

# Stock\_R\_Squared\_Chart

September 29, 2021

## 1 Stock R-Squaared Chart

```
[1]: # Library
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

import warnings
warnings.filterwarnings("ignore")

from pandas_datareader import data as pdr
import yfinance as yf
yf.pdr_override()

[2]: start = '2016-01-01' #input
end = '2020-07-01' #input
symbol1 = '^GSPC' #input
symbol2 = 'AMD' #input

[3]: market = yf.download(symbol1, start=start, end=end)['Adj Close']
stocks = yf.download(symbol2, start=start, end=end)['Adj Close']

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

[4]: market_returns = market.pct_change().dropna()
stocks_returns = stocks.pct_change().dropna()

[5]: def r_squared(stocks_returns, market_returns):
    correlation_matrix = np.corrcoef(stocks_returns, market_returns)
    correlation_xy = correlation_matrix[0,1]
    r_squared = correlation_xy**2
    return r_squared

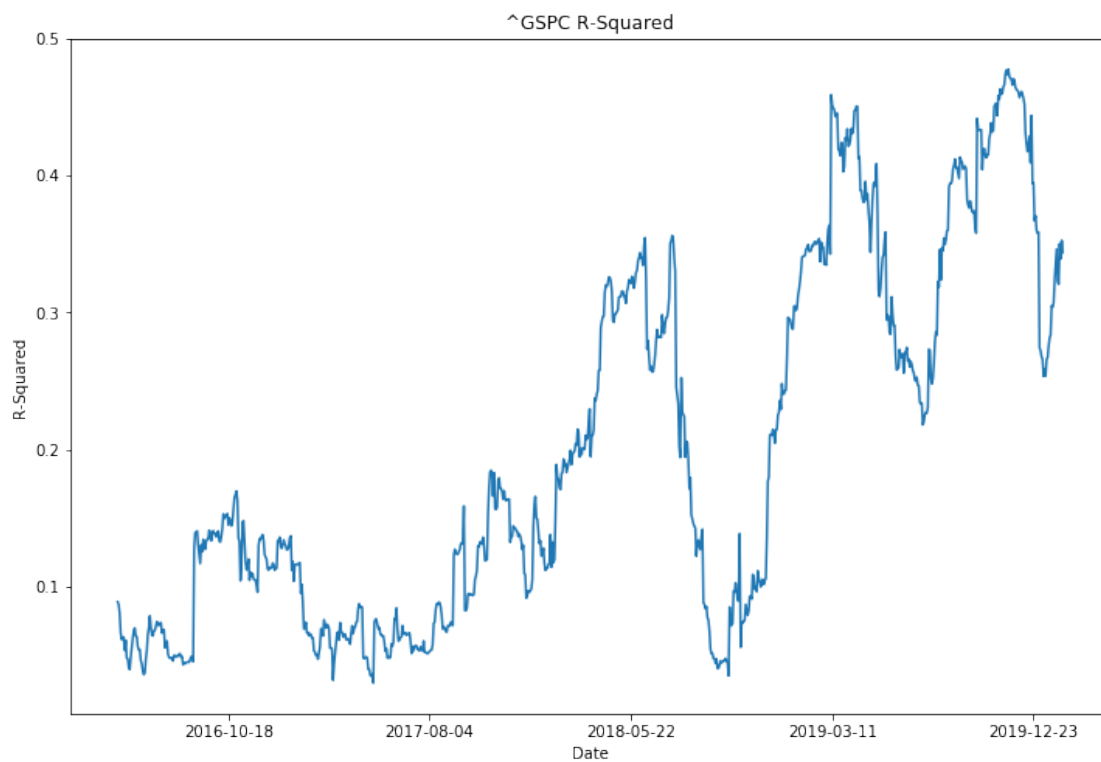
[6]: # Compute the running Beta
running = [r_squared(stocks_returns[i-90:i], market_returns[i-90:i]) for i in
↳range(90, len(stocks_returns))]
```

```

# Plot running Beta up to 100 days before the end of the data set
_, ax1 = plt.subplots(figsize=(12,8))
ax1.plot(range(90, len(stocks_returns)-100), running[: -100])
ticks = ax1.get_xticks()
ax1.set_xticklabels([stocks.index[int(i)].date() for i in ticks[: -1]]) # Label_
    ↳ x-axis with dates
plt.title(symbol1 + ' R-Squared')
plt.xlabel('Date')
plt.ylabel('R-Squared')

```

```
[6]: Text(0, 0.5, 'R-Squared')
```



```
[7]: r_squared(stocks_returns, market_returns)
```

```
[7]: 0.19608261131339016
```