Vysoká škola ekonomická v Praze

Quantitative analysis of capital market

4ST441

Statistické metody a kapitálové trhy (v angličtině)

LS 2018/2019

Jaroslav Novák

xnovj151

Obsah

Introduction

In this paper I will focus on companies which are included in S&P 500, S&P 400 and S&P 600 indices. Each index includes companies of different size. The Size of the companies is based on market capitalization.

Firstly, I will investigate the relation between mean log returns and standard deviation of log returns for all the companies mentioned above. Then I will try to find out if the size of companies has any effect on this relationship.

Finally, I will choose one of the companies with the best mean log returns and risk (standard deviation of log returns) ratio and enough trading days in considered period. For the chosen company I will make ARIMA – GARCH model.

The analysis will be conducted in R.

1 Stock returns

The goal of trading on stock markets is to make a profit. The amount of a profit or a loss from a trading strategy depends on changes in prices of stocks and the amount of stocks being traded. Hence traders are rather interested in a relative measure of their profits in order to estimate how well a certain strategy performs. Returns are able to provide this measure because they are expressed as a relation of changes in price to the initial price. Returns are scale-free, meaning that they do not depend on units.

1.1 Gross returns

Let Pt be the price of a stock at time index t. Assuming that a stock pays no dividend. For one period from date t - 1 to date t the simple gross return can be calculated as:

$$G_t = \frac{P_t}{P_{t-1}} = 1 + R_t$$

The gross return over the k periods is the product of the k single-period gross returns. Thus, the k-period simple gross return is just the product of the k one-period simple gross returns. This is called a compound or cumulative return:

$$G_t(k) = \frac{P_t}{P_{t-k}} = \frac{P_t}{P_{t-1}} \frac{P_{t-1}}{P_{t-2}} \times \cdots \times \frac{P_{t-k+1}}{P_{t-k}}$$

1.2 Net returns

Net return over the period from time t - 1 to time t is:

$$R_t = \frac{P_t}{P_{t-1}} - 1 = \frac{P_t - P_{t-1}}{P_{t-1}}$$

Pt - Pt-1 in the numerator is the revenue over the period from time t - 1 to time t and Pt-1 is the initial price of a stock. Therefore, the net return can be considered as the relative profit.

The formula for net returns over k periods is as follows:

$$R_t(k) = \frac{P_t}{P_{t-k}} - 1 \; = \frac{P_t - P_{t-k}}{P_{t-k}}:$$

1.3 Log returns

Log returns are denoted by r_t and can be obtained as natural logarithm of simple gross returns.

$$r_t = ln\left(\frac{P_t}{P_{t-1}}\right)$$

The log returns have some interesting properties:

- *k*-period log return is simply sum of the single period log returns:

$$r_t(k) = r_t + r_{t-1} + r_{t-2} + \cdots r_{t-k+1}$$

- Small values of log return can be approximately interpreted as the simple net return.
- Thus, log returns can be approximately interpreted as percentage change of the price.
- Under assumption of log normally distributed prices the log returns are normally distributed.

2 ARIMA and GARCH models

2.1 ARIMA model

ARIMA stands for autoregressive integrated moving average and it has 3 parameters *p, d, q*.

Autoregression means that a value of a time series is possible to express as linear combination of some number of lagged observations. The *p* parameter shows how many lagged observations to be taken in consideration. Autoregressive model can be express like this:

$$X_t = \phi_0 + \sum_{i=1}^{p} \phi_i \, X_{t-i} \; + \; \varepsilon_t$$

Integrated means applying differencing of order *d* to observations of a time series. The order of differencing expresses how many times will be the differencing applied.

Moving average model uses past values of the white noise process *at* to express the value of an observation of a time series. It can be expressed as follows:

$$X_t = c_0 + \varepsilon_t + \sum_{i=1}^{q} \Theta_i \, \varepsilon_{t-i}$$

The parameter *q* denotes the lag of the error component.

The whole ARIMA model (assuming $d = 0$) can be expressed in following form:

$$X_t = \phi_0 + \sum_{i=1}^{p} \phi_i X_{t-i} + \varepsilon_t + \sum_{i=1}^{q} \Theta_i \varepsilon_{t-i}$$

Where ϕ_0, ϕ_i, Θ_i are parameters and ε_t is Gaussian white noise with zero mean and variance σ^2.

2.2 GARCH model

GARCH models are often used with financial time series. By using GARCH models it is possible to achieve more accurate models and prediction than by using just ARIMA models. There is one main reason for that and that is volatility clustering (volatility changes over time).

Volatility is the conditional standard deviation (or variance) of future value of given time series conditioned on the history of given time series. ARIMA model assumes constant volatility. Therefore, models such as GARCH are being used to capture the changes in volatility.

In financial markets volatility is becoming higher during periods of financial political crisis or other events and on the other hand becoming lower during periods of calm economic growth.

A GARCH model stands for Generalized Autoregressive Conditional Heteroskedasticity model and it is given as:

$$\varepsilon_t = \sigma_t w_t$$

Where $\{w_t\}$ is white noise with zero mean and unit variance and σ_t^2 is given by:

$$\sigma_t^2 = \alpha_0 + \sum_{i=1}^{q} \alpha_i \varepsilon_{t-i}^2 + \sum_{j=1}^{p} \beta_j \sigma_{t-j}^2$$

A GARCH(1,1) model is then specified like this:

$$\sigma_t^2 = \alpha_0 + \alpha_1 \varepsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2$$

3 Data

3.1 Introduction

As stated above this analysis will concern just companies from the mentioned indices. Firstly, I will shortly introduce the indices. These indices were developed and are being published by S&P Dow Jones Indices. The indices are based on companies listed on American stock exchange such as NYSE or NASDAQ. Each index contains company of different size based on market capitalization.

Market capitalization is the aggregate market value of a company represented in dollar amount. It is equal to share price of a company multiplied by the number of outstanding shares. S&P 500 includes 500 largest companies with market capitalization greater than 5.3 billion USD. S&P 400 contains 400 companies with medium market capitalization between 1.6 billion

and 6.8 billion USD. Lastly, S&P 600 is made of 600 companies with small market capitalization which range from 450 million to 2.1 billion USD. Various sources provide different numbers, so they are just approximate to get the idea.

3.2 Preprocesing

In order to acquire the data, it was necessary to do some web-scraping and some data transformation. To do so I was inspired by (Dancho, 2019). I extended his work for including also companies from S&P 400 and S&P 600 index. I will briefly describe the steps I made to make the R code more understandable.

The first task was to get the ticker symbols for all the companies. I used the fact that Wikipedia contains the tables of companies and their ticker symbols for S&P 500, S&P 400, S&P 400 and S&P 600 combined. The structure of wikipedia web pages for S&P 500 and S&P 400 was the same. These web pages contained only one table with company names, ticker symbols and information about industry sector and subsector. It was easy to web-scrape these data using the *rvest* package. I basically loaded html code of the pages into R, then searched the code for the table and then transformed it into a data frame. I also added to this data frame column that denotes the size of the company.

To web-scrape the data from the Wikipedia webpage that contained table for companies from S&P 400 and S&P 600 combined was little bit trickier, because the webpage had slightly different structure and contained more than one table. More detailed description is in the enclosed R script.

When I got all the ticker symbols and other information about the companies into data frames, it was needed to transform the data so functions from *quantmod* package would be applicable. The data from Wikipedia had different format of ticker symbols than that is being used by yahoo finance. There were also problems due to duplicities in the data.

After all this data transformation done there were still problems with downloading the stock prices. The companies that causes error during downloading had to be manually removed from the dataset. One of the reasons of the errors was that the data from Wikipedia were not up to date and the companies listed in Wikipedia table could not be found on yahoo finance anymore. There were also other reasons of the errors that I was not able to explain. The problems were caused just by less than 30 companies, so they were removed from the dataset.

It is possible that other problems with downloading the data might occur in the future. That is why I saved the data as csv file after downloading. So, in case of any future problems with downloading the data it will be still possible to redo and check the analysis.

After all the data was web-scraped and preprocessed it was needed to download the stock prices for all the companies and calculate the log returns based on the adjusted price. The best way to do that was to create a nested data frame. It means that the original data frame with the web-scraped data was extended by new columns. The new columns are actually lists and each element of the list is a data frame. This makes possible to save the webscraped data and the stock prices for each company in one dataframe and consequently calculate mean daily log returns and standard deviation of daily log returns for each company.

First, there were created function to download the stock prices and function to calculate the daily log returns. Applying *map()* function and the function for downloading the stock prices it was possible to create the nested table. The *map()*[1] function returns list of values, in this case list of dataframes containing the downloaded prices. It basically loop trough the original dataset and it downloads the stock prices for each ticker symbol. Daily log returns were calculated in similar way using the *map()* function. For each element of the list containing dataframes of stock prices it applies function to compute daily log returns and save it again as list of dataframes where each dataframe contains the daily log returns for each company. Consequently, was computed mean of daily log returns and standard deviation of daily log returns. The columns containing nested data were then deleted because their were not needed anymore. The reduced dataframe contains just the webscraped data, mean of the daily log returns and its standard deviation.

4 Data analysis

After the data was preprocessed mean and standard deviation of log returns were calculated. The goal is to compare all the company's stocks based on risk versus reward and see if the size of company matters. The second goal is to choose one stock with the best ratio of risk vs reward and built an ARIMA–GARCH model.

4.1 Risk versus Reward

The Risk/Reward ratio might help investors to decide what stocks are the most suitable for investment. Since log returns can be interpreted as relative returns, we can consider their mean as a measure of reward. The more volatile the log returns are the higher their standard deviation is. The volatility of log returns is associated with investment risk. Therefore, the standard deviation of log returns is good measure of risk. The investors are the most interested in stocks with the highest mean of log returns and the lowest standard deviations of log returns. The considered period is from 1.1.2010 to 14.3.2019.

Before choosing company with the best ratio I will present the relation between stock risk and reward using scatterplots. I created interactive scatterplots using *plotly* package. After running the code, it is possible to examine the data in detail thanks to the plot's interactivity. In this paper will be presented just snapshots from the interactive plots. In all the figures is the size of company distinguished by color and the size of points refers to the number of trading days during the considered period.

The first figure shows the risk and reward measures for all stocks of considered companies. There are some outliers with very high risk measure. It makes the plot hard to read. That is why all stocks with standard deviation of log returns higher than 0.06 will not be shown in the other figures.

[1] More information about the *map()* function and nested dataframe can be found here: https://r4ds.had.co.nz/many-models.html

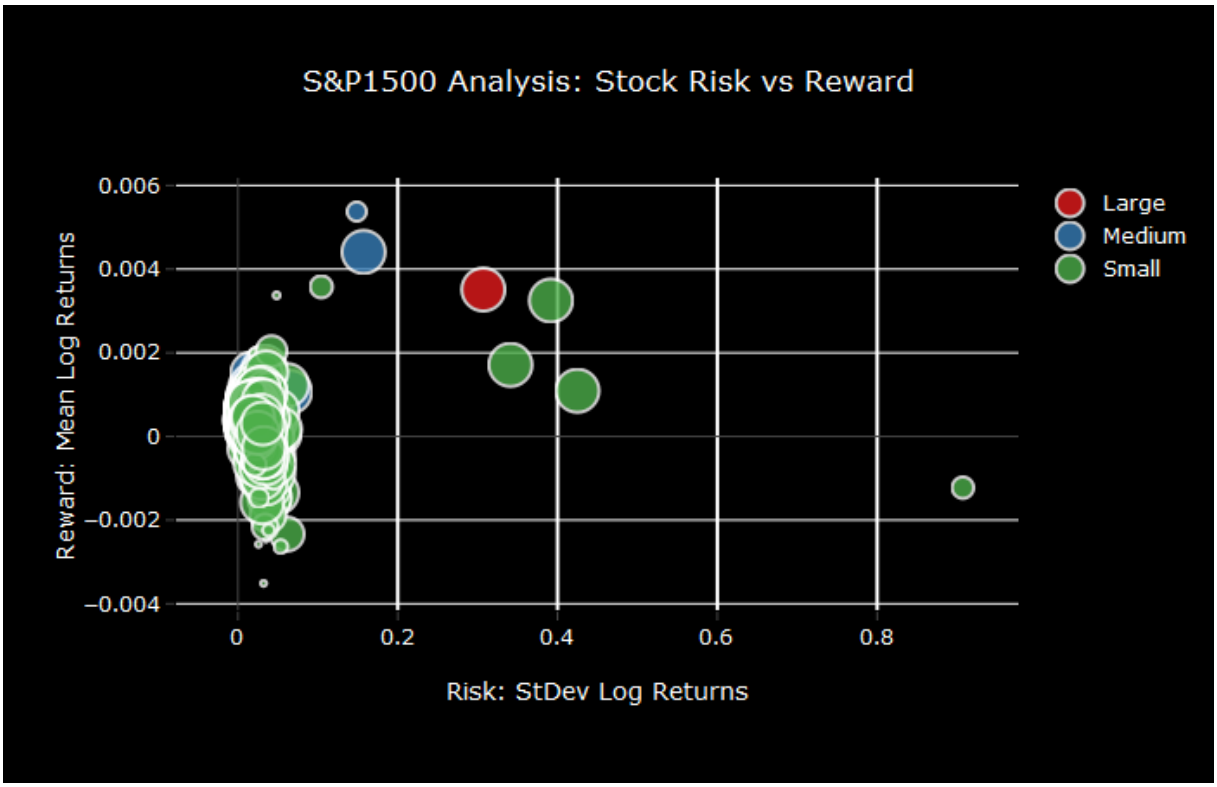

Figure 1: Risk/Reward scatterplot for all companies.

Figure 2: Risk/Reward scatterplot for all companies (zoomed)

The second figure presents all companies with risk measure lower than 0.06. As we can see most of the companies has standard deviation of log returns between 0.01 and 0.04 and the mean of log returns between -0.001 and 0.01. Since there are approximately 1500 stocks, the snapshot of the interactive plot is still hard to read.

The other figures show the same plot for stocks that belongs to companies of different size. There are clearly visible differences. These differences stand out even more while exploring the interactive plots and not just snapshots.

There is much higher frequency of companies with mean log returns higher than zero within the large companies. The relative frequency of companies with positive mean log returns is approximately 95 % for the large companies, 85 % for the medium size companies and 76 % for the small size companies. The plots also show that the negative mean log returns tend to be higher (in absolute value) for smaller company. In the plots can be also seen that the variance of mean log returns grows with the standard deviation of log returns, especially in case of the small companies.

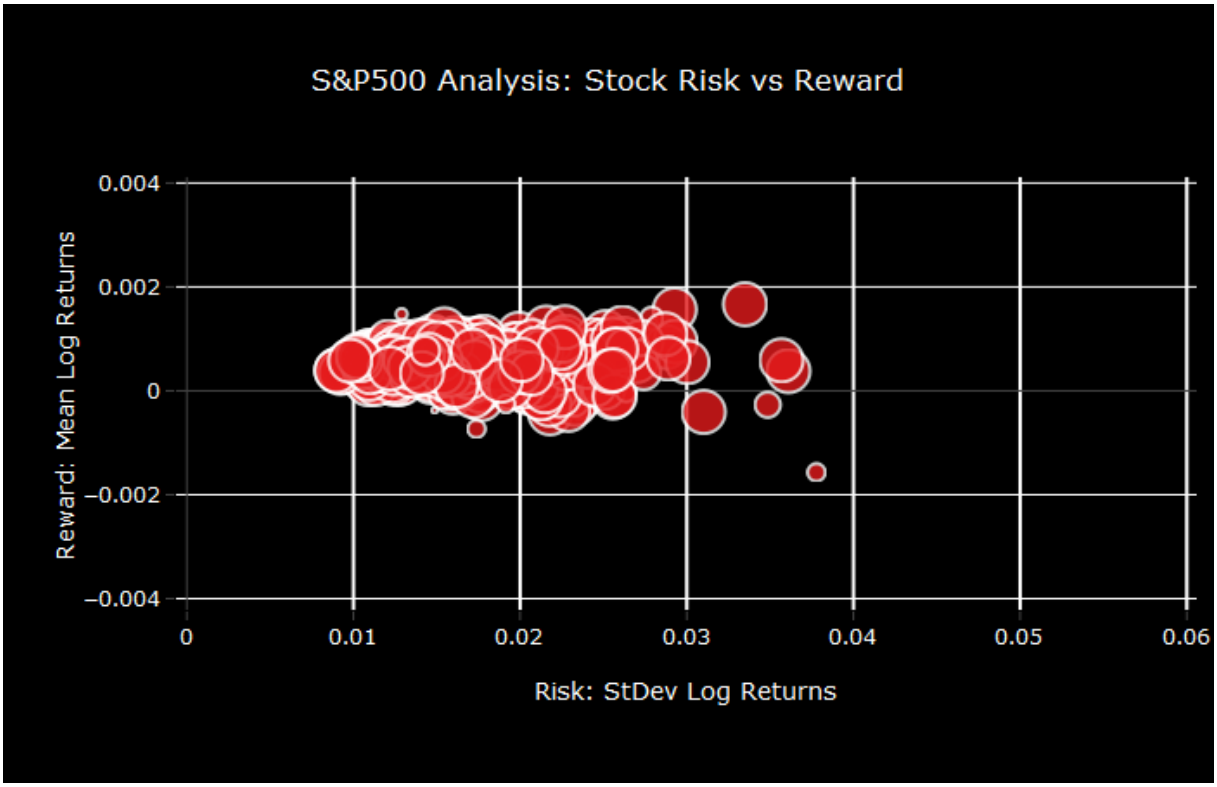

Figure 3: Risk/Reward scatterplot for large companies

From the figures it also seems that there might be a trend. The mean log returns of stocks of large companies seems to grow with standard deviation. On the other hand, there can be observed decreasing trend for medium size company. In case of small companies, it is hard to decide about a trend just from the plot because of much higher variance than in the previous cases. I did not test for statistical significance of the trend in any case. It might be subject for other analysis.

Based on these plots I decide that the companies with the best Risk/Reward ratio should have mean of daily log return of their stocks higher than 0.001 and the standard deviation of log returns lower than 0.025. I wanted to take in to consideration only stocks with more than 1000 trading days. After this condition was applied just 28 companies remained.

Figure 4: Risk/Reward scatterplot for medium size companies

Figure 5: Risk/Reward scatterplot for small companies

4.2 ARIMA-GARCH

In this step I will pick one of the companies with the best Risk/Reward ratio and build an ARIMA-GARCH model for log returns of this company. In the table below, there are 10 companies with the lowest Risk/Reward ratio. I chose Domino's Pizza Inc. It does not have the best value of the ratio, but it also possesses sufficient number of trading days possible during the considered period.

Symbol	Security	Sector	Sub-industry	Size	Trading days	Risk/Reward
LW	Lamb Weston Holdings Inc	Consumer Staples	Packaged Foods & Meats	Medium	586	8.726286
NGVT	Ingevity Corporation	Materials	Specialty Chemicals	Small	720	11.368803
DPZ	Domino's Pizza Inc	Consumer Discretionary	Restaurants	Medium	2313	11.870871
NI	NiSource Inc.	Utilities	Multi-Utilities	Large	2313	12.820590
TDG	TransDigm Group	Industrials	Aerospace & Defense	Large	2313	13.077705
PRAH	PRA Health Sciences	Health Care	Life Sciences Tools & Services	Medium	1088	13.620327
CTAS	Cintas Corporation	Industrials	Diversified Support Services	Large	2313	13.707059
OGS	ONE Gas	Utilities	Gas Utilities	Medium	1297	14.054627
EXR	Extra Space Storage	Real Estate	Specialized REITs	Large	2313	14.066517
FISV	Fiserv Inc	Information Technology	Internet Software & Services	Large	2313	14.074735

Figure 6 shows the time series of adjusted price on the left and on the right series of log returns. The adjusted price rose till the June 2018 exponentially and then it fluctuated around value of 250 USD. The log returns have no significant trend and they fluctuate around value close to zero. It is clear, that its volatility changes over time. The smallest volatility was during the first half of year 2014. There were also periods of high volatility, for example beginning of year 2010.

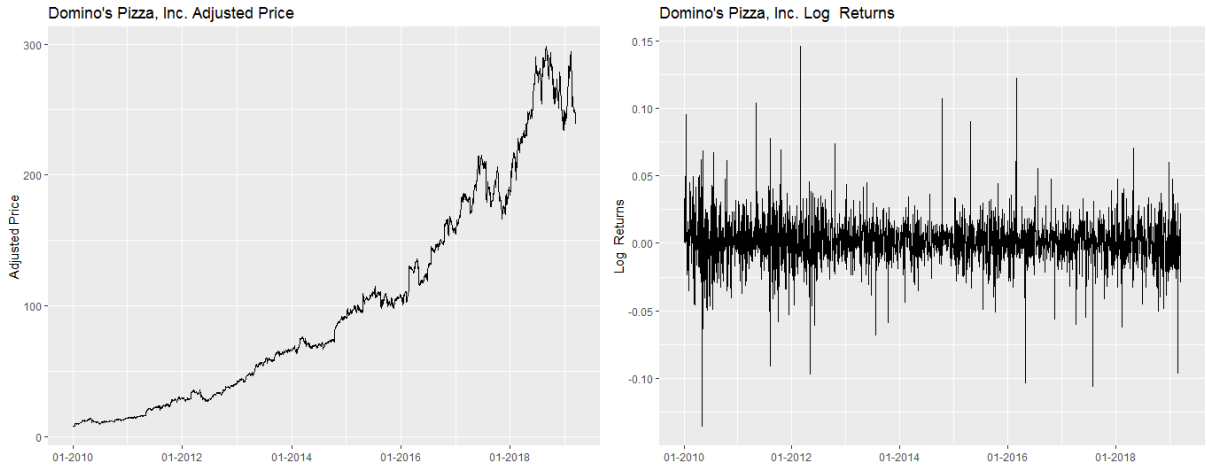

Figure 6: Series of adjusted price and log returns of DPZ stocks.

Firstly, I create an ARIMA model for the log returns. To find parameters of the model that would fit the series the best I used *auto.arima()* function. This function returns best ARIMA model according to its AIC. I set the maximum for *p* and *q* parameter to 5 and for *d* parameter to 2. The function searched trough all the possible models and return the one with the best value of AIC. The best model was ARIMA(4,0,4).

Figure 7 on the left presents the ACF of residuals and squared residuals of the ARIMA(4,0,4) model. According ACF of residuals it seems there is no correlation between the

series values at any lag up to 35. The residuals of this model look like realization of white noise. To test for conditional heteroskedasticity It is needed to have a look on ACF of squared residuals. It is shown in figure 7 on the right. There is significant correlation in first 4 lags of the time series. It leads to the conclusion that heteroskedasticity is present in log returns series. Therefor it is meaningful to create a GARCH model.

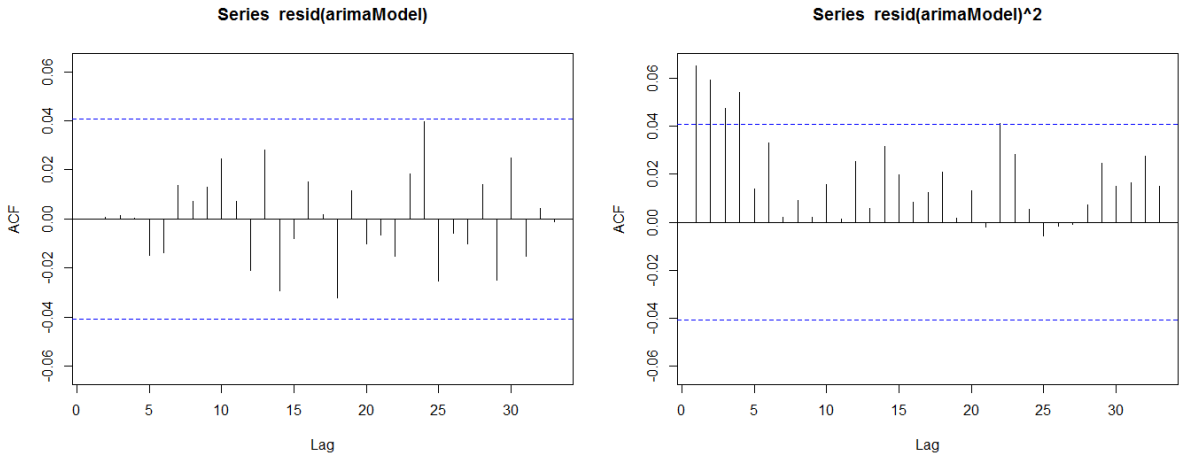

Figure 7: ACF for residuals and squared residuals – ARIMA(4, 0, 1)

At this point of the analysis It is time to fit a GARCH model. To model the conditional variance GARCH(1,1) was used. In the figure 9 there is ACF of residuals and squared residuals of this model. There is not significant evidence of serial correlation in any of the ACFs. The residuals and squared residuals seem like realization of white noise. It is indicating a good fit.

I used *ugarchspec()* and *ugarchfit()* fuctions to get the fited values. The first function is used to specify the model. The function needs to be fed with model of mean and with model of variance. As mean model I used the ARIMA(4, 0, 4) model. For a variance I used the GARCH(1,1) model. The inputs of *ugarchfit()* function are the model specification and data. The fitted values from this model are presented in the figure 8 on the left. The absolute value of the original time series and the modeled conditional standard deviations is shown in figure 8 on the right.

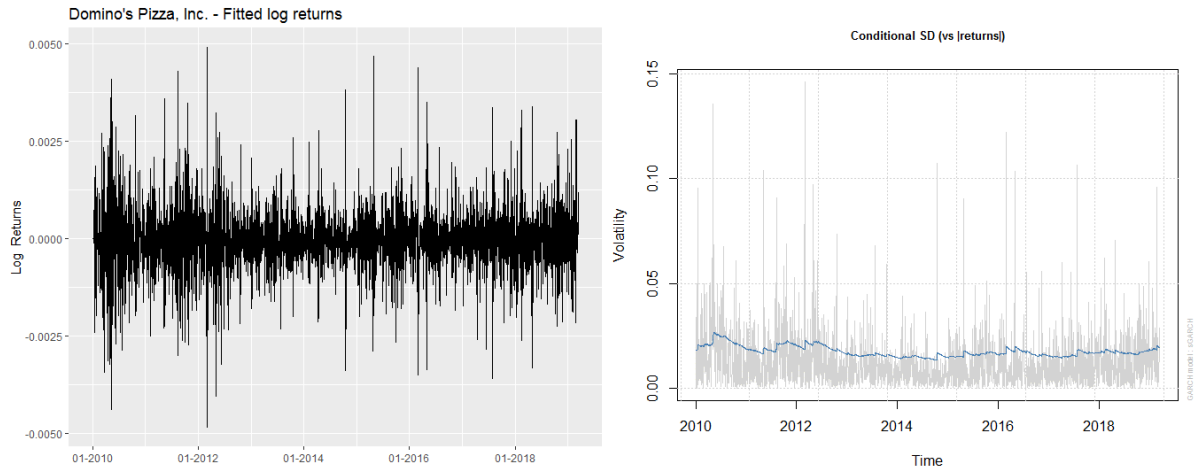

Figure 8: Fitted values and original time series with conditional SD – GARCH(1, 1)

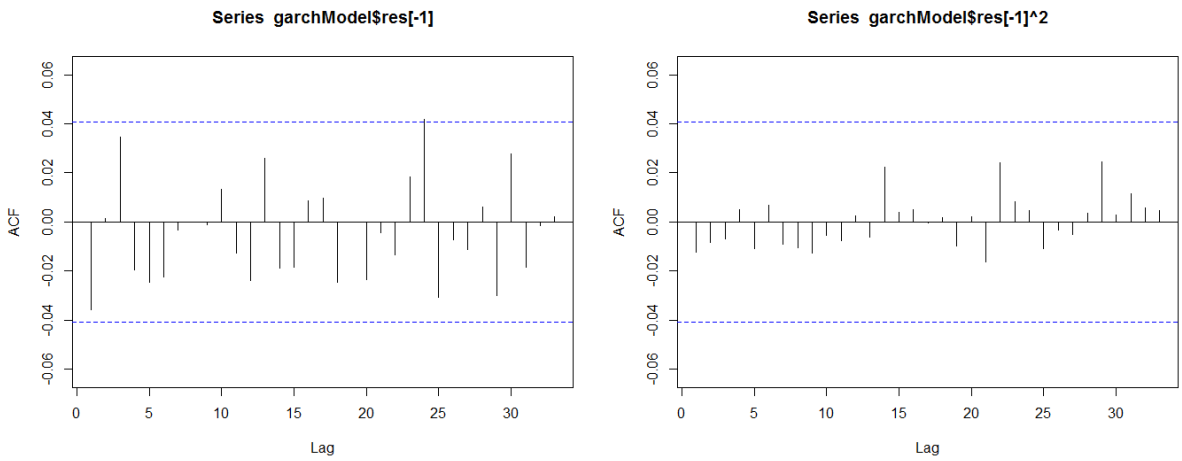

Figure 9: Figure 7: ACF for residuals and squared residuals – GARCH(1,1)

5 Conclusion

The main goal of the paper was to find company with the best Risk/Reward ratio and built a suitable model for log returns of its stock. Considering the number of trading days company with the best Risk/Reward ratio is Domino's Pizza Inc. I found the most suitable ARIMA model for its log returns based on AIC. In order to make the model more precise I used GARCH model that coped with conditional heteroskedasticity. The final model can be further used for prediction and creating trading strategy like in (QuantStart, 2019).

The secondary goal was to shortly describe how distribution of mean log returns and standard deviation differ based on company size. Since I gather data about lot of companies It would be interesting to continue with this work and create a portfolio of 10 or 20 companies in similar manner like in (Dancho, 2019). Build trading strategy for this portfolio consequently and test how does this strategy perform.

While running the R code it is possible that some errors with downloading the date will occur. In order to get the same results I recommend to use enclosed csv file *sp_1500_rd*.

6 References

Dancho, M., 2019. *Quantitative Stock Analysis Tutorial: Screening the Returns for Every S&P500 Stock in Less than 5 Minutes.* [Online]
Available at: https://www.business-science.io/investments/2016/10/23/SP500_Analysis.html

Christoph Hanck, M. A. A. G. a. M. S., 2019. *Introduction to Econometrics with R.* [Online]
Available at: https://www.econometrics-with-r.org/index.html

Kang, E., 2019. *Time Series: ARIMA Model.* [Online]
Available at: https://medium.com/@kangeugine/time-series-arima-model-11140bc08c6

QuantStart, T., 2019. *ARIMA-GARCH Trading Strategy on the SP500 Stock Market Index Using R.* [Online]
Available at: https://www.quantstart.com/articles/ARIMA-GARCH-Trading-Strategy-on-the-SP500-Stock-Market-Index-Using-R

QuantStart, T., 2019. *Generalised Autoregressive Conditional Heteroskedasticity GARCH(p, q) Models for Time Series Analysis.* [Online]
Available at: https://www.quantstart.com/articles/Generalised-Autoregressive-Conditional-Heteroskedasticity-GARCH-p-q-Models-for-Time-Series-Analysis