UNIVERSIDAD DE EL SALVADOR

FACULTAD MULTIDISCIPLINARIA DE OCCIDENTE

DEPARTAMENTO DE MATEMATICAS

INDICES EN RSTUDIO

CARRERA:

LICENCIATURA EN ESTADÍSTICA.

ASIGNATURA:

ANÁLISIS DE SISTEMAS

DOCENTE:

LICDO.JAIME ISAAC PEÑA

PRESENTADO POR:

JEIMY MERARY ROMERO RS19036

FECHA:

04 DE JUNIO DE 2024

Tabla de contenidos

Operaciones de Apple

Listado de Figuras

Listado de Tablas

lang: es-ES

Introducción

En el análisis financiero y la investigación de mercados, disponer de datos precisos y actualizados es esencial para tomar decisiones informadas. Yahoo Finance es una de las fuentes más populares y accesibles de información financiera, proporcionando datos sobre acciones, índices, bonos, divisas, materias primas y mucho más. Integrar estos datos en RStudio, una potente herramienta de análisis y visualización de datos, amplía enormemente las capacidades analíticas de los usuarios.

Y a continuación se hara un eejemplo de como hacerlo en RStudio con python.

EJEMPLO:

```
!pip install yfinance
```

```
Requirement already satisfied: yfinance in c:\users\dell\appdata\local\programs\python\python312\lib\si
Requirement already satisfied: pandas>=1.3.0 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: numpy>=1.16.5 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: requests>=2.31 in c:\users\dell\appdata\local\programs\python\python312\l
Requirement already satisfied: multitasking>=0.0.7 in c:\users\dell\appdata\local\programs\python\pytho
Requirement already satisfied: lxml>=4.9.1 in c:\users\dell\appdata\local\programs\python\python312\lib
Requirement already satisfied: platformdirs>=2.0.0 in c:\users\dell\appdata\local\programs\python\pytho
Requirement already satisfied: pytz>=2022.5 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: frozendict>=2.3.4 in c:\users\dell\appdata\local\programs\python\python3
Requirement already satisfied: peewee>=3.16.2 in c:\users\dell\appdata\local\programs\python\python312\l
Requirement already satisfied: beautifulsoup4>=4.11.1 in c:\users\dell\appdata\local\programs\python\py
Requirement already satisfied: html5lib>=1.1 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: soupsieve>1.2 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: six>=1.9 in c:\users\dell\appdata\local\programs\python\python312\lib\si
Requirement already satisfied: webencodings in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: python-dateutil>=2.8.2 in c:\users\dell\appdata\local\programs\python\py
Requirement already satisfied: tzdata>=2022.7 in c:\users\dell\appdata\local\programs\python\python312\l
Requirement already satisfied: charset-normalizer<4,>=2 in c:\users\dell\appdata\local\programs\python\p
Requirement already satisfied: idna<4,>=2.5 in c:\users\dell\appdata\local\programs\python\python312\li
Requirement already satisfied: urllib3<3,>=1.21.1 in c:\users\dell\appdata\local\programs\python\python3
```

```
Requirement already satisfied: certifi>=2017.4.17 in c:\users\dell\appdata\local\programs\python\python3
```

```
import yfinance as yf
import pandas as pd
```

se descarga la llama la base a usar

```
apple = yf.Ticker("AAPL")
```

```
apple_info=apple.info
apple_info
```


```
{'address1': 'One Apple Park Way',
 'city': 'Cupertino',
 'state': 'CA',
 'zip': '95014',
 'country': 'United States',
 'phone': '408 996 1010',
 'website': 'https://www.apple.com',
 'industry': 'Consumer Electronics',
 'industryKey': 'consumer-electronics',
 'industryDisp': 'Consumer Electronics',
 'sector': 'Technology',
 'sectorKey': 'technology',
 'sectorDisp': 'Technology',
 'longBusinessSummary': 'Apple Inc. designs, manufactures, and markets smartphones, personal computers,
 'fullTimeEmployees': 150000,
 'companyOfficers': [{'maxAge': 1,
   'name': 'Mr. Timothy D. Cook',
   'age': 62,
   'title': 'CEO & Director',
   'yearBorn': 1961,
   'fiscalYear': 2023,
   'totalPay': 16239562,
   'exercisedValue': 0,
```

```
 'unexercisedValue': 0},
{'maxAge': 1,
 'name': 'Mr. Luca  Maestri',
 'age': 60,
 'title': 'CFO & Senior VP',
 'yearBorn': 1963,
 'fiscalYear': 2023,
 'totalPay': 4612242,
 'exercisedValue': 0,
 'unexercisedValue': 0},
{'maxAge': 1,
 'name': 'Mr. Jeffrey E. Williams',
 'age': 59,
 'title': 'Chief Operating Officer',
 'yearBorn': 1964,
 'fiscalYear': 2023,
 'totalPay': 4637585,
 'exercisedValue': 0,
 'unexercisedValue': 0},
{'maxAge': 1,
 'name': 'Ms. Katherine L. Adams',
 'age': 59,
 'title': 'Senior VP, General Counsel & Secretary',
 'yearBorn': 1964,
 'fiscalYear': 2023,
 'totalPay': 4618064,
 'exercisedValue': 0,
 'unexercisedValue': 0},
{'maxAge': 1,
 'name': "Ms. Deirdre  O'Brien",
 'age': 56,
 'title': 'Senior Vice President of Retail',
 'yearBorn': 1967,
 'fiscalYear': 2023,
 'totalPay': 4613369,
```

```
   'exercisedValue': 0,
   'unexercisedValue': 0},
  {'maxAge': 1,
   'name': 'Mr. Chris  Kondo',
   'title': 'Senior Director of Corporate Accounting',
   'fiscalYear': 2023,
   'exercisedValue': 0,
   'unexercisedValue': 0},
  {'maxAge': 1,
   'name': 'Mr. James  Wilson',
   'title': 'Chief Technology Officer',
   'fiscalYear': 2023,
   'exercisedValue': 0,
   'unexercisedValue': 0},
  {'maxAge': 1,
   'name': 'Suhasini  Chandramouli',
   'title': 'Director of Investor Relations',
   'fiscalYear': 2023,
   'exercisedValue': 0,
   'unexercisedValue': 0},
  {'maxAge': 1,
   'name': 'Mr. Greg  Joswiak',
   'title': 'Senior Vice President of Worldwide Marketing',
   'fiscalYear': 2023,
   'exercisedValue': 0,
   'unexercisedValue': 0},
  {'maxAge': 1,
   'name': 'Mr. Adrian  Perica',
   'age': 49,
   'title': 'Head of Corporate Development',
   'yearBorn': 1974,
   'fiscalYear': 2023,
   'exercisedValue': 0,
   'unexercisedValue': 0}],
 'auditRisk': 6,
```

```
'boardRisk': 1,
'compensationRisk': 2,
'shareHolderRightsRisk': 1,
'overallRisk': 1,
'governanceEpochDate': 1717200000,
'compensationAsOfEpochDate': 1703980800,
'irWebsite': 'http://investor.apple.com/',
'maxAge': 86400,
'priceHint': 2,
'previousClose': 194.03,
'open': 194.72,
'dayLow': 193.0342,
'dayHigh': 195.32,
'regularMarketPreviousClose': 194.03,
'regularMarketOpen': 194.72,
'regularMarketDayLow': 193.0342,
'regularMarketDayHigh': 195.32,
'dividendRate': 1.0,
'dividendYield': 0.0050999997,
'exDividendDate': 1715299200,
'payoutRatio': 0.14930001,
'fiveYearAvgDividendYield': 0.71,
'beta': 1.264,
'trailingPE': 30.225508,
'forwardPE': 26.918283,
'volume': 45319734,
'regularMarketVolume': 45319734,
'averageVolume': 61275561,
'averageVolume10days': 49016610,
'averageDailyVolume10Day': 49016610,
'bid': 194.29,
'ask': 194.34,
'bidSize': 100,
'askSize': 100,
'marketCap': 2980182491136,
```

'fiftyTwoWeekLow': 164.08,
'fiftyTwoWeekHigh': 199.62,
'priceToSalesTrailing12Months': 7.8092318,
'fiftyDayAverage': 178.0664,
'twoHundredDayAverage': 181.4276,
'trailingAnnualDividendRate': 0.96,
'trailingAnnualDividendYield': 0.0049476884,
'currency': 'USD',
'enterpriseValue': 3017618751488,
'profitMargins': 0.26306,
'floatShares': 15308320742,
'sharesOutstanding': 15334099968,
'sharesShort': 99287450,
'sharesShortPriorMonth': 101912593,
'sharesShortPreviousMonthDate': 1713139200,
'dateShortInterest': 1715731200,
'sharesPercentSharesOut': 0.0064999997,
'heldPercentInsiders': 0.052199997,
'heldPercentInstitutions': 0.57666,
'shortRatio': 1.53,
'shortPercentOfFloat': 0.0064999997,
'impliedSharesOutstanding': 15663899648,
'bookValue': 4.837,
'priceToBook': 40.179867,
'lastFiscalYearEnd': 1696032000,
'nextFiscalYearEnd': 1727654400,
'mostRecentQuarter': 1711756800,
'earningsQuarterlyGrowth': -0.022,
'netIncomeToCommon': 100389003264,
'trailingEps': 6.43,
'forwardEps': 7.22,
'pegRatio': 2.8,
'lastSplitFactor': '4:1',
'lastSplitDate': 1598832000,
'enterpriseToRevenue': 7.907,

```
'enterpriseToEbitda': 23.279,
'52WeekChange': 0.092959166,
'SandP52WeekChange': 0.23990989,
'lastDividendValue': 0.25,
'lastDividendDate': 1715299200,
'exchange': 'NMS',
'quoteType': 'EQUITY',
'symbol': 'AAPL',
'underlyingSymbol': 'AAPL',
'shortName': 'Apple Inc.',
'longName': 'Apple Inc.',
'firstTradeDateEpochUtc': 345479400,
'timeZoneFullName': 'America/New_York',
'timeZoneShortName': 'EDT',
'uuid': '8b10e4ae-9eeb-3684-921a-9ab27e4d87aa',
'messageBoardId': 'finmb_24937',
'gmtOffSetMilliseconds': -14400000,
'currentPrice': 194.35,
'targetHighPrice': 275.0,
'targetLowPrice': 164.0,
'targetMeanPrice': 204.31,
'targetMedianPrice': 200.0,
'recommendationMean': 2.1,
'recommendationKey': 'buy',
'numberOfAnalystOpinions': 39,
'totalCash': 67150000128,
'totalCashPerShare': 4.379,
'ebitda': 129629003776,
'totalDebt': 104590000128,
'quickRatio': 0.875,
'currentRatio': 1.037,
'totalRevenue': 381623009280,
'debtToEquity': 140.968,
'revenuePerShare': 24.537,
'returnOnAssets': 0.22073999,
```

```
 'returnOnEquity': 1.4725,
 'freeCashflow': 84726874112,
 'operatingCashflow': 110563000320,
 'earningsGrowth': 0.007,
 'revenueGrowth': -0.043,
 'grossMargins': 0.45586,
 'ebitdaMargins': 0.33968,
 'operatingMargins': 0.30743,
 'financialCurrency': 'USD',
 'trailingPegRatio': 2.0542}
```

```
apple_info['country']
```

```
'United States'
```

```
historico_apple = apple.history(start='2010-01-01', end='2024-05-20')
print(historico_apple)
```

```
                                 Open        High         Low       Close  \
Date
2010-01-04 00:00:00-05:00    6.444462    6.476770    6.412758    6.461975
2010-01-05 00:00:00-05:00    6.479792    6.509685    6.439029    6.473149
2010-01-06 00:00:00-05:00    6.473147    6.498813    6.363540    6.370183
2010-01-07 00:00:00-05:00    6.393737    6.401286    6.312211    6.358408
2010-01-08 00:00:00-05:00    6.349954    6.401286    6.312514    6.400681
...                               ...         ...         ...         ...
2024-05-13 00:00:00-04:00  185.440002  187.100006  184.619995  186.279999
2024-05-14 00:00:00-04:00  187.509995  188.300003  186.289993  187.429993
2024-05-15 00:00:00-04:00  187.910004  190.649994  187.369995  189.720001
2024-05-16 00:00:00-04:00  190.470001  191.100006  189.660004  189.839996
2024-05-17 00:00:00-04:00  189.509995  190.809998  189.179993  189.869995

                              Volume  Dividends  Stock Splits
Date
2010-01-04 00:00:00-05:00  493729600        0.0           0.0
2010-01-05 00:00:00-05:00  601904800        0.0           0.0
```

```
2010-01-06 00:00:00-05:00  552160000        0.0           0.0
2010-01-07 00:00:00-05:00  477131200        0.0           0.0
2010-01-08 00:00:00-05:00  447610800        0.0           0.0
...                              ...        ...           ...
2024-05-13 00:00:00-04:00   72044800        0.0           0.0
2024-05-14 00:00:00-04:00   52393600        0.0           0.0
2024-05-15 00:00:00-04:00   70400000        0.0           0.0
2024-05-16 00:00:00-04:00   52845200        0.0           0.0
2024-05-17 00:00:00-04:00   41282900        0.0           0.0

[3618 rows x 7 columns]
```

```
historico_apple.head()
```

Tabla 1: tabla de los 10 primeros registros

Date	Open	High	Low	Close	Volume	Dividends	Stock Splits
2010-01-04 00:00:00-05:00	6.444462	6.476770	6.412758	6.461975	493729600	0.0	0.0
2010-01-05 00:00:00-05:00	6.479792	6.509685	6.439029	6.473149	601904800	0.0	0.0
2010-01-06 00:00:00-05:00	6.473147	6.498813	6.363540	6.370183	552160000	0.0	0.0
2010-01-07 00:00:00-05:00	6.393737	6.401286	6.312211	6.358408	477131200	0.0	0.0
2010-01-08 00:00:00-05:00	6.349954	6.401286	6.312514	6.400681	447610800	0.0	0.0

Si se quiere mostrar los diez primero registros de los datos de una sola columna; por ejemplo la columna de precio de la acción al cierre de la jornada

```
print(historico_apple["Close"].head(10))
```

```
Date
2010-01-04 00:00:00-05:00    6.461975
2010-01-05 00:00:00-05:00    6.473149
2010-01-06 00:00:00-05:00    6.370183
2010-01-07 00:00:00-05:00    6.358408
2010-01-08 00:00:00-05:00    6.400681
2010-01-11 00:00:00-05:00    6.344217
```

```
2010-01-12 00:00:00-05:00    6.272052
2010-01-13 00:00:00-05:00    6.360521
2010-01-14 00:00:00-05:00    6.323684
2010-01-15 00:00:00-05:00    6.218003
Name: Close, dtype: float64
```

```
historico_apple.reset_index(inplace=True)
```

Se grafica el precio al cierre por fecha.

```
import matplotlib.pyplot as plt
historico_apple.plot(x="Date", y="Close", color="black")
plt.show()
```

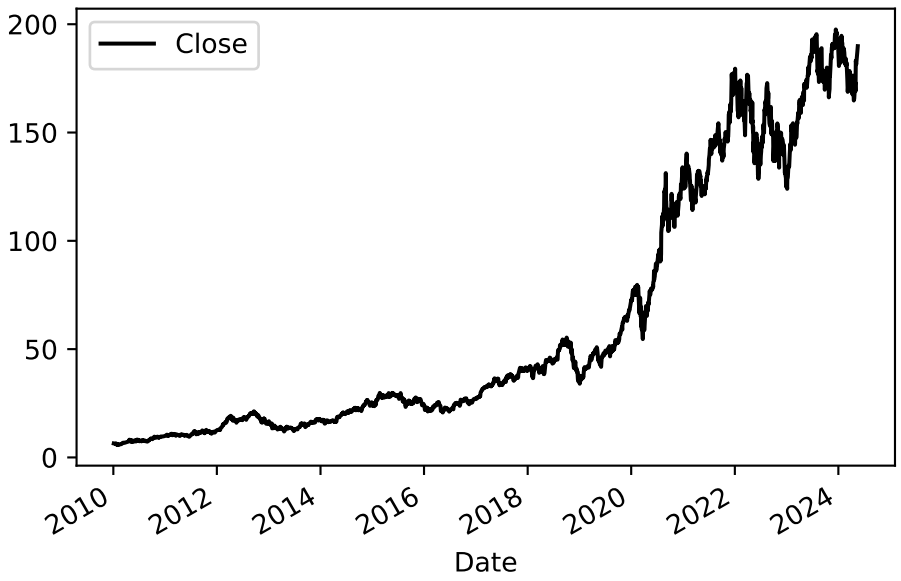


Figura 1: Gráfico de la evolución del precio de cierre de las acciones de Apple

AL observar el grafico se puede decir que muestra la evolución del precio de cierre de las acciones de Apple a lo largo del tiempo donde se observar que el precio ha tendido a aumentar desde el año 2020 en adelante.

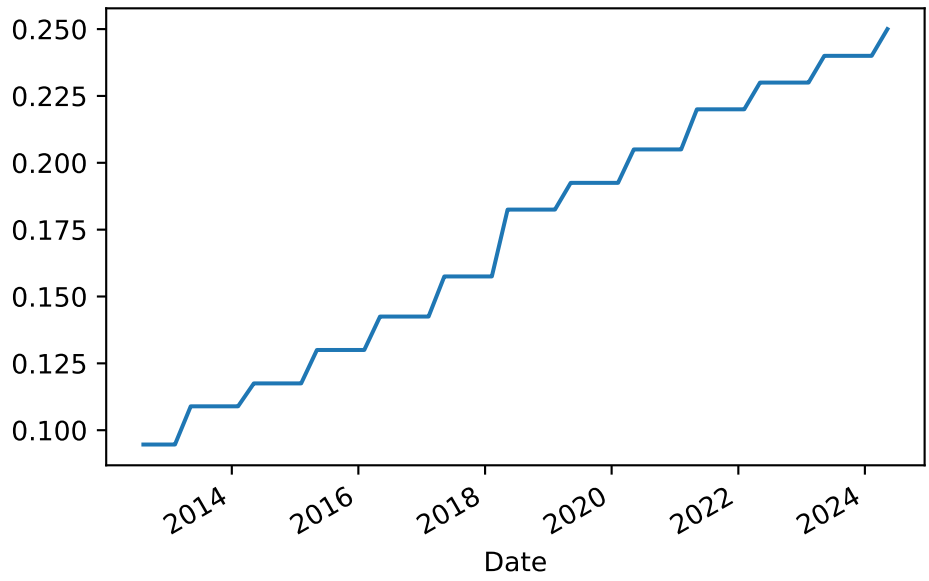
```
apple.dividends
```

```
Date
```

```
2012-08-09 00:00:00-04:00    0.094643
2012-11-07 00:00:00-05:00    0.094643
2013-02-07 00:00:00-05:00    0.094643
2013-05-09 00:00:00-04:00    0.108929
2013-08-08 00:00:00-04:00    0.108929
2013-11-06 00:00:00-05:00    0.108929
2014-02-06 00:00:00-05:00    0.108929
2014-05-08 00:00:00-04:00    0.117500
2014-08-07 00:00:00-04:00    0.117500
2014-11-06 00:00:00-05:00    0.117500
2015-02-05 00:00:00-05:00    0.117500
2015-05-07 00:00:00-04:00    0.130000
2015-08-06 00:00:00-04:00    0.130000
2015-11-05 00:00:00-05:00    0.130000
2016-02-04 00:00:00-05:00    0.130000
2016-05-05 00:00:00-04:00    0.142500
2016-08-04 00:00:00-04:00    0.142500
2016-11-03 00:00:00-04:00    0.142500
2017-02-09 00:00:00-05:00    0.142500
2017-05-11 00:00:00-04:00    0.157500
2017-08-10 00:00:00-04:00    0.157500
2017-11-10 00:00:00-05:00    0.157500
2018-02-09 00:00:00-05:00    0.157500
2018-05-11 00:00:00-04:00    0.182500
2018-08-10 00:00:00-04:00    0.182500
2018-11-08 00:00:00-05:00    0.182500
2019-02-08 00:00:00-05:00    0.182500
2019-05-10 00:00:00-04:00    0.192500
2019-08-09 00:00:00-04:00    0.192500
2019-11-07 00:00:00-05:00    0.192500
2020-02-07 00:00:00-05:00    0.192500
2020-05-08 00:00:00-04:00    0.205000
2020-08-07 00:00:00-04:00    0.205000
2020-11-06 00:00:00-05:00    0.205000
2021-02-05 00:00:00-05:00    0.205000
```

```
2021-05-07 00:00:00-04:00    0.220000
2021-08-06 00:00:00-04:00    0.220000
2021-11-05 00:00:00-04:00    0.220000
2022-02-04 00:00:00-05:00    0.220000
2022-05-06 00:00:00-04:00    0.230000
2022-08-05 00:00:00-04:00    0.230000
2022-11-04 00:00:00-04:00    0.230000
2023-02-10 00:00:00-05:00    0.230000
2023-05-12 00:00:00-04:00    0.240000
2023-08-11 00:00:00-04:00    0.240000
2023-11-10 00:00:00-05:00    0.240000
2024-02-09 00:00:00-05:00    0.240000
2024-05-10 00:00:00-04:00    0.250000
Name: Dividends, dtype: float64
```

```
apple.dividends.plot()
```




```
import matplotlib.pyplot as plt
import matplotlib.pyplot as plt
import pandas as pd


# Calcular el promedio móvil
historico_apple['MA'] = historico_apple['Close'].rolling(window=30).mean()
```

```
# Graficar
plt.figure(figsize=(10, 6))
plt.plot(historico_apple['Date'], historico_apple['Close'], label='Precio de cierre')
plt.plot(historico_apple['Date'], historico_apple['MA'], label='Promedio móvil (30 días)')
plt.xlabel('Fecha')
plt.ylabel('Precio')
plt.title('Precio de cierre de las acciones de Apple con promedio móvil')
plt.legend()
plt.show()
```

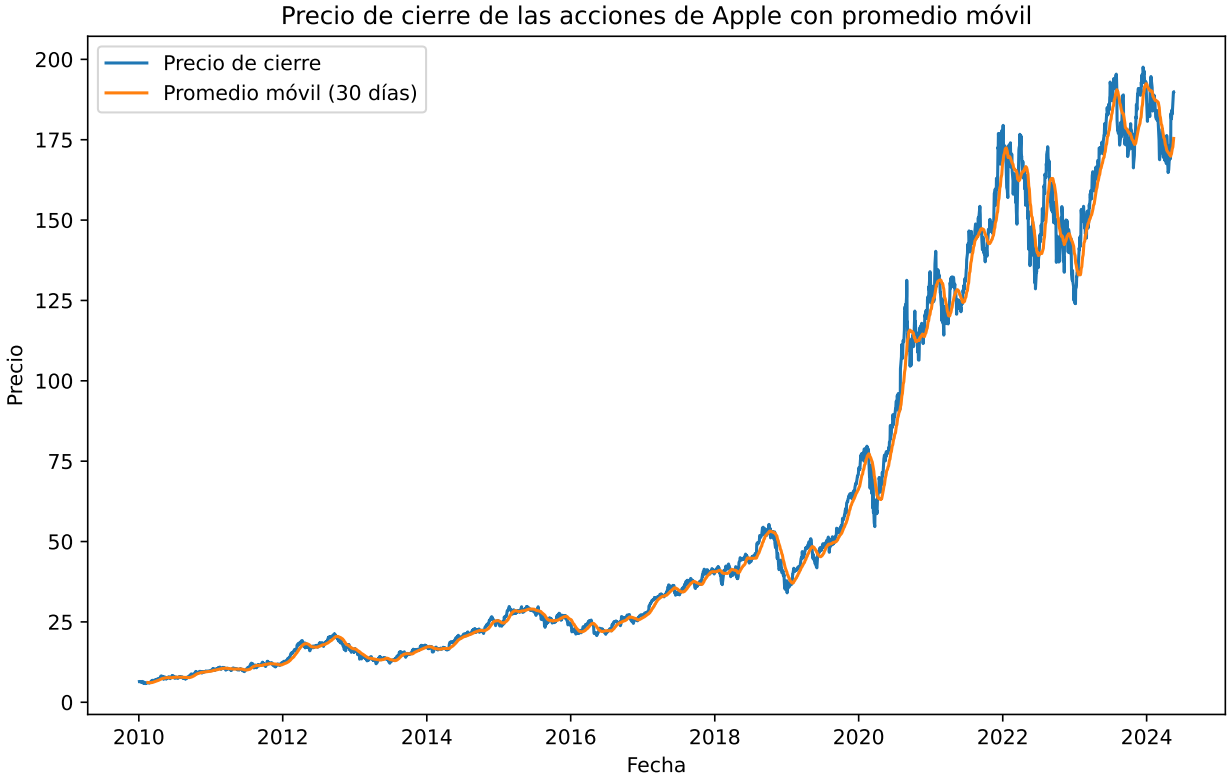


Figura 2: Gráfico lineal del Precio del Cierre de las acciones de Apple

El código calcula el promedio móvil de 30 días del precio de cierre de las acciones de Apple y luego lo grafica junto con el precio de cierre original en función de la fecha.

Se pueden observar cruces entre el precio de cierre y el promedio móvil donde no hay un cambio significativo.

Gráfico de barras de volumen de operaciones:

El volumen de operaciones es la cantidad de acciones que se negocian en un período de tiempo determinado. Un gráfico de barras que muestra el volumen de operaciones junto con el precio de cierre puede proporcionar información sobre el interés de los inversores en un determinado período.

```
import matplotlib.pyplot as plt
historico_apple['Volume'].plot(kind='bar', figsize=(10, 6), color='black')
plt.xlabel('Fecha')
plt.ylabel('Volumen de operaciones')
plt.title('Volumen de operaciones de las acciones de Apple')
plt.show()
```

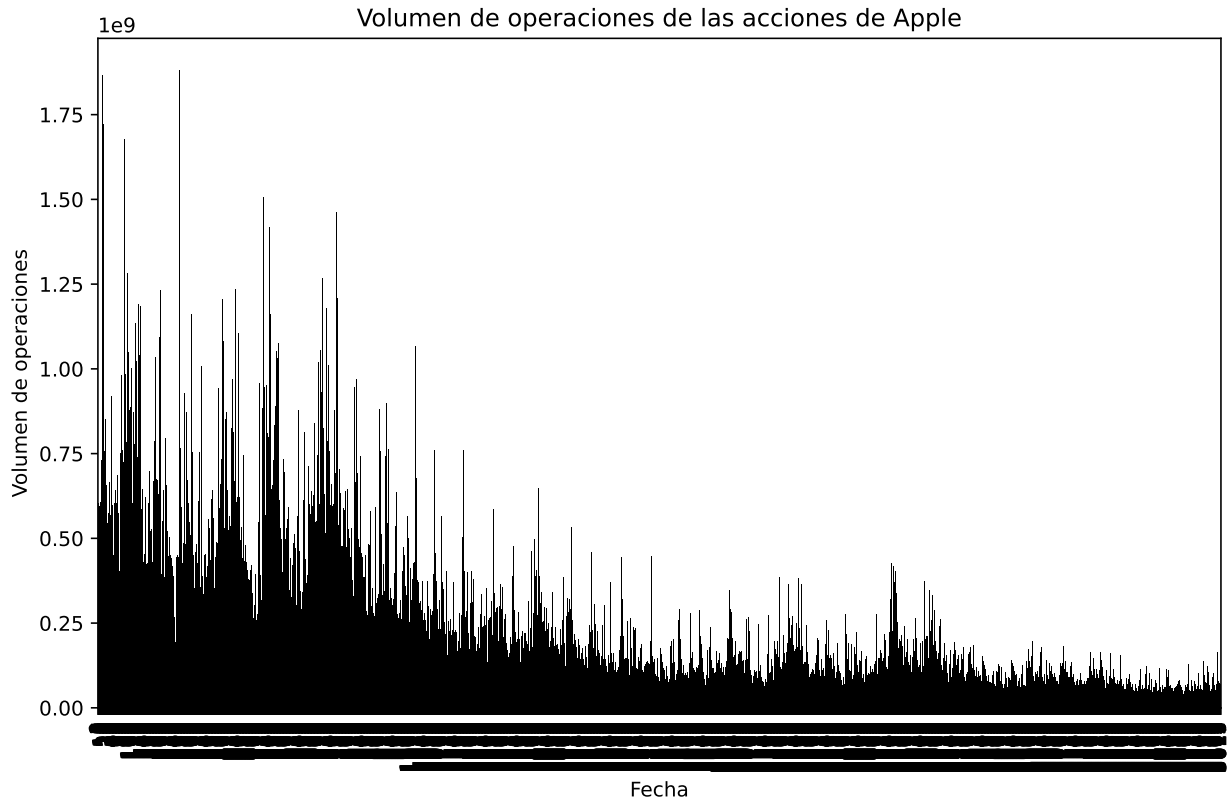


Figura 3: Gráfico de Barras del volumen de las Operaciones en Apple