



**Fecomércio  
Sesc**

**Big Data**

**Prof. Marco Mialaret**

Abril

2024



# Big Data



---

Onde me encontrar:

<https://www.linkedin.com/in/marco-mialaret-junior/>

e

<https://github.com/MatmJr>

# Uma luz no fim do túnel: AtlasDB

# Big Data

MongoDB. Products ▾ Resources ▾ Solutions ▾ Company ▾ Pricing

Q Support Sign In

Try Free

MONGODB ATLAS

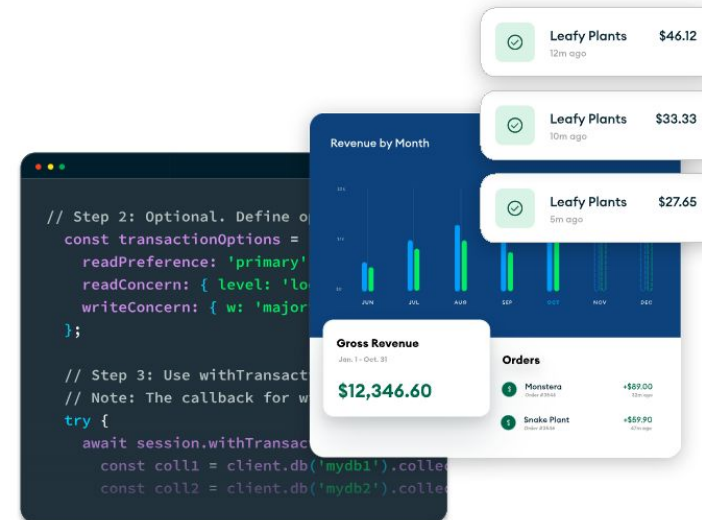
Loved by developers.  
Built for

## Transactions

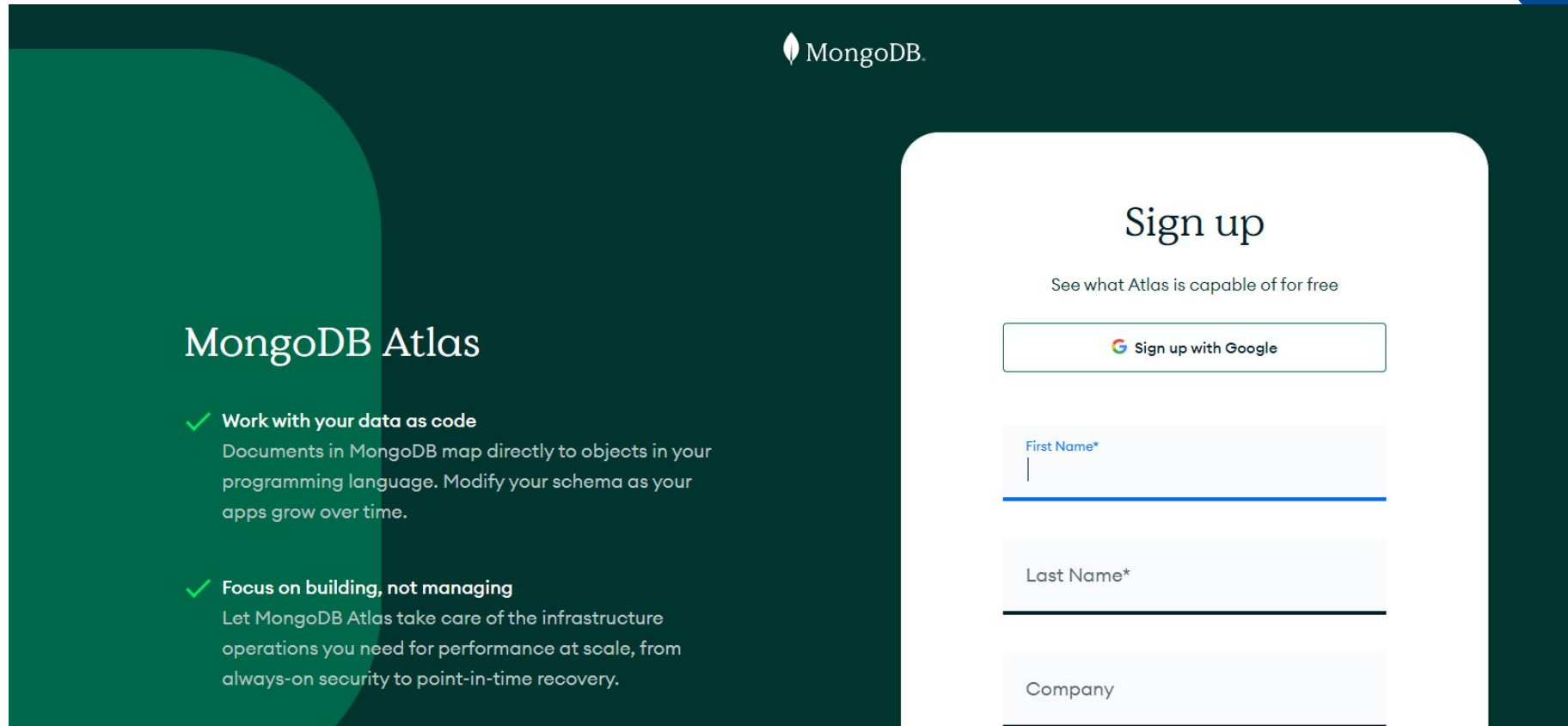
You don't need a separate database to support transactions, rich search, or genAI. The world's most popular document database is now the world's most versatile developer data platform.

Start Free

Learn More →



# Big Data



The screenshot shows the MongoDB Atlas sign-up page. On the left, the MongoDB logo is at the top right. Below it, the text 'MongoDB Atlas' is displayed. Underneath, there are two green checkmarks with bold text and descriptive paragraphs. On the right, there is a white sign-up form with the title 'Sign up', a subtext 'See what Atlas is capable of for free', a 'Sign up with Google' button, and three input fields for 'First Name\*', 'Last Name\*', and 'Company'.


MongoDB

## MongoDB Atlas

- ✓ **Work with your data as code**  
Documents in MongoDB map directly to objects in your programming language. Modify your schema as your apps grow over time.
- ✓ **Focus on building, not managing**  
Let MongoDB Atlas take care of the infrastructure operations you need for performance at scale, from always-on security to point-in-time recovery.

### Sign up

See what Atlas is capable of for free

 Sign up with Google

First Name\*

Last Name\*

Company



# Big Data

MARCO'S ORG - 2024-04-22 > PROJECT 0

## Overview



### Create a deployment

Choose your cloud provider, region, and specs.

[+ Create](#)

#### Toolbar

#### Featured Resources

##### GENERAL

[Get Started with Atlas](#)

[Reference MongoDB Documentation](#)

[Develop Applications with the Developer Center](#)

[Ask the MongoDB Community](#)

#### New On Atlas

**2 NEW**

[Learn about the latest feature enhancements on Atlas.](#)

# Big Data

## Deploy your database

Use a template below or set up advanced configuration options. You can also edit these configuration options once the cluster is created.

☐ M10

\$0.12/hour

For production applications with sophisticated workload requirements.

STORAGE	RAM	vCPU
10 GB	2 GB	2 vCPUs

☐ Serverless

For application development and testing, or workloads with variable traffic.

STORAGE	RAM	vCPU
Up to 1 TB	Auto-scale	Auto-scale

☒ M0

Free

For learning and exploring MongoDB in a cloud environment.

STORAGE	RAM	vCPU
612 MB	Shared	Shared

✔ **Free forever!** Your M0 cluster is ideal for experimenting in a limited sandbox. You can upgrade to a production cluster anytime.

### Name

You cannot change the name once the cluster is created.

Cluster0

☒ Automate security setup ⓘ

☒ Add sample dataset ⓘ

### Provider



Google Cloud

Azure

# Big Data



## Edit IP Access List Entry ✕

Atlas only allows client connections to a cluster from entries in the project's IP Access List. Each entry should either be a single IP address or a CIDR-notated range of addresses. [Learn more.](#)

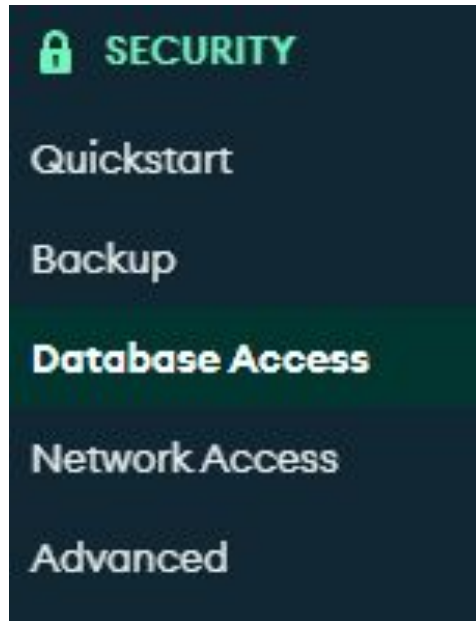
**ALLOW ACCESS FROM ANYWHERE**

**Access List Entry:**

**Comment:**



# Big Data



**Authentication Method**

Password

Certificate

AWS IAM  
(MongoDB 4.4 and up)



**PREVIEW**  
Federated Auth  
(MongoDB 7.0 and up) ⓘ

MongoDB uses [SCRAM](#) as its default authentication method.

**Password Authentication**

marcomialaret

Enter new password SHOW

 Autogenerate Secure Password  Copy

# Big Data

Usar a connection string fornecida pelo atlas

### Connect to ClusterTest

✓

✓

3

Set up connection securityChoose a connection methodConnect

#### Connecting with MongoDB for VS Code

**1. Install MongoDB for VS Code.**  
In **VS Code**, open "Extensions" in the left navigation and search for "MongoDB for VS Code." Select the extension and click install.

**2. In VS Code, open the Command Palette.**  
Click on "View" and open "Command Palette."  
Search "MongoDB: Connect" on the Command Palette and click on "Connect with Connection String."

**3. Connect to your MongoDB deployment.**  
Paste your connection string into the Command Palette.

`mongodb+srv://marcomialaret:<password>@cluster-test.5nzhzqx.mongodb.net/`

Replace **<password>** with the password for the **marcomialaret** user. Ensure any options are [URL encoded](#).

# Acessando o banco com python

# Big Data

---

Criar um venv com o requirements.txt da semana 07. Acesse o diretório no qual o requirements se encontra e execute o comando

```
python -m venv venv && .\venv\Scripts\activate  
&& pip install -r requirements.txt
```

# Big Data



No VScode (se você tem o docker configurado):

```
from pymongo import MongoClient

# Crie uma conexão com o MongoDB (ajuste o host e a porta conforme
# necessário)
client = MongoClient("mongodb://root:root@localhost:27017/admin")
```

✓ 2.7s

Python

# Big Data

---

No Colab (se você tem o atlas configurado):

```
from pymongo.mongo_client import MongoClient
from pymongo.server_api import ServerApi

uri = "mongodb+srv://<username>:<password>@<cluster>/admin"

client = MongoClient(uri, server_api=ServerApi('1'))

try:
    client.admin.command('ping')
    print("Pinged your deployment. You successfully connected to MongoDB!")
except Exception as e:
    print(e)
```



# Big Data

---

```
# Seleciona o banco de dados
db = client["aula07"]

# Seleciona a coleção
collection = db["cotacoes"]
```

✓ 0.0s

Python

# Big Data

---

```
import json

# Carrega os dados do arquivo JSON
with open("cotacoes.json", "r") as file:
    data = json.load(file)
```

✓ 0.0s

Python

# Big Data

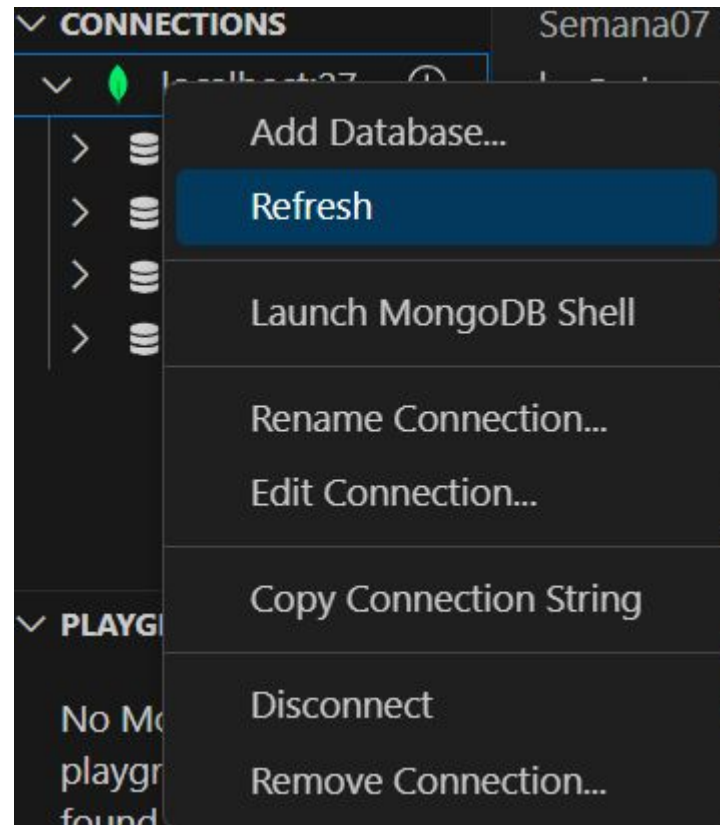
---

```
collection.insert_one(data)
```

✓ 0.9s

# Big Data

---



# Big Data



The screenshot shows a MongoDB interface. On the left, the 'CONNECTIONS' panel is expanded, showing a connection to 'localhost:27017...'. Under this connection, the 'aula07' database is selected, and the 'cotacoes' collection is expanded. A document with the key '661fe1414d0...' is selected. The main panel on the right displays the JSON document structure for this document. The document has a top-level key 'aula07.cotacoes' with a value of a JSON array containing one object. This object has three fields: '\_id' (a string), 'USDBRL=X' (an object), and 'Open' (an object). The 'USDBRL=X' object has a single key 'Open' with a value of an array of four floating-point numbers. The 'Open' object has four keys representing dates from 2023-04-17 to 2023-04-20, each with a corresponding floating-point value.

```
{  
  "aula07.cotacoes": [  
    {  
      "_id": "661fe1414d0d69f8d2e883c1",  
      "USDBRL=X": {  
        "Open": {  
          "2023-04-17": 4.908699989318848,  
          "2023-04-18": 4.939799785614014,  
          "2023-04-19": 4.984300136566162,  
          "2023-04-20": 5.0742998123168945,  
        }  
      }  
    }  
  ]  
}
```

# Usando o dataset para fazer previsão



# Big Data

---

Agora dispomos de um serviço de banco de dados na nuvem que contém um arquivo específico. Em seguida, simularemos o acesso a esse arquivo para realizar previsões de cenários futuros.

O dataset em questão representa a cotação do dólar (em reais) ao longo dos últimos 10 anos.

Primeiro, vamos iniciar o processo acessando o JSON que estava no banco de dados.

# Big Data

---

```
import json
```

```
with open('cotacoes.json', 'r') as _json:  
    data = json.loads(_json.read())
```

# Big Data

---

Dentro do arquivo JSON, vamos acessar a tag que contém a cotação do dólar de fechamento da bolsa.

```
close = data["USDBRL=X"]["Close"]
```

# Big Data

---

Por fim, convertendo o arquivo JSON para um objeto pandas.Series.

```
import pandas as pd
```

```
# Transformar em uma pandas.Series  
data_series = pd.Series(close)
```

```
# Converter o índice para tipo data  
data_series.index = pd.to_datetime(data_series.index)
```

# Big Data

---

Para visualizar o gráfico da cotação do dólar, podemos utilizar a biblioteca matplotlib, uma das ferramentas mais populares para a criação de gráficos em Python. Ela oferece uma ampla gama de funcionalidades que permitem a construção de gráficos estáticos, animados e interativos de maneira simples e eficaz. Através do matplotlib, é possível customizar gráficos com títulos, legendas e escalas, tornando-os uma escolha versátil para análise e apresentação de dados.

# Big Data

---

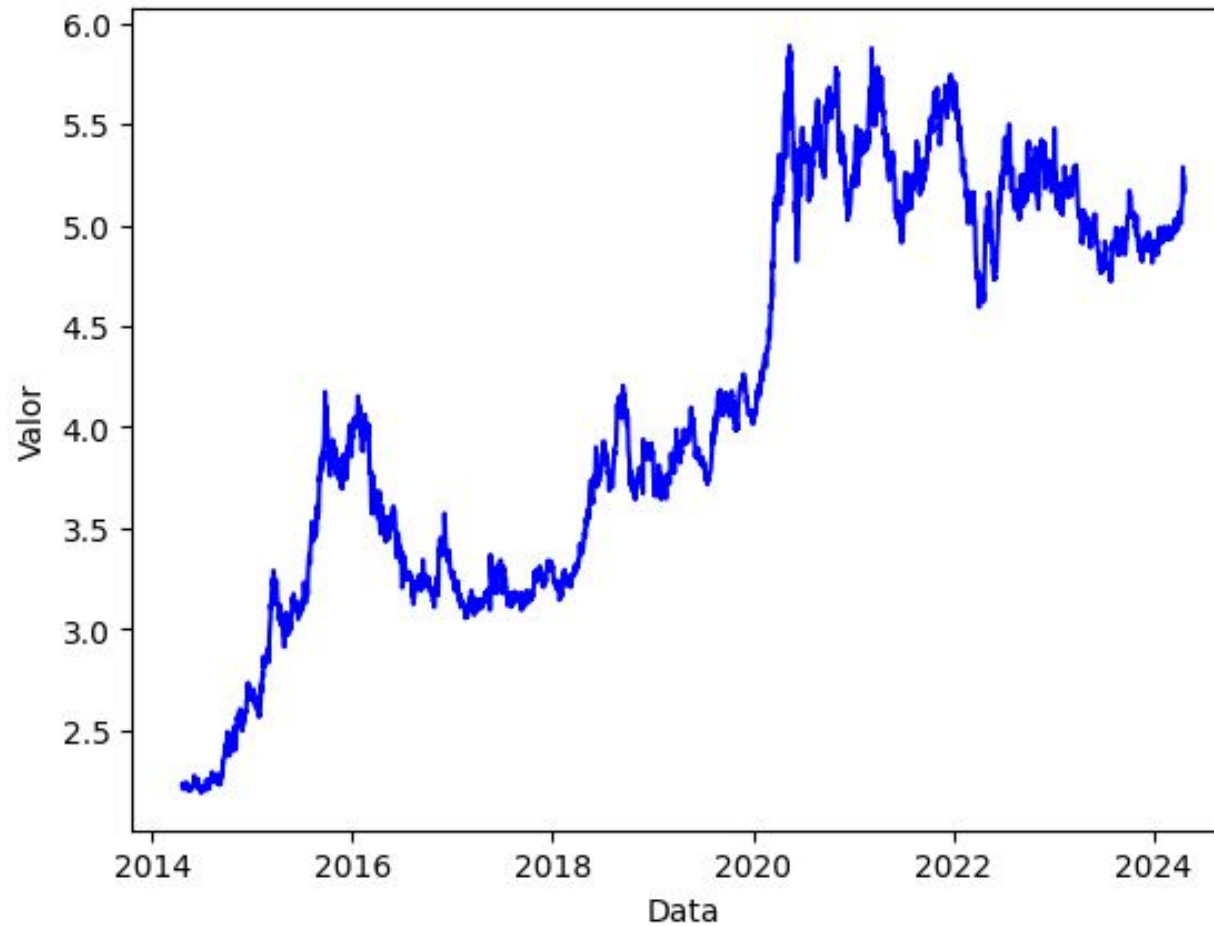
```
import matplotlib.pyplot as plt

plt.plot(data_series, linestyle='-', color='b') # Plota a série
temporal
plt.title('Gráfico da Série Temporal') # Título do gráfico
plt.xlabel('Data') # Rótulo do eixo X
plt.ylabel('Valor') # Rótulo do eixo Y
plt.show() # Exibe o gráfico
```



# Big Data

Gráfico da Série Temporal



# Big Data



---

Para realizar previsões de cenários futuros, podemos utilizar a biblioteca Prophet, desenvolvida pelo Facebook. O Prophet é especializado em previsões para séries temporais, sendo robusto a dados faltantes e mudanças de tendência, e capaz de lidar com efeitos sazonais de diferentes granularidades. Essa ferramenta é particularmente útil para dados que exibem padrões sazonais fortes e várias temporadas de histórico. O Prophet facilita a modelagem de séries temporais, permitindo que analistas e desenvolvedores produzam previsões confiáveis com facilidade.

# Big Data

---

```
from prophet import Prophet

data_frame = data_series.reset_index()
data_frame.columns = ['ds', 'y']

# Criar e ajustar o modelo
model = Prophet(daily_seasonality=True)
model.fit(data_frame)
```

# Big Data

---

```
# Criar um DataFrame para as datas futuras  
future_dates = model.make_future_dataframe(periods=30)
```

```
# Fazer as previsões  
forecast = model.predict(future_dates)
```

```
# Plotar os resultados  
model.plot(forecast)  
plt.show()
```

# Big Data



# Big Data

---

Exibir componentes da previsão (tendência, sazonalidade)

```
model.plot_components(forecast)  
plt.show()
```



# Big Data

---

Esta é uma solução inicial, e percebemos que o modelo ainda não está captando a tendência de alta na cotação que está ocorrendo atualmente. Muitos ajustes são necessários para melhorar a qualidade do ajuste. Ajustar parâmetros, incorporar variáveis externas e refinar a modelagem sazonal são algumas das estratégias que podem ser exploradas para aprimorar as previsões do modelo.

# Big Data

---

Por exemplo, podemos informar ao modelo as datas que foram feriados no Brasil e os períodos que tivemos crise financeira (levei em consideração o período da copa/olimpíadas 2015-2016 e o período da covid 2021-2022).

# Big Data

---

```
from prophet import Prophet
```

```
# Preparar os dados
```

```
data_frame = data_series.reset_index()
```

```
data_frame.columns = ['ds', 'y']
```

```
# Criar o modelo com changepoints manuais para os períodos de crise
```

```
changepoints = ['2015-01-01', '2016-01-01', '2021-01-01', '2022-01-01']
```

# Big Data

---

```
model = Prophet(changepoints=changepoints, daily_seasonality=True)
model.add_country_holidays(country_name='BR')
```

```
# Ajustar o modelo
model.fit(data_frame)
```

```
# Criar um DataFrame para as datas futuras
future_dates = model.make_future_dataframe(periods=30)
```

# Big Data

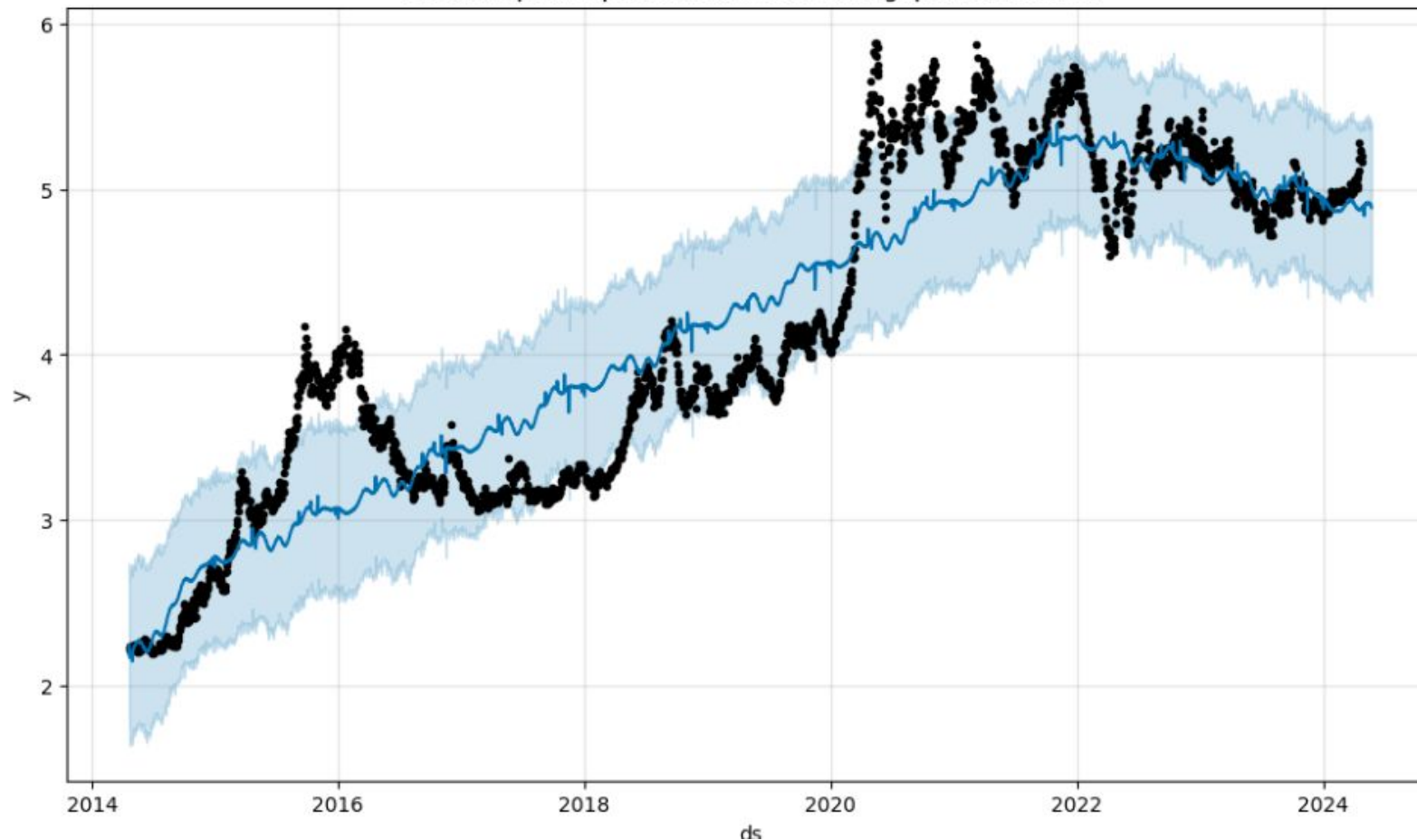
---

```
# Fazer as previsões com o modelo ajustado  
forecast = model.predict(future_dates)
```

```
# Plotar as previsões  
fig = model.plot(forecast)  
plt.title('Previsão para o próximo ano com Changepoints de Crise')  
plt.show()
```

# Big Data

Previsão para o próximo ano com Changepoints de Crise



# Big Data

---

Com os ajustes realizados, houve uma perda de precisão nos dados históricos, porém, os dados mais recentes passaram a se alinhar com a faixa esperada. É importante destacar que este modelo ainda está distante de ser considerado confiável. Para alcançar um modelo mais preciso, seria essencial realizar um estudo aprofundado em séries temporais e economia, o que permitiria refinar as técnicas de modelagem e integrar conhecimentos específicos do mercado financeiro.

# Big Data

---

Exercício: Fazer um modelo para o preço de abertura.



# Dúvidas?

---



**Marco Mialaret, MSc**

**Telefone:**

**81 98160 7018**

**E-mail:**

**marcomialaret@gmail.com**

