

Stock_Adj_Sharpe_Ratio_Chart

September 29, 2021

1 Stock Adjusted Sharpe Ratio Chart

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

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
symbol = 'AMD'

[3]: df = yf.download("AMD", start, end)

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

[4]: returns = df['Adj Close'].pct_change()[1:].dropna()

[5]: # risk free
rf = yf.download('BIL', start=start, end=end)['Adj Close'].pct_change()[1:]

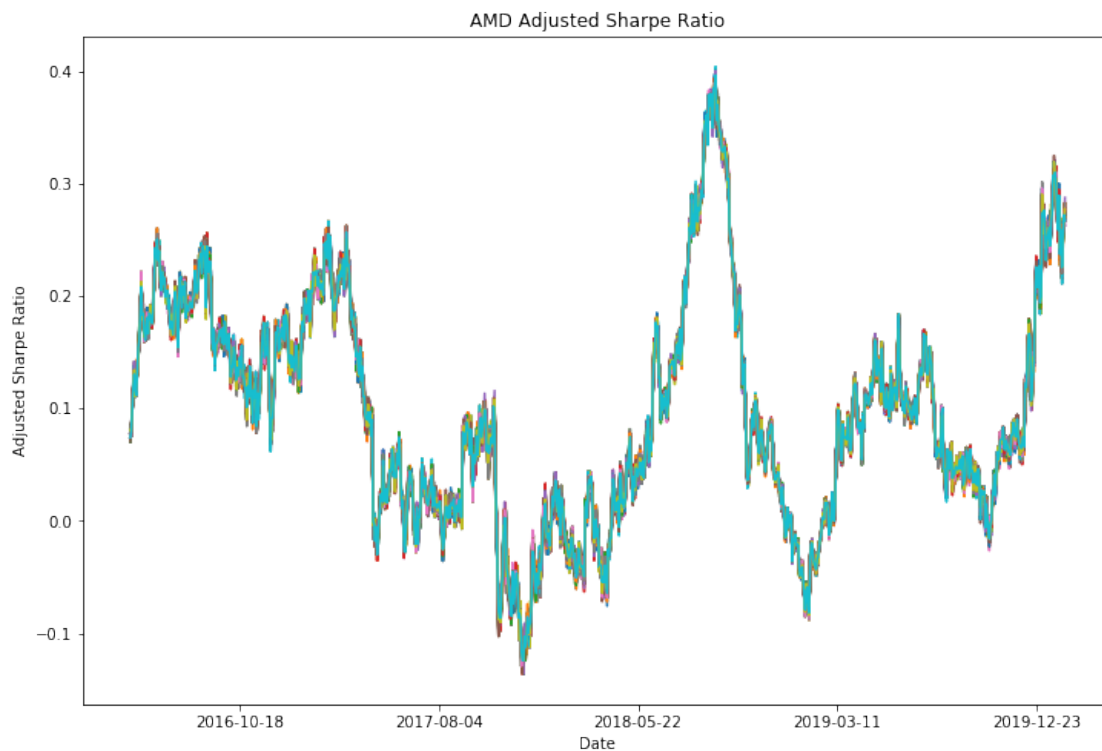
[*****100%*****] 1 of 1 completed

[6]: def adj_sharpe_ratio(stock_returns, rf):
    sharpe_ratio = (stock_returns.mean() - rf) / stock_returns.std()
    stock_skewness = scs.skew(stock_returns)
    stock_kurtosis = stock_returns.kurtosis()
    Adj_SR = sharpe_ratio * (1 + (stock_skewness / 6.0) * sharpe_ratio +
    ↪(stock_kurtosis - 3) / 24.0 * sharpe_ratio**2)
    return Adj_SR
```

```
[7]: # Compute the running Adjusted Sharpe ratio
running_sharpe = [adj_sharpe_ratio(returns[i-90:i], rf[i-90:i]) for i in
↳range(90, len(returns))]

# Plot running Adjusted Sharpe ratio up to 100 days before the end of the data
↳set
_, ax1 = plt.subplots(figsize=(12,8))
ax1.plot(range(90, len(returns)-100), running_sharpe[:-100])
ticks = ax1.get_xticks()
ax1.set_xticklabels([df['Adj Close'].index[int(i)].date() for i in ticks[:-1]])
↳# Label x-axis with dates
plt.title(symbol + ' Adjusted Sharpe Ratio')
plt.xlabel('Date')
plt.ylabel('Adjusted Sharpe Ratio')
```

```
[7]: Text(0, 0.5, 'Adjusted Sharpe Ratio')
```



```
[8]: adj_sharpe_ratio(returns, rf)
```

```
[8]: Date
2016-01-05    0.085122
2016-01-06    0.096359
2016-01-07    0.079548
```

2016-01-08	0.085122
2016-01-11	0.085122
2016-01-12	0.079540
2016-01-13	0.090731
2016-01-14	0.090723
2016-01-15	0.085122
2016-01-19	0.079548
2016-01-20	0.085122
2016-01-21	0.079540
2016-01-22	0.096359
2016-01-25	0.079548
2016-01-26	0.090723
2016-01-27	0.079548
2016-01-28	0.085122
2016-01-29	0.079540
2016-02-01	0.096359
2016-02-02	0.085122
2016-02-03	0.079548
2016-02-04	0.079540
2016-02-05	0.085122
2016-02-08	0.090731
2016-02-09	0.079540
2016-02-10	0.090731
2016-02-11	0.079540
2016-02-12	0.090731
2016-02-16	0.085122
2016-02-17	0.085122
...	
2020-05-19	0.085122
2020-05-20	0.085122
2020-05-21	0.082334
2020-05-22	0.087917
2020-05-26	0.082334
2020-05-27	0.087917
2020-05-28	0.085122
2020-05-29	0.085122
2020-06-01	0.082334
2020-06-02	0.085122
2020-06-03	0.085122
2020-06-04	0.085122
2020-06-05	0.090720
2020-06-08	0.082333
2020-06-09	0.085122
2020-06-10	0.082334
2020-06-11	0.090720
2020-06-12	0.079551
2020-06-15	0.085122

2020-06-16	0.085122
2020-06-17	0.085122
2020-06-18	0.087917
2020-06-19	0.085122
2020-06-22	0.085122
2020-06-23	0.085122
2020-06-24	0.085122
2020-06-25	0.082334
2020-06-26	0.087917
2020-06-29	0.085122
2020-06-30	0.085122

Name: Adj Close, Length: 1130, dtype: float64