# 1 Models

## 1.1 MIDAS

In this section we will introduce the MIDAS model, as it was first developed at **?** We have $y_t$ the low-frequency variable for t = 1, ..., T (say, monthly) and $x_t^{(m)}$ the high frequency variable can be observed (say, daily or m = 22). In the case of m equals to 1, there will be only low-frequency variables. 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 $\Theta$. $\beta_1$ 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:

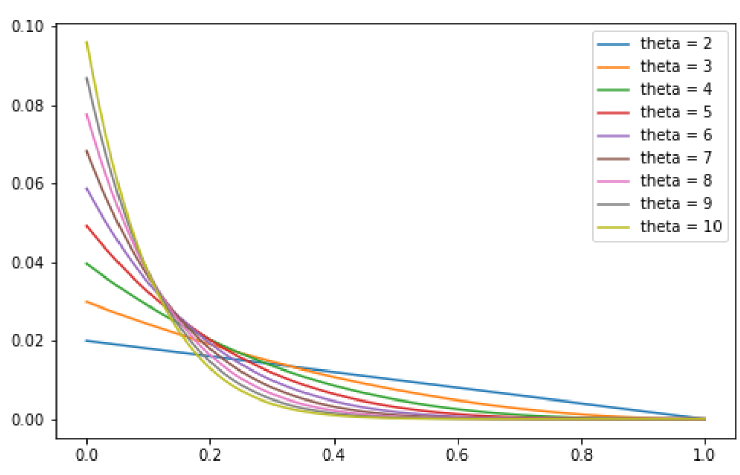


Figure 1: Plot of Beta Lag weighting function in equation **??** with K = 100, $\theta_1 = 1$ and $\theta_2 = 2, ..., 10$

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 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:

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

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

#### 1.1.3 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.

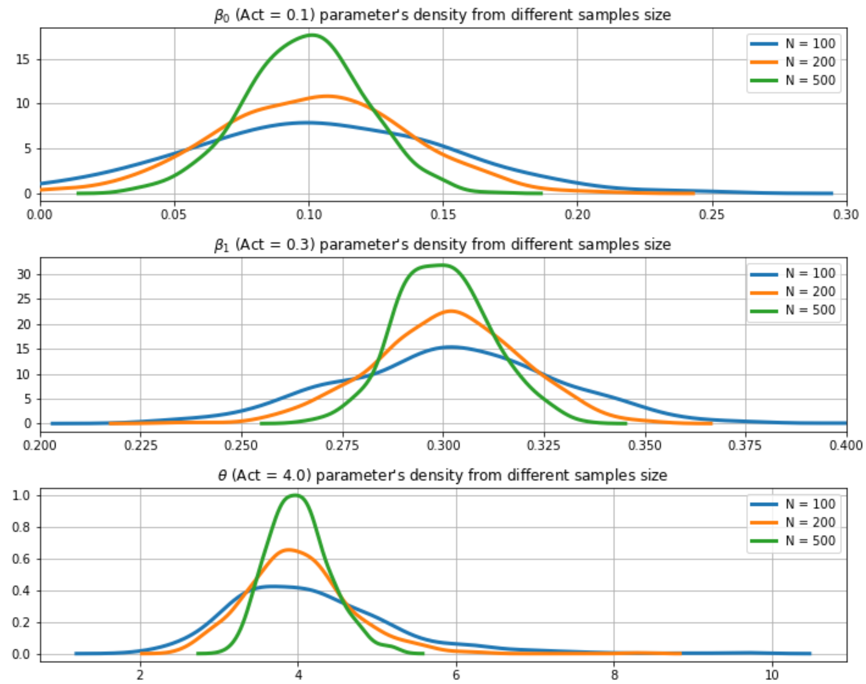


Figure 2: Plot of estimated parameter distributions with sample sizes of 100, 200, 500

## 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 $\epsilon_t$ 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 $\Theta$ is a compact subspace of an Euclidean space such that $\epsilon_t$ process finite second moments. The loglikelihood function for a sample of N observation is:

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

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

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

### 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 $\epsilon_t$ every step with the current state of $\sigma_t^2$. 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:

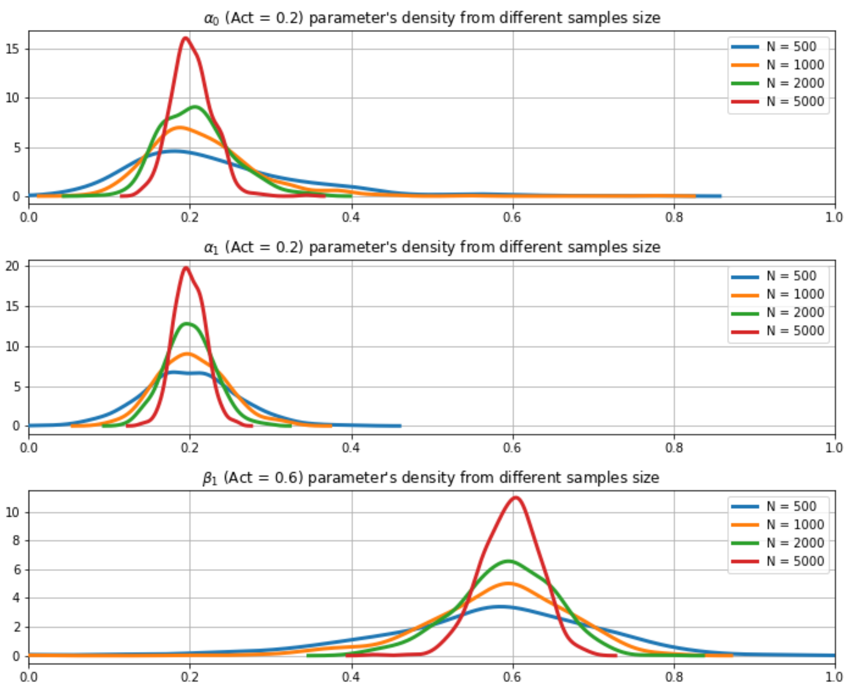


Figure 3: Plot of estimated parameter distributions with sample sizes of 500, 1000, 2000

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

## 1.3 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:

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

$$\sigma_{i,t}^2 = \mu_i(1-\alpha-\beta) + \alpha\epsilon_{i,t-1}^2 + \beta\sigma_{i,t-1}^2 \tag{12}$$

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.

### 1.3.1 Parameter estimation

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

$$\hat{\mu_t} = \frac{1}{N}\sum_{i=1}^{N} r_{i,t}^2 \tag{13}$$

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

$$\log L(\alpha, \beta \mid \mu_i) = \sum_{i=1}^{N}\sum_{t=1}^{T} -\frac{1}{2}\log 2\pi - \frac{1}{2}\sigma_{i,t}^2 - -\frac{1}{2}\frac{r_{i,t}}{\sigma_{i,t}^2} \tag{14}$$

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

### 1.3.2 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:

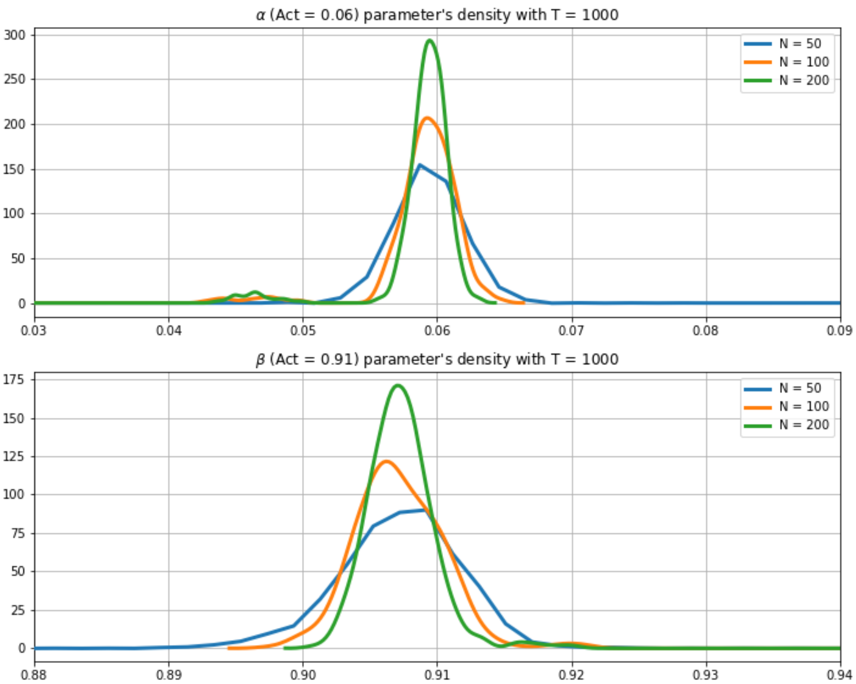


Figure 4: Plot of estimated parameter distributions with sample sizes of 1000 and N = 500, 1000, 2000

## 1.4 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:

$$r_{i,t} = \sigma_{i,t} c_t \epsilon_{i,t} \tag{15}$$

$$c_t = (1-\phi) + \phi \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{r_{it-1}}{\sigma_{it-1}c_{t-1}} - \frac{1}{N}\sum_{i=1}^{N}\frac{r_{it-1}}{\sigma_{it-1}c_{t-1}}\right)^2} \tag{16}$$

$$\sigma_{i,t}^2 = \mu_i(1-\alpha-\beta) + \alpha\epsilon_{i,t-1}^2 + \beta\sigma_{i,t-1}^2 \tag{17}$$

### 1.4.1 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 $\alpha, \beta$ and $\phi$.

### 1.4.2 Simulations

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

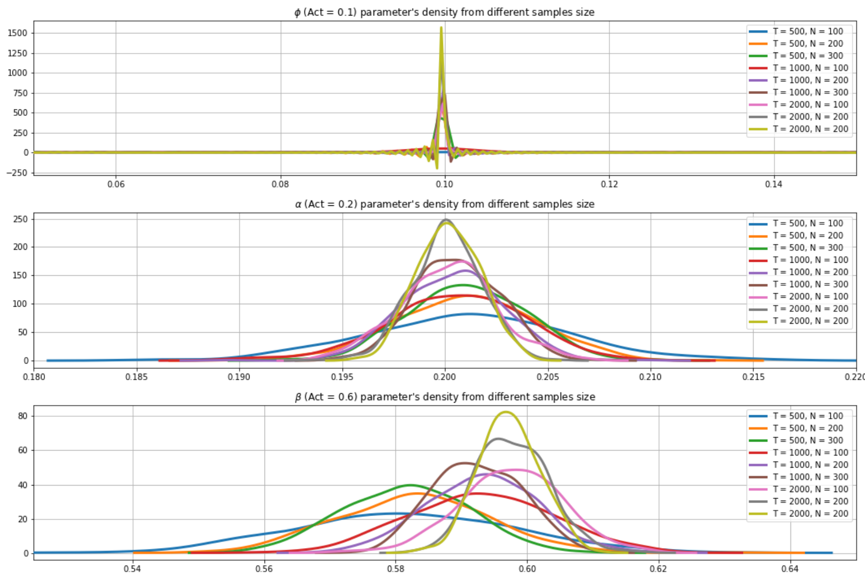


Figure 5: Plot of estimated parameter distributions with sample sizes of 1000 and N = 500, 1000, 2000

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