Financial Data Science Homework 2

Evan Mc Garry

March 2023

1 Introduction

This report will examine the Goldmansachs stock (GS) through its returns (calculated as "log returns") and its relationship with the market returns (with the chosen proxy for the market being the S&P500 Market Index). The final part of this report will then examine the expected volatility of GS over the life of all upcoming options by extracting the upcoming option data and using this data to plot a 3-D implied volatility surface.

The price information history of the stock and the market will be downloaded using "yfinance" and subsequently the *daily* log returns will be calculated from this information. The code for these processes will be available in Appendix A.1. Additionally, a table of GS log returns will be provided in Appendix B.1 and a table of S&P500 log returns in Appendix B.2.

2 Data Plotting

2.1 Scatter Plot - GS vs. S&P500

First it is necessary to examine the relationship between GS returns and S&P500 returns which will be achieved through a scatter plot (the code for which can be found in Appendix A.2):
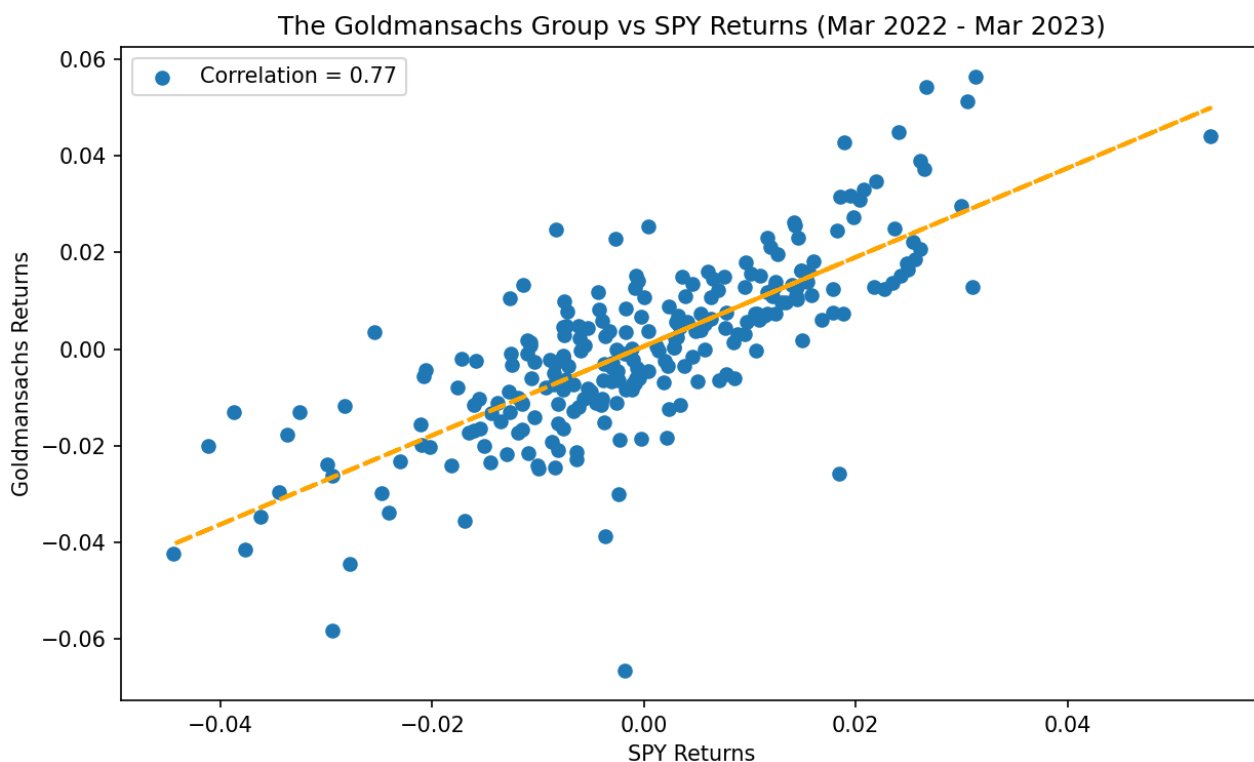

Figure 1: "**Scatter Plot**" which includes the *regression line* and *correlation coefficient.*

As we can see there is quite a high positive correlation coefficient which means that a movement in market returns will likely result in a similar movement in stock returns (*i.e.* a 1% increase in market returns will lead to a 0.77% increase in GS returns). In terms of strategy, the GS stock will not provide great diversification benefits in a portfolio with the S&P500 Market Index. However, if you expect the S&P500 returns to increase then holding GS stock would be beneficial. Conversely, if you expect the S&P500 to take a hit then shorting GS would be a viable strategy.

2.2 Histogram of GS Returns

This report will now produce a histogram of GS stock returns in order to give an idea of the stock returns frequency distribution. The code for producing this graph will be provided in Appendix A.3:
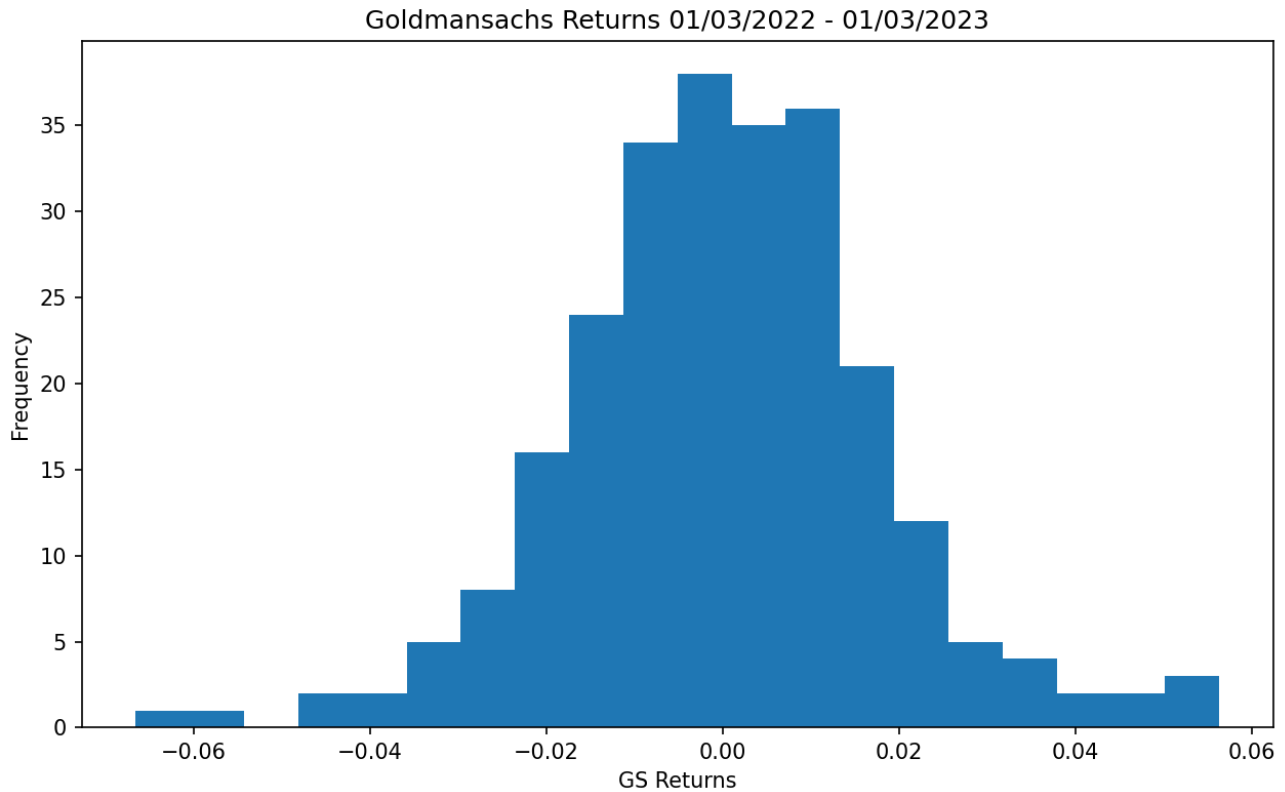

Figure 2: "Histogram" of Goldmansachs stock returns.

This histogram will be very useful if it is necessary to compare GS with another stock. It will allow us to compare the performance of the stocks and to determine which stock has more spread about the mean.

2.3 Box Plots of GS Returns and S&P500 Returns

The box plots of these graphs will offer a lot of information on the statistical distribution of the data at a glance. We can determine the interquartile range, median, maximum and minimum values for GS returns and S&P500 returns very quickly. In addition to this we can see any outliers. The code for producing this plot will be provided in Appendix A.4:
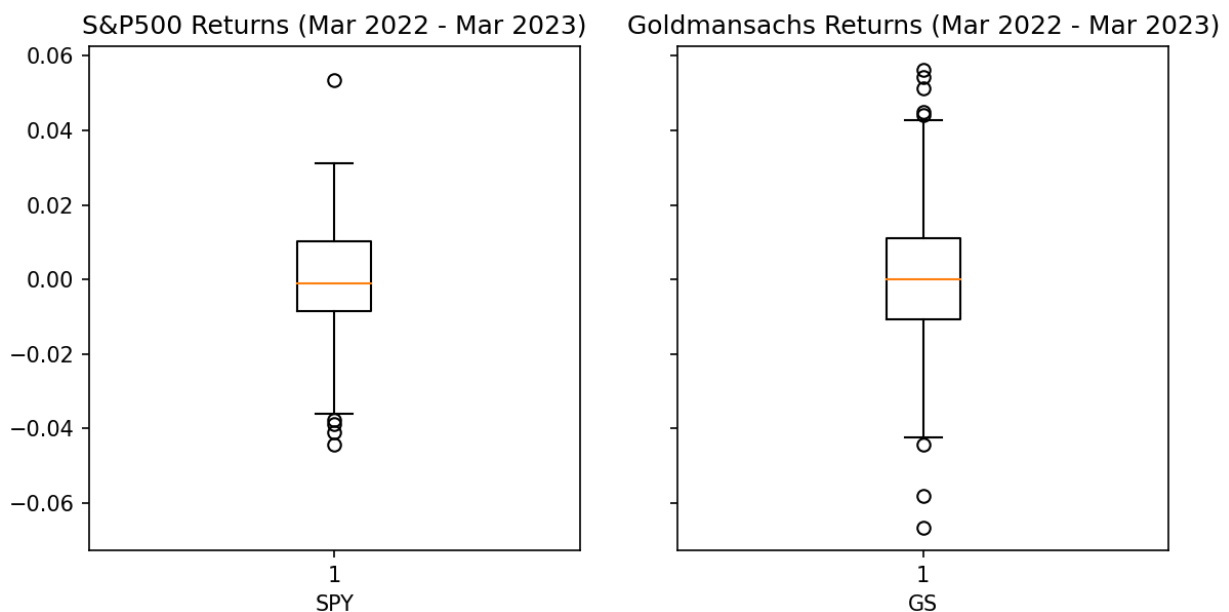

Figure 3: "Histogram" of Goldmansachs stock returns.

It may seem counter-intuitive that there are outliers beyond the maxima and minima of these plots - however this is because the minimum value is calculated as "1.5 times the inter-quartile range" subtracted from the 25th percentile (Q1). Conversely, the maximum value is calculated as "1.5 times the inter-quartile range" added to the 75th percentile (Q3). Anything outside of this range will be counted as an outlier.

These outliers are treated as abnormalities that can distort the final insights. For the S&P500 returns there are many negative outliers (below the minimum) and only 1 positive outlier (above the maximum). In comparison, GS returns have more outliers overall with 3 negative and multiple positive outliers. GS also has a much higher maximum value and a slightly lower minimum value.

2.4 Trading Strategy using Buy / Sell Signals

In order to develop the basis for a trading strategy, it is necessary to plot the stock price with a moving average indicator. While there are many indicators that can be used to generate buy/sell signals we will look at the simple moving average indicator, taking 2 moving averages with one being a short-period average and the other being a long-period average. The code for generating this plot and the signals will be provided in Appendix A.5:
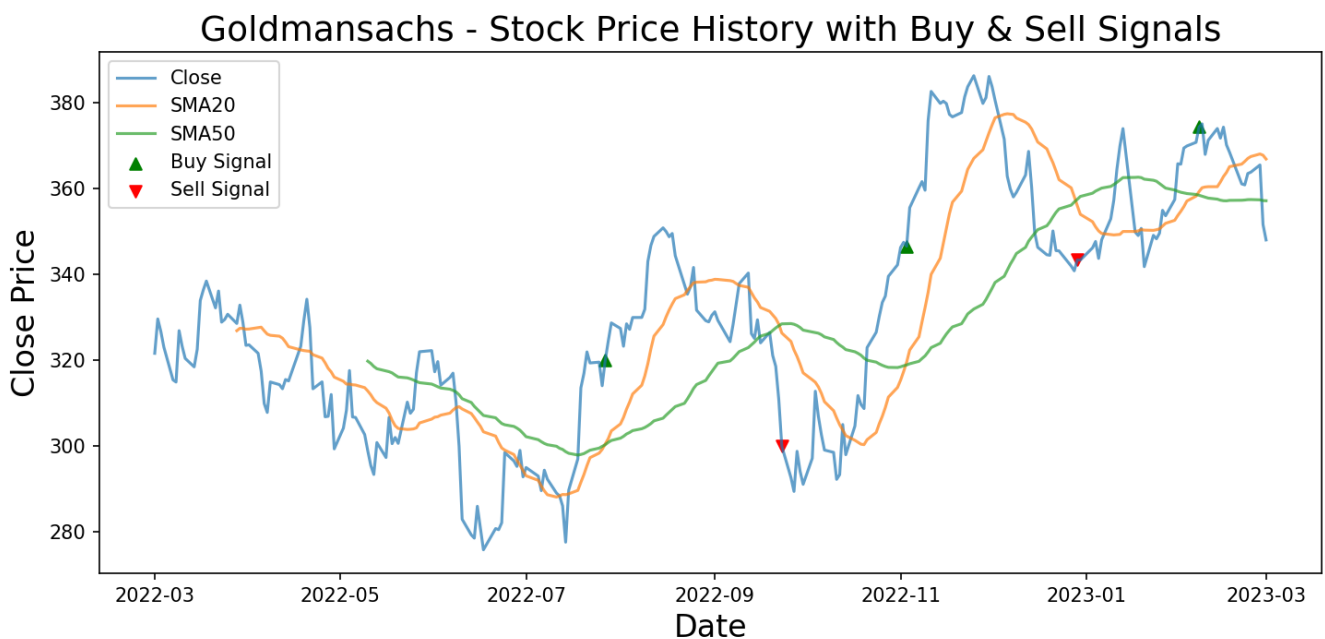
Figure 4: **Plot of Goldmansachs stock returns with Moving Average Indicators and Buy/Sell Signals**

The buy signal is generated where the short-period moving average exceeds the long-period moving average. Looking at the first buy signal it seems this strategy indicates to buy at approx. $320USD however the subsequent sell signal is at approx. $300 which will mean we take a loss as a result of this strategy. Similarly for the next pair there seems to be a slight loss - indicating that this is a poor strategy and perhaps a different moving average indicator would be better suited.

2.5   3-D Implied Volatility Surface - Option Data for GS

Finally this report will analyse the volatility of the GS stock price over the duration of all upcoming options by extracting all upcoming option data. The amount of data extracted is too vast to include in this report but the full data set can be examined through the code, as well as how to generate this 3-D Surface, in Appendix A.6:
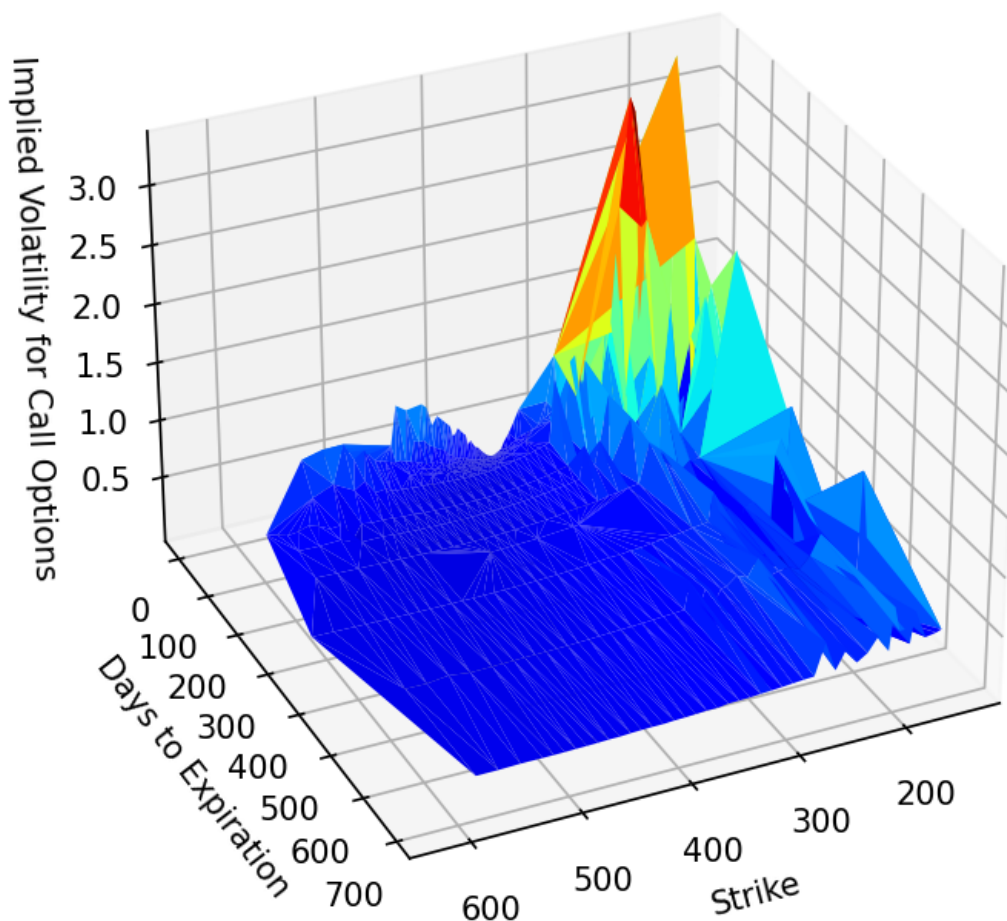

Figure 5: 3-D Implied Volatility Surface of Goldmansachs stock option data

As we can see, the results are to be expected with the highest implied volatility occurring for options close to expiration with the lower strike prices.

A Appendix A - Python Code

A.1 Python Code - Price Information History

This This is the code necessary to successfully download the **price information history** of Goldmansachs:

```python
import numpy as np
import matplotlib.pyplot as plt
import yfinance as yf
import pandas as pd
import matplotlib.dates as dt
import talib
import mplfinance as fplt
from mpl_toolkits import mplot3d
from datetime import datetime
from itertools import chain
from matplotlib import cm
# Q1.2 Yahoo Finance
# Obtaining Price Information History
ticker_name1 = 'GS'
yticker1 = yf.Ticker(ticker_name1)
GS = yticker1.history(period="1y")

ticker_name2 = 'SPY'
yticker2 = yf.Ticker(ticker_name2)
SPY = yticker2.history(period="1y")

# Calculating Log Returns from Price Information History
GS['Return'] = np.log(GS['Close']/GS['Close'].shift(1))
R_GS = GS['Return']
Y = R_GS[1:]

SPY['Return'] = np.log(SPY['Close']/SPY['Close'].shift(1))
R_SPY= SPY['Return']
X = R_SPY[1:]
```

Return to Report

A.2 Python Code - Scatter Plot of GS Returns vs. Market Returns

This is the code necessary to generate the scatter plot of GS Returns vs. Market Returns which includes the regression line and correlation coefficient:

```python
# Q1.3 Scatter Plot - GS v SPY500
fig1, ax1 = plt.subplots()
a, b = np.polyfit(X, Y, 1)
plt.scatter(X, Y, label=f'Correlation = {np.round(np.corrcoef(X,Y)[0,1], 2)}')
plt.plot(X, a*X + b, color='orange', linestyle='--', linewidth=2)
plt.rcParams.update({'figure.figsize':(10,8), 'figure.dpi':150})
plt.title('The Goldmansachs Group vs SPY Returns (Mar 2022 - Mar 2023)')
plt.xlabel('SPY Returns')
plt.ylabel('Goldmansachs Returns')
plt.legend(loc='upper left')
plt.show()
```

Return to Report

A.3 Python Code - Histogram of GS Returns

This is the code necessary to generate the histogram of GS Returns:

```python
# Q1.4 Histogram - GS Returns
fig2, ax2 = plt.subplots()
ax2.hist(Y, bins=20)
ax2.set(xlabel = "GS Returns")
```

```
ax2.set(ylabel = "Frequency")
plt.title('Goldmansachs Returns 01/03/2022 - 01/03/2023')
plt.show()
```

Return to Report

A.4   Python Code - Box Plots of GS Returns and Market Returns

This is the code necessary to generate the box plots of GS Returns and S&P500 Returns and compare their statistical distributions:

```
# Q1.5 Box Plots
fig3, ax3 = plt.subplots(1, 2, sharey=True)
ax3[0].boxplot(X)
ax3[1].boxplot(Y)
ax3[0].set(xlabel="SPY")
ax3[1].set(xlabel="GS")
ax3[0].title.set_text("S&P500 Returns (Mar 2022 - Mar 2023)")
ax3[1].title.set_text("Goldmansachs Returns (Mar 2022 - Mar 2023)")
plt.show()
```

Return to Report

A.5   Python Code - Stock Price plot with Moving Average Indicator and Signals

This is the code necessary to produce the plot of GS stock price which includes a long-period simple moving average, short-period moving average and buy/sell signals based on these 2 indicators:

```
# Q1.6 Generating Buy/Sell Signals - Moving Average Indicator
# Stock Price Data
df = pd.DataFrame(GS)

# Store Simple Moving Average (SMA) Data
df['SMA20'] = talib.SMA(GS.Close, 20)
df['SMA50'] = talib.SMA(GS.Close, 50)

# Obtain Buy and Sell Signals
df['Signal'] = np.where(df['SMA20'] > df['SMA50'], 1, 0)
df['Position'] = df['Signal'].diff()
df['Buy'] = np.where(df['Position'] == 1, df['Close'], np.NAN)
df['Sell'] = np.where(df['Position'] == -1, df['Close'], np.NAN)

# Plot Stock Price, SMA and Signals
df = pd.DataFrame(GS)
plt.figure(figsize=(16,8))
plt.title('Goldmansachs - Stock Price History with Buy & Sell Signals', fontsize=18)
plt.plot(df['Close'], alpha = 0.7, label='Close')
plt.plot(df['SMA20'], alpha = 0.7, label='SMA20')
plt.plot(df['SMA50'], alpha = 0.7, label='SMA50')
plt.scatter(df.index, df['Buy'], alpha = 1, label = 'Buy Signal', marker = '^', color = 'green')
plt.scatter(df.index, df['Sell'], alpha = 1, label = 'Sell Signal', marker = 'v', color = 'red')
plt.xlabel('Date', fontsize = 16)
plt.ylabel('Close Price', fontsize = 16)
plt.legend()
plt.show()
```

Return to Report

A.6 Python Code - 3-D Implied Volatility Surface of Upcoming GS Option Data

This is the code necessary to generate the 3-D Implied Volatility Surface using all of the upcoming option data for GS:

```python
# Q1.7 3D Implied Volatility Surface
# Extraction of Option Data
sTicker = "GS"
stock = yf.Ticker(sTicker)

# Store the Maturities of Options
lMaturity = list(stock.options)
today = datetime.now().date()

lDTE = [] # DTE = Days to Expiration
lData_CALL = []

for maturity in lMaturity:
    maturity_date = datetime.strptime(maturity, '%Y-%m-%d').date()
    lDTE.append((maturity_date - today).days)
    # Store Data for Calls
    lData_CALL.append(stock.option_chain(maturity).calls)

# View all Option Data
print(lData_CALL)

# Store Data in Lists
# Empty Lists
lStrike = []
lDTE_ext = []
lImpVol = []

for i in range(0,len(lData_CALL)):
    lStrike.append(lData_CALL[i]["strike"])
    lDTE_ext.append(np.repeat(lDTE[i], len(lData_CALL[i])))
    lImpVol.append(lData_CALL[i]["impliedVolatility"])

# Delist 'List of Lists'
lStrike = list(chain(*lStrike))
lDTE_ext = list(chain(*lDTE_ext))
lImpVol = list(chain(*lImpVol))

# Prepare Plot Title
S = GS['Close'].values[-1]
T = today

# Plot 3-D Imp Vol Surface
fig5 = plt.figure(figsize=(15,15))
axs = fig5.add_subplot(111, projection='3d')
axs.plot_trisurf(lStrike, lDTE_ext, lImpVol, cmap=cm.jet)
axs.view_init(30, 65)
axs.set_xlabel('Strike')
axs.set_ylabel('Days to Expiration')
axs.set_zlabel('Implied Volatility for Call Options')
axs.set_title("Implied Volatility Surface for "+str(sTicker)+"\n Current Price: $"+str(round(S, ndigits
# plt.title('Implied Volatility Surface for %s - Current Price: %s - Date: %s' %
#           (sTicker, '{0:.4g}'.format(stock['Close'].values[-1]), today))
plt.show()
```

Return to Report

B   Appendix B - Tabulated Price Information History (with Python Code for conversion)

B.1   Price Information History - GS Returns

This is the code necessary to generate the table of GS Returns:

```python
# Converting to LaTex table
LR_GS = Y.to_frame()
print(LR_GS.style.to_latex())
```

Which produces the following output:

Date	Return
2022-03-02 00:00:00-05:00	0.024618
2022-03-03 00:00:00-05:00	-0.008839
2022-03-04 00:00:00-05:00	-0.011311
2022-03-07 00:00:00-05:00	-0.023882
2022-03-08 00:00:00-05:00	-0.001617
2022-03-09 00:00:00-05:00	0.037320
2022-03-10 00:00:00-05:00	-0.011123
2022-03-11 00:00:00-05:00	-0.008829
2022-03-14 00:00:00-04:00	-0.006197
2022-03-15 00:00:00-04:00	0.012780
2022-03-16 00:00:00-04:00	0.034634
2022-03-17 00:00:00-04:00	0.007339
2022-03-18 00:00:00-04:00	0.006128
2022-03-21 00:00:00-04:00	-0.018645
2022-03-22 00:00:00-04:00	0.011760
2022-03-23 00:00:00-04:00	-0.021810
2022-03-24 00:00:00-04:00	0.001846
2022-03-25 00:00:00-04:00	0.003740
2022-03-28 00:00:00-04:00	-0.006510
2022-03-29 00:00:00-04:00	0.012920
2022-03-30 00:00:00-04:00	-0.012055
2022-03-31 00:00:00-04:00	-0.016495
2022-04-01 00:00:00-04:00	0.000363
2022-04-04 00:00:00-04:00	-0.006105
2022-04-05 00:00:00-04:00	-0.013033
2022-04-06 00:00:00-04:00	-0.024055
2022-04-07 00:00:00-04:00	-0.006694
2022-04-08 00:00:00-04:00	0.022785
2022-04-11 00:00:00-04:00	-0.001962
2022-04-12 00:00:00-04:00	-0.003060
2022-04-13 00:00:00-04:00	0.006825
2022-04-14 00:00:00-04:00	-0.001025
2022-04-18 00:00:00-04:00	0.025296
2022-04-19 00:00:00-04:00	0.018233
2022-04-20 00:00:00-04:00	0.015096
2022-04-21 00:00:00-04:00	-0.020020
2022-04-22 00:00:00-04:00	-0.044437
2022-04-25 00:00:00-04:00	0.005147
2022-04-26 00:00:00-04:00	-0.026162
2022-04-27 00:00:00-04:00	0.000351
2022-04-28 00:00:00-04:00	0.016402
2022-04-29 00:00:00-04:00	-0.041423
2022-05-02 00:00:00-04:00	0.016009
2022-05-03 00:00:00-04:00	0.013566
2022-05-04 00:00:00-04:00	0.029590
2022-05-05 00:00:00-04:00	-0.034656
2022-05-06 00:00:00-04:00	-0.000351

2022-05-09 00:00:00-04:00	-0.013090
2022-05-10 00:00:00-04:00	-0.012477
2022-05-11 00:00:00-04:00	-0.011573
2022-05-12 00:00:00-04:00	-0.007122
2022-05-13 00:00:00-04:00	0.025001
2022-05-16 00:00:00-04:00	-0.011631
2022-05-17 00:00:00-04:00	0.030924
2022-05-18 00:00:00-04:00	-0.020139
2022-05-19 00:00:00-04:00	0.004781
2022-05-20 00:00:00-04:00	-0.004553
2022-05-23 00:00:00-04:00	0.031475
2022-05-24 00:00:00-04:00	-0.008437
2022-05-25 00:00:00-04:00	0.003021
2022-05-26 00:00:00-04:00	0.027346
2022-05-27 00:00:00-04:00	0.015179
2022-05-31 00:00:00-04:00	0.000826
2022-06-01 00:00:00-04:00	-0.015416
2022-06-02 00:00:00-04:00	0.007429
2022-06-03 00:00:00-04:00	-0.017327
2022-06-06 00:00:00-04:00	0.005726
2022-06-07 00:00:00-04:00	0.003115
2022-06-08 00:00:00-04:00	-0.021472
2022-06-09 00:00:00-04:00	-0.033805
2022-06-10 00:00:00-04:00	-0.058199
2022-06-13 00:00:00-04:00	-0.012975
2022-06-14 00:00:00-04:00	-0.002757
2022-06-15 00:00:00-04:00	0.026302
2022-06-16 00:00:00-04:00	-0.017703
2022-06-17 00:00:00-04:00	-0.018380
2022-06-21 00:00:00-04:00	0.017783
2022-06-22 00:00:00-04:00	-0.000948
2022-06-23 00:00:00-04:00	0.005712
2022-06-24 00:00:00-04:00	0.056321
2022-06-27 00:00:00-04:00	-0.006528
2022-06-28 00:00:00-04:00	-0.004298
2022-06-29 00:00:00-04:00	0.012575
2022-06-30 00:00:00-04:00	-0.020857
2022-07-01 00:00:00-04:00	0.007413
2022-07-05 00:00:00-04:00	-0.006807
2022-07-06 00:00:00-04:00	-0.011642
2022-07-07 00:00:00-04:00	0.016342
2022-07-08 00:00:00-04:00	-0.007159
2022-07-11 00:00:00-04:00	-0.011159
2022-07-12 00:00:00-04:00	-0.002219
2022-07-13 00:00:00-04:00	-0.008169
2022-07-14 00:00:00-04:00	-0.029946
2022-07-15 00:00:00-04:00	0.042685
2022-07-18 00:00:00-04:00	0.024836
2022-07-19 00:00:00-04:00	0.054235
2022-07-20 00:00:00-04:00	0.010634
2022-07-21 00:00:00-04:00	0.015710
2022-07-22 00:00:00-04:00	-0.008025
2022-07-25 00:00:00-04:00	0.000586
2022-07-26 00:00:00-04:00	-0.017334
2022-07-27 00:00:00-04:00	0.018629
2022-07-28 00:00:00-04:00	0.013983
2022-07-29 00:00:00-04:00	0.012921
2022-08-01 00:00:00-04:00	-0.003937
2022-08-02 00:00:00-04:00	-0.012728
2022-08-03 00:00:00-04:00	0.016005
2022-08-04 00:00:00-04:00	-0.003910
2022-08-05 00:00:00-04:00	0.008402
2022-08-08 00:00:00-04:00	0.000030

2022-08-09 00:00:00-04:00	0.005780
2022-08-10 00:00:00-04:00	0.032989
2022-08-11 00:00:00-04:00	0.010778
2022-08-12 00:00:00-04:00	0.006067
2022-08-15 00:00:00-04:00	0.005721
2022-08-16 00:00:00-04:00	-0.002476
2022-08-17 00:00:00-04:00	-0.003471
2022-08-18 00:00:00-04:00	0.002203
2022-08-19 00:00:00-04:00	-0.014919
2022-08-22 00:00:00-04:00	-0.019807
2022-08-23 00:00:00-04:00	-0.006563
2022-08-24 00:00:00-04:00	0.005160
2022-08-25 00:00:00-04:00	0.013219
2022-08-26 00:00:00-04:00	-0.029553
2022-08-29 00:00:00-04:00	-0.007399
2022-08-30 00:00:00-04:00	-0.000869
2022-08-31 00:00:00-04:00	0.004640
2022-09-01 00:00:00-04:00	0.002492
2022-09-02 00:00:00-04:00	-0.006075
2022-09-06 00:00:00-04:00	-0.015168
2022-09-07 00:00:00-04:00	0.012389
2022-09-08 00:00:00-04:00	0.014476
2022-09-09 00:00:00-04:00	0.013976
2022-09-12 00:00:00-04:00	0.007265
2022-09-13 00:00:00-04:00	-0.042303
2022-09-14 00:00:00-04:00	-0.003447
2022-09-15 00:00:00-04:00	0.013235
2022-09-16 00:00:00-04:00	-0.016448
2022-09-19 00:00:00-04:00	0.007482
2022-09-20 00:00:00-04:00	-0.016567
2022-09-21 00:00:00-04:00	-0.007920
2022-09-22 00:00:00-04:00	-0.024590
2022-09-23 00:00:00-04:00	-0.035620
2022-09-26 00:00:00-04:00	-0.024641
2022-09-27 00:00:00-04:00	-0.011058
2022-09-28 00:00:00-04:00	0.031784
2022-09-29 00:00:00-04:00	-0.015681
2022-09-30 00:00:00-04:00	-0.010388
2022-10-03 00:00:00-04:00	0.020602
2022-10-04 00:00:00-04:00	0.051215
2022-10-05 00:00:00-04:00	-0.018819
2022-10-06 00:00:00-04:00	-0.014112
2022-10-07 00:00:00-04:00	-0.011853
2022-10-10 00:00:00-04:00	-0.001795
2022-10-11 00:00:00-04:00	-0.021287
2022-10-12 00:00:00-04:00	0.003732
2022-10-13 00:00:00-04:00	0.039050
2022-10-14 00:00:00-04:00	-0.023327
2022-10-17 00:00:00-04:00	0.022154
2022-10-18 00:00:00-04:00	0.023013
2022-10-19 00:00:00-04:00	-0.006682
2022-10-20 00:00:00-04:00	-0.003052
2022-10-21 00:00:00-04:00	0.044951
2022-10-24 00:00:00-04:00	0.011013
2022-10-25 00:00:00-04:00	0.011194
2022-10-26 00:00:00-04:00	0.009849
2022-10-27 00:00:00-04:00	0.004429
2022-10-28 00:00:00-04:00	0.013667
2022-10-31 00:00:00-04:00	0.007839
2022-11-01 00:00:00-04:00	0.011745
2022-11-02 00:00:00-04:00	0.003465
2022-11-03 00:00:00-04:00	-0.002634
2022-11-04 00:00:00-04:00	0.025582

2022-11-07 00:00:00-05:00	0.012908
2022-11-08 00:00:00-05:00	0.004019
2022-11-09 00:00:00-05:00	-0.005565
2022-11-10 00:00:00-05:00	0.044070
2022-11-11 00:00:00-05:00	0.017971
2022-11-14 00:00:00-05:00	-0.007322
2022-11-15 00:00:00-05:00	0.001359
2022-11-16 00:00:00-05:00	-0.001438
2022-11-17 00:00:00-05:00	-0.006692
2022-11-18 00:00:00-05:00	-0.001528
2022-11-21 00:00:00-05:00	0.002660
2022-11-22 00:00:00-05:00	0.009580
2022-11-23 00:00:00-05:00	0.006181
2022-11-25 00:00:00-05:00	0.006735
2022-11-28 00:00:00-05:00	-0.016857
2022-11-29 00:00:00-05:00	0.003525
2022-11-30 00:00:00-05:00	0.012875
2022-12-01 00:00:00-05:00	-0.006078
2022-12-02 00:00:00-05:00	-0.008451
2022-12-05 00:00:00-05:00	-0.024040
2022-12-06 00:00:00-05:00	-0.023502
2022-12-07 00:00:00-05:00	-0.008273
2022-12-08 00:00:00-05:00	-0.005125
2022-12-09 00:00:00-05:00	0.002956
2022-12-12 00:00:00-05:00	0.011186
2022-12-13 00:00:00-05:00	0.015058
2022-12-14 00:00:00-05:00	-0.022797
2022-12-15 00:00:00-05:00	-0.029712
2022-12-16 00:00:00-05:00	-0.009997
2022-12-19 00:00:00-05:00	-0.005037
2022-12-20 00:00:00-05:00	-0.000348
2022-12-21 00:00:00-05:00	0.016325
2022-12-22 00:00:00-05:00	-0.013166
2022-12-23 00:00:00-05:00	-0.000203
2022-12-27 00:00:00-05:00	-0.010299
2022-12-28 00:00:00-05:00	-0.003222
2022-12-29 00:00:00-05:00	0.007482
2022-12-30 00:00:00-05:00	-0.000146
2023-01-03 00:00:00-05:00	0.008237
2023-01-04 00:00:00-05:00	0.004266
2023-01-05 00:00:00-05:00	-0.011396
2023-01-06 00:00:00-05:00	0.012489
2023-01-09 00:00:00-05:00	0.014036
2023-01-10 00:00:00-05:00	0.012276
2023-01-11 00:00:00-05:00	0.019728
2023-01-12 00:00:00-05:00	0.014869
2023-01-13 00:00:00-05:00	0.010915
2023-01-17 00:00:00-05:00	-0.066551
2023-01-18 00:00:00-05:00	-0.002375
2023-01-19 00:00:00-05:00	0.004744
2023-01-20 00:00:00-05:00	-0.025731
2023-01-23 00:00:00-05:00	0.021130
2023-01-24 00:00:00-05:00	-0.002323
2023-01-25 00:00:00-05:00	0.003725
2023-01-26 00:00:00-05:00	0.015158
2023-01-27 00:00:00-05:00	-0.003584
2023-01-30 00:00:00-05:00	0.010574
2023-01-31 00:00:00-05:00	0.023091
2023-02-01 00:00:00-05:00	-0.000273
2023-02-02 00:00:00-05:00	0.010229
2023-02-03 00:00:00-05:00	0.001298
2023-02-06 00:00:00-05:00	0.002295
2023-02-07 00:00:00-05:00	0.009662

2023-02-08 00:00:00-05:00	0.001868
2023-02-09 00:00:00-05:00	-0.019137
2023-02-10 00:00:00-05:00	0.008901
2023-02-13 00:00:00-05:00	0.007353
2023-02-14 00:00:00-05:00	-0.006007
2023-02-15 00:00:00-05:00	0.006862
2023-02-16 00:00:00-05:00	-0.011121
2023-02-17 00:00:00-05:00	-0.004603
2023-02-21 00:00:00-05:00	-0.020230
2023-02-22 00:00:00-05:00	-0.000637
2023-02-23 00:00:00-05:00	0.007316
2023-02-24 00:00:00-05:00	0.000852
2023-02-27 00:00:00-05:00	0.004607
2023-02-28 00:00:00-05:00	-0.038712
2023-03-01 00:00:00-05:00	-0.010261

B.2 Price Information History - S&P500 Returns (with Python Code for conversion)

This is the code necessary to generate the table of S&P500 returns:

```
# Converting to LaTex table
LR_SPY = X.to_frame()
print(LR_SPY.to_latex())
```

Which produces the following output:

Date	Return
2022-03-02 00:00:00-05:00	0.018229
2022-03-03 00:00:00-05:00	-0.004991
2022-03-04 00:00:00-05:00	-0.008158
2022-03-07 00:00:00-05:00	-0.029922
2022-03-08 00:00:00-05:00	-0.007611
2022-03-09 00:00:00-05:00	0.026458
2022-03-10 00:00:00-05:00	-0.004526
2022-03-11 00:00:00-05:00	-0.012796
2022-03-14 00:00:00-04:00	-0.007335
2022-03-15 00:00:00-04:00	0.021752
2022-03-16 00:00:00-04:00	0.021932
2022-03-17 00:00:00-04:00	0.012433
2022-03-18 00:00:00-04:00	0.010893
2022-03-21 00:00:00-04:00	-0.000293
2022-03-22 00:00:00-04:00	0.011633
2022-03-23 00:00:00-04:00	-0.012962
2022-03-24 00:00:00-04:00	0.014962
2022-03-25 00:00:00-04:00	0.004872
2022-03-28 00:00:00-04:00	0.007088
2022-03-29 00:00:00-04:00	0.012295
2022-03-30 00:00:00-04:00	-0.006194
2022-03-31 00:00:00-04:00	-0.015511
2022-04-01 00:00:00-04:00	0.002830
2022-04-04 00:00:00-04:00	0.008530
2022-04-05 00:00:00-04:00	-0.012712
2022-04-06 00:00:00-04:00	-0.010050
2022-04-07 00:00:00-04:00	0.005026
2022-04-08 00:00:00-04:00	-0.002678
2022-04-11 00:00:00-04:00	-0.017240
2022-04-12 00:00:00-04:00	-0.003712
2022-04-13 00:00:00-04:00	0.011388

2022-04-14 00:00:00-04:00	-0.012530
2022-04-18 00:00:00-04:00	0.000411
2022-04-19 00:00:00-04:00	0.016014
2022-04-20 00:00:00-04:00	-0.000742
2022-04-21 00:00:00-04:00	-0.015066
2022-04-22 00:00:00-04:00	-0.027823
2022-04-25 00:00:00-04:00	0.005781
2022-04-26 00:00:00-04:00	-0.029388
2022-04-27 00:00:00-04:00	0.002808
2022-04-28 00:00:00-04:00	0.024946
2022-04-29 00:00:00-04:00	-0.037656
2022-05-02 00:00:00-04:00	0.006001
2022-05-03 00:00:00-04:00	0.004574
2022-05-04 00:00:00-04:00	0.029998
2022-05-05 00:00:00-04:00	-0.036190
2022-05-06 00:00:00-04:00	-0.005987
2022-05-09 00:00:00-04:00	-0.032541
2022-05-10 00:00:00-04:00	0.002308
2022-05-11 00:00:00-04:00	-0.016014
2022-05-12 00:00:00-04:00	-0.001044
2022-05-13 00:00:00-04:00	0.023626
2022-05-16 00:00:00-04:00	-0.004066
2022-05-17 00:00:00-04:00	0.020362
2022-05-18 00:00:00-04:00	-0.041147
2022-05-19 00:00:00-04:00	-0.006143
2022-05-20 00:00:00-04:00	0.000437
2022-05-23 00:00:00-04:00	0.018537
2022-05-24 00:00:00-04:00	-0.007663
2022-05-25 00:00:00-04:00	0.008796
2022-05-26 00:00:00-04:00	0.019784
2022-05-27 00:00:00-04:00	0.024253
2022-05-31 00:00:00-04:00	-0.005627
2022-06-01 00:00:00-04:00	-0.008121
2022-06-02 00:00:00-04:00	0.018864
2022-06-03 00:00:00-04:00	-0.016548
2022-06-06 00:00:00-04:00	0.003040
2022-06-07 00:00:00-04:00	0.009546
2022-06-08 00:00:00-04:00	-0.010932
2022-06-09 00:00:00-04:00	-0.024070
2022-06-10 00:00:00-04:00	-0.029424
2022-06-13 00:00:00-04:00	-0.038708
2022-06-14 00:00:00-04:00	-0.003018
2022-06-15 00:00:00-04:00	0.014156
2022-06-16 00:00:00-04:00	-0.033656
2022-06-17 00:00:00-04:00	0.002153
2022-06-21 00:00:00-04:00	0.024862
2022-06-22 00:00:00-04:00	-0.001815
2022-06-23 00:00:00-04:00	0.009755
2022-06-24 00:00:00-04:00	0.031299
2022-06-27 00:00:00-04:00	-0.003827
2022-06-28 00:00:00-04:00	-0.020645
2022-06-29 00:00:00-04:00	-0.000815
2022-06-30 00:00:00-04:00	-0.008157
2022-07-01 00:00:00-04:00	0.010521
2022-07-05 00:00:00-04:00	0.001887
2022-07-06 00:00:00-04:00	0.003372
2022-07-07 00:00:00-04:00	0.014866
2022-07-08 00:00:00-04:00	-0.000823
2022-07-11 00:00:00-04:00	-0.011489
2022-07-12 00:00:00-04:00	-0.008888
2022-07-13 00:00:00-04:00	-0.005265
2022-07-14 00:00:00-04:00	-0.002431
2022-07-15 00:00:00-04:00	0.018925

2022-07-18 00:00:00-04:00	-0.008291
2022-07-19 00:00:00-04:00	0.026661
2022-07-20 00:00:00-04:00	0.006353
2022-07-21 00:00:00-04:00	0.010132
2022-07-22 00:00:00-04:00	-0.009321
2022-07-25 00:00:00-04:00	0.001214
2022-07-26 00:00:00-04:00	-0.011902
2022-07-27 00:00:00-04:00	0.025635
2022-07-28 00:00:00-04:00	0.012464
2022-07-29 00:00:00-04:00	0.014473
2022-08-01 00:00:00-04:00	-0.002966
2022-08-02 00:00:00-04:00	-0.006619
2022-08-03 00:00:00-04:00	0.015538
2022-08-04 00:00:00-04:00	-0.000676
2022-08-05 00:00:00-04:00	-0.001692
2022-08-08 00:00:00-04:00	-0.001162
2022-08-09 00:00:00-04:00	-0.003979
2022-08-10 00:00:00-04:00	0.020786
2022-08-11 00:00:00-04:00	0.000000
2022-08-12 00:00:00-04:00	0.016787
2022-08-15 00:00:00-04:00	0.004112
2022-08-16 00:00:00-04:00	0.001957
2022-08-17 00:00:00-04:00	-0.007123
2022-08-18 00:00:00-04:00	0.002902
2022-08-19 00:00:00-04:00	-0.013529
2022-08-22 00:00:00-04:00	-0.021042
2022-08-23 00:00:00-04:00	-0.002422
2022-08-24 00:00:00-04:00	0.003196
2022-08-25 00:00:00-04:00	0.014019
2022-08-26 00:00:00-04:00	-0.034435
2022-08-29 00:00:00-04:00	-0.006634
2022-08-30 00:00:00-04:00	-0.011039
2022-08-31 00:00:00-04:00	-0.007638
2022-09-01 00:00:00-04:00	0.003133
2022-09-02 00:00:00-04:00	-0.010600
2022-09-06 00:00:00-04:00	-0.003780
2022-09-07 00:00:00-04:00	0.017805
2022-09-08 00:00:00-04:00	0.006515
2022-09-09 00:00:00-04:00	0.015416
2022-09-12 00:00:00-04:00	0.010690
2022-09-13 00:00:00-04:00	-0.044456
2022-09-14 00:00:00-04:00	0.003809
2022-09-15 00:00:00-04:00	-0.011418
2022-09-16 00:00:00-04:00	-0.007658
2022-09-19 00:00:00-04:00	0.007725
2022-09-20 00:00:00-04:00	-0.011545
2022-09-21 00:00:00-04:00	-0.017598
2022-09-22 00:00:00-04:00	-0.008435
2022-09-23 00:00:00-04:00	-0.016897
2022-09-26 00:00:00-04:00	-0.009942
2022-09-27 00:00:00-04:00	-0.002556
2022-09-28 00:00:00-04:00	0.019485
2022-09-29 00:00:00-04:00	-0.021110
2022-09-30 00:00:00-04:00	-0.015584
2022-10-03 00:00:00-04:00	0.026059
2022-10-04 00:00:00-04:00	0.030516
2022-10-05 00:00:00-04:00	-0.002331
2022-10-06 00:00:00-04:00	-0.010369
2022-10-07 00:00:00-04:00	-0.028290
2022-10-10 00:00:00-04:00	-0.007665
2022-10-11 00:00:00-04:00	-0.006353
2022-10-12 00:00:00-04:00	-0.003304
2022-10-13 00:00:00-04:00	0.026049

2022-10-14 00:00:00-04:00	-0.023052
2022-10-17 00:00:00-04:00	0.025372
2022-10-18 00:00:00-04:00	0.011681
2022-10-19 00:00:00-04:00	-0.007112
2022-10-20 00:00:00-04:00	-0.008421
2022-10-21 00:00:00-04:00	0.024011
2022-10-24 00:00:00-04:00	0.012162
2022-10-25 00:00:00-04:00	0.015842
2022-10-26 00:00:00-04:00	-0.007563
2022-10-27 00:00:00-04:00	-0.005354
2022-10-28 00:00:00-04:00	0.023512
2022-10-31 00:00:00-04:00	-0.007249
2022-11-01 00:00:00-04:00	-0.004385
2022-11-02 00:00:00-04:00	-0.025416
2022-11-03 00:00:00-04:00	-0.010350
2022-11-04 00:00:00-04:00	0.014291
2022-11-07 00:00:00-05:00	0.009520
2022-11-08 00:00:00-05:00	0.005381
2022-11-09 00:00:00-05:00	-0.020817
2022-11-10 00:00:00-05:00	0.053497
2022-11-11 00:00:00-05:00	0.009632
2022-11-14 00:00:00-05:00	-0.008543
2022-11-15 00:00:00-05:00	0.008493
2022-11-16 00:00:00-05:00	-0.007658
2022-11-17 00:00:00-05:00	-0.003065
2022-11-18 00:00:00-05:00	0.004530
2022-11-21 00:00:00-05:00	-0.003643
2022-11-22 00:00:00-05:00	0.013367
2022-11-23 00:00:00-05:00	0.006282
2022-11-25 00:00:00-05:00	-0.000224
2022-11-28 00:00:00-05:00	-0.016086
2022-11-29 00:00:00-05:00	-0.001719
2022-11-30 00:00:00-05:00	0.031015
2022-12-01 00:00:00-05:00	-0.000736
2022-12-02 00:00:00-05:00	-0.001154
2022-12-05 00:00:00-05:00	-0.018153
2022-12-06 00:00:00-05:00	-0.014520
2022-12-07 00:00:00-05:00	-0.001703
2022-12-08 00:00:00-05:00	0.007803
2022-12-09 00:00:00-05:00	-0.007498
2022-12-12 00:00:00-05:00	0.014314
2022-12-13 00:00:00-05:00	0.007541
2022-12-14 00:00:00-05:00	-0.006414
2022-12-15 00:00:00-05:00	-0.024766
2022-12-16 00:00:00-05:00	-0.011876
2022-12-19 00:00:00-05:00	-0.008516
2022-12-20 00:00:00-05:00	0.001367
2022-12-21 00:00:00-05:00	0.014842
2022-12-22 00:00:00-05:00	-0.014369
2022-12-23 00:00:00-05:00	0.005736
2022-12-27 00:00:00-05:00	-0.003951
2022-12-28 00:00:00-05:00	-0.012506
2022-12-29 00:00:00-05:00	0.017840
2022-12-30 00:00:00-05:00	-0.002638
2023-01-03 00:00:00-05:00	-0.004219
2023-01-04 00:00:00-05:00	0.007691
2023-01-05 00:00:00-05:00	-0.011479
2023-01-06 00:00:00-05:00	0.022673
2023-01-09 00:00:00-05:00	-0.000567
2023-01-10 00:00:00-05:00	0.006988
2023-01-11 00:00:00-05:00	0.012569
2023-01-12 00:00:00-05:00	0.003634
2023-01-13 00:00:00-05:00	0.003872

2023-01-17 00:00:00-05:00	-0.001834
2023-01-18 00:00:00-05:00	-0.015914
2023-01-19 00:00:00-05:00	-0.007306
2023-01-20 00:00:00-05:00	0.018458
2023-01-23 00:00:00-05:00	0.011927
2023-01-24 00:00:00-05:00	-0.001074
2023-01-25 00:00:00-05:00	0.000375
2023-01-26 00:00:00-05:00	0.010930
2023-01-27 00:00:00-05:00	0.002295
2023-01-30 00:00:00-05:00	-0.012626
2023-01-31 00:00:00-05:00	0.014596
2023-02-01 00:00:00-05:00	0.010572
2023-02-02 00:00:00-05:00	0.014452
2023-02-03 00:00:00-05:00	-0.010686
2023-02-06 00:00:00-05:00	-0.006130
2023-02-07 00:00:00-05:00	0.012994
2023-02-08 00:00:00-05:00	-0.010995
2023-02-09 00:00:00-05:00	-0.008707
2023-02-10 00:00:00-05:00	0.002331
2023-02-13 00:00:00-05:00	0.011671
2023-02-14 00:00:00-05:00	-0.000460
2023-02-15 00:00:00-05:00	0.003242
2023-02-16 00:00:00-05:00	-0.013864
2023-02-17 00:00:00-05:00	-0.002501
2023-02-21 00:00:00-05:00	-0.020265
2023-02-22 00:00:00-05:00	-0.001379
2023-02-23 00:00:00-05:00	0.005305
2023-02-24 00:00:00-05:00	-0.010740
2023-02-27 00:00:00-05:00	0.003400
2023-02-28 00:00:00-05:00	-0.003703
2023-03-01 00:00:00-05:00	-0.005707

Return to Report