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

José de Ribamar Mendes Farias

CLUSTERIZAÇÃO DO ÍNDICE DE VACINAÇÃO NO ESTADO DO MARANHÃO COM BASE EM INDICADORES SOCIOECONÔMICOS

São Luís
2022

José de Ribamar Mendes Farias

CLUSTERIZAÇÃO DO ÍNDICE DE VACINAÇÃO NO ESTADO DO MARANHÃO COM BASE EM INDICADORES SOCIOECONÔMICOS

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.

São Luís

2022

SUMÁRIO

1. INTRODUÇÃO

1.1. Contextualização

> *O conceito de desenvolvimento humano nasceu definido como um processo de ampliação das escolhas das pessoas para que elas tenham capacidades e oportunidades para serem aquilo que desejam ser (UNDP).*

O IDH[1] (Índice de Desenvolvimento Humano) é uma medida geral e sintética que modificou a classificação do desenvolvimento dos países, passando a avaliar a **qualidade de vida**, tendo por base dimensões sociais: **saúde**, **educação** e **renda**, em contraponto à classificação econômica baseada no PIB (Produto Interno Bruto) (INFOESCOLA).

Os índices do IDH são usados para verificar se o crescimento econômico de determinado país reflete desenvolvimento humano dos seus cidadãos (KAMBIENTAL).

Os critérios de avaliação consideram algumas variáveis relativas a:

- **saúde**: expectativa de vida das pessoas, condições de saneamento, nutrição e políticas públicas de saúde (campanhas de vacinação, educação em saúde, fornecimento de medicações e sistema público de saúde).
- **educação**: índices de analfabetismo, expectativa de escolaridade e tempo efetivo de escolaridade.
- **renda**: índice calculado pela razão entre a renda nacional bruta (RNB) e a paridade do poder de compra (PPC).

Com base nessas dimensões o país recebe uma pontuação entre 0 e 1, sendo classificado nas seguintes faixas:

[1] IDH - desenvolvido em 1990 pelos economistas Amartya Sen e Mahbub ul Haq é usado, desde 1993, pelo Programa das Nações Unidas para o Desenvolvimento (PNUD) no seu relatório anual (WIKIPEDIA).

●	Muito Alto	0,800 - 1,000
●	Alto	0,700 - 0,799
●	Médio	0,600 - 0,699
●	Baixo	0,500 - 0,599
●	Muito Baixo	0,000 - 0,499

O relatório do PNUD (Programa das Nações Unidas para o Desenvolvimento) em 2020 situa o Brasil, com índice de 0,765, na 84º posição do ranking mundial (entre 189 países), apresentando melhoras nas dimensões saúde e educação. Entretanto, se aplicado o índice Gini[2], que mede o grau de concentração de renda, despencaria 20 posições, ficando com pontuação de 0,570 (PNUD).

1.1.1 Renda *per capita* Maranhão

Segundo o Instituto Brasileiro de Geografia e Estatística (IBGE), o Maranhão vem se mantendo com o menor rendimento domiciliar *per capita* do país, com apenas R$ R$ 635,59 em 2019. A renda por pessoa no estado ficou abaixo da metade da média nacional, medida em R$ 1.439,00.

Figura 01 – Renda *per capita* Brasil em 2019

[2] Coeficiente de Gini - índice criado por Conrado Gini, matemático italiano, que calcula o grau de concentração de renda.

1.1.2 Pandemia Covid-19

A pandemia da Covid-19[3] causou instabilidades em todas as áreas, impactando diretamente na economia, saúde e educação em todo mundo.

No Brasil, a pandemia encontrou um cenário econômico que já passava por uma crise financeira e política. A pandemia agravou este cenário econômico e social afetando o emprego, as micro e pequenas empresas e os trabalhadores autônomos, fazendo com que diminuísse a renda e aumentasse a taxa de desemprego (PREIRA; RODRIGUES).

O efeito da pandemia de Covid-19 paralisou, parcial ou totalmente, diversas atividades econômicas. O consumo despencou e os investimentos encolheram.

O PIB 2020 foi o pior resultado da economia brasileira em 30 anos (ALVARENGA et al.). A aceleração da inflação e o avanço da pandemia contribuíram para a diminuição do consumo.

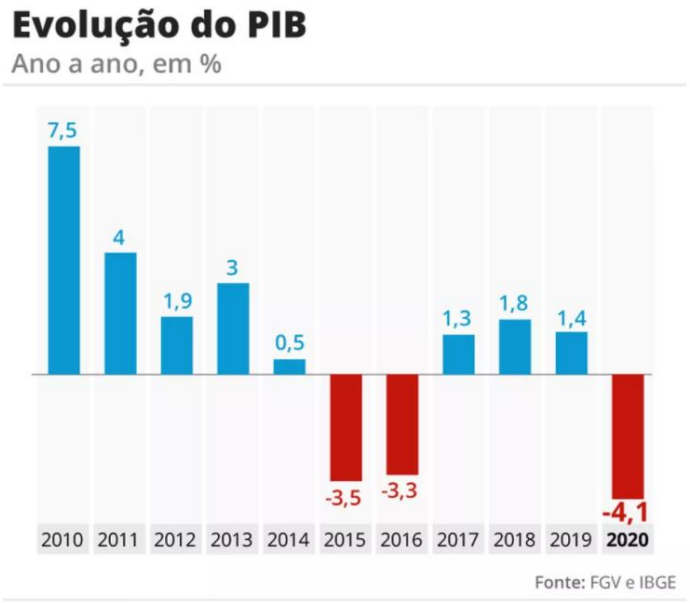


Figura 02 – Evolução do PIB

A taxa de desemprego no Brasil atingiu 13,9% da força de trabalho no quarto trimestre de 2020. A população ocupada diminuiu 8,9% em relação ao mesmo trimestre do ano anterior (SENADO).

[3] Covid-19 - pandemia causada pelo coronavírus SARS CoV-2. Surgiu no mundo em 2019 em Wuhan, China e em 2020 em São Paulo, Brasil.

1.1.3 Hesitação vacinal

A Organização Mundial de Saúde (OMS) define o atraso ou recusa à administração das vacinas como **hesitação vacinal**, apesar da disponibilidade.

"A hesitação vacinal compreende um amplo espectro de posturas, desde o receio até a total recusa, assumindo diversos níveis. É um fenômeno social complexo e reflete um ideal coletivo, que expressa seus questionamentos em dimensões como a liberdade individual, por exemplo" (COUTO, BARBIERE, MATOS). Envolve aspectos culturais, geográficos, psicossociais, econômicos, religiosos, políticos, fatores cognitivos e de gênero.

A hesitação vacinal apresenta três categorias inter-relacionadas:

- falta de confiança (na eficácia, na segurança, no sistema de saúde, nas motivações dos gestores e formuladores de políticas);
- complacência (baixa percepção do risco – "a vacinação não é necessária") e;
- falta de conveniência (disponibilidade, acessibilidade e campanha do serviço de imunização) (OLIVEIRA, BRUNO).

Uma elevada hesitação vacinal tem por consequência a baixa demanda da vacina e uma cobertura vacinal insatisfatória e preocupante.

1.2. O problema proposto

O objeto deste trabalho é a utilização de algoritmos de aprendizado de máquina (*machine learning*) não supervisionado para classificação dos municípios maranhenses e do índice de cobertura da vacinação contra a Covid-19 durante o ano de 2021. Analisar os fatores locais que podem motivar o baixo desempenho registrado, com base nos indicadores socioeconômicos dos municípios.

Para sistematização do problema foi utilizada a técnica dos 5W's.

Why (porque)

Em onze meses da campanha da vacinação contra Covid-19, apenas treze cidades maranhenses alcançaram o índice de 70% da população acima de 12 anos com o esquema completo de vacinação contra a Covid-19.

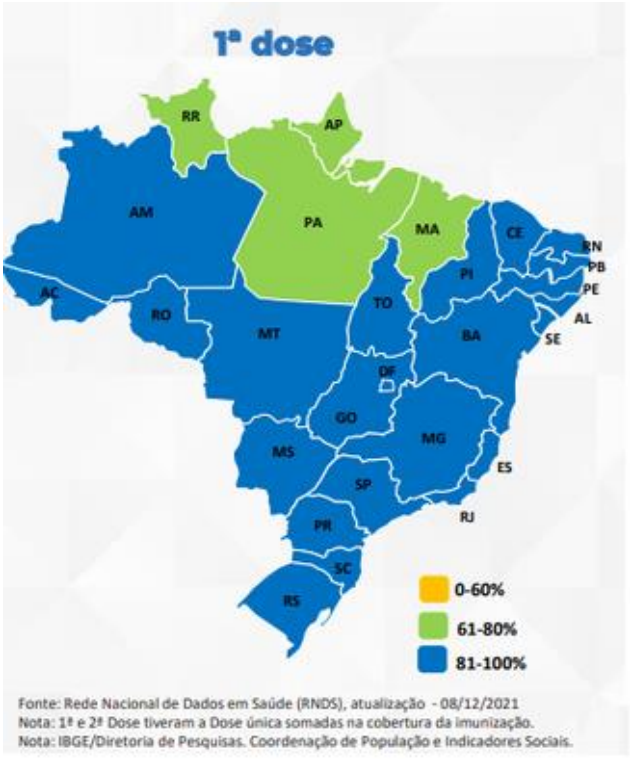


Figura 03 – Cobertura da Imunização Covid-19, Brasil, 2021

Para melhorar o índice de cobertura vacinal o poder público precisa definir a população-alvo, os grupos prioritários, garantir a distribuição e a segurança da vacina,

direcionar políticas públicas e orientar sobre o processo da vacinação. Para alcançar a população não imunizada é necessário conhecer suas condições sociais e econômicas.

Who (quem)

O objetivo deste trabalho é analisar dados extraídos em sites públicos (registros de vacinação Covid-19, base de dados do IDHM e população dos municípios) e construir modelos de aprendizagem de máquina de classificação.

What (o quê)

Com base nas informações do IDHM do município, esta análise pretende identificar as principais características que impedem um melhor desempenho da vacinação no Maranhão e usar modelos matemáticos para agrupar os municípios de acordo com o índice de cobertura vacinal, evidenciando similaridades econômicas e sociais.

When (quando)

A análise compreende o período de janeiro a dezembro de 2021.

Where (onde)

No estado do Maranhão e em seus 217 municípios.

1.3. Objetivo

Criação de modelos e utilização de algoritmos de classificação e agrupamento de dados para análise dos 217 municípios maranhenses.

Para o alcançar este objetivo e no desenvolvimento deste trabalho utilizou-se a linguagem de programação Python, versão 3.8.5, e diversas bibliotecas de software como Pandas, Numpy, Matplotlib, Seaborn e Scikit-learn.

O software Jupyter Notebook, versão 6.1.4, foi o ambiente desenvolvimento e execução dos comandos Python e Pandas, testes, tratamento e análise dos dados, criação dos modelos e verificação dos resultados, conforme figura 04.

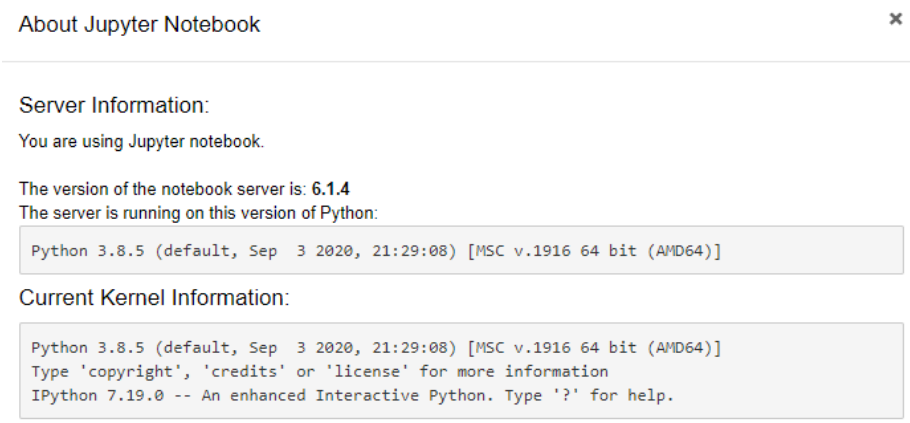


Figura 04 – Ambiente de desenvolvimento

2. COLETA DE DADOS

Os dados usados neste trabalho são públicos e estão disponíveis em sites abertos como openDataSUS, AtlasBR e IBGE.

a. Registros de Vacinação Covid-19

No site openDataSUS são encontrados os arquivos da Campanha Nacional de Vacinação contra a Covid-19 de todo Brasil. São várias informações sobre o paciente (anonimizado) e sobre a vacina aplicada.

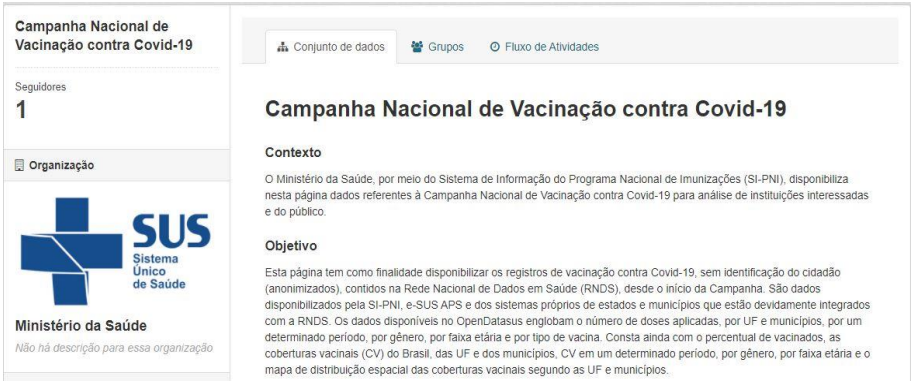
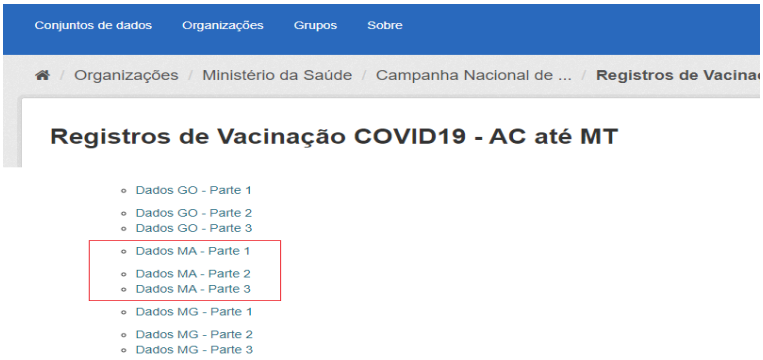


Figura 05 – Bases da Campanha Nacional de Vacinação contra Covid-19

Link: https://opendatasus.saude.gov.br/dataset/covid-19-vacinacao

O download dos arquivos foi realizado em 10/02/2022.



Em Registros de Vacinação COVID19 – AC até MT encontra-se os dados do MA. São três arquivos em formato “csv” (Character-separated values ou valores separados por delimitador) contendo as informações da vacinação, com cerca de 1,5 GB cada arquivo.

Após o download, foi executada a integração desta base de dados e está documentada no Jupyter Notebook **TCC_PUC_00 - JFarias_Vacina_MA - Junção das bases.ipynb**.

Na figura 06 está o dicionário de dados do dataset Registros de Vacinação COVID-19.

Ordem	Campo	Descrição
1	document_id	Identificador do documento
2	paciente_id	Identificador do vacinado
3	paciente_idade	Idade do vacinado
4	paciente_dataNascimento	Data de nascimento do vacinado
5	paciente_enumSexoBiologico	Sexo do vacinado (M=masculino, F=Feminino)
6	paciente_racaCor_codigo	Código da raça/cor do vacinado (1; 2; 3; 4; 99)
7	paciente_racaCor_valor	Descrição da raça/cor do vacinado (1 = Branca, 2 = Preta, 3 = Parda; 4 = Amarela; 99 = sem informação)
8	paciente_endereco_coibgeMunicipio	Código IBGE do município de endereçco do vacinado
9	paciente_endereco_coPais	Código do país de endereço do vacinado
10	paciente_endereco_nmMuncipio	Nome do município de endereçco do vacinado
11	paciente_endereco_nmPais	Nome do país de endereço do vacinado
12	paciente_endereco_uf	Sigla da UF de endereço do vacinado
13	paciente_endereco_cep	5 dígitos para anonimizado e 7 dígitos para identificado
14	paciente_nacionalidade_enumNacionalidade	Nacionalidade do vacinado
15	estabelecimento_valor	Código do CNEs do estabelecimento que realizou a vacinação
16	estabelecimento_razaoSocial	Nome/Razão Social do estabelecimento
17	estabelecimento_noFantasia	Nome fantasia do estabelecimento
18	estabelecimento_municipio_codigo	Código do município do estabelecimento
19	estabelecimento_municipio_nome	Nome do município do estabelecimento
20	estabelecimento_uf	Sigla da UF do estabelecimento
21	vacina_grupo_atendimento_codigo	Código do grupo de atendimento ao qual pertence o vacinado
22	vacina_grupo_atendimento_nome	Nome do grupo de atendimento ao qual pertence o vacinado
23	vacina_categoria_codigo	Código da categoria
24	vacina_categoria_nome	Nome da Categoria
25	vacina_lote	Número do lote da vacina
26	vacina_fabricante_nome	Nome do fabricante/fornecedor
27	vacina_fabricante_referencia	CNPJ do fabricante/fornecedor
28	vacina_dataAplicacao	Data de aplicação da vacina
29	vacina_descricao_dose	Descrição da dose
30	vacina_codigo	Código da vacina
31	vacina_nome	Nome da vacina/produto
32	sistema_origem	Nome do sistema de origem
33	data_importacao_rnds	Data de importação
34	id_sistema_origem	ID do sistema de origem

Figura 06 – Dicionário de dados Registros de Vacinação Covid-19

b. IDH Municípios

Esta base de dados foi obtida no site AtlasBR.

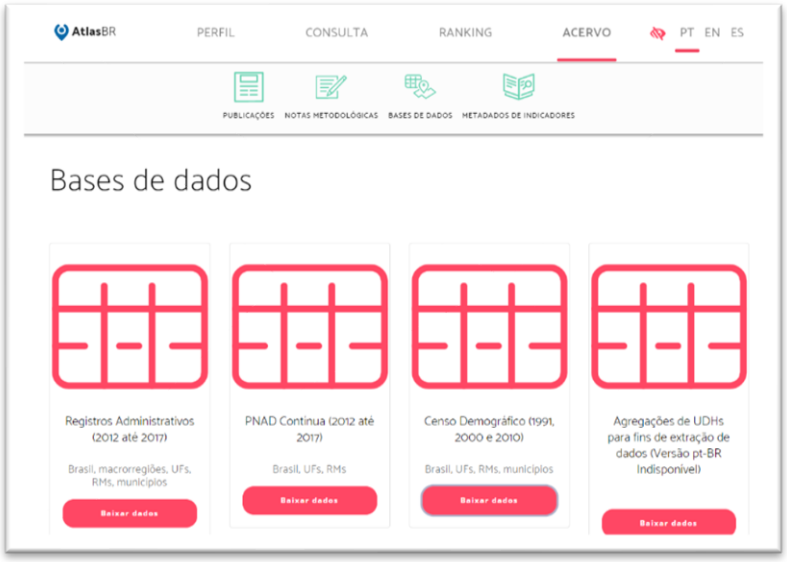


Figura 07 – Site AtlasBR - Bases de dados

Clicando na guia Acervo, e procurando em: Bases de dados -> Censo Demográfico (1991, 2000 e 2010) -> Baixar dados, pode ser feito o download do arquivo compactado ***Bases Censo.zip***, onde encontramos o arquivo ***Atlas 2013_municipal, estadual e Brasil.xlsx***. Neste arquivo, na planilha "**MUN 91-00-10**", temos diversas informações socioeconômicas dos municípios do todo Brasil, em 16.695 linhas e 237 colunas.

Link: http://www.atlasbrasil.org.br/acervo/biblioteca

Download dos arquivos realizado em 10/02/2022.

Segue o dicionário de dados (resumido) do dataset IDHM.

Nome do atributo	Descrição	Tipo
Codmun6	Código utilizado pelo IBGE para identificação do município (6 dígitos)	int64
Codmun7	Código utilizado pelo IBGE para identificação do município (7 dígitos)	int64
Município	Nome do município	object
MORT1	Mortalidade infantil	float64
T_AGUA	Percentual da população que vive em domicílios com água encanada	float64
T_LUZ	Percentual da população que vive em domicílios com energia elétrica	float64
AGUA_ESGOTO	Percentual de pessoas em domicílios com abastecimento de água e esgotamento sanitário inadequados	float64
PESOTOT	População total	int64
RENOCUP	Rendimento médio dos ocupados	float64
RDPC	Renda per capita média	float64
T_ANALF18M	Taxa de analfabetismo da população de 18 anos ou mais de idade	float64
I_ESCOLARIDADE	Escolaridade fundamental da população adulta	float64
IDHM	Índice de Desenvolvimento Humano Municipal	float64
IDHM_E	Índice de Desenvolvimento Humano Municipal - Dimensão Educação	float64
IDHM_L	Índice de Desenvolvimento Humano Municipal - Dimensão Longevidade	float64
IDHM_R	Índice de Desenvolvimento Humano Municipal - Dimensão Renda	float64

Figura 08 – Dicionário de dados IDHM

c. População estimada dos Municípios 2021

O IBGE disponibilizou na pasta Downloads as estimativas da população para 2021, figura 09.

Para este trabalho utilizou-se o arquivo ***estimativa_dou_2021.xls***, planilha "**Municípios**", que apresenta a população de 5.570 municípios brasileiros, com data de referência de 01 de julho de 2021.

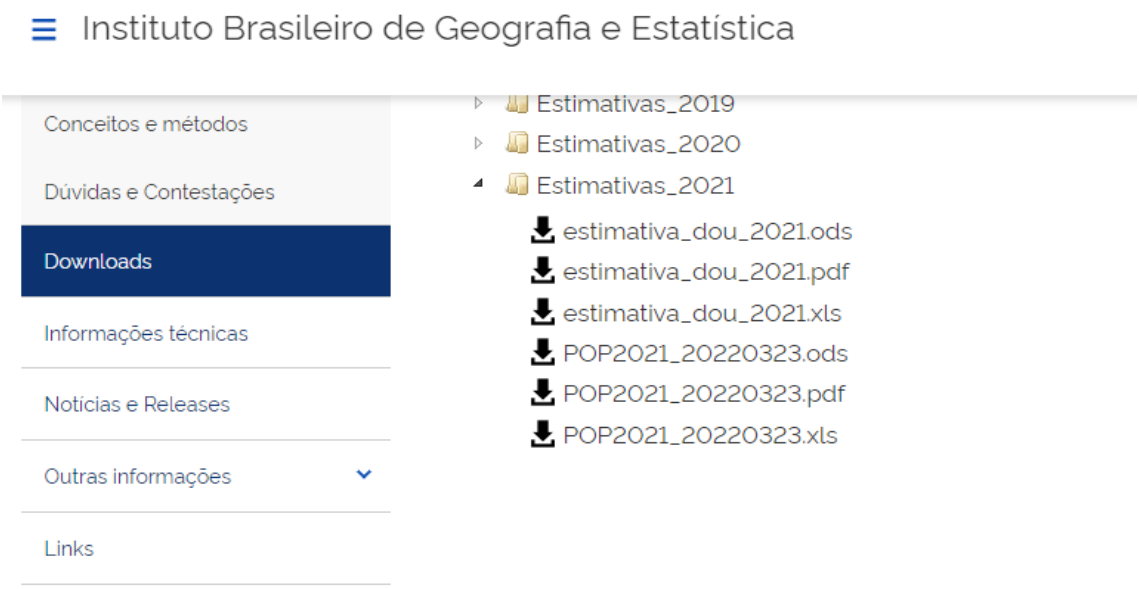


Figura 10 – Bases IBGE - estimativas população dos municípios para 2021

Link: https://www.ibge.gov.br/estatisticas/sociais/populacao/9103-estimativas-de-populacao.html?=&t=downloads

Dados obtidos em 10/02/2022.

Na figura abaixo temos o dicionário de dados deste dataset.

Nome da coluna	Descrição	Tipo
UF	Unidade da Federação	Texto
COD. UF	Código IBGE para a Unidade da Federação	Numérica
COD. MUNIC	Código IBGE para o Município	Numérica
NOME DO MUNICÍPIO	Nome do Município	Texto
POPULAÇÃO ESTIMADA	População estimada do Município	Numérica

Figura 11 – Dicionário de dados da base IBGE - estimativas população dos municípios para 2021

3. PROCESSAMENTO/TRATAMENTO DE DADOS

Os comandos utilizados nos procedimentos descritos nesta seção estão documentados no arquivo Jupyter Notebook **TCC_PUC_01 - JFarias_Vacina_MA - Tratamento de dados.ipynb**, disponível no repositório indicado no final deste trabalho.

A vacinação contra a Covid-19 no Maranhão iniciou em 18/01/2021, com a aplicação das primeiras doses da vacina Coronavac, produzida pelo Instituto Butantan em parceria com o laboratório chinês Sinovac, em cinco maranhenses:

- quatro profissionais da saúde e
- uma indígena.

A partir deste marco, tenta-se analisar a vacinação contra Covid-19 durante o ano 2021.

3.1 Bibliotecas

```
import pandas as pd
from datetime import datetime, date
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Para tratamento dos dados utilizou-se as principais bibliotecas Python:

- Pandas – manipulação de DataFrames;
- NumPy – biblioteca matemática;
- Matplotlib - biblioteca para criação de gráficos e visualizações de dados em geral;
- Seaborn - biblioteca de visualização de dados de alto nível.

3.2 Tratamento dataset Registros da Vacinação no Maranhão (openDataSUS)

3.2.1 Carregando a planilha com os registros da vacinação Covid-19/MA

Conforme falado na seção de coleta de dados, os três arquivos baixados no site openDataBR com os registros da vacinação do Maranhão, foram juntados (concatenados) em um só arquivo, em formato “csv”, e gravados com o nome **RegVacina_MA.csv**.

Para executar o tratamento dos dados foi feita a importação deste arquivo para o Pandas.

```
# Carregando a planilha registros da vacinação Covid-19/MA (extração 19/02/2022)
df_regvac = pd.read_csv('RegVacina_MA.csv', encoding='utf-8', sep=',', header=0, low_memory=False)
```

O arquivo foi convertido em um DataFrame formado de 9.564.664 linhas e 33 colunas.

3.2.2 Verificando os dados carregados

```
df_regvac.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9564664 entries, 0 to 9564663
Data columns (total 33 columns):
 #   Column                                  Dtype
---  ------                                  -----
 0   Unnamed: 0                              int64
 1   document_id                             object
 2   paciente_id                             object
 3   paciente_idade                          float64
 4   paciente_dataNascimento                 object
 5   paciente_enumSexoBiologico              object
 6   paciente_racaCor_codigo                 float64
 7   paciente_racaCor_valor                  object
 8   paciente_endereco_coIbgeMunicipio       object
 9   paciente_endereco_coPais                object
 10  paciente_endereco_nmMunicipio           object
 11  paciente_endereco_nmPais                object
 12  paciente_endereco_uf                    object
 13  paciente_endereco_cep                   object
 14  paciente_nacionalidade_enumNacionalidade object
 15  estabelecimento_valor                   int64
 16  estabelecimento_razaoSocial             object
 17  estalecimento_noFantasia                object
 18  estabelecimento_municipio_codigo        int64
 19  estabelecimento_municipio_nome          object
 20  estabelecimento_uf                      object
 21  vacina_grupoAtendimento_codigo          int64
 22  vacina_grupoAtendimento_nome            object
 23  vacina_categoria_codigo                 float64
 24  vacina_categoria_nome                   object
 25  vacina_lote                             object
 26  vacina_fabricante_nome                  object
 27  vacina_fabricante_referencia            object
 28  vacina_dataAplicacao                    object
 29  vacina_descricao_dose                   object
 30  vacina_codigo                           int64
 31  vacina_nome                             object
 32  sistema_origem                          object
dtypes: float64(3), int64(5), object(25)
memory usage: 2.4+ GB
```

Este DataFrame contém informações detalhadas sobre pacientes (anonimizados), estabelecimentos (locais) de vacinação e das vacinas aplicadas. Algumas informações não são de interesse para esta análise.

3.2.3 Definindo o escopo do projeto - vacinações 2021

```
# Definindo como base de estudo - vacinações em 2021
df_regvac2 =  df_regvac2.query('vacina_dataAplicacao >= "2021-01-01" & vacina_dataAplicacao <= "2021-12-31"')

pd.unique(df_regvac2['vacina_dataAplicacao'].dt.year)
```

```
array([2021], dtype=int64)
```

Tendo como proposta analisar as vacinações ocorridas em 2021 e em virtude do download ter sido feito em fevereiro/2022, houve a necessidade de filtrar e eliminar os registros de 2022.

Esta ação reduziu o DataFrame para 8.710.112 linhas em 33 colunas.

3.2.4 Verificando a origem dos pacientes registrados

No processo de conhecer os dados a serem trabalhados, verificou-se a procedência dos vacinados.

Embora a grande maioria (92,2%) sejam maranhenses, existem pacientes de todas as unidades federativas do Brasil, residentes ou passageiros. Observou-se também alguns dados faltantes ou registrados incorretamente.

	paciente_endereco_uf	paciente_endereco_coIbgeMunicipio	paciente_endereco_nmMunicipio	vacina_dataAplicacao	vacina_nome
10622	XX	999999	INVALIDO	2021-04-01	COVID-19 SINOVAC/BUTANTAN - CORONAVAC
50087	XX	999999	INVALIDO	2021-05-01	COVID-19 ASTRAZENECA/FIOCRUZ - COVISHIELD
129978	XX	999999	INVALIDO	2021-05-24	COVID-19 ASTRAZENECA - ChAdOx1-S
192395	XX	999999	INVALIDO	2021-06-13	COVID-19 ASTRAZENECA/FIOCRUZ - COVISHIELD
204649	XX	999999	INVALIDO	2021-10-20	COVID-19 ASTRAZENECA/FIOCRUZ - COVISHIELD

A informação da naturalidade dos pacientes com XX no atributo **paciente_endereco_uf** poderia ser recuperada utilizando-se os dois primeiros dígitos do atributo código IGBE **paciente_endereco_coIbgeMunicipio**, mas neste caso, esta opção se mostrou inviável.

Optou-se pela exclusão destes registros.

3.2.5 Excluindo colunas que não serão usadas no projeto

```
# Eliminando as colunas que não serão tratadas neste trabalho

col_drop = ['Unnamed: 0','document_id','paciente_racaCor_codigo','paciente_endereco_coPais','paciente_endereco_nmPais',
            'paciente_endereco_uf','paciente_endereco_cep','paciente_nacionalidade_enumNacionalidade','estabelecimento_valor',
            'estabelecimento_razaoSocial','estalecimento_noFantasia','estabelecimento_municipio_codigo','estabelecimento_uf',
            'vacina_grupoAtendimento_codigo','vacina_grupoAtendimento_nome','vacina_categoria_codigo','vacina_lote',
            'vacina_fabricante_nome','vacina_fabricante_referencia', 'vacina_codigo','sistema_origem']

df_regvac2 = df_regvac2.drop(col_drop, axis=1)
```

Foram eliminadas algumas colunas no processo de limpeza dos dados. Principalmente informações e códigos específicos, importantes apenas para a gestão da saúde.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 8647943 entries, 0 to 9564663
Data columns (total 12 columns):
 #   Column               Dtype         
---  ------               -----         
 0   paciente_id          object        
 1   paciente_idade       float64       
 2   paciente_dt_nasc     datetime64[ns]
 3   paciente_sexo        object        
 4   paciente_raca        object        
 5   paciente_cod_municp  object        
 6   paciente_nm_municp   object        
 7   vacina_nm_municp     object        
 8   vacina_categoria     object        
 9   vacina_dt_aplic      datetime64[ns]
 10  vacina_desc_dose     object        
 11  vacina_nome          object        
dtypes: datetime64[ns](2), float64(1), object(9)
memory usage: 857.7+ MB
```

Com isso reduziu-se o DataFrame para 12 as colunas.

3.2.6 Ajustando o dataset

Para facilitar a interpretação dos dados, o nome comercial das vacinas foi substituído pelo nome conhecido popularmente, p.ex. COVID-19 ASTRAZENECA/FIOCRUZ – COVISHIELD foi trocado por AstraZeneca, e foram renomeadas também algumas colunas, para traduzir mais expressamente seu conteúdo.

```
dic_nmvacina = {'COVID-19 ASTRAZENECA/FIOCRUZ - COVISHIELD':'AstraZeneca','COVID-19 SINOVAC/BUTANTAN - CORONAVAC':'Coronavac',
                'COVID-19 PEDIÁTRICA - PFIZER COMIRNATY':'Pfizer','COVID-19 PFIZER - COMIRNATY':'Pfizer',
                'COVID-19 ASTRAZENECA - ChAdOx1-S':'AstraZeneca','COVID-19 JANSSEN - Ad26.COV2.S':'Janssen'}
df_regvac2 = df_regvac2.replace({'vacina_nome':dic_nmvacina})
```

3.2.7 Excluindo registros de vacinação em crianças abaixo de 12 anos

A vacinação pediátrica no Maranhão foi iniciada oficialmente em 20/01/2022. Mesmo assim, foram encontrados registros de vacinação em pacientes abaixo dos 12 anos em 2021.

Optou-se pela exclusão destes registros.

3.2.8 Tratamento da categoria da vacina

O atributo **vacina_categoria** informa em qual grupo o paciente se enquadrava ao ser atendido na campanha, ou seja, se a pessoa vacinada possuía ou não prioridades, de acordo com as regras estabelecidas pelo Ministério da Saúde. Observou-se que este atributo estava fortemente relacionado ao paciente e não a vacina, como originalmente registrado.

Resolveu-se renomear este atributo para **paciente_categoria**.

```
Faixa Etária                                      6702409
Comorbidades                                       505453
Trabalhadores de Saúde                             430448
Trabalhadores da Educação                          354993
Povos e Comunidades Tradicionais                   255699
Trabalhadores Industriais                           98787
Gestantes                                           49228
Trabalhadores de Transporte                         39886
Forças de Segurança e Salvamento                    38344
Pessoas com Deficiência                             29897
Pessoas de 60 anos ou mais institucionalizadas      27369
Povos Indígenas                                     27357
Outros                                              20508
Trabalhadores Portuários                            17675
Trabalhadores de Limpeza Urbana                     13309
Puérperas                                           11191
Funcionário do Sistema de Privação de Liberdade      9058
População Privada de Liberdade                       7150
Forças Armadas (membros ativos)                      2235
Pessoas em Situação de Rua                            398
Name: vacina_categoria, dtype: int64
```

3.2.9 Verificando registros duplicados

A quantidade de registros precisava ser trabalhada, pois o volume de dados, mais de 8 milhões de registros, dificultaria as etapas seguintes desta análise.

```
# Verificando a quantidade de registros duplicados
df_regvac2.duplicated().sum()
```

```
7613
```

Inicialmente foram verificadas possíveis duplicidades nos registros e encontrou-se 7.613 registros duplicados.

Os registros foram eliminados.

Depois verificou-se quantos registros existiam por paciente.

Normalmente, um paciente deveria ter 3 registros de vacinação, correspondentes as 1ª e 2ª dose e dose reforço.

```
df_regvac2.paciente_id.value_counts()

f30dce82db0c8fc285117e50531f40b6797ac8362e653cb692fcf459fb1b9b62    8
4d7f46a384ef4a4b0b2ae21bc51c8eadcf81add910f73d4a51cad51b48403690    8
357fb31c6bda2296e11d6c2545bec1dd2e443a924985ccd954ae73475c6c45fb    8
b3952f5aafaa7274c01d40a9623b4a8e2cfcdc5d29ac3a9fd8a6cbcfb5773d64    7
6b41ab6ba1648611621059 65f40be6843fb4911b28ee1e5cf2546ee2e62f454b   7
                                                                   ..
0b6be027a8c5294e5ba1c4cf6c594e618db1bf99528b81c178f1a448908f1e1b    1
8330d591d22c7a65a6a05ade353c5b1c7a846909f4e37f792d367531da05d113    1
d7e49a98d49f578def0224786eaeeeac04d673cda30be0c218611d68794eaf72b    1
b0378975e983b3159444e9d08011fe69fdca1282b6139a84ff650e845745b7b7    1
94b1e1669fc7a38b904e000038481a8ca902fbdadac45ff5986b0e9141fa5236    1
Name: paciente_id, Length: 4614250, dtype: int64
```

Como observado acima, a quantidade de pacientes na base é de 4.614.250. Entretanto, aparecem pacientes com até 8 registros de vacinação, muito acima do esperado.

Uma solução para o problema poderia ser agregar as doses recebidas em um único registro por paciente, mas o objetivo do trabalho é analisar os pacientes que receberam a 1ª dose e não completaram o ciclo de vacinação. Então, resolveu-se distribuir os registros em novos DataFrames, separando as doses recebidas.

3.2.10 Separando os registros por doses

3.2.10.1 DataFrame 1ª dose

O novo DataFrame recebeu 4.432.440 registros, ficando apenas com as informações da 1ª dose. Os atributos com as informações da vacina foram renomeados com o final _d1 para não haver sobreposição de dados numa futura junção dos DataFrames.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4432440 entries, 6274449 to 4453538
Data columns (total 12 columns):
 #   Column               Dtype
---  ------               -----
 0   paciente_id          object
 1   paciente_idade       float64
 2   paciente_dt_nasc     datetime64[ns]
 3   paciente_sexo        object
 4   paciente_raca        object
 5   paciente_cod_municp  object
 6   paciente_nm_municp   object
 7   paciente_categoria   object
 8   vacina_nm_municp_d1  object
 9   vacina_dt_aplic_d1   datetime64[ns]
 10  vacina_desc_dose_d1  object
 11  vacina_nome_d1       object
dtypes: datetime64[ns](2), float64(1), object(9)
memory usage: 439.6+ MB
```

3.2.10.2 DataFrame 2ª dose

No caso do DataFrame da 2ª dose, foi preciso observar a existência de registros referentes a dose da vacina Janssen (Dose), tomada em dose única, e outras “doses adicionais”.

```
2ª Dose                 3539385
Dose                     110987
Dose Adicional            39619
1ª Dose Revacinação           2
Name: vacina_desc_dose, dtype: int64
```

Todos estes registros foram reunidos no DataFrame 2º dose que representa os pacientes que completaram o ciclo de vacinação. Foram 3.689.993 registros.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 3689993 entries, 0 to 3689992
Data columns (total 12 columns):
 #   Column               Dtype
---  ------               -----
 0   paciente_id          object
 1   paciente_idade       float64
 2   paciente_dt_nasc     datetime64[ns]
 3   paciente_sexo        object
 4   paciente_raca        object
 5   paciente_cod_municp  object
 6   paciente_nm_municp   object
 7   paciente_categoria   object
 8   vacina_nm_municp_d2  object
 9   vacina_dt_aplic_d2   datetime64[ns]
 10  vacina_desc_dose_d2  object
 11  vacina_nome_d2       object
dtypes: datetime64[ns](2), float64(1), object(9)
memory usage: 337.8+ MB
```

Os atributos com as informações da vacina foram renomeados com o final _d2.

3.2.11 Remontando o DataFrame dos registros de vacinação

Separados os DataFrames referentes as doses recebidas temos o problema dos pacientes que constam em ambos datasets. Ou seja, que receberam as duas doses.

Para retirar o registro destes pacientes do DataFrame da 1ª dose deveria ser aplicada uma junção tipo LEFT EXCLUDING JOIN, figura 12, não disponível no Pandas.

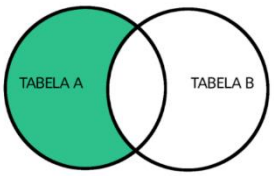


Figura 12 – Junção LEFT EXCLUDING JOIN

Optou-se por fazer uma junção tipo FULL JOIN, figura 13, e depois remover os registros com informações da 2ª dose. Obteve-se efeito semelhante a junção LEFT EXCLUDING JOIN.

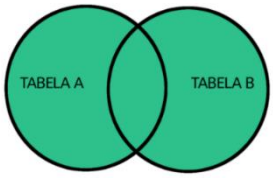


Figura 13 – Junção FULL JOIN

```
df_regvac3 = df_regvac3.query('vacina_desc_dose_d2 != "2ª Dose" ')
```

Executados estes comandos restaram no DataFrame de trabalho (df_regvac3) 973.892 registros de pacientes que receberam só a 1ª dose da vacina.

Foi verificado novamente possíveis duplicidades de registros e encontrou-se 4.520 duplicações, que foram eliminadas.

```
# Excluindo registros repetidos
df_regvac3 = df_regvac3.drop_duplicates(subset=ck_duplic, keep='first', inplace=False)
```

Restava considerar o intervalo entre as doses.

O Ministério da Saúde estabelece como intervalo entre as doses, 28 dias para a vacina Coronavac e 60 dias no caso das vacinas Pfizer e AstraZeneca.

Foram retirados da base os registros após 01/11/2021 para os pacientes que receberam as vacinas Pfizer e AstraZeneca e os registros após 01/12/2021 dos pacientes que receberam a vacina Coronavac.

Estes procedimentos reduziram o DataFrame para 822.693 linhas com 12 colunas, estando pronto para a junção com as outras bases.

```
df_regvac3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 822693 entries, 0 to 822692
Data columns (total 12 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   paciente_id         822693 non-null  object
 1   paciente_idade      822693 non-null  float64
 2   paciente_dt_nasc    822693 non-null  datetime64[ns]
 3   paciente_sexo       822693 non-null  object
 4   paciente_raca       822693 non-null  object
 5   paciente_cod_municp 822693 non-null  object
 6   paciente_nm_municp  822693 non-null  object
 7   paciente_categoria  822371 non-null  object
 8   vacina_nm_municp    822693 non-null  object
 9   vacina_dt_aplic     822693 non-null  datetime64[ns]
 10  vacina_desc_dose    822693 non-null  object
 11  vacina_nome         822693 non-null  object
dtypes: datetime64[ns](2), float64(1), object(9)
memory usage: 75.3+ MB
```

3.3 Tratamento dataset IDH Municípios (AtlasBR)

Foi executada a importação do arquivo **Atlas 2013_municipal, estadual e Brasil.xlsx** e gerado um DataFrame Pandas com 16.695 linhas e 237 colunas.

Inicialmente foi filtrado ano (2010) e a UF (MA). Em seguida, foram selecionas os atributos de interesse.

Com estes procedimentos obteve-se um DataFrame com 217 linhas (total de municípios maranhenses) e 16 colunas.

```
df_idhm3.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 217 entries, 0 to 216
Data columns (total 16 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   municp_cod6            217 non-null    int64
 1   municp_codibge         217 non-null    int64
 2   municp_nome            217 non-null    object
 3   municp_mort_inf        217 non-null    float64
 4   municp_tx_analf        217 non-null    float64
 5   municp_rend_pcap       217 non-null    float64
 6   municp_rend_med_ocup   217 non-null    float64
 7   municp_domic_agua      217 non-null    float64
 8   municp_domic_luz       217 non-null    float64
 9   municp_domic_agua&esg  217 non-null    float64
 10  municp_pop_total       217 non-null    int64
 11  municp_niv_escol_pop   217 non-null    float64
 12  municp_idhm            217 non-null    float64
 13  municp_idhm_educ       217 non-null    float64
 14  municp_idhm_longev     217 non-null    float64
 15  municp_idhm_renda      217 non-null    float64
dtypes: float64(12), int64(3), object(1)
memory usage: 27.2+ KB
```

3.4 Tratamento dataset Estimativa População 2021 (IBGE)

Importado o arquivo **estimativa_dou_2021.xls**, gerou-se um DataFrame com 5.593 li nhas e 5 colunas.

```
df_pop.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 5593 entries, 0 to 5592
Data columns (total 5 columns):
 #   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   UF                  5591 non-null   object
 1   COD. UF             5570 non-null   object
 2   COD. MUNIC          5570 non-null   object
 3   NOME DO MUNICÍPIO   5570 non-null   object
 4   POPULAÇÃO ESTIMADA  5570 non-null   object
dtypes: object(5)
memory usage: 218.6+ KB
```

Foi criado o atributo **cod_municp**, no padrão das outras bases, para servir como chave na junção das bases.

3.4.1 Criando atributo cobertura vacinal

Foram criados os atributos **municp_qtdevac** e **municp_cobert**. O primeiro é a quantidade de vacinas aplicadas no município e o segundo é a razão entre esta quantidade e

a população do município (municp_cobert = municp_qtdevac/pop_estimada). Desta forma, a cobertura vacinal passou a fazer parte das informações ligadas ao município.

DataFrame pronto para junção de bases.

```
df_pop3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 217 entries, 0 to 216
Data columns (total 5 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   pop_estimada    217 non-null    int32
 1   cod_municp      217 non-null    int32
 2   nm_municp       217 non-null    object
 3   municp_qtdevac  217 non-null    int64
 4   municp_cobert   217 non-null    float64
dtypes: float64(1), int32(2), int64(1), object(1)
memory usage: 6.9+ KB
```

3.5 Junção dos datasets

3.5.1 Junção datasets IDHM e População dos Municípios

Foram juntados inicialmente as bases IDHM e População 2021, resultando em um DataFrame com 217 linhas e 21 colunas.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 217 entries, 0 to 216
Data columns (total 21 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   pop_estimada           217 non-null    int32
 1   cod_municp             217 non-null    int32
 2   nm_municp              217 non-null    object
 3   municp_qtdevac         217 non-null    int64
 4   municp_cobert          217 non-null    float64
 5   municp_cod6            217 non-null    int64
 6   municp_codibge         217 non-null    int64
 7   municp_nome            217 non-null    object
 8   municp_mort_inf        217 non-null    float64
 9   municp_tx_analf        217 non-null    float64
 10  municp_rend_pcap       217 non-null    float64
 11  municp_rend_med_ocup   217 non-null    float64
 12  municp_domic_agua      217 non-null    float64
 13  municp_domic_luz       217 non-null    float64
 14  municp_domic_agua&esg  217 non-null    float64
 15  municp_pop_total       217 non-null    int64
 16  municp_niv_escol_pop   217 non-null    float64
 17  municp_idhm            217 non-null    float64
 18  municp_idhm_educ       217 non-null    float64
 19  municp_idhm_longev     217 non-null    float64
 20  municp_idhm_renda      217 non-null    float64
dtypes: float64(13), int32(2), int64(4), object(2)
memory usage: 35.6+ KB
```

Eliminados atributos semelhantes e renomeados outros, resultou uma base com 217 linhas e 18 colunas.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 217 entries, 0 to 216
Data columns (total 18 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   municp_populacao       217 non-null    int32
 1   municp_codigo          217 non-null    int32
 2   municp_qtdevac         217 non-null    int64
 3   municp_cobert          217 non-null    float64
 4   municp_cod6            217 non-null    int64
 5   municp_nome            217 non-null    object
 6   municp_mort_inf        217 non-null    float64
 7   municp_tx_analf        217 non-null    float64
 8   municp_rend_pcap       217 non-null    float64
 9   municp_rend_med_ocup   217 non-null    float64
 10  municp_domic_agua      217 non-null    float64
 11  municp_domic_luz       217 non-null    float64
 12  municp_domic_agua&esg  217 non-null    float64
 13  municp_niv_escol_pop   217 non-null    float64
 14  municp_idhm            217 non-null    float64
 15  municp_idhm_educ       217 non-null    float64
 16  municp_idhm_longev     217 non-null    float64
 17  municp_idhm_renda      217 non-null    float64
dtypes: float64(13), int32(2), int64(2), object(1)
memory usage: 30.5+ KB
```

3.5.2 Junção datasets PopIDHM e Registros de Vacinação

Após a junção das bases PopIDHM e Registros de Vacinação, foram removidos os atributos de código IBGE que serviram de chave para as junções, ficando o DataFrame final com 737.004 linhas e 28 colunas.

Esta será a base utilizada nas próximas etapas deste trabalho.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 737004 entries, 0 to 737003
Data columns (total 28 columns):
 #   Column                 Non-Null Count   Dtype
---  ------                 --------------   -----
 0   paciente_id            737004 non-null  object
 1   paciente_idade         737004 non-null  float64
 2   paciente_dt_nasc       737004 non-null  datetime64[ns]
 3   paciente_sexo          737004 non-null  object
 4   paciente_raca          737004 non-null  object
 5   paciente_nm_municp     737004 non-null  object
 6   paciente_categoria     736696 non-null  object
 7   vacina_nm_municp       737004 non-null  object
 8   vacina_dt_aplic        737004 non-null  datetime64[ns]
 9   vacina_desc_dose       737004 non-null  object
 10  vacina_nome            737004 non-null  object
 11  municp_populacao       737004 non-null  int32
 12  municp_codigo          737004 non-null  int32
 13  municp_qtdevac         737004 non-null  int64
 14  municp_cobert          737004 non-null  float64
 15  municp_nome            737004 non-null  object
 16  municp_mort_inf        737004 non-null  float64
 17  municp_tx_analf        737004 non-null  float64
 18  municp_rend_pcap       737004 non-null  float64
 19  municp_rend_med_ocup   737004 non-null  float64
 20  municp_domic_agua      737004 non-null  float64
 21  municp_domic_luz       737004 non-null  float64
 22  municp_domic_agua&esg  737004 non-null  float64
 23  municp_niv_escol_pop   737004 non-null  float64
 24  municp_idhm            737004 non-null  float64
 25  municp_idhm_educ       737004 non-null  float64
 26  municp_idhm_longev     737004 non-null  float64
 27  municp_idhm_renda      737004 non-null  float64
dtypes: datetime64[ns](2), float64(14), int32(2), int64(1), object(9)
memory usage: 157.4+ MB
```

4. ANÁLISE E EXPLORAÇÃO DOS DADOS

Análise exploratória de dados (AED) é uma etapa muito importante em ciência de dados para analisar e investigar conjuntos de dados e resumir suas principais características usando métodos de visualização de dados. É essencial que o cientista de dados seja capaz de entender a natureza dos dados.

Ela permite determinar a melhor forma de controlar as fontes de dados para obter as respostas que você precisa, tornando mais fácil descobrir padrões, detectar anomalias, testar hipóteses ou verificar suposições(x).

Os comandos utilizados nos procedimentos descritos nesta seção estão documentados em detalhes no arquivo Jupyter Notebook **TCC_PUC_02 - JFarias_Vacina_MA - AED.ipynb**, disponível no repositório indicado no final deste trabalho.

4.1 Bibliotecas

```
import pandas as pd
from datetime import datetime, date
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

Utilizadas as mesmas bibliotecas Python da etapa anterior.

4.2 Carregando a planilha de trabalho

```
#Carregando a planilha registros da vacinação Covid-19/MA
df_aed = pd.read_csv('RegVacinaMA_TratDados.csv', encoding='utf-8', sep=',', header=0, low_memory=False)
```

Importado para este notebook o arquivo final da etapa Tratamento de dados.

4.3 Análise Univariada

4.3.1 Variáveis Qualitativas

4.3.1.1 Análise da faixa etária dos pacientes

```
df_aed['paciente_class_idade'] = pd.cut(df_aed['paciente_idade'],
                                        bins=[11.0, 19.0, 29.0, 39.0, 49.0, 59.0, 69.0, 79.0, 129.0],
                                        labels=['12-18 anos','19-29 anos','30-39 anos','40-49 anos','50-59 anos','60-69 anos',
                                                '70-79 anos','80 ou mais'])
```

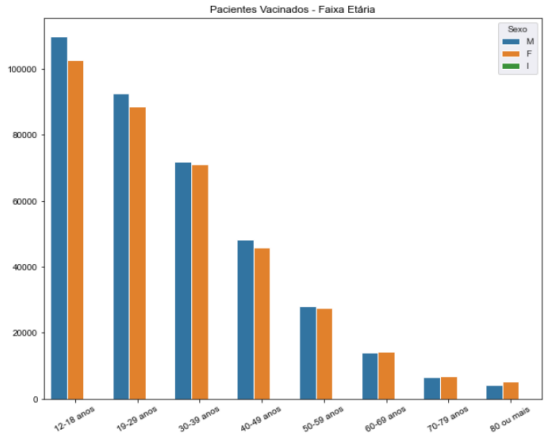
Criado o atributo de classificação de idades (**paciente_class_idade**) que estabelece as faixas etárias dos pacientes.

Lembrete: Estes são os registros dos pacientes que receberam a 1ª dose mas não retornaram para receber a 2ª dose durante o ano de 2021.

```
fe = df_aed.paciente_class_idade.value_counts()
tfe =df_aed.paciente_class_idade.count()
round(fe/tfe*100,2)
```

```
12-18 anos    28.85
19-29 anos    24.56
30-39 anos    19.41
40-49 anos    12.76
50-59 anos     7.53
60-69 anos     3.82
70-79 anos     1.81
80 ou mais     1.26
Name: paciente_class_idade, dtype: float64
```

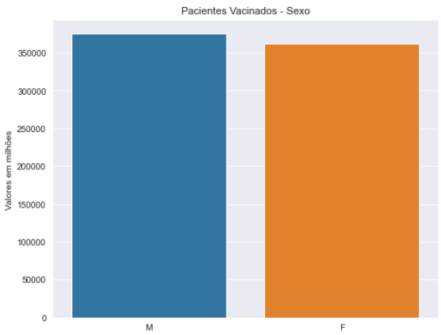
Observa-se que a faixa de 12-18 anos foi a mais vacina com 28,85% das aplicações.



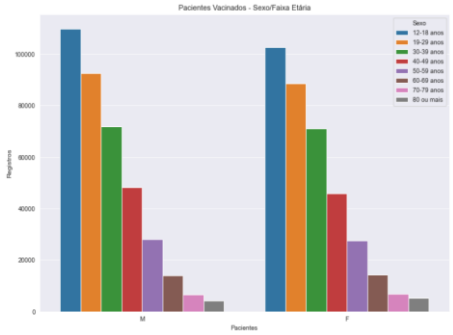
4.3.1.2 Analisando o sexo declarado

```
df_aed.paciente_sexo.value_counts()
```

```
M    375260
F    361743
Name: paciente_sexo, dtype: int64
```



Nota-se uma pequena diferença dos pacientes masculinos (50,92%) sobre os femininos (49,08%).



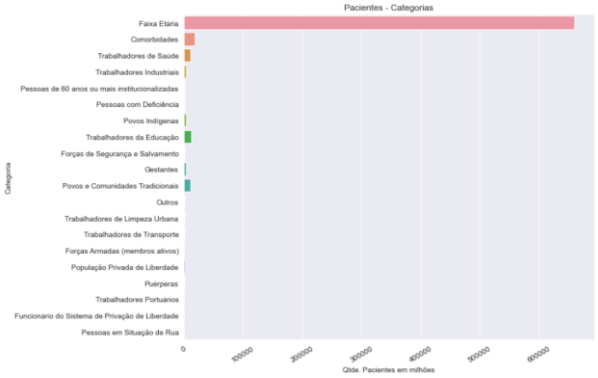

```
paciente_class_idade  paciente_sexo
12-18 anos            M                109965
                      F                102647
19-29 anos            M                 92508
                      F                 88505
30-39 anos            M                 71876
                      F                 71185
40-49 anos            M                 48202
                      F                 45811
50-59 anos            M                 28011
                      F                 27513
60-69 anos            F                 14173
                      M                 13992
70-79 anos            F                  6769
                      M                  6560
80 ou mais            F                  5140
                      M                  4146
Name: paciente_sexo, dtype: int64
```

A partir da faixa dos 60 anos há uma inversão e as mulheres são, ligeiramente, mais numerosas.

4.3.1.3 Analisando o atributo categoria

A ordem de aplicação da vacina seguiu as normas estabelecidas pelo PNI (Programa Nacional de Imunização).

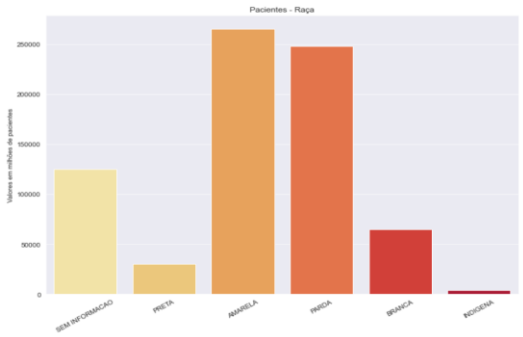
Algumas categorias como trabalhadores da saúde, povos indígenas, comorbidades e puérperas tiveram a prioridade na aplicação das doses.



Neste grupo, a maioria dos não imunizados estão da categoria Faixa Etária (89,62%), seguidos dos pacientes com comorbidade (2,47%), trabalhadores da educação (1,73%) e Povos e Comunidades Tradicionais (1,50%).

4.3.1.4 Análise da raça declarada

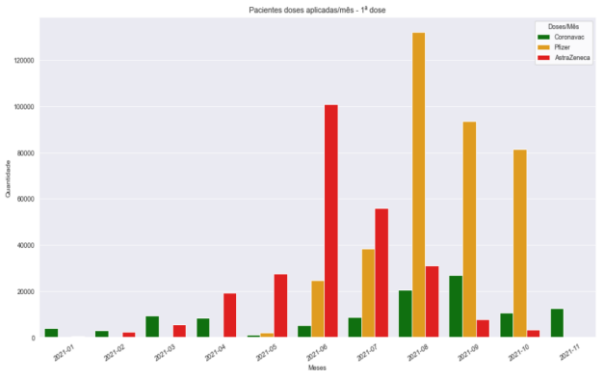
```
AMARELA          265222
PARDA            247823
SEM INFORMACAO   124965
BRANCA            64931
PRETA             30346
INDIGENA           3716
Name: paciente_raca, dtype: int64
```



Resultado não esperado uma vez que a raça negra é predominante na população maranhense. Uma correção poderia ser a soma dos declarados pardos e pretos.

4.3.1.5 Análise da vacinação por mês

```
2021-08    183503
2021-06    131027
2021-09    128310
2021-07    102814
2021-10     95567
2021-05     30743
2021-04     27754
2021-03     15003
2021-11     12522
2021-02      5415
2021-01      4345
Name: vacina_dose_mes, dtype: int64
```




```
Pfizer         372275
AstraZeneca    254036
Coronavac      110692
Name: vacina_nome, dtype: int64
```

Os meses com maior concentração de vacinações foram agosto, junho, setembro e julho/21.

A Pfizer for a vacina mais administrada nos pacientes deste grupo.

4.3.2 Variáveis Quantitativas

4.3.2.1 Análise da idade dos pacientes

```
df_aed.paciente_idade.describe()
```

```
count    737003.000000
mean         31.598502
std          16.313271
min          12.000000
25%          18.000000
50%          28.000000
75%          41.000000
max         128.000000
Name: paciente_idade, dtype: float64
```

A média de idade do grupo é 31, a mediana 28 e o desvio padrão 16 anos. A idade máxima 128 e a mínima 12 anos.

4.3.2.2 Análise dos indicadores socioeconômicos dos municípios

4.3.2.2.1 Índice IDHM

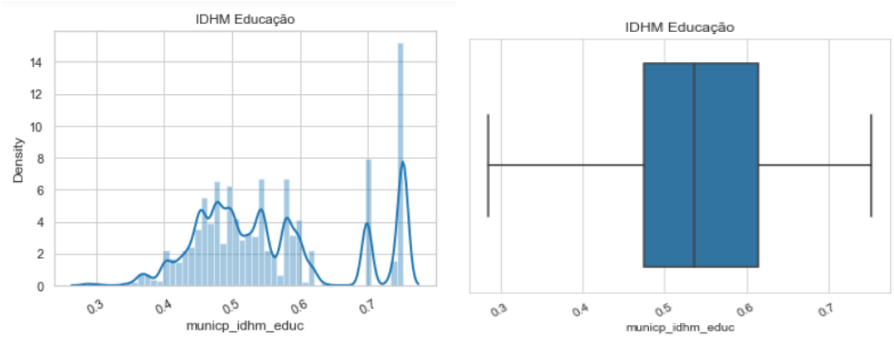
```
# Aplicando a classificação no índice IDHM
#
# A classificação padrão do indice IDHM se divide em 5 faixas de desenvolvimento (entre 0 e 1):
# {Muito alto  : 0,800 - 1,000
#  Alto        : 0,700 - 0,799
#  Médio       : 0,600 - 0,699
#  Baixo       : 0,500 - 0,599
#  Muito Baixo : 0,000 - 0,499}

# Criando atributo class para os índices IDHM
df_aed['municp_idhm_class'] = pd.cut(df_aed['municp_idhm'], bins=[0,0.5,0.6,0.7,0.8,1],
                                     labels=['Muito Baixo','Baixo','Médio','Alto','Muito Alto'])
df_aed['municp_idhm_class_educ'] = pd.cut(df_aed['municp_idhm_educ'], bins=[0,0.5,0.6,0.7,0.8,1],
                                     labels=['Muito Baixo','Baixo','Médio','Alto','Muito Alto'])
df_aed['municp_idhm_class_longev'] = pd.cut(df_aed['municp_idhm_longev'], bins=[0,0.5,0.6,0.7,0.8,1],
                                     labels=['Muito Baixo','Baixo','Médio','Alto','Muito Alto'])
df_aed['municp_idhm_class_renda'] = pd.cut(df_aed['municp_idhm_renda'], bins=[0,0.5,0.6,0.7,0.8,1],
                                     labels=['Muito Baixo','Baixo','Médio','Alto','Muito Alto'])
```

Aplicada a classificação IDH nos índices sociais para tornar mais evidente as condições socioeconômicas do município.

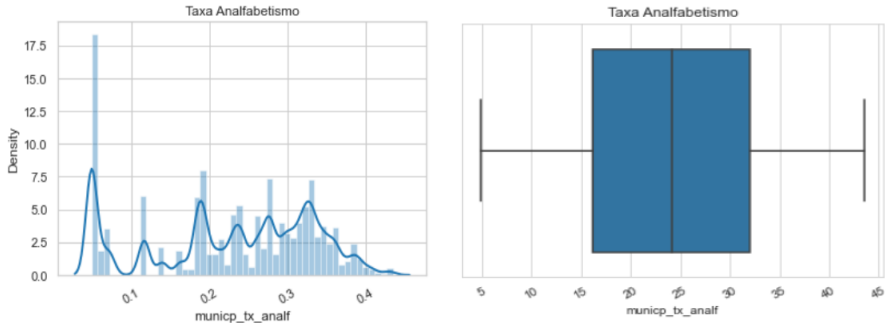
4.3.2.2.2 Indicadores de educação

	municp_idhm_educ	municp_tx_analf	municp_niv_escol_pop
count	737003.000000	737003.000000	737003.000000
mean	0.557906	22.843686	0.435874
std	0.111330	10.706437	0.161220
min	0.286000	4.920000	0.154000
25%	0.475000	16.180000	0.311000
50%	0.536000	24.160000	0.379000
75%	0.615000	31.980000	0.488000
max	0.752000	43.530000	0.735000

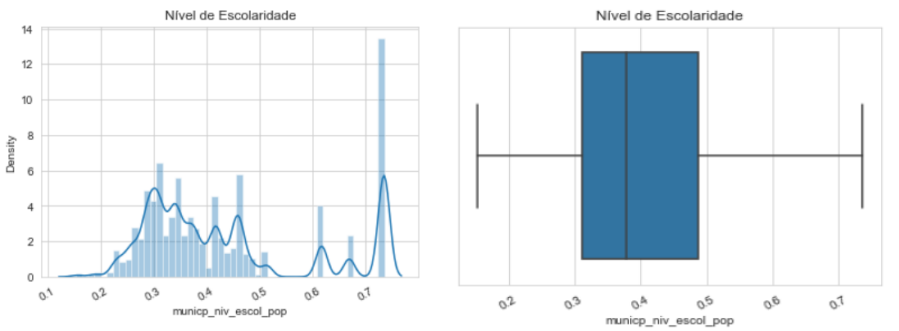
Observa-se que o atributo **municp_tx_analf** precisa ser normalizado para estar no mesmo domínio dos outros indicadores.



A distribuição do índice IDHM Educação é assimétrica à direita ou assimétrica positiva. Não se observa outliers.



O indicador da taxa de analfabetismo é assimétrico à esquerda, ou negativo, tendo correlação negativa com o índice IDHM Educação, como esperado.

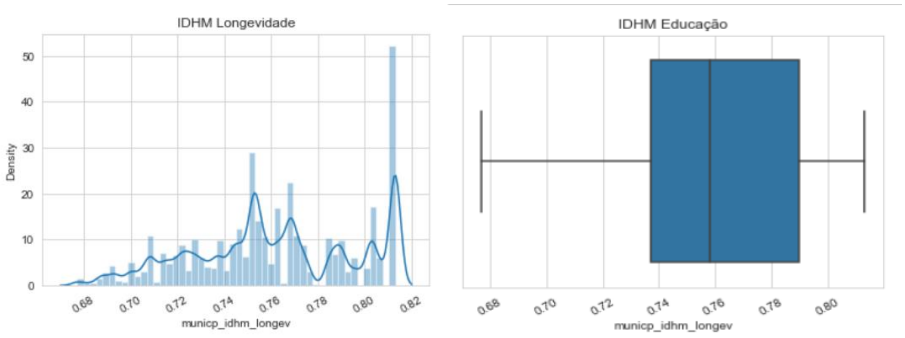


O indicador nível de escolaridade tem alta correlação com o índice IDHM Educação, podendo ser substituído por este.

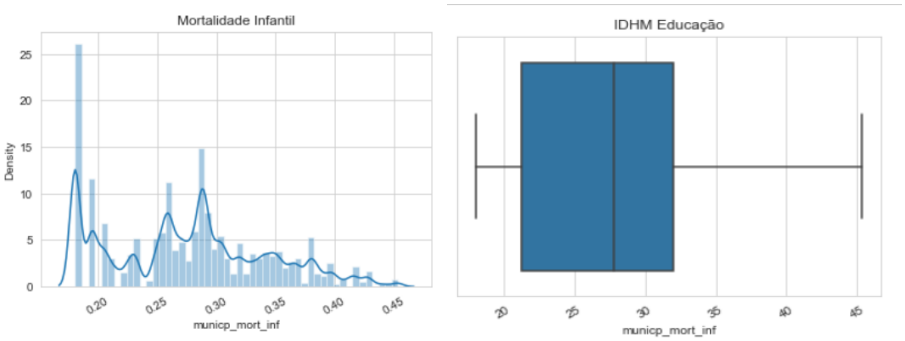
4.4.2.3 Indicadores de saúde

	municp_idhm_longev	municp_mort_inf	municp_domic_agua&esg
count	737003.000000	737003.000000	737003.000000
mean	0.761193	27.550102	23.179646
std	0.035149	6.741804	15.915259
min	0.677000	18.100000	1.940000
25%	0.737000	21.300000	7.430000
50%	0.758000	27.800000	18.250000
75%	0.790000	32.000000	34.300000
max	0.813000	45.300000	73.010000

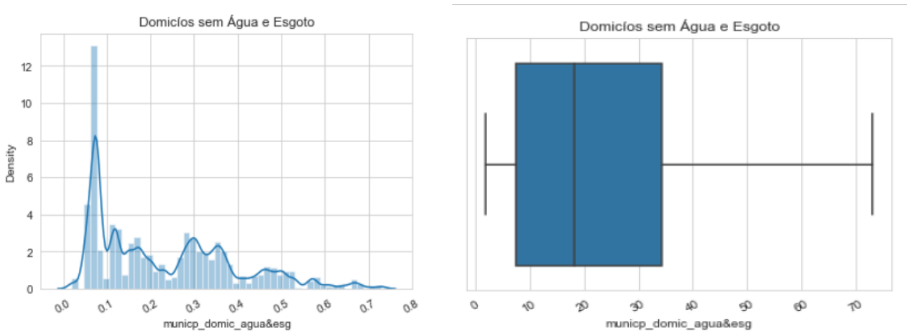
Os atributos **municp_mort_inf** e **municp_domic_agua&esg** precisam ser normalizados para estarem no mesmo domínio do indicador **municp_idhm_longev**.



A distribuição do índice IDHM Longevidade (saúde) é assimétrica à direita. Não se observa outliers.



O indicador da mortalidade infantil é assimétrico à esquerda, apresentando correlação negativa com o índice IDHM Longevidade, como esperado.

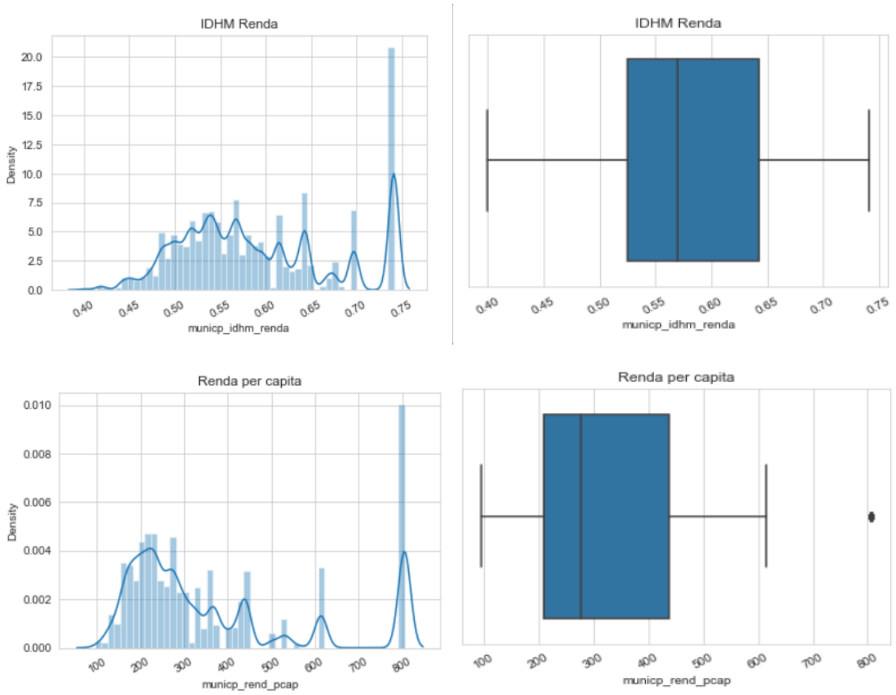


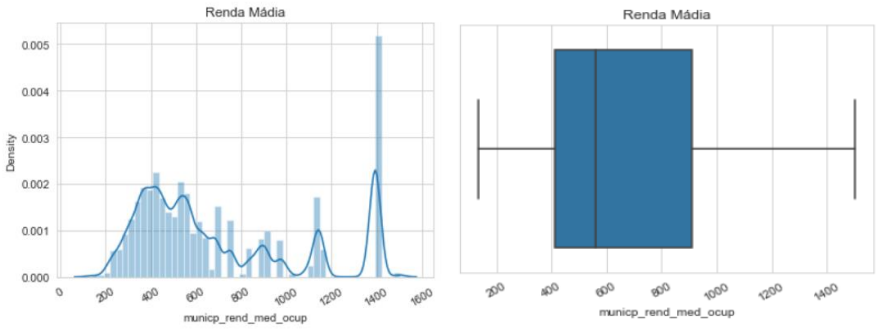
Semelhante ao indicador mortalidade infantil, este indicador é assimétrico à esquerda, tendo por diferença a sua mediana mais próxima do primeiro quartil.

4.4.2.4 Indicadores de renda

	municp_idhm_renda	municp_rend_pcap	municp_rend_med_ocup
count	737003.000000	737003.000000	737003.000000
mean	0.590240	364.898178	693.702958
std	0.084617	212.005949	367.498380
min	0.400000	96.250000	136.420000
25%	0.525000	210.300000	412.150000
50%	0.570000	277.190000	560.400000
75%	0.643000	438.560000	909.170000
max	0.741000	805.360000	1501.640000

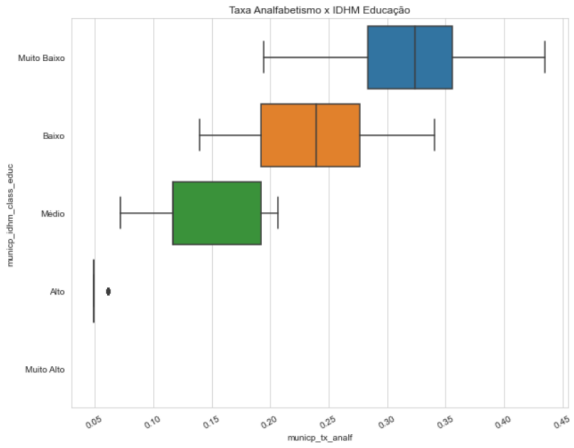
Os atributos **municp_rend_pcap** e **municp_rend_med_ocup** precisam ser normalizados para estarem no mesmo domínio do indicador **municp_idhm_renda**.



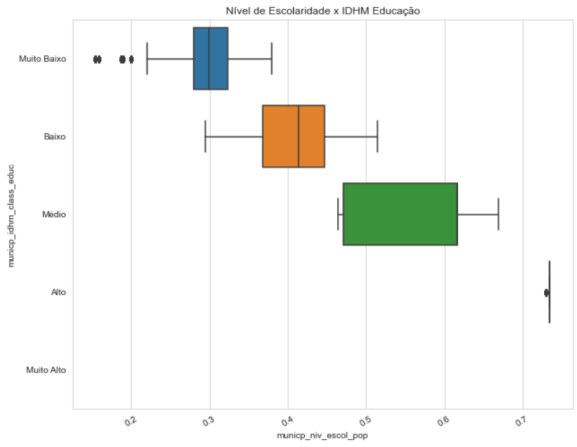


Os indicadores renda per capita e renda média têm alta correlação com o índice IDHM Renda, como esperado.

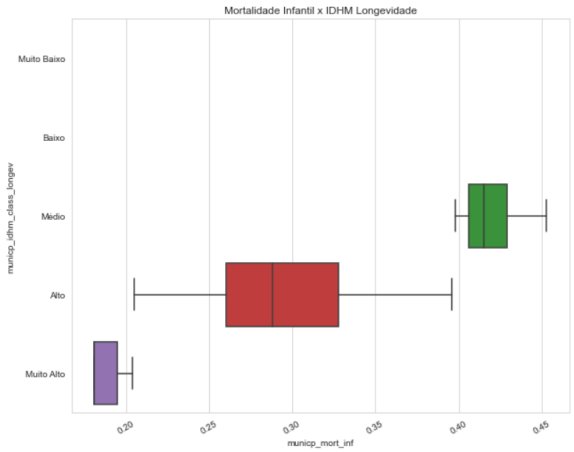
4.4 Análises Bivariadas



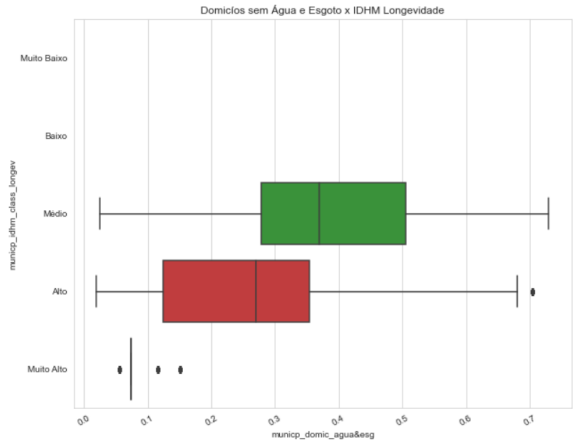
Observa-se que a taxa de analfabetismo apresenta maiores valores nos municípios de classes Muito Baixo e Baixo, chegando a valores de 40%. Nos municípios de classes Médio ou Alto apresentam valores entre 5 e 20%.



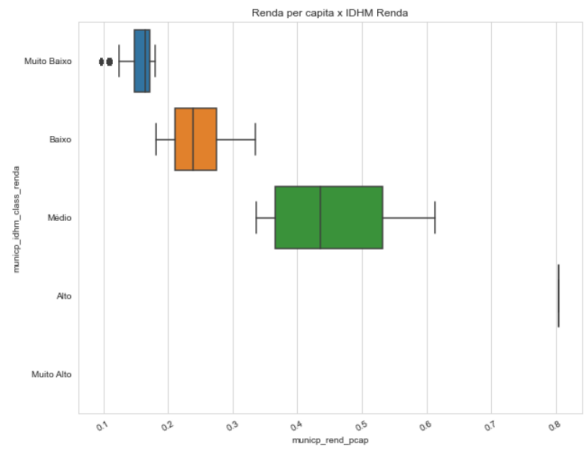
O nível de escolaridade apresenta melhores números nos municípios de classe Média, ficando entre 45 e 65%.



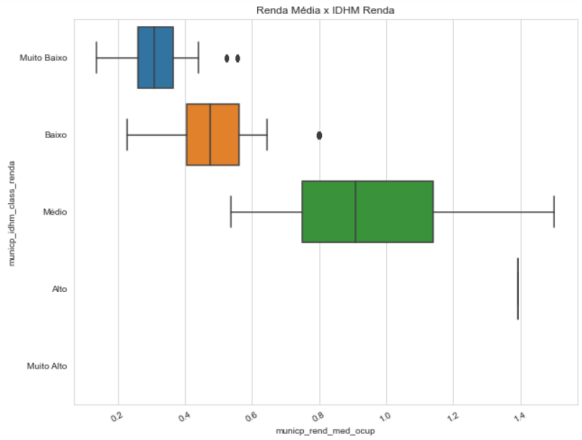
A mortalidade infantil tem grande dispersão em municípios da classe Alta, com valores entre 20% e 40%, resultado não esperado.



Já os municípios sem fornecimento de água permanente e rede de esgoto situam-se nas classes Média e Alto, com alguns outliers na classe Muito Alto.



Como falado na contextualização a renda *per capita* dos municípios maranhenses é a pior do Brasil, principalmente nos das classes Muito Baixo, Baixo e Médio.

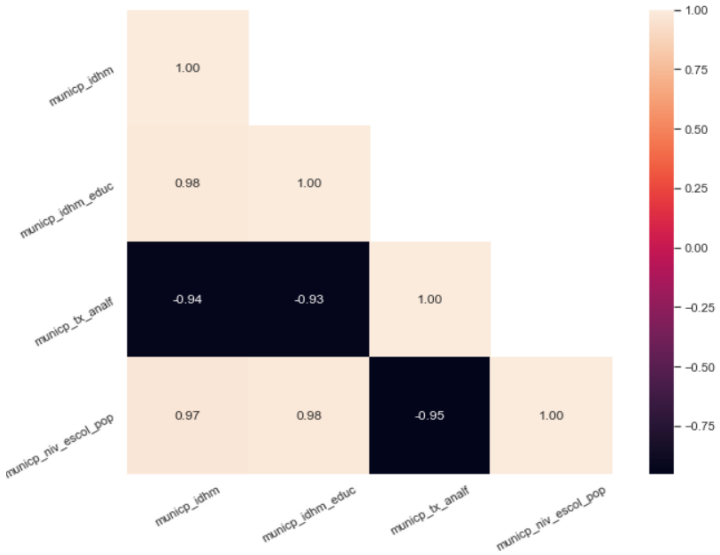


Situação semelhante observamos no caso da renda média.

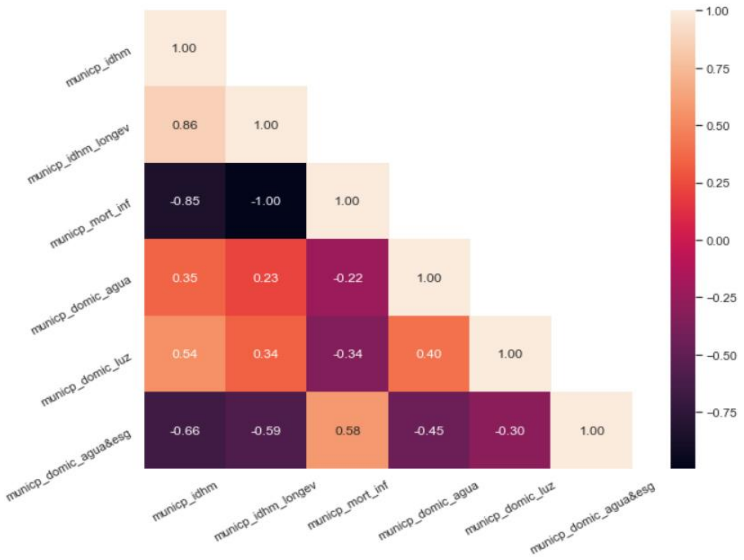
4.5 Análise Multivariada (matriz de correlação)

4.5.1 Indicadores de educação

A matriz de correlação de Pearson apresentou estes resultados para os indicadores agrupados por dimensão:



O indicador taxa de analfabetismo tem correlação negativa com os índices IDHM Educação (-0,93) e IDHM (-0,94).



O indicador mortalidade infantil tem correlação negativa com os índices IDHM Longevidade (-1) e IDHM (-0,85).

O indicador domicílio sem água e esgoto tem correlação negativa com os índices IDHM Longevidade (-0,59) e IDHM (-0,66).

A correlação dos indicadores renda per capita e renda média com o índice IDHM Renda é acima de 90%.

5. CRIAÇÃO DE MODELOS DE MACHINE LEARNING

"Aprendizado de Máquina é o campo de estudo que dá aos computadores a habilidade de aprender sem ser explicitamente programado—Arthur Samuel, 1959" (GÉRON, Aurélien).

Segundo Escovedo, aprendizado de máquina busca descobrir padrões ou fórmulas matemáticas que expliquem o relacionamento entre os dados e estuda formas de automatização de tarefas inteligentes que seriam difíceis ou até mesmo impossíveis de serem realizadas manualmente por seres humanos (ESCOVEDO, Tatiana).

O objetivo deste trabalho é a criação de modelos de *machine learning* não-supervisionado pela utilização de algoritmos de agrupamento de dados (*clusterização*) para análise dos 217 municípios maranhenses.

A clusterização tem por objetivo agrupar os dados de interesse, ou separar os registros de um conjunto de dados em subconjuntos ou grupos (*clusters*), de tal forma que elementos em um *cluster* compartilhem um conjunto de propriedades comuns que os diferencie dos elementos de outros *clusters* (*ESCOVEDO, Tatiana*).

Os procedimentos descritos nesta seção estão documentados no arquivo Jupyter Notebook **TCC_PUC_03 - JFarias_Vacina_MA - Modelo ML.ipynb**, disponível no repositório indicado no final deste trabalho.

5.1 Bibliotecas

```
import pandas as pd
from datetime import datetime, date
import numpy as np
import seaborn as sns

import matplotlib.pyplot as plt
get_ipython().run_line_magic('matplotlib', 'inline')

from matplotlib import pyplot as plt
import scipy.cluster.hierarchy as sch

from sklearn import metrics
from sklearn.cluster import AgglomerativeClustering, KMeans
from sklearn.cluster import KMeans
import skfuzzy

from sklearn.preprocessing import StandardScaler
from sklearn.preprocessing import LabelEncoder

import warnings
warnings.filterwarnings('ignore')
```

Nesta etapa foram acrescentados vários módulos do Scikit-learn (sklearn), biblioteca de código aberto (*open source*) de aprendizado de máquina para Python.

```
#pip install scikit-fuzzy --upgrade
```

Também foi necessário a instalação do módulo scikit-fuzzy (algoritmo Fuzzy C-Means), que será utilizado na criação do modelo de classificação.

5.2 Carregando a planilha de trabalho

```
# Carregando a planilha registros da vacinação Covid-19/MA
df_mml = pd.read_csv('RegVacinaMA_AED.csv', encoding='utf-8', sep=',', header=0, low_memory=False)
```

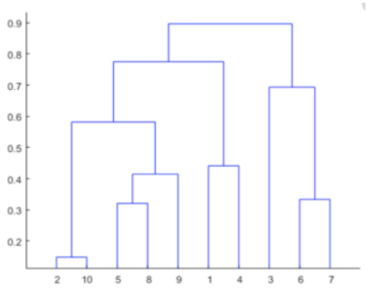
Carregado o dataset trabalhado na etapa anterior.

Os algoritmos de clusterização podem ser divididos em duas categorias:

- **Hierárquicos** e
- **Particionais**

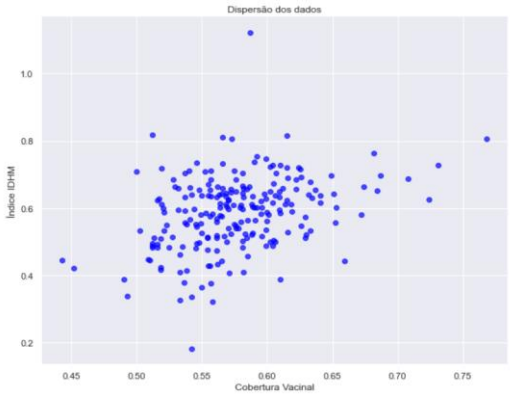
5.3 Algoritmos Hierárquicos

Os algoritmos **hierárquicos** criam uma decomposição hierárquica do conjunto de dados, representada por um *dendrograma*, uma árvore que iterativamente divide o conjunto de dados em subconjuntos menores até que cada subconjunto consista em somente um objeto. Cada nó da árvore representa um cluster do conjunto de dados e a união dos clusters em um determinado nível da árvore corresponde ao cluster no nível exatamente acima.



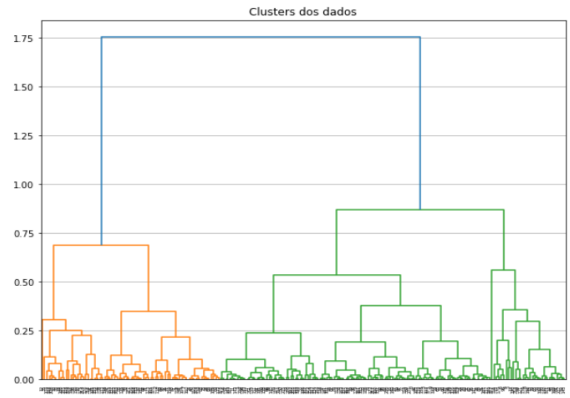
Exemplo de um dendrograma

5.3.1 Algoritmo Dendrograma



Visualização da dispersão dos dados. Observa-se que a maioria dos municípios estão na faixa de 40 a 70% da cobertura vacinal.

```
schX = sch.linkage(df_mmlX, 'ward')

plt.figure(figsize=(10, 8))
plt.grid(axis='y')
plt.title('Clusters dos dados')
plt.xticks(rotation=90)
denX = sch.dendrogram(schX, labels=list(df_mmlX.index))
```



O gráfico gerado pelo algoritmo apresenta algumas opções de clusters, de acordo com o nível de similaridade escolhido. O nível de similaridade é medido no eixo vertical. Por exemplo, se escolhido o nível 0.5 teremos 6 clusters.

Optou-se pelo nível 0.75, para gerar 3 clusters.

```
n_clusters = 3
cluster = AgglomerativeClustering(n_clusters=n_clusters, affinity='euclidean', linkage='ward')
den_cluster = cluster.fit_predict(df_mmlX)
den_cluster
```

```
array([2, 2, 0, 2, 1, 1, 1, 0, 1, 2, 0, 2, 0, 1, 1, 1, 2, 2, 0, 2, 2, 0,
       2, 0, 0, 2, 2, 2, 0, 2, 0, 2, 2, 0, 2, 0, 2, 0, 0, 2, 0, 0, 2, 2,
       0, 0, 2, 0, 2, 1, 2, 0, 1, 2, 1, 1, 2, 2, 2, 2, 2, 2, 2, 2, 1,
       2, 2, 0, 0, 1, 0, 2, 1, 2, 2, 2, 0, 0, 1, 2, 1, 2, 0, 2, 2, 0, 0,
       0, 0, 2, 0, 2, 0, 2, 0, 1, 2, 2, 2, 2, 2, 2, 2, 0, 0, 2, 2, 0, 0,
       0, 0, 1, 2, 2, 1, 2, 2, 0, 2, 0, 1, 2, 0, 0, 0, 0, 2, 2, 2, 1, 2,
       0, 0, 2, 0, 1, 2, 0, 2, 2, 2, 2, 1, 2, 0, 2, 2, 0, 2, 0, 2, 2, 0,
       2, 0, 0, 2, 2, 0, 0, 2, 2, 2, 0, 2, 0, 2, 0, 1, 1, 1, 2, 2, 0, 0,
       0, 0, 2, 0, 2, 2, 0, 2, 2, 0, 1, 2, 2, 2, 1, 2, 2, 1, 2, 0, 2, 2,
       0, 2, 2, 2, 0, 0, 2, 2, 0, 1, 2, 2, 2, 2, 2, 1, 0, 0, 2],
      dtype=int64)
```

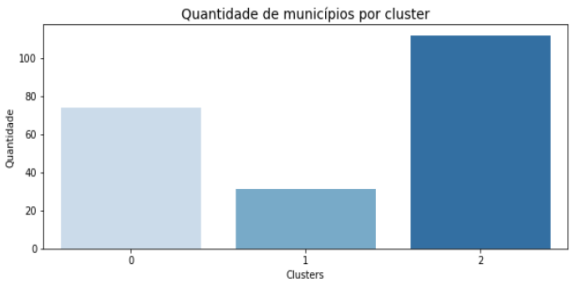
```
n_clusters
```

```
3
```

O algoritmo gerou os três clusters, distribuindo 74 municípios para o cluster 0, 31 para o cluster 1 e 112 para o cluster 2.

```
df_dend.cluster.value_counts()
```

```
2    112
0     74
1     31
Name: cluster, dtype: int64
```



5.4 Algoritmos Particionais

Os algoritmos particionais dividem a base de dados em k-grupos, onde o número k é dado pelo usuário. Inicialmente, o algoritmo escolhe k objetos como sendo os centros dos k clusters. Os objetos são divididos entre os k clusters de acordo com a distância entre o objeto e o centro do mesmo.

5.4.1 Fuzzy C-Means (FCM)

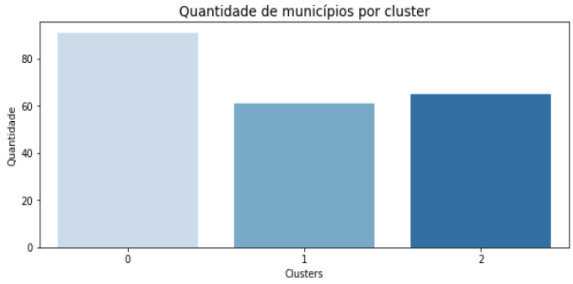
```
# Normalizando os valores
f_scaler = StandardScaler()
df_norm = f_scaler.fit_transform(df_X)
```

```
f_fuzzy = skfuzzy.cmeans(data = df_norm.T, c = 3, m = 2, error=0.005, maxiter=1000, init=None)
#f_fuzzy
```

O algoritmo Fuzzy c-means foi executado com k=3 clusters. O agrupamento ficou mais equilibrado, 91 municípios no cluster 0, 61 no cluster 1 e 65 no cluster 2.

```
df_fuzzy.cluster.value_counts()
```

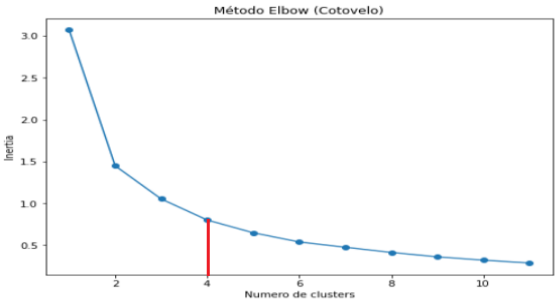
```
0    91
2    65
1    61
Name: cluster, dtype: int64
```



5.4.2 Algoritmo K-Means

O algoritmo k-means também requer que seja informado o número de clusters k. Alguns métodos auxiliam a determinação do melhor valor para k, como o método Elbow (cotovelo), que testa a variância dos dados em relação ao número de clusters, e o método silhouette (silhueta), que avalia os particionamentos encontrados e permite visualizar graficamente os agrupamentos.

Utilizou-se o método Elbow e foram definidos k=4 clusters .




```
df_kmcluster.cluster.value_counts()
```

```
0    91
2    66
3    35
1    25
Name: cluster, dtype: int64
```

O k-means distribuiu 91 municípios para o cluster 0, 25 para o cluster 1, 66 para o cluster 2 e 35 para o cluster 3.

5.5 Análise dos Clusters

5.5.1 Algoritmo Dendrograma

5.5.1.1 Análise Cluster 0

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	74.00	74.00	74.00	74.00
mean	25601.46	12087.73	0.47	0.56
std	17576.68	9254.30	0.07	0.05
min	6856.00	3270.00	0.18	0.44
25%	14604.00	6430.00	0.43	0.53
50%	19968.00	9845.50	0.49	0.56
75%	33095.75	15800.25	0.52	0.59
max	113783.00	65960.00	0.58	0.67

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Baixo	0.47	24712.48	11331.71	56
Muito Baixo	0.40	12428.25	4864.75	4
Médio	0.51	32921.14	17175.50	14

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
ALDEIAS ALTAS	0.513029	26979	13841
ALTO ALEGRE DO MARANHÃO	0.511687	28066	14361
AMARANTE DO MARANHÃO	0.429874	42017	18062
ARAGUANÃ	0.409952	15675	6426
ARAME	0.495963	32825	16280

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
0	74	47%	0,56	4	56	14		

5.5.1.2 Análise Cluster 1

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	31.00	31.00	31.00	31.00
mean	83487.71	62524.42	0.73	0.63
std	202296.57	161799.07	0.09	0.05
min	5936.00	4281.00	0.63	0.51
25%	11280.50	8087.50	0.69	0.60
50%	19090.00	14362.00	0.72	0.62
75%	64540.00	44833.50	0.75	0.66
max	1115932.00	898898.00	1.12	0.77

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Alto	0.71	420380.50	322663.75	4
Baixo	0.83	12953.14	10691.14	7
Médio	0.70	40796.25	28638.20	20

municip_nome	municip_cobert	municip_populacao	municip_qtdevac
ALCÂNTARA	0.807014	22126	17856
ALTO PARNAÍBA	0.678091	11233	7617
BALSAS	0.696455	96951	67522
BARÃO DE GRAJAÚ	0.754862	19026	14362
BELÁGUA	0.819141	7586	6214

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
1	31	73%	0,63		7	20	4	

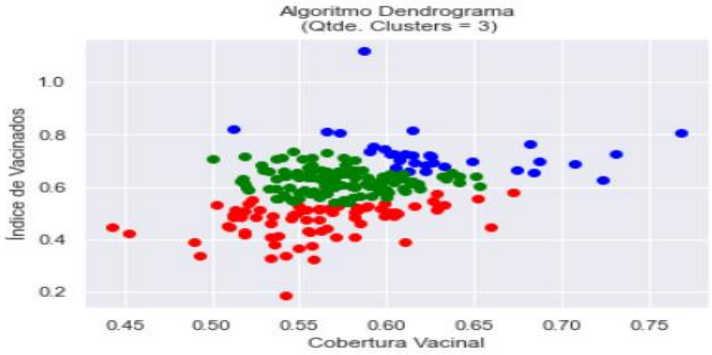
5.5.1.3 Análise Cluster 2

	municip_populacao	municip_qtdevac	municip_cobert	municip_idhm
count	112.00	112.00	112.00	112.00
mean	23844.96	14744.12	0.62	0.57
std	21811.93	13551.71	0.05	0.03
min	4334.00	2760.00	0.54	0.50
25%	10589.25	6369.50	0.59	0.55
50%	18095.00	11178.50	0.62	0.57
75%	27233.25	16340.75	0.65	0.59
max	123368.00	76738.00	0.73	0.65

municip_idhm_class	municip_cobert	municip_populacao	municip_qtdevac	nº de municípios
Baixo	0.62	20392.43	12538.55	92
Muito Baixo	0.71	12731.00	9011.00	1
Médio	0.62	41147.37	25725.42	19

municip_nome	municip_cobert	municip_populacao	municip_qtdevac
AFONSO CUNHA	0.663399	6631	4399
ALTAMIRA DO MARANHÃO	0.579879	8250	4784
ALTO ALEGRE DO PINDARÉ	0.584165	31967	18674
AMAPÁ DO MARANHÃO	0.599404	7047	4224
ANAJATUBA	0.645970	27170	17551

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
2	112	62%	0,57	1	92	19		



5.5.2 Algoritmo Fuzzy k-means

5.5.2.1 Análise Cluster 0

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	91.00	91.00	91.00	91.00
mean	19982.85	12387.84	0.63	0.56
std	17425.91	10467.29	0.06	0.02
min	4334.00	3069.00	0.53	0.50
25%	8823.50	5568.50	0.59	0.55
50%	16971.00	10310.00	0.63	0.57
75%	22991.00	13646.00	0.66	0.58
max	123368.00	76738.00	0.82	0.60

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Baixo	0.63	20063.42	12425.36	90
Muito Baixo	0.71	12731.00	9011.00	1

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
AFONSO CUNHA	0.663399	6631	4399
ALCÂNTARA	0.807014	22126	17856
ALTAMIRA DO MARANHÃO	0.579879	8250	4784
ALTO ALEGRE DO PINDARÉ	0.584165	31967	18674
AMAPÁ DO MARANHÃO	0.599404	7047	4224

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
0	91	63%	0,56	1	90			

5.5.2.2 Análise Cluster 1

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	65.00	65.00	65.00	65.00
mean	23590.37	10814.72	0.46	0.54
std	14283.32	6641.60	0.07	0.03
min	6856.00	3270.00	0.18	0.44
25%	14274.00	6168.00	0.43	0.52
50%	19521.00	8908.00	0.48	0.55
75%	29143.00	14187.00	0.51	0.56
max	73595.00	38436.00	0.56	0.61

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Baixo	0.47	24076.67	11090.05	58
Muito Baixo	0.40	12428.25	4864.75	4
Médio	0.46	29071.33	13425.00	3

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
ALDEIAS ALTAS	0.513029	26979	13841
ALTO ALEGRE DO MARANHÃO	0.511687	28066	14361
AMARANTE DO MARANHÃO	0.429874	42017	18062
ARAGUANÃ	0.409952	15675	6426
ARAME	0.495963	32825	16280

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
1	65	46%	0,54	4	58	3		

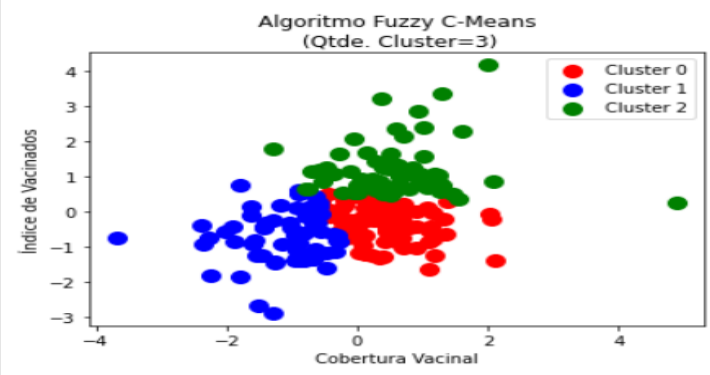
5.5.2.3 Análise Cluster 2

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	61.00	61.00	61.00	61.00
mean	62318.84	43505.57	0.65	0.63
std	146147.17	116807.94	0.10	0.04
min	4700.00	2760.00	0.44	0.59
25%	12923.00	8749.00	0.59	0.61
50%	23677.00	14597.00	0.64	0.62
75%	52852.00	32328.00	0.70	0.64
max	1115932.00	898898.00	1.12	0.77

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Alto	0.71	420380.50	322663.75	4
Baixo	0.72	21217.14	14493.71	7
Médio	0.63	39428.14	25234.58	50

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
ALTO PARNAÍBA	0.678091	11233	7617
ARARI	0.547178	30014	16423
AXIXÁ	0.638712	12234	7814
AÇAILÂNDIA	0.579700	113783	65960
BACABAL	0.642130	105094	67484

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
2	61	65%	0,63		7	50	4	



5.5.3 Algoritmo K-Means

5.5.3.1 Análise Cluster 0

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	66.00	66.00	66.00	66.00
mean	23473.14	12222.77	0.52	0.57
std	14265.51	7536.36	0.03	0.04
min	6261.00	3320.00	0.44	0.50
25%	13942.75	6709.50	0.50	0.55
50%	19614.50	10517.00	0.52	0.57
75%	29216.50	15952.25	0.55	0.59
max	73105.00	38436.00	0.58	0.66

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Baixo	0.52	22726.82	11895.45	55
Médio	0.51	27204.73	13859.36	11

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
ALDEIAS ALTAS	0.513029	26979	13841
ALTO ALEGRE DO MARANHÃO	0.511687	28066	14361
ARAME	0.495963	32825	16280
ARARI	0.547178	30014	16423
BACABEIRA	0.510891	17446	8913

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
0	66	52%	0,57		55	11		

5.5.3.2 Análise Cluster 1

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	91.00	91.00	91.00	91.00
mean	28622.32	17906.37	0.63	0.58
std	27271.43	17011.06	0.03	0.04
min	4682.00	2760.00	0.57	0.52
25%	11231.00	7047.00	0.60	0.56
50%	19616.00	12420.00	0.63	0.58
75%	32757.50	20622.50	0.65	0.61
max	125265.00	78359.00	0.69	0.72

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Alto	0.63	125265.00	78359.00	1
Baixo	0.63	21627.48	13531.00	64
Médio	0.63	42123.35	26351.42	26

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
AFONSO CUNHA	0.663399	6631	4399
ALTAMIRA DO MARANHÃO	0.579879	8250	4784
ALTO ALEGRE DO PINDARÉ	0.584165	31967	18674
AMAPÁ DO MARANHÃO	0.599404	7047	4224
ANAJATUBA	0.645970	27170	17551

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
1	91	63%	0,58		64	26	1	

5.5.3.3 Análise Cluster 2

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	25.00	25.00	25.00	25.00
mean	24720.76	9468.28	0.39	0.53
std	15969.46	6142.11	0.06	0.04
min	7757.00	3270.00	0.18	0.44
25%	15675.00	6102.00	0.38	0.52
50%	17123.00	7079.00	0.41	0.54
75%	29121.00	10589.00	0.43	0.56
max	73595.00	31587.00	0.46	0.61

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Baixo	0.39	26959.25	10297.15	20
Muito Baixo	0.40	12428.25	4864.75	4
Médio	0.39	29121.00	11305.00	1

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
AMARANTE DO MARANHÃO	0.429874	42017	18062
ARAGUANÃ	0.409952	15675	6426
BOM JARDIM	0.413663	42010	17378
BOM JESUS DAS SELVAS	0.323066	35095	11338
BOM LUGAR	0.432803	16578	7175

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
2	25	39%	0,53	4	20	1		

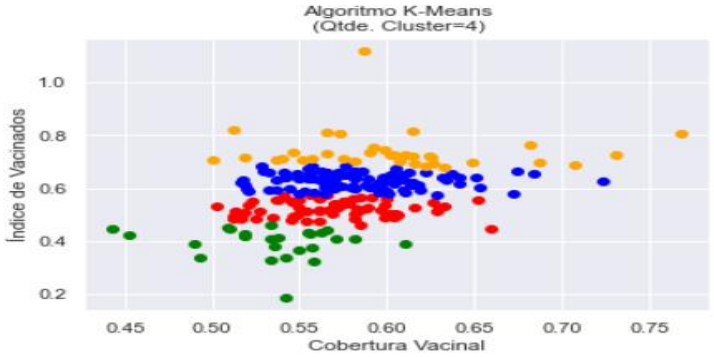
5.5.3.4 Análise Cluster 3

	municp_populacao	municp_qtdevac	municp_cobert	municp_idhm
count	35.00	35.00	35.00	35.00
mean	68039.57	51748.57	0.74	0.60
std	192264.54	153652.20	0.08	0.06
min	4334.00	3069.00	0.68	0.50
25%	8952.50	6448.00	0.70	0.57
50%	12662.00	9011.00	0.72	0.60
75%	24094.50	19553.50	0.74	0.62
max	1115932.00	898898.00	1.12	0.77

municp_idhm_class	municp_cobert	municp_populacao	municp_qtdevac	nº de municípios
Alto	0.74	518752.33	404098.67	3
Baixo	0.76	11084.75	8549.00	16
Muito Baixo	0.71	12731.00	9011.00	1
Médio	0.72	42336.07	30207.27	15

municp_nome	municp_cobert	municp_populacao	municp_qtdevac
ALCÂNTARA	0.807014	22126	17856
ALTO PARNAÍBA	0.678091	11233	7617
BALSAS	0.696455	96951	67522
BARÃO DE GRAJAÚ	0.754862	19026	14362
BELÁGUA	0.819141	7586	6214

Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
				Muito Baixo	Baixo	Médio	Alto	Muito Alto
3	35	74%	0,6	1	16	15	3	



6. INTERPRETAÇÃO DOS RESULTADOS

Como demonstrado, os fatores socioeconômicos influenciam a população e sua adesão aos programas de vacinação.

Índices de baixa cobertura vacinal são encontrados justamente nos municípios com classificação IDHM Muito Baixa ou Baixa, caso da maioria dos municípios maranhenses. Enquanto os melhores índices estão nos municípios classificados com o IDHM Médio ou Alto.

Os algoritmos avaliados executaram os agrupamentos propostos, dividindo a base de dados por similaridades e apresentaram, com diferença apenas na formação dos agrupamentos, o mesmo resultado (tabela abaixo).

O algoritmo k-Means apresentou maior interação com a base de dados e seria a escolha para aprimoramentos deste trabalho.

Algoritmo	Cluster	Qtde. Municípios	Cobertura Vacinal média	IDHM médio	Classificação IDHM				
					Muito Baixo	Baixo	Médio	Alto	Muito Alto
Dendrograma	0	74	47%	0,56	4	56	14		
	1	31	73%	0,63		7	20	4	
	2	112	62%	0,57	1	92	19		
	Total	217			5	155	53	4	
	%	100,00%			2,30%	71,43%	24,42%	1,84%	0,00%
Fuzzy c-Means	0	91	63%	0,56	1	90			
	1	65	46%	0,54	4	58	3		
	2	61	65%	0,63		7	50	4	
	Total	217			5	155	53	4	
	%	100,00%			2,30%	71,43%	24,42%	1,84%	0,00%
k-Means	0	66	52%	0,57		55	11		
	1	91	63%	0,58		64	26	1	
	2	25	39%	0,53	4	20	1		
	3	35	74%	0,60	1	16	15	3	
	Total	217			5	155	53	4	
	%	100,00%			2,30%	71,43%	24,42%	1,84%	0,00%

7. APRESENTAÇÃO DOS RESULTADOS

Abaixo, workflow do trabalho pelo modelo de *Canvas* proposto por Vasandani:

<table>
<tr><td colspan="3">Título: CLUSTERIZAÇÃO DO ÍNDICE DE VACINAÇÃO NO ESTADO DO MARANHÃO COM BASE EM INDICADORES SOCIOECONÔMICOS</td></tr>
<tr>
<td>Definição do Problema:

Criação de modelos e utilização de algoritmos de classificação e agrupamento de dados para análise dos 217 municípios maranhenses.</td>
<td>Resultados/previsões:

Classificação dos municípios maranhenses pelos indicares socioeconômicos</td>
<td>Aquisição de dados:

Dados obtidos em sites públicos como:
Site openDataSUS;
Site AtlasBR
Site do IBGE</td>
</tr>
<tr>
<td>Modelagem:

Matriz de correlação de Pearson para seleção de atributos

Normalização dos dados com StandardScaler

Algoritmos de aprendizado de máquina não-supervisionado:
• Dendrograma
• Fuzzy c-Means
• k-Means</td>
<td>Avaliação do Modelo:

Os algoritmos executaram a clusterização na forma proposta</td>
<td>Preparação dos dados:

Tratamento de dados com filtragem, exclusão de atributos e junção de datasets.</td>
</tr>
</table>

8. LINKS

Link para o vídeo: https://youtu.be/-GAJCLZPRTo

Link para o repositório: https://github.com/fariasjm/TCC-Ci-ncias-de-Dados-2020

REFERÊNCIAS

ALVARENGA, Darlan; SILVEIRA, Daniel. Economia.
Disponível em: <https://g1.globo.com/economia/noticia/2021/03/03/pib-do-brasil-despenca-41percent-em-2020.ghtml>. Acesso em: 07 jan. 2022

COUTO, Marcia Thereza; BARBIERI, Carolina Luisa Alves; MATOS, Camila Carvalho de Souza Amorim. Considerações sobre o impacto da covid-19 na relação indivíduo-sociedade: da hesitação vacinal ao clamor por uma vacina. Disponível em: <https://www.scielo.br/j/sausoc/a/rQFs3PMLgZprt3hkJMyS8mN/>. Acesso em: 07 jan. 2022

ESCOVEDO, Tatiana. Machine Learning: conceitos e modelos – parte I: aprendizado supervisionado. Disponível em: <https://tatianaesc.medium.com/machine-learning-conceitos-e-modelos-f0373bf4f445>. Acesso em 07 jan. 2022.

_____. Machine Learning: conceitos e modelos – parte II: aprendizado não supervisionado. Disponível em: <https://tatianaesc.medium.com/machine-learning-conceitos-e-modelos-parte-ii-aprendizado-n%C3%A3o-supervisionado-fb6d83e4a520>. Acesso em: 07 jan. 2022.

GÉRON, Aurélien. **Mãos à obra aprendizado de máquina com scikit-learn & tensorflow**: conceitos, ferramentas e técnicas para a construção de sistemas inteligentes. Trad. Rafael Contatori. Rio de Janeiro: AltaBooks, 2019.

IBM Cloud Lear Hub. Análise exploratória de dados. Disponível em: <https://www.ibm.com/br-pt/cloud/learn/exploratory-data-analysis>. Acesso em: 05 jan. 2022

IDH (Índice de Desenvolvimento Humano). Disponível em: <https://www.infoescola.com/geografia/idh-indice-de-desenvolvimento-humano/>. Acesso em: 05 jan. 2022.

_____. Disponível em: <https://blog.brkambiental.com.br/idh-brasil/>. Acesso em: 05 jan. 2022

_____. Disponível em: <https://pt.wikipedia.org/wiki/%C3%8Dndice_de_Desenvolvimento_Humano>. Acesso em: 05 jan. 2022

OLIVEIRA, Bruno Luciano Carneiro Alves de. Prevalência e fatores associados à hesitação vacinal contra a COVID-19 no Maranhão, Brasil. Disponível em: <https://www.scielo.br/j/rsp/a/tQzJW4JDcNVLtjhh7crg3tz/?lang=pt>. Acesso em: 07 jan. 2022.

PEREIRA, Handrey dos Santos; RODRIGUES, Fábio da Silva. Efeitos da pandemia de COVID-19 no IDH do Brasil: uma pesquisa bibliográfica com análise documental. V EIGEDIN (19 a 22out. 2021 - Online). Disponível em: <https://desafioonline.ufms.br/index.php/EIGEDIN/article/download/14350/9563/>. Acesso em: 06 jan. 2022.

PNUD no Mundo. Disponível em: <https://www.br.undp.org/content/brazil/pt/home/idh0.html>. Acesso em: 05 jan. 2022.

PNUD. Relatório de desenvolvimento humano 2020: síntese. Disponível em: http://hdr.undp.org/sites/default/files/hdr_2020_overview_portuguese.pdf p.29>. Acesso em: 06 jan. 2022

SENADO Notícias. Disponível em: <https://www12.senado.leg.br/noticias/materias/2021/03/22/demora-na-vacinacao-traz-maior-impacto-economico-aponta-relatorio-da-ifi>. Acesso em: 07 jan. 2022.