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

Leandra Inácio de Paula

ANÁLISE DA PROBABILIDADE DE UM ALUNO DE ENSINO BÁSICO TER FUMADO AO MENOS UMA VEZ EM RELAÇÃO A DADOS SOCIAIS APRESENTADOS

Belo Horizonte
2023

Leandra Inácio de Paula

ANÁLISE DA PROBABILIDADE DE UM ALUNO DE ENSINO BÁSICO TER FUMADO AO MENOS UMA VEZ EM RELAÇÃO A DADOS SOCIAIS APRESENTADOS

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 6
1.1. Contextualização 6
1.2. O problema proposto 7
1.3. Objetivos 8
2. Coleta de Dados 9
3. Processamento/Tratamento de Dados 11
3.1 Importação das bibliotecas 11
3.2 Importação e tratamento da base de dados principal 12
3.3 Importação e tratamento da base de dados de enriquecimento 15
3.4 Enriquecimento da base de dados principal 16
4. Análise e Exploração dos Dados 18
4.1 Identificação de outliers 18
4.2 Visualização gráfica para dados nominais 19
4.2.1 Quantidade de alunos que já experimentaram cigarros 19
4.2.2 Representação gráfica de outros dados da base 20
4.3 Mapas de calor e combinações de dados do conjunto 25
5. Criação de Modelos de Machine Learning 29
5.1 Importação das bibliotecas 29
5.2 Processamento dos dados para criação dos modelos de aprendizado de máquina 30
5.3 Tunning 32
5.3.1 Tunning para Randon Forest 32
5.3.2 Tunning para árvore de decisão 33
5.3.3 Tunning para Redes Neurais 34
5.4 Implementação dos modelos de machine learning 35
5.4.1 Treinamento com Random Forest 36
5.4.2 Treinamento com Árvore de Decisão 36
5.4.3 Treinamento com Redes Neurais 37
5.5 Validação Cruzada 37

6. Interpretação dos Resultados 38

7. Conclusão 42

8. Links 43

Referências 44

APÊNDICE 45

1. Funções de transformações das respostas da pesquisa de formato numérico para formato descritivo: 45

2. Processamento de dados da base de enriquecimento 49

2.1 Enriquecimento dos dados 49

3. Exploração e Representação dos dados 49

3.1 Mapas de calor para interpretação dos dados da base de estudo 51

4. Criação dos modelos de machine learning 55

4.1 Predições do modelo de Randon Forest 55

4.2 Predições do modelo de Árvore de Decisão 55

4.3 Predições do modelo de Redes Neurais 56

5. Cross Validation 57

Lista de Figuras

Figura 1: Bibliotecas importadas para o tratamento e exploração dos dados *12*
Figura 2: Comando para renomear colunas importantes para o trabalho *12*
Figura 3: Implementação da transformação dos dados de numéricos para descritivos *14*
Figura 4: Trecho de código para remoção das linhas em que não há resposta para colunas específicas *15*
Figura 5: Separação do nome da cidade e a sigla do estado *16*
Figura 6: Exemplo da base de dados após o tratamento *16*
Figura 7: Informações e tipo de cada coluna da base *17*
Figura 8: Verificação de campos nulos na base de dados final *17*
Figura 9: Boxplot para identificação de outliers na coluna de IDH_EDUCACAO da base de dados *18*
Figura 10: Representação da quantidade de alunos que já experimentaram ou não cigarros ao menos uma vez *20*
Figura 11: Idade predominante dos alunos que participaram da pesquisa *21*
Figura 12: Etnia autodeclarada pelos alunos que participam da amostragem analisada *22*
Figura 13: Ano escolar predominante dos alunos analisados *23*
Figura 14: Incidência de pais e responsáveis fumantes *24*
Figura 15: Incidência de amigos que fumaram perto do aluno analisado nos últimos 30 dias *25*
Figura 16: Mapa de calor entre alunos que já experimentaram cigarros e etnia autodeclarada *26*
Figura 17: Mapa de calor entre responsável fumante e alunos que já experimentaram cigarros *28*
Figura 18: Mapa de calor de amigos que fuma em relação a alunos que já experimentaram cigarros *29*
Figura 19: Bibliotecas importantes para criação dos modelos de machine learning *30*
Figura 20: código de separação da classe alvo e dos dados de entrada seguido do escalonamento dos dados de entrada *31*
Figura 21: Dados de alunos que já experimentaram cigarros (1.0) e que nunca experimentaram cigarros (2.0) após utilização do undersampling *31*
Figura 22: Preparação dos parâmetros do modelo e implementação do tunning para Random Forest *33*
Figura 23: Preparação dos parâmetros do modelo e implementação do tunning para árvore de decisão *34*
Figura 24: Preparação dos parâmetros do modelo e implementação do tunning para Redes Neurais *35*
Figura 25: Implementação do treinamento de Random Forest *36*
Figura 26: Implementação do treinamento de Árvore de Decisão *36*
Figura 27: Implementação do treinamento de Redes Neurais *37*
Figura 28: Implementação do Cross Validation para Random Forest *37*
Figura 29: Implementação do Cross Validation para árvore de decisão *38*
Figura 30: Implementação do Cross Validation para redes neurais *38*
Figura 31: Matriz de confusão Random Forest *39*
Figura 32: Matriz de confusão Árvore de Decisão *40*
Figura 33: Matriz de confusão Redes Neurais *41*
Figura 34: Representação do sexo predominante entre os alunos analisados *50*
Figura 35: Mapa de calor dos municípios analisados em relação a alunos que já experimentaram cigarros *51*
Figura 36: Mapa de calor da relação entre tipo da escola e alunos que já experimentaram cigarros *52*
Figura 37: Mapa de calor da relação entre sexo do aluno e se aluno já experimentou cigarros *52*
Figura 38: Mapa de calor da relação entre o ano escolar e alunos que já experimentaram cigarros *53*
Figura 39: Mapa de calor da relação entre alunos que já experimentaram narguilé e alunos que já experimentaram cigarros *53*
Figura 40: Mapa de calor da relação entre alunos que já experimentaram cigarros eletrônico e já experimentaram cigarros de tabaco *54*
Figura 41: Mapa de calor da relação entre o IDH da cidade analisada e alunos que já experimentaram cigarros *54*

1. Introdução

1.1. Contextualização

Em outubro de 2019 o Instituto Nacional de Câncer – INCA, divulgou uma pesquisa com informações estatísticas sobre a prevalência do tabagismo no Brasil (INCA, 2022). Pesquisas como esta são realizadas desde 1997 com o intuito de os resultados serem capazes de monitorar tendências do consumo do tabaco, assim como adotar medidas mais eficazes para a luta contra o tabagismo no país. Pesquisas apontaram que os dados de fumantes adultos no Brasil, isto é, pessoas maiores de 18 anos, tem regredido continuamente nos últimos anos. Em 1989 cerca de 34% da população acima de 18 anos era fumante, este dado reduziu para 18,5% em 2008 e para 12,6% em 2019 no último estudo realizado.

No entanto, entre os jovens menores de 18 anos, mais especificamente jovens de 13 a 17 anos, as pesquisas feitas por diversas instituições no país, revelaram que a experimentação de cigarro ou uso do tabaco tiveram um aumento, sendo de 6,6% em 2015 para 6,8% em 2019. Além disso, foi identificado que entre os jovens que já experimentaram ou fazem uso de cigarros, a maioria é do sexto masculino e frequentam a rede pública de ensino. É possível identificar na mesma pesquisa que houve também um crescimento na popularidade de cigarros eletrônicos conhecidos como vaper ou e-cig, desta forma as pesquisas também mostraram que para este tipo de cigarro a maior população de jovens usuários ou que já experimentaram se concentra em escolas de rede privada, além de também possuir um maior percentual na região centro-oeste do país.

Sendo assim, é possível perceber que há um alerta de saúde pública para o aumento do consumo de tabaco entre jovens e adolescentes, pois como é amplamente divulgado o consumo de cigarros podem causar diversas doenças, cerca de 50 (Silva), causadas principalmente pela nicotina presente no tabaco. Visto isso, é de grande importância social estudos que procuram entender o comportamento de jovens que podem ter uma tendência em experimentar ou se tornarem usuários de tabaco, através de dados sociais que possibilitem a identificação destes jovens e que possa contribuir para medidas mais eficazes e direcionas para o público-alvo.

1.2. O problema proposto

Como citado no tópico anterior, o uso de cigarro pode provocar diversas doenças, principalmente respiratórias, devido às aproximadamente 4.720 substâncias tóxicas presentes na fumaça. Desta forma, um estudo capaz de identificar um padrão social em jovens e adolescentes que possuem uma tendência em se tornarem fumantes é de grande importância, visto que campanhas de prevenção ao fumo podem ser direcionadas principalmente a este público com características em comum.

Os dados utilizados para realização desta pesquisa são do PeNSE – Pesquisa Nacional de Saúde do Escolar, realizada pelo IBGE e o Ministério da Saúde, com o apoio também do Ministério da Educação. O questionário respondido pelos alunos possui muitos dados sociais e escolares, que vão desde qualidade da alimentação cotidiana dos alunos até a experimentação de drogas ou outros entorpecentes. Sendo assim, para esta análise especificamente serão utilizados dados sociais respondidos por jovens, a fim de identificar um padrão social para àqueles que já experimentaram cigarro ao menos uma vez na vida.

Por fim, para realização desta pesquisa foram filtrados apenas dados sociais que foram julgados relevantes para o estudo, buscando analisar somente alunos de rede pública ou privada das capitais da região sudeste do país. Além disso, foram consideradas apenas respostas de sim ou não para a pergunta “Alguma vez na vida, você já fumou cigarro, mesmo uma ou duas tragadas?”. Os dados utilizados são do relatório de microdados do PeNSE 2019, juntamente com uma fonte de enriquecimento dos dados, acrescentando o IDH (Índice de Desenvolvimento Humano) de cada cidade de estudo. Sendo os dados de IDH colhidos pelo último censo do IBGE em 2010.

1.3. Objetivos

O objetivo desde trabalho é:

- Identificar um padrão social que leva jovens de 13 a 17 anos, ou mais, utilizarem cigarro, ao menos uma vez na vida;
- Verificar qual dos dados sociais tem maior impacto na probabilidade da experimentação de cigarros pelos jovens;
- Obter resultados que possam auxiliar nas ações de medidas preventivas ao fumo entre adolescentes e jovens.

2. Coleta de Dados

O dataset utilizado no trabalho é resultado da união entre dois datasets diferentes obtidos em diferentes fontes. O dataset principal para a composição da base de pesquisa utilizada, foi obtido através de uma pesquisa em escolas públicas e privadas entre alunos de 13 à 17 anos, em sua maioria, que responderam um questionário realizado pelo PeNSE – Pesquisa Nacional de Saúde Escolar. Este dataset foi obtido no endereço eletrônico https://www.ibge.gov.br/estatisticas/sociais/educacao/9134-pesquisa-nacional-de-saude-do-escolar.html?=&t=downloads localizado no site do IBGE (IBGE, 2022). O arquivo utilizado foi o realizado em 2019 e possui dois docmentos, um no formato csv que é o que possui os resultados da pesquisa realizada e um outro arquivo em formato xls , que é o dicionario de dados para possibilitar a interpretação dos dados coletados.

O arquivo principal possui um total de 306 colunas e ao total possui 165.838 linhas. A Tabela 1 mostra apenas os campos foram utilizados neste trabalho, apenas campos que foram julgados importantes e influentes na relação com o cigarro entre os alunos. É possível acessar o dataset principal e o dicionário de dados nos arquivos que acompanham este trabalho.

Tabela 1: Colunas do dataset PeNSE que foram utilizadas nesta pesquisa

Nome da coluna/campo	Nome original no dataset PeNSE	Descrição / perguntas do questionário	Tipo
MUNICIPIO_CAP	MUNICIPIO_CAP	Município da capital	Numérico
TIPO_ESCOLA	DEP_ADMIN	Dependência administrativa, tipo da escola do aluno	Numérico
SEXO	B01001a	Sexo do aluno	Numérico
IDADE	B01003	Idade do aluno	Numérico
ETINIA	B01002	Cor ou raça do aluno	Numérico
ANO_ESCOLAR	B01021a	Ano escolar do aluno	Numérico
MORA_MAE	B01006	Se o aluno mora com a mãe	Numérico
MORA_PAI	B01007	Se o aluno mora com o pai	Numérico
JA_FUMOU	B04001	Alguma vez na vida, você já fumou cigarro, mesmo uma ou duas tragadas?	Numérico

Nome da coluna/campo	Nome original no dataset PeNSE	Descrição / perguntas do questionário	Tipo
JA_NARGUILE	B04013	Alguma vez na vida você já experimentou narguilé (cachimbo de água)?	Numérico
JA_CIGARRO_ELETRONICO	B04014	Alguma vez na vida você já experimentou cigarro eletrônico (e-cigarrette)?	Numérico
JA_USOU_TABACO	B04015	Alguma vez na vida você já experimentou outros produtos do tabaco, SEM CONTAR narguilé e cigarro eletrônico?	Numérico
RESPONSAVEL_FUMA	B04006b	Sua mãe, pai ou responsável fuma?	Numérico
FUMA_PROXIMO_ALUNO	B04005a	NOS ÚLTIMOS 7 DIAS, em quantos dias pessoas fumaram em sua presença na sua casa?	Numérico
AMIGO_FUMA	B04016	NOS ÚLTIMOS 30 DIAS, algum dos seus amigos fumou na sua presença?	Numérico

Além do dataset principal, também foram utilizado um dataset de IDH municipal de 2010 para que pudesse acrescentar informações ao dataset anterior, desta forma enriquecendo os dados e auxiliando no treinamento e detecção propostos por este trabalho. Este dataset possui 6 colunas e 5.565 linhas e carrega informações sobre os índices de desenvolvimento humano nos municípios brasileiros. A obtenção deste dataset foi através do portal eletrônico https://www.undp.org/pt/brazil/idhm-munic%C3%ADpios-2010 localizado no site do Programa das Nações Unidas para o Desenvolvimento (PNUD, 2022).

A Tabela 2 mostra as colunas do dataset de IDH, no entanto também é importante ressaltar que apenas o IDHM de educação foi utilizado para enriquecimento do dataset principal. Os dados de IDH foram obtidos através de uma tabela no site indicado acima e transformado em CSV para ser tratado e manuseado no arquivo prático do trabalho de conclusão.

Tabela 2: Colunas dataset de IDH dos municípios

Nome da coluna/campo	Descrição	Inclusão no dataset principal	Tipo
Ranking IDHM 2010	Posição ordenada dos municípios com maiores IDHs do Brasil	Não	Numérico
Município	Nome do município analisado	Não	Numérico
IDHM 2010	IDH geral baseado no IBGE 2010	Não	Numérico
IDHM Renda 2010	IDH de renda no município basea-do no IBGE 2010	Não	Numérico
IDHM Longevidade 2010	IDH de longevidade da população baseado no IBGE 2010	Não	Numérico
IDHM Educação 2010	IDH de educação baseado no IBGE 2010	sim	Numérico

Por fim, o dataset final que foi utilizado para o a realização do trabalho, possui 16 colunas ao total, onde quinze destas são preenchidas por dados retirados do dataset principal do PeNSE e uma coluna do IDH educacional dos municípios estudados, sendo preenchida pela base de IDH de 2010 obitido através da base do UNDP Brasil. Os tratamentos realizados em cada uma das bases e a inclusão dos dados de enriquecimento serão tratados no próximo tópico.

3. Processamento/Tratamento de Dados

3.1 Importação das bibliotecas

Inicialmente para que fosse possível todo o tratamento da base, a fim de deixa-la ideal para seguir com os treinamentos e exploração dos dados, foi necessária a importação de bibliotecas que ajudam e facilitam toda manipulação de dados e de gráficos. Desta forma, para este primeiro tópico, foram instaladas as bibliotecas Plotly, utilizada para visualizações gráficas no Python e a biblioteca Unidecode, que será utilizada para os tratamentos de padronização de strings nas bases de dados analisadas, estas instalações foram necessárias, pois o projeto não possuía estas bibliotecas especificamente na área de desenvolvimento. O comando para esta instalação foi: *!pip instal "nome da biblioteca".*

Após a instalação destas duas principais bibliotecas, foram feitas também outras importações de bibliotecas que já existiam no ambiente de desenvolvimento, sendo desta

forma importadas através do comando *import nome da biblioteca as "nickname da biblioteca"*, de livre escolha. A Figura 1 mostra as bibliotecas que foram importadas para o tratamento de dados e posteriormente para a exploração dos dados também.

```
import pandas as pd
import json
from pandas import json_normalize
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px
import unidecode
import unicodedata
```

Figura 1: Bibliotecas importadas para o tratamento e exploração dos dados

3.2 Importação e tratamento da base de dados principal

Após a importação das bibliotecas, foi iniciado a importação da base de dados PeNSE 2019, que possui todos os dados sociais de alunos que participaram da pesquisa realizada pelo IBGE, juntamente com os ministérios da saúde e da educação. Como citado anteriormente, esta base de dados possui 306 colunas e mais de 165 mil linhas, no entanto as colunas que serão tratadas neste trabalho serão apenas 15 destas.

O primeiro tratamento utilizado para a base de dados principal, foi a substituição dos nomes das colunas que serão utilizadas no trabalho, por nomes de fácil identificação, pois conforme mostrado na tabela 1, a maioria das colunas indicadas nesta base foram nomeadas com uma ordem numérica, que pode ter seu objetivo identificado através do dicionário dos dados. A Figura 2 mostra o comando para renomear todos os campos que são de interesse para este trabalho.

```
base.rename(columns={'DEP_ADMIN':'TIPO_ESCOLA','B01001A':'SEXO','B01003':'IDADE','B01002':'ETINIA','B01021A':'ANO_ESCOLAR'}, inplace = True)
base.rename(columns={'B01006':'MORA_MAE','B01007':'MORA_PAI','B04001':'JA_FUMOU','B04013':'JA_NARGUILE','B04014':'JA_CIGARRO_ELETRONICO'}, inplace = True)
base.rename(columns={'B04015':'JA_USOU_TABACO','B04006B':'RESPONSAVEL_FUMA','B04005A':'FUMA_PROXIMO_ALUNO','B04016':'AMIGO_FUMA'}, inplace = True)
```

Figura 2: Comando para renomear colunas importantes para o trabalho

Estas colunas foram escolhidas como fundamentais para o trabalho em detrimento de outras, pois elas trazem inicialmente dados sociais do aluno, como tipo da escola, sexo, idade, etnia, ano escolar e se aluno mora com os pais. Estas colunas foram consideradas como dados que podem influenciar no início dos hábitos negativos à saúde, seja por influência do meio educacional, da idade ou sexo, seja pela etnia indicando uma tendência maior em pessoas de uma mesma raça ou seja pela influência de hábitos dos pais.

Para uma segunda análise, foram selecionadas colunas do setor de tabagismo da pesquisa, estas colunas são se o aluno já experimentou ou não cigarros, se o aluno já experimentou narguilé ou cigarro eletrônico, se já usou tabaco de outras formas além de cigarros, se o responsável pelo aluno tem o hábito de fumar e se responsáveis ou amigo próximo fumou perto do aluno nos últimos 7 ou 30 dias, respectivamente. Estes dados foram considerados importantes, pois tratam da influência familiar ou de amizades no desenvolvimento do habito de fumar de uma pessoa, além de também trazer informações sobre a experimentação de outros tipos de "fumo" e qual a influência deles na probabilidade de o aluno ter ou não experimentado cigarro de tabaco.

O outro tratamento que também foi realizado para um filtro no banco de dados principal, foi a limitação da região e dos municípios analisados. Para as colunas da base chamadas de REGIAO e MUNICIPIO_CAP, os filtros aplicados foram de REGIAO somente para o valor 3(três) e MUNICIPIO_CAP somente para os valores diferentes de zero. Estes valores significam segundo o dicionário de dados, respectivamente, que a base será analisada apenas na região sudeste e para municípios que são capitais dos estados desta região, ou seja, será considerado na pesquisa apenas as cidades de Belo Horizonte, Rio de Janeiro, São Paulo e Vitória.

Outro ponto também, é sobre a desistência ou falta de resposta na pergunta principal deste trabalho, se o aluno já fumou ou não. Para que pudesse ser entendido a quantidade de desistência dentro da base, já com os filtros de região, foi aplicado um comando de verificação da quantidade de dados para cada possível resposta para esta coluna. Desta forma verificou-se que para abandono de questionário (valor -2), pulo do questionário (valor -1) e sem resposta (valor 9), no total de respostas somavam uma quantidade de alunos de 6, 2.395 e 22 respectivamente. As respostas de 1 e 2 como sim ou não possuem um somatório

total de 2.771 e 8.831, respectivamente. Visto que a quantidade de alunos que responderam à esta pergunta com sim ou não somam uma quantidade significativa e satisfatória para este trabalho, todas as outras repostas possíveis para esta pergunta foram filtradas e retiradas da base de dados.

Por fim, com todos os filtros realizados conforme foi descrito acima, foram retiradas todas as outras colunas que não fazem parte das citadas anteriormente como essenciais para a pesquisa. Restando assim um total de 15 colunas. É importante ressaltar que o modelo dos dados da pesquisa feita pelo PeNSE possui apenas respostas em formato numérico, mesmo que sua interpretação seja de variáveis categóricas, sendo possível interpretá-las através do dicionário de dados. Desta forma, para que o entendimento da base fosse mais claro, foi necessária a transformação destas variáveis categóricas que antes estavam em formato numérico em formato descritivo, utilizando o dicionário de dados como suporte para esta transformação. Sendo assim, quatorze funções diferentes foram criadas para que se pudesse transformar estes dados, deixando de fora da transformação apenas a coluna JÁ_FUMOU, mantendo assim os valores de 1 ou 2 para as respostas dos alunos. Na seção Apêndice, são encontradas as funções de transformação e na Figura 3 é possível verificar a implementação das funções para cada coluna analisada.

```
base['MUNICIPIO_CAP'] = base['MUNICIPIO_CAP'].apply(lambda x : ajuste_municipio(x))
base['TIPO_ESCOLA'] = base['TIPO_ESCOLA'].apply(lambda x : ajuste_escola(x))
base['SEXO'] = base['SEXO'].apply(lambda x : ajuste_sexo(x))
base['IDADE'] = base['IDADE'].apply(lambda x : ajuste_idade(x))
base['ETINIA'] = base['ETINIA'].apply(lambda x : ajuste_raca(x))
base['ANO_ESCOLAR'] = base['ANO_ESCOLAR'].apply(lambda x : ajuste_ensino_medio(x))
base['MORA_MAE'] = base['MORA_MAE'].apply(lambda x : ajuste_mora_mae(x))
base['MORA_PAI'] = base['MORA_PAI'].apply(lambda x : ajuste_mora_pai(x))
base['JA_NARGUILE'] = base['JA_NARGUILE'].apply(lambda x : ajuste_narguile(x))
base['JA_CIGARRO_ELETRONICO'] = base['JA_CIGARRO_ELETRONICO'].apply(lambda x : ajuste_cigarro_eletronico(x))
base['RESPONSAVEL_FUMA'] = base['RESPONSAVEL_FUMA'].apply(lambda x : ajuste_responsavel_fuma(x))
base['FUMA_PROXIMO_ALUNO'] = base['FUMA_PROXIMO_ALUNO'].apply(lambda x : ajuste_fumar_perto_7dias(x))
base['JA_USOU_TABACO'] = base['JA_USOU_TABACO'].apply(lambda x : ajuste_consumiu_tabaco(x))
base['AMIGO_FUMA'] = base['AMIGO_FUMA'].apply(lambda x : ajuste_amigo_fuma(x))
```

Figura 3: Implementação da transformação dos dados de numéricos para descritivos

A escolha de não transformar, por enquanto, a coluna alvo foi para que ao salvar a base para ser utilizada no treinamento de Machine Learning, a coluna alvo já estivesse em formato numérico e não acontecesse nenhum ajuste indesejado na base ao ser utilizada a função de dummies, que será tratada na seção 5. Para finalização do tratamento da base principal, após a transformação dos dados numéricos categóricos para dados descritivos,

observou-se um índice recorrente de dados sem resposta, isto porque os alunos podem ter pulado a pergunta do questionário ou não respondido satisfatoriamente. Desta forma, as colunas SEXO, IDADE, ETNIA, ANO_ESCOLAR, JA_NARGUILE, JA_CIGARRO_ELETRONICO, JA_USOU_TABACO, RESPONSAVEL_FUMA, FUMA_PROXIMO_ALUNO e AMIGO_FUMA que possuem dados representados como “sem Resposta” tiveram as linhas que possuíam dados neste formato excluídas. A decisão de excluir as linhas, e não preenche-las possivelmente com média ou a moda da coluna, foi tomada devido à quantidade de dados para o treinamento dos modelos de machine learning era satisfatória e se enquadrava nas exigências do desenvolvimento deste trabalho. A Figura 4 mostra o trecho do código que foi necessário para remoção dessas linhas. Sendo assim, a base final, que antes possuía 11.602 linhas, após as remoções possui 11.390 linhas finais.

```
base.drop(base[base['SEXO'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['IDADE'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['ETNIA'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['ANO_ESCOLAR'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['JA_NARGUILE'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['JA_CIGARRO_ELETRONICO'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['JA_USOU_TABACO'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['RESPONSAVEL_FUMA'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['FUMA_PROXIMO_ALUNO'] == "Sem Resposta"].index, axis=0, inplace=True)
base.drop(base[base['AMIGO_FUMA'] == "Sem Resposta"].index, axis=0, inplace=True)
```

Figura 4: Trecho de código para remoção das linhas em que não há resposta para colunas específicas

3.3 Importação e tratamento da base de dados de enriquecimento

Para enriquecimento da base principal, foi utilizada uma base de dados de IDH que possui o índice de desenvolvimento humano de todas as cidades do país. No entanto, a base de IDH possui a descrição das cidades com acentuações e diferenciação de letras maiúsculas e minúsculas, desta forma uma equação foi utilizada para padronizar a descrição das cidades a fim de possibilitar a comparação com os municípios da base principal. A função para esta padronização está listada como equação 15 nos Apêndices.

Outro tratamento realizado foi a separação da coluna Município em duas colunas, a coluna CIDADE e a COLUNA ESTADO no dataset, pois na coluna município deste dataset as cidades são acompanhadas pela sigla do estado entre parênteses o que prejudicaria o processo de enriquecimento da base principal. Desta forma, foi utilizada uma função *split* a fim

de separar a sigla do estado e colocá-la na coluna ESTADO o restante ser direcionado à coluna CIDADE. A Figura 5 mostra o comando de criação das duas colunas, juntamente com a separação do conteúdo da coluna município. A Figura 6 é um exemplo de como ficou a base de dados após o tratamento.

```
base_IDH[['CIDADE', 'ESTADO']] = base_IDH['Município'].str.split('(', expand=True, n=1)
base_IDH.head(10)
```

Figura 5: Separação do nome da cidade e a sigla do estado

	Ranking IDHM 2010	Município	IDHM 2010	IDHM\nRenda\n2010	IDHM Longevidade 2010	IDHM Educação 2010	CIDADE	ESTADO
0	1 º	SAO CAETANO DO SUL (SP)	0,862	0,891	0,887	0,811	SAO CAETANO DO SUL	SP)
1	2 º	AGUAS DE SAO PEDRO (SP)	0,854	0,849	0,89	0,825	AGUAS DE SAO PEDRO	SP)
2	3 º	FLORIANOPOLIS (SC)	0,847	0,87	0,873	0,8	FLORIANOPOLIS	SC)
3	4 º	BALNEARIO CAMBORIU (SC)	0,845	0,854	0,894	0,789	BALNEARIO CAMBORIU	SC)
4	4 º	VITORIA (ES)	0,845	0,876	0,855	0,805	VITORIA	ES)
5	6 º	SANTOS (SP)	0,84	0,861	0,852	0,807	SANTOS	SP)
6	7 º	NITEROI (RJ)	0,837	0,887	0,854	0,773	NITEROI	RJ)
7	8 º	JOACABA (SC)	0,827	0,823	0,891	0,771	JOACABA	SC)
8	9 º	BRASILIA (DF)	0,824	0,863	0,873	0,742	BRASILIA	DF)
9	10 º	CURITIBA (PR)	0,823	0,85	0,855	0,768	CURITIBA	PR)

Figura 6: Exemplo da base de dados após o tratamento

Após o tratamento da base, foram considerados apenas duas colunas relevantes para o enriquecimento da base principal. As colunas relevantes para este caso são a de CIDADE, para que seja possível comparar com a coluna MUNICIPIO_CAP da base principal, e a coluna IDHM Educação 2010, para que se possa enriquecer o dataset principal com os valores de IDH de educação para as cidades de base de estudo.

3.4 Enriquecimento da base de dados principal

Ao finalizar o tratamento da base principal e da base de IDH, o enriquecimento será realizado através de uma função que irá percorrer toda a base de IDH comparando se o valor passado para a função era igual ao valor da coluna CIDADE, caso seja igual uma nova coluna na base principal será criada e a linha para aquela cidade será preenchida com o valor do IDH de educação analisado em 2010 para aquele município. A Equação 16 pode ser conferida na seção de Apêndice. No entanto, a base de IDH possui os valores separados por vírgula, sendo necessário já na base principal substituir a vírgula por ponto, para que o campo seja reconhecido como valor numérico.

Por fim, após a transformação dos valores de IDH em dados numéricos na base principal, a base foi finalizada sendo possível salvá-la em um novo dataset que será utilizado para a exploração dos dados e a criação dos modelos de Machine Learning. Vale ressaltar que a base final possui 11.390 linhas e não possui campos nulos, não sendo necessário preenchê-los com dados de média ou alguma outra estratégia pertinente. Também é importante verificar que os a base final, agora com 16 colunas, possui apenas 2 colunas representadas como dados numéricos, a coluna de IDH é uma variável contínua já a coluna de JA_FUMOU é uma variável categórica representada por valores numéricos (1 = sim, 2 = não). Todos os outros campos da base são variáveis categóricas.

```
Int64Index: 11390 entries, 99531 to 125065
Data columns (total 16 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   MUNICIPIO_CAP         11390 non-null  object
 1   TIPO_ESCOLA           11390 non-null  object
 2   SEXO                  11390 non-null  object
 3   IDADE                 11390 non-null  object
 4   ETNIA                 11390 non-null  object
 5   ANO_ESCOLAR           11390 non-null  object
 6   MORA_MAE              11390 non-null  object
 7   MORA_PAI              11390 non-null  object
 8   JA_FUMOU              11390 non-null  float64
 9   JA_NARGUILE           11390 non-null  object
 10  JA_CIGARRO_ELETRONICO 11390 non-null  object
 11  JA_USOU_TABACO        11390 non-null  object
 12  RESPONSAVEL_FUMA      11390 non-null  object
 13  FUMA_PROXIMO_ALUNO    11390 non-null  object
 14  AMIGO_FUMA            11390 non-null  object
 15  IDH_EDUCACAO          11390 non-null  float64
dtypes: float64(2), object(14)
```

Figura 7: Informações e tipo de cada coluna da base

```
base.isna().sum()
```

```
MUNICIPIO_CAP           0
TIPO_ESCOLA             0
SEXO                    0
IDADE                   0
ETINIA                  0
ANO_ESCOLAR             0
MORA_MAE                0
MORA_PAI                0
JA_FUMOU                0
JA_NARGUILE             0
JA_CIGARRO_ELETRONICO   0
JA_USOU_TABACO          0
RESPONSAVEL_FUMA        0
FUMA_PROXIMO_ALUNO      0
AMIGO_FUMA              0
IDH_EDUCACAO            0
dtype: int64
```

Figura 8: Verificação de campos nulos na base de dados final

4. Análise e Exploração dos Dados

Antes de iniciar a exploração dos dados, a base final foi importada a fim de ser utilizada para a identificação de outliers e da predominância dos dados sociais de alunos que já experimentaram cigarro ao menos uma vez. No entanto, conforme citado anteriormente, a representação dos dados de resposta para a pergunta do questionário de se o aluno já fumou ou não, está de forma numérica. Sendo assim, antes de iniciar a análise dos dados, uma função foi aplicada na coluna JA_FUMOU do dataset final, a fim de transformar os dados representados por 1 e 2 para a representação de sim e não. A equação de transformação destes dados está representada na seção de Apêndice como Equação 17.

4.1 Identificação de outliers

A Coluna IDH_EDUCACAO, após a transformação da coluna JÁ_FUMOU, é a única coluna com representação de dados numéricos no dataset analisado. Desta forma, para que fosse possível a identificação de outliers na coluna analisada, foi aplicado uma representação gráfica do tipo boxplot a fim de identificar se algum valor na coluna estava fora do limite admissível para este tipo de dado. A Figura 8, mostra o gráfico gerado nesta análise e conforme é possível verificar, os dados de IDH não possuem nenhuma variação inesperada, pois o range de valores representado realmente pertence aos valores dos IDHs das cidades que estão sendo analisadas neste projeto.

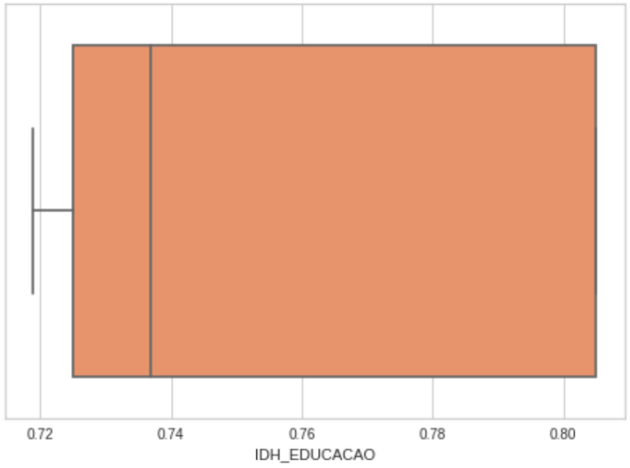

Figura 9: Boxplot para identificação de outliers na coluna de IDH_EDUCACAO da base de dados

4.2 Visualização gráfica para dados nominais

Como o restante dos dados apresentados no dataset de estudo são dados nominais, o melhor tipo de gráfico encontrado para esta representação, a fim de identificar o somatório dos dados e possíveis outliers que estão presentes nesta parte dos dados, foi o modelo de countplot, também conhecido como gráfico de barras. Para que a formatação dos gráficos fosse a mesma, a fim de manter uma harmonia entre as representações e uma padronização, uma função de formatação foi implementada. Desta forma a função recebe quatro atributos que serão necessários para a montagem do gráfico, a classe que será analisada, o título do eixo x, o título do eixo y e o título geral do gráfico. Esta equação pode ser encontrada na seção de Apêndice, sendo nomeada como Equação 18.

4.2.1 Quantidade de alunos que já experimentaram cigarros

A primeira representação implementada foi referente à classe alvo. A coluna JA_FUMOU representa o principal interesse deste projeto, pois através dela, com uma parcela dos dados da base, será possível treinar os algoritmos de aprendizado e prever a probabilidade de alunos com as mesmas características sociais experimentarem cigarros ainda na adolescência e juventude. Este gráfico mostra que para os estados da região analisada, capitais do sudeste brasileiro, assim como utilizando somente alunos que participaram desta parte da pesquisa, os dados de alunos nunca experimentaram cigarros superam os dados de alunos que já experimentaram ao menos uma vez, sendo estes cerca de 76% dos dados totais analisados.

No entanto, os dados para este caso ainda são alarmantes, pois como citado na introdução deste projeto, nos últimos anos a porcentagem de jovens menores de 18 anos que fumam é cerca de 7%, enquanto para a amostra analisada neste projeto, a porcentagem de experimentação de cigarros em jovens menos de idade foi de cerca de 23%. A Figura 10 mostra o resultado gráfico para a coluna de respostas à experimentação ou não de cigarros pelos alunos que participaram da pesquisa.

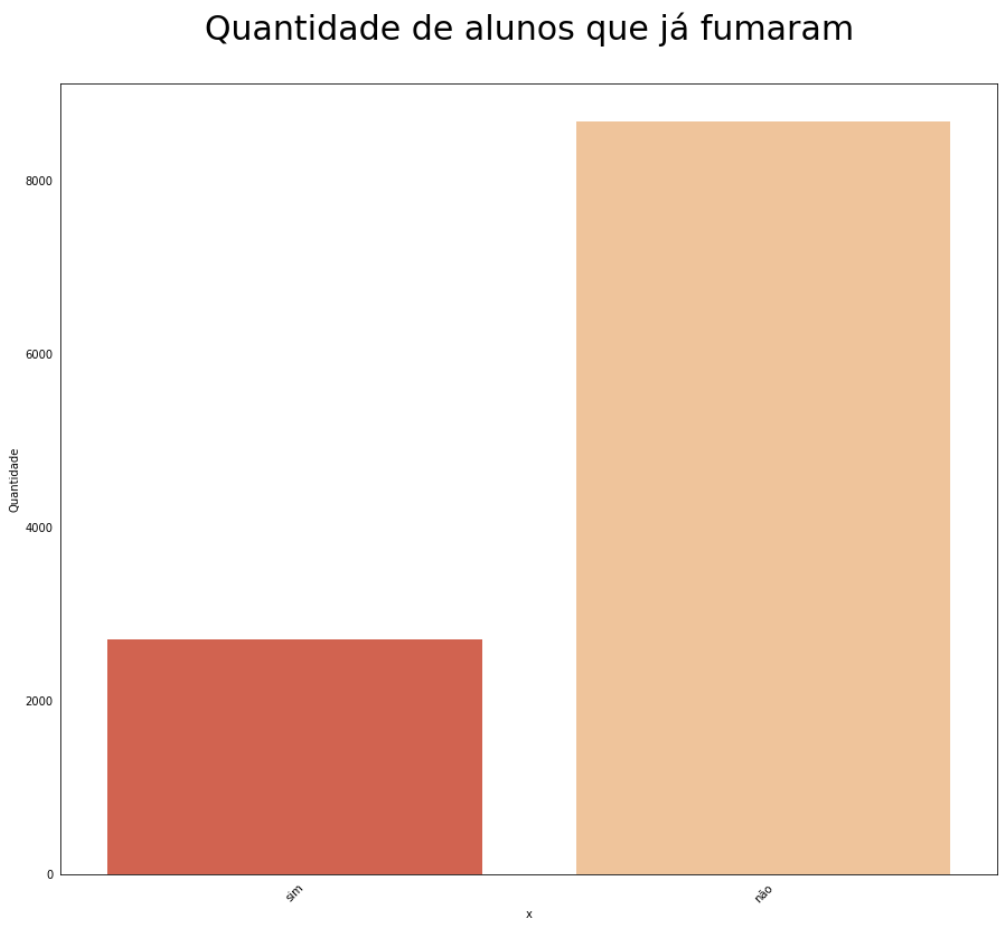

Figura 10: Representação da quantidade de alunos que já experimentaram ou não cigarros ao menos uma vez

4.2.2 Representação gráfica de outros dados da base

A segunda representação é referente à idade dos alunos analisados. O questionário para esta pergunta permitia que os alunos preenchessem a resposta que fizesse parte de um intervalo de idades. No caso dos filtros aplicados a predominância dos alunos estão na faixa etária entre 13 e 15 anos, representando mais de sete mil alunos nestas idades. Seguidas por 16 e 17 anos e 18 anos ou mais, sendo a menor representatividade nesta análise. A seguir está a Figura 11, indicando de forma gráfica os dados apresentados por esta coluna.

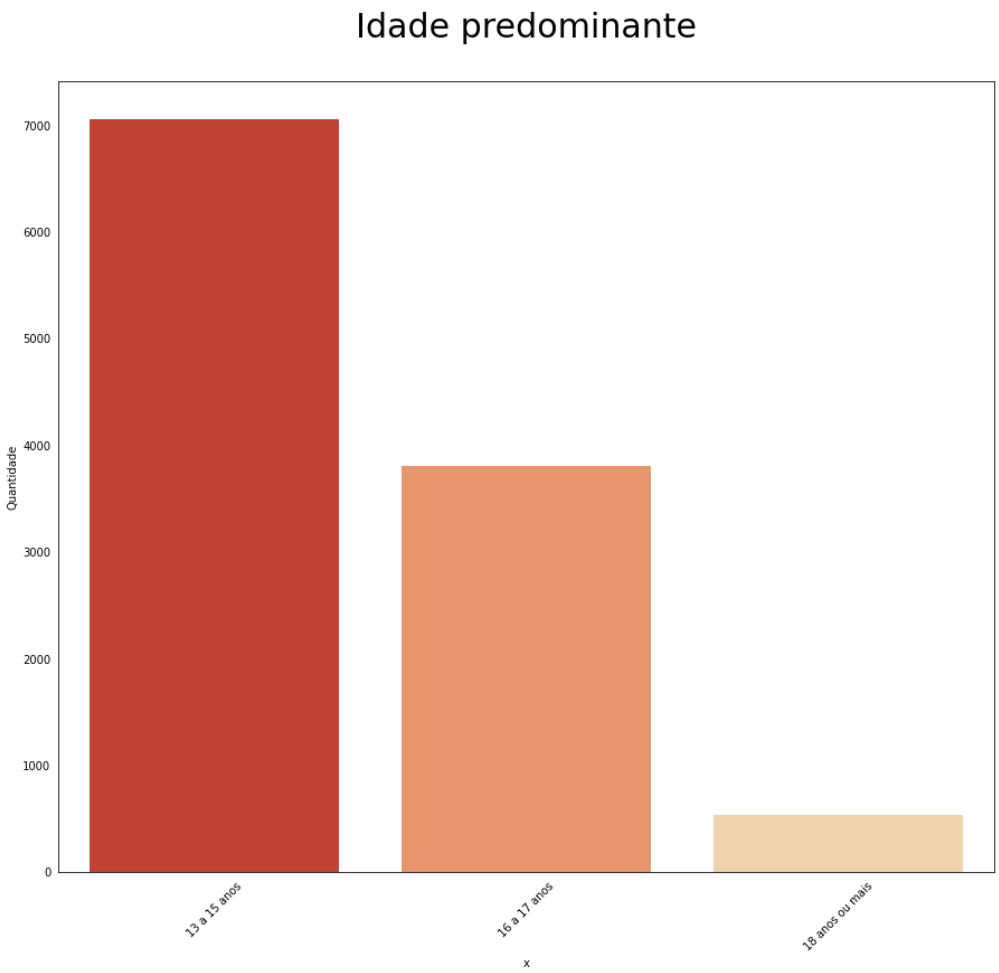

Figura 11: Idade predominante dos alunos que participaram da pesquisa

Outra representação também analisada foi o sexo dos alunos analisados. Para a amostragem proposta os gêneros masculino e feminino são basicamente equivalentes, sendo o primeiro pouco predominante. A representação desta classe está na seção de Apêndices indicada por Figura 34. A etnia entre os alunos também foi considerada como um ponto importante para a pesquisa, pois é notório que no Brasil etnias mais à margem da sociedade tendem a ter hábitos de saúde, educação e desenvolvimento piores. Para a amostragem analisada, maioria dos alunos de identificaram como sendo brancos, seguidos pele parda e preta respectivamente, dado que é coerente com a pesquisa realizada pelo (Adjuto, 2022), em que no sudeste brasileiro a maioria dos habitantes se consideram brancos, seguidos de pardos e pretos. A Figura 12 representa o somatório de estudantes que identificaram com alguma etnia apresentada.

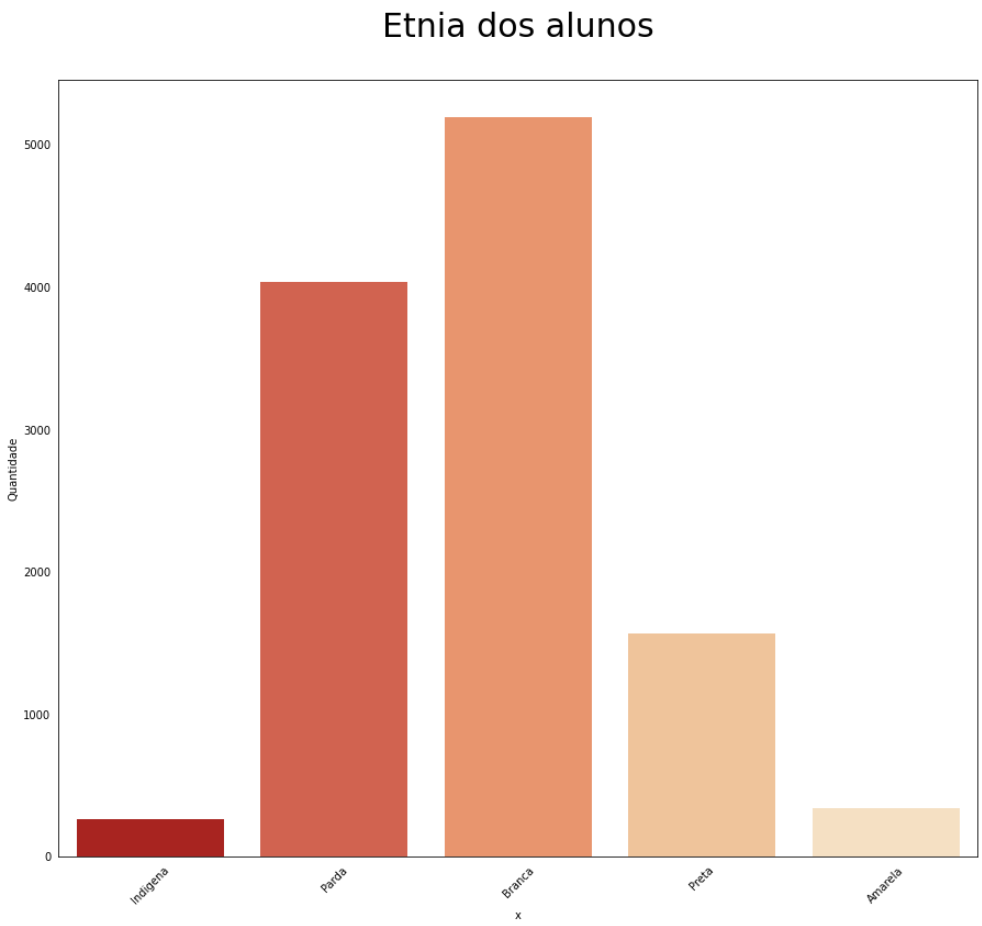

Figura 12: Etnia autodeclarada pelos alunos que participam da amostragem analisada

Um outro gráfico que também foi analisado foi o do ano escolar predominante entre os alunos que participam da amostragem. A predominância fica entre os alunos de 9º ano, seguido pelo 1º ano do ensino médio e posteriormente pelo 8º ano do ensino fundamental, o menor índice de alunos que participam da pesquisa e responderam as questões de tabagismo estão localizados no 6º ano.

Entrando agora nos dados sobre tabagismo, dois gráficos foram gerados para identificar a quantidade total de pessoas próximas ao aluno que fumam ou fumaram. Na Figura 15 e 16 são mostrados o resultado para as perguntas se o responsável pelo aluno fuma e se algum amigo próximo ao aluno fumou perto dele nos últimos 30 dias. Em ambos os gráficos um fato é possível ser identificado, para pais ou responsáveis fumantes a resposta de nenhum deles tem cerca de nove mil repetições, enquanto respostas para só o pai ou responsável masculino, ou só a mãe ou responsável feminino ou os dois somam cerca de dois mil resultados.

Da mesma forma, observa-se as respostas para um amigo próximo que fumou na frente do aluno analisado, as respostas para não são predominantes e possuem cerca de oito mil registros, enquanto para sim equivale a um pouco mais de três mil. O Interessante nessa última análise é que o somatório de alunos que nunca fumaram é muito próximo ao somatório de pais ou responsável que nunca fumaram e amigos que não fumaram. O contrário, consequentemente acontece, o somatório de alunos que já experimentaram cigarros está muito próximo à alunos que ao menos um dos responsáveis fuma e ao somatório de respostas “sim” para amigos que fumaram próximo ao aluno analisado, nos últimos 30 dias. Desta forma, pode-se perceber que a influência, seja familiar ou de amizades, se mostra um ponto importante para alunos que experimentam cigarros ao menos uma vez durante a adolescência e juventude. No entanto, será possível analisar melhor estes dados com as representações de mapa de calor que será abordada no próximo tópico.

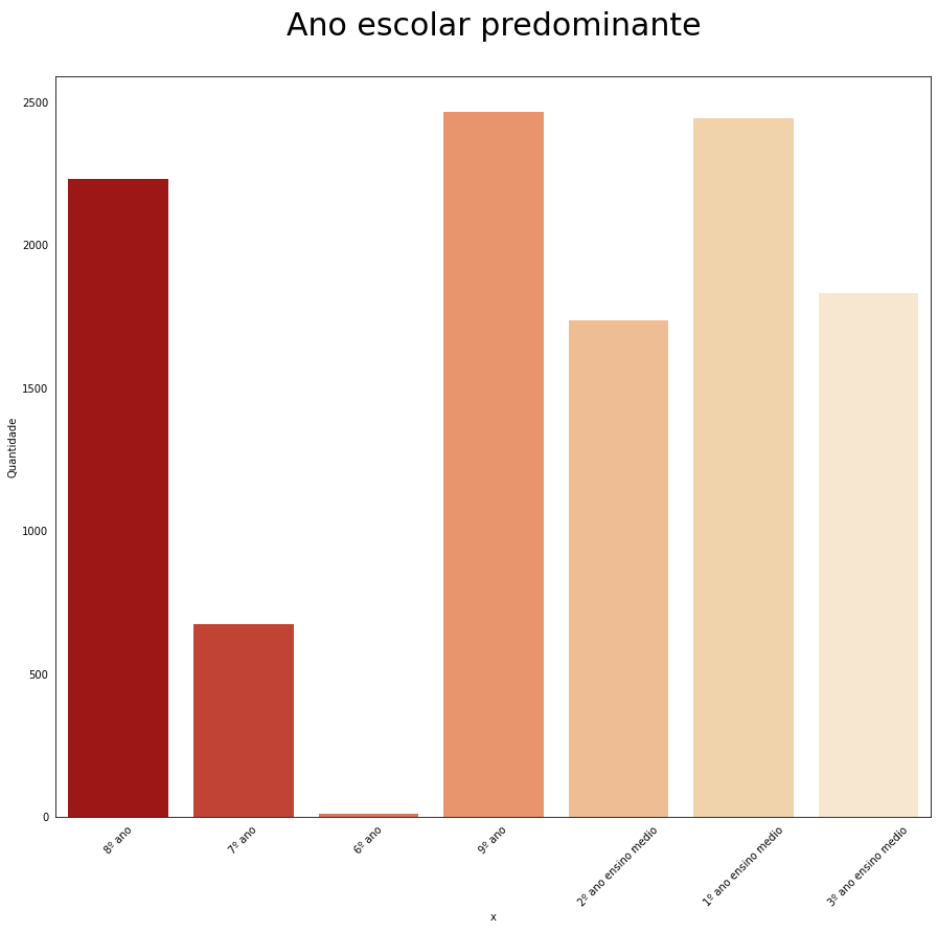

Figura 13: Ano escolar predominante dos alunos analisados

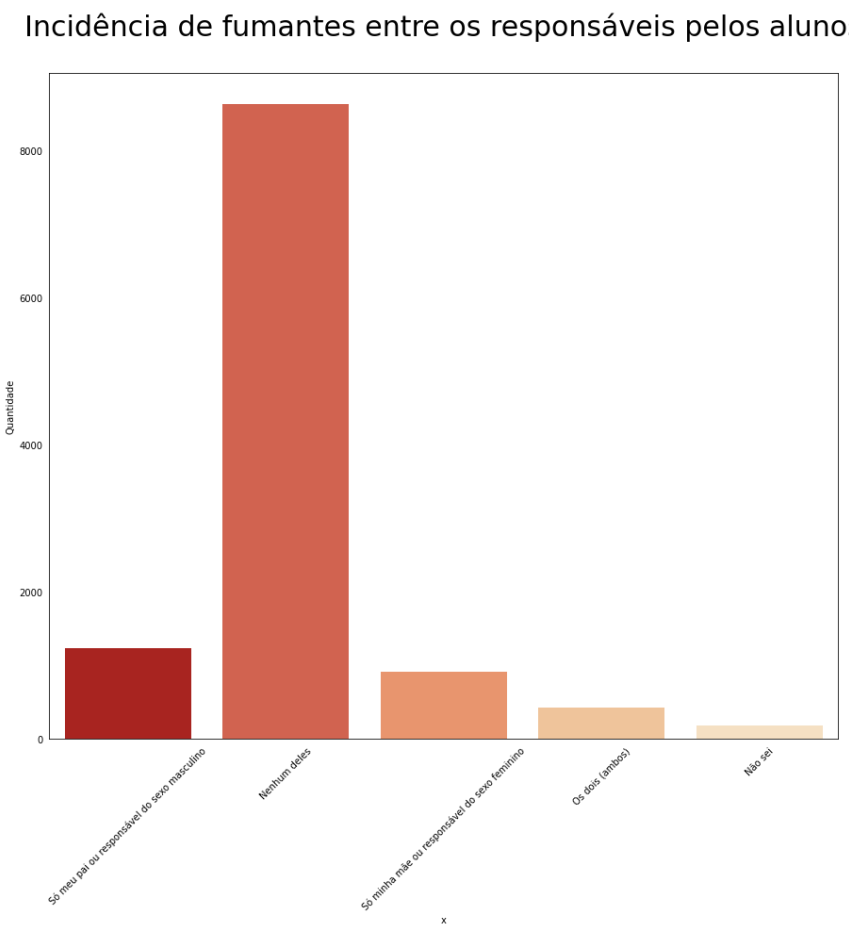

Figura 14: Incidência de pais e responsáveis fumantes

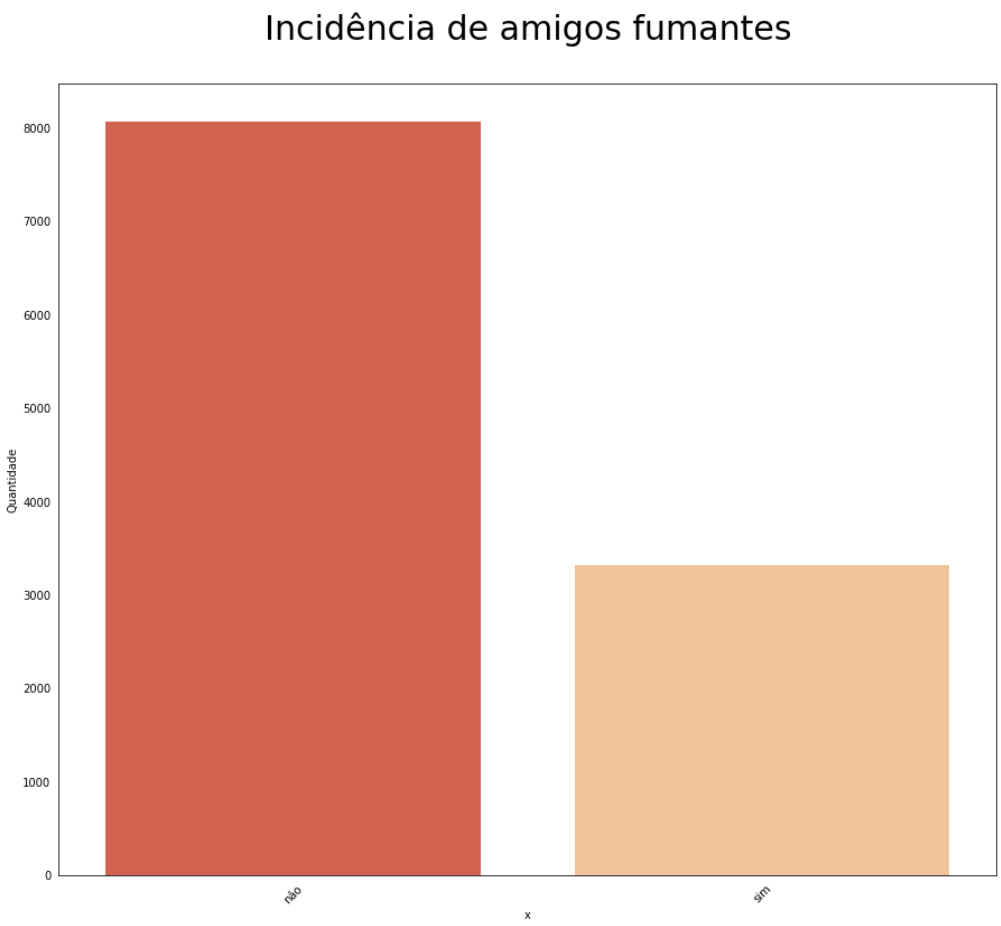

Figura 15: Incidência de amigos que fumaram perto do aluno analisado nos últimos 30 dias

4.3 Mapas de calor e combinações de dados do conjunto

Assim como foi feita uma função para padronização dos gráficos de barras, para a representação dos mapas de calor também foi utilizada uma padronização de formatação, a Equação 19 na seção de Apêndice mostra a implementação desta função. Descrevendo brevemente a implementação, a função tem o objetivo de receber duas colunas do dataset, a principal e a comparativa, assim como os títulos do eixo x e do eixo y, por fim também é passado o título do gráfico. Após receber estes valores a função irá formatar o gráfico heatmap, indicando qual coluna é a principal e qual é a comparativa. Após a instanciação do gráfico, toda a parte de formatação é iniciada, indicando os nomes dos eixos, o tamanho das letras, a posição da escrita e o tamanho da figura que será gerada.

A análise do mapa de calor foi dividida em dez gráficos diferentes que foram analisados individualmente no arquivo do código python. No entanto, apenas três destes gráficos serão analisados durante o decorrer deste tópico, que são as relações entre JA_FUMOU e ETNIA, JA_FUMOU e RESPONSAVEL_FUMA e JA_FUMOU e AMIGO_FUMA, os outros gráficos poderão ser encontrados na seção de Apêndice a partir do tópico 4.1.

O primeiro mapa de calor que será analisado é a relação entre alunos que já experimentaram cigarros e a raça autodeclarada destes alunos. No tópico anterior podemos perceber que as etnias principais declarada pelos alunos foram de branca, parda e preta respectivamente, desta forma, segue a Figura 16 com a relação entre alunos que já experimentaram cigarros e a etnia declarada por eles.

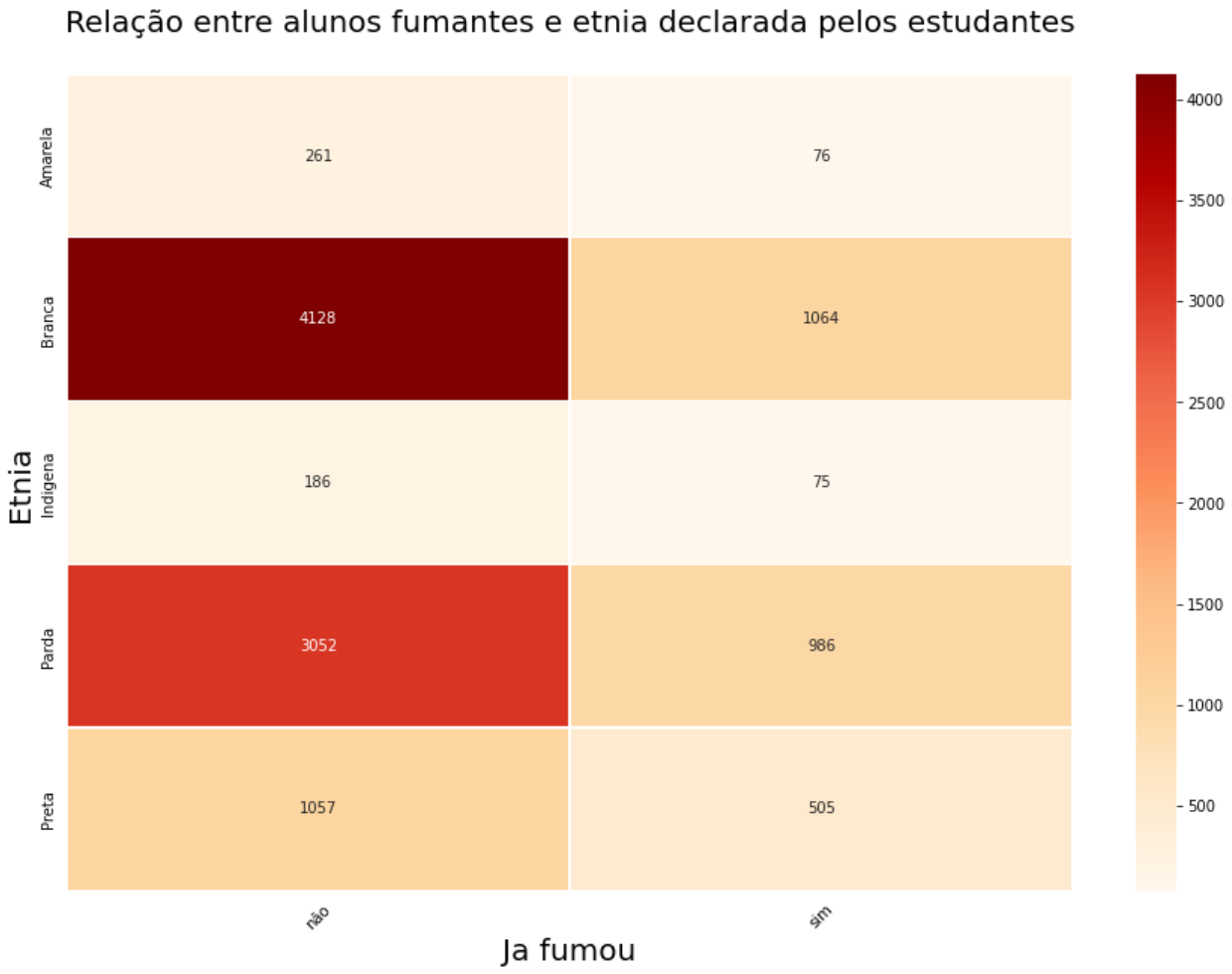

Figura 16: Mapa de calor entre alunos que já experimentaram cigarros e etnia autodeclarada

Neste gráfico é possível identificar que consequentemente o maior número de estudantes que já experimentaram ou não cigarros, estão na ordem de brancos, pardos e

pretos. No entanto, um ponto é importante ser ressaltado no total de alunos brancos que responderam a pesquisa cerca de 20,5% dos alunos já experimentaram cigarros ao menos uma única vez o mesmo se repete para os alunos que se autodeclaram pardos, cerca de 24% destes já experimentaram cigarros ao menos uma vez. No entanto, para alunos que se autodeclaram pretos a porcentagem sobe para cerca de 32,33% de alunos que já experimentaram cigarros. Desta forma, uma reflexão importante para este gráfico é a porcentagem crescente de alunos que já fumaram em etnias que estão mais à margem da sociedade brasileira.

Outro mapa de calor importante para este estudo é a relação entre responsável que fuma e se o aluno já experimentou cigarros ao menos uma vez. Para analisar este ponto, segue a Figura 17 que indica, como esperado, maiores volumes de alunos que não fumam para responsáveis que também não fumam. Um ponto importante a se observar neste mapa é a relação entre alunos que já fumaram ou não em relação ao total de alunos para as repostas de responsáveis que não fumam, somente o responsável masculino fuma ou somente o responsável feminino fuma. Para responsáveis que não fumam, um total de 8.625 alunos responderam ao questionário, sendo que cerca de 20% destes alunos já experimentaram cigarros. No entanto quando analisamos as respostas de responsável masculino fuma e responsável feminino fuma as porcentagens de alunos que já fumaram sobe para em torno de 32% e 37% respectivamente, mostrando que a influência da mãe ou responsável feminina fumar é a mais expressiva nas respostas. Por fim analisando alunos que possuem os dois responsáveis fumantes é possível chegar a conclusão que destes 41,6% já experimentaram cigarros ao menos uma vez, o que reforça a importância da influência familiar no consumo de cigarros por adolescentes e jovens.

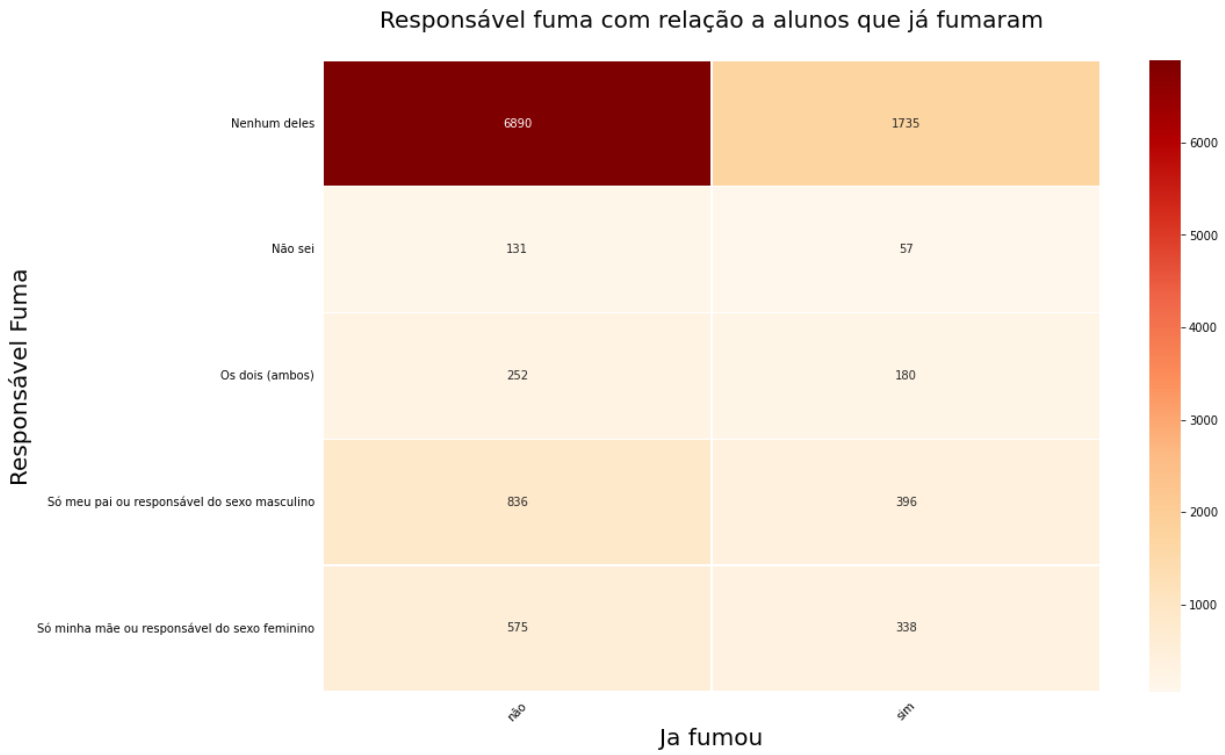

Figura 17: Mapa de calor entre responsável fumante e alunos que já experimentaram cigarros

O terceiro e último gráfico que será analisado é a influência da amizade no consumo de cigarros pelos alunos que participaram da pesquisa. Como esperado, a maioria dos alunos não possuem amigos que fumaram perto deles nos últimos 30 dias. No entanto, assim como nos gráficos anteriores as porcentagens analisadas mostram uma realidade diferente. Para alunos que não possuem amigos que fumam, cerca de 13% já experimentaram cigarros, porém a porcentagem muda significativamente quando analisamos alunos que responderam “sim” para amigos que fumam, pois destes cerca de 48,72% também já experimentaram cigarros ao menos uma vez, indicando assim a importância da influência da amizade nos hábitos de tabagismo.

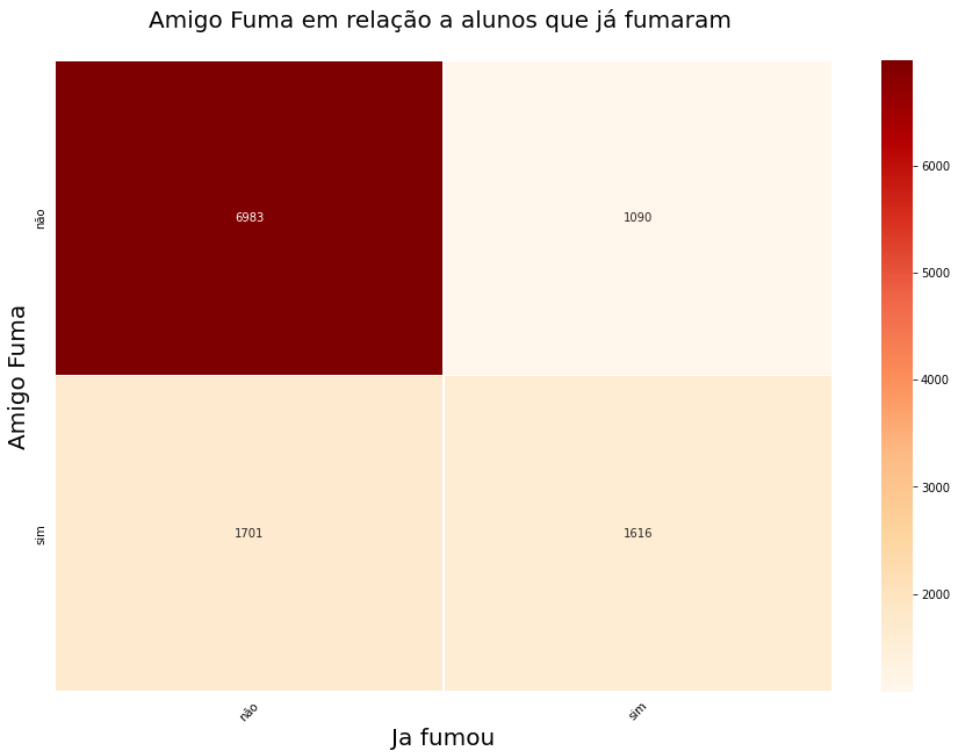

Figura 18: Mapa de calor de amigos que fuma em relação a alunos que já experimentaram cigarros

5. Criação de Modelos de Machine Learning

Após a análise dos dados e as informações que conseguimos retirar deles através da visualização de gráficos, foi analisado o quanto as características e situações sociais influenciam na probabilidade de o aluno experimentar ou não cigarros durante a adolescência e juventude. Desta forma, três modelos de aprendizado de máquina foram utilizados, sendo eles Randon Forest, árvore de decisão e Redes neurais. Estes algoritmos serão discutidos mais adiante, juntamente com a manipulação de processamento dos dados para execução dos modelos.

5.1 Importação das bibliotecas

Para esta parte do projeto um total de 11 bibliotecas foram importadas e aplicadas no código. A Figura 19 mostra estas importações que tem como principal objetivo fornece-rem métodos capazes de realizar o treinamento dos modelos, a representação das acurácias, a representação da matriz de confusão, a utilização do método de tunning para identificação

dos melhores parâmetros e a utilização de cross validation para identificação da melhor separação de treinamento e teste da base.

```
from sklearn.preprocessing import MinMaxScaler
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score, classification_report
from sklearn.model_selection import train_test_split, cross_val_score, KFold
from sklearn import tree
from sklearn.tree import DecisionTreeClassifier
from sklearn.neural_network import MLPClassifier
from yellowbrick.classifier import ConfusionMatrix
from imblearn.under_sampling import NearMiss
from sklearn.model_selection import GridSearchCV
from sklearn.model_selection import cross_val_score, KFold
```

Figura 19: Bibliotecas importantes para criação dos modelos de machine learning

5.2 Processamento dos dados para criação dos modelos de aprendizado de máquina

Após a importação das bibliotecas a base foi novamente importada para o ambiente de desenvolvimento e a retirada da coluna MUNICIPIO_CAP foi realizada. Logo após este tratamento, todos os dados foram transformados em dados numéricos novamente. Neste passo do processamento foi utilizado um método da biblioteca pandas chamado de *get_dummies* e a base importada é passada como parâmetro para este método. Neste mesmo método, a transformação os dados categóricos em numéricos não influencia na preferência de alguns valores em relação a outros, sendo assim o treinamento dos modelos não fica comprometido. Finalizado o processo de transformação dos dados, a classe alvo e os dados de entrada foram separados em variáveis chamadas de X, para os dados de entrada, e y, para a classe alvo (coluna JÁ_FUMOU). Uma etapa importante também no processamento dos dados antes do treinamento dos modelos é o escalonamento dos dados, o escalonamento ocorre para que a diferença de escala dos dados não seja um fator que impacte no treinamento dos modelos que serão analisados. A Figura 20 mostra o trecho de código que separa os dados de entrada da classe alvo e escalona os dados de entrada.

```
X = base_ML.drop('JA_FUMOU', axis=1)
y = base_ML['JA_FUMOU']

scaler = MinMaxScaler()
X = scaler.fit_transform(X)
```

Figura 20: código de separação da classe alvo e dos dados de entrada seguido do escalonamento dos dados de entrada

Conforme visto anteriormente, os valores para alunos que responderam que nunca experimentaram cigarros é de 8.684, assim como alunos que já experimentaram cigarros é de aproximadamente 2.700 alunos. Desta forma, para que a quantidade predominante de alunos que nunca experimentaram cigarros não influenciasse no treinamento dos modelos foi aplicado um undersampling, fazendo com que o treinamento dos dados fosse o mais próximo do real sem sofre influência da quantidade de dados. Desta forma, a biblioteca *NearMiss* foi aplicada e ambas as respostas, tanto para sim quanto para não, possuem a mesma quantidade de alunos, totalizando neste ponto 5.412 dados no total. Desta forma, a base de treinamento possui 4.600 dados de entrada e a base de teste possui 812, sendo indicada neste caso como 15% da base total para realização de testes.

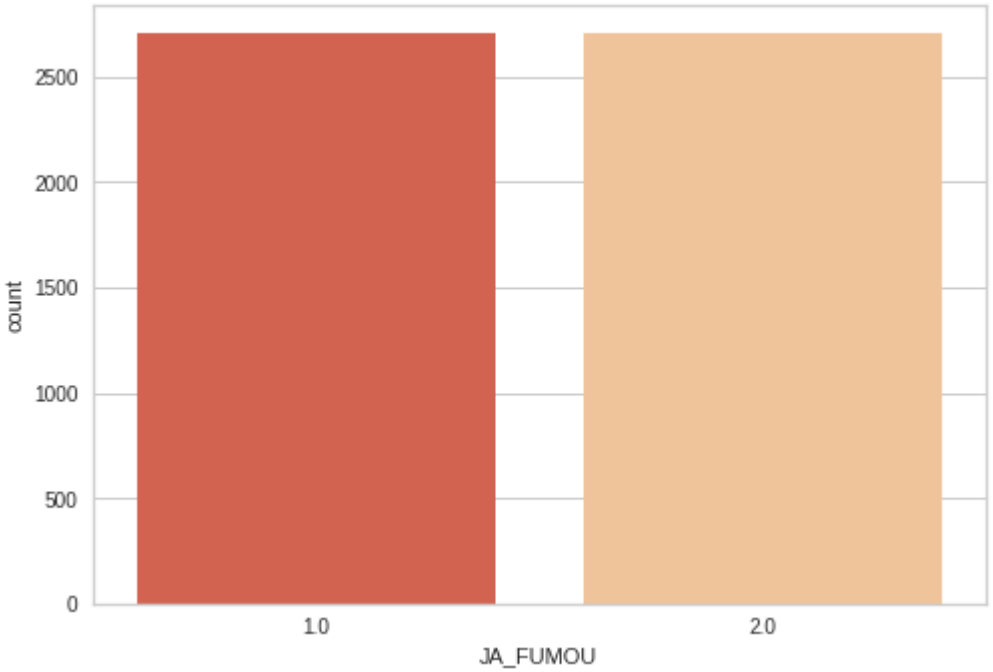

Figura 21: Dados de alunos que já experimentaram cigarros (1.0) e que nunca experimentaram cigarros (2.0) após utilização do undersampling

5.3 Tunning

Antes de criar os modelos de treinamento, foi utilizado o processo de tunning para identificação dos melhores parâmetros de treinamento para cada modelo. Para isso, foi necessário concatenar novamente as bases de treinamento e de teste, tanto dos dados de entrada quanto os dados a classe alvo, totalizando 5.412 dados após o undersampling. Cada modelo possui diferentes dados de entrada que influenciam no resultado do treinamento, desta forma nos tópicos a seguir será discutido os parâmetros para cada modelo utilizado (Skit Learn, 2023).

5.3.1 Tunning para Randon Forest

Os parâmetros utilizados para o tunning deste modelo estão listados na Tabela x, assim como a indicação da importância para o treinamento deste modelo.

Tabela 3: Parâmetros de entrada e resultado do tunning para Random Forest

Parâmetros	Descrição	Dados de entrada	Dados de saída
criterion	A função para medir a qualidade de uma divisão	gini / entropy	entropy
n_estimators	O número de árvores na floresta.	10,40,100,150	100
min_samples_split	O número mínimo de amostras necessárias para dividir um nó interno	2,5,10	2
min_samples_leaf	O número mínimo de amostras necessárias para estar em um nó folha	1,5,10	5

Como mostrado na tabela 3, os dados de saída para o tunning de Random forest é o critério de entropia, a quantidade de 100 árvores na floresta, o número de amostras para divisão de um nó folha de 5 e a quantidade de amostras para divisão de um nó interno de 2. Estes parâmetros serão utilizados posteriormente no treinamento do algoritmo para obtenção do melhor resultado do treinamento. A declaração dos parâmetros para preparação e a implementação do tunning para este algoritmo estão indicadas na Figura 22.

```
paramentros_RF = {'criterion':['gini','entropy'],
                  'n_estimators':[10,40,100,150],
                  'min_samples_split':[2,5,10],
                  'min_samples_leaf':[1,5,10]}
```

```
gridRN = GridSearchCV(estimator = RandomForestClassifier(),param_grid=paramentros_RF)
gridRN.fit(X_df,Y_df)
melhoresParametros = gridRN.best_params_
melhorResultado = gridRN.best_score_
```

Figura 22: Preparação dos parâmetros do modelo e implementação do tunning para Random Forest

5.3.2 Tunning para árvore de decisão

Assim como para Random Forest, o modelo de árvore de decisão possui parâmetros de entrada que combinados resultam no melhor resultado do treinamento. Dos quatro parâmetros que serão manipulados por tunning, três deles são os mesmos de Random Forest e apenas o n_estimators não faz parte de árvore de decisão. No entanto, um novo parâmetro chamado splitter será indicado, desta forma a Tabela 4 mostra os parâmetros de árvore de decisão que passaram por tunning, a descrição de cada parâmetro, as opções de entrada e o retorno dos parâmetros para melhor resultado.

Tabela 4: Parâmetros de entrada e resultado do tunning para árvore de decisão

Parâmetros	Descrição	Dados de entrada	Dados de saída
criterion	A função para medir a qualidade de uma divisão	gini / entropy	gini
splitter	Indica a estratégia usada para escolher a divisão em cada nó	Best/ random	Best
min_samples_split	O número mínimo de amostras necessárias para dividir um nó interno	2,5,10	5
min_samples_leaf	O número mínimo de amostras necessárias para estar em um nó folha	1,5,10	10

Da mesma forma como para Random Forest, os dados de saída do resultado de tunning serão utilizados na implementação dos treinamentos de árvore de decisão. A Figura 23

mostra a atribuição dos parâmetros e as linhas de código necessárias para o tratamento de tunning para este modelo de machine learning.

```
paramentros_DT = {'criterion':['gini','entropy'],
                  'splitter':['best','random'],
                  'min_samples_split':[2,5,10],
                  'min_samples_leaf':[1,5,10]}
```

```
gridDT = GridSearchCV(estimator = DecisionTreeClassifier(),param_grid=paramentros_DT)
gridDT.fit(X_df,Y_df)
melhoresParametros = gridDT.best_params_
melhorResultado = gridDT.best_score_
```

Figura 23: Preparação dos parâmetros do modelo e implementação do tunning para árvore de decisão

5.3.3 Tunning para Redes Neurais

O modelo de Redes Neurais, dos três modelos aplicados, é o que mais se diferencia na metodologia de treinamento e nos parâmetros de entrada. A característica principal de cada modelo será abordada em tópicos posteriores, mas é possível dizer que árvore de decisão e Random Forest são modelos mais similares, que desempenham uma metodologia parecida durante o treinamento, tendo desta forma parâmetros de entrada parecidos. Para o tratamento de tunning em redes neurais, apenas três parâmetros serão indicados e a Tabela 5 mostra exatamente quais são, a descrição de cada parâmetro, os dados de entrada e os dados de saída que posteriormente serão utilizados no treinamento do modelo. Da mesma forma, a Figura 24 também mostra a implementação dos parâmetros de entrada para redes neurais e as linhas de código que implementaram o processo de tunning.

Tabela 5: Parâmetros de entrada e resultado do tunning para redes neurais

Parâmetros	Descrição	Dados de entrada	Dados de saída
activation	Função de ativação da camada oculta	Relu/ logistic/ tahn	logistic
solver	O solucionador para otimização de peso.	Adam/ sgd	adam
batch_size	Tamanho de minilotes para otimizadores estocásticos	10 ,50	50

```
paramentros_RN = {'activation': ['relu','logistic','tahn'],
                  'solver': ['adam','sgd'],
                  'batch_size':[10,50]}

gridRN= GridSearchCV(estimator = MLPClassifier(),param_grid=paramentros_RN)
gridRN.fit(X_df, Y_df)
melhoresParametros = gridRN.best_params_
```

Figura 24: Preparação dos parâmetros do modelo e implementação do tunning para Redes Neurais

5.4 Implementação dos modelos de machine learning

Para a identificação da probabilidade de um adolescente ou jovem experimentar cigarro através das características sociais deste jovem, foram utilizados três diferentes modelos de aprendizado de máquina. O primeiro modelo, como já citado anteriormente, é o de Random Forest, este algoritmo resumidamente é um conjunto de árvores de decisões, outro modelo que será tratado posteriormente, e combina o resultado destas diversas árvores para enfim chegar em um resultado final. Este algoritmo é bastante poderoso e de simples compreensão, cada variável de entrada possui uma importância diferente e tem um impacto distinto nas predições realizadas. Uma desvantagem deste modelo é ser um algoritmo de aprendizado supervisionado, sendo possível ter como resposta apenas uma variável do conjunto de dados (ICMC JÚNIOR, 2023).

O segundo modelo utilizado é o de árvore de decisão, que como citado anteriormente é bem parecido com o Random Forest, sendo neste caso apenas uma árvore utilizada para realizar as predições necessárias. Uma árvore de decisão começa com um nó, que se divide em possíveis caminhos sendo que cada um desses resultados leva a nós adicionais, que se ramificam em outras possibilidades. Desta forma, um formato parecido com uma árvore é criado, dando o nome deste modelo e ao modelo citado no parágrafo anterior. Árvore de decisão é um algoritmo importante para quando há um problema de diversos rótulos. Ou seja, quando as categorias de classificação são múltiplas, e não apenas duas (como sim ou não), no caso deste trabalho a classe alvo é separada em apenas duas respostas, porém caso seja utilizado em outros trabalhos que possuem um maior range de respostas finais, árvore de decisão é um algoritmo que lida bem com estes tipos de soluções (Blog Somos Tera, 2023).

Por fim, o algoritmo de redes neurais possivelmente é o mais conhecido quando se fala sobre machine learning. A característica mais marcante das redes neurais, é sua estruturação semelhante à rede de neurônios em nosso cérebro. O sistema de predição é composto por vários nós que se interconectam, sendo que cada nó possui informações que serão passadas para o próximo nó e assim por

diante. Este modelo é também indicado para problemas mais complexos e de aprendizado profundo (deep learning) e é formado por três camadas principais, a camada de entrada, quando as informações externas entram na rede, sendo processadas, categorizadas e enviadas para a próxima etapa. A camada oculta é a que fica no meio do processo, podendo ser inúmeros camadas que consomem informações internas, processam ainda mais os dados e enviam para o final do fluxo. Por fim, a camada de saída é a que fornece os resultados processados através dos dados de entrada, esta camada pode ter vários nós, mas no caso deste projeto a camada de saída terá apenas um nó que será classificado como sim ou não para a experimentação de cigarros por alunos que participaram da pesquisa (Amazon, 2023).

5.4.1 Treinamento com Random Forest

Para o treinamento do modelo de Random Forest foram utilizados parâmetros indicados por tunning a fim de encontrar o melhor resultado de resposta. A Figura 25 mostra a implementação do treinamento deste modelo, que retornou um resultado de 88,67% de acurácia, ou seja, acertando conforme dados socias de entrada a possibilidade de um aluno ter ou não experimentado cigarros durante a adolescência e juventude. As previsões do modelo estão na seção de Apêndice.

```
random_forest = RandomForestClassifier(criterion='entropy', min_samples_leaf= 5, min_samples_split= 2, n_estimators= 100)
random_forest.fit(X_treinamento, y_treinamento)
```

Figura 25: Implementação do treinamento de Random Forest

5.4.2 Treinamento com Árvore de Decisão

O treinamento de árvore de decisão está representado na Figura 26, conforme indicado anteriormente os parâmetros retornados por tunning foram aplicados na chamada do método para treinamento do modelo. A acurácia para o algoritmo de árvore de decisão foi de 88,05% e da mesma forma que o modelo anterior, as previsões do modelo estão representadas na seção de Apêndice.

```
arvore_decisao = tree.DecisionTreeClassifier(criterion= 'gini', min_samples_leaf= 10, min_samples_split= 5, splitter= 'best')
```

```
arvore_decisao.fit(X, y)
arvore_decisao.score(X, y)
```

Figura 26: Implementação do treinamento de Árvore de Decisão

5.4.3 Treinamento com Redes Neurais

Da mesma forma como o outros dois algoritmos, os parâmetros indicados por tunning foram passados para o treinamento de redes neurais. Desta forma, o retorno de acurácia para este modelo foi de 88,05% e a Figura 27 mostra a implementação do treinamento, sendo que as previsões dadas pelo modelo com as entradas de teste estão indicadas na seção de Apêndices.

```
rede_neural = MLPClassifier(activation= 'logistic', batch_size= 50, solver= 'adam')
rede_neural.fit(X_treinamento, y_treinamento)
```

Figura 27: Implementação do treinamento de Redes Neurais

5.5 Validação Cruzada

Conforme já citado neste trabalho a separação entre a base de treinamento e a base de teste foram de 85% e 15% respectivamente, desta forma o conjunto de treinamento são exemplos pelos quais o modelo irá aprender e o conjunto de teste é utilizado para simular os resultados que por fim serão comparados com a classe alvo de teste. A validação cruzada, portanto, é uma metodologia que visa reformular os subconjuntos realizando os treinamentos e testes para verificar qual destes sub-conjuntos possui melhor desempenho para o treinamento do modelo. Desta forma, para cada um dos modelos já descritos anteriormente foram aplicados à validação cruzada em um range de 30 vezes, subdividindo os conjuntos e apresentando a acurácia obtida, assim como a média das acurácias. As três figuras seguintes mostram a implementação da validação cruzada para cada modelo aplicado, na seção de Apêndice as tabelas com a acurácia de cada uma das iterações dos modelos poderão ser encontradas.

```
resultadosVC_RF =[]

for i in range(30):
  kfold = KFold(n_splits=10 , shuffle=True, random_state=i)
  RF = RandomForestClassifier(criterion='entropy', min_samples_leaf= 5, min_samples_split= 2, n_estimators= 100)
  scores = cross_val_score(RF,X_df, Y_df, cv= kfold )
  resultadosVC_RF.append(scores.mean())
resultadosVC_RF
```

Figura 28: Implementação do Cross Validation para Random Forest

```
resultadosVC_arvore =[]

for i in range(30):
  kfold = KFold(n_splits=10 , shuffle=True, random_state=i)
  DT = DecisionTreeClassifier(criterion= 'gini', min_samples_leaf= 10, min_samples_split= 5, splitter= 'best')
  scores = cross_val_score(DT,X_df, Y_df, cv= kfold )
  resultadosVC_arvore.append(scores.mean())
resultadosVC_arvore
```

Figura 29: Implementação do Cross Validation para árvore de decisão

```
resultadosVC_RN =[]

for i in range(30):
  kfold = KFold(n_splits=10 , shuffle=True, random_state=i)
  RN = MLPClassifier(activation= 'logistic', batch_size= 50, solver= 'adam')
  scores = cross_val_score(RN,X_df, Y_df, cv= kfold )
  resultadosVC_RN.append(scores.mean())
resultadosVC_RN
```

Figura 30: Implementação do Cross Validation para redes neurais

Desta forma, através da validação cruzada é possível identificar as médias de acurácia para cada modelo e compará-las com a acurácia que foi obtida no treinamento escolhido para realização do trabalho; o desvio padrão também pode ser analisado além dos valores máximos e mínimos de acurácia para cada modelo. Um ponto importante para avaliação é o coeficiente de variação de cada modelo escolhido, podendo ser um ponto crucial para decisão da escolha de um modelo ideal para o treinamento da base de dados escolhida. Desta forma o modelo de Random Forest tem uma variação entre os resultados do treinamento de 0,16% aproximadamente. O algoritmo de Redes Neurais foi o modelo que possui a menor variação dos três escolhidos, possuindo 0,12% de variação aproximadamente. Já o modelo de Árvore de Decisão é o modelo que possui maior variação, chegando a quase 0,30% entre os resultados obtidos.

6. Interpretação dos Resultados

Um ponto importante a ser analisado para interpretação dos resultados obtidos é a análise da matriz de confusão de cada modelo implementado. A matriz de confusão é um dashboard importante para analisara quantidade de erros e acertos da classificação feita por cada modelo, desta forma a Figura 31 representa a matriz de confusão do primeiro modelo de aplicado, o de Random Forest.

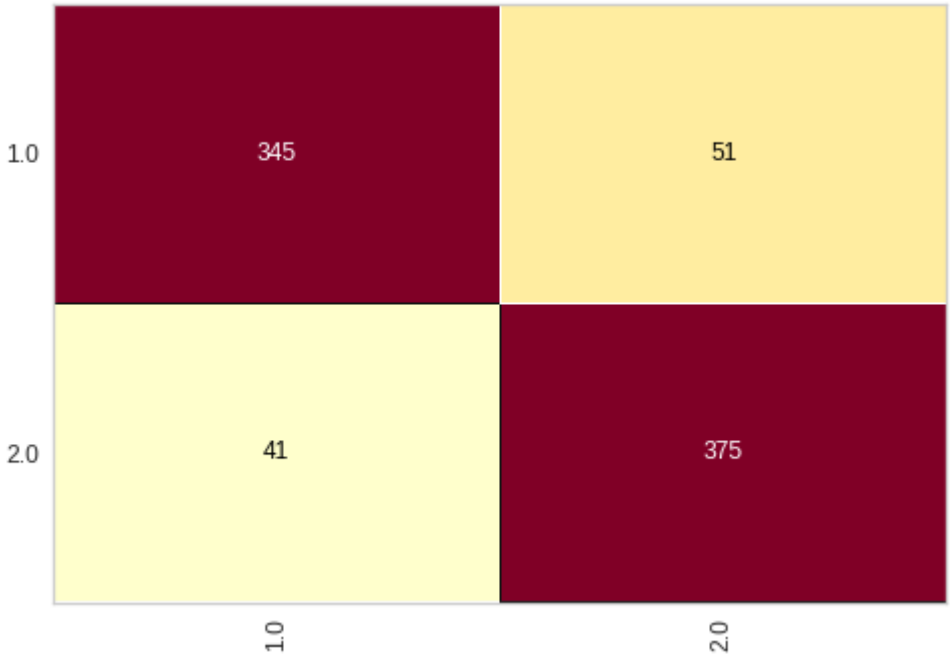

Figura 31: Matriz de confusão Random Forest

Neste modelo, que conforme indicado no tópico anterior é o que possui o segundo menor coeficiente de variação nos resultados, 0,16%, de Cross Validation, mas conforme a acurácia obtida no treinamento realizado de 0,8866995073891626 e a matriz de confusão acima, este modelo é o que representa o menor número de erros da predição em relação à classe alvo de teste. Do total de 812 alunos que estavam no conjunto de teste, 92 destes obtiveram classificação errada para os resultados, ou seja, são alunos que foram classificados como já experimentaram cigarros e na realidade nunca experimentaram e também o contrário.

O modelo de Árvore de Decisão é o modelo que possui maior coeficiente de variação entre os resultados de acurácia resultante do Cross Validation, 0,27%, sendo também o modelo que mais possui erros de predição e um valor de acurácia menor que o modelo analisado anteriormente. Dos 812 alunos utilizados para predições, o modelo errou a classificação de 97. A Figura 32 mostra a matriz de confusão para o modelo de Árvore de Decisão.

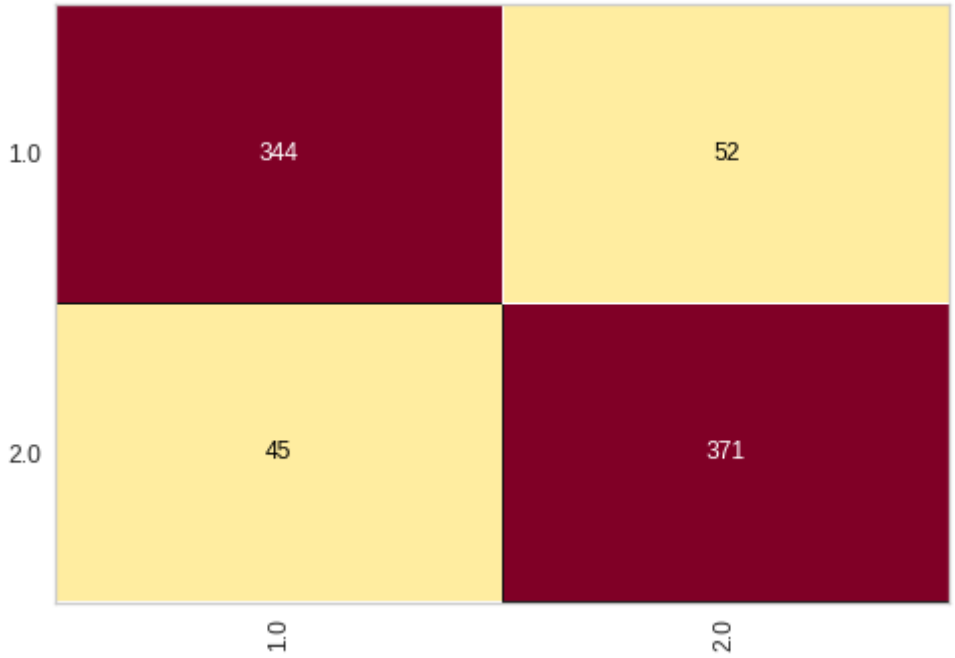

Figura 32: Matriz de confusão Árvore de Decisão

Redes Neurais é o modelo que possui a menor variação entre os resultados das acurácias resultantes do Cross Validation, 0,12%, e erro de classificação menor que o modelos de Random Forest e igual ao de Árvore de Decisão. Este modelo, conforme já citado anteriormente é o modelo mais famoso dentre os modelos de machine learning e o algoritmo que melhor lida com dados mais complexos e resultados que vão além das respostas de sim e não. A Figura 33 mostra a matriz de confusão para este modelo.

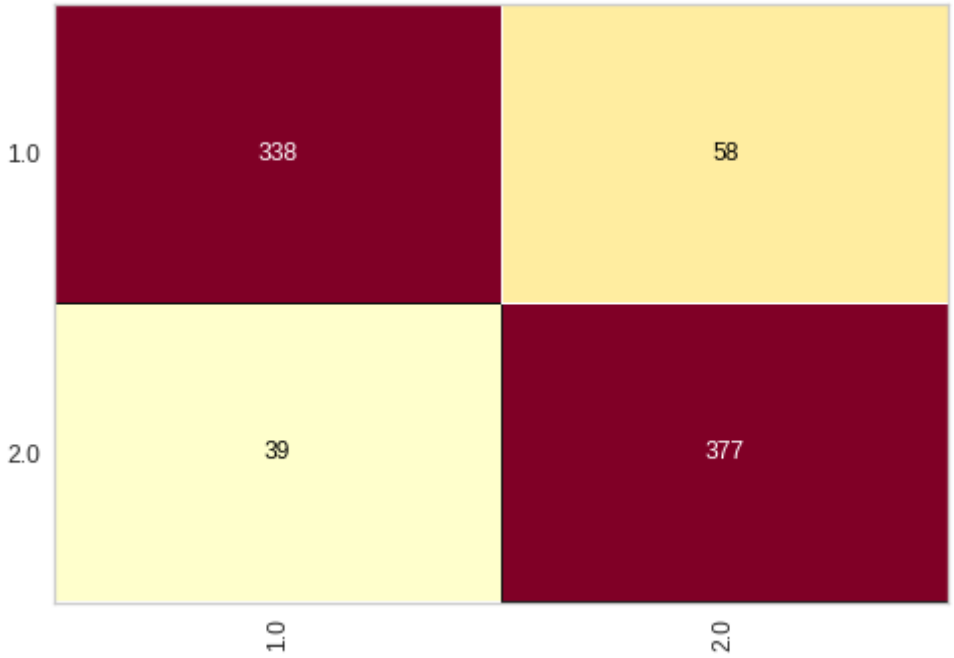

Figura 33: Matriz de confusão Redes Neurais

Sendo assim, para escolha do melhor modelo de classificação analisado para este trabalho é necessário analisar os seguintes pontos: ser o modelo que possui um baixo coeficiente de variação e um baixo índice de erro de classificação. O Random Forest neste caso é o que possui menor erro, errando apenas 92 dos alunos da classe de teste, este modelo apresenta-se como o melhor desempenho deste trabalho se comparada a acurácia em relação aos outros modelos. Além disso, o coeficiente de variação é intermediário, indicando uma boa coerência entre os resultados independente das subdivisões dos conjuntos de treinamento e teste, conseguindo ter um bom aprendizado com qualquer que seja o conjunto de dados de entrada.

Analisando agora os resultados alcançados e as influências dos dados de entrada para este trabalho, conforme citado anteriormente dados como etnia, influência dos pais e influência dos amigos para experimentação de cigarros ainda na adolescência e juventude dos jovens brasileiros são os dados que preferencialmente influenciam o hábito de fumar. A crescente porcentagem de alunos que já experimentaram cigarros quando são, pretos, algum responsável, seja masculino ou feminino fuma e quando algum amigo próximo também fuma, são nitidamente maiores. Chegando em todas as três análises em cerca de 40% de alunos experimentando cigarros quando estão sob estas condições. Outro ponto também importante de ressaltar é influência de um pai, ou responsável masculino fumante em relação à outro responsável que também fuma. Esta influência pode também está vinculada à maioria dos alunos serem, mesmo que sutilmente, do sexo masculino, tendo assim o pai como referência de hábitos e comportamentos.

7. Conclusão

Por fim, após a análise dos dados, implementação dos modelos de machine learning e aplicação do Cross Validation é possível perceber que para uma entrada de dados sociais é possível através de modelos de aprendizado de máquina prever com cerca de 88% de precisão se o aluno experimentou ou não cigarros durante a adolescência. Este resultado mostra que o ambiente familiar e de amizades influenciam de forma direta na possibilidade de um aluno se tornar fumante na fase adulta, considerando que a experimentação de cigarros na adolescência é um fator importante para o desenvolvimento deste hábito.

O objetivo deste projeto foi analisar exatamente este fato, o quanto as características sociais influenciam em hábitos ruins à saúde do aluno e através principalmente do algoritmo de Random Forest foi possível prever com 88,93% de acurácia esta influência. Através deste resultado, é possível que escolas, pais ou responsáveis, ministérios de saúde e educação consigam prever a probabilidade de um aluno desenvolver o hábito do fumo e aplicar políticas especiais e focadas para estes alunos, a fim de afastar esta possibilidade e minimizar a incidência do aumento de consumo de cigarros por adolescentes e jovens no Brasil.

Outros estudos podem ser desenvolvidos utilizando como base este trabalho, como por exemplo a probabilidade do desenvolvimento do alcoolismo entre os jovens brasileiros, utilizando também as condições sociais como base de entrada para o treinamento. Este tema também se torna importante devido ao grande consumo de álcool no Brasil e principalmente o quanto o álcool pode ser porta de entrada para outras drogas e para violência doméstica e mortalidade de jovens em acidentes de trânsito.

8. Links

Abaixo estão os links para acesso à breve apresentação do tema e para a documentação utilizada para realização deste trabalho, como arquivo python e as bases de dados.

Link para o vídeo: https://youtu.be/4NQpaKEmpZw

Link para o repositório: https://github.com/LeandraInacio/TCC_PUC_Pos-graduacao_Ciencia_de_dados_e_BigData

Referências

Adjuto, G. (27 de Dezembro de 2022). *agenciabrasil.ebc.com.br*. Fonte: AgênciaBrasil: https://agenciabrasil.ebc.com.br/economia/noticia/2017-11/populacao-brasileira-e-formada-basicamente-de-pardos-e-brancos-mostra-ibge#:~:text=Na%20Regi%C3%A3o%20Sudeste%2C%20a%20que,205%2C5%20milh%C3%B5es%20de%20pessoas.

Amazon. (04 de Janeiro de 2023). Fonte: AWS AMAZON: https://aws.amazon.com/pt/what-is/neural-network/

Blog Somos Tera. (04 de Janeiro de 2023). Fonte: ÁRVORE DE DECISÃO: ENTENDA ESSE ALGORITMO DE MACHINE LEARNING: https://blog.somostera.com/data-science/arvores-de-decisao

IBGE. (23 de Dezembo de 2022). *Instituto Brasileiro de Geografia e Estatística*. Fonte: IBGE: https://www.ibge.gov.br/

ICMC JÚNIOR. (04 de Janeiro de 2023). Fonte: RANDOM FOREST: https://icmcjunior.com.br/random-forest/#:~:text=O%20algoritmo%20Random%20Forest%20(Floresta,para%20chegar%20no%20resultado%20final.

INCA. (19 de outubro de 2022). *Instituto Nacional de Câncer - INCA*. Fonte: gov.br: https://www.gov.br/inca/pt-br/assuntos/gestor-e-profissional-de-saude/observatorio-da-politica-nacional-de-controle-do-tabaco/dados-e-numeros-do-tabagismo/prevalencia-do-tabagismo

PNUD. (23 de Dezembro de 2022). *PNUD NO BRASIL*. Fonte: PNUD: https://www.undp.org/pt/brazil

Silva, I. (s.d.). *TABAGISMO – O Mal da Destruição em Massa.* Fonte: Fiocruz: http://www.fiocruz.br/biosseguranca/Bis/infantil/tabagismo.htm

Skit Learn. (04 de Janeiro de 2023). Fonte: Skit Learn: https://scikit-learn.org

APÊNDICE

1. **Funções de transformações das respostas da pesquisa de formato numérico para formato descritivo:**

```
def ajuste_municipio(valor):
  if valor == 3106200:
    return "BELO HORIZONTE"
  elif valor == 3205309:
    return "VITORIA"
  elif valor == 3304557:
    return "RIO DE JANEIRO"
  elif valor == 3550308:
    return "SAO PAULO"
```

Equação 1: Transformação de campo numérico para descritivo de municípios

```
def ajuste_escola(valor):
  if valor == 1:
    return "Publica"
  else:
    return "Privada"
```

Equação 2: Transformação de campo numérico para descritivo de tipo de escola

```
def ajuste_sexo(valor):
  if valor == 1:
    return "Homem"
  elif valor == 2:
    return "Mulher"
  else:
    return "Sem Resposta"
```

Equação 3: Transformação de campo numérico para descritivo de sexo do aluno

```
def ajuste_mora_mae(valor):
  if valor == 1:
    return "sim"
  else:
    return "não"
```

Equação 4: Transformação de campo numérico para descritivo de mora com a mãe

```
def ajuste_idade(valor):
  if valor == 1:
    return "Menos de 13 anos"
  elif valor == 2:
    return "13 a 15 anos"
  elif valor == 3:
    return "16 a 17 anos"
  elif valor == 4:
    return "18 anos ou mais"
  else:
    return "Sem Resposta"
```

Equação 5: Transformação de campo numérico para descritivo de idade

```
def ajuste_raca(valor):
  if valor == 1:
    return "Branca"
  elif valor == 2:
    return "Preta"
  elif valor == 3:
    return "Amarela"
  elif valor == 4:
    return "Parda"
  elif valor == 5:
    return "Indigena"
  else:
    return "Sem Resposta"
```

Equação 6: Transformação de campo numérico para descritivo de raça do aluno

```
def ajuste_mora_pai(valor):
  if valor == 1:
    return "sim"
  else:
    return "não"
```

Equação 7: Transformação de campo numérico para descritivo de mora com o pai

```
def ajuste_narguile(valor):
  if valor == 1:
    return "sim"
  elif valor == 2:
    return "não"
  else:
    return "Sem Resposta"
```

Equação 8: Transformação de campo numérico para descritivo de já experimentou narguilé

```
def ajuste_ensino_medio(valor):
  if valor == 1:
    return "6º ano"
  elif valor == 2:
    return "7º ano"
  elif valor == 3:
    return "8º ano"
  elif valor == 4:
    return "9º ano"
  elif valor == 5:
    return "1º ano ensino medio"
  elif valor == 6:
    return "2º ano ensino medio"
  elif valor == 7:
    return "3º ano ensino medio"
  else:
    return "Sem Resposta"
```

Equação 9: Transformação de campo numérico para descritivo de ano escolar do aluno

```
def ajuste_cigarro_eletronico(valor):
  if valor == 1:
    return "sim"
  elif valor == 2:
    return "não"
  else:
    return "Sem Resposta"
```

Equação 10: Transformação de campo numérico para descritivo de já experimentou cigarro eletrônico

```
def ajuste_consumiu_tabaco(valor):
  if valor == 1:
    return "sim"
  elif valor == 2:
    return "não"
  else:
    return "Sem Resposta"
```

Equação 11: Transformação de campo numérico para descritivo de já consumiu tabaco de outras formas

```
def ajuste_amigo_fuma(valor):
  if valor == 1:
    return "sim"
  elif valor == 2:
    return "não"
  else:
    return "Sem Resposta"
```

Equação 12: Transformação de campo numérico para descritivo de amigo fumou nos últimos 30 dias

```
def ajuste_responsavel_fuma(valor):
  if valor == 1:
    return "Nenhum deles"
  elif valor == 2:
    return "Só meu pai ou responsável do sexo masculino"
  elif valor == 3:
    return "Só minha mãe ou responsável do sexo feminino"
  elif valor == 4:
    return "Os dois (ambos)"
  elif valor == 5:
    return "Não sei"
  else:
    return "Sem Resposta"
```

Equação 13: Transformação de campo numérico para descritivo de responsável fuma

```
def ajuste_fumar_perto_7dias(valor):
  if valor == 1:
    return "Nenhum dia nos últimos 7 dias"
  elif valor == 2:
    return "1 ou 2 dias"
  elif valor == 3:
    return "3 ou 4 dias"
  elif valor == 4:
    return "5 ou 6 dias"
  elif valor == 5:
    return "Todos os dias"
  else:
    return "Sem Resposta"
```

Equação 14: Transformação de campo numérico para descritivo de frequência que fumaram perto do aluno nos últimos sete dias

2. Processamento de dados da base de enriquecimento

```
def padronizacao_acentuacao(valor):
    return unidecode.unidecode(valor)
```

Equação 15: Equação para padronização da descrição das cidades

2.1 Enriquecimento dos dados

```
def enriquecimento_IDH(valor):
  a = str(valor).strip()
  for i in base_IDH.index:
    b = str(base_IDH['CIDADE'][i]).strip()
    if a == b:
      return base_IDH['IDHM Educação 2010'][i]
```

Equação 16: Equação de enriquecimento com dados de IDH à base principal

3. Exploração e Representação dos dados

```
def ajuste_ja_fumou(valor):
  if valor == 1:
    return "sim"
  else:
    return "não"
```

```
df['JA_FUMOU'] = df['JA_FUMOU'].apply(lambda x : ajuste_ja_fumou(x))
```

Equação 17: Função de transformação dos dados da coluna JA_FUMOU

```
def countplot_format(classe, eixox, eixoy, titulo):
  grafico = plt.subplots(figsize=(15, 13))
  grafico = sns.countplot(x=df[classe],palette = 'OrRd_r')
  grafico.set_title(f'{titulo}\n', fontsize=30);
  grafico.set_xlabel(eixox, fontsize=10);
  grafico.set_ylabel(eixoy, fontsize=10);
  return grafico
```

Equação 18: Função de padronização da formatação de gráficos countplot

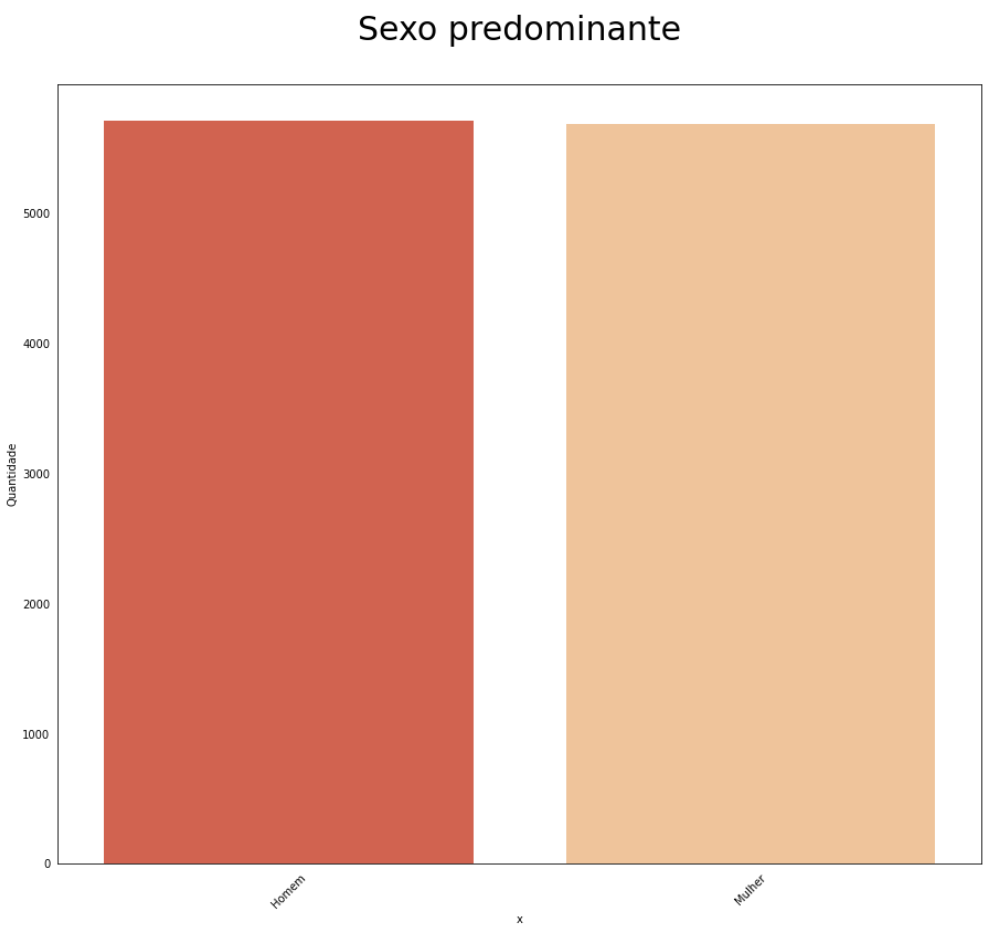

Figura 34: Representação do sexo predominante entre os alunos analisados

```
def mapaDeCalor(principal, comparativo, eixoX, eixoY, titulo):
  bd = df.loc[~df[principal].isin(['NAO SE APLICA', 'NS/NR']),[principal, comparativo]]# "~" para negar
  info_adequacao = pd.pivot_table(bd, index=principal, columns=comparativo, aggfunc=len, fill_value=0)
  grafico = sns.heatmap(info_adequacao, cmap="OrRd", annot=True, fmt ='d', linewidths=.3, xticklabels=True, yticklabels=True)
  grafico.set_xlabel(eixoX, fontsize=20)
  grafico.set_ylabel(eixoY, fontsize=20)
  grafico.set_title(f'{titulo}\n', fontsize=20)

  plt.tick_params(left = False, right = False , labelleft = True ,labelbottom = True, bottom = False)
  plt.xticks(rotation=45)
  plt.gcf().set_size_inches(15, 10)
  return grafico
```

Equação 19: Função de padronização e formatação dos mapas de calor

3.1 Mapas de calor para interpretação dos dados da base de estudo

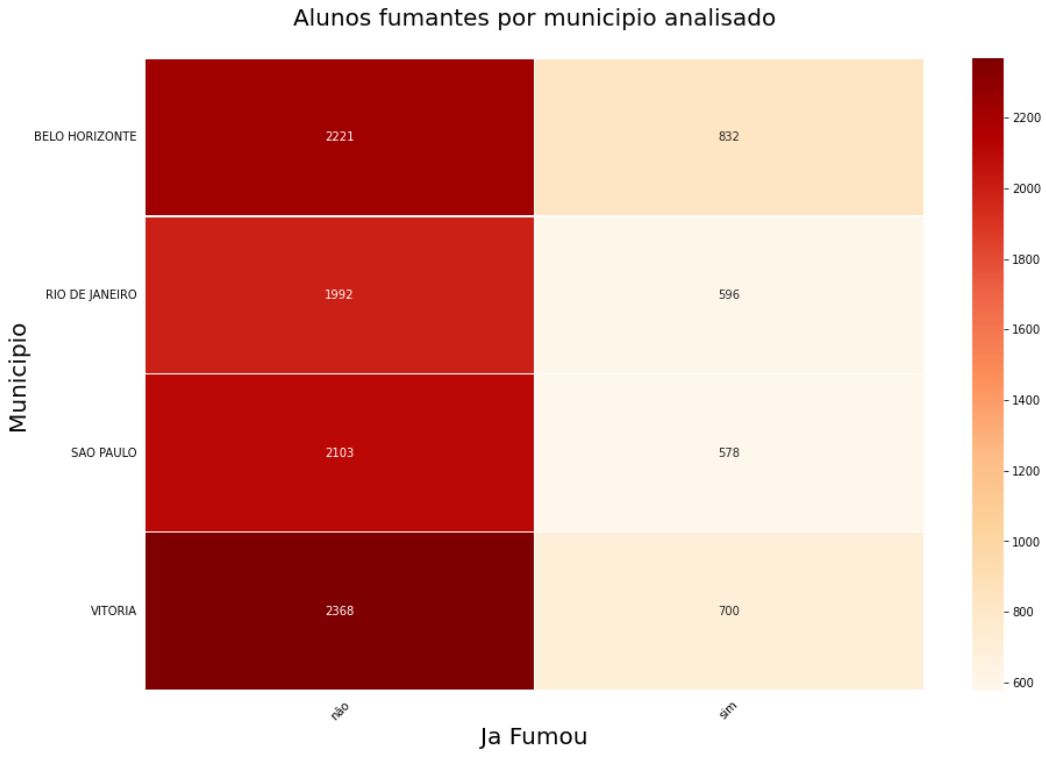

Figura 35: Mapa de calor dos municípios analisados em relação a alunos que já experimentaram cigarros

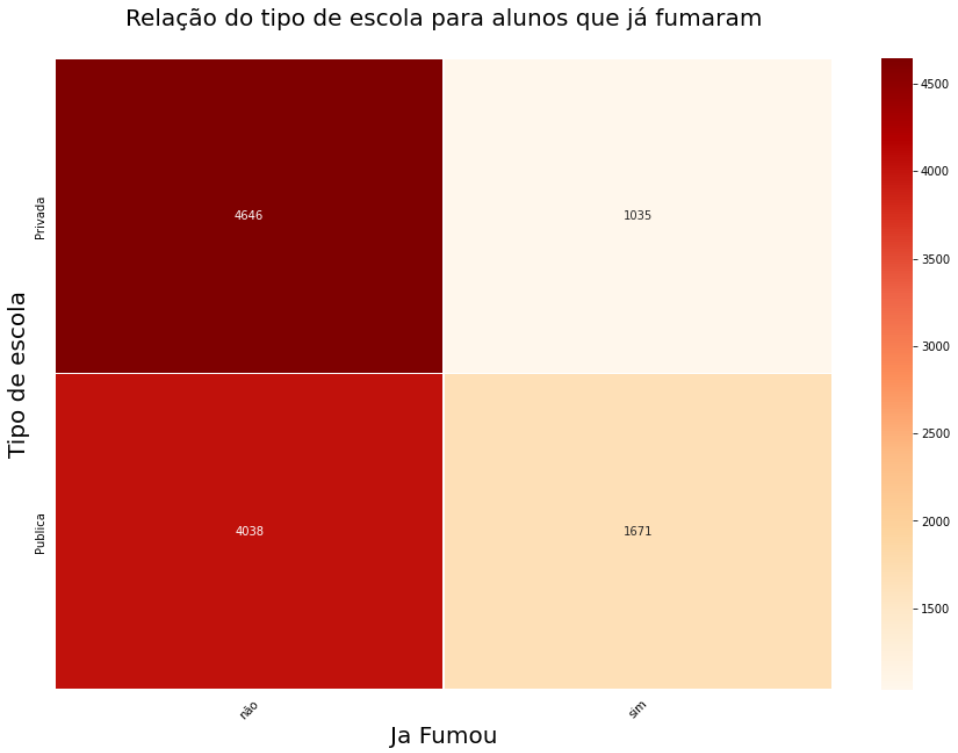

Figura 36: Mapa de calor da relação entre tipo da escola e alunos que já experimentaram cigarros

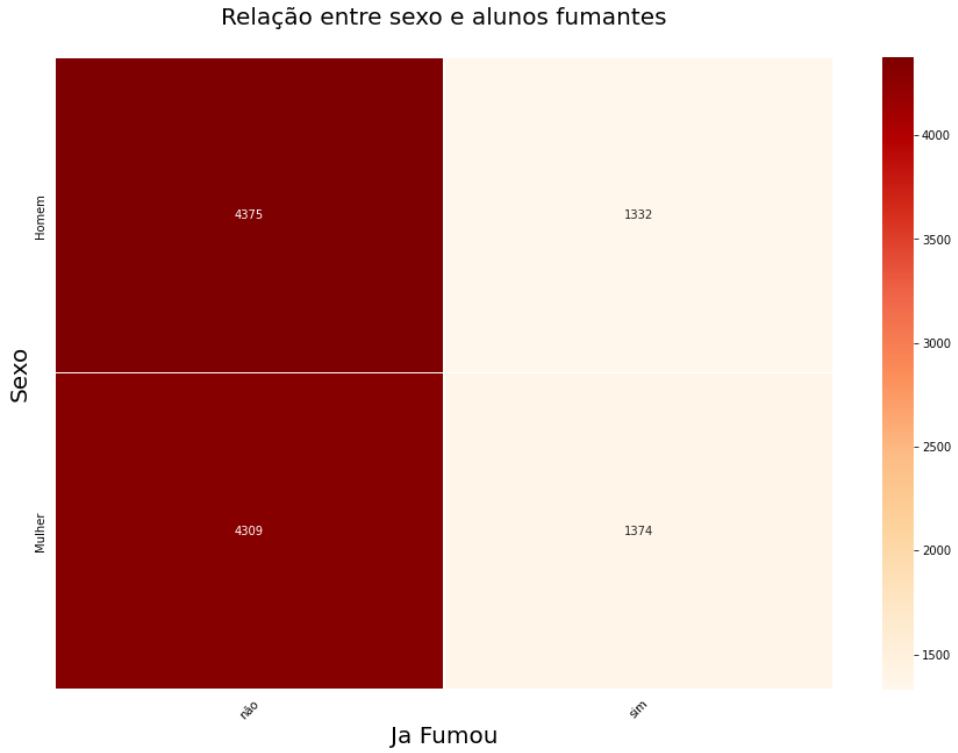

Figura 37: Mapa de calor da relação entre sexo do aluno e se aluno já experimentou cigarros

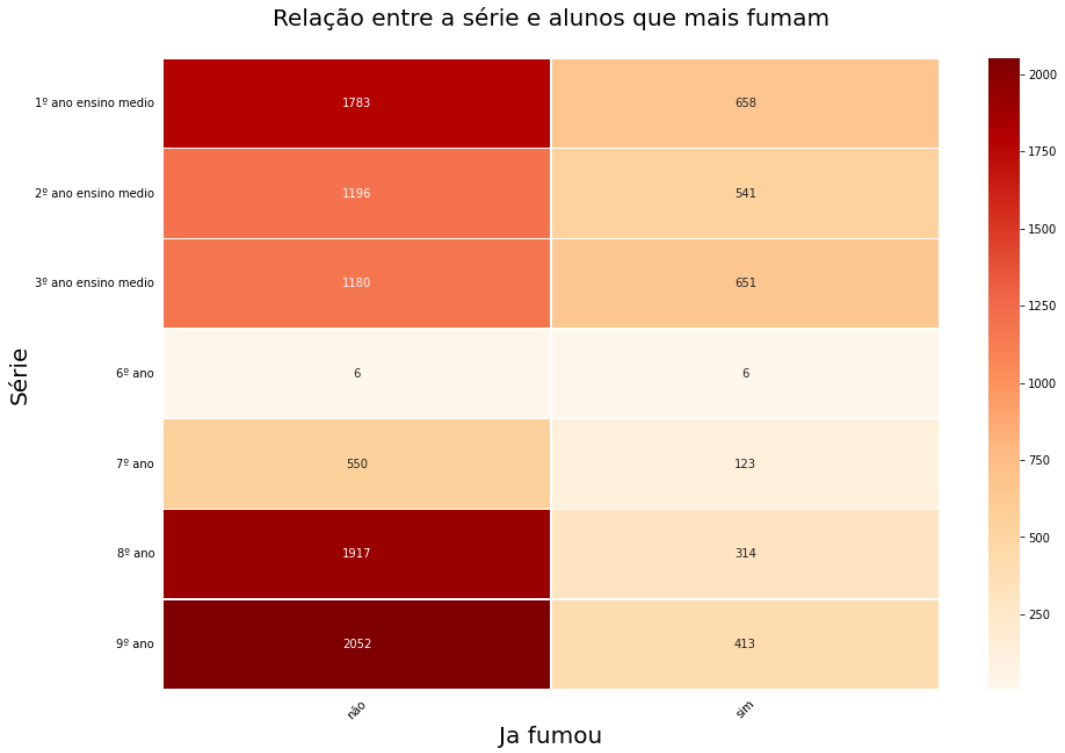

Figura 38: Mapa de calor da relação entre o ano escolar e alunos que já experimentaram cigarros

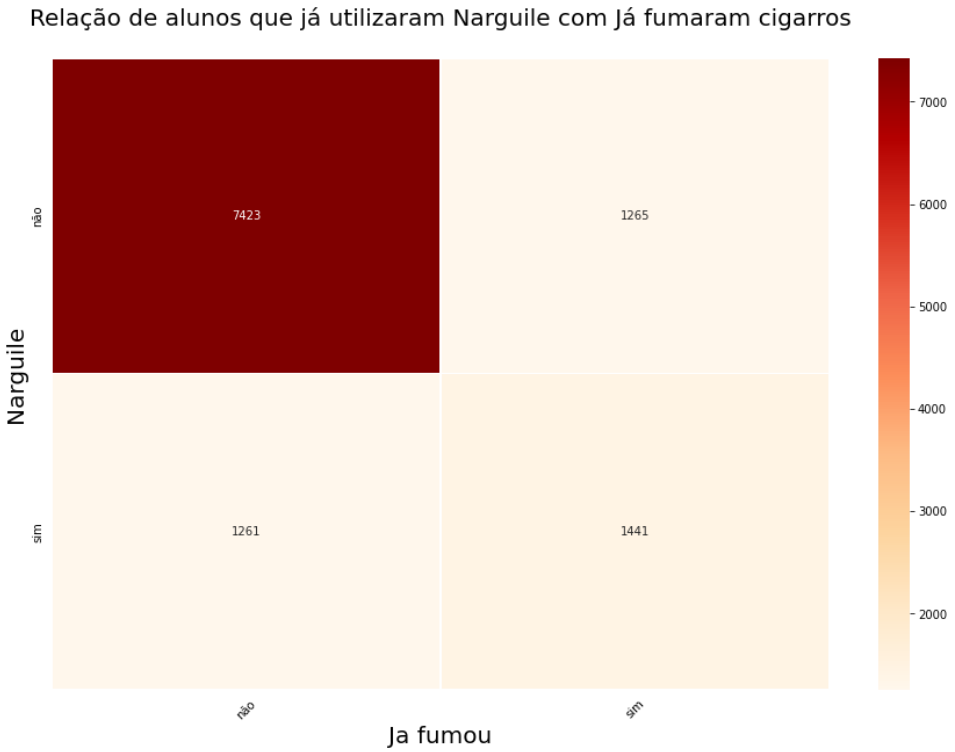

Figura 39: Mapa de calor da relação entre alunos que já experimentaram narguilé e alunos que já experimentaram cigarros

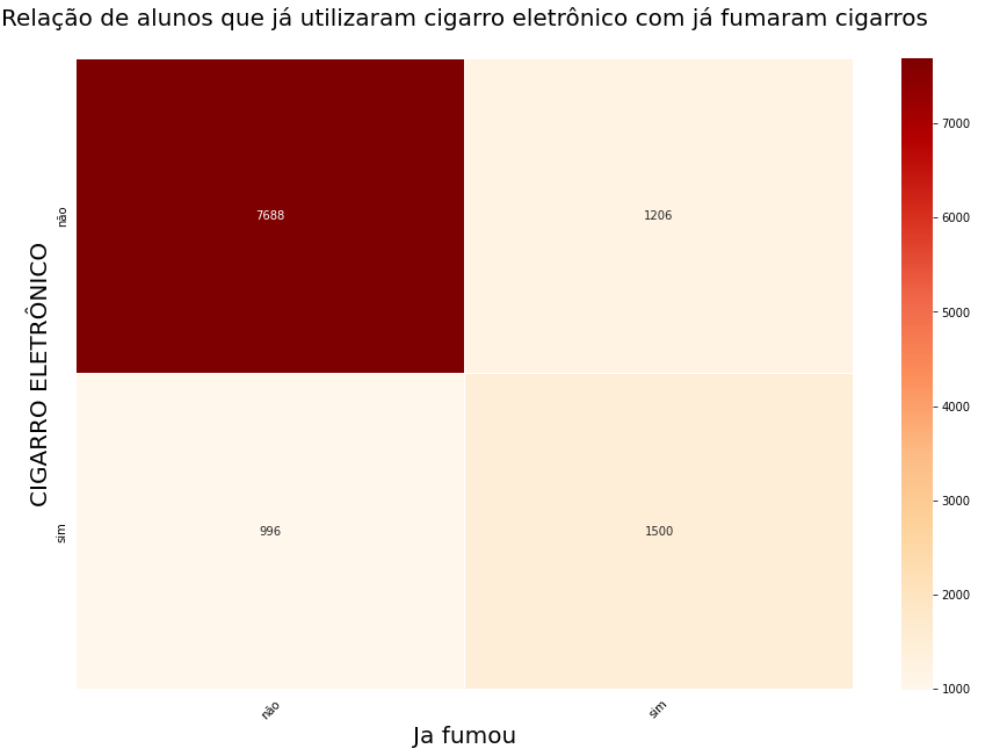

Figura 40: Mapa de calor da relação entre alunos que já experimentaram cigarros eletrônico e já experimentaram cigarros de tabaco

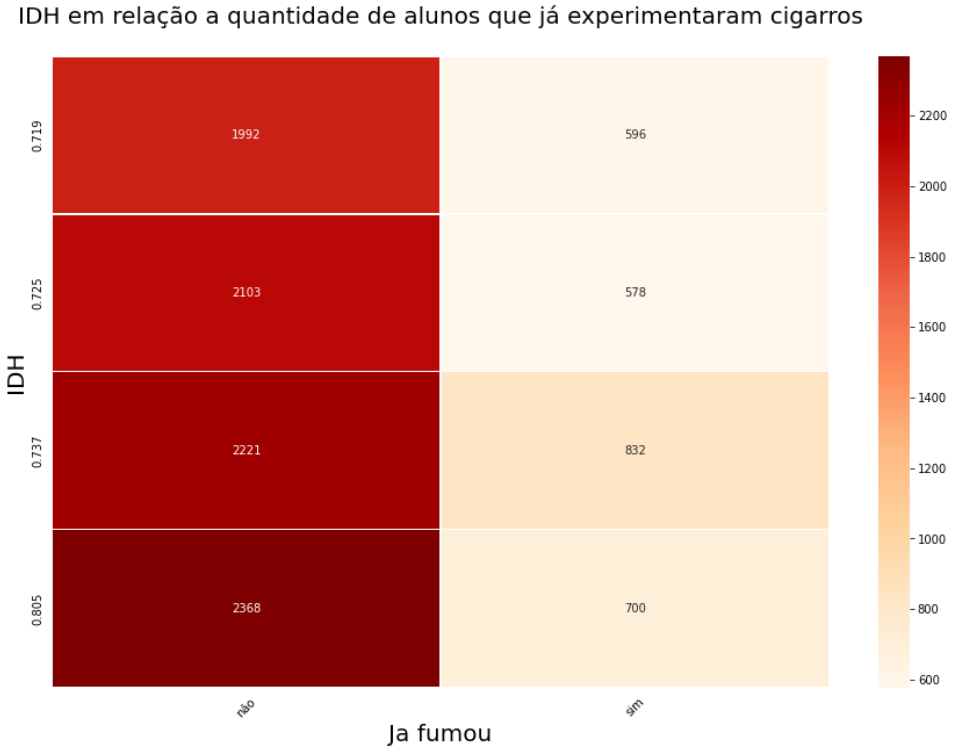

Figura 41: Mapa de calor da relação entre o IDH da cidade analisada e alunos que já experimentaram cigarros

4. Criação dos modelos de machine learning

4.1 Predições do modelo de Randon Forest

([2., 2., 2., 1., 1., 1., 2., 2., 1., 1., 2., 2., 2., 1., 2., 1., 1., 1., 2., 2., 1., 2., 2.,
2., 1., 1., 1., 2., 1., 1., 1., 2., 1., 2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 1., 2., 2., 1., 1., 2., 1., 2., 2., 2.,1.,
1., 1., 2., 1., 1., 1., 2., 2., 2., 1., 1., 1., 2., 1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 2., 2., 2., 1., 1., 2., 2.,2.,
1., 1., 1., 2., 2., 1., 2., 2., 2., 1., 2., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 2., 2., 1., 2., 1., 1., 2.,1.,
2., 2., 2., 1., 2., 2., 1., 2., 1., 1., 1., 1., 2.,1., 2., 1., 1., 1., 2., 1., 2., 2., 1., 1., 2., 1., 2., 2., 1., 2., 2.,
1., 1., 1., 1., 1., 2., 1., 2., 1., 2., 2., 2., 1., 1., 1., 1., 2., 1., 1., 2., 1., 1., 2., 1., 2., 2., 2., 1., 2., 2.,2.,
1., 2., 1., 1., 1., 1., 1., 2., 2., 1., 1., 2., 2., 1., 2., 1., 1., 1., 2., 1., 1., 2., 2., 2., 1., 2., 2., 2., 1., 1.,1.,
2., 2., 1., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1., 2., 2., 2., 1., 2., 2., 2., 2., 2., 2., 2., 2., 1., 1.,2.,
1., 1., 1., 1., 2., 2., 1., 2., 1., 2., 2., 2., 1., 1., 2., 1., 2., 2., 2., 2., 1., 2., 1., 1., 2., 1., 1., 1., 2., 1.,1.,
2., 2., 2., 2., 1., 1., 2., 2., 1., 2., 1., 1., 1., 2., 2., 1., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1.,1.,
1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 1., 2., 1., 2., 2., 1., 1., 1., 1., 2., 1., 1., 2., 2., 1., 2., 2., 2., 1., 2.,2.,
1., 2., 1., 2., 1., 1., 2., 2., 1., 2., 2., 2., 1., 2., 2., 1., 2., 2., 1., 1., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2.,2.,
2., 2., 1., 2., 2., 1., 1., 1., 1., 2., 2., 1., 1., 2., 1., 1., 1., 1., 1., 2., 2., 2., 1., 1., 1., 1., 1., 2., 2.,2.,
2., 1., 1., 2., 2., 1., 2., 1., 1., 2., 2., 2., 1., 1., 2., 1., 2., 2., 1., 2., 2., 2., 2., 2., 2., 2., 1., 2., 2., 2.,1.,
1., 1., 1., 2., 2., 1., 1., 2., 1.,1., 2., 2., 1., 1., 2., 2., 2., 1., 2., 2., 1., 1., 2., 2., 2., 1., 1., 1., 1., 1., 1.,
1., 1., 1., 2., 1., 2., 2., 2., 1., 2., 2., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 2., 1., 2.,1.,
1., 2., 2., 1., 2., 1., 2., 1., 2., 2., 2., 2., 2., 2., 1., 2., 1., 2., 2., 2., 1., 2., 2., 1., 2., 2., 1., 2., 1.,2.,
2., 2., 1., 2., 1., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2., 2., 2., 2.,1., 1., 2., 2., 2., 2., 2., 1., 1., 2., 2., 1.,
2., 2., 2., 2.,2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 1., 2., 1., 2., 1., 2., 2.,1., 2., 2., 2., 1., 2., 2., 1., 2., 1.,
2., 1., 2., 1., 2., 2., 1.,1., 1., 1., 2., 1., 2., 2., 1., 1., 1., 2., 2., 1., 1., 2., 1., 1.,2., 1., 1., 2., 2., 1., 1.,
2., 1., 2., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 2., 2., 1., 2., 2., 1., 2., 2., 2., 2., 1., 1., 2., 2., 2.,2.,
1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1., 2., 1.,2., 2., 2., 2., 2., 1., 2., 2., 2., 1., 2., 2., 2., 1., 1., 2., 1., 1.,
1., 2., 1., 1., 2., 2., 2., 1., 1., 1., 2., 1., 1., 1., 1., 1., 2., 1., 1., 2., 2., 1., 2., 1., 2., 1., 2., 1., 1.,1.,
1., 1., 2., 1., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 2., 1., 1., 1.,2., 2., 2., 2., 2., 2., 2., 2., 2., 2., 1., 2.,
1., 2., 1., 1., 2.,2., 2., 1., 1., 1., 2., 1., 1., 1., 2., 2., 1., 2., 1., 2., 2., 2.,2., 1., 2., 2., 1., 1., 1., 1., 1.,
1., 2., 2., 1., 2., 1., 2., 2.,2., 1., 1., 2., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 1., 2.,2., 1., 1., 2., 2., 2.,
2., 2., 1., 1., 2., 1., 1.])

4.2 Predições do modelo de Árvore de Decisão

([2., 2., 2., 1., 1., 1., 2., 2., 1., 2., 2., 2., 2., 1., 2., 2., 1.,1., 2., 2., 1., 2., 1., 1., 1., 2., 1., 1., 2., 2.,
2., 1., 1., 1.,2., 1., 1., 1., 2., 1., 2., 2., 1., 2., 2., 1., 1., 2., 1., 2., 1.,2., 2., 1., 1., 2., 1., 2., 2., 1., 1.,
1., 1., 2., 1., 1., 1., 2.,2., 2., 2., 1., 1., 2., 1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 2.,2., 2., 1., 1., 2., 2., 2.,
1., 1., 1., 2., 2., 1., 2., 2., 2., 1.,2., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 2., 2., 1., 2.,1., 1., 2., 2.,
2., 2., 2., 1., 2., 1., 1., 2., 1., 1., 1., 1., 2.,1., 2., 1., 1., 1., 2., 1., 2., 2., 1., 1., 2., 1., 2., 1., 1., 2.,2.,
1., 1., 1., 1., 1., 2., 1., 2., 1., 2., 2., 2., 1., 1., 1., 1.,2., 1., 1., 2., 1., 2., 2., 1., 2., 2., 2., 1., 2., 2., 2.,
1., 2.,1., 1., 1., 1., 2., 2., 2., 1., 1., 2., 2., 1., 2., 1., 1., 1., 2.,1., 1., 2., 2., 2., 1., 2., 2., 2., 1., 1., 1.,
2., 2., 1., 2., 2.,1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1., 1., 2., 2., 1., 2., 2.,2., 2., 2., 2., 2., 2., 2., 1., 2.,
1., 1., 1., 1., 2., 2., 1., 2.,1., 2., 2., 2., 1., 1., 2., 1., 2., 2., 2., 2., 1., 2., 1., 1., 2.,1., 1., 1., 2., 1., 1.,
2., 2., 2., 2., 1., 1., 2., 2., 1., 2., 1.,1., 1., 1., 2., 1., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 2.,1., 1., 1.,
1., 2., 2., 2., 1., 2., 1., 1., 1., 1., 1., 2., 1., 2.,2., 1., 1., 1., 1., 2., 1., 1., 2., 2., 1., 2., 2., 2., 1., 2., 2.,

1., 2., 2., 2., 2., 1., 2., 2., 2., 2., 2., 2., 1., 2., 2., 1., 2.,2., 1., 1., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2., 2.,
2., 2., 1.,2., 2., 1., 1., 1., 1., 2., 2., 1., 1., 2., 1., 1., 1., 1., 1., 2.,2., 2., 1., 1., 1., 1., 1., 1., 2., 2., 2.,
2., 1., 1., 2., 2., 1.,2., 1., 1., 2., 1., 2., 1., 1., 2., 1., 2., 1., 1., 1., 2., 2., 2.,2., 2., 2., 1., 2., 2., 2., 1.,
1., 1., 1., 2., 2., 1., 1., 1., 1.,1., 2., 2., 1., 1., 2., 2., 2., 1., 2., 2., 1., 1., 2., 2., 2., 1.,1., 1., 1., 1., 1.,
1., 1., 1., 2., 1., 2., 1., 2., 1., 2., 2., 1.,1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 2., 1.,2., 1.,
1., 2., 2., 1., 2., 1., 2., 1., 2., 2., 2., 2., 2., 2., 2.,1., 2., 1., 2., 2., 2., 1., 2., 2., 1., 2., 2., 1., 2., 1., 2.,
2.,2., 1., 2., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2., 2., 2., 2.,1., 1., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2., 1.,
2., 2., 2., 2.,1., 2., 1., 2., 1., 1., 1., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2.,1., 2., 2., 2., 1., 2., 2., 1., 2., 1.,
2., 1., 2., 1., 2., 2., 1.,1., 1., 1., 2., 1., 2., 2., 1., 1., 1., 2., 2., 1., 1., 2., 1., 1.,2., 1., 1., 2., 2., 1., 1.,
2., 1., 2., 1., 1., 1., 1., 1., 1.,1., 1., 1., 1., 1., 2., 2., 1., 2., 2., 1., 2., 2., 2., 2., 1., 1.,1., 2., 2., 2.,
1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1., 2., 1.,2., 2., 2., 2., 1., 1., 2., 2., 1., 1., 2., 2., 2., 1., 1., 2., 1.,1.,
1., 2., 1., 1., 2., 2., 2., 1., 1., 1., 2., 1., 1., 1., 1.,1., 2., 1., 1., 1., 2., 1., 2., 1., 2., 1., 2., 1., 1., 1.,
1., 1.,2., 1., 2., 2., 1., 2., 2., 2., 1., 1., 1., 2., 2., 2., 1., 1., 1.,2., 2., 2., 2., 2., 2., 2., 2., 2., 2., 1., 2.,
1., 2., 1., 2., 1.,2., 2., 1., 1., 1., 2., 1., 1., 1., 2., 2., 1., 2., 1., 2., 2., 2.,2., 1., 2., 2., 1., 1., 1., 1., 1.,
1., 2., 2., 1., 2., 1., 2., 2.,2., 1., 2., 2., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 1., 2.,1., 1., 1., 2., 2., 2.,
2., 2., 1., 1., 2., 2., 1.])

4.3 Predições do modelo de Redes Neurais

([2., 2., 2., 1., 1., 1., 2., 2., 1., 2., 2., 2., 2., 1., 2., 2., 1.,1., 2., 2., 1., 2., 1., 1., 1., 2., 1., 1., 2., 2.,
2., 1., 1., 1.,2., 1., 1., 1., 2., 1., 2., 2., 1., 2., 2., 1., 1., 2., 2., 2., 1.,2., 2., 1., 1., 2., 1., 2., 2., 1., 1.,
1., 1., 2., 1., 1., 1., 2.,2., 2., 2., 1., 1., 2., 1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 2.,2., 2., 1., 1., 2., 2., 2.,
1., 1., 1., 2., 2., 1., 2., 2., 2., 1.,2., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 1., 2., 2., 2., 2.,1., 1., 2., 2.,
2., 2., 2., 1., 2., 2., 1., 2., 1., 1., 1., 1., 2.,1., 2., 1., 1., 1., 2., 1., 2., 2., 1., 1., 2., 1., 2., 1., 1., 2.,2.,
1., 1., 1., 1., 1., 2., 1., 1., 1., 2., 2., 2., 1., 1., 1., 1.,2., 1., 1., 2., 1., 2., 2., 1., 2., 2., 2., 1., 2., 2., 2.,
1., 2.,1., 1., 1., 1., 1., 2., 2., 1., 1., 2., 2., 1., 2., 1., 1., 1., 2.,1., 1., 2., 2., 2., 1., 2., 2., 2., 1., 1., 1.,
2., 2., 1., 2., 2.,1., 2., 2., 2., 1., 2., 2., 2., 2., 1., 1., 1., 2., 2., 1., 2., 2.,2., 2., 2., 2., 2., 2., 2., 1., 2.,
1., 1., 1., 1., 2., 2., 1., 2.,1., 2., 2., 2., 1., 1., 2., 1., 2., 2., 2., 2., 1., 2., 1., 1., 2.,1., 1., 1., 2., 1., 1.,
2., 2., 2., 2., 1., 1., 2., 2., 1., 2., 1.,1., 1., 2., 2., 1., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 2.,1., 1., 1.,
1., 2., 2., 1., 1., 2., 2., 1., 1., 1., 1., 1., 1., 1., 2., 2., 1., 2., 2., 2., 1., 2.,
2.,1., 2., 1., 2., 2., 1., 2., 2., 2., 2., 2., 1., 2., 2., 1., 2.,2., 1., 1., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2.,
2., 2., 2., 1.,2., 2., 1., 1., 1., 1., 2., 2., 1., 1., 2., 1., 1., 1., 1., 1., 2.,2., 2., 1., 1., 1., 1., 1., 2., 2.,
2., 2., 1., 2., 2., 2., 1.,2., 1., 1., 2., 2., 2., 1., 1., 2., 1., 2., 2., 1., 2., 2., 2., 2.,2., 2., 2., 1., 2., 2., 2.,
1., 1., 1., 1., 2., 2., 1., 1., 2., 1.,1., 2., 2., 1., 1., 2., 2., 2., 1., 2., 2., 1., 1., 2., 2., 2., 1.,1., 1., 1., 1.,
2., 1., 1., 1., 2., 1., 2., 2., 2., 1., 2., 2., 1.,1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 2., 1.,1.,
1., 1., 2., 2., 1., 2., 1., 2., 1., 2., 2., 2., 2., 2., 2.,1., 2., 1., 2., 2., 2., 1., 2., 2., 1., 2., 2., 1., 2., 1.,
2., 2.,2., 1., 2., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2., 2., 2.,1., 2., 2., 2., 2., 2., 2., 2., 1., 1., 2., 2.,
1., 2., 2., 2., 2.,2., 2., 1., 2., 1., 1., 1., 2., 1., 2., 1., 2., 1., 2., 1., 2., 2.,1., 2., 2., 2., 1., 2., 2., 1., 2.,
1., 2., 1., 2., 1., 2., 2., 1.,1., 1., 1., 2., 1., 2., 2., 1., 1., 1., 1., 2., 1., 1., 2., 1., 1.,2., 1., 1., 2., 2., 1.,
1., 2., 1., 2., 1., 1., 1., 1., 1., 1., 1.,1., 1., 1., 1., 1., 2., 2., 1., 2., 2., 1., 2., 2., 2., 1., 1., 1.,2., 2., 2.,
2., 2., 2., 2., 2., 1., 2., 1., 2., 2., 1., 1., 2., 1.,2., 2., 2., 2., 1., 1., 2., 2., 2., 1., 2., 2., 2., 1., 1., 2., 1.,
1., 1., 2., 1., 1., 2., 2., 2., 1., 1., 2., 2., 1., 1., 1., 1., 1.,1., 2., 1., 1., 2., 2., 1., 2., 1., 2., 1., 2.,1., 1., 1.,
1., 1., 1.,2., 1., 2., 2., 1., 2., 2., 2., 1., 2., 1., 2., 2., 2., 1., 1., 1.,2., 2., 2., 2., 2., 2., 2., 2., 2., 2., 1.,
2., 1., 2., 1., 1., 2.,2., 2., 1., 1., 1., 2., 1., 2., 1., 2., 2., 1., 2., 1., 2., 2., 2.,2., 1., 2., 2., 1., 1., 1., 1.,
1., 1., 2., 2., 1., 2., 1., 2., 2.,2., 1., 1., 2., 1., 1., 1., 1., 2., 2., 2., 1., 2., 1., 2., 1., 2.,2., 1., 1., 2., 2.,
2., 2., 2., 1., 1., 2., 1., 1.])

5. Cross Validation

Tabela 6: Resultados de cada iteração para Random Forest

Resultados das iterações de Cross Validation para Random Forest	0.8721361289398477, 0.8756488258043392, 0.8730610254346537, 0.8730562508952261, 0.8747242703480639, 0.8739791011588489, 0.8739848988138681, 0.8732414348173057, 0.873609415391751, 0.8741745162368446, 0.8728765235896352, 0.8706601141797, 0.8743521973112521, 0.8739750086964826, 0.872134423747195, 0.8728748183969826, 0.8728710669731466, 0.8730603433575925, 0.8741697416974169, 0.8702877001043576, 0.87564916684287, 0.8712098682909195, 0.8715778488653647, 0.8734290060090988, 0.8710345744862253, 0.876016465340254, 0.8714018729836097, 0.8728748183969823, 0.8734266187393851, 0.8726872472051893]

Tabela 7 : Resultados de cada iteração para Árvore de Decisão

Resultados das iterações de Cross Validation para Árvore de Decisão	[0.862529073534728, 0.8621627981529352, 0.8641895901398939, 0.8584628711351808, 0.8612341502342934, 0.8632646936450881, 0.8610506715048667, 0.8595685180511694, 0.861784586422574, 0.8610527177360499, 0.8617910661546541, 0.8595705642823527, 0.8623415023429347, 0.8608562795424627,

Resultados das iterações de Cross Validation para Árvore de Decisão	0.8630818969927223, 0.8619748859226115, 0.8588305106710956, 0.8566147833382216, 0.8608637823901345, 0.863079850761539, 0.8571751096438875, 0.8580928443295524, 0.8592005374767242, 0.8614189931178424, 0.8590208101711332, 0.8636336973351249, 0.8629014876100701, 0.8542135310447374, 0.8641906132554856, 0.8595692001282306]

Tabela 8: Resultados de cada iteração para Redes Neurais

Resultados das iterações de Cross Validation para Redes Neurais	[0.8819314376138216, 0.8824839200332854, 0.8808227213510582, 0.8832161297583401, 0.8813745216934608, 0.8813707702696251, 0.8813734985778694, 0.8819266630743942, 0.8834043830271943, 0.8819345069605964, 0.8810058590419546, 0.8798961196635997, 0.8837754329484145, 0.8826633063003457, 0.8819307555367606, 0.8800819856627402, 0.8826656935700596, 0.881191042964034, 0.8830404949151154, 0.8815576593843572, 0.8795298442818069, 0.8815569773072962, 0.8815559541917046, 0.8813738396163998, 0.8824897176883043, 0.8826708091480177, 0.8793446603597275, 0.8832226094904202, 0.8824818738021021, 0.8821115059579432]