École polytechnique

Statistical Modeling
MAP565

Modeling S&P 500 Using Time Series techniques

Authors:
M. El mendili

Supervisors:
M. Rosenbaum

June 8, 2020

Contents

1 Introduction

In this report, we will work on S&P 500 index. Using **YahooStockDataSource** Python module, we scrapped its values from 2000 to 2019.

First, we will apply Linear Time Series and GARCH model to its attribute: **Open**. We will implement a methodology to fit and to find the best ARIMA model by minimizing AIC over a non-trivial combinations of parameters. Then, we will use this selected ARIMA model to fit a GARCH model. This methodolody gives residuals that looks like descrete white noise and models well the **Volatility Clustering** phenomenon studied in lectures.

Second, we will implement a simple trading strategy using ARIMA-GARCH models and compare its results to a naive "Hold and Buy" strategy. This methods seems to perform well on periods of high volatility. This is due to the very definition of GARCH model which takes into account the conditional volatility phenomenon.

Finally, we will work on the daily **Volumes** of trades associated to S&P 500. We will fit a Univariate Hawkes model on Extreme volumes of trading events. The fitting will be used by the classical MLE approach. We will also validate our modeling using the Ogata's test on residuals (By QQ plot of residuals Vs Exp(1) distribution).

2 Data Exploratory

We plotted in **Figure 1** the variations of S&P 500 Open values between 2000 and 2019.

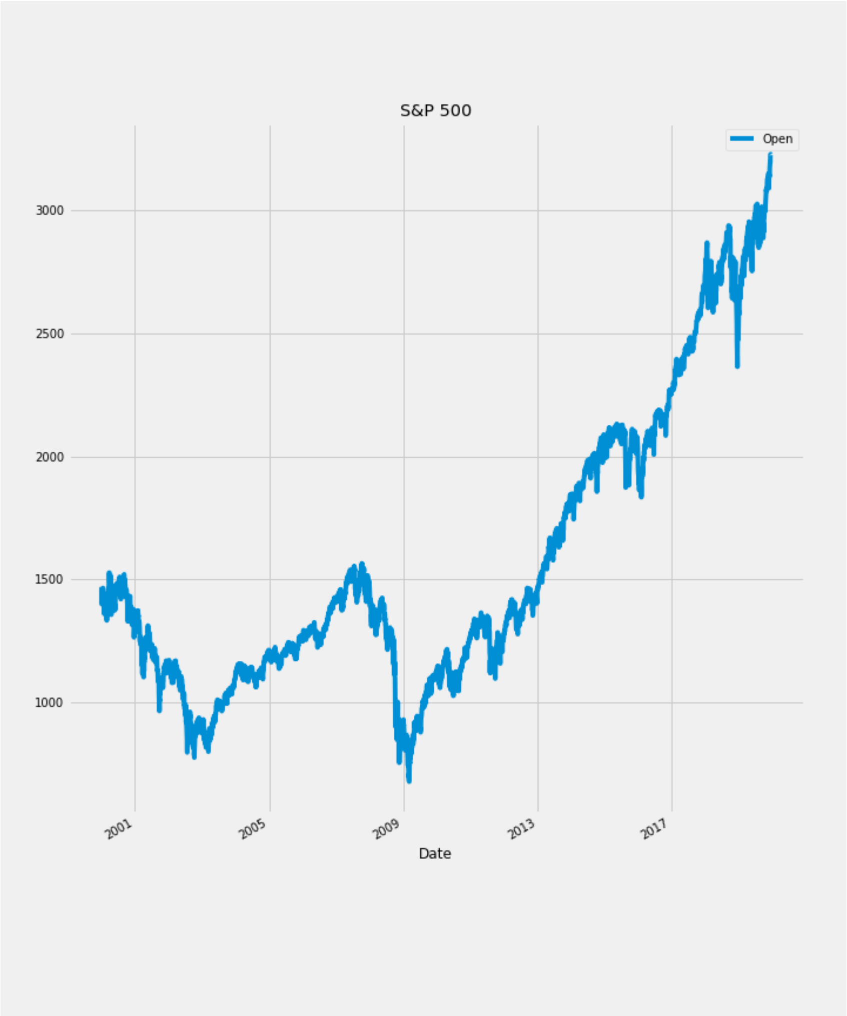


Figure 1: S&P 500 **Open** values between 2000 and 2019.

This time series seems to have trends and to be non-stationary. As seen in lectures, we will use a **Seasonal Decomposition by Moving Averages** to decompose this serie to a trend, seasonal and residual components. The result is plotted in **Figure 2**.

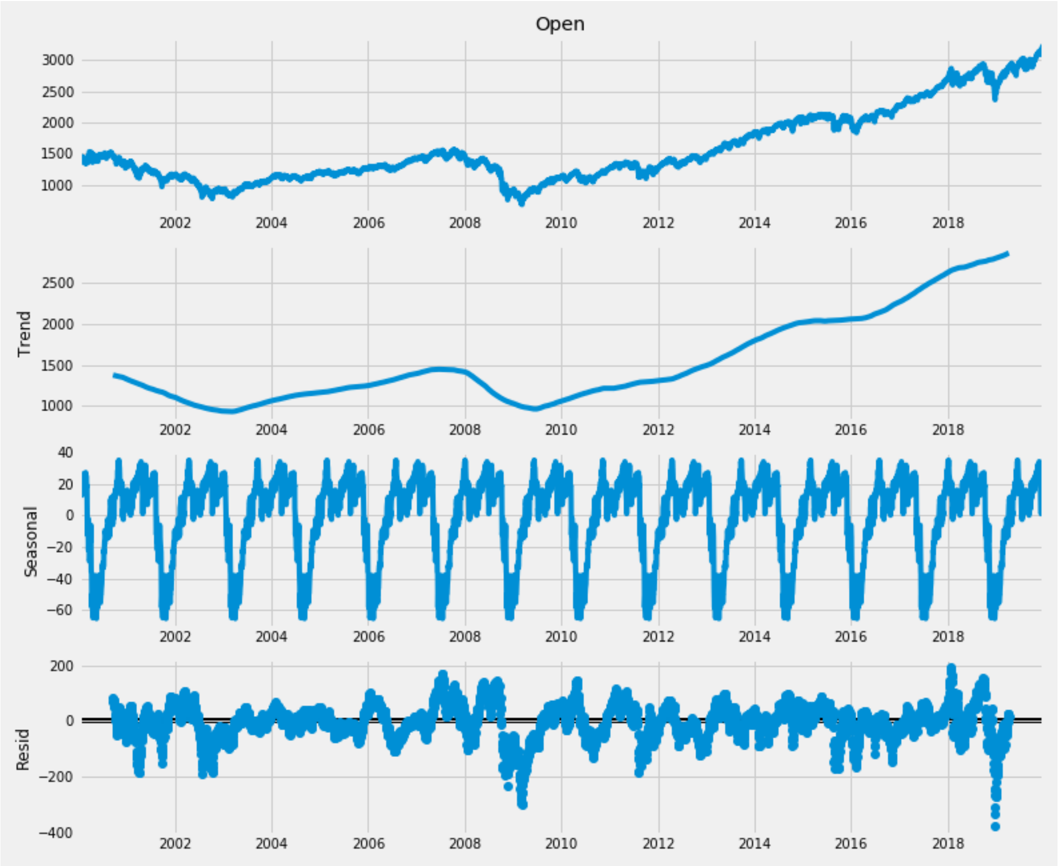


Figure 2: S&P 500 **Open** Moving Average Decomposition between 2000-2019.

In order to apply *Linear Time Series* methods, the time serie should be at least stationary of order 2. To do so, we will take the log returns of our time serie. Precisely, if we denote (p_t) the daily Open values of S&P 500, the log returns are :

$$L_t = log(p_t) - log(p_{t-1})$$

This transformation leads (most of times) to a time series which is stationary (almost) of second order. We plotted our log returns in **Figure 3**

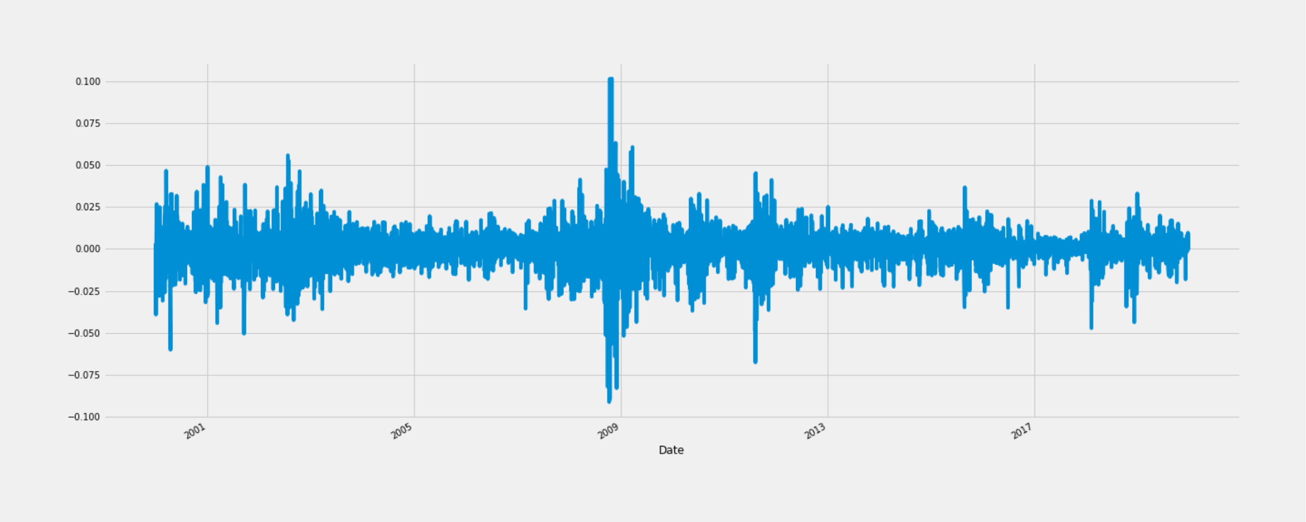


Figure 3: Log stock returns (for Open market values) of SP 500 index from 2000 to 2019.

This Time Series seems to be better to analyze than the previous one. Although it appears to be a white noise, it is not. The variance is clearly non-constant. Naively, we plotted in **Figure 4** the variations of the *deviation* of (L_t) on a sliding window of size 150 days. We can notice periods of **high volatility** such as 2007-2008 (Economic crisis).

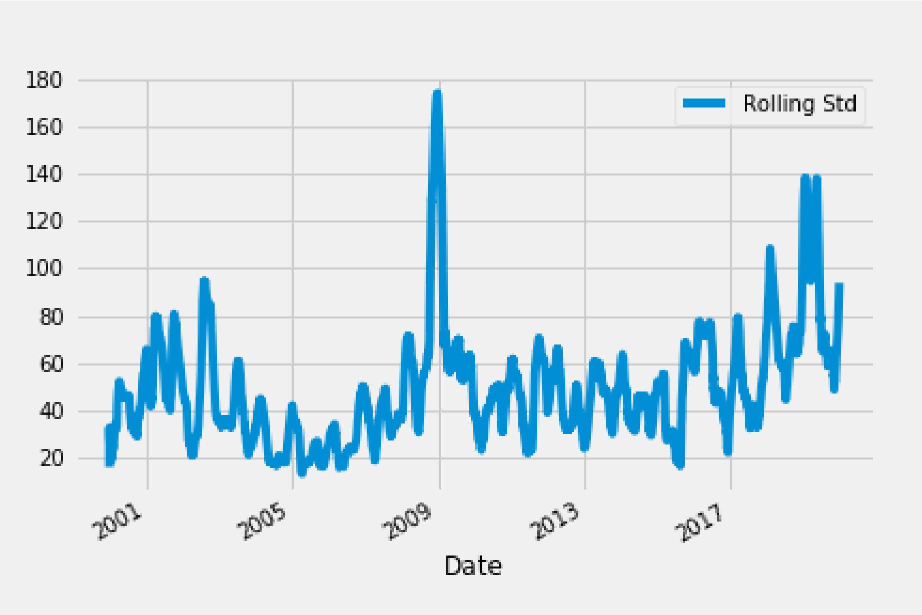


Figure 4: The evolution of standard deviation over time between 2000 and 2019 for the S&P 500.

Statistical Analysis of the log returns (L_t)

In this section, we will analyze (L_t) using the ACF and PACF measures discussed in lectures. More precisely, we will compute, for (L_t), the following:

- *Autocorrelation (ACF)*: Correlations between lags of (L_t).
- *Partial Autocorrelation (PACF):* Correlation between lags without considering the effects of inter-lags (by substracting the linear projection of each lag with respect to all inter-lags).

In **Figure 5**, we plotted these quantities (second row) for the first 30 lags. We have also compared the distribution of L_t to a normal distribution using the Q-Q plot test (Third row).

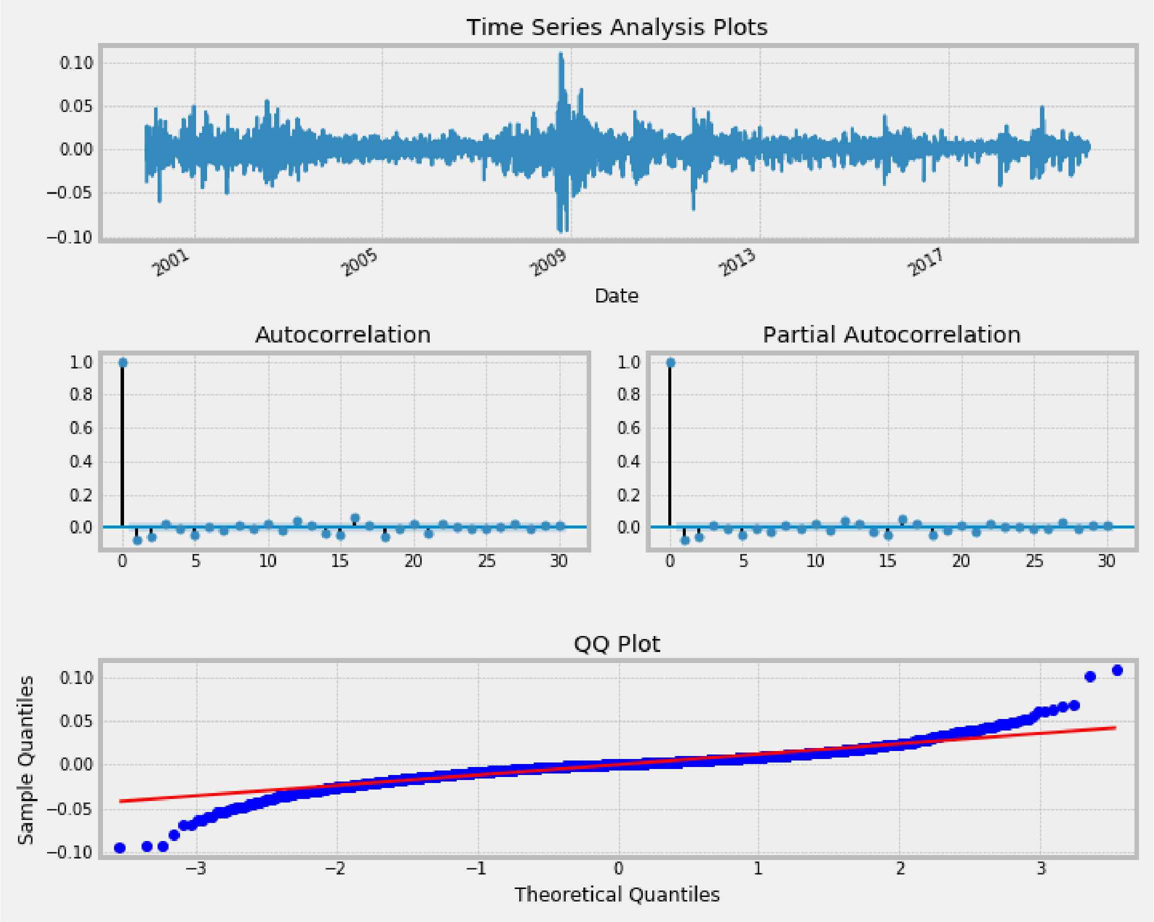


Figure 5: ACF and PACF quantities for the first 30 lags of L_t and QQ plots against a normal distribution.

Now we can fit Time Series models on L_t. In fact, there is no PACF nor ACF for lags superior to 1.
We can also notice the presence of the **Heavy tails** phenomenon in L_t. As discussed in lectures, L_t have more *extreme events* than a simple normal distributions.

3 Application of Autoregressive Integrated Moving Average Models ARIMA(p, d, q)

In this section we will try to fit an ARIMA process on our log returns and see the results on residuals. In fact, a model that explains well the data is a model which gives **white noise residuals**.

3.1 ARIMA Models

An ARIMA(p, d, q) assumes the following stochastic process for L_t:

$$(1-L)^d L_t = \mu + a_1(1-L)^d X_{t-1} + ... + (1-L)^d X_{t-p} + \epsilon_t - \theta_1 \epsilon_{t-1} - ... - \theta_q \epsilon_{t-q}$$

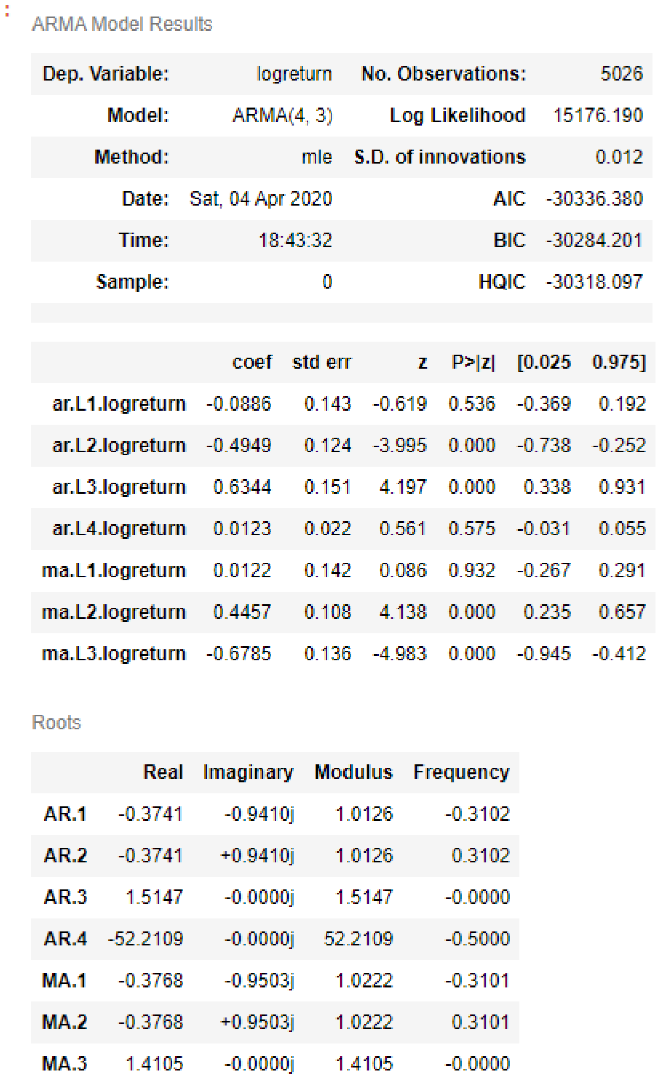
where $p \neq 0$, $\theta_q \neq 0$ and (ϵ_t) is a weak white noise. L is the classical Lag operator. An ARIMA(p, d, q) process gives an ARMA(p,q) process when differentiating d times.

In our estimations, we will fit ARIMA models using the Exact maximum likelihood via Kalman filter.

3.2 Model Selection

The idea is to fit different non-trivial combinations of (p,d,q) on our data and to pick the model which minimizes tha **AIC** measure.
Precisely, we fitted models for $p \in \{0,1,2,3,4\}, q \in \{0,1,2,3,4\}$ and $d \in \{0,1\}$. We obtained that $(\mathbf{p}, \mathbf{d}, \mathbf{q}) = (\mathbf{4}, \mathbf{0}, \mathbf{3})$ is the model that minimizes **AIC** for our data (with value $-\mathbf{30336.38}$)

Remark: Since we are already working on the log returns, it is obvious that the parameter d should be equal to 0 since L_t is already almost stationary (of order 2).
We plotted in **Figure 6** the training results for ARIMA(4,0,3).



ARMA Model Results

Dep. Variable:	logreturn	**No. Observations:**	5026
Model:	ARMA(4, 3)	**Log Likelihood**	15176.190
Method:	mle	**S.D. of innovations**	0.012
Date:	Sat, 04 Apr 2020	**AIC**	-30336.380
Time:	18:43:32	**BIC**	-30284.201
Sample:	0	**HQIC**	-30318.097

	coef	std err	z	P>|z|	[0.025	0.975]
ar.L1.logreturn	-0.0886	0.143	-0.619	0.536	-0.369	0.192
ar.L2.logreturn	-0.4949	0.124	-3.995	0.000	-0.738	-0.252
ar.L3.logreturn	0.6344	0.151	4.197	0.000	0.338	0.931
ar.L4.logreturn	0.0123	0.022	0.561	0.575	-0.031	0.055
ma.L1.logreturn	0.0122	0.142	0.086	0.932	-0.267	0.291
ma.L2.logreturn	0.4457	0.108	4.138	0.000	0.235	0.657
ma.L3.logreturn	-0.6785	0.136	-4.983	0.000	-0.945	-0.412

Roots

	Real	Imaginary	Modulus	Frequency
AR.1	-0.3741	-0.9410j	1.0126	-0.3102
AR.2	-0.3741	+0.9410j	1.0126	0.3102
AR.3	1.5147	-0.0000j	1.5147	-0.0000
AR.4	-52.2109	-0.0000j	52.2109	-0.5000
MA.1	-0.3768	-0.9503j	1.0222	-0.3101
MA.2	-0.3768	+0.9503j	1.0222	0.3101
MA.3	1.4105	-0.0000j	1.4105	-0.0000

Figure 6: ARIMA(4,0,3) model with lowest AIC measure.

Let's try now to analyse the residuals of our model. We plot PACF and ACF for residuals in Figure 7.

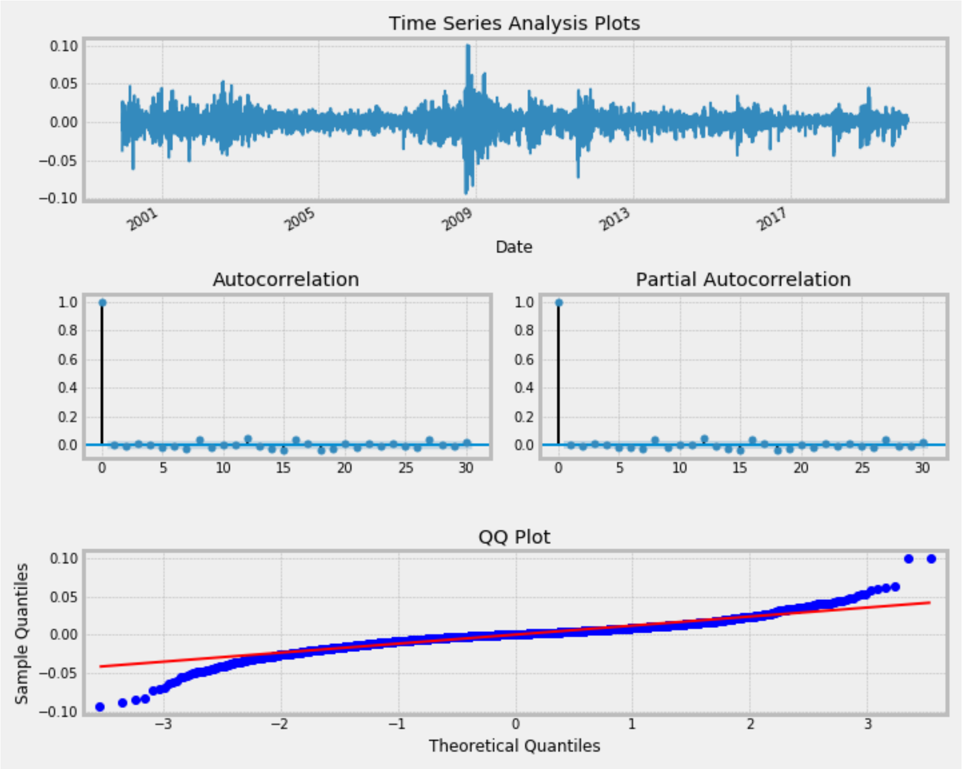


Figure 7: Residuals of ARIMA(4,0,3) when fitted on our data from 2000 to 2019 included.

We can notice some non-neglictible autocorrelations (lag 15 for example).
As discussed in the course, this graph is misleading and we should also analyze the **square** o residuals (see next section) to verify whether ARIMA deals with the **Conditional Variance** issue.

4 Application of Generalized Autoregressive Conditionally Heteroskedastic Models GARCH(p,q)

4.1 Motivation

As discussed in the previous section, we found that residuals looks like a descrete white noise. Now, we will look at the **Square of residuals** and see whether they also look like a white noise. **Figure 8** presents the results for ARMA(4,3) model.

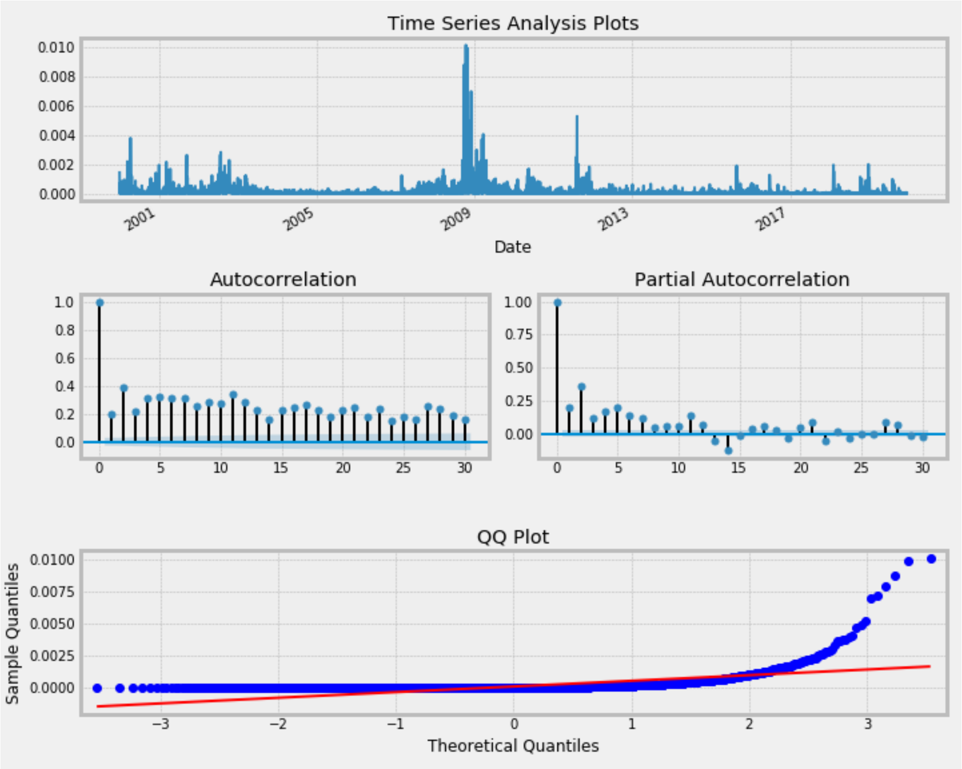


Figure 8: Square of Residuals of ARIMA(4,0,3) when fitted on our data from 2000 to 2019 included.

Now, we can see strong evidence of conditional volatility. This proves that ARIMA models could not cope with **Volatility Clustering** issues as the residuals have autocorrelations when squared.
This justifies the use of **GARCH** model.

4.2 Definition of GARCH(p,q) model

Let's denote $(\eta_t)_t$ an iid random variables such as : $\mathbf{E}(\eta_t) = 0$ and $\mathbf{Var}(\eta_t) = 1$. GARCH model assumes that

$$L_t = \sigma_t \eta_t$$

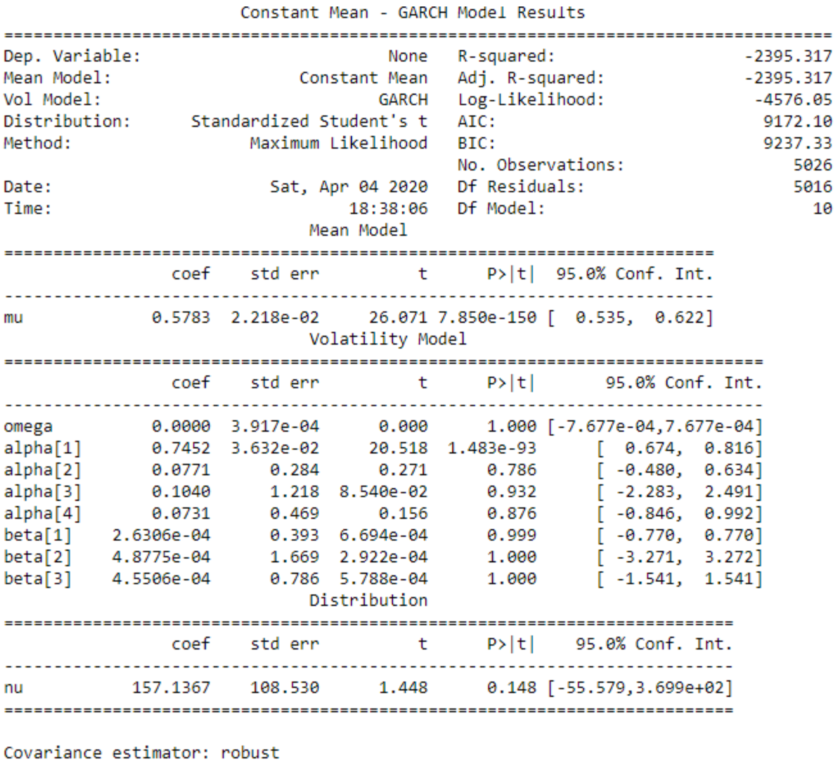
where

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

with $w \geq 0$, $\beta_j \geq 0$, $\alpha_i \geq 0$.

4.3 Model Selection

As GARCH model is well adapted to our case. We will use it along with ARIMA model to explain the variations of L_t.
As discussed in lectures, we will take a GARCH model with the same parameter as the best ARIMA model. Hence, we will train a GARCH model on the residuals of our ARIMA model and see if it could explain them. **Figure 9** plots the results of training.



```
                     Constant Mean - GARCH Model Results
==================================================================================
Dep. Variable:                      None   R-squared:                   -2395.317
Mean Model:                Constant Mean   Adj. R-squared:              -2395.317
Vol Model:                         GARCH   Log-Likelihood:               -4576.05
Distribution:   Standardized Student's t   AIC:                           9172.10
Method:               Maximum Likelihood   BIC:                           9237.33
                                           No. Observations:                 5026
Date:                   Sat, Apr 04 2020   Df Residuals:                     5016
Time:                           18:38:06   Df Model:                           10
                              Mean Model
========================================================================
                 coef    std err          t       P>|t|  95.0% Conf. Int.
------------------------------------------------------------------------
mu             0.5783  2.218e-02     26.071 7.850e-150 [  0.535,  0.622]
                               Volatility Model
=============================================================================
                 coef    std err          t       P>|t|       95.0% Conf. Int.
-----------------------------------------------------------------------------
omega          0.0000  3.917e-04      0.000       1.000 [-7.677e-04,7.677e-04]
alpha[1]       0.7452  3.632e-02     20.518   1.483e-93      [  0.674,  0.816]
alpha[2]       0.0771      0.284      0.271       0.786      [ -0.480,  0.634]
alpha[3]       0.1040      1.218  8.540e-02       0.932      [ -2.283,  2.491]
alpha[4]       0.0731      0.469      0.156       0.876      [ -0.846,  0.992]
beta[1]    2.6306e-04      0.393  6.694e-04       0.999      [ -0.770,  0.770]
beta[2]    4.8775e-04      1.669  2.922e-04       1.000      [ -3.271,  3.272]
beta[3]    4.5506e-04      0.786  5.788e-04       1.000      [ -1.541,  1.541]
                              Distribution
==========================================================================
                 coef    std err          t       P>|t|     95.0% Conf. Int.
--------------------------------------------------------------------------
nu           157.1367    108.530      1.448       0.148 [-55.579,3.699e+02]
==========================================================================

Covariance estimator: robust
```

Figure 9: GARCH(4,0,3) fitted on our data from 2000 to 2019 included.

We can notice that the square of residuals now looks like a discrete white noise (See **Figure 10**) and hence we have succeeded to deal with **Volatility Clustering** problem using this GARCH modeling.

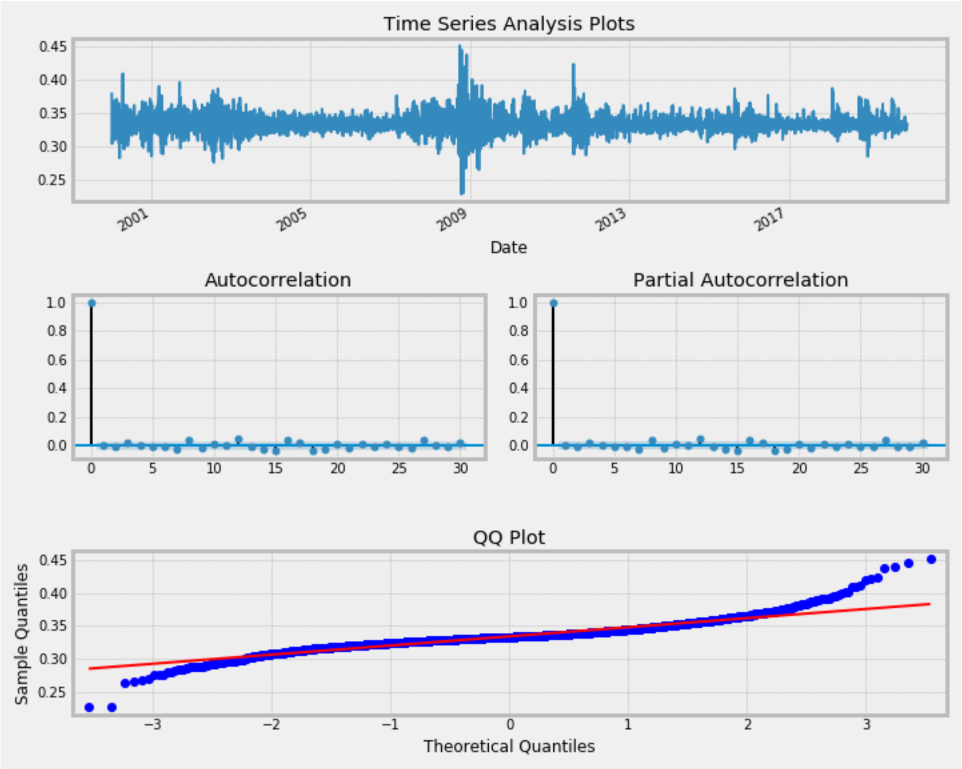


Figure 10: Square of residuals of GARCH(4,0,3) fitted on the residuals of ARIMA(4,0,3) between 2000 and 2019.

5 A simple trading strategy using ARIMA and GARCH

We will try to exhibit a simple trading strategy using the predictions of ARIMA and GARCH described above. To do so, we will do the following:

- Fix T=250 days as a length of a sliding window
- Find the best ARIMA model in the sliding window
- Fit the corresponding GARCH model using the methodology described above
- Predict the next value of stock returns using GARCH
- If positive returns: Buy / Else: Sell

We launched and compared the returns of our strategy with a naive **Buy and Hold** strategy for different timespans. Our conclusion is that ARIMA-GARCH strategy outperformed the naive one in periods of high volatility such as the economic crisis (2007-2008) and performed poorly during recent relative stable periods such as 2019. This is due to the use of GARCH model which captures well the conditional volatility. **Figures 11 and 12** present the returns of the backtested stategies (during two different timespans).

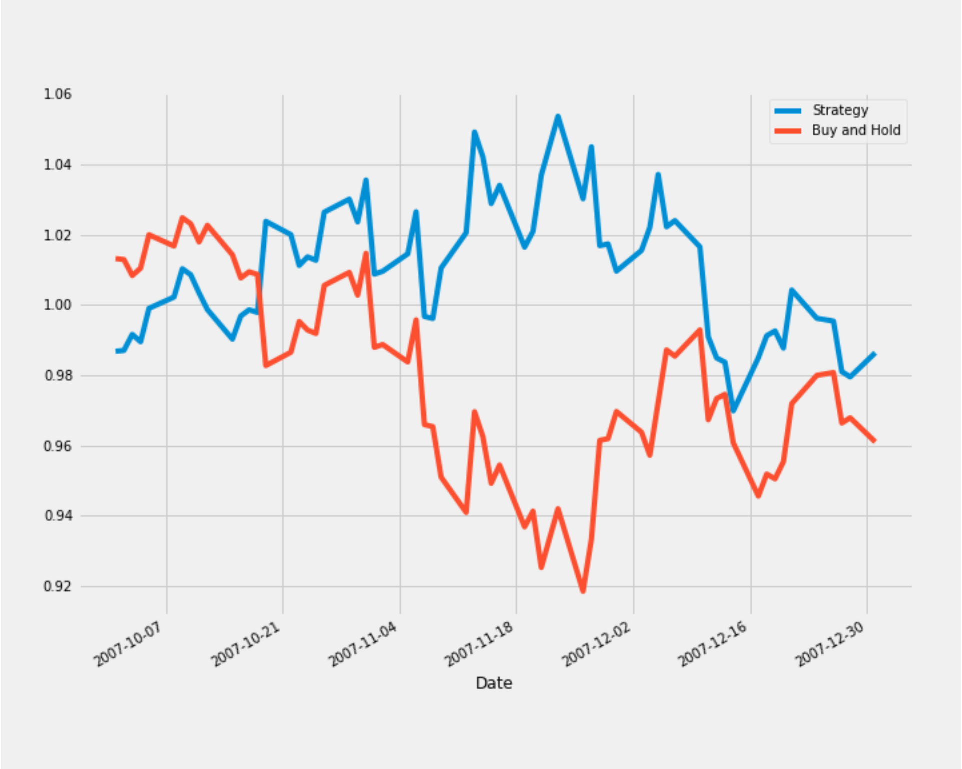


Figure 11: The strategy ARIMA-GARCH comparey to Buy and Hold during the volatile period 2007. ARIMA-GARCH forcasting appears to be efficient in periods of high volatility.

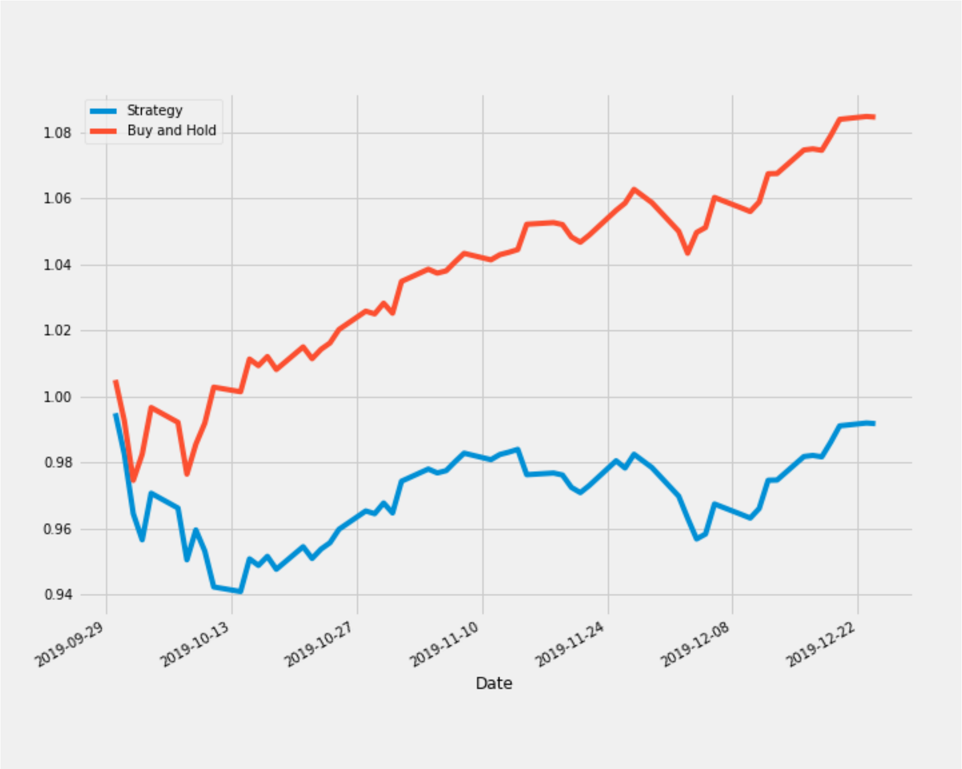


Figure 12: The strategy ARIMA-GARCH comparey to Buy and Hold during late 2019. Buy and Hold strategy outperform ARIMA-GARCH.

6 Hawkes Process to model Extreme Trades' Volumes Arrivals

6.1 Motivation and Data

It is known in finance that trades do not arrive in evenly spaced intervals but as clusters over time. There are many reasons that could explain this, such as the fact that traders split up their order to smaller blocks or the reactions of a market to an event or a certain exchange event. Our first idea was to model Transaction's dates arrivals using Hawkes process, but we haven't found intraday data for any stock freely in the net.
In this section, we will rather model the arrivals of the event : **Volume of transactions of S&P 500 in a day is extreme**. This is motivated by the observation that *higher* volumes transactions come as clusters in the market. For example, in the economic crisis 2008-2009 the market entered in panic mode and the volumes of transactions increased highly for all this period.

Data

In our analysis, we will use only Volumes data of S&P 500 between 2011-2019. An event is considered as extreme when **Volumes are higher than the quantile of order 90 of our data**. **Figure 13** illustrates the arrivals of these events from the starting date: 2011-01-03.

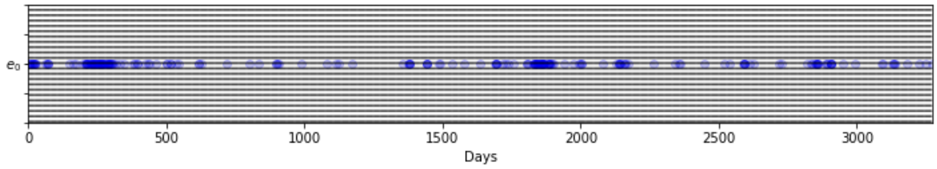


Figure 13: Extreme Volumes arrivals starting 2011-01-03

Figure 14 illustrates the variation of volumes between 2011 and 2019 (The horizontal line is the quantile(90)

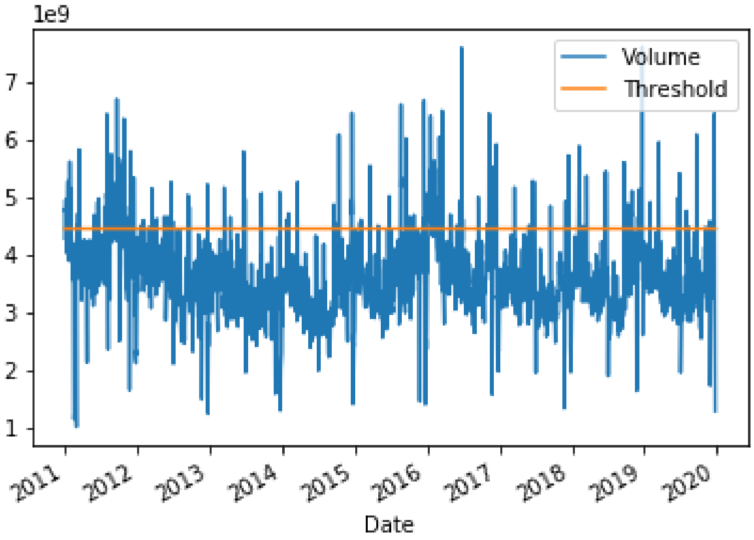


Figure 14: Variation of Volumes between 2011-2019.

6.2 Hawkes Process

Hawkes process models the time-varying event occurrence of a process, which is partially determined by the history of the process. In our study, we use a hawkes process with **exponential** kernel. Precisely, we train a Hawkes model with itensity:

$$\lambda(t) = \mu + \sum_{t_i < t} \alpha\beta \exp -\beta(t - t_i)$$

where t_i are the timestamps of events that occurred before t, α is the branching ratio (or the endogenity of the process) which describes the fraction of events that are endogenously generated (results of previous occurrences), $\beta \geq 0$ is the exponential decay and μ is the base rate of the process.

6.3 Fitting Extreme Volumes Arrival to a Hakwes Process

After each occurrence, a hawkes process follows a Inhomogoneous Poisson law of parameter $\lambda(t)$. This could be siumalted by the famous Thinning Ogata Algorithm (which we used in our simulations). For the fitting with our data, we used a maximum of likelihood approach.

Note that β is a hyper-parameter in our fitting process. We chose it using cross-validation. The fitting gave the following parameters: $\mu = 0.03842521$, $\alpha = 0.44121204$ and $\beta = 0.23$.

They can be interpreted as follows:

- The baseline of extreme Trades' volumes is $\mu = 0.04$ events per day.
- Each event prior to t contributes of $\alpha\beta = 0.1$ in the intensity just after in its occurence. This contribution decrease exponentially with a rate $\beta = 0.23$
- The average intensity over the whole period is $\mathbf{E}(\lambda) = \frac{\mu}{1-\alpha} = 0.06$ events per day.

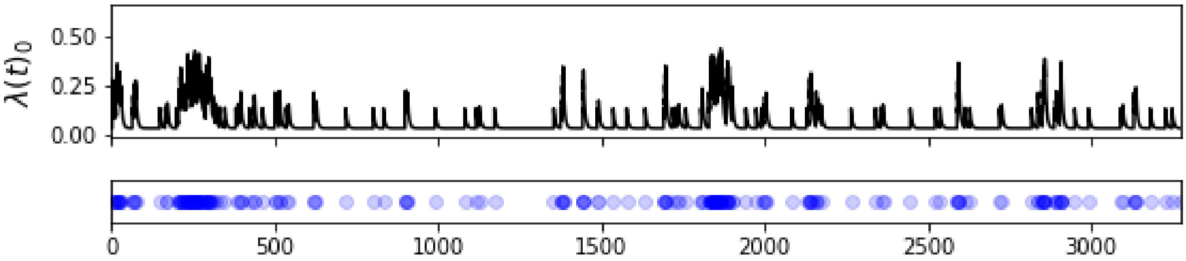


Figure 15: Intensity of the fitted Hawkes model for data between 2011-2019. **Parameters:** $\mu = 0.03842521$, $\alpha = 0.44121204$ and $\beta = 0.23$

6.4 Goodness-of-fit

In order to test the goodness-of-fit of our hawkes process regarding our data, we will use the Ogata's residuals analysis.
In fact, this method calls $\{\hat{t_n} = \int_0^{t_n} \lambda(s)ds = \mu - \alpha \sum_{t_i<t_n} e^{-\beta(t_n-t_i)} - 1\}$ the residuals process. According to the random time change theorem, the residual process should be close to a standard Poisson process if the estimated intensity $\lambda(t)$ is close to the true intensity. This can be tested by comparing the distribution of $\{\tau_n = \hat{t_n} - \hat{t_{n-1}}\}$ to an Exp(1) distribution.
We implemented these quantities and used the Q-Q plot to compare the distributions. We found also that $R^2 = 0.98$ for our estimated model.

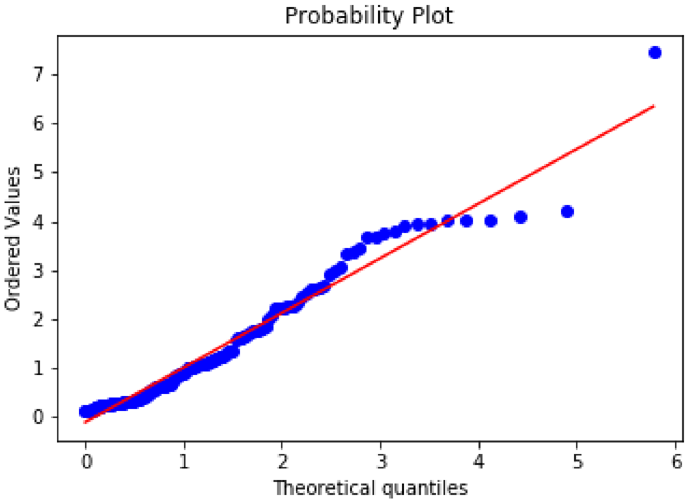


Figure 16: Q-Q plot of τ_n residuals (using our best model) against Exp(1) distribution. We found also $R^2 = 0.98$ for the regressions used in QQ plot.