Assignment_8

March 21, 2022

[1]:
```
import pandas_datareader as pdr
import pandas as pd
import numpy as np
import datetime as dt
from dateutil import relativedelta

NOW = dt.datetime.now()

ticker_list = [str(x) for x in input('Enter Stock Tickers Separated by a Space:␣
 ↪').upper().split()]
set_index = str(input('Enter an index to use: ').upper())
ticker_list.insert(0, set_index)

while True:
    try:
        start_date = dt.datetime.strptime(input('Enter a start date␣
 ↪(YYYY-MM-DD): '), '%Y-%m-%d')
        if start_date >= NOW:
            print('Date cannot be in the future')
            continue
        while True:
            try:
                end_date = dt.datetime.strptime(input('Enter a end date␣
 ↪(YYYY-MM-DD): '), '%Y-%m-%d')
                if end_date <= start_date:
                    print('End Date must be later than Start Date')
                    continue
                elif end_date >= NOW:
                    print('Date cannot be in the future')
                    continue
            except ValueError:
                print('Wrong format')
                continue
            else:
                break
    except ValueError:
        print('Wrong format')
```

```
        continue
    else:
        break


class Capm:
    def __init__(self, ticker, start_date, end_date):
        self.ticker = ticker
        self.start_date = start_date
        self.end_date = end_date
        self.df = None
        self.np_array = []
        self.np_index = []
        print('init called')


    def read_yahoo(self):
        yahoo = pdr.get_data_yahoo(self.ticker, self.start_date, self.end_date,␣
 ↪interval="m")
        self.df = yahoo['Adj Close'].copy()

        print("yahoo called")


    def to_numpy(self):

        for stock in self.ticker:
            close = self.np_array.append(self.df[stock])

        close = np.asarray(self.np_array)
        shift = np.roll(self.np_array, 1, axis=1)

        for x in range(0,len(shift)):
            shift[x,0] = np.nan

        ln_ret = np.log(close/shift)

        self.np_array = ln_ret[:,1:]
        self.np_index = ln_ret[0,1:]

        print('np_array created')


    def beta_alpha(self):
        risk_free = 2.14 #Rate on a 10 year T-bill on Mar 17th

        var = np.var(self.np_index)
```

```
        cov = np.cov(self.np_array)

        time = max(1, relativedelta.relativedelta(end_date, start_date).years)

        index_return = ((self.df.iloc[-1,0] / self.df.iloc[0,0]) ** (1/time) -␣
↪1) * 100

        print(f'Between {self.start_date.strftime("%x")} and {self.end_date.
↪strftime("%x")}:\n--------------')

        for stock in range(1,len(self.ticker)):
            beta = cov[0,stock] / var

            stock_return = self.df.iloc[-1,stock] / self.df.iloc[0,stock]
            stock_return = (stock_return ** (1/time) - 1) * 100

            expected_return = risk_free + (beta * (index_return - risk_free))

            alpha = stock_return - expected_return

            print(f'{self.ticker[stock]} has a beta of {beta:,.4f} and alpha of␣
↪{alpha:,.4f}.')

beta = Capm(ticker_list, start_date, end_date)
beta.read_yahoo()
beta.to_numpy()
```

```
Enter Stock Tickers Separated by a Space: goog ko jnj
Enter an index to use: ^GSPC
Enter a start date (YYYY-MM-DD): 2018-01-01
Enter a end date (YYYY-MM-DD): 2022-01-01
init called
yahoo called
np_array created
```

[2]:
```
display(beta.df.head(), beta.df.tail())
beta.beta_alpha()
```

Symbols	^GSPC	GOOG	KO	JNJ
Date				
2018-01-01	2823.810059	1169.939941	41.475910	123.464935
2018-02-01	2713.830078	1104.729980	37.667351	116.040405
2018-03-01	2640.870117	1031.790039	37.850372	115.227913
2018-04-01	2648.050049	1017.330017	37.991058	113.735291
2018-05-01	2705.270020	1084.989990	37.806423	107.558029

Symbols	^GSPC	GOOG	KO	JNJ
Date				

```
2021-09-01  4307.540039  2665.310059  51.283665  159.425674
2021-10-01  4605.379883  2965.409912  55.511295  160.787949
2021-11-01  4567.000000  2849.040039  51.651012  153.927200
2021-12-01  4766.180176  2893.590088  58.760201  169.978882
2022-01-01  4515.549805  2713.969971  60.546524  171.191086

Between 01/01/18 and 01/01/22:
--------------
GOOG has a beta of 1.0722 and alpha of 10.2158.
KO has a beta of 0.7338 and alpha of 0.2121.
JNJ has a beta of 0.7195 and alpha of -1.0464.
```

[ ]: