1 Models

In this section we will describe the models that have been implemented throughout the process and highlight the changed and simplifications. To make a framework for model development we created a metaclass, so called BaseModel, where we created all the common functions that all these models share. We manage to implement a strategy to create bounds with existing parameter's transformation. Not only give us a unique opportunity to modeling with various parameter transformations, but in optimizations brought as a more stable results. The optimizer that we use the Python's scipy.optimize.minimize function with the setting of the L-BFGS-B method.

1.1 MIDAS

The first model was the MIDAS that was first introduced [**?**], where they compare distributed lag models with MIDAS regression. The MIDAS model can be described with the expressions used in [**?**]. Suppose y_t is the low-frequency dependent variable that can be observed once within a time-step t (say, monthly), then $x_t^{(m)}$ the high-frequency explanatory variable can be observed m times during one time-step (say, daily or m = 22). We want to describe the relationship between y_t and $x_t^{(m)}$, in the sense of using lagged observations of $x_t^{(m)}$. The model is the following:

$$y_t = \beta_0 + \beta_1 B(L^{\frac{1}{m}}, \theta) x_t^{(m)} + \epsilon_t^{(m)} \tag{1}$$

where $B(L^{\frac{1}{m}}, \theta) = \sum_{k=0}^{K} B(k, \theta) L^{\frac{k}{m}}$, where $L^{\frac{k}{m}}$ is a lag operator such that $L^{\frac{1}{m}} x_t^{(m)} = x_{t-\frac{1}{m}}^{(m)}$. The lag coefficients in $B(k, \theta)$ of the corresponding lag operator $L^{\frac{k}{m}}$ are parameterized as a function of a small-dimensional vector of parameters Θ. β_1 is a scale parameter for the lag coefficients

1.1.1 Specification of Weighting Function

In the MIDAS literature there is one weighting function that used the most, namely "Beta" Lag. [**???**]. For completeness, I mention the others, these are the Exponential Weighting and the Exponential Almon Lag. Beta Lag involves two parameters, $\Theta = (\theta_1, \theta_2)$, and the parametrization:

$$B(k, \theta_1, \theta_2) = \frac{f(\frac{k}{K}), \theta_1, \theta_2)}{\sum_{k=1}^{K} f(\frac{k}{K}), \theta_1, \theta_2)} \tag{2}$$

where

$$f(x, a, b) = \frac{x^{a-1}(1-x)^{b-1}\Gamma(a+b)}{\Gamma(a)\Gamma(b)} \tag{3}$$

$$\Gamma(a) = \int_0^\infty e^{-x} x^{a-1} dx \tag{4}$$

The following figure will deonstrate how flexiable it is correspond to different parameters:

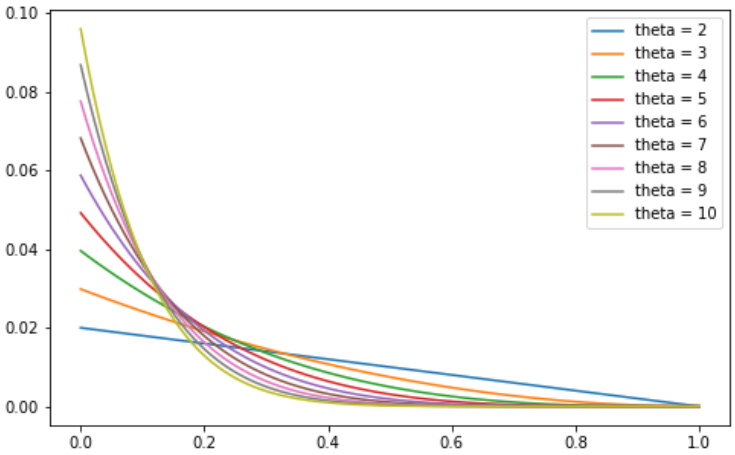

Figure 1: Plot of Beta Lag weighting function in equation **??** with K = 100, $\theta_1 = 1$ and $\theta_2 = 2, ..., 10$

We can see that if we choose to fix $\theta_1 = 1$ and in the case of $\theta_2 > 1$ cause a monoton decliyin weighting structure. This weight function specification provide us positive coefficients, which is crutual when we want to modeling volatility.

1.1.2 Parameter Transformation

In this section we will describe an approach to make parameter estmation more consistant and stabil, it is so called parameter transformation. The main idea behind this strategy is that estimators can treat bounds, but in practice it is much more convenient to transform our parameters. With this approach we can create bounds without explicitly programming to the estimator function. First we describe the transform and the back-transform function, then show how they incoperate to the function that will be estimated. Let denote θ with the parameter that we want to work with:

$$\tilde{\theta} = \begin{cases} \log(\theta) & \text{,if 'pos'} \\ \log(\theta) - \log(1-\theta) & \text{,if '01'} \\ \theta & \text{otherwise.} \end{cases}$$

$$\theta = \begin{cases} \exp(\tilde{\theta}) & \text{,if 'pos'} \\ \frac{1}{1+\exp(-\tilde{\theta})} & \text{,if '01'} \\ \tilde{\theta} & \text{otherwise.} \end{cases}$$

In the log likelihood function instead of calculting with the actual θ, we will make the estimation with θ. Than we transform back as the estimation finished. One issue raise from this estimation strategy, is that the standard error won't be correct. So there is another function called gradient. θ^* marked as the estimated parameters that were previously transformed.

$$gradient = \begin{cases} \exp(\theta^*) & \text{,if 'pos'} \\ \frac{\exp(\theta^*)}{(1+\exp(\theta^*))^2)} & \text{,if '01'} \\ 1 & \text{otherwise.} \end{cases}$$

As the L-BFGS-B method relies ont he approximation to the Hessian matrix of the loss function, se as we take advantage of information matrix equality we can calculate the standard errors easily.

1.1.3 Parameter Estimation

In the parameter estimation we will use the Python's function from scipy.optimize library, called minimize. I applied L-BFGS-B method, this method allow us to define bounds for parameters, and the biggest advantage is that approximate the inverse Hessian matrix. The estimation is happening throughout the sum of squared estimat of error:

$$SSE = \epsilon^T \epsilon = \sum_{t=1}^{T} (y_t - \beta_0 - \beta_1 B(L^{\frac{1}{m}}, \theta) x_t^{(m)})^2 \tag{5}$$

$$\arg \min_{\beta_0, \beta_1, \theta_2} SSE$$

1.1.4 Simulations

The Monte-Carlo simulation was from [**?**]. Suppose we have X_t is an AR(1) process, that:

$$X_{i,t} = \phi X_{i-1,t} + \epsilon_t$$

where t = 1, ..., T show the low-frequency time-steps, i = 1, ...,I_t is the high-frequency. Set the I_t equals to 22, $\phi = 0.9$ and $\epsilon_t \sim \mathcal{N}(0, 1)$ standard normal variable. The MIDAS equation will be:

$$y_t = \beta_0 + \beta_1 \sum_{k=0}^{K} \xi_k(1.0, \theta) X_{i-k,t} + z_t$$

with the parameters $\beta_0 = 0.1, \beta_1 = 0.3, \theta = 4.0$ and $z_t \sim \mathcal{N}(0, 0.5)$. We made simulations with T = 100, 200, 500. A A simulare approach was described in [**?**] with the difference of simulating quarterly/monthly data and they found out, as they increase the sample ssize the more accurate their parameter estimations will be, furthermore the more parsimonious will be the model's computational cost.

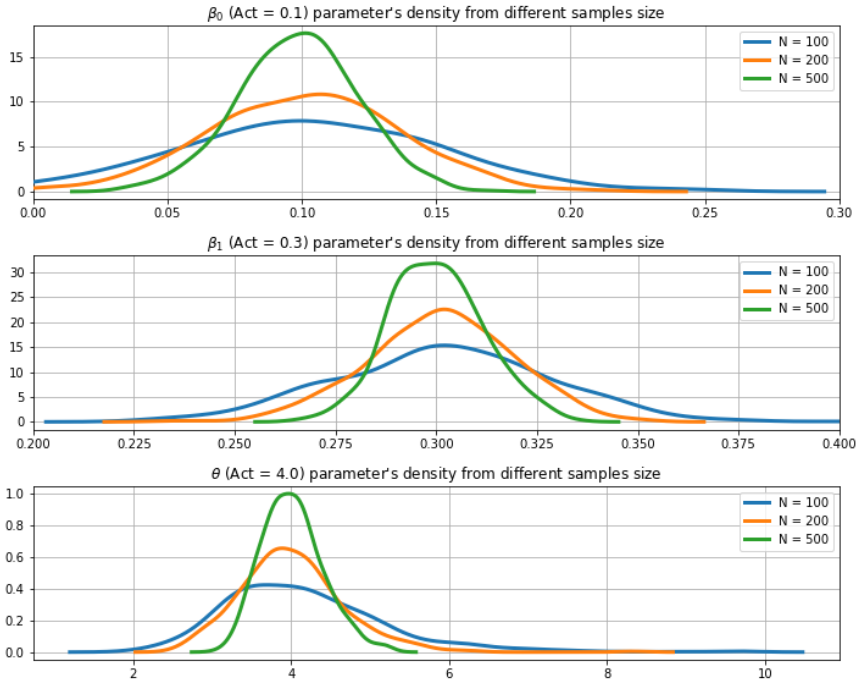

Figure 2: Plot of estimated parameter distributions with sample sizes of 100, 200, 500

1.2 Generalized Autoregressive Conditional Heteroscedasticity

In this section we would like to specify the vanilla GARCH(1, 1) model. First we assign the r_t to the daily log return ($r_t = \log P_t - \log P_{t-1}$, where P_t is the stock price at time t) for t = 1, ..., T. Assume, that the conditional mean of the returns are contants:

$$r_t = \mu + \epsilon_t \tag{6}$$

where ϵ_t denote a real-valued discrete-time stochastic process and $\mathcal{F}_t$ the information set of all information through time t. [**?**]

$$\epsilon_t \mid \mathcal{F}_{t-1} \sim \mathcal{N}(0, \sigma_t^2) \tag{7}$$

Than the GARCH(1, 1) process

$$\sigma_t^2 = \alpha_0 + \alpha_1 \epsilon_{t-1}^2 + \beta_1 \sigma_{t-1}^2 \tag{8}$$

where $\alpha_0 > 0, \alpha_1 \geq 0, \beta_1 \geq 0$ and $1 > \alpha_1 + \beta_1$ enough wide-sense stacionarity.

1.2.1 Parameter Estimation

The estimation happens through maximum likelihood estimation. Let $\theta \in \Theta$, where $\theta = (\mu, \alpha_0, \alpha_1, \beta_1)$ and Θ is a compact subspace of an Euclidean space such that ϵ_t process finite second moments. The loglikelihood function for a sample of N observation is:

$$l_t(\Theta) = -\frac{1}{2} log 2\pi - \frac{1}{2} \log \sigma_t^2 - \frac{1}{2} \frac{\epsilon_t^2}{\sigma_t^2} \tag{9}$$

$$L_N(\Theta) = \frac{1}{N}\sum_{t=1}^{N} l_t(\Theta)$$

$$\arg\min_{\Theta} -L_N(\Theta)$$

1.2.2 Simulations

We preform Monte-Carlo simulations for GARCH(1, 1) model with different sample sizes. The generation happens through

$$\epsilon_t \sim \mathcal{N}(0, \sigma_t^2) \tag{10}$$

For t = 1, ..., T. The equation means, that we generate an ϵ_t every step with the current state of σ_t^2. We expect that as we increase the sample size, the more accurate estimation we get. The following figure will show the results of the simulation, where we apply the kernel density estimation process in order to get a smoother plot:

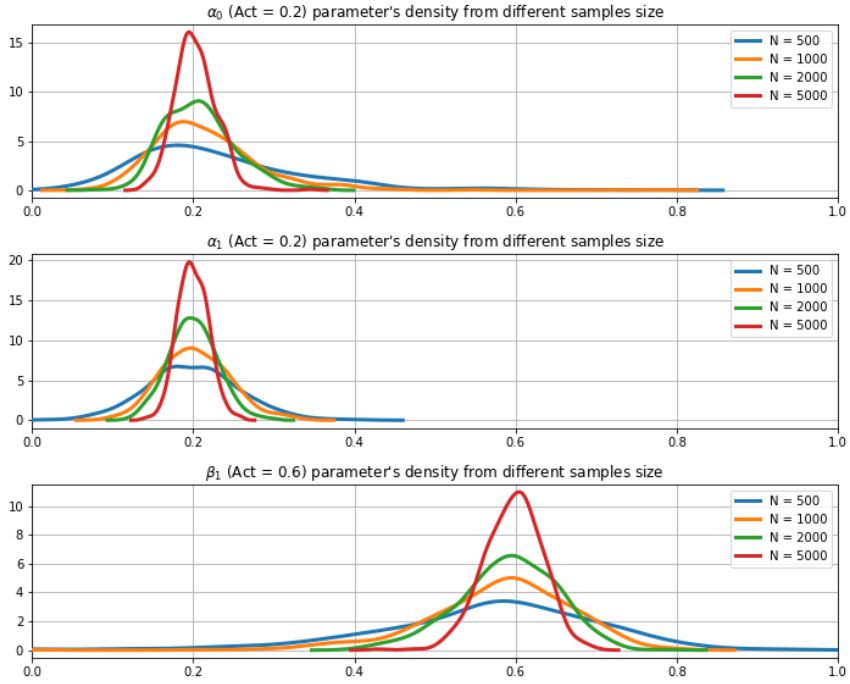

Figure 3: Plot of estimated parameter distributions with sample sizes of 500, 1000, 2000

As we expected, the parameter estimation get better and better, when we increase the sample size.

1.3 GARCH-MIDAS

This GARCH-MIDAS framework give us the opportunity to incorporate macroeconomic variables sampled at different frequency. In the recent years many study examined the effectiveness of this approach. In this model class we can utilies both models strengths. Let r_t the daily log return for t = 1, ..., T period (say

monthly) and i = 1, ..., I_t (say, daily or $I_t = 22$) to the days within that period. Assume the daily log return on i in t follows:

$$r_{i,t} = \mu + \epsilon_{i,t} \tag{11}$$

$$\epsilon_{i,t} = \sqrt{g_{i,t}\tau_t} Z_{i,t} \tag{12}$$

where $\epsilon_{i,t} \mid \mathcal{F}_{i-1,t} \sim \mathcal{N}(0, g_{i,t}\tau_t)$ with $\mathcal{F}_{i-1,t}$ is the information set up to day i-1 of period t. We assuem the short-term volatility component $g_{i,t}$ is a GARCH(1, 1) process, as **?** described:

$$g_{i,t} = (1 - \alpha - \beta) + \alpha \frac{\epsilon^2_{i-1,t}}{\tau_t} + \beta g_{i-1,t} \tag{13}$$

and define the long-term volatility component τ_t in the spirit of MIDAS regression:

$$\tau_t = \beta_0 + \sum_{j=1}^{M} \beta_j \sum_{k=0}^{K} \phi_k(1.0, \theta_j) X^{(j)}_{t-k} \tag{14}$$

where $X^{(j)}_{t-k}$ referes to the j-th macroeconomic variable. τ_t is constant over the periods. As macroeconomic variables can be positive or negative values, we change the specification of tau_t as [**?** and **?**] previously did it in the following way:

$$\log \tau_t = \beta_0 + \sum_{j=1}^{N} \beta_j \sum_{k=0}^{K} \phi_k(1.0, \theta_j) X^{(j)}_{t-k} \tag{15}$$

1.3.1 Parameter Estimation

GARCH-MIDAS estimation made by maximum likelihood estimation, where the parameters for the two volatility component estimated by one-step. For our implementation and backtest raise some identification issue. The Loglikelihood function is:

$$\log L_N(\Theta) = \frac{1}{N} \sum_{j=1}^{N} \frac{1}{I_t} \sum_{i=1}^{I_t} \frac{1}{2} \log 2\pi + \frac{1}{2} \log g_{i,t}\tau_t + \frac{1}{2} \frac{\epsilon^2_{i,t}}{g_{i,t}\tau_t} \tag{16}$$

$$\arg\min_{\Theta} \log L_N(\Theta)$$

1.3.2 Simulations

The simulation is based totally on [**?**]. It was a good base for us to compare our results with the one, that they got. We want to mention, that we not only examine the case of T = 480, but run the simulation with T = 240, 960. The reason behind, in modeling situations we don't have that much data.

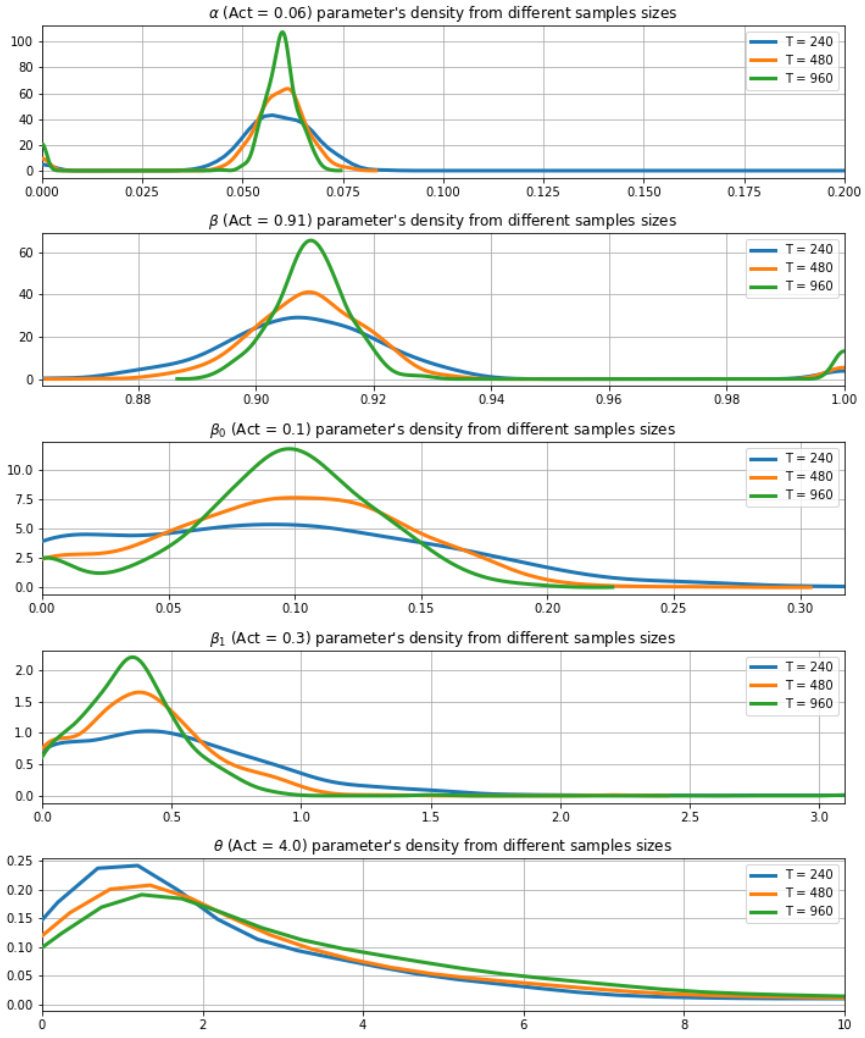

Figure 4: Plot of estimated parameter distributions with sample sizes of T = 100, 200, 500

In this figure you can see, that as we decrease T to 240, the estimated MIDAS parameters are far from the theoretical ones. One of the reason we found, that as we want to optimized the product of $g_{i,t}$ and τ_t identification issues arised. In the following sections we will give an other estimation strategy for this problem.

1.4 Panel MIDAS

Let $r_{i,t}^{(j)}$ is the j-th stock's log return for j = 1, ..., M, where M is the number of stocks. We sign month index by t = 1, ..., T, the length of sequence is T and days in the t-th month sign with i = 1, ..., I_t. We assume that these stocks share the same underlying long-term volatility component, that will be marked as τ_t. τ_t is

the t-th month's level of volatility whick can be explained with macroeconomic variables, so the equation:

$$\tau_t = \beta_0 + \sum_{m=1}^{N} \beta_m \sum_{k=1}^{K_m} \phi_k(1.0, \theta_m) X_{t-k}^{(m)} \tag{17}$$

where N is the number of macroeconomic variables, K is the number of lag. For simplicity we choose $K \equiv K_m$. 1 + N * 2 parameters will be estimated.

1.4.1 Parameter Estimation

The parameter estimation happens through maximum likelihood estimation as we did previously, where the loglikelihood function will connent the log returns and the long-term volatility component:

$$l_t^{(j)}(\Theta) = \frac{1}{2}\log 2\pi + \frac{1}{2}\tau_t + \frac{1}{2}\frac{(r_{i,t}^{(j)})^2}{\tau_t} \tag{18}$$

We sum them up:

$$\log l(\Theta) = \sum_{j=1}^{M}\sum_{t=1}^{T} -\frac{1}{2}\log 2\pi + \frac{1}{2}\tau_t + \frac{1}{2}\frac{(r_{i,t}^{(j)})^2}{\tau_t} \tag{19}$$

then minimize the value:

$$\arg\min_{\Theta} \log l(\Theta) \tag{20}$$

1.4.2 Simulations

The simulation was conducted in the spirit of MIDAS simulation with same changes. Let suppose we have one explanatory variable that define the volatility say X_t is an AR(1) process:

$$X_t = \phi X_{t-1} + \epsilon_t \tag{21}$$

where t = 1, ..., T, $\phi = 0.9$ and $\epsilon_t \sim \mathcal{N}(0, 1)$ standard normal variable, than the MIDAS model will be:

$$\log \tau_t = \beta_0 + \beta_1 \sum_{k=0}^{K} \xi_k(1.0, \theta) X_{t-k} \tag{22}$$

where $\beta_0 = 0.1, \beta_1 = 0.3$ and $\theta = 4.0$.τ_t remains the same throughout the whole period. The τ_t will determine the return's volatility, the returns are generated from normal distribution with zero mean and τ_t variance:

$$r_{i,t} \sim \mathcal{N}(0, \tau_t) \tag{23}$$

as τ_t is set to be a monthly variable, we generate daily return, so i mark mean that the i-th day of t-th month, i = 1, ..., I_t, where $I_t = 22$.

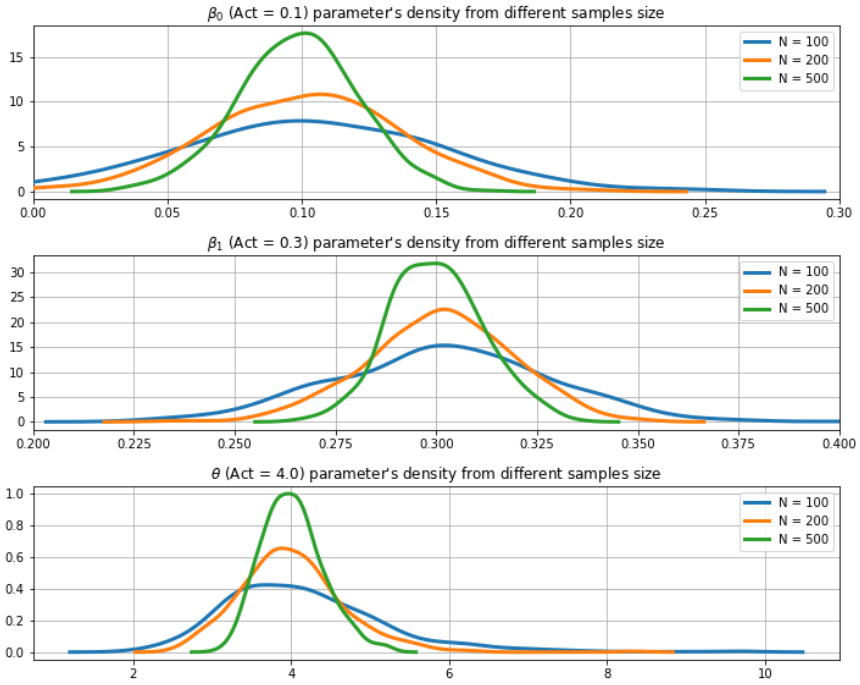

Figure 5: Plot of estimated parameter distributions with sample sizes of T = 100, 200, 500

1.5 Panel GARCH

In this section we specify th panel version of the GARCH(1, 1) model. We have daily returns $r_{i,t}$ for i = 1, ..., N and t = 1, ..., T, where i assigned the i-th stock's return and t is the t-th time step. We assume that the parameters which drive the dynamics of the return volatilites are common to all stocks. However the unconditional means of the volatilities are asset specific:

$$r_{i,t} = \sigma_{i,t}\epsilon_{i,t} \tag{24}$$

$$\sigma^2_{i,t} = \mu_i(1 - \alpha - \beta) + \alpha\epsilon^2_{i,t-1} + \beta\sigma^2_{i,t-1} \tag{25}$$

This means we have N + 2 numbers of parameters. This can be challenging to estimate as the number of assets increases. To tackle this issue we use the following estimation procedure.

1.5.1 Parameter estimation

First we calculate μ_i by moment matching. As μ_i is the unconditional variance of the returns we can estimate the μ_i parameters by averaging the squared returns.

$$\hat{\mu_t} = \frac{1}{N}\sum_{i=1}^{N} r^2_{i,t} \tag{26}$$

In the second step given the unconditional variance estimates we can estimate the remaining two parameters by maximum likelihood:

$$\log L(\alpha, \beta \mid \mu_i) = \sum_{i=1}^{N} \sum_{t=1}^{T} -\frac{1}{2} \log 2\pi - \frac{1}{2}\sigma_{i,t}^2 - -\frac{1}{2}\frac{r_{i,t}}{\sigma_{i,t}^2} \tag{27}$$

where $\sigma_{i,t}^2$ is the function of the α, β.

1.5.2 Simulations

We applied the same simulation technic, that we described in the GARCH section. The aim of the simulation was still the same, but now we examined the impact of the increment in the size of N. The results can be seen in the following figures:

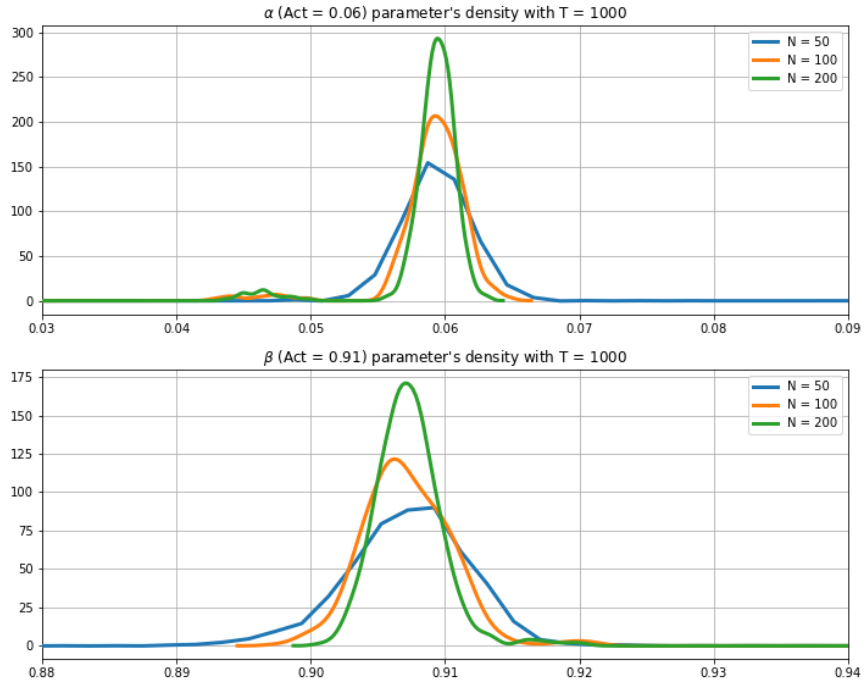

Figure 6: Plot of estimated parameter distributions with sample sizes of 1000 and N = 500, 1000, 2000

1.6 Panel GARCH with cross sectional adjustment

In this section we want to specify what we changed in contrast to the Panel GARCH section. We have:

$$r_{i,t} = \sigma_{i,t} c_t \epsilon_{i,t} \tag{28}$$

$$c_t = (1 - \phi) + \phi \sqrt{\frac{1}{N} \sum_{i=1}^{N} (\frac{r_{it-1}}{\sigma_{it-1} c_{t-1}} - \frac{1}{N} \sum_{i=1}^{N} \frac{r_{it-1}}{\sigma_{it-1} c_{t-1}})^2} \tag{29}$$

$$\sigma_{i,t}^2 = \mu_i(1-\alpha-\beta) + \alpha\epsilon_{i,t-1}^2 + \beta\sigma_{i,t-1}^2 \tag{30}$$

1.6.1 Parameter estimation

Estimation is done the same as the panel GARCH case. First we do the unconditional means by matching the second moment. We do the MLE on α, β and ϕ.

1.6.2 Simulations

The simulations remain the same, so the random generation happens through ϵ_t. We examined the cases of increase the sample sizes and the number of stocks. The results are the following:

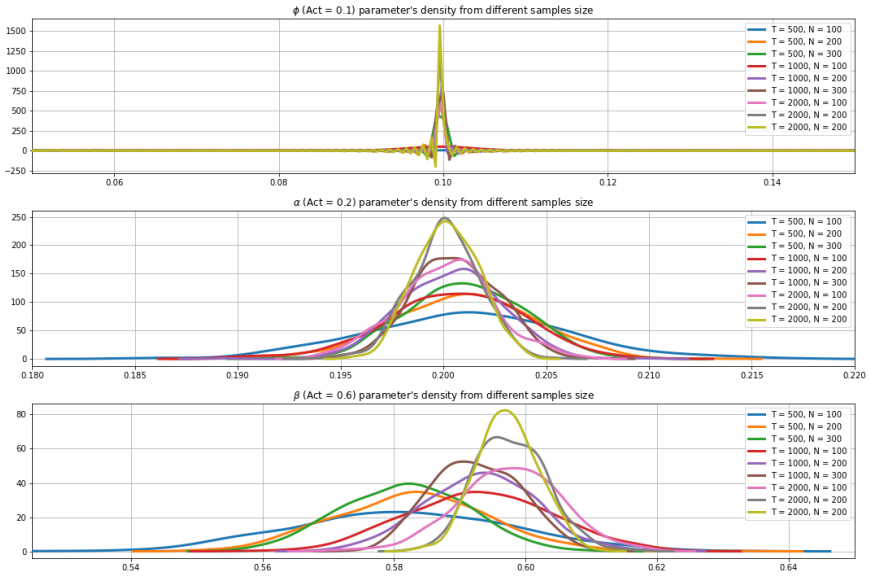

Figure 7: Plot of estimated parameter distributions with sample sizes of 1000 and N = 500, 1000, 2000

We can see that for the first to parameters (ϕ and α) the distributions of the parameter estimation's median is equal to the given parameters. In the case of β as we increase the sample size and number of stocks, we get closer and closer to the theoretical parameter.

2 Data

In this section, we will go through the macroeconomic variables we used, and what transformation or changes we made. We want to mention, all of our data came from resources that free for everyone. In the selection of macroeconomic variable we mainly rely on those that was previously used in research papers, such as **?** where they used several variables, that will be presented. Both of these time series data start at 1997-01-01 and end at 2020-11-01. We make use of the following time series:

- The AAII Investor Sentiment Survey (AAII) measures the percentage of individual investors who are bullish, bearish, and neutral on the stock market for the next months. The series reported on a weekly basis.
  https://www.aaii.com/files/surveys/sentiment.xls

- Moody's Seasoned BAA Corporate Bond Yield Relative to Yield on 10 Year Treasury Constant Maturity (BAA10Y) is a daily series.
  https://fred.stlouisfed.org/series/BAA10Y

- The Chicago Fed National Activity Index (CFNAI) is a weighted average of 85 monthly filtered and standardized economic indicators. Whereas positive CFNAI values indicate an expanding US-economy above its historical trend rate, negative values indicate the opposite. **?**
  https://alfred.stlouisfed.org/series?seid=CFNAI

- Consumer Price Index for All Urban Consumers: All Items in U.S. City Average (CPIAUCSL) is a measure of the average monthly change in the price for goods and services paid by urban consumers between any two time periods.
  https://alfred.stlouisfed.org/series?seid=CPIAUCSL

- Case-Shiller U.S. National Home Price Index (CSUSHPINSA) is a monthly index the leading measures of U.S. residential real estate prices, tracking changes in the value of residential real estate nationally.
  https://fred.stlouisfed.org/series/CSUSHPINSA

- 10-Year Treasury Constant Maturity Rate (DGS10) is a daily percent.
  https://fred.stlouisfed.org/series/DGS10

- 3-Month Treasury Bill: Secondary Market Rate (DTB3) is a daily percent. **?**
  https://alfred.stlouisfed.org/series?seid=DTB3

- Housing Starts Total: New Privately Owned Housing Units Started (HOUST) is a monthly unit. **?**
  https://fred.stlouisfed.org/series/HOUST

- Industrial Production: Total Index (INDPRO) is a monthly economic indicator that measures real output for all facilities located in the U.S. **?**
  https://alfred.stlouisfed.org/series?seid=INDPRO

- M2 Money Stock (M2SL) is a monthly value in units of dollar billions.
  https://fred.stlouisfed.org/series/M2SL

- Chicago Fed National Financial Conditions Index (NFCI) provides a weekly update on U.S. financial conditions in money markets. Positive values of the NFCI indicate financial conditions that ar tighter than average, negative values indicate financial conditions that are looser than average. **?**
  https://fred.stlouisfed.org/series/NFCI

- Producer Price Index by Commodity: All Commodities (PPIACO) is a monthly index.
  https://alfred.stlouisfed.org/series?seid=PPIACO
- Unemployment Rate (UNRATE) represents the number of unemployed as a monthly percentage of the labor force. **?**
  https://fred.stlouisfed.org/series/UNRATE
- CBOE Volatility Index: VIX (VIXCLS) is a daily close index that measures market expectation of near term volatility conveyed by stock index option prices. **?**
  https://fred.stlouisfed.org/series/VIXCLS

The two most commonly used variables for calculating inflation are the differences of Consumer Price Index and Producer Price Index, as we will show in the correlation matrix they have a solid correlation between them. We are going to mark Inflation with the differences of CPIAUCSL and Δ PPI is the differences of PPIACO. We make the same transformation for the M2 Money Stock, Case-Shiller U.S. National Home Price Index, Housing Starts Total and Industrial Production. We wanted to measure the slope of the yield curve we subtracted the 10-Year Treasury Constant Maturity Rate with the 3-Month Treasury Bill as **?** they specified. Those variables the are observed weekly or daily we finally take there monthly mean. In order to keep our models as simply as possible, we will only use monthly macroeconomic variables for modeling, but we didn't want to miss them out, so we took those variables monthly average. The time-series of final dataset we will use for modeling:

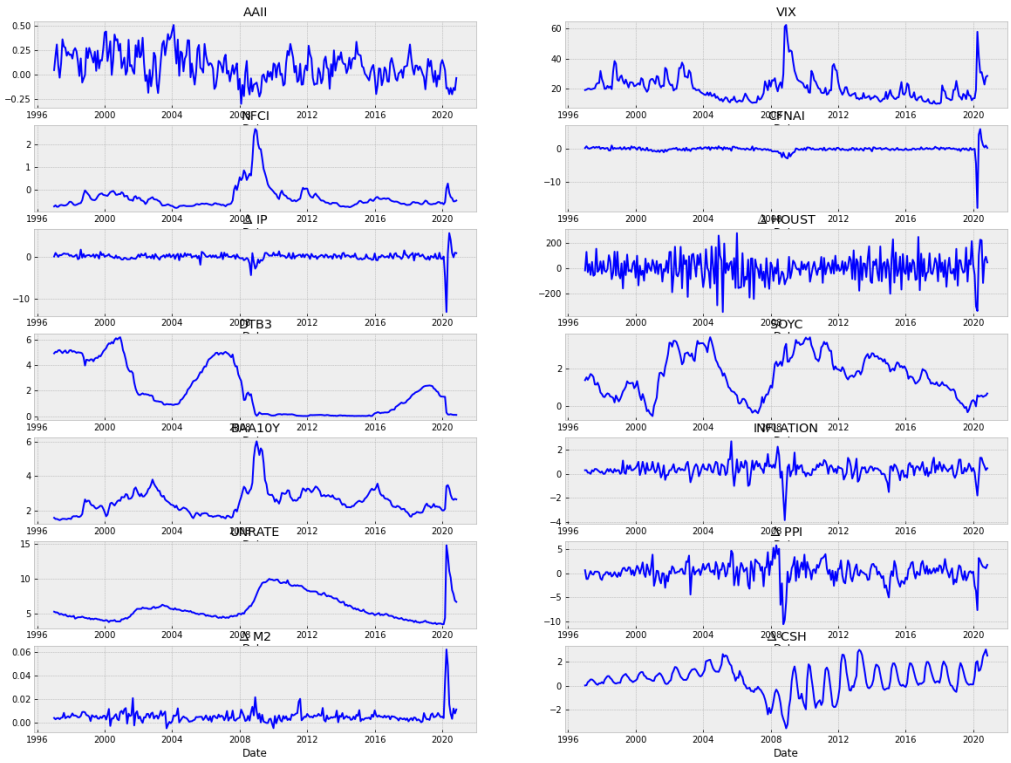

Figure 8: Time-series of macroeconomic variables

The stock prices we used for modeling are the prices of the S&P 500 Index components between 1999-12-01 and 2020-10-31, so due to the recent COVID-19's selloffs in the first quarter of 2020, we can examine two stressed period with our models. These data were downloaded by Python's package called *yfinance*.

	Min.	Max.	Mean	Median	Sd.	Skew.	Kurt
AAII	-0.30	0.51	0.08	0.08	0.15	0.21	-0.30
VIX	10.12	62.25	20.43	19.17	8.27	1.87	5.56
NFCI	-0.80	2.68	-0.36	-0.51	0.51	3.35	13.73
CFNAI	-17.73	5.96	-0.10	-0.01	1.28	-8.89	128.41
Δ IP	-13.26	5.74	0.09	0.14	1.13	-5.80	72.17
Δ HOUST	-343.00	279.00	0.73	-3.00	99.74	-0.26	0.79
DTB3	0.01	6.17	2.00	1.41	1.97	0.61	-1.14
Soyc	-0.53	3.69	1.64	1.62	1.11	0.03	-1.04
BAA10Y	1.45	6.01	2.53	2.50	0.75	1.57	4.73
Inflation	-3.84	2.70	0.36	0.39	0.59	-1.48	10.54
UNRATE	3.50	14.80	5.81	5.10	1.92	1.34	1.72
Δ PPI	-10.50	5.70	0.24	0.30	1.98	-1.28	5.75
Δ M2	-0.01	0.06	0.01	0.01	0.01	5.64	46.91
Δ Csh	-3.53	3.04	0.52	0.51	1.14	-0.60	0.92

Table 1: *Notes:* The table presents summary statistics for the variables.