**Python para Engenharia de Dados**

**Projeto Final**

**COVID-19**

**Anne Karoline Fortunato do Carmo**

**2023**

**Introdução**

Recentemente, o mundo passou por uma pandemia, conhecida como pandemia da COVID-19. Essa pandemia impactou o mundo inteiro desde o seu surgimento em dezembro de 2019 e o Brasil registrou um alto número de casos e de óbitos ao longo dos últimos três anos.

Este documento é referente a uma análise, onde foram explorados os dados relacionados à pandemia no Brasil. Os dados utilizados foram obtidos a partir do portal Brasil.IO, que disponibiliza informações detalhadas sobre os casos confirmados, casos de óbitos e outros indicadores relacionados à doença.

Os dados disponibilizados permitiram obter insights extremamente relevantes sobre a propagação e impacto da COVID-19 no país.

**Objetivo**

O objetivo desta análise é investigar e obter insights relevantes sobre como foi a evolução da COVID-19 no Brasil.

Foram realizadas explorações analíticas que foram capazes de fornecer como foi a evolução dos casos no país e quais foram os padrões e as tendências.

**Metodologia**

Para realizar esta análise, utilizou-se a linguagem de programação Python e algumas de suas bibliotecas de análise de dados como o Pandas e o Matplotlib.

Os dados disponibilizados pelo Brasil I.O foram carregados em um Dataframe do Pandas para facilitar a análise dos dados e a sua manipulação.

Os procedimentos da análise foram:

1. Carregamentos dos dados.
2. Exploração dos dados
3. Limpeza e pré-processamento dos dados.
4. Análise Exploratória dos dados.
5. Apresentação dos resultados utilizando gráficos e tabelas.

**Carregamento dos dados**

Para o carregamentos dos dados foi utilizado o link fornecido (https://data.brasil.io/dataset/covid19/caso\_full.csv.gz) e carregado em um Dataframe do Pandas para a análise.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Os 5 primeiros resultados estão exibidos no dataframe abaixo:

Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Exploração dos dados**

Foi realizado alguns comandos para conhecer a base de dados. O df.columns retornou todas as colunas do Dataframe.

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

A descrição das colunas do Dataframe se encontra abaixo:

* **city:** nome do município (pode estar em branco quando o registro é referente ao estado, pode ser preenchido com Importados/Indefinidos também).
* **city\_ibge\_code:** código IBGE do local.
* **date:** data de coleta dos dados no formato YYYY-MM-DD.
* **epidemiological\_week:** número da semana epidemiológica no formato YYYYWW.
* **estimated\_population:** população estimada para esse município/estado em 2020, segundo o IBGE. (acesse o script que faz o download e conversão dos dados de população).
* **estimated\_population\_2019:** população estimada para esse município/estado em 2019, segundo o IBGE. ATENÇÃO: essa coluna possui valores desatualizados, prefira usar a coluna estimated\_population.
* **is\_last**: campo pré-computado que diz se esse registro é o mais novo para esse local, pode ser True ou False (caso filtre por esse campo, use is\_last=True ou is\_last=False, não use o valor em minúsculas).
* **is\_repeated:** campo pré-computado que diz se as informações nesse registro foram publicadas pela Secretaria Estadual de Saúde no dia date ou se o dado é repetido do último dia em que o dado está disponível (igual ou anterior a date). Isso ocorre pois nem todas as secretarias publicam boletins todos os dias. Veja também o campo last\_available\_date.
* **last\_available\_confirmed:** número de casos confirmados do último dia disponível igual ou anterior à data date.
* **last\_available\_confirmed\_per\_100k\_inhabitants:** número de casos confirmados por 100.000 habitantes (baseado em estimated\_population) do último dia disponível igual ou anterior à data date.
* **last\_available\_date:** data da qual o dado se refere.
* **last\_available\_death\_rate:** taxa de mortalidade (mortes / confirmados) do último dia disponível igual ou anterior à data date.
* **last\_available\_deaths:** número de mortes do último dia disponível igual ou anterior à data date.
* **order\_for\_place:** número que identifica a ordem do registro para este local. O registro referente ao primeiro boletim em que esse local aparecer será contabilizado como 1 e os demais boletins incrementarão esse valor.
* **place\_type:** tipo de local que esse registro descreve, pode ser city ou state.
* **state:** sigla da unidade federativa, exemplo: SP.
* **new\_confirmed:** número de novos casos confirmados desde o último dia (note que caso is\_repeated seja True, esse valor sempre será 0 e que esse valor pode ser negativo caso a SES remaneje os casos desse município para outro).
* **new\_deaths:** número de novos óbitos desde o último dia (note que caso is\_repeated seja True, esse valor sempre será 0 e que esse valor pode ser negativo caso a SES remaneje os casos desse município para outro).

O df.describe() retornar um resumo estatísticos dos dados como contagem, média, desvio padrão, valores mínimos e máximos. Com essas informações também é possível identificar se há valores inconsistentes ou ausentes no Dataframe.

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Limpeza e pré-processamento dos dados**

Após conhecer um pouco sobre a base de dados que será trabalhada, foi utilizado o comando df.isnull() que retorna True para valores nulos ou False para campos preenchidos. É possível perceber que há campos nulos no Dataframe estudado.

Tela de computador

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Foi realizado uma etapa de limpeza para remoção de dados ausentes ou inconsistentes afim de garantir a qualidade dos dados utilizados.

Para verificar os dados ausentes, foi utilizado a função df.isnull().sum() para ver a soma de valores ausentes no Dataframe para casa coluna.

A próxima imagem mostra que as colunas city, city\_ibge\_cod, estimated\_population, estimated\_population\_2019 e last\_available\_confirmed\_per\_100k\_inhabitants possuem valores nulo. Após essa verificação, havia duas opções: deletar os registros que possuem valores ausentes(usando o comando df.dropna()) ou preencher os valores ausentes com algum valor adequado (usando o comando df.fillna()).

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

De acordo com as definições das colunas, o campo **city** é referente ao nome do município e pode estar em branco quando o registro é referente ao estado. Desta forma, pode ser preenchido com Importados/Indefinidos. Foi inserido indefinidos nos campos nulos.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

A coluna city\_ibge\_code, quando nula, recebeu valor de 0.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Como a coluna **estimated\_population\_2019** está desatualizada, ela foi removida do Dataframe.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Os registros que possuíam 'estimated\_population' nulo foram removidos do Dataframe.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

De acordo com a descrição das colunas, last\_available\_confirmed\_per\_100k\_inhabitants é número de casos confirmados por 100.000 habitantes (baseado em estimated\_population) do último dia disponível igual ou anterior à data date. Foi realiza esse tratamento nos dados para os campos que possuíam valores nulos.

Uma imagem contendo Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Após realizar a limpeza dos registros nulos, a próxima etapa foi converter os tipos de dado, quando necessário. Para isso, foi necessário verificar o schema atual do Dataframe com o comando df.dtypes().

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

De posse dessas informações, as únicas colunas que optou para converter o tipo de dado foi a coluna ‘date’ e ‘last\_available\_date’ que se encontra como tipo object (string) e é do tipo date e a coluna ‘estimated\_population’ que se encontrava no tipo float e é do tipo int.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

A nova configuração do schema ficou:

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

last\_available\_date é a data da qual o dado se refere. Portando, iremos criar três novas colunas no dataframe, referente a ano, mes e dia.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

Para realizar algumas análises, também foi realizado a divisão do df em dois dataframes. Um para place\_type igual a city e outro para place\_type igual a state.

Forma

Descrição gerada automaticamente com confiança média

Disponível no notebook trabalho-final.ipynb

**Tela de computador com fundo preto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Tela de computador com fundo preto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Análise Exploratória dos dados**

Abaixo estão as análises realizadas:

**Distribuição dos casos por estado**

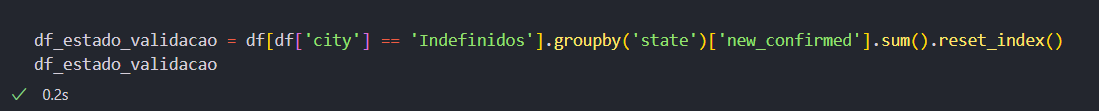
Abaixo estão as análises realizadas para obter a quantidade de casos confirmados por estado.

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Também foi realizado um código de validação, para verificar se a consulta estava sendo realizada corretamente.

****

Disponível no notebook trabalho-final.ipynb

Após obter o Dataframe, foi criado um gráfico para visualizar os resultados.

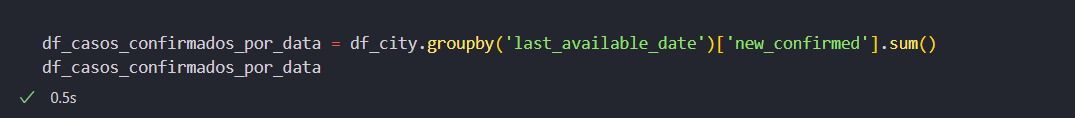
**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Evolução de casos de COVID-19 por data**

Para obter os casos confirmados por data, foi utilizado a coluna de nossos casos confirmados e a coluna de data correspondente ao registro.

****

Disponível no notebook trabalho-final.ipynb

Ordenando os dados do maior para o menor:

**Tela preta com letras brancas

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Código utilizado para a criação do gráfico:

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Total de mortes por COVID**

Realizando a soma dos registros da coluna new\_deaths para obter o total de pessoas que vieram a óbito pela doença:

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Total de mortes por ano**

Obtendo o total de mortes, separando os resultados pelos anos:

**Interface gráfica do usuário, Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Código utilizado para a criação do gráfico:

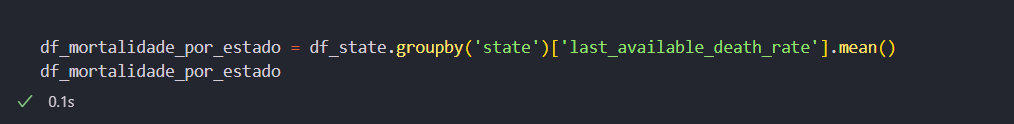
**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Taxa de mortalidade por estado**

Obtendo a taxa de mortalidade por estado do país:

****

Disponível no notebook trabalho-final.ipynb

Ordenando os registros em ordem crescente:

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Código utilizado para a criação do gráfico:

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Análise regional do Brasil**

Para fazer uma análise por região do brasil, é necessário criar uma nova coluna no dataframe.

Desta forma, foi possível agrupar os dados por região geográfica baseado na sigla dos estados. é possível realizar uma comparação de casos confirmados óbito.

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Agrupando os registros por região e obtendo a soma dos casos confirmados, soma dos óbitos e a media da taxa de mortalidade:

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Ordenando os registros obtidos:

**Tela de computador com texto preto sobre fundo branco

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

Código utilizado para a criação do gráfico:

**Texto

Descrição gerada automaticamente**

Disponível no notebook trabalho-final.ipynb

**Resultados**

Os resultados da análise estão apresentados em forma de gráficos e comentários descritivos. Os insights obtidos com a propagação da COVID-19 no brasil indica que:

**Distribuição dos casos por estado**

Com base no gráfico gerado, percebe-se que alguns estados brasileiros apresentam um número significativamente maior de casos em comparação com outros. Pode-se concluir que a propagação da COVID-19 não foi uniformemente distribuída pelo país.

Gráfico, Gráfico de barras, Histograma

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Total de mortes por ano**

O primeiro registro foi realizado em 23/02/2020 e o último registro do dataset foi realizado em 27/03/2022.

Texto

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

O gráfico abaixo mostra a quantidades de mortes registradas nos anos de 2020, 2021 e 2022, dentro do intervalo entre a primeira e a última data registrada.

Gráfico, Gráfico de barras

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Taxa de mortalidade por estado**

A taxa de mortalidade por estado está exibida no gráfico abaixo. De acordo com a população do estado, o Rio de Janeiro foi o estado com a maior Taxa de mortalidade e em segundo lugar ficou o estado de Pernambuco.

Gráfico, Histograma

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Evolução de casos de COVID-19 por data**

O dia 03/09/2021 é considerado o pico de casos confirmados no Brasil. Foi registrado 356.442 casos. O segundo pico é do dia 21/04/2021 e o terceiro no dia 02/10/2021.

Gráfico, Gráfico de linhas, Histograma

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Análise regional do Brasil**

A região com o maior número de casos confirmados foi a Sudeste, seguido de Nordeste, Sul, Centro-Oeste e Norte. Percebe-se também que a COVI-19 afetou as regiões de maneiras distintas, podem principalmente ter sido influenciados pelos fatores de densidade populacional da região, infraestrutura de saúde e das medidas de controle adotadas pelos governos.

Gráfico, Gráfico de barras

Descrição gerada automaticamente

Disponível no notebook trabalho-final.ipynb

**Conclusão**

A conclusão dessa análise forneceu uma visão geral dos principais aspectos da pandemia da COVID-19 no Brasil.

Análises como a de taxa de crescimentos dos casos por data nos permite identificas os períodos em que houve aumento e diminuição da propagação da COVI-19. Essa análise acaba sendo primordial para monitorar a evolução da pandemia no país ao longo do tempo e, consequentemente, auxiliar no controle da doença. Ao analisar os casos por região, percebe-se uma disparidade. Tal disparidade pode ter sido ocasionada pelas políticas públicas adotadas, pela infraestrutura da região e pela densidade populacional.

Em resumo, pode-se concluir que a COVID-19 teve impactos bem heterogêneos no Brasil, variando entre cidades, estados e regiões.