Summer Project

Report

The Vattikuti-Viswanathan Ratio

Bachelor of Science

in

Physics and Computer Science

Submitted by

UT eid	Names of Students
vv8379	Venkata Hasith Vattikuti hasith@utexas.edu
av37768	Anand Viswanathan anand.viswanathan@utexas.edu

Department of Physics and the Department of Computer Science

The University of Texas at Austin

Austin, Texas, United States – 78712

Summer 2022

Abstract

Volatility is an investment term indicating when a market or security experiences periods of unpredictable, and sometimes sharp, price movements. The beta value has been referred to as a measurement of how volatile a security is in relation to the market. However, beta performs the best under large time frames and has underlying assumptions that can make the ratio lose meaning and even mislead analysts. In this project, we propose a new measurement to determine the volatility of a security based on how linear the local price movements are. Using segmentation, we can analyze complex security charts in a simple, linear matter over any time frame.

Contents

List of Figures

Chapter 1

The Problem

The beta value's derivation from the Security Characteristic Line comes with the assumption that the security correlates with the market; in this increasingly interconnected world, interdependence is a fair assumption, yet there are quite a few securities that deviate from it. "Rogue" securities' beta values will reflect little information about its volatility and can confuse investors about the risk of said securities.

Furthermore, beta's effectiveness is limited to a large time frame where asset return on investments and market return on investments are measured with ample time to determine clear correlation and to have enough data to run a proper regression. In order to combat these shortcomings, we need a performance metric that can accurately and reliably measure the volatility of complicated, unpredictable securities over any time period.

Chapter 2

Introduction

2.1 Background and Recent Research

2.1.1 Beta values' misleading calculations

Beta values taken at face value may mislead investors into thinking a security's performance is related to the market's performance, but analyzing the correlation between beta and multiple data points of an asset's performance divided by market performance show that many securities' have no relation with the market whatsoever. For example our calculated one-year (limited to one-year by the data we had access to) beta for Netflix is 1.7 meaning that–ideally–for every one percent return on investment in the market, Netflix would return 1.7 percent on investment. However, looking at the correlation between our calculated beta and Netflix's market adjusted ROI points, you can see that there is very little correlation between Netflix and the market ($r^2 = 0.08$). Refer figure 2.1.

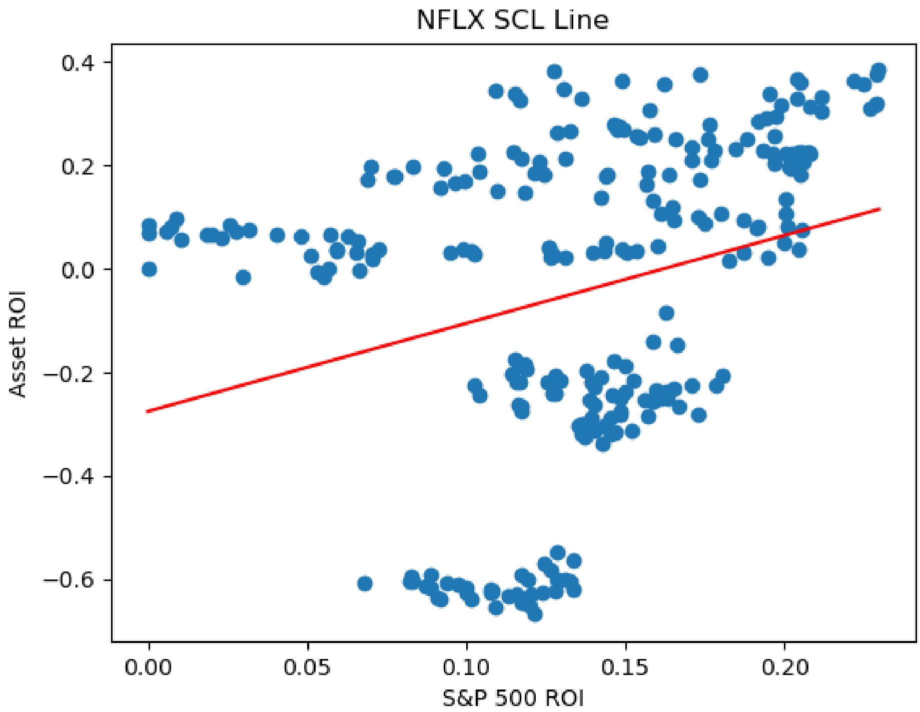

Figure 2.1: The clearly uncorrelated relationship between the market and Netflix

Chapter 3

Work Done

3.1 Slope and Segmentation

Using python and python libraries such as pandas, numpy, and matplotlib, we created a program to plot out the historical closing prices over a one year time period. Then added the ability to split the year into multiple 30 day time periods and created local lines of best fit for each individual time period. This allows us to observe and analyze more localized linear trends of our choosing rather than just yearly overall trends. in our ratio, the average slope (m) is utilized display whether an asset is decreasing or increasing in value to add context to our volatility measurement. Refer figure 3.1.

```python
period = 30
avg_slope = 0

# dataframe of the roi of an asset
df["ROI"] = roi_from_start(df["close"])

# for loop breaks up the time frame into periods of 30 days
for i in range(math.floor(df.shape[0] / period)):
    # each period has a line of best fit calculated for it
    slope, intercept, r_value, p_value, std_err =
        scipy.stats.linregress(df.index[i * period:(i + 1) *
        period], df["ROI"][i * period:(i + 1) * period])

    # we will have the average slope at the end of the computation
    avg_slope += slope / (math.floor(df.shape[0] / period))
```

Refer figure 3.1.

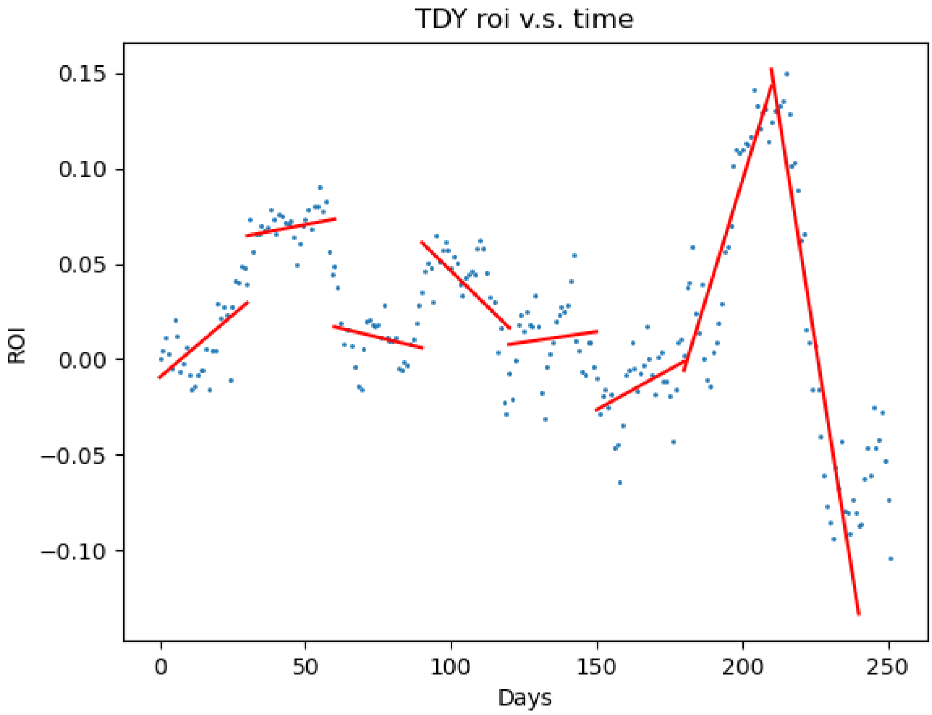

Figure 3.1: segmented line of best fit for the TDY stock. The avg correlation($\bar{r^2}$) and avg slope ($\bar{m}$) were calculated from the r^2 and slope(m) from each segment.

3.1.1 Correlation

To determine how volatile a stock is, we take the r^2 value of the roi v.s. time to get a estimate on how "predictable" a security's movement was for that time period and we average the r^2 values across all time periods to get $\bar{r^2}$. $\bar{r^2}$ is, as we found, a good measure of how unpredictable a security's movement has been. Refer figure 3.2.

```
# adding this inside the for loop above gives us the average r2
    value
avg_r2 += (r_value**2) / (math.floor(df.shape[\(\bar{m}\)0] /
    period))
```

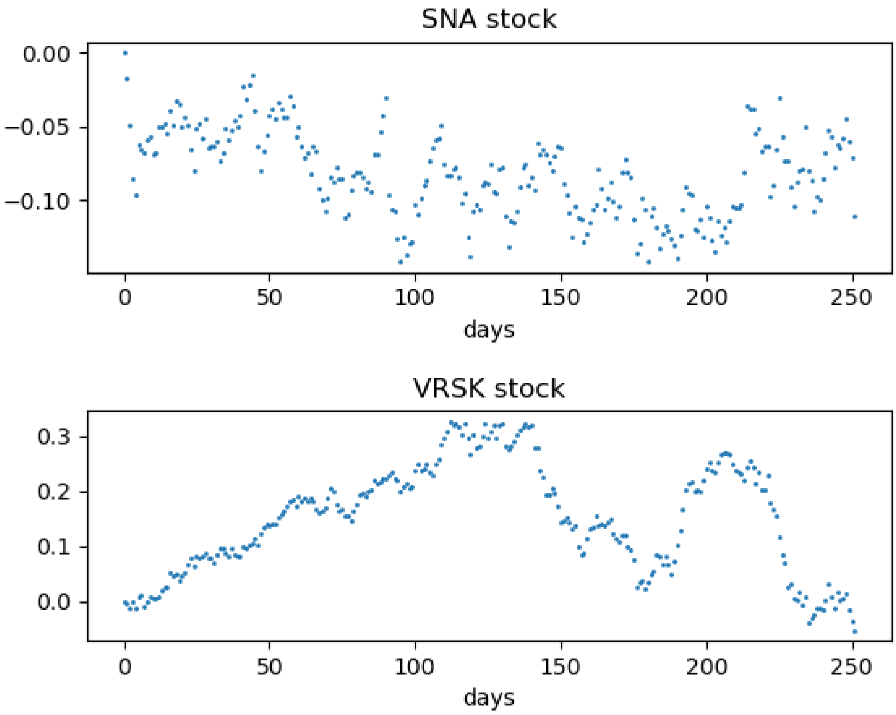

Figure 3.2: The extreme $\bar{r^2}$ values in the S&P 500. The stock SNA has the lowest $\bar{r^2}$ at .072 and the stock VRSK had the highest $\bar{r^2}$ value at .683.

3.1.2 Putting them together

The average slopes of all the lines of best fit also tell us the stocks movement across these periods and the correlation tells us how volatile the stocks are in regards to the local trend. Dividing the slopes by the correlation creates our ratio ($\frac{\bar{m}}{\bar{r^2}}$) that roughly tells us the average reward for our calculated volatility.

```
# the ratio is defined as the computed average slope over the
    average r2 for each of the 30 day periods across the time frame
vv = avg_slope/avg_r2
```

3.1.3 Advantages vs. Weaknesses

The advantages of our ratio are that we can use it to analyze any time period whereas beta works best only with time frames that show a consistent correlation in relation to the market–which are usually large. Because we are using and presenting the correlation in the ratio, we can be certain that the ratio will accurately represent the data and not misconstrue the facts.

If a time segment is unusually uncorrelated ($\bar{r^2} << 1$), then the resulting ratio of $\frac{\bar{m}}{r^2}$ for a single segment will be extraordinarily large and skew the final average in favor of that one period and reflect less of the overall trend. Also, the average $\bar{r^2}$ and $\bar{m}$ values can overpower one another to distort the ratio. For example a volatile stock with a really low roi and an even lower correlation could have the same ratio as a stable stock with a really high roi and a close to perfect correlation. Unlike beta, the ratio will not explicitly show risk because there is no comparison to the market as a standard. Refer figure 3.3.

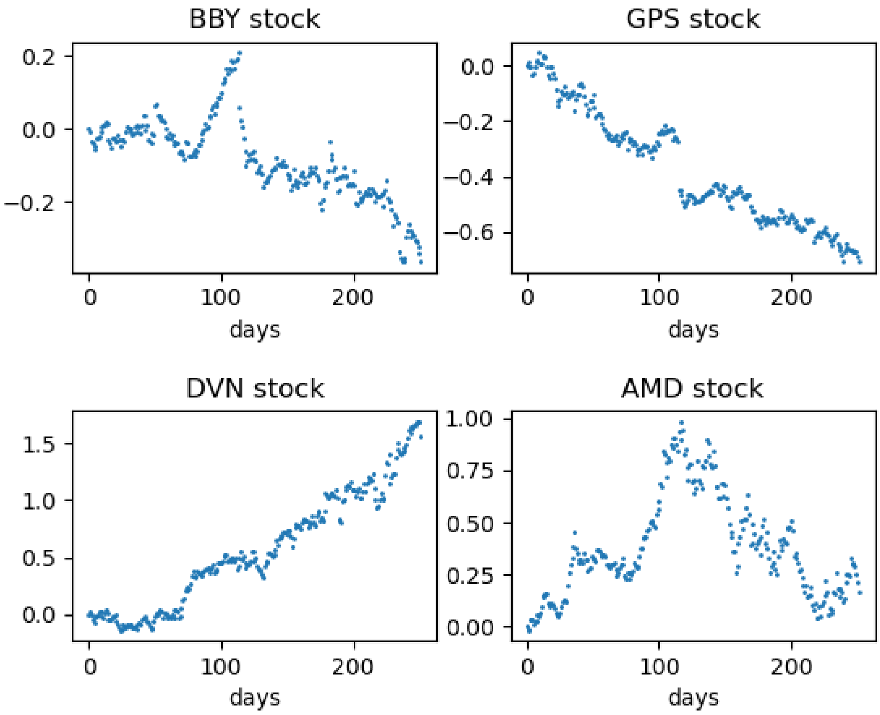

Figure 3.3: The stock BBY has the lowest vv-ratio in the S&P 500 at -.0071 with $\bar{m} = -.0013$ and an avg $\overline{r^2} = -.183$ which shows a very volatile stock with a low, negative rate of gains.The stock GPS has one of the lowest vv-ratios in the S&P 500 at -.0052 with $\bar{m} = -.0022$ and an avg $\overline{r^2} = .417$ which shows a very stable stock decreasing at a high, negative rate. The stock DVN has the highest vv-ratio in the S&P 500 at .0138 with $\bar{m} = .0069$ and an avg $\overline{r^2} = .500$ which shows a very stable stock with a high, positive rate of gains. The stock AMD has the second highest vv-ratio in the S&P 500 at .0132 with $\bar{m} = .0026$ and an avg $\overline{r^2} = .217$ which shows a very volatile stock with a low, positive rate of gains.

These stocks are prime examples of how the vv-ratio can accurately depict the rewards of an asset with a stable stock with a high rate of gains versus a volatile stock with a low rate of gains producing similar rewards. However, since the vv-ratio abstracts the relationship between $\bar{m}$ and $\overline{r^2}$ into one number, the risk from the volatility isn't shown at all. Therefore it is best to use the vv-ratio alongside the $\bar{m}$ and $\overline{r^2}$ in order to contextualize the risk into the asset's possible rewards.

3.1.4 Examples

The main assets we looked at all came from the S&P 500 index which we also used as our market. After computing the average r^2 value and the average slope for each asset, we look at the distribution of the average r^2 and vv-ratio (average slope/average r^2). Refer figure 3.4.

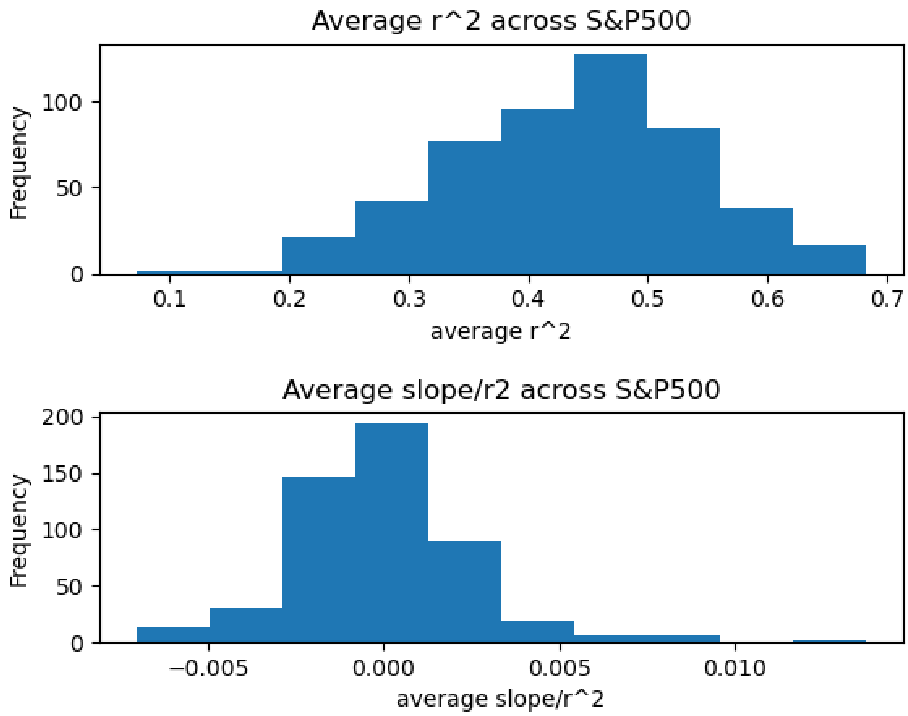

Figure 3.4: approximately normal distributions of the vv-ratios and r^2 for all assets

3.1.5 Full code

```
def get_moving_avg_r2(ticker):
    df = pd.read_csv("S&P500 Prices/" + ticker +
       "_historical_closing_prices.csv", sep="\t",
       encoding='utf-8')

    period = 30
```

```python
    avg_r2 = 0
    avg_slope = 0


    df["ROI"] = roi_from_start(df["close"])

    for i in range(math.floor(df.shape[0] / period)):
        slope, intercept, r_value, p_value, std_err =
            scipy.stats.linregress(df.index[i * period:(i + 1) *
            period], df["ROI"][i * period:(i + 1) * period])


        avg_slope += slope / (math.floor(df.shape[0] / period))
        avg_r2 += (r_value**2) / (math.floor(df.shape[0] / period))


    return avg_slope, avg_r2
```

for the full repository with the data sets we used, visit our github repo at `https://github.com/Screeeech/vv-ratio`