Análise de dados e Políticas Públicas

Introdução

Utilizar análises de dados é muito importante para criar ou melhorar políticas públicas mais eficientes. Aqui vamos falar sobre como números e dados podem ser amigos das políticas públicas. Isso é importante porque queremos que representantes do estado democrático façam coisas que funcionem, certo?

Políticas públicas são basicamente os planos do governo para fazer a sociedade ficar melhor. Eles podem ser sobre saúde, educação, dinheiro, ou até mesmo coisas divertidas como cultura. Às vezes, todos nós ajudamos a pensar nelas!

A ideia é que essas políticas públicas sigam as regras que estão escritas na Constituição de 1988, que é como o manual das leis aqui no Brasil. Mas como saber o que fazer e onde investir nosso dinheiro? Aí é onde entram os dados.

Os dados são como pistas que nos ajudam a entender o que está acontecendo na sociedade. Eles nos mostram coisas como quanto dinheiro as pessoas ganham, se têm acesso a serviços como saúde e educação, e até mesmo se todo mundo está tendo as mesmas oportunidades.

Por exemplo, temos o Instituto Brasileiro de Geografia e Estatística - IBGE. Eles estão coletando informações sobre tudo, desde quantas pessoas vivem em uma cidade até quanto tempo leva para as pessoas irem ao trabalho.

A transparência é fundamental aqui. Precisamos ter certeza de que todos podem ver e entender esses dados, porque isso ajuda a manter as coisas justas. Existem até leis, como a Lei de Acesso à Informação, que garantem que você pode pedir essas informações ao governo. E também temos a Lei Geral de Proteção de Dados (LGPD), que protege suas informações pessoais.

Então, resumindo, dados são como dicas valiosas para criar políticas públicas melhores. E é importante que todos possam acessá-los e que nossos dados pessoais sejam protegidos. Afinal, estamos todos juntos nessa jornada para uma sociedade mais justa!

Tutorial

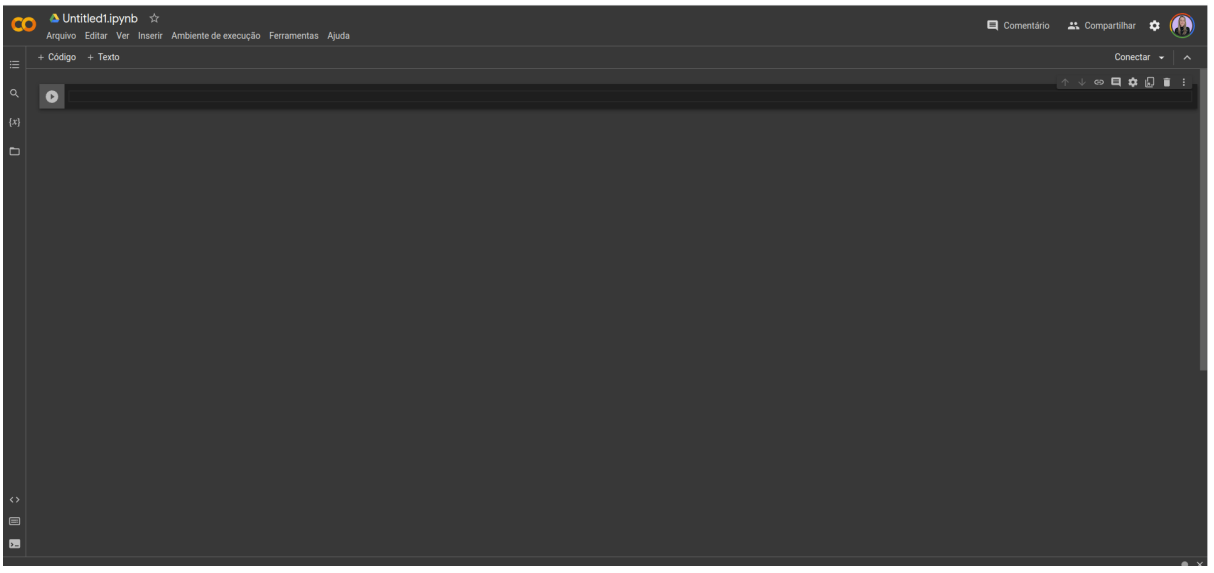
Vamos realizar uma Análise de Dados simples utilizando Python, Pandas, Matplotlib, e o Google Colab! :)

1. **Vamos acessar o Google Colab:**

    - Nesse link aqui: https://colab.google/

2. **Vamos clicar em "New Notebook":**

    Para isso precisamos ter uma conta no Google. Vai abrir uma nova página no navegador com seu 'Notebook' aberto. O legal é que nessa plataforma podemos simular um ambiente virtual para trabalhar com códigos, e podemos armazenar esses arquivos em diversos locais, na sua máquina, no github, no drive, etc…
    [image] ()



3. **Vamos alterar o nome do arquivo**:

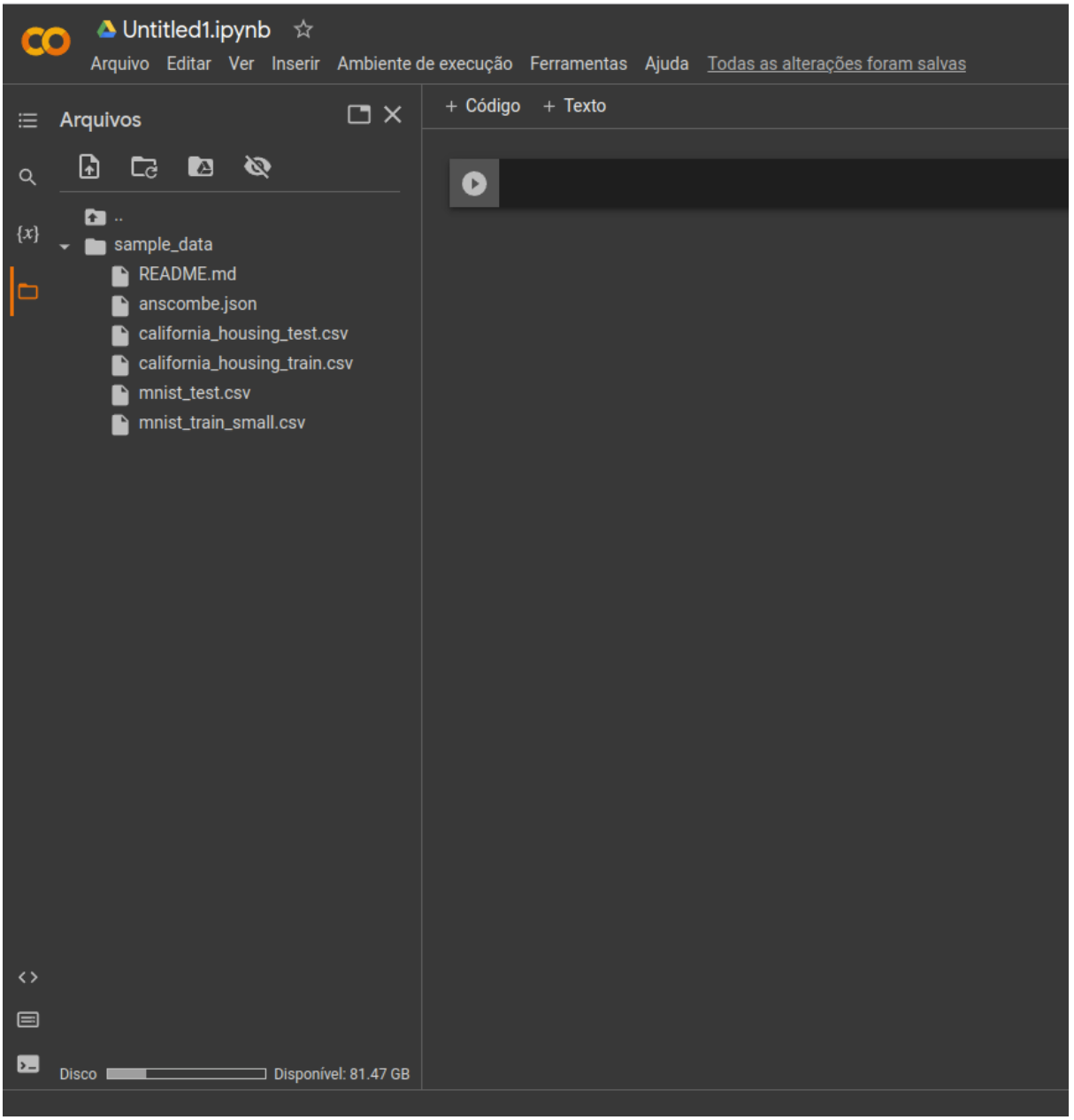
    - Na parte superior do arquivo, basta clicar no nome do arquivo com extensão .ipynb e trocar para **'aula1'**.
    - Se quiser, podemos armazenar/salvar este arquivo no Drive, Github, etc…

4. **Vamos configurar nosso projeto**:

    - No lado esquerdo do arquivo temos um opção para configurar o 'arquivo', vamos clicar nela, e vai aparecer o ícone de uma pasta, é o nosso projeto!

- No ‘Google Colab’ já temos um ambiente preparado para realizar análise de dados, porém pode utilizar outras tecnologias, como jupyter, anaconda, tudo depende do que vamos analisar. :)

[image] ()



5. **Vamos copiar, ou arrastar o arquivo .csv para a pasta: ‘sample-data’ ou integrar com o Google Drive**

    - *Montar o Google Drive no Colab, tem que criar uma célula(bloco de código) no notebook com o seguinte conteúdo:*

      ```sh
      from google.colab import drive
      drive.mount('/content/drive')
      ```

```

- *Ao fazer isso ira aparecer uma mensagem como esta:*

```sh
Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=[um-valor-bem-longo]
Enter your authorization code:
```

- *Acesse o URL acima, escolha uma conta Google, copie o token gerado e cole no Colab, aperte enter.*
- *Feito isso a célula atualiza e aparece a seguinte mensagem:*

```sh
……
Mounted at /content/drive
```

- *Enviar o arquivo para Google Drive*
- *Com o drive montado, vá no seu Google Drive e faça upload do arquivo*.

Exemplo: Datasets/imdb-reviews-pt-br.csv.

- *Abrir Dataset no Colab: Com o arquivo no Google Drive, crie uma célula com os seguintes valores:*

```sh
import pandas as pd

df = pd.read_csv('/content/drive/My Drive/Datasets/imdb-reviews-pt-br.csv')
df.head()
```

- *Você já deve ser capaz de visualizar as primeiras linhas do seu dataset!!! :)*

6. **Agora vamos acessar o site que disponibiliza dados abertos para fazermos a análise**

- vamos acessar o site do INEP no link abaixo: https://www.gov.br/inep/pt-br/acesso-a-informacao/dados-abertos/microdados/enem

- ao clicar no link será realizado o download do arquivo .zip(compactado) com todos os dados
```

- descompactar o arquivo .zip e verificar quais são as pastas e tipos de arquivo que contém
- vamos escolher a do microdados do Enem

7. **Agora vamos programar! Se vc não sabe nada de python, não tem problema, é só copiar e colar o bloco de código abaixo:**

   Mas é fundamental estudar esta linguagem de programação se você pretende avançar na carreira na área de Ciência de Dados. S2

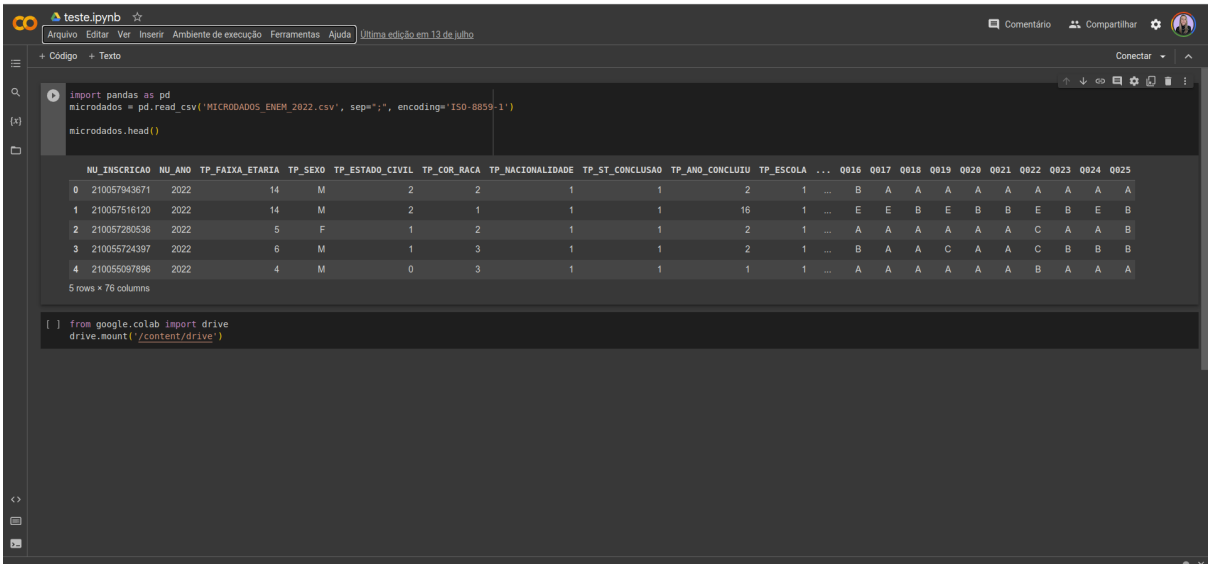
- criar uma variavel para armazenar os dados que vão ser importados com o panda:

  https://pandas.pydata.org/docs/reference/api/pandas.read_csv.html

  ```sh
  microdados = pd.read_csv('caminho-do-arquivo'), sep=";", enconding='ISO-8859-1'
  ```

- Depois de copiar o código e colar no Google Colab, clique na tecla run dentro do bloco de código e veja a mágica acontecer!
- Ao rodar o comando teremos o data Frame que é a estrutura de dados do pandas que é encapsulado e lido por ele, ou seja, uma tabela com linhas e colunas :)

[image] ()



8. **A partir daqui iniciamos a análise:**

   - Nesse momento é hora de realizar os questionamentos
   - O que queremos analisar?
   - Quais as perguntas que os dados, e índices podem apontar respostas, ou hipóteses?

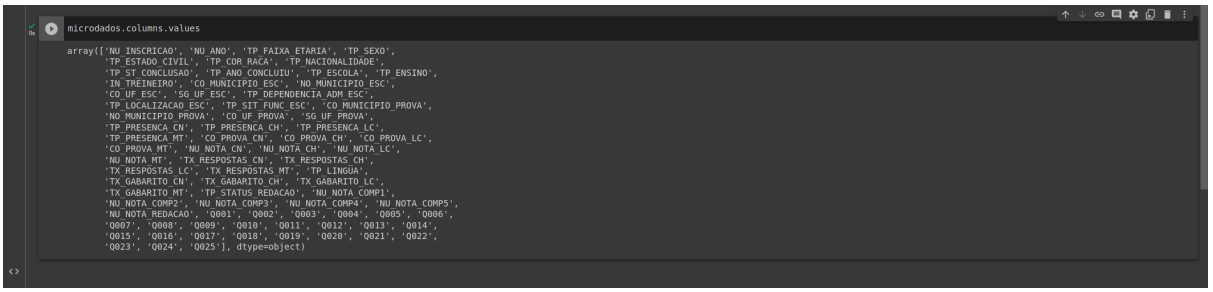
- Como esse resultado pode impactar nas decisões tanto no público como no privado?

9. **Para continuar, vamos fazer uma breve análise exploratória a partir da seguinte questão:**

    - Como organizar as colunas para ter insights?
    - Vamos digitar o código abaixo para verificar os dados

    ```sh
    microdados.columns.values
    ```

[image] ()



```
microdados.columns.values

array(['NU_INSCRICAO', 'NU_ANO', 'TP_FAIXA_ETARIA', 'TP_SEXO',
       'TP_ESTADO_CIVIL', 'TP_COR_RACA', 'TP_NACIONALIDADE',
       'TP_ST_CONCLUSAO', 'TP_ANO_CONCLUIU', 'TP_ESCOLA', 'TP_ENSINO',
       'IN_TREINEIRO', 'CO_MUNICIPIO_ESC', 'NO_MUNICIPIO_ESC',
       'CO_UF_ESC', 'SG_UF_ESC', 'TP_DEPENDENCIA_ADM_ESC',
       'TP_LOCALIZACAO_ESC', 'TP_SIT_FUNC_ESC', 'CO_MUNICIPIO_PROVA',
       'NO_MUNICIPIO_PROVA', 'CO_UF_PROVA', 'SG_UF_PROVA',
       'TP_PRESENCA_CN', 'TP_PRESENCA_CH', 'TP_PRESENCA_LC',
       'TP_PRESENCA_MT', 'CO_PROVA_CN', 'CO_PROVA_CH', 'CO_PROVA_LC',
       'CO_PROVA_MT', 'NU_NOTA_CN', 'NU_NOTA_CH', 'NU_NOTA_LC',
       'NU_NOTA_MT', 'TX_RESPOSTAS_CN', 'TX_RESPOSTAS_CH',
       'TX_RESPOSTAS_LC', 'TX_RESPOSTAS_MT', 'TP_LINGUA',
       'TX_GABARITO_CN', 'TX_GABARITO_CH', 'TX_GABARITO_LC',
       'TX_GABARITO_MT', 'TP_STATUS_REDACAO', 'NU_NOTA_COMP1',
       'NU_NOTA_COMP2', 'NU_NOTA_COMP3', 'NU_NOTA_COMP4', 'NU_NOTA_COMP5',
       'NU_NOTA_REDACAO', 'Q001', 'Q002', 'Q003', 'Q004', 'Q005', 'Q006',
       'Q007', 'Q008', 'Q009', 'Q010', 'Q011', 'Q012', 'Q013', 'Q014',
       'Q015', 'Q016', 'Q017', 'Q018', 'Q019', 'Q020', 'Q021', 'Q022',
       'Q023', 'Q024', 'Q025'], dtype=object)
```

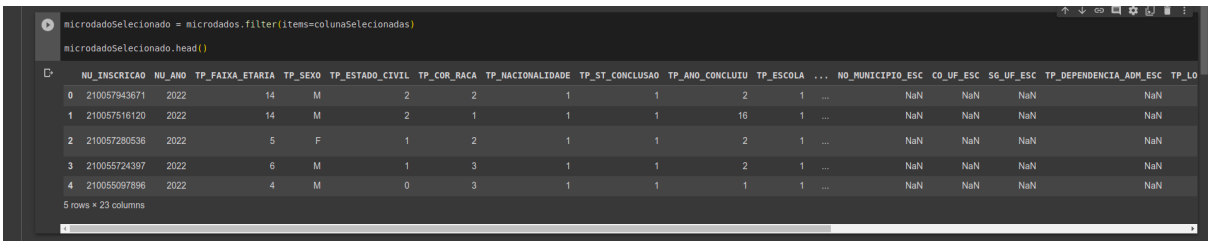
- Esse comando irá retornar um array com um vetor e com o nome de todas as colunas

10. **Como hoje vamos analisar somente um conjunto de dados, vamos selecionar apenas algumas colunas para isso, ou seja, vamos criar um data frame para esta análise. Vamos digitar o seguinte código:**

    - O método que vamos utilizar é o filter (para filtrar as colunas que queremos analisar)

    ```sh
    microdadoSelecionado = microdados.filter(items=colunaSelecionadas)

    microdadoSelecionado.head()
    ```

[image] ()



```
microdadoSelecionado = microdados.filter(items=colunaSelecionadas)

microdadoSelecionado.head()
```

	NU_INSCRICAO	NU_ANO	TP_FAIXA_ETARIA	TP_SEXO	TP_ESTADO_CIVIL	TP_COR_RACA	TP_NACIONALIDADE	TP_ST_CONCLUSAO	TP_ANO_CONCLUIU	TP_ESCOLA	...	NO_MUNICIPIO_ESC	CO_UF_ESC	SG_UF_ESC	TP_DEPENDENCIA_ADM_ESC	TP_LO
0	210057943671	2022	14	M	2	2	1	1	2	1	...	NaN	NaN	NaN	NaN	
1	210057516120	2022	14	M	2	1	1	1	16	1	...	NaN	NaN	NaN	NaN	
2	210057280536	2022	5	F	1	2	1	1	2	1	...	NaN	NaN	NaN	NaN	
3	210055724397	2022	6	M	1	3	1	1	2	1	...	NaN	NaN	NaN	NaN	
4	210055097896	2022	4	M	0	3	1	1	1	1	...	NaN	NaN	NaN	NaN	

5 rows × 23 columns

11. **Vamos analisar a distribuição de alunos por município**

https://pandas.pydata.org/docs/getting_started/intro_tutorials/06_calculate_statistics.html

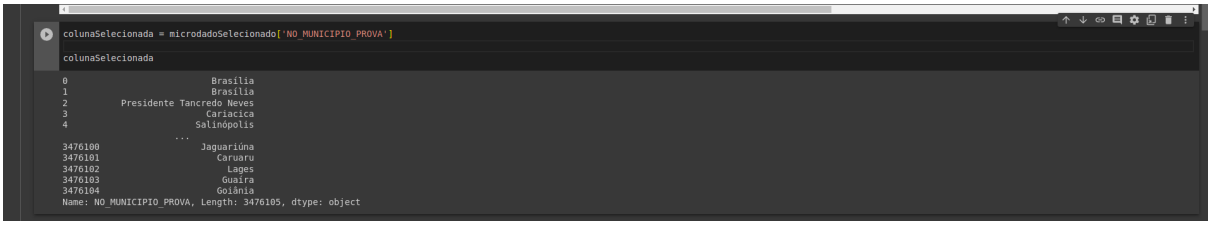
https://pandas.pydata.org/docs/reference/index.html

Ou seja, quantas linhas tem para cada município:

- pegar o nome da variável da coluna e o método da biblioteca do Pandas, e também podemos ordenar os dados e procurar o município que queremos, por exemplo:

  ```sh
  colunaSelecionada = microdadoSelecionado['NO_MUNICIPIO_PROVA']

  colunaSelecionada
  ```

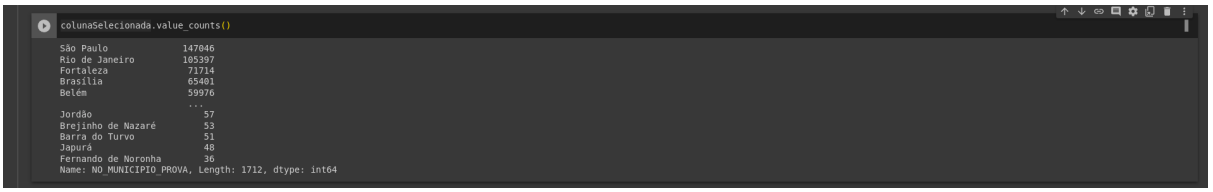
[image] ()



e

```sh
colunaSelecionada.value_counts()
```

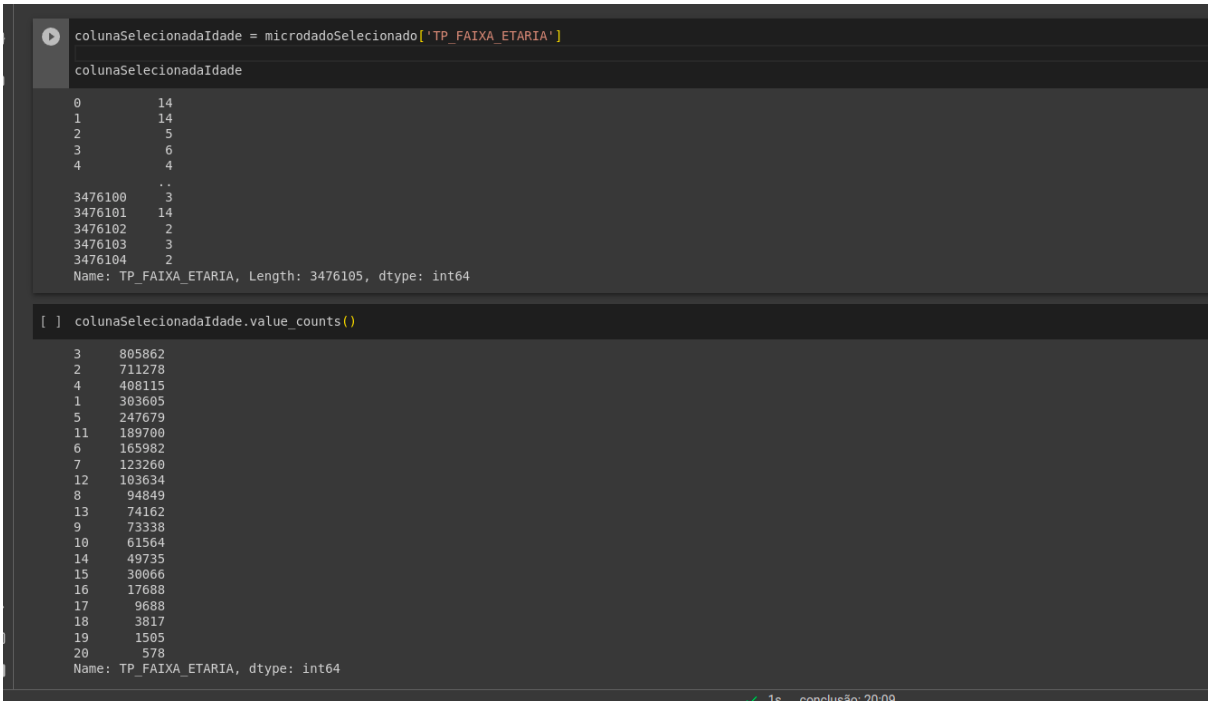
[image] ()



- Agora vamos fazer para a faixa etária:

  ```sh
  colunaSelecionadaIdade = microdadoSelecionado['TP_FAIXA_ETARIA']

  colunaSelecionadaIdade.value_counts()
  ```

[image] ()



12. E agora para visualizar os dados vamos utilizar a biblioteca matplotlib

https://matplotlib.org/stable/api/

- Importamos ela lá no início do projeto, lembra? Aqui está como importamos:

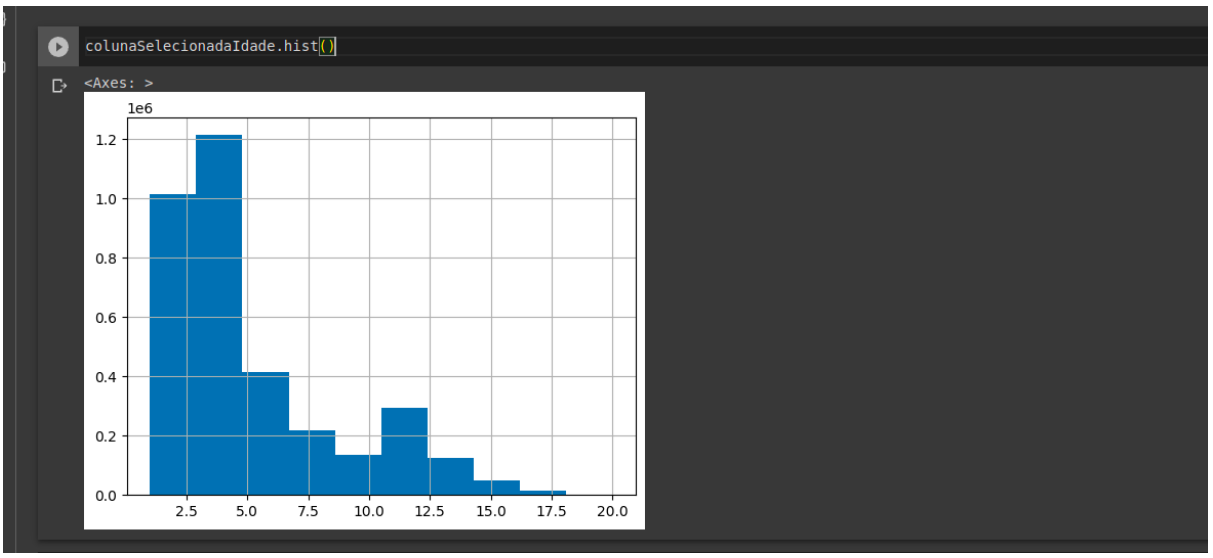
```sh
import matplotlib
```

[image] ()

- E podemos rodar esse comando:

```sh
colunaSelecionadaIdade.hist()
```

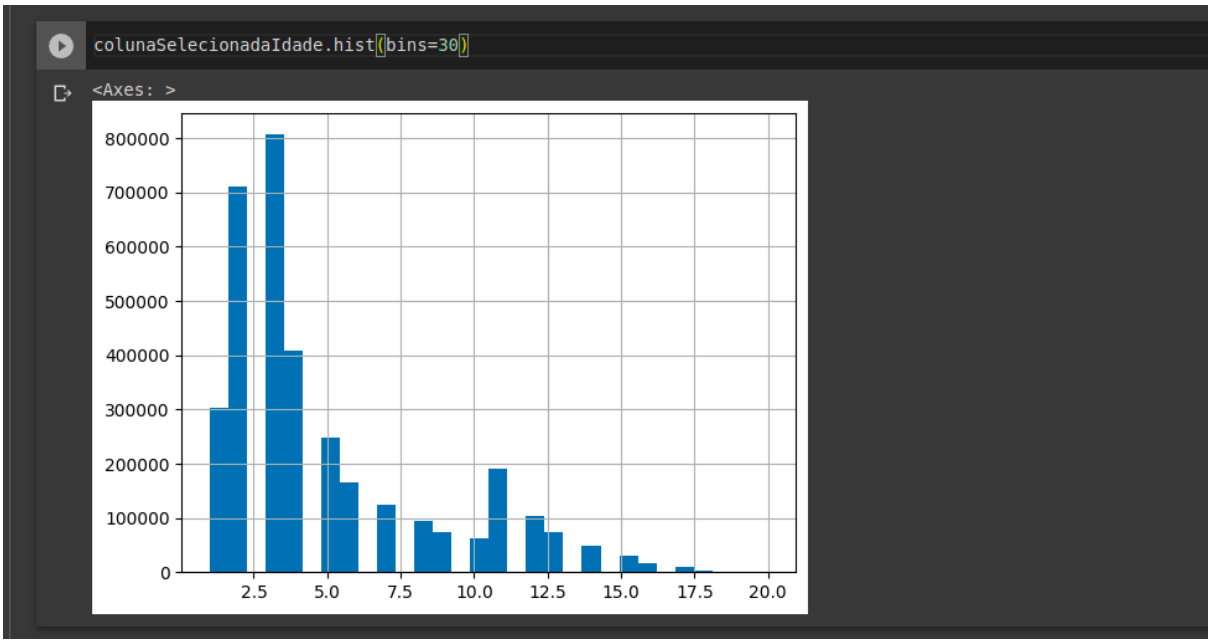
[image] ()



- E se quisermos aumentar a distribuição de dados para melhor utilização, podemos usar o parâmetro ‘bins’:

  ```sh
  colunaSelecionadaIdade.hist(bins=30)
  ```

[image] ()

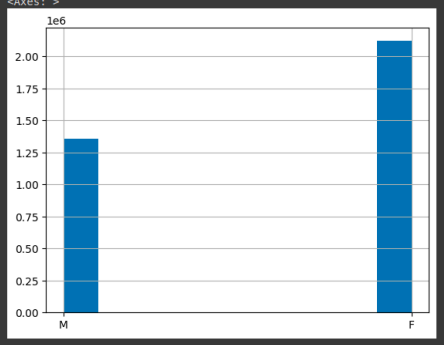


- Agora vamos fazer a análise de gênero:

  ```sh
  colunaGenero = microdadoSelecionado['TP_SEXO']
  colunaGenero.hist()
  ```

[image] ()

```
[ ] colunaGenero = microdadoSelecionado['TP_SEXO']

    generoIndex = colunaGenero.value_counts()

[ ] percentGenero = [100*x/generoIndex.sum() for x in generoIndex]

    percentGenero

    [61.00273150552127, 38.99726849447873]

    colunaGenero.hist()

    <Axes: >
```



spoiler!
Opa, mas o Enem não cadastra por gênero, temos apenas a opção de masculino e feminino, será que esses dados não podem auxiliar a uma política pública para que tenha mais opções de gênero?

Conclusão:

Agora, é hora de se aprofundar na Análise de Dados. Vamos explorar mais, fazer perguntas diferentes e experimentar tecnologias novas. Lembrem-se, o que vimos foi um tutorial básico para mostrar como usar dados públicos em análises simples ou complexas.

O uso de dados é essencial para políticas públicas eficazes. Ajuda as organizações governamentais e não-governamentais a tomar decisões informadas, alinhadas com as necessidades reais da sociedade, e também a avaliar o desempenho das políticas após a implementação. Vamos continuar explorando a análise de dados para atender melhor às necessidades da sociedade!

Fonte:

Cursos gratuitos Do Governo Federal para a area de Ciência de Dados: https://www.gov.br/governodigital/pt-br/capacita/ciencia-de-dados

Pesquisa sobre Ciência de Dados e Educação: https://www.institutounibanco.org.br/iniciativas/centro-de-pesquisa-transdisciplinar-em-educacao-cpte/ciencia-de-dados-na-educacao/

2022 - Ciência de dados em políticas públicas: uma experiência de formação Escola Nacional de Administração Publica (Brasil); De Toni, Jackson (Organizador); Dorneles, Rachel (Organizadora) - https://repositorio.enap.gov.br/handle/1/7472