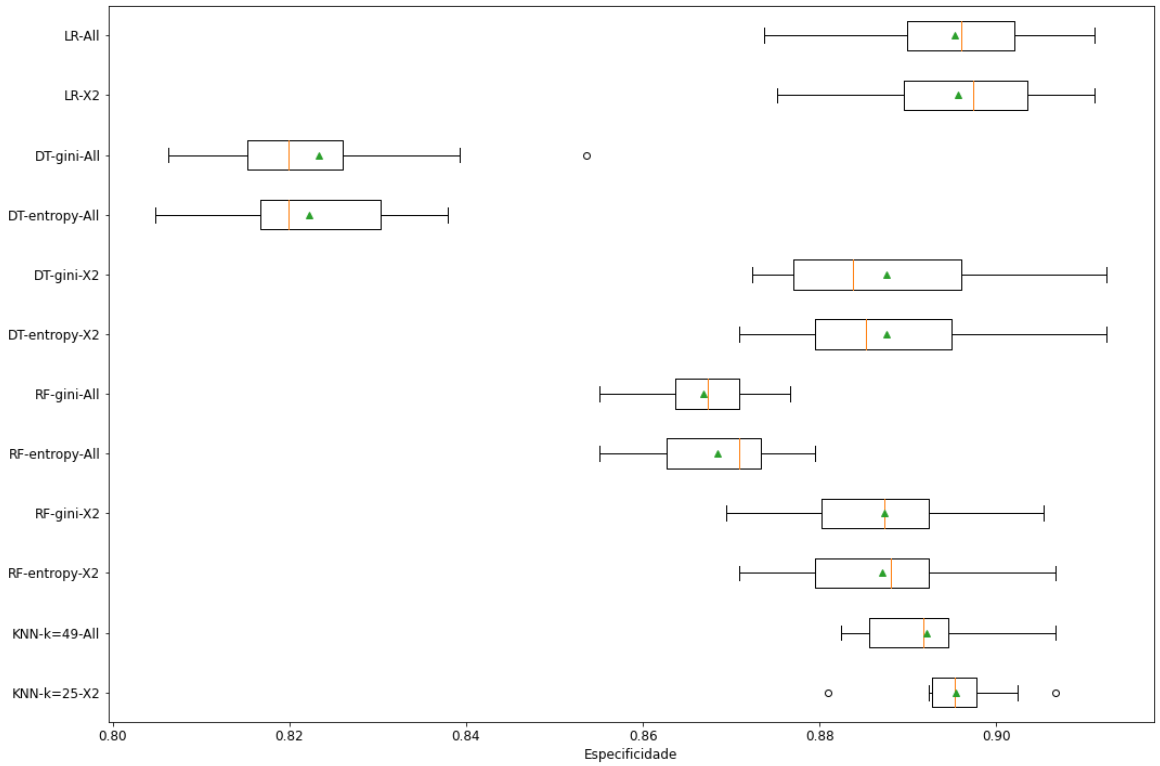

Figura 38 – Boxplots dos valores de especificidade nas validações cruzadas para os modelos avaliados

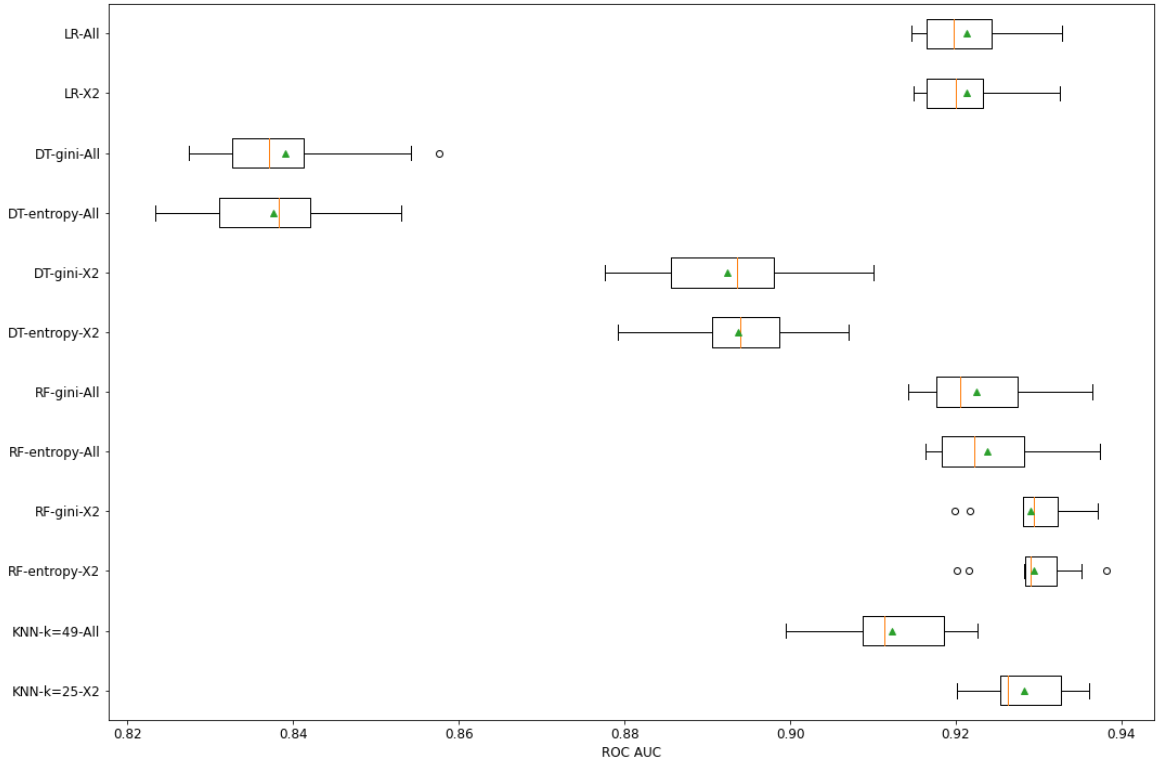

Figura 39 – Boxplots dos valores de ROC AUC nas validações cruzadas para os modelos avaliados

A Tabela 6 apresenta as métricas dos modelos avaliados em cada conjunto de dados.

Tabela 6 – Métricas dos modelos avaliados

Modelo	**Acurácia média**	**Recall**	**Especificidade**	**ROC AUC**
KNN-k=25-X2	0,8914	0,8874	0,8954	0,9283
RF-entropy-X2	0,8852	0,8833	0,8871	0,9294
RF-gini-X2	0,8851	0,8827	0,8874	0,9291
LR-X2	0,8840	0,8722	0,8957	0,9213
LR-All	0,8836	0,8719	0,8953	0,9213
RF-entropy-All	0,8775	0,8865	0,8684	0,9238
DT-entropy-X2	0,8770	0,8665	0,8875	0,8938
RF-gini-All	0,8768	0,8868	0,8669	0,9225
DT-gini-X2	0,8767	0,8659	0,8875	0,8924
KNN-k=49-All	0,8756	0,8592	0,8921	0,9123
DT-gini-All	0,8348	0,8463	0,8234	0,8391
DT-entropy-All	0,8341	0,8460	0,8222	0,8377

6. Interpretação dos Resultados

Todos os modelos avaliados apresentaram desempenho acima de 80% em todas as métricas e o modelo com melhor desempenho geral foi o K-nearest neighbors com 89% de acurácia média. O conjunto de dados utilizado neste modelo foi o selecionado pelo teste Qui-quadrado e o melhor número de vizinhos encontrado foi 25.

Após a definição do modelo com melhor desempenho, o modelo KNN com 25 vizinhos foi aplicado aos dados de teste que conta com 5.468 registros, sendo 2.324 de vítimas sem ferimentos e 3.144 de vítimas com ferimentos. Como o conjunto de variáveis que obteve o melhor desempenho foi o selecionado pelo teste Qui-quadrado, foram selecionadas apenas as devidas colunas do dataset de teste, como apresentado na Figura 40.

```
# Selecionando apenas as mesmas colunas do dataset X_train_chi2
X_test_chi2 = X_test[X_train_chi2.columns]
```

```
X_test_chi2.columns
```

```
Index(['Condutor', 'Sexo', 'Cinto', 'Embriaguez', 'Pedestre', 'Passageiro',
       'AUTOMOVEL', 'MOTOCICLETA', 'BICICLETA', 'CAMINHAO', 'ONIBUS',
       'CAMINHONETE', 'Num envolvidos cat', 'Num veiculos cat',
       'Num logradouros cat', 'Faixa etaria', 'Turno_Madrugada', 'Turno_Tarde',
       'Turno_Noite', 'Dia Semana_Domingo', 'Dia Semana_Sábado',
       'Dia_Dia útil', 'Dia_Feriado/Recesso', 'Dia_Fim de semana'],
      dtype='object')
```

Figura 40 – Seleção das variáveis selecionadas pelo teste Qui-quadrado no dataset de teste

O modelo foi ajustado nos dados de treinamento e foi realizada a previsão com os dados de teste já com as colunas adequadas, conforme Figura 41.

```
mod1 = KNeighborsClassifier(metric='hamming',n_neighbors=25).fit(X=X_train_chi2, y=y_train)
prevteste = mod1.predict(X_test_chi2)
```

Figura 41 – Aplicação do modelo selecionado nos dados de treinamento e previsão nos dados de teste

Avaliando a matriz de confusão, apresentada na Figura 42, é possível verificar que o modelo obteve ótimo desempenho com acurácia média igual a 0,9012, recall igual a 0,9030 e especificidade igual a 0,8993. A área abaixo da curva ROC foi igual a 0,9351.

```
disp = ConfusionMatrixDisplay(confusion_matrix(y_test,prevteste),display_labels=['Sem ferimentos','Com ferimentos'])
disp.plot(cmap='Blues');
```

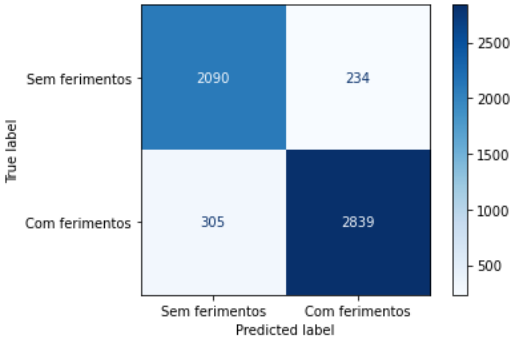

Figura 42 – Matriz de confusão dos valores preditos do dataset de teste

7. Apresentação dos Resultados

Para a apresentação do processo de desenvolvimento do projeto, foi utilizado o modelo proposto por Vasandrini (disponível em https://towardsdatascience.com/a-data-science-workflow-canvas-to-kickstart-your-projects-db62556be4d0), apresentado na Figura 43.

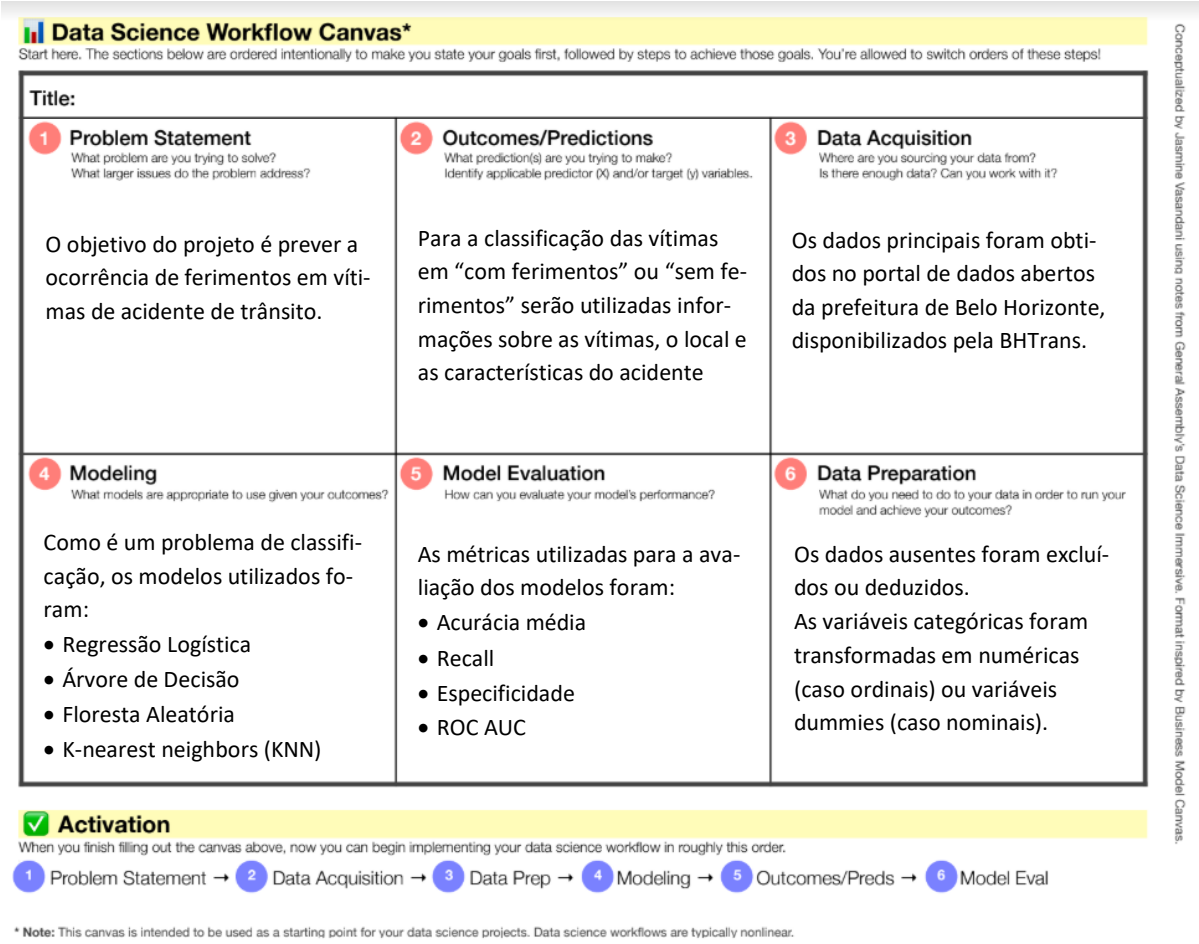

Figura 43 – Workflow do projeto de ciência de dados conforme modelo de Vasandrini

As variáveis que se mostraram mais relevantes para a previsão da ocorrência de ferimentos em vítimas de acidentes de trânsito foram o sexo e a faixa etária da vítima, se o envolvido era condutor, passageiro ou pedestre, se estava utilizando cinto de segurança, se estava sob efeito de álcool, em que automóvel estava (automóvel, motocicleta, bicicleta, caminhão, ônibus ou caminhonete), em que turno ocorreu o acidente (madrugada, tarde ou noite), se o acidente ocorreu no sábado ou no

domingo, qual era o tipo de dia (dia útil, fim de semana ou feriado/recesso) e o número de vítimas, de veículos e de logradouros envolvidos no acidente.

Das 21.869 vítimas analisadas, 12.575 tiveram ferimentos. Comparando as vítimas com e sem ferimentos, as variáveis que se mostraram mais relevantes para diferenciação entre os grupos foram a faixa etária, se o envolvido era condutor, passageiro ou pedestre, se utilizava cinto de segurança, se estava em automóvel ou motocicleta, se era o único envolvido no acidente e se o acidente envolveu apenas 1 veículo. Vítimas com idade entre 18 e 29 anos e que estavam em motocicletas se destacaram com uma maior frequência de ferimentos.

Após a implementação dos quatro modelos de machine learning utilizados (regressão logística, árvore de decisão, floresta aleatória e KNN), o modelo que apresentou melhor desempenho foi o KNN com 25 vizinhos e utilizando apenas as variáveis que apresentaram relação significativa com a ocorrência de ferimentos a 5% de significância pelo teste Qui-quadrado.

Aplicando o modelo selecionado aos dados de teste (que não foram utilizados na escolha do modelo), foi obtida uma acurácia média de 90% e uma área abaixo da curva ROC de 0,9351. A Figura 44 a seguir mostra a curva ROC.

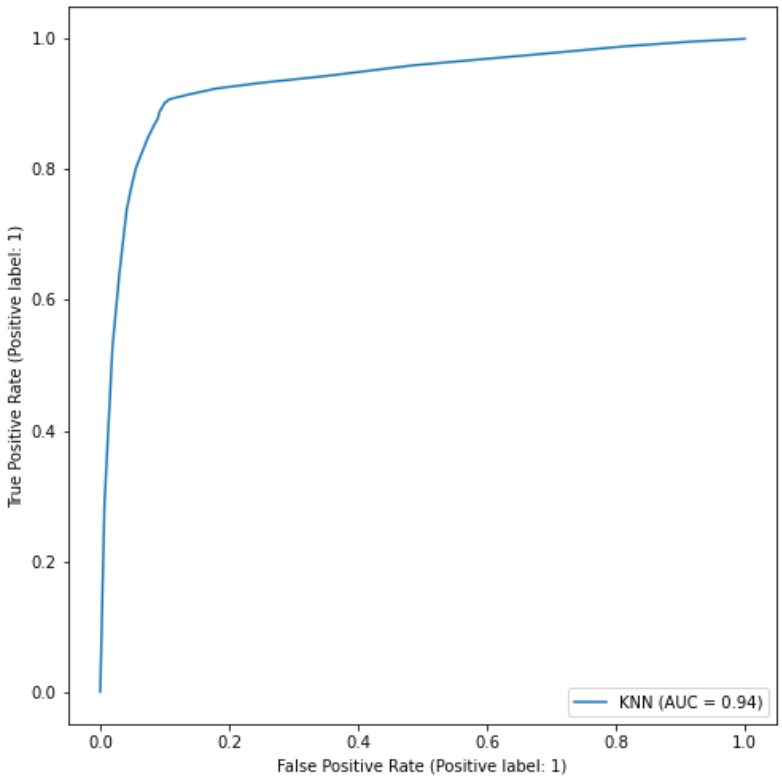

Figura 44 – Curva ROC para a classificação da severidade das vítimas

O modelo com melhor desempenho encontrado foi capaz de prever corretamente a ocorrência de ferimentos nas vítimas em cerca de 90% dos casos, podendo ser utilizado para orientar os locais e momentos em que seria necessário maior fiscalização e atendimento médico mais rápido. Considerando as características e comportamento das vítimas, também é possível conscientizar os grupos mais vulneráveis e que costumam se ferir mais em acidentes a fim de minimizar comportamentos de risco.

8. Links

Link para o vídeo: https://youtu.be/5r-uLOwQBgI

Link para o repositório com os dados utilizados e o código criado: https://github.com/Raquel-Marinho/TCC-Ciencia-de-Dados-PUC-Minas

REFERÊNCIAS

Organização Pan-Americana da Saúde. **Trânsito: um olhar da saúde para o tema.** Brasília: OPAS; 2018.

PREFEITURA DE BELO HORIZONTE. **Portal de Dados Abertos.** Disponível em: https://dados.pbh.gov.br/. Acesso em 27 de julho de 2023.

MCKINNEY, W. **Data structures for statistical computing in python.** Proceedings of the 9th Python in Science Conference, vol. 445, 2010.

HARRIS, C.R., et al. **Array programming with NumPy.** Nature vol. 585, p. 357-362 (2020).

HUNTER, J. D. **Matplotlib: A 2D Graphics Environment.** Computing in Science & Engineering, vol. 9, no. 3, p. 90-95, 2007.

PEDREGOSA el al. **Scikit-learn: Machine Learning in Python.** Journal of Machine Learning Research, vol. 12, p. 2825-2830, 2011.

APÊNDICE

Programação/Scripts

O script e os arquivos utilizados estão disponíveis no repositório do GitHub: https://github.com/Raquel-Marinho/TCC-Ciencia-de-Dados-PUC-Minas