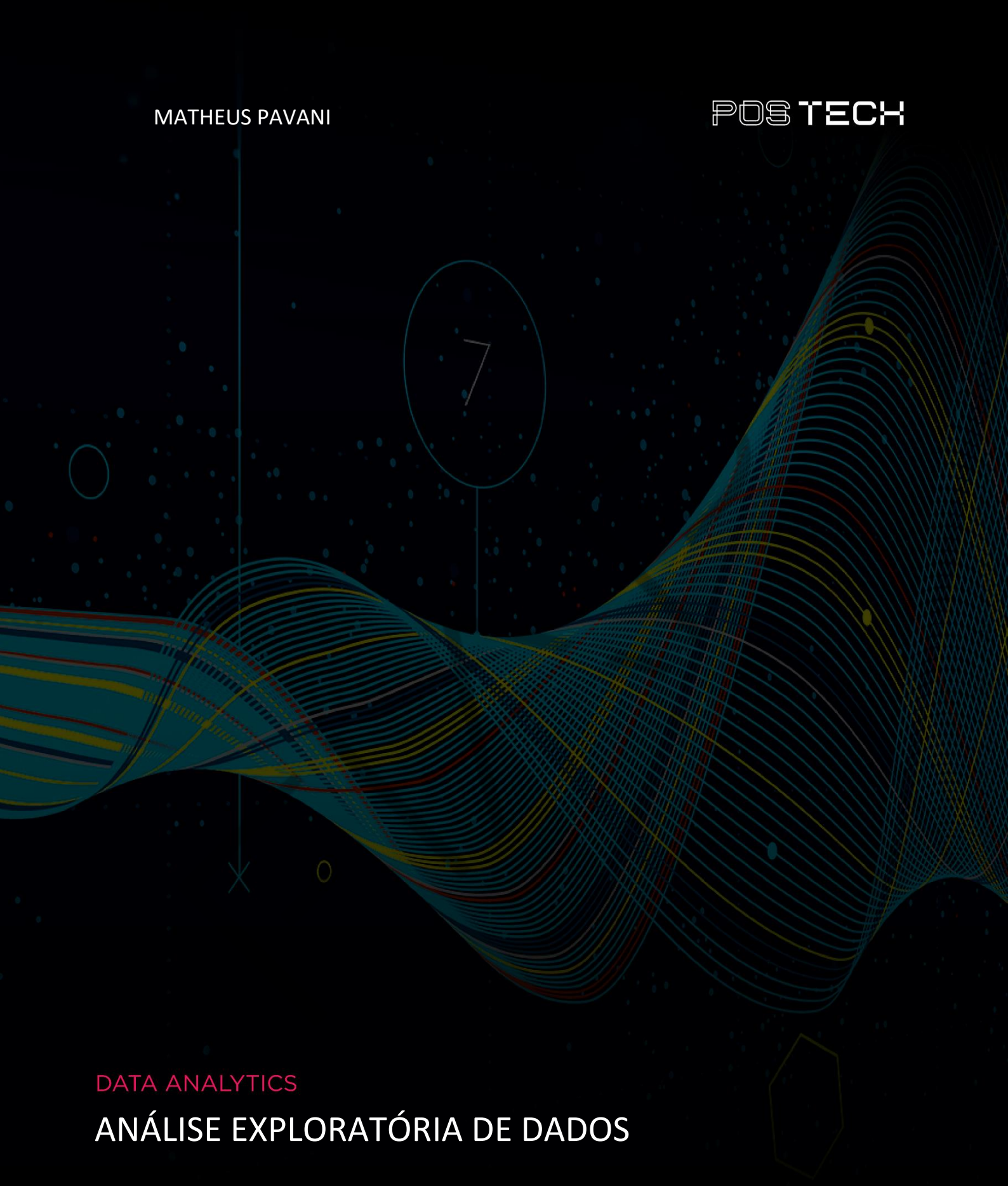
MATHEUS PAVANI



DATA ANALYTICS

ANÁLISE EXPLORATÓRIA DE DADOS

AULA 01

SUMÁRIO

O QUE VEM POR AÍ?

Nesta aula, vamos aprender como iniciar a nossa coleta de dados reais focando em um ambiente propício para testes, como o site do Google Colab, que tem estruturas parecidas com o Jupyter. Mas, nada te impede de usar o próprio Jupyter Notebook: site do Jupyter Notebook, Kaggle: site do Kaggle ou outras IDE's de sua preferência. Além disso, utilizaremos a biblioteca Pandas: site do Pandas, do Python, para começar a trazer esses dados reais para o universo dos algoritmos.

Ao final da disciplina, temos um desafio que faz parte da sua jornada de aprendizado, e você poderá acessá-lo na área de atividades da plataforma.

Este desafio vai te ajudar a praticar os conhecimentos adquiridos durante as aulas e te preparar para o projeto da fase!

HANDS ON

Agora, chegou o momento de ver, na prática, como começar a importar nossos dados e trabalhar com eles via programação. O ambiente utilizado é o Google Colab e as bases de dados que foram disponibilizadas no início deste documento. A ideia é não se limitar apenas ao código explícito no hands on, então recomendamos que procurem a documentação das bibliotecas, explorem novas funcionalidades e muito mais!

A seguir, temos uma ideia de código que foi desenvolvido no vídeo da aula, mas é importante que, ao olhar para ele, você consiga se desenvolver e trabalhar de maneira livre. Isso significa que é interessante mexer no código e praticar, explorar as funcionalidades e parâmetros. Por isso, as documentações das bibliotecas são tão importantes!

```
import pandas as pd #1ª célula

dados = pd.read_csv('/content/A150850189_28_143_208.csv', encoding='ISO-8859-1', skiprows=3, sep=';', skipfooter=12, thousands='.', decimal=',') #2ª célula

dados.head()#3ª célula

dados.tail()#4ª célula

dados.info()#5ª célula

pd.options.display.float_format = '{:.2f}'.format #6ª célula

dados.mean() #7ª célula
```

Notebook Aula 1 – Produção Hospitalar.ipynb
Fonte: Elaborado pelo autor (2023)

SAIBA MAIS

Quando pensamos em uma linguagem de programação que nos ajude a trabalhar de forma mais eficiente dentro da ciência do dado, de imediato pensamos na linguagem Python <https://www.python.org/>, com sua sintaxe simples e seu grande leque de bibliotecas, que tornam a vida de quem desenvolve algoritmos preditivos mais fácil.

Porém, para escrever nossas primeiras linhas de código precisamos de um ambiente que interprete o Python. Com isso, o Google Colab se torna uma opção de fácil acesso, pois basta ter uma conta Google para usá-lo.

Ao acessar o site do Google Colab com a sua conta, você terá uma visão parecida com essa:

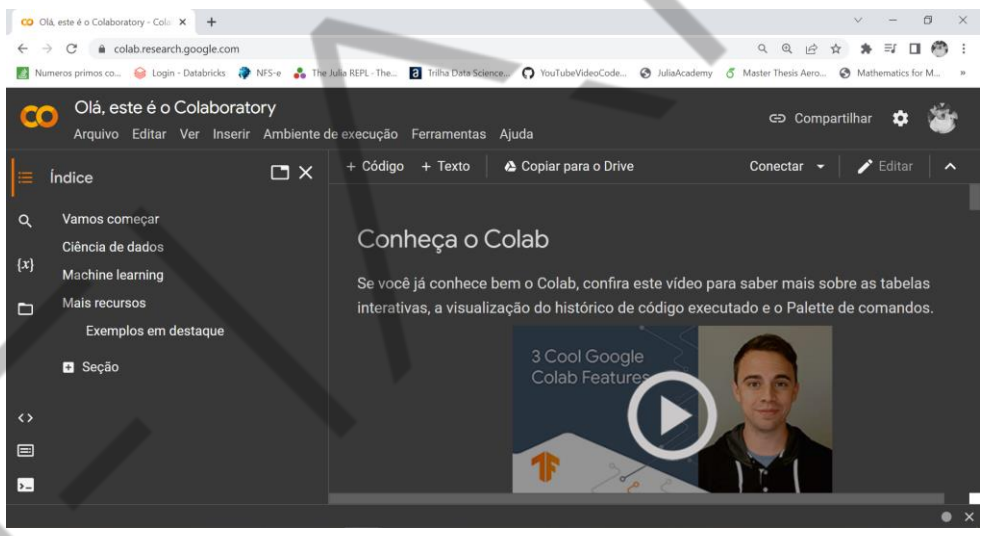


Figura 1 – Acessando o Google Colab
Fonte: Elaborado pelo autor (2023)

Para iniciar um notebook novo, basta clicar em Arquivo > Novo notebook e uma nova aba do navegador se abrirá:

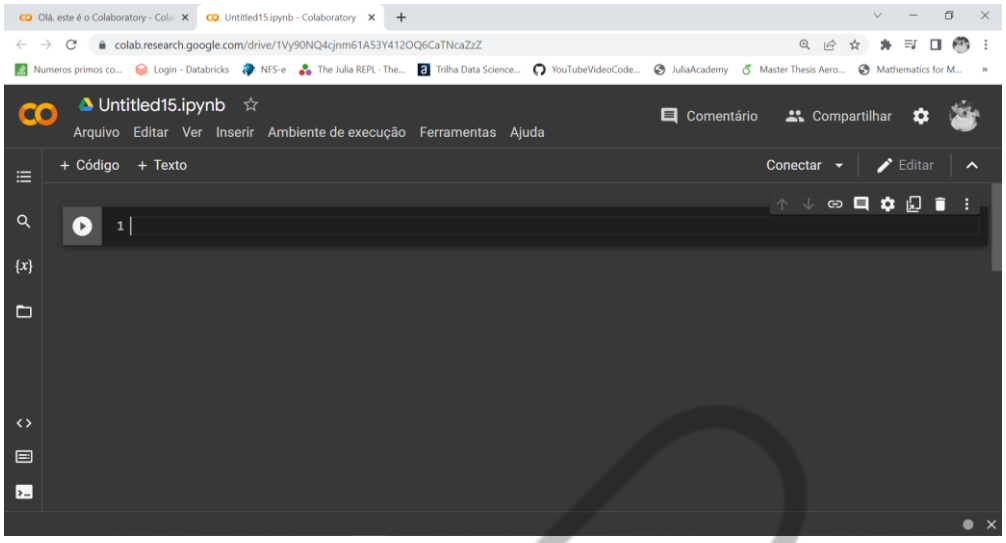


Figura 2 – Abrindo um novo notebook
Fonte: Elaborado pelo autor (2023)

A partir daqui você consegue codificar em Python de maneira tranquila e com muitos recursos do próprio ambiente em nuvem do Google. Mas, antes de começar, é importante que iniciemos o kernel da máquina virtual que suporta o notebook. Para isso, clique no botão conectar:

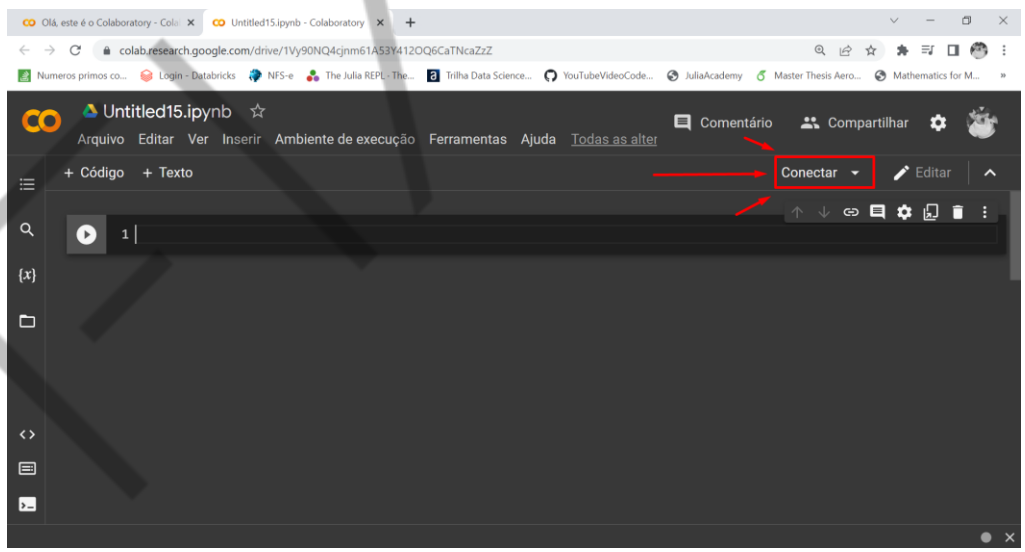


Figura 3 – Conectando ao kernel na máquina em nuvem
Fonte: Elaborado pelo autor (2023)

Após conectar seu ambiente, você verá a seguinte mudança:

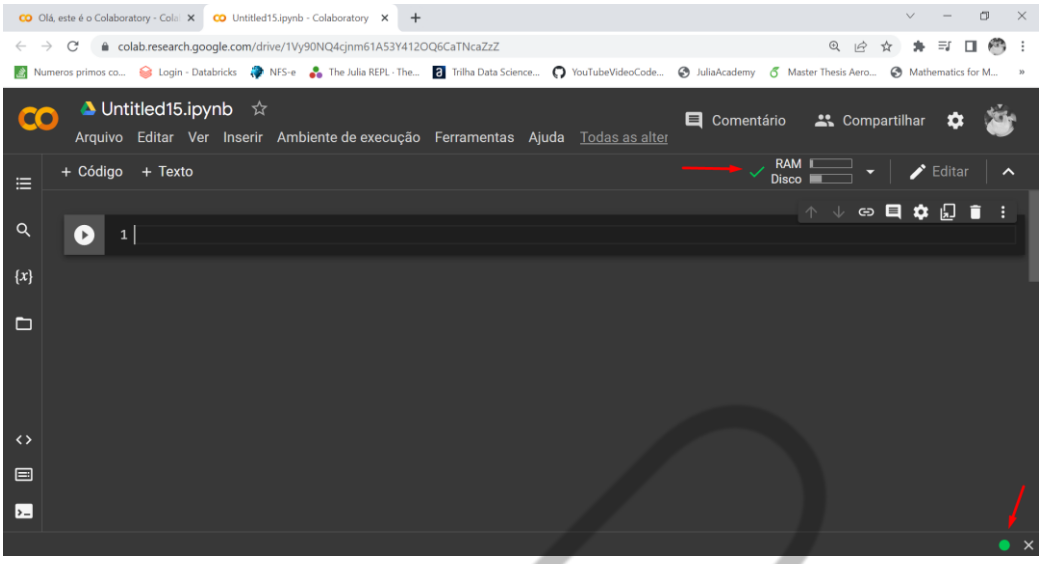


Figura 4 – Verificando a conexão do ambiente
Fonte: Elaborado pelo autor (2023)

Agora está tudo pronto para você começar a escrever suas primeiras linhas de código nessas células.

Dica de leitura:

A própria plataforma do Google Colab conta com um tutorial <https://colab.research.google.com/drive/16pBJQePbqkz3QFV54L4NIkOn1kwpuRrj> para trabalhar no ambiente.

Antes de fazer os primeiros códigos em Python manipulando bases de dados reais, é sempre importante entender por onde começar a direcionar e trazer dados reais, em que será possível testar nossos conhecimentos.

É evidente que, para cada desafio que você encontrar no dia a dia, será necessário adquirir dados de diferentes fontes e também de diferentes formatos.

Um excelente lugar de dados nacionais para consulta e análise é o DATASUS <https://datasus.saude.gov.br/informacoes-de-saude-tabnet/> (link direto para Produção Hospitalar <http://www2.datasus.gov.br/DATASUS/index.php?area=0202&id=11633&VObj=http://tabnet.datasus.gov.br/cgi/deftohtm.exe?sih/cnv/qi>) usando o sistema Tabnet. Uma maneira fácil de você começar a fazer as primeiras manipulações e posteriormente acompanhar nosso HANDS ON, é acessar as bases já trazidas em arquivos .csv em nosso Github <https://github.com/alura-tech/pos-datascience-analise-e-exploracao-de-dados/archive/refs/heads/Dados.zip>.

Voltando ao Python

Agora vamos ver e entender um pouquinho das bibliotecas que nos ajudarão na manipulação dos dados e também na visualização dos mesmos (Pandas <https://pandas.pydata.org/> e Matplotlib <https://matplotlib.org/>, que inclusive, já vem instaladas no ambiente do Google Colab).

A biblioteca Pandas é uma das mais completas quando o assunto é analisar dados, porque ela nos permite ler e gravar arquivos em diversas extensões (.csv, .xlsx, .parquet, .txt, .sas, .pkl, .html, .hdf , ...), além de ler querys e tabelas em bancos de dados (desde que você conecte o python no banco que desejar).

Já a biblioteca matplotlib é a biblioteca mais usada para visualização de dados em forma de gráfico, além de ser base para a criação de outras bibliotecas (Seaborn <https://seaborn.pydata.org/> e Plotly <https://plotly.com/>). Aqui temos várias possibilidades no que diz respeito a criar e manipular gráficos, desde o tipo, até ajustes visuais, como plano de fundo, eixos, tickets, títulos, legendas, subplots etc.

Um dos recursos mais importantes do Matplotlib é sua capacidade de funcionar bem com muitos sistemas operacionais e back-ends gráficos. O Matplotlib suporta dezenas de back-ends e tipos de saída, o que significa que você pode contar com ele para funcionar independentemente de qual sistema operacional você está usando ou qual formato de saída você deseja. Essa abordagem de plataforma cruzada, tudo para todos, tem sido um dos grandes pontos fortes do Matplotlib. Isso levou a uma grande base de usuários, que por sua vez levou a uma base ativa de desenvolvedores e às poderosas ferramentas e onipresença do Matplotlib no mundo científico do Python.

Instalar essas bibliotecas é relativamente simples! Basta apenas que executemos os seguintes comandos no terminal:

```
pip install matplotlib
```

e

```
pip install pandas
```

A visualização e manipulação de dados são habilidades importantes para qualquer pessoa que tente extrair e comunicar insights de dados. No campo do aprendizado de máquina, a visualização desempenha um papel fundamental em todo o processo de análise.

O QUE VOCÊ VIU NESTA AULA?

Nesta aula, vimos como iniciar no ambiente do Google Colab, além de fazer nossa primeira leitura e manipulação de dados com a biblioteca Pandas, ressaltando algumas possíveis maneiras de se ler um dado vindo de uma fonte real.

Daqui para a frente, é importante que você replique os conhecimentos adquiridos para fortalecer mais suas bases e conhecimentos, já que um bom ou uma boa cientista de dados não é somente aquele(a) que é uma enciclopédia humana, mas sim aquele(a) que sabe ler um problema e atuar com eficácia.

IMPORTANTE: não esqueça de praticar com o desafio da disciplina, para que assim você possa aprimorar os seus conhecimentos!

Você não está sozinho ou sozinha nesta jornada! Te esperamos no Discord e nas *lives* com os nossos especialistas, onde você poderá tirar dúvidas, compartilhar conhecimentos e estabelecer conexões!

REFERÊNCIAS

DATASUS. **<https://datasus.saude.gov.br/>.** Acesso em: 06 fev 2023.

DOCUMENTAÇÃO PANDAS. **<https://pandas.pydata.org/>.** Acesso em: 06 fev 2023.

GOOGLE COLAB. **<https://colab.research.google.com/>.** Acesso em: 06 fev 2023.

TABNET. **<https://datasus.saude.gov.br/informacoes-de-saude-tabnet/>.** Acesso em: 06 fev 2023.

PALAVRAS-CHAVE

Python. Pandas. Dataframe.