Python初級數據分析員證書

（六）數據分析 及可視化專案

13. 數據分析專案
Demo3 – TA library

Review

- Statistics
- Hypothesis testing
- Algebra
- Linear regression
- Propositional logic
- Python
- R
- SQL

- Pandas, NumPy, SciPy
- Data Visualization, Matplotlib, Seaborn, Plotly
- Dashboard Visualization, Business Intelligence
- Storytelling

Chapter Summary

- Leading and Lagging Indicators
- TA-Lib introduction and installation
- STOCH RSI
- Bollinger Band
- MACD
- ADOSC
- ATR & TR

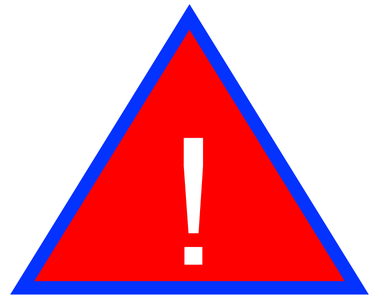
Recap

在上一章中，我們學習了多個移動平均線組合的簡單回測。Google Colab 是在雲上長時間運行計算的絕佳選擇。

在本章中，我們將發現更多 技術分析帶有庫的方法。TA 不僅用於金融和股票市場分析，還是分析時間序列問題的統計工具。我們經常使用股票數據，因為它的多樣性和來源的便利性。

Disclaimer

我們課程的所有內容僅用於一般性質的資訊和教育目的，不涉及任何特定個人或實體的任何情況。請勿將任何此類資訊或材料解釋為投資、財務、專業或任何其他建議。



領先指標和滯後指標

領先 和 滯後指標, 是評估效能時使用的兩種類型, 度量商業機構或經濟活動.
領先指標 被認為指向未來活動.

- Heads-up(提醒): 對於希望預測的經濟學家和投資者趨勢.
- 債券收益率: 被認為是股市的一個很好的領先指標 因為債券交易員對經濟趨勢進行預測和推測。

滯後指標 被視為確認 正在進行的模式。

- 廣為人知活動結束後，但這並不能使它們變得毫無用處。
- 他們可以澄清和確認 一種隨時間推移而發生的模式。
- 失業率是最可靠的滯後指標之一。但視乎要預測的是什麼，也可以是領先指標。

領先指標和滯後指標

金融業的熱門領先指標

- 相對強弱指數(RSI)
- Stochastic oscillator隨機(統計)振蕩
- Williams %R 威廉
- On-balance volume (OBV)

金融業流行的滯後指標

- 移動平均線
- MACD 指標
- Bollinger bands 保力加

僅僅依賴任何一個都可能對策略產生負面影響，這就是為什麼許多交易者的目標是找到一個**兩者的平衡。**

TA library

交易者會使用兩種常用的TA技術分析庫:

- Pandas-ta
  - 提供超過130+ 指標. 使用更方便.
  - Pandas TA - https://github.com/twopirllc/pandas-ta
- ta-lib
  - 快，並提供其他語言版本，如 C, Java, Perl
  - 由於它的後端是用C, 速度更快
  - TA Lib - https://github.com/TA-Lib/ta-lib-python

Ta-lib

TA-Lib is widely used by trading software developers requiring to perform technical analysis of financial market data.廣泛使用於：金融市場、數據技術分析、交易軟體開發。

- 包括150+ indicators such as ADX, MACD, RSI, Stochastic, Bollinger Bands, etc.
- Candlestick pattern recognition蠟燭圖形態識別
- Open-source API for C/C++, Java, Perl, Python and 100% Managed .NET
- Although it might be a little tricky while installation, it definitely a great tool afterwards. 雖然安裝時可能會有點棘手，但之後它絕對是一個很棒的工具。

Ta-lib

在本章中，我們將重點使用ta-lib. 首先，我們來安裝 ta library.

https://github.com/TA-Lib/ta-lib-python

Add two lines of scripts in .zshrc

Then source .zshrc

```
156  # TA-Lib
157  export TA_INCLUDE_PATH="$(brew --prefix ta-lib)/include"
158  export TA_LIBRARY_PATH="$(brew --prefix ta-lib)/lib"
159
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS

```
source .zshrc
```

Finally, in ipython, pip install ta-lib.

```
1  pip install ta-lib
```

```
Collecting ta-lib
  Using cached TA-Lib-0.4.26.tar.gz (272 kB)
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Installing backend dependencies ... done
  Preparing metadata (pyproject.toml) ... done
```

```
1 import pandas as pd
2 import numpy as np
3 import talib
4 import math
5 import yfinance as yf
6 import plotly.express as px
7 import plotly.graph_objects as go
8 from plotly.subplots import make_subplots
```

```
1 df_nvda = yf.download('NVDA', start='2022-04-01',
2                       end='2023-01-01')
3 df_nvda
```

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

Date	Open	High	Low	Close	Adj Close	Volume
2022-04-01	273.750000	274.959991	262.670013	267.119995	266.870575	51723500
2022-04-04	267.279999	275.579987	266.130005	273.600006	273.344543	39712000

STOCHASTIC RSI

Stochastic RSI (StochRSI) is an indicator used in technical analysis that ranges between(是技術分析中使用的指標，範圍介於) zero and one (or zero and 100 on some charting platforms) and is created by applying the Stochastic oscillator formula to a set of relative strength index (RSI) values rather than to standard price data.

StochRSI reading above 0.8 is considered overbought, while a reading below 0.2 is considered oversold. On the zero to 100 scale, above 80 is overbought, and below 20 is oversold.

Lowest RSI = Lowest RSI reading over last 14 periods (or chosen lookback period);

Highest RSI = Highest RSI reading over last 14 period (or lookback period).

STOCHASTIC RSI

The Formulas For the Stochastic RSI (StochRSI) are:

$$\text{StochRSI} = \frac{RSI - \min[RSI]}{\max[RSI] - \min[RSI]}$$

where:

$RSI = \text{Current RSI reading}$

$\min[RSI] = \text{Lowest RSI reading over the last 14 periods}$
$(\text{or your chosen lookback interval})$

$\max[RSI] = \text{Highest RSI reading over the last 14 periods}$
$(\text{or your chosen lookback interval})$

STOCHASTIC & RSI

Using talib.STOCHRSI to prepare k and d line. Function setting as follow:

k, d = talib.STOCH(df_nvda['High'], df_nvda['Low'], df_nvda['Close'],

fastk_period, slowk_period, slowk_matype, slowd_period, slowd_matype)

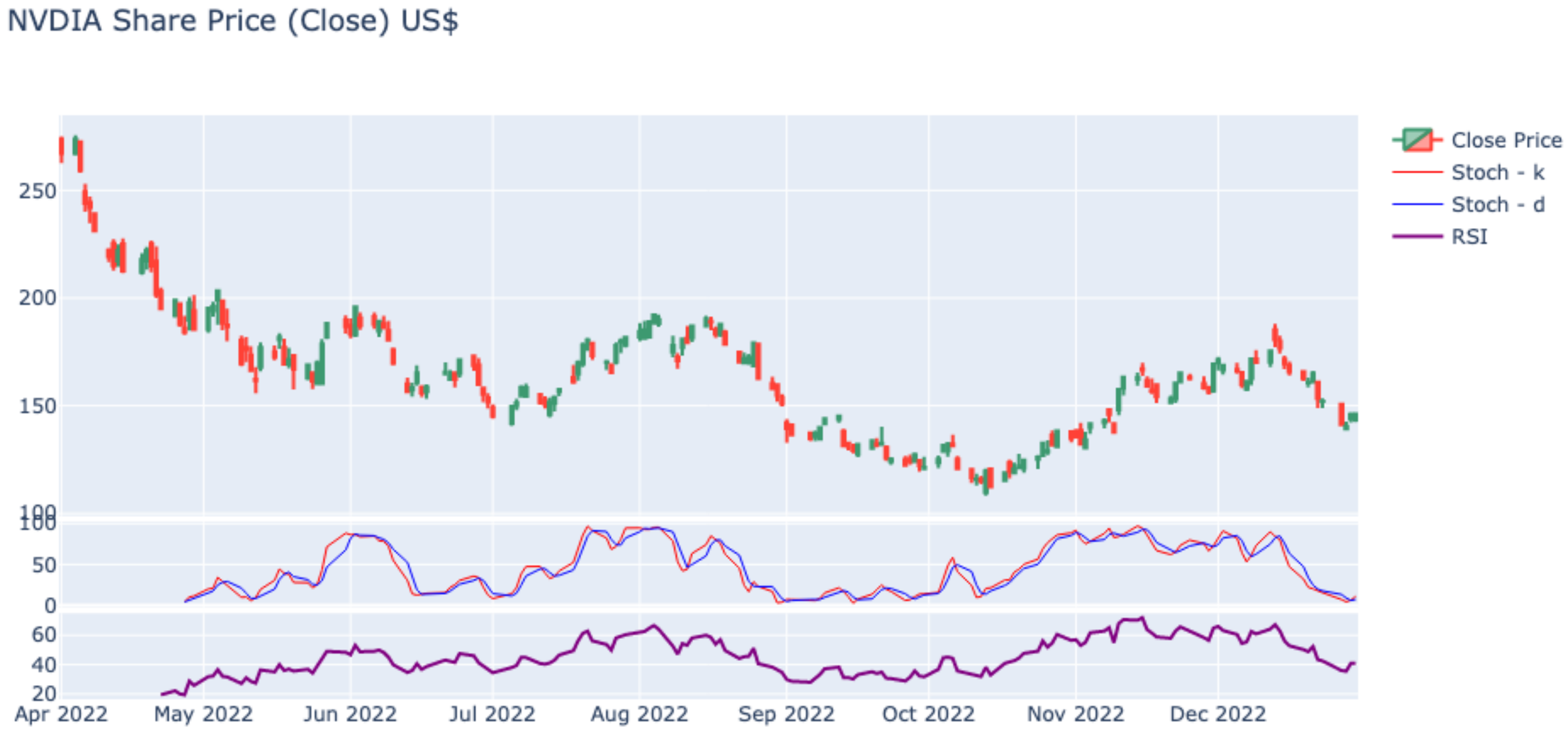
K line is the fast line.

```
rsi = talib.RSI(df_nvda['Close'], timeperiod=14)
k, d = talib.STOCH(df_nvda['High'], df_nvda['Low'], df_nvda['Close'],14)
```

STOCHASTIC & RSI

```
fig = make_subplots(rows=3, cols=1, shared_xaxes=True,
                    vertical_spacing=0.01, row_heights=[0.7, 0.15, 0.15])
fig.add_trace(go.Candlestick(x=df_nvda.index,
              open=df_nvda['Open'], high=df_nvda['High'],
              low=df_nvda['Low'], close=df_nvda['Close'],
              showlegend=True, name='Close Price'),row=1,col=1 )
fig.add_trace(go.Scatter(x=df_nvda.index,
                         y=k, name='Stoch - k',
                         line=dict(color='red', width=1)
                        ), row=2, col=1)
fig.add_trace(go.Scatter(x=df_nvda.index,
                         y=d, name='Stoch - d',
                         line=dict(color='blue', width=1)
                        ), row=2, col=1)
fig.add_trace(go.Scatter(x=df_nvda.index, y=rsi, name='RSI',
                    marker=dict(color ='purple' )), row=3, col=1)
fig.update_layout(title="NVDIA Share Price (Close) US$",
                  xaxis_rangeslider_visible=False)
fig.show()
```

STOCHASTIC & RSI



Bollinger Bands

Bollinger Bands are envelopes plotted at a standard deviation level above and below a simple moving average of the price. Because the distance of the bands is based on standard deviation, they adjust to volatility swings in the underlying price. Bollinger Bands use 2 parameters, Period and Standard Deviations.

Typical values used:

Short term: 10 day moving average, bands at 1.5 standard deviations. (1.5 times the standard dev. +/- the SMA)

Medium term: 20 day moving average, bands at 2 standard deviations.

Long term: 50 day moving average, bands at 2.5 standard deviations.

Bollinger Bands

```
1  from talib import MA_Type
2
3  upper, middle, lower = talib.BBANDS(df_nvda['Close'], matype=MA_Type.T3)
```

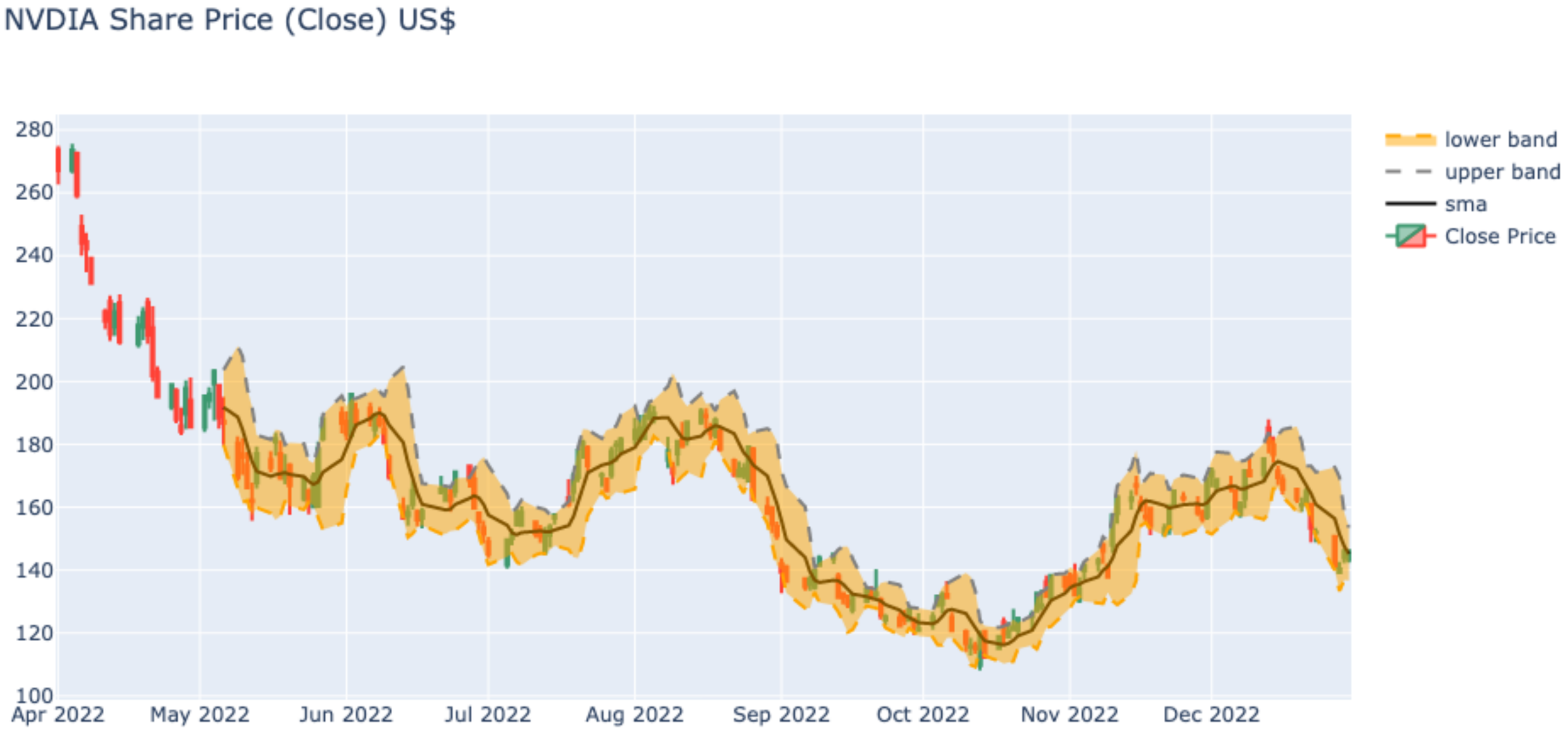
Note (for your ref):

MA_Type.T3 = talib.MA(closed,timeperiod=10, matype=3)

Bollinger Bands

```
fig=go.Figure(data=[go.Candlestick(x=df_nvda.index,
            open=df_nvda['Open'], high=df_nvda['High'],
            low=df_nvda['Low'], close=df_nvda['Close'],
            showlegend=True, name='Close Price')] )

# Moving Average
fig.add_trace(go.Scatter(x = df_nvda.index, y = middle,
            line_color = 'black', name = 'sma'))

# Upper Bound
fig.add_trace(go.Scatter(x = df_nvda.index, y = upper,
            line_color = 'gray', line = {'dash': 'dash'},
            name = 'upper band', opacity = 0.5))

# Lower Bound fill in between with parameter 'fill': 'tonexty'
fig.add_trace(go.Scatter(x = df_nvda.index, y = lower,
            line_color = 'orange', line = {'dash': 'dash'},
            fill = 'tonexty', name = 'lower band', opacity = 0.4))

fig.update_layout(title="NVDIA Share Price (Close) US$",
                  xaxis_rangeslider_visible=False)
```

Bollinger Bands



MACD

Moving average convergence divergence (MACD) is one of the most popular technical indicators in trading. The MACD is appreciated by traders worldwide for its simplicity and flexibility, as it can be used as a trend or momentum indicator and signal opportunities to enter and exit positions.

MACD

There are three different elements involved with the histogram, which is mapped out around a baseline:

- The MACD line (subtracting a long-term exponential moving average (EMA) from a shorter-term EMA)
- The signal line (subtracting the two EMAs and creating a 9-day moving average)
- The histogram (subtracting the MACD line from the signal line)

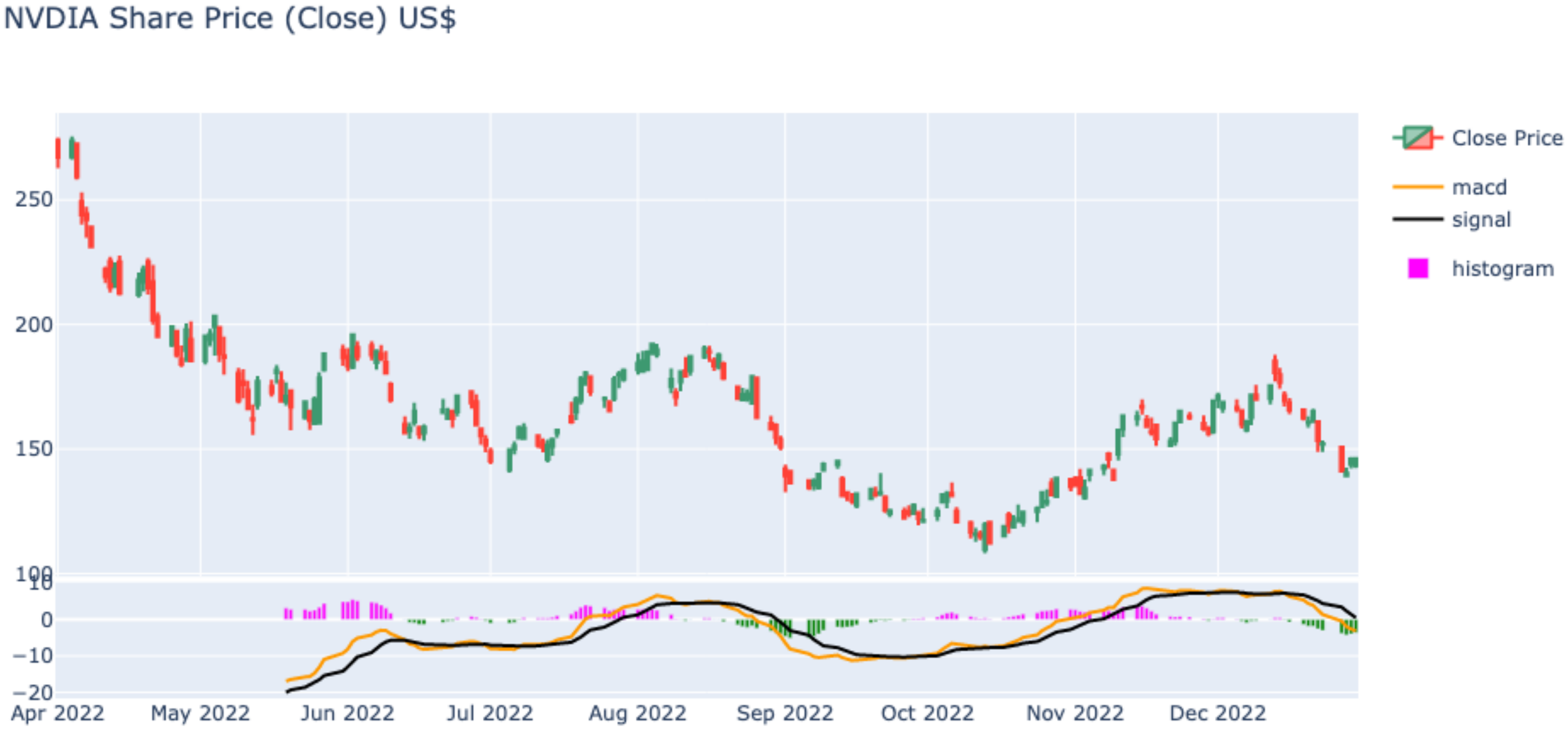
TA lib setting

```
Macd, macdsignal, macdhist = talib.MACD(close, fastperiod=12, slowperiod=26, signalperiod=9)
```

MACD

```
# Colorize the histogram values
colors = np.where(macdhist < 0, '#008000', '#FF00FF')

fig = make_subplots(rows=2, cols=1, shared_xaxes=True,
                    vertical_spacing=0.01, row_heights=[0.8, 0.2])
fig.add_trace(go.Candlestick(x=df_nvda.index,
            open=df_nvda['Open'], high=df_nvda['High'],
            low=df_nvda['Low'], close=df_nvda['Close'],
            showlegend=True, name='Close Price'),row=1,col=1 )
# MACD
fig.append_trace(
    go.Scatter(x=df_nvda.index, y=macd,
        line=dict(color='#ff9900', width=2), name='macd',
        legendgroup='2',), row=2, col=1)

# MACD Signal
fig.append_trace(
    go.Scatter(x=df_nvda.index, y=macdsignal,
        line=dict(color='#000000', width=2),
        legendgroup='2', name='signal'), row=2, col=1)

# histogram
fig.append_trace(
    go.Bar(x=df_nvda.index, y=macdhist, name='histogram',
        marker_color=colors,), row=2, col=1)

fig.update_layout(title="NVDIA Share Price (Close) US$",
                  xaxis_rangeslider_visible=False)
fig.show()
```

MACD


NVDIA Share Price (Close) US$
Close Price
macd
signal
histogram
250
200
150
100
10
0
−10
−20
Apr 2022
May 2022
Jun 2022
Jul 2022
Aug 2022
Sep 2022
Oct 2022
Nov 2022
Dec 2022

ADOSC

Accumulation/Distribution Oscillator (ADOSC) is a volume indicator. The ADOSC is a cumulative indicator that uses volume and price to assess whether a stock is being accumulated or distributed.

The Accumulation/Distribution Indicator (A/D) Formula

$$\text{MFM} = \frac{(\text{Close} - \text{Low}) - (\text{High} - \text{Close})}{\text{High} - \text{Low}}$$

where:

$\text{MFM} = \text{Money Flow Multiplier}$

$\text{Close} = \text{Closing price}$

$\text{Low} = \text{Low price for the period}$

$\text{High} = \text{High price for the period}$

$$\text{Money Flow Volume} = \text{MFM} \times \text{Period Volume}$$

$$\text{A/D} = \text{Previous A/D} + \text{CMFV}$$

where:

$\text{CMFV} = \text{Current period money flow volume}$

ADOSC

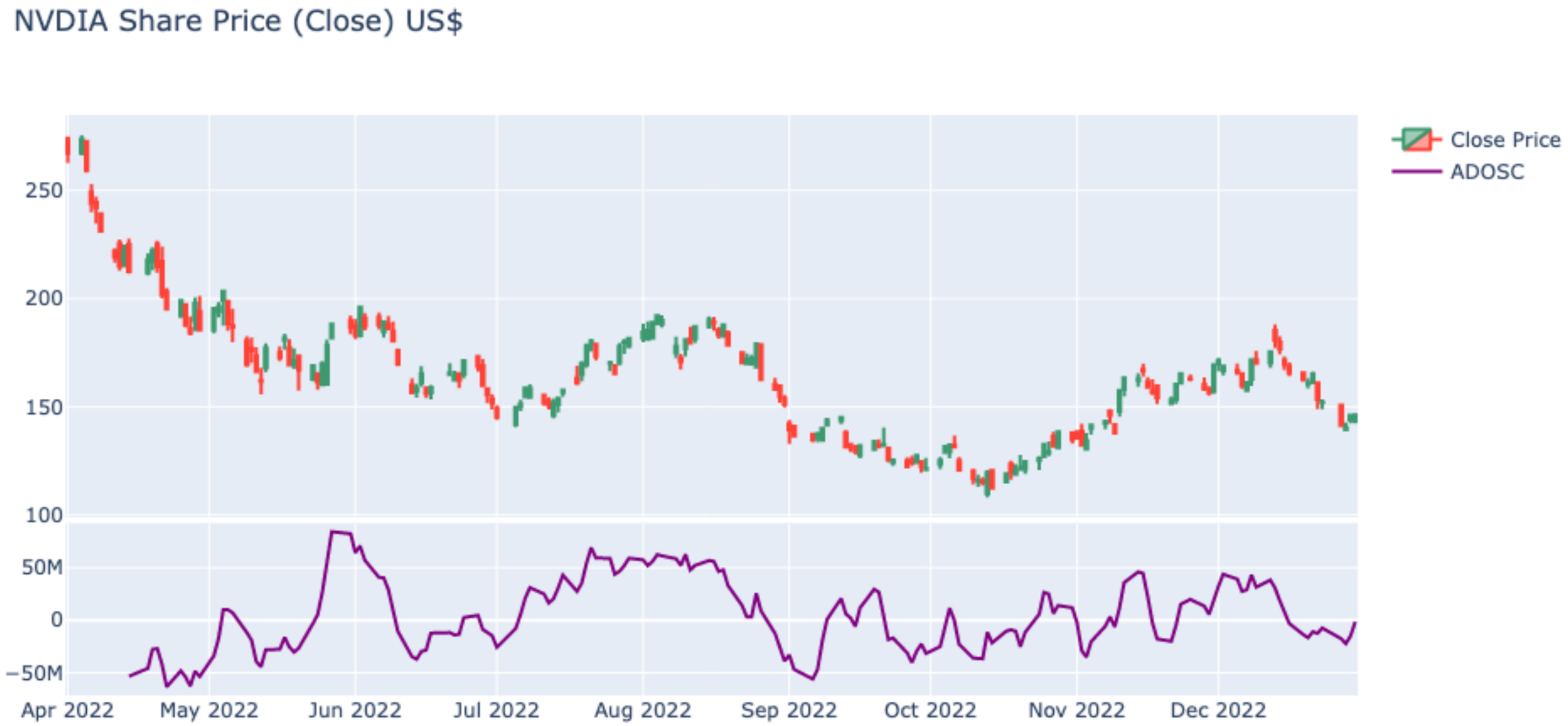
TA Lib setting

real = talib.ADOSC(high, low, close, volume, fastperiod=3, slowperiod=10)

```
adosc = talib.ADOSC(df_nvda['High'], df_nvda['Low'],df_nvda['Close'],
                    df_nvda['Volume'],fastperiod=3, slowperiod=10)
```

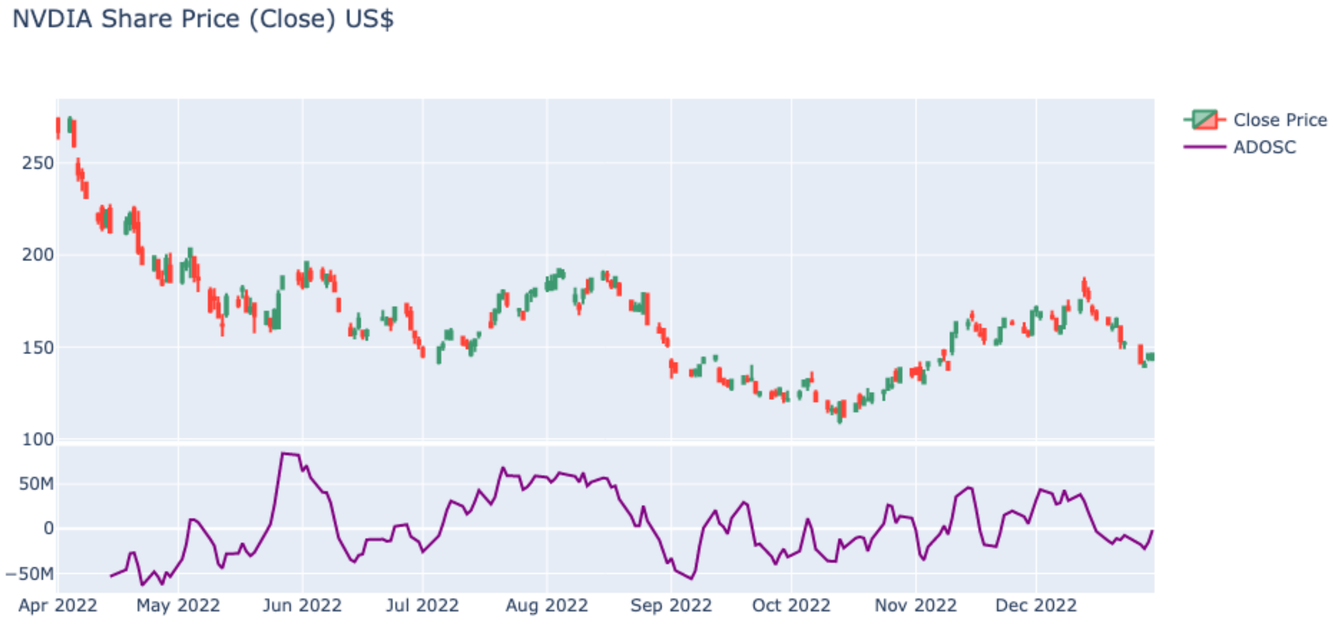
```
fig = make_subplots(rows=2, cols=1, shared_xaxes=True,
                    vertical_spacing=0.01, row_heights=[0.7, 0.3])
fig.add_trace(go.Candlestick(x=df_nvda.index,
              open=df_nvda['Open'], high=df_nvda['High'],
              low=df_nvda['Low'], close=df_nvda['Close'],
              showlegend=True, name='Close Price'),row=1,col=1 )
fig.add_trace(go.Scatter(x=df_nvda.index, y=adosc, name='ADOSC',
                    marker=dict(color ='purple' )), row=2, col=1)
fig.update_layout(title="NVDIA Share Price (Close) US$",
                  xaxis_rangeslider_visible=False)
fig.show()
```

ADOSC


NVDIA Share Price (Close) US$
Close Price
ADOSC
250
200
150
100
50M
0
−50M
Apr 2022
May 2022
Jun 2022
Jul 2022
Aug 2022
Sep 2022
Oct 2022
Nov 2022
Dec 2022

ADOSC

- The accumulation/distribution (ADOSC) indicates supply and demand of an asset or security by looking at where the price closed within the period's range and then multiplying that by volume.
- The ADOSC indicator is cumulative, meaning one period's value is added or subtracted from the last.
- In general, a rising ADOSC helps confirm a rising price trend, while a falling ADOSC helps confirm a price downtrend.



ADOSC – Further Consideration

You may check its holder of of the stock on morningstars or yfinance. And calculate the approximately market shares and the non-institute share portion. That makes hint for you to estimate the minimum and maximum ADOSC.

Major Holders

Breakdown

4.18%	% of Shares Held by All Insider
68.08%	% of Shares Held by Institutions
71.06%	% of Float Held by Institutions
4,015	Number of Institutions Holding Shares

https://finance.yahoo.com/quote/NVDA/holders?p=NVDA

ATR

The average true range (ATR) is a technical analysis indicator introduced by market technician J. Welles Wilder Jr. in his book *New Concepts in Technical Trading Systems* that measures market volatility by decomposing the entire range of an asset price for that period.

ATR & TR

- ATR is typically derived from the 14-day simple moving average of a series of true range indicators.
- The ATR was initially developed for use in commodities markets but has since been applied to all types of securities.
- ATR shows investors the average range prices swing for an investment over a specified period.
- True Range is the daily based swing range.

ATR

TA lib setting

```
ATR = talib.ATR(high, low, close, timeperiod=14)
TR = talib.TRANGE(high, low, close)
```

The Average True Range (ATR) Formula

The formula to calculate ATR for an investment with a previous ATR calculation is :

$$\frac{\text{Previous ATR}(n-1) + \text{TR}}{n}$$

where:

$n = \text{Number of periods}$

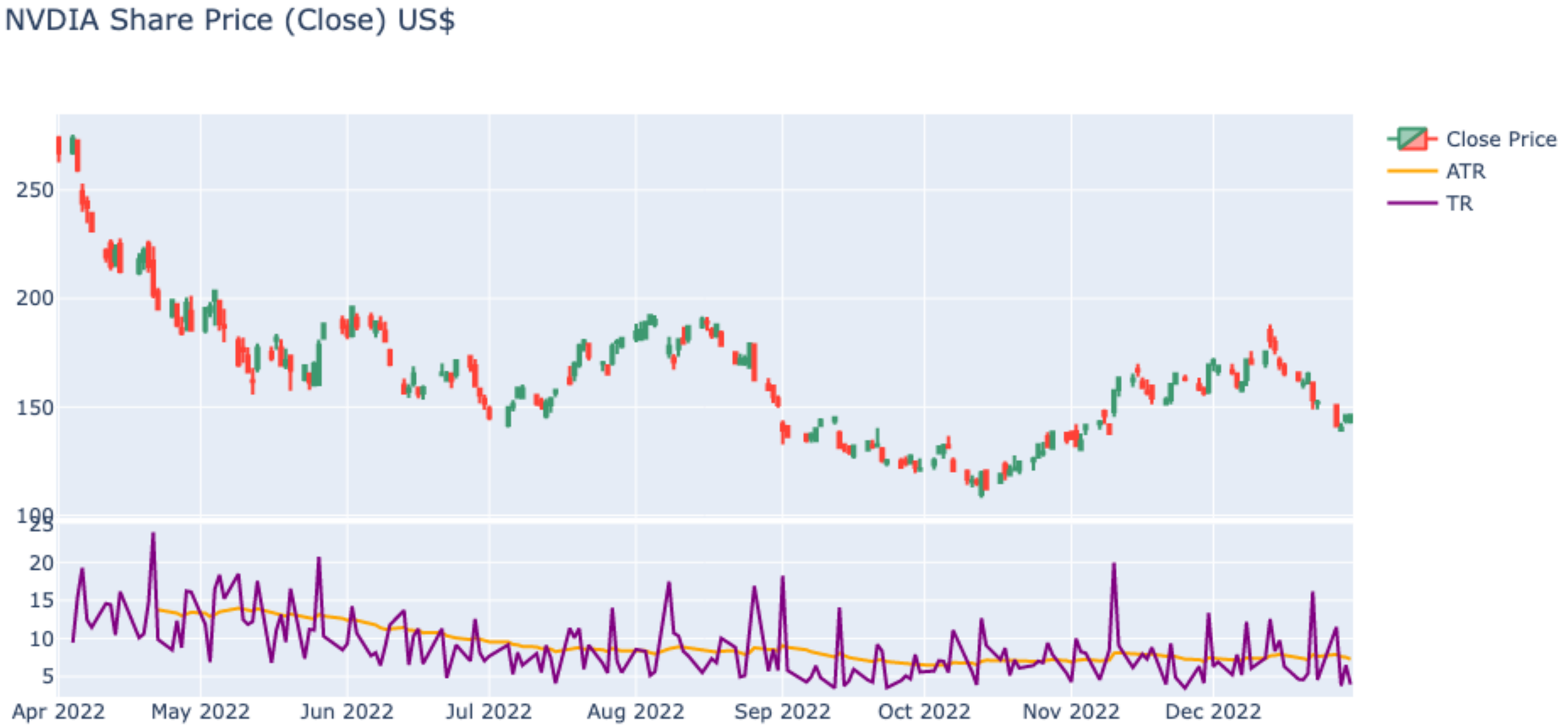
$\text{TR} = \text{True range}$

ATR & TR

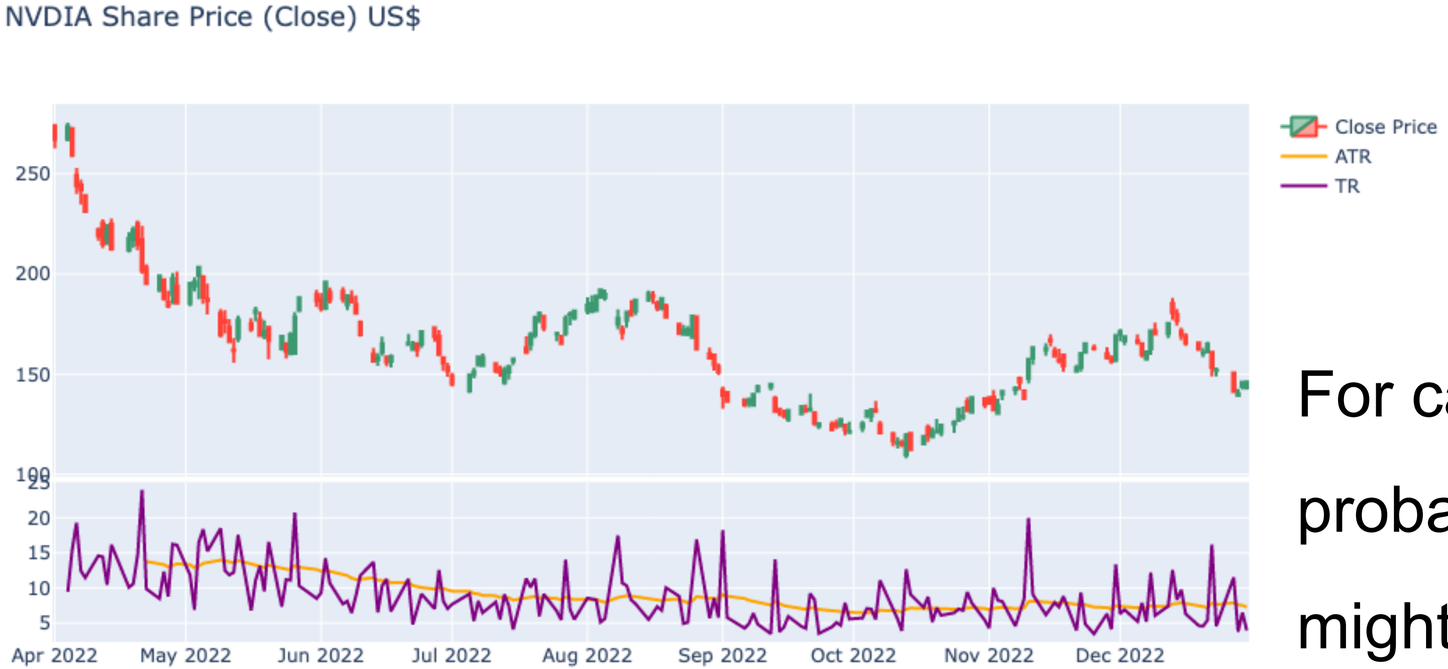
```
atr = talib.ATR(df_nvda['High'], df_nvda['Low'],df_nvda['Close'], timeperiod=14)
tr = talib.TRANGE(df_nvda['High'], df_nvda['Low'],df_nvda['Close'])
```

```
fig = make_subplots(rows=2, cols=1, shared_xaxes=True,
                    vertical_spacing=0.01, row_heights=[0.7, 0.3])
fig.add_trace(go.Candlestick(x=df_nvda.index,
              open=df_nvda['Open'], high=df_nvda['High'],
              low=df_nvda['Low'], close=df_nvda['Close'],
              showlegend=True, name='Close Price'),row=1,col=1 )
fig.add_trace(go.Scatter(x=df_nvda.index, y=atr, name='ATR',
                    marker=dict(color ='orange' )), row=2, col=1)
fig.add_trace(go.Scatter(x=df_nvda.index, y=tr, name='TR',
                    marker=dict(color ='purple' )), row=2, col=1)
fig.update_layout(title="NVDIA Share Price (Close) US$",
                  xaxis_rangeslider_visible=False)
fig.show()
```

ATR & TR


NVDIA Share Price (Close) US$
Close Price
ATR
TR
250
200
150
100
25
20
15
10
5
Apr 2022
May 2022
Jun 2022
Jul 2022
Aug 2022
Sep 2022
Oct 2022
Nov 2022
Dec 2022

ATR & TR



For case TR greater than ATR, that probably means the current trend might change soon.

If you are an option trader, you should pay more attention to the asset volatility.

Chapter Wrapping

TA-Lib is a C backend library support Python and other main stream core language. We only demonstrated 6 out of 130+ TA with plotly graphs. Statistical TA data analysis will be discussed in later chapters.

Academic Resources

TA Lib - https://github.com/TA-Lib/ta-lib-python

TA Lib Documentation - https://ta-lib.github.io/ta-lib-python/doc_index.html

Pandas TA - https://github.com/twopirllc/pandas-ta

QuantConnect - https://www.quantconnect.com/

Google Colab - https://colab.research.google.com/

Google Scholar - https://scholar.google.com/

Research Gate - https://www.researchgate.net/

JSTOR - https://www.jstor.org/

Research SPJ - https://spj.science.org/