

Indonesia Bank Stock Visualization

Feni Ismiati | ismiatifeni6@gmail.com | 6282210023177



Background

The Indonesia Bank Stock Visualization project aims to provide a comprehensive analysis of historical stock data by focusing on 6 bank stocks. This project explores the patterns and relationships among bank stocks over the period of time. Understanding these patterns and relationships are crucial for analysts looking to be informed based on historical performance.

Objectives

The project aims to achieve the following objectives

1. Maximum and Minimum Closing Price

Identify the maximum and minimum closing prices for each bank stock

2. Specific Dates for Maximum and Minimum Returns

Determine the specific dates when each bank stock reached its maximum and minimum return

3. Average Return for Each Bank Stock

Calculate the average return to assess the overall performance of each bank stocks

4. Volatility Analysis

Identify the most volatile and the most stable bank stocks based on the standard deviation of returns

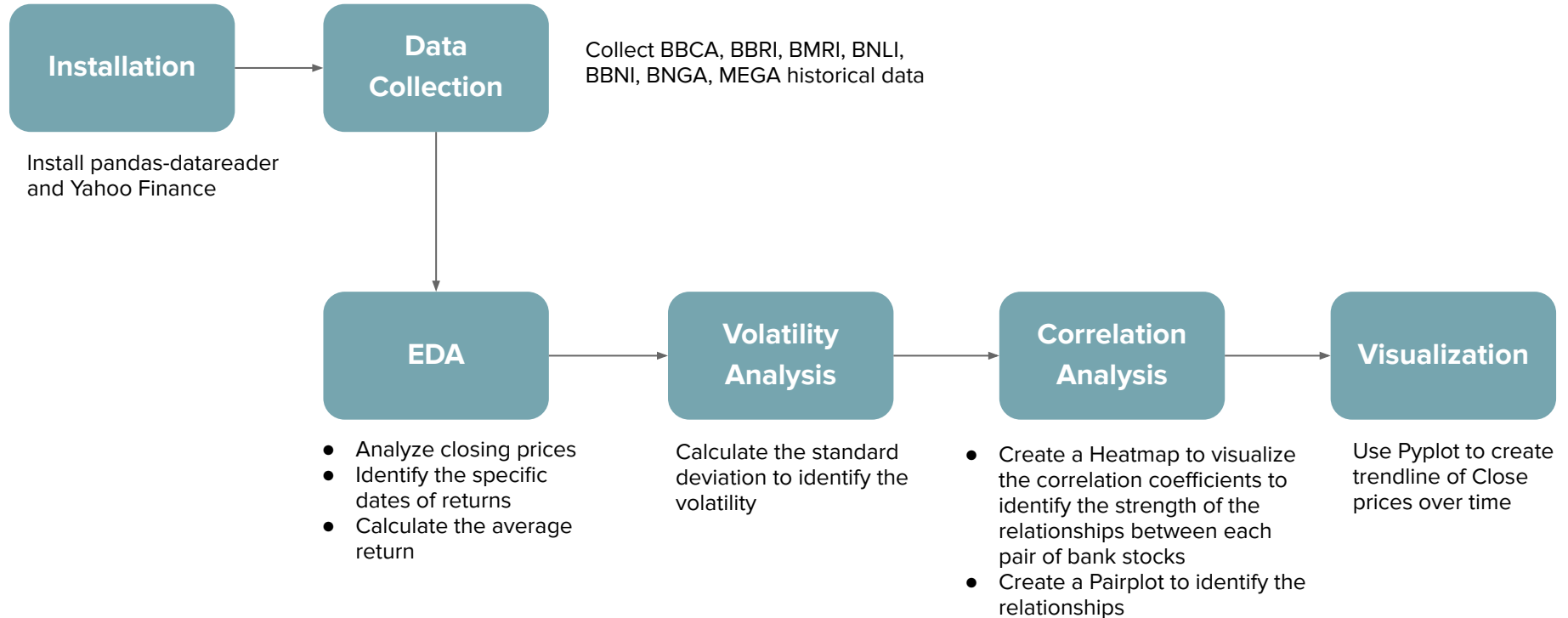
5. Correlation Analysis

Examine the correlation between each pair of bank stocks to determine the strength of their relationships

! Disclaimer !

This project is intended for educational and analytical purposes. It shows the actual historical data of Indonesian Bank Stocks and doesn't serve as a recommendation or endorsement of any specific bank stock. **The information provided here should not be interpreted as investment advice.** Please conduct your own research or consult with a financial advisor before making any investment decisions.

Method



Installation

Installation

```
# Install pandas-datareader  
!pip install pandas-datareader
```

```
# Install Yahoo Finance  
!pip install yfinance
```

- **Install pandas-datareader:** it is a Python library that allows easy access to financial data from various online sources directly into Pandas dataframes
- **Install Yahoo Finance:** in this project, Yahoo Finance source is used to fetching the data. Yahoo Finance also provides a wide range of financial data, including stock prices, dividends, and splits

Data Collection

1. Import Libraries

There are several libraries used in this project

1. **Data Collection:** `pandas_datareader` and `yfinance` to gather the up-to-date financial data
2. **Data Analysis:** `pandas` and `numpy` to robust data manipulation and essential for calculating returns, average, and standard deviation
3. **Visualization:** `matplotlib`, `seaborn`, and `plotly` provide tools for creating interactive graphs, and explore trends, correlations, and patterns.
4. **Integration:** `cufflinks` and `plotly.express` to create interactive plots from dataframes and explanatory analysis

```
import pandas_datareader.data as web
import yfinance as yf
import pandas as pd
import numpy as np
import datetime
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.graph_objects as go
%matplotlib inline
sns.set_style('whitegrid')

# For Plotly
import plotly
import cufflinks as cf
cf.go_offline()
# to avoid deprecated in newer versions of libraries like Pandas or Plotly
import plotly.express as px
```

2. Determine the Data

In this project, the historical of bank stock data is from period **31 December 2013** to **31 December 2023**

These are the Indonesia's Bank Stocks used in this project:

1. PT Bank Central Asia Tbk (**BBCA**)
2. PT Bank Rakyat Indonesia (Persero) Tbk (**BBRI**)
3. PT Bank Mandiri (Persero) Tbk (**BMRI**)
4. PT Bank Permata Tbk (**BNLI**)
5. PT Bank Negara Indonesia (Persero) Tbk (**BBNI**)
6. PT Bank CIMB Niaga Tbk (**BNGA**)
7. PT Bank Mega Tbk (**MEGA**)

```
# define the start - end date
start_date = datetime.datetime(2013, 12, 31).strftime('%Y-%m-%d')
end_date = datetime.datetime(2023, 12, 31).strftime('%Y-%m-%d')
```

```
# List of Indonesian bank stock tickers
bank_tickers = ['BBCA.JK', 'BBRI.JK', 'BMRI.JK', 'BNLI.JK', 'BBNI.JK', 'BNGA.JK', 'MEGA.JK']
```

3. Download the Data

- Retrieve financial data from **Yahoo Finance** using **yfinance** library to download the data.
- The **download** code is a method from the **yfinance** library that retrieves stock data for the specified ticker.
- The **ticker** used is the bank stock symbol to download the data and store in the **stock_data** dataframe

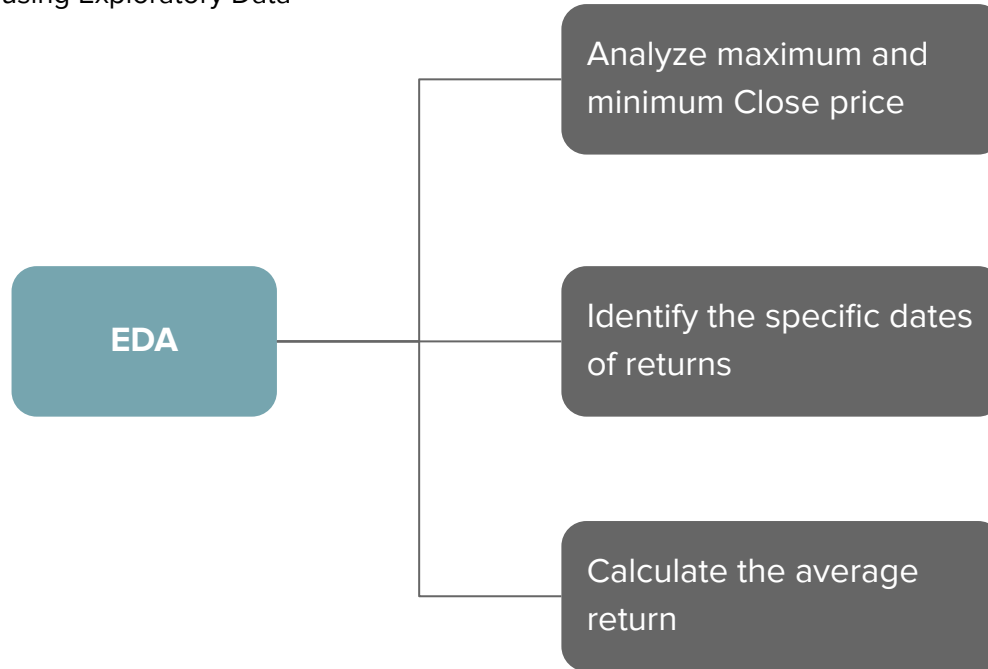
```
# create a loop from Yahoo Finance to Bank Tickers
for ticker in bank_tickers:
    stock_data[ticker] = yf.download(ticker, start=start_date, end=end_date)
```

```
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
[*****100%*****] 1 of 1 completed
```

EDA

(Exploratory Data Analysis)

In this section, there are several objectives that need to be answered using Exploratory Data Analysis (EDA)



Analyze maximum and minimum Close price

What is the maximum and minimum closing price for each bank stock during the specified period?

Create `max()` and `min()` from pandas library to determine the maximum and minimum of Close price for each bank stock. It shows that the range of **maximum close price is 1,790 - 9,400**. Meanwhile the range of **minimum close price is 382 - 1,865**.

Maximum Close price

```
# checking the maximum of Close price
max_close_prices = close_prices.max()

print(max_close_prices)
```

Bank Ticker	
BBCA.JK	9400.000000
BBRI.JK	5725.000000
BMRI.JK	6125.000000
BNLI.JK	2930.000000
BBNI.JK	5375.000000
BNGA.JK	1790.000000
MEGA.JK	7496.647461

dtype: float64

Minimum Close price

```
# checking the minimum of Close price
min_close_prices = close_prices.min()

print(min_close_prices)
```

Bank Ticker	
BBCA.JK	1865.000000
BBRI.JK	1277.250610
BMRI.JK	1860.000000
BNLI.JK	382.846039
BBNI.JK	1580.000000
BNGA.JK	440.000000
MEGA.JK	1089.378906

dtype: float64

Identify the specific dates of returns

On which specific dates did each bank stock reach its maximum and minimum return values?

First, we should know what does **Return** mean? **Return** is basically represent the gain or loss of an investment over a specified period. Therefore, **Return** calculation is:

$$r_t = \frac{p_t - p_{t-1}}{p_{t-1}} = \frac{p_t}{p_{t-1}} - 1$$

rt: the return of the stock at time **t**. This represents the percentage change in the stock's price from the previous day

pt: the stock's closing price at time **t**. In this project **t** is current day

pt-1: the stock's closing price at time **t-1** or the previous day

Processing

pct_change use to define the percent change

```
# create Loop through bank ticker and returns for the 'Close' prices
for tick in bank_tickers:
    # Calculate the percentage change for 'Close' prices and store it in the 'returns' DataFrame
    returns[tick + ' Return'] = df_stock[tick]['Close'].pct_change()

# Display the first few rows of the returns DataFrame
print(returns.head())
```

So, to calculate the Return in bank stock, **pct_change()** function from **Pandas** is used to calculate the percentage change. This basically calculates the percentage change in the closing prices for each bank stock between the current and previous

Identify the specific dates of returns

On which specific dates did each bank stock reach its maximum and minimum return values?

Minimum Return

```
# specific dates did each bank stock reach its minimum returns  
returns.idxmin()
```

```
BBCA.JK Return    2020-03-17  
BBRI.JK Return    2020-02-27  
BMRI.JK Return    2020-03-17  
BNLI.JK Return    2019-11-07  
BBNI.JK Return    2020-03-09  
BNGA.JK Return    2015-04-24  
MEGA.JK Return    2017-02-10  
dtype: datetime64[ns]
```

Maximum Return

```
# specific dates did each bank stock reach its maximum returns  
returns.idxmax()
```

```
BBCA.JK Return    2020-03-26  
BBRI.JK Return    2020-03-26  
BMRI.JK Return    2020-03-26  
BNLI.JK Return    2020-10-09  
BBNI.JK Return    2020-06-08  
BNGA.JK Return    2016-07-20  
MEGA.JK Return    2019-03-05  
dtype: datetime64[ns]
```

Findings

From the result above, it can be seen that mostly of the bank stock experience **minimum** and **maximum** returns on **2020** between **February - March**.

Calculate the average return

What is the average return for each bank stock over the analyzed period?

Calculate the average of Return for each of bank stock using `.mean()` function, *100 also to show it in percent format.

Findings

If we look at the result, it can be seen that the highest average of Return is coming from **MEGA (0.094997%)** compared to all the bank stocks. Meanwhile, the lowest average of Return is **BNLI (0.031911%)** over a period of given time

```
# The average return for each bank stock? (in percent)
```

```
returns.mean() * 100 # in percent, to be more readable
```

```
BBCA.JK Return    0.073608
```

```
BBRI.JK Return    0.078642
```

```
BMRI.JK Return    0.063475
```

```
BNLI.JK Return    0.031911
```

```
BBNI.JK Return    0.061046
```

```
BNGA.JK Return    0.047097
```

```
MEGA.JK Return    0.094997
```

```
dtype: float64
```

Volatility Analysis

Calculate the standard deviation to identify the volatility

Which bank stock is the most volatile, and which is the most stable, based on their standard deviation of returns?

```
# using Standard Deviation / .std() to indicate how much the returns deviate from their average
# so it can be said that the higher standard deviation means the stock's returns are more spread out and more volatile (risky)

returns.std()
```

```
BBCA.JK Return    0.014471
BBRI.JK Return    0.019925
BMRI.JK Return    0.019700
BNLI.JK Return    0.026902
BBNI.JK Return    0.020379
BNGA.JK Return    0.021407
MEGA.JK Return    0.027163
dtype: float64
```

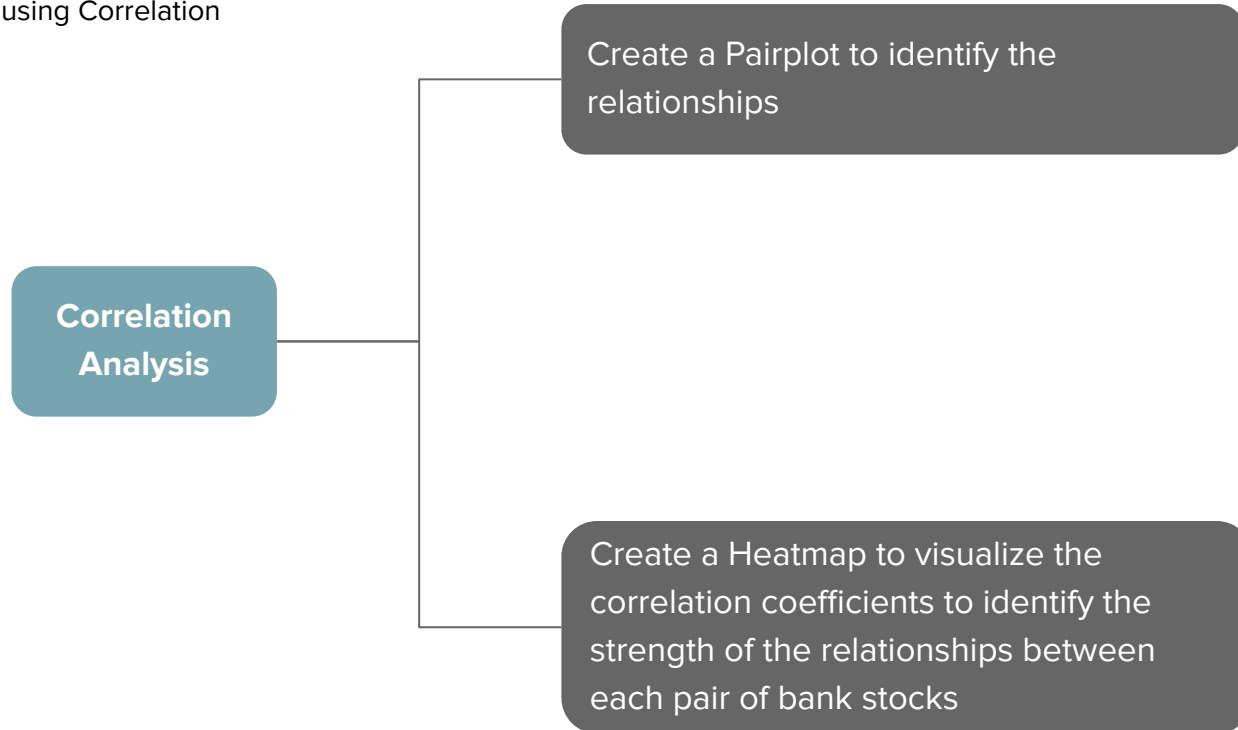
To answer which bank stock is the most volatile and most stable, `.std()` or standard deviation which used to indicate how much the returns deviate from their average. The conclusion is **the higher standard deviation value means that the stock returns are more volatile or more riskiest.**

Findings

The result above shows that **MEGA (0.027163)** has the highest of standard deviation value which it can be said that MEGA is the most volatile or riskiest. Meanwhile, **BBCA (0.014471)** has the lowest of standard deviation, so it can be conclude that BBKA is the most stable compared to all bank stocks.

Correlation Analysis

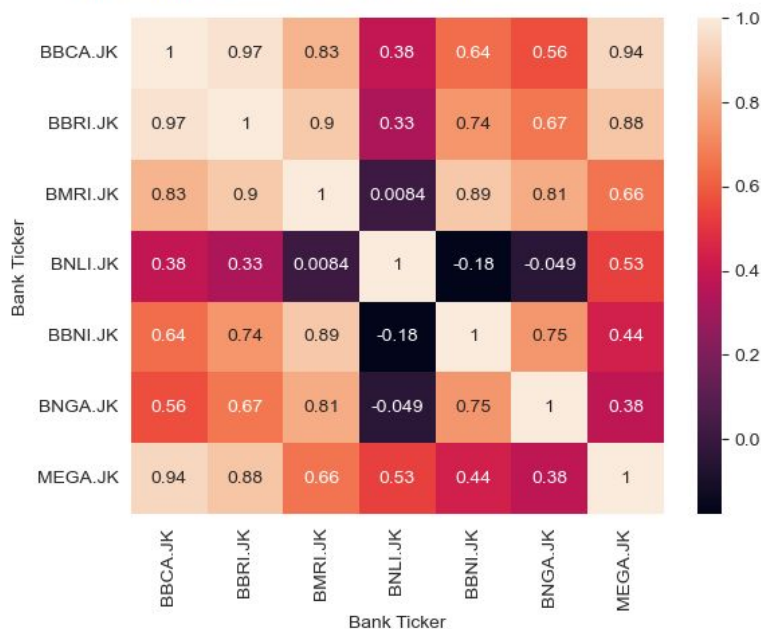
In this section, there are several objectives that need to be answered using Correlation



1. Create a Heatmap to visualize the correlation coefficients

What is the correlation between each pair of bank stocks, and how strong are these relationships?

<Axes: xlabel='Bank Ticker', ylabel='Bank Ticker'>



Heatmap function `.heatmap()` to shows the strength of the correlation between different bank stocks, highlighting which stocks move together and which others don't

Findings

If we look at the Heatmap Correlation between each pair of stocks, **BBKA has a positive correlation with BBRI (0.97), MEGA (0.94), and BMRI (0.83)** which means that if BBKA increases, they would also tend to increase.

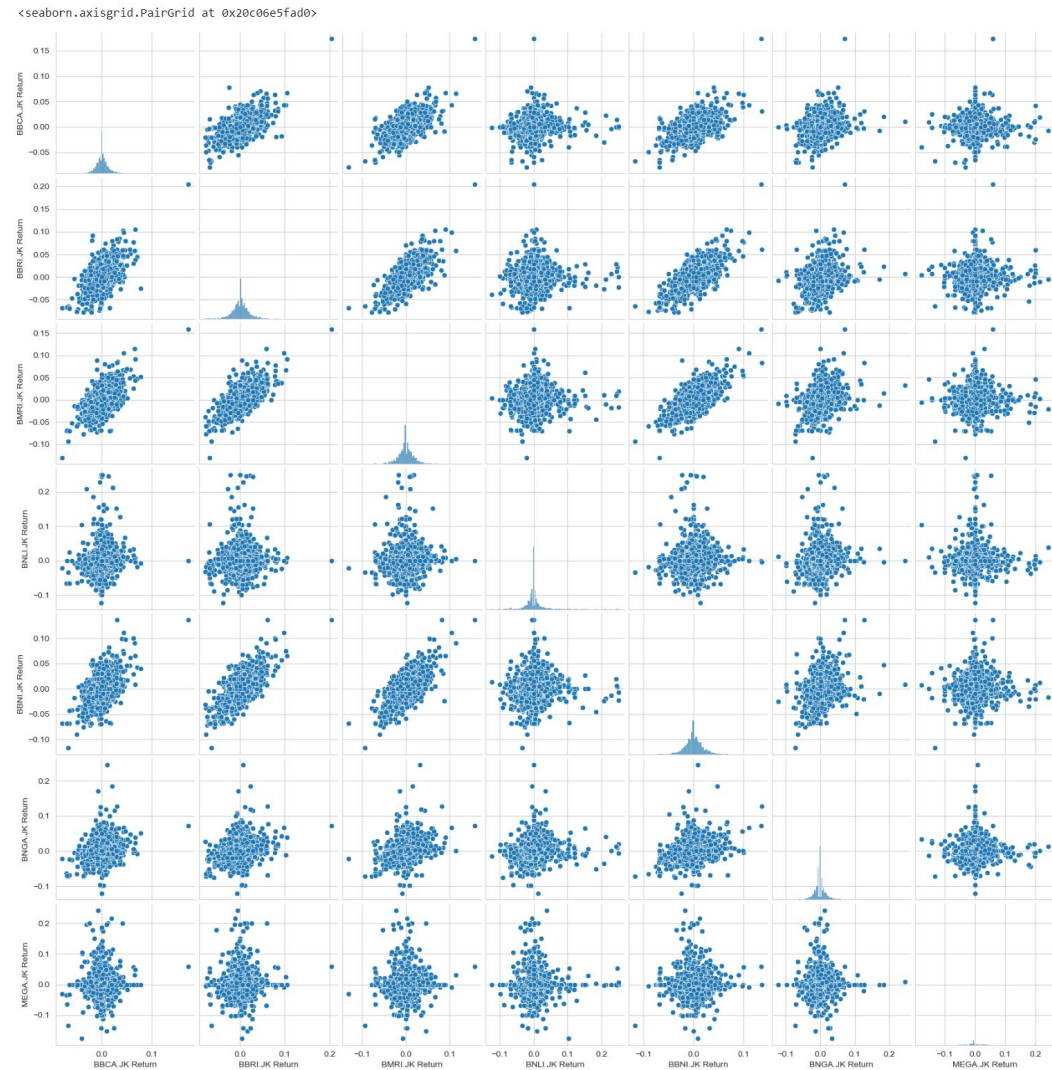
2. Create a Pairplot to identify the relationships

What is the correlation between each pair of bank stocks, and how strong are these relationships?

To identify the relationship, `.pairplot()` function is used to identify correlations, trends, and distribution patterns of each bank stocks

Findings

From the Pairplot visualization, it can be seen that **BBCA** has a linear positive correlation with **BBRI**, **BMRI**, and **BBNI**. So, if we compare it with MEGA it is more spread out.



Visualization

Use Pyplot to create trendline of Close prices over time

How is the Close price trendline using Pyplot?



Using `.iplot()` from Pyplot to create detailed and interactive exploration of Close price for each of bank stocks. You can also point your cursor to see the detailed of Close price for the bank stocks you want to see.

Findings

If we look at the Pyplot chart above, it can be seen that most of the bank stocks **tend to increase from 2013 to 2023**. But, in the in **2020 around February - March** most of the each bank stocks have experience decreased.

Summary

1. The range of maximum Close price from each of bank stock from 31 December 2013 to 31 December 2023 is 1,790 - 9,400. Meanwhile the minimum is 382 - 1,865
2. Most of the minimum and maximum returns of all the bank stocks are occurred in 2020
3. The highest return came from MEGA (0.094997%), but MEGA (0.027163) also has the highest of standard deviation value which it can be said that MEGA is the most volatile or riskiest
4. Based standard deviation, the lowest value of standard deviation is coming from BBKA (0.014471) which can be said that BBKA is the most stable, with the average of return is 0.073608%
5. To see the strength of correlation between each pair of bank stocks, we can see that BBKA shows a strong positive correlation with BBRI (0.97), MEGA (0.94), and BMRI (0.83) indicating that their prices tend to move together. But, Pairplot analysis reveals that the correlation with MEGA is less linear and more scattered compared to the more consistent linear relationships with BBRI, BMRI, and BBNI. This suggests that BBKA stock price movements are more predictably linked with BBRI, BMRI, and BBNI than with MEGA

Thank You!

Feni Ismiati | ismatifeni6@gmail.com | 6282210023177