Lecture Notes: part 2

Financial Econometrics

2024-2025

Francisco Blasques
and
Paolo Gorgi

Contents

Part II

Parameter Driven Models

Chapter 9

Univariate Stochastic Volatility model

Until now, we have worked with observation-driven models of conditional variances and conditional covariances between multiple stock returns. In the univariate case, we considered models of the form

$$y_t = \sigma_t \varepsilon_t,$$

where the conditional variance σ_t^2 is specified as a function of past observations $\{y_{t-1}, y_{t-2}, \dots\}$. The simplest conditional volatility model we considered was the ARCH(1) model where $\sigma_t^2 = \omega + \alpha_1 y_{t-1}^2$. As you may recall, the ARCH(1) model is said to be *observation-driven* since the time-varying conditional variance is a given constant when we condition on the past observed data Y^{t-1}.

In this chapter we will introduce the class of *parameter-driven models*. In particular, we first study a parameter-driven model for time varying conditional variance known as the *Stochastic Volatility* (SV) model. The SV model is a parameter-driven model because the time varying conditional variance σ_t^2 evolves exogenously according to its own dynamic equation, rather than being determined by the past observed data Y^{t-1}. As a result, in the SV model, the conditional variance σ_t^2 is a random variable, even if we condition on the past observations Y^{t-1}.

As we shall see, this peculiarity of observation-driven models makes estimation of static parameter more challenging. This happens because we cannot derive an analytic expression for the likelihood-function, as we did, for example, for GARCH models. Instead, we will have to introduce new simulation-based estimation procedures. Specifically, we will learn how to estimate parameters by the method of *indirect inference*, which can be used in general situations, even when an analytic expression for the likelihood is not available.

9.1 The SV model

The *observation equation* of an SV model looks the same as for GARCH models, namely

$$y_t = \sigma_t \varepsilon_t,$$

where $\{\varepsilon_t\}_{t\in\mathbb{Z}}$ is an $NID(0,1)$ random sequence. The peculiarity of the SV model comes from the specification of the *transition* or *updating equation*. Note that in observation-driven models, as the ARCH/-GARCH model, the equation that specifies the time varying variance is called *updating equation*. Instead, for parameter-driven models this equation is called *transition equation*. for σ_t^2 as

$$\sigma_t^2 = \exp(f_t), \quad f_t = \omega + \beta f_{t-1} + \eta_t,$$

where $\{\eta_t\}_{t\in\mathbb{Z}}$ is $NID(0, \sigma_\eta^2)$ and independent of $\{\varepsilon_t\}_{t\in\mathbb{Z}}$. Despite the the exponential link function $\exp(\cdot)$, which is needed to ensure that the conditional volatility σ_t^2 is positive, a key feature of the SV model is that the volatility parameter $\sigma_t^2 = \exp(f_t)$ is specified as an unobserved process and not as a function of past observations. Therefore, given the past Y^{t-1} the volatility σ_t^2 is not a known constant. This is the main difference with respect to GARCH models.

The similarity between the GARCH model and the SV model is that they are both able to model the *volatility clustering* observed in financial returns. In particular, assuming β is close to 1, we have that when f_t is large then the volatility of y_t is large as well and it is likely that also f_{t+1} will be large and thus the volatility of y_{t+1} as well. In the following, we study some properties of the SV model that allow us to better understand dynamic behavior of this model. First, we introduce the *log-normal* distribution that will become useful in the rest of the chapter.

The **log-normal** distribution is defined on the basis of the normal distribution. In particular, assume we have a normal random variable X with mean μ and variance σ^2, i.e. $X \sim N(\mu, \sigma^2)$. Then the variable $Y = \exp(X)$ is a log-normal random variable with parameters μ and σ^2, i.e. $Y \sim \text{log-}N(\mu, \sigma^2)$.
The probability density function f(y) of the log-Normal for $y \in (0, \infty)$ is given by

$$f(y) = \frac{1}{y\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(\log y - \mu)^2}{2\sigma^2}\right).$$

The mean and the variance of a log-normal random variable Y are

$$\mathbb{E}(Y) = \exp(\mu + \sigma^2/2),$$
$$\mathbb{V}ar(Y) = (\exp(\sigma^2) - 1)\exp(2\mu + \sigma^2).$$

To study the SV process a first important step is to understand the properties of the unobserved process $\{f_t\}_{t\in\mathbb{Z}}$. Indeed this is very simple as f_t is a simple AR(1) process. Therefore, you already know all its properties from your previous time series course. The properties are summarized in the following theorem

Theorem 9.1. *When $|\beta| < 1$, the unobserved random sequence $\{f_t\}_{t\in\mathbb{Z}}$ is a weakly stationary AR(1) process with the following properties*

1. *The unconditional mean is $\mu_f = \mathbb{E}(f_t) = \omega/(1-\beta)$.*
2. *The unconditional variance is $\sigma_f^2 = \mathbb{V}ar(f_t) = \sigma_\eta^2/(1-\beta^2)$.*
3. *The unconditional distribution is $f_t \sim N(\mu_f, \sigma_f^2)$.*
4. *The autocorrelation function is $\rho_f(l) = \mathbb{C}ov(f_t, f_{t-l})/\mathbb{V}ar(f_t) = \beta^l$, for $l = 0, 1, 2, \ldots$.*

First we note that, similarly to ARCH/GARCH models, the conditional mean (conditional on the past Y^{t-1}) of the SV model is equal to zero. These result is provided by the next theorem

Theorem 9.2. *Let $\{y_t\}_{t\in\mathbb{Z}}$ be generated by an SV model with $|\beta| < 1$, then y_t is uncorrelated, $\mathbb{C}ov(y_t, y_{t-l}) = 0$ for $l > 0$, and the conditional mean of y_t given the past Y^{t-1} is equal to zero, i.e. $\mathbb{E}(y_t|Y^{t-1}) = 0$.*

Proof. The conditional mean can be obtained as follows

$$\mathbb{E}(y_t|Y^{t-1}) = \mathbb{E}(\sigma_t\varepsilon_t|Y^{t-1}) = \mathbb{E}(\sigma_t|Y^{t-1})\mathbb{E}(\varepsilon_t|Y^{t-1}) = \mathbb{E}(\sigma_t|Y^{t-1}) \times 0 = 0.$$

The second equality follows because σ_t and ε_t are independent and the third equality follows because ε_t is independent of the past Y^{t-1} and therefore $\mathbb{E}(\varepsilon_t|Y^{t-1}) = \mathbb{E}(\varepsilon_t) = 0$. Furthermore, this immediately implies that y_t is uncorrelated because

$$\mathbb{C}ov(y_t, y_{t-l}) = \mathbb{E}(y_t y_{t-l}) = \mathbb{E}(\mathbb{E}(y_t y_{t-l}|Y^{t-1})) = \mathbb{E}(y_{t-l}\mathbb{E}(y_t|Y^{t-1})) = \mathbb{E}(y_{t-l} \times 0) = 0,$$

where the second equality follows from the law of total expectation, the third from the fact that y_{t-l} is a constant conditional on Y^{t-1} and the fourth from the fact that $\mathbb{E}(y_t|Y^{t-1}) = 0$. ∎

We now focus our attention to the variance of y_t. The first thing to note is that the conditional variance of y_t is time varying but unfortunately there is no close form available for it. It is important to understand that $\sigma_t^2 = \exp(f_t)$ is not the conditional variance of y_t given Y^{t-1}. This because σ_t^2 is not a function of past observations because σ_t^2 depends on the unobservable Ar(1) process f_t.

Remark 9.1. *The conditional variance of y_t given Y^{t-1}, i.e. $\mathbb{V}ar(y_t|Y^{t-1})$, is time varying but there is no close form expression available.*

Although the conditional variance is time-varying, the SV model has a constant unconditional variance. This is shown in the next theorem. To obtain this result we make use of the log-normal distribution properties

Theorem 9.3. *Let $\{y_t\}_{t\in\mathbb{Z}}$ be generated by an SV model with $|\beta| < 1$, then the unconditional variance of y_t is*

$$\mathbb{V}ar(y_t) = \exp\left(\frac{\omega}{1-\beta} + \frac{\sigma_\eta^2}{2(1-\beta^2)}\right).$$

Proof. The first step is to recognize that the unconditional distribution of f_t is normal with mean $\mu_f = \omega/(1-\beta)$ and variance $\sigma_f^2 = \sigma_\eta^2/(1-\beta^2)$, i.e. $f_t \sim N(\mu_f, \sigma_f^2)$. Therefore, the unconditional distribution of $\sigma_t^2 = \exp(f_t)$ is log-normal $\sigma_t^2 \sim \text{log-}N(\mu_f, \sigma_f^2)$. Knowing this we obtain that

$$\mathbb{V}ar(y_t) = \mathbb{E}(y_t^2) = \mathbb{E}(\sigma_t^2\varepsilon_t^2) = \mathbb{E}(\sigma_t^2)\mathbb{E}(\varepsilon_t^2) = \mathbb{E}(\sigma_t^2) \times 1 = \mathbb{E}(\sigma_t^2) = \exp(\mu_f + \sigma_f^2/2),$$

where the last equality follows from the fact that σ_t^2 has a log-normal distribution. Therefore given the expressions of μ_f and σ_f^2 we can conclude that

$$\mathbb{V}ar(y_t) = \exp\left(\frac{\omega}{1-\beta} + \frac{\sigma_\eta^2}{2(1-\beta^2)}\right).$$

■

The next results shows that the unconditional distribution of y_t generated by the SV model is not normal. In particular, the unconditional distribution has fatter tails than the normal i.e. the kurtosis of y_t is bigger than 3. This is coherent with the high kurtosis often observed for log-returns.

Theorem 9.4. *Let $\{y_t\}_{t\in\mathbb{Z}}$ be generated by an SV model with $|\beta| < 1$, then the Kurtosis of y_t is given by*

$$k_u = \frac{\mathbb{E}(y_t^4)}{\mathbb{E}(y_t^2)^2} = 3\exp\left(\frac{\sigma_\eta^2}{1-\beta^2}\right).$$

Therefore $k_u > 3$ as long as $\sigma_\eta^2 > 0$.

Proof. The first step is to note that $\sigma_t^4 = \exp(2f_t)$. The unconditional distribution of $2f_t$ is normal with mean $2\mu_f$ and variance $4\sigma_f^2$, i.e. $f_t \sim N(2\mu_f, 4\sigma_f^2)$. Therefore, the unconditional distribution of $\sigma_t^4 = \exp(2f_t)$ is log-normal $\sigma_t^2 \sim \text{log-}N(2\mu_f, 4\sigma_f^2)$. Knowing this we obtain that

$$\mathbb{E}(y_t^4) = \mathbb{E}(\sigma_t^4\varepsilon_t^4) = \mathbb{E}(\sigma_t^4)\mathbb{E}(\varepsilon_t^4) = \mathbb{E}(\sigma_t^4) \times 3 = 3\mathbb{E}(\sigma_t^4) = 3\exp(2\mu_f + 2\sigma_f^2),$$

where the second-last equality follows because the fourth moment of a standard normal is equal to 3, $\mathbb{E}(\varepsilon_t^4) = 3$, and the last equality follows from the fact that σ_t^4 has a log-normal distribution. Now we know that $\mathbb{E}(y_t^2) = \exp(\mu_f + \sigma_f^2/2)$ and thus $\mathbb{E}(y_t^2)^2 = \exp(2\mu_f + \sigma_f^2)$. As a result we conclude that

$$k_u = \frac{\mathbb{E}(y_t^4)}{\mathbb{E}(y_t^2)^2} = \frac{3\exp(2\mu_f + 2\sigma_f^2)}{\exp(2\mu_f + \sigma_f^2)} = 3\exp(\sigma_f^2).$$

Therefore given the expressions of σ_f^2 we can conclude that

$$k_u = \frac{\mathbb{E}(y_t^4)}{\mathbb{E}(y_t^2)^2} = 3\exp(\sigma_f^2) = 3\exp\left(\frac{\sigma_\eta^2}{1-\beta^2}\right).$$

■

The next result shows that the SV model with with $|\beta| < 1$ is in fact a weakly stationary White Noise process.

Theorem 9.5. *Let $\{y_t\}_{t\in\mathbb{Z}}$ be generated by an SV model with $|\beta| < 1$, then $\{y_t\}_{t\in\mathbb{Z}}$ is a weakly stationary white noise sequence.*

Remark 9.2. *The condition $|\beta| < 1$ is also necessary and sufficient for strict stationarity.*

For GARCH models, we have seen that squared log-returns are autocorrelated because of the ARMA representation. Similarly, also squared log-returns from an SV model are autocorrelated. Figure 9.1 shows the autocorrelation function of returns and squared log-returns generated from an SV model. As we can see returns are uncorrelated instead squared log-returns show autocorrelation.

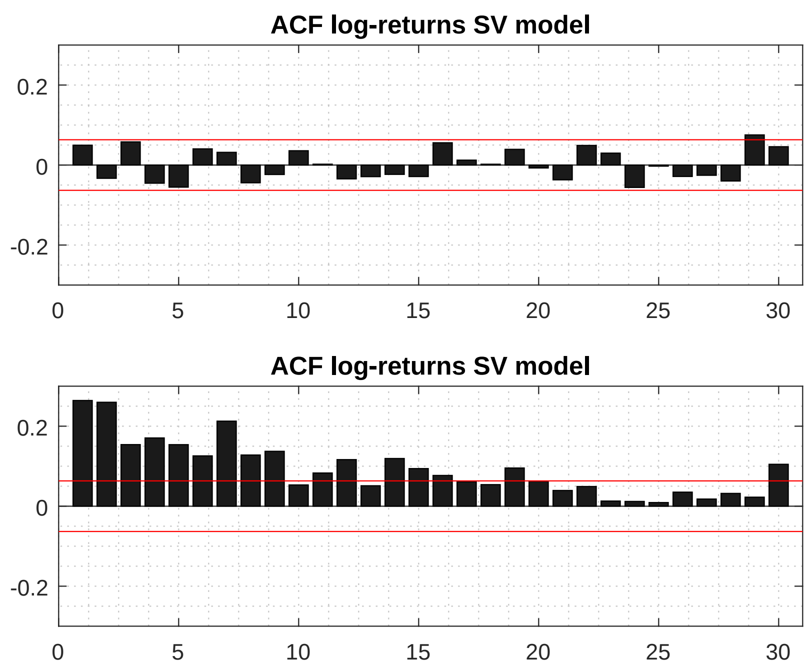

Figure 9.1: Sample ACF of y_t (first plot) and Sample ACF of y_t^2 (second plot). The series is generated from an SV model.

9.2 The SV-ARMA(p,q) model

The SV model presented in the previous section considers an AR(1) process for the unobserved f_t. This is the most popular specification and it is common practice to refer to this model as "the SV model". However, more general specifications for the unobserved f_t can be considered. In general we can have an ARMA(p,q) specification for f_t as

$$y_t = \sigma_t \varepsilon_t, \quad \sigma_t^2 = \exp(f_t),$$
$$f_t = \omega + \sum_{i=1}^{p} \beta_i f_{t-i} + \eta_t + \sum_{i=1}^{q} \alpha_i \eta_{t-i},$$

We call this model the SV-ARMA(p,q) model.

As for the SV model of the previous section, the properties of the SV-ARMA(p,q) model can be studied starting from the properties of the unobserved process f_t. This process is an ARMA model and therefore you should know how to obtain conditions for stationarity as well as obtain the unconditional mean and variance.

9.3 Simulate from an SV model with R

In the following, we show how to simulate from an SV model using R. The code can be found in the R file `Simulate_SV.R`.
The first step is to choose the sample size `n` and parameter values ω, β and σ_η^2 that are labeled `omega`, `beta` and `sig2f` respectively.

```
n <- 2500

omega <- 0
beta <-  0.95
sig2f <- 0.3
```

Second, we generate the sequences $\{\varepsilon_t\}_{t=1}^T$ and $\{\eta_t\}_{t=1}^T$ as follows.

```
epsilon <- rnorm(n)
eta <- sqrt(sig2f)*rnorm(n)
```

Then, we define 2 vectors `x` and `f` that will contain the the generated series and f_t.

```
f <- rep(0,n)
x <- rep(0,n)
```

Next, we initialize the AR(1) process f_t generating from its unconditional distribution and generate the first observation y_1.

```
f[1] <- omega/(1-beta) + sqrt(sig2f/(1-beta^2))*rnorm(1)
x[1] <- exp(f[1]/2) * epsilon[1]
```

Finally, we use a for loop to obtain the simulated series and stochastic volatility by means of the observation equation as well as the transition equation.

```
for(t in 2:n){
  f[t] <- omega + beta * f[t-1] + eta[t]
  x[t] <- exp(f[t]/2) * epsilon[t]
}
```

Figure 9.2 shows a time series as well as the unobserved stochastic volatility generated by an SV model.

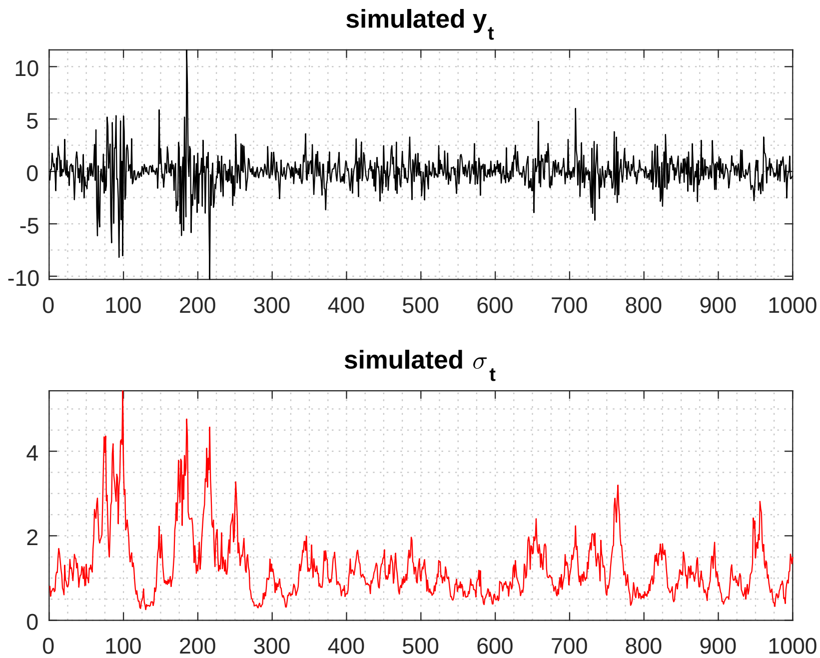

Figure 9.2: Generated time series (first plot) and generated stochastic volatility σ_t (second plot). The parameter value is $(\omega, \beta, \sigma_\eta^2) = (0, 0.95, 0.15)$.

9.4 Problems estimating an SV model by Maximum Likelihood

In the first part of the course we have seen that it is very easy to obtain the log-likelihood function for GARCH models. This is due to the fact that for GARCH models σ_t^2 is a constant conditional on Y^{t-1}. Therefore the distribution of y_t given Y^{t-1} is $N(0, \sigma_t^2)$ and from this we can easily obtain the log-likelihood. For the SV model this is not the case. The volatility σ_t^2 is not a constant given the past Y^{t-1}. As a result, we would need to solve an integral to obtain the log-likelihood function. Unfortunately this integral is intractable. In particular, given a sample of data $\{y_t\}_{t=1}^T$, the log-likelihood function of an SV model is given by

$$L(\theta; y_1, \dots, y_T) = \log\left(\int_{-\infty}^{\infty} \cdots \int_{-\infty}^{\infty} \prod_{t=1}^{T} p(y_t|f_t) p(f_1, \dots, f_T; \theta) df_1 \dots df_T\right),$$

where θ is the parameter vector, $p(y_t|f_t)$ is the conditional distribution of y_t given the unobserved f_t and $p(f_1, \dots, f_T; \theta)$ is the joint distribution of the vector $(f_1, \dots, f_T)$. This integral cannot be solved in closed form, i.e. there is no analytical solution. Furthermore, we note that the integral above is multidimensional. The dimensionality is equal to the sample size T. Therefore as you can imagine the dimension of this integral can be easily more than 1000. This prevents the use of numerical integration methods such as *Quadrature Methods* to obtain a solution. Numerical methods to solve integrals are not practically applicable when the dimensionality is too large.

For the reasons mentioned above Maximum Likelihood estimation is not easy to be implemented for SV models. In the next chapter we will see an an alternative approach to estimate SV model that is called Indirect Inference.

Chapter 10

Multivariate Stochastic volatility model

As we have discussed in the first part of the course, we are very often interested in modeling multiple time series. For instance, we have seen the importance of multivariate GARCH models for dynamic portfolio optimization. Therefore, we need to extend the univariate SV model to the multivariate case. As for multivariate GARCH models, there are several ways to specify multivariate SV models. In this section, we introduce a multivariate version of the SV model presented in the previous chapter. We call this model the *multivariate stochastic volatility* (MSV) model.

10.1 Multivariate Stochastic volatility model

The MSV model features an observation equation that is exactly the same as for multivariate GARCH models. Namely, the vector of returns $\boldsymbol{y}_t = (y_{1t}, ..., y_{nt})$ at time t, is modeled as the product of a vector of n Gaussian innovations $\boldsymbol{\varepsilon}_t$ with the square-root of a matrix $\boldsymbol{\Sigma}_t$,

$$\boldsymbol{y}_t = \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\varepsilon}_t.$$

A fundamental difference between the MSV and multivariate GARCH models lies however on the fact that $\boldsymbol{\Sigma}_t^{1/2}$ is a diagonal matrix, containing no cross-terms, and that the elements of the vector of innovations $\boldsymbol{\varepsilon}_t$ are allowed to be dependent, so the variance covariance matrix of $\boldsymbol{\varepsilon}_t$ is no longer the identity matrix. In particular, we have that

$$\boldsymbol{\Sigma}_t = \text{diag}\{\exp(\boldsymbol{f}_t)\} \qquad \text{and} \qquad \{\boldsymbol{\varepsilon}_t\}_{t\in\mathbb{Z}} \sim NID_n(\mathbf{0}, \boldsymbol{R})$$

where $\boldsymbol{f}_t$ is an n-dimensional vector of time-varying parameters, and $\boldsymbol{R}$ is a matrix of static coefficients that defines the covariance structure of the innovations $\boldsymbol{\varepsilon}_t$. The matrix $\boldsymbol{R}$ is also normalized as a correlation matrix since its diagonal elements are all equal to 1, and its off-diagonal elements take values in the interval $[-1, 1]$. It is important to note however that the matrix $\boldsymbol{\Sigma}_t$ is *not* the conditional variance-covariance matrix of $\boldsymbol{y}_t$. First, the matrix $\boldsymbol{\Sigma}_t$ contains only diagonal elements. Second, even conditioning on the past observations Y^{t-1}, the matrix $\boldsymbol{\Sigma}_t$ is still random. Hence, the reasoning that we used for deriving the conditional distribution of the returns $\boldsymbol{y}_t$ of a multivariate GARCH model cannot be applied here.

In any case, from an intuitive sense, it is convenient to note that if $\boldsymbol{\Sigma}_t$ were given (not random), then its diagonal elements would indeed contain the variances of the vector of returns $\boldsymbol{y}_t$, and that the matrix $\boldsymbol{R}$ would correspond to the correlation matrix of the returns. This follows from the fact that, for a constant Σ_t, we would have $\mathbb{V}\text{ar}(\boldsymbol{\Sigma}_t^{1/2}\boldsymbol{\varepsilon}_t) = \boldsymbol{\Sigma}_t^{1/2}\boldsymbol{R}\boldsymbol{\Sigma}_t^{1/2}$, where $\boldsymbol{R}$ is then the correlation matrix because any variance-covariance matrix C can always be decomposed as $\boldsymbol{C} = \boldsymbol{V}\boldsymbol{R}\boldsymbol{V}$, where $\boldsymbol{V}$ is a matrix that contains the variances in the diagonal, and $\boldsymbol{R}$ is the correlation matrix.

For the bivariate case ($n = 2$), we thus have that the observation equation of the MSV model is given by

$$\begin{bmatrix} y_{1t} \\ y_{2t} \end{bmatrix} = \begin{bmatrix} \exp(f_{1t}) & 0 \\ 0 & \exp(f_{2t}) \end{bmatrix}^{1/2} \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix} \qquad \text{where} \qquad \begin{bmatrix} \varepsilon_{1t} \\ \varepsilon_{2t} \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix} \right).$$

The updating equation for the vector of time-varying parameters $\boldsymbol{f}_t$ of the MSV model is a direct extension of the autoregressive dynamics encountered in the univariate SV model. In particular, it takes the form

$$\boldsymbol{\Sigma}_t = \text{diag}\{\exp(\boldsymbol{f}_t)\} \qquad \text{and} \qquad \boldsymbol{f}_{t+1} = \boldsymbol{\omega} + \boldsymbol{\beta} \odot \boldsymbol{f}_t + \boldsymbol{\eta}_t \qquad \text{where} \qquad \{\boldsymbol{\eta}_t\}_{t\in\mathbb{Z}} \sim NID_n(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}}).$$

In the bivariate case ($n = 2$), the updating equation can thus be written as follows

$$\begin{bmatrix} f_{1t+1} \\ f_{2t+1} \end{bmatrix} = \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} + \begin{bmatrix} \beta_1 f_{1t} \\ \beta_2 f_{2t} \end{bmatrix} + \begin{bmatrix} \eta_{1t} \\ \eta_{2t} \end{bmatrix}, \quad \text{where} \quad \begin{bmatrix} \eta_{1t} \\ \eta_{2t} \end{bmatrix} \sim N\left(\begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \sigma_{1\eta}^2 & \sigma_{12\eta} \\ \sigma_{12\eta} & \sigma_{2\eta}^2 \end{bmatrix} \right).$$

This updating equation allows for all the dynamics of vector autoregressive (VAR) models that you have learned in your introductory time-series courses. Furthermore, it can naturally be extended to the VAR(p) case where

$$\boldsymbol{f}_{t+1} = \boldsymbol{\omega} + \boldsymbol{\beta}_1 \odot \boldsymbol{f}_t + ... + \boldsymbol{\beta}_p \odot \boldsymbol{f}_{t-p} + \boldsymbol{\eta}_t.$$

Taking all these elements together, we obtain the specification for the n-dimensional multivariate SV model as follows

$$\begin{aligned} &\boldsymbol{y}_t = \boldsymbol{\Sigma}_t^{1/2} \boldsymbol{\varepsilon}_t, \\ &\boldsymbol{\Sigma}_t = \text{diag}\{\exp(\boldsymbol{f}_t)\}, \\ &\boldsymbol{f}_{t+1} = \boldsymbol{\omega} + \boldsymbol{\beta} \odot \boldsymbol{f}_t + \boldsymbol{\eta}_t, \\ &\{\boldsymbol{\varepsilon}_t\}_{t\in\mathbb{Z}} \sim NID_n(\mathbf{0}, \boldsymbol{R}), \\ &\{\boldsymbol{\eta}_t\}_{t\in\mathbb{Z}} \sim NID_n(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}}), \end{aligned}$$

where $\{\boldsymbol{\varepsilon}_t\}_{t\in\mathbb{Z}}$ is independent of $\{\boldsymbol{\eta}_t\}_{t\in\mathbb{Z}}$, $\boldsymbol{\Sigma}_t$ is a diagonal matrix, $\boldsymbol{R}$ is a correlation matrix and $\boldsymbol{\Sigma}_{\boldsymbol{\eta}}$ a covariance matrix. Note that since $\boldsymbol{f}_t$ is of the same dimension as $\boldsymbol{y}_t$, the curse of dimensionality is somewhat attenuated, at least when compared to the multivariate GARCH VECH model discussed in previous chapters.

Naturally, the conditional correlation between y_{1t} and y_{2t} is constant and equal to ρ_{12} but the conditional covariance is time-varying. Figure 10.1 shows σ_{1t}^2, σ_{2t}^2 and $\sigma_{12t} = \rho_{12}\sigma_{1t}\sigma_{2t}$ generated by a bivariate SV model. Also for the multivariate SV model the conditional covariance matrix as well as the conditional distribution are not available in closed form. The matrix $\boldsymbol{\Sigma}_t$ is not the conditional covariance matrix given Y^{t-1}! The reason is that, as for the univariate SV, the matrix $\boldsymbol{\Sigma}_t$ is not a constant conditional on the past Y^{t-1}. As we for the univariate SV model, the likelihood function of the multivariate SV model is not available in closed form.

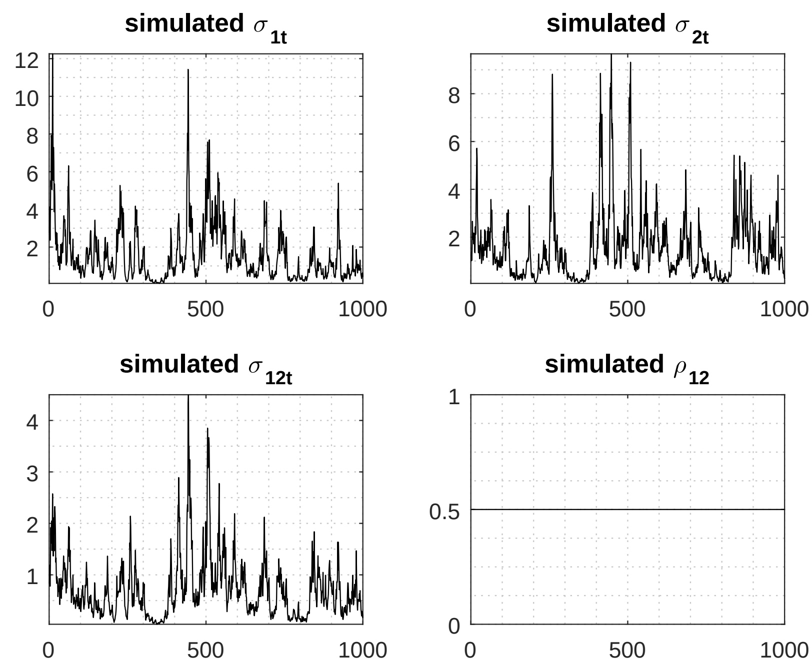

Figure 10.1: Simulated series from a bivariate Stochastic Volatility model.

10.2 Simulate from a bivariate SV model with R

In the following, we show how to simulate from a bivariate SV model using R. The code can be found in the R file `Simulate_multivariate_SV.R`.
The first step is to choose the sample size T, which is labeled `n`, and parameter values ω, β_1, β_2, $\sigma^2_{1\eta}$, $\sigma^2_{2\eta}$, $\sigma_{12\eta}$ and ρ_{12} that are labeled `omega`, `beta1`, `beta1`, `sig2f1`, `sig2f2`, `sigf12` and `rho` respectively.

```
omega1 <- 0
omega2 <- 0

beta1 <- 0.95
beta2 <- 0.95

sig2f1 <-  0.10
sig2f2 <-  0.10
sigf12 <-  0.05

rho <- 0.5

R <- cbind(c(1,rho),c(rho,1))
Sf <- cbind(c(sig2f1,sigf12),c(sigf12,sig2f2))
```

Second, we generate the sequences of error terms $\{\boldsymbol{\eta}_t\}_{t=1}^T$ and $\{\boldsymbol{\varepsilon}_t\}_{t=1}^T$ from a $NID_2(\mathbf{0}, \boldsymbol{R})$ and a $NID_2(\mathbf{0}, \boldsymbol{\Sigma_\eta})$ respectively.

```
epsilon <- mvrnorm(n,rep(0,2),R)
eta <- mvrnorm(n,rep(0,2),Sf)

x <- matrix(0,nrow=n,ncol=2)
f <- matrix(0,nrow=n,ncol=2)
```

Then, we generate the initial value of $\boldsymbol{f}_t$ at $t = 1$ drawing from its unconditional distribution.

```
umf <- c(omega1/(1-beta1), omega2/(1-beta2))
uSf <- matrix(0,nrow=2,ncol=2)
uSf[1,1] <- sig2f1/(1-beta1^2)
uSf[2,2] <- sig2f2/(1-beta2^2)
uSf[2,1] <- sigf12/(1-beta1*beta2)
uSf[1,2] <- sigf12/(1-beta1*beta2)

f[1,] <- mvrnorm(1,umf,uSf)
```

Next, we generate the first observation $\boldsymbol{y}_1$.

```
x[1,1] <- exp(f[1,1]/2) * epsilon[1,1]
x[1,2] <- exp(f[1,2]/2) * epsilon[1,2]
```

Finally, we recursively generate the $\boldsymbol{f}_t$ and $\boldsymbol{y}_t$ using the observation equation and transition equation respectively.

```
for(t in 2:n){
  f[t,1] <- omega1 + beta1*f[t-1,1] + eta[t,1]
  f[t,2] <- omega2 + beta2*f[t-1,2] + eta[t,2]
```

```
  x[t,1] <- exp(f[t,1]/2) * epsilon[t,1]
  x[t,2] <- exp(f[t,2]/2) * epsilon[t,2]
}
```

Figure 10.2 plots a time series generated by the bivariate SV model.

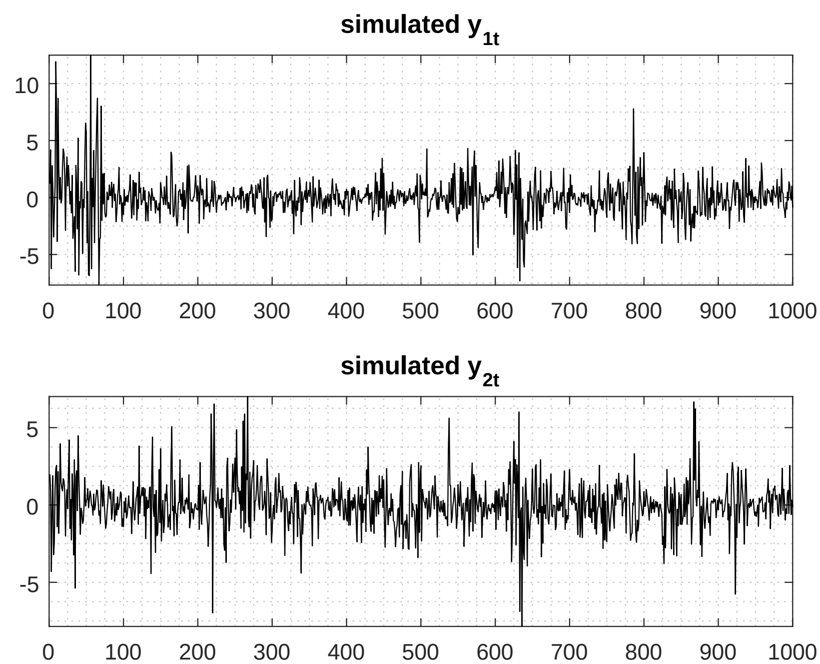

Figure 10.2: Simulated series from a bivariate Stochastic Volatility model.

Chapter 11

Indirect Inference Estimation

11.1 Indirect Inference

In Chapters 4 and 7, we learned how to estimate GARCH and multivariate GARCH models. We noted that, while it was difficult to obtain an expression for the estimator, it was still possible to write down the log-likelihood function and optimize it numerically.

Unfortunately, as we have discussed for the SV model, the log-likelihood function is typically intractable in the context of parameter driven models. This renders useless the numerical optimization procedures discussed in Chapters 4 and 7. *If we cannot write down the log likelihood function, then we must find another way of proceeding.* The solution to this problem typically consists of adopting *simulation-based procedures.* In this chapter we will learn how to estimate parameter driven models by the method of *indirect inference*, proposed by Smith (1993) and Gourieroux, Monfort and Renault (1993).

The idea behind indirect inference is simple. Suppose that we wish to estimate the vector or parameters θ from a parameter-driven model. Then, first, we describe the properties of the observed data $y_1, ..., y_T$ using a vector of *auxiliary statistics* $\hat{B}_T$ that we find relevant. The vector $\hat{B}_T$ may contain moments (like the mean, variance and covariances), or parameters of models that are simple to estimate (like regression models, ARMA models, ADL models, etc.). All that matters is that the vector of *auxiliary statistics* $\hat{B}_T$ provides a description of the dynamic properties of the observed data.

Next, for a given parameter value θ, we simulate a very long sample of data from our parameter-driven model $\tilde{y}_1(\theta), ..., \tilde{y}_H(\theta)$, and again, we use the vector of *auxiliary statistics* $\tilde{B}_H(\theta)$ to describe the properties of the simulated data. Notice that the vector $\tilde{B}_H(\theta)$ can describe very accurately the properties of the simulated data because we can simulate a very long path $\tilde{y}_1(\theta), ..., \tilde{y}_H(\theta)$ from our model.

Finally, we just need to find the value of the parameter vector θ that makes the properties of the simulated data $\tilde{B}_H(\theta)$ as close as possible to the properties $\hat{B}_T$ of the observed data. Once we find that parameter, then we call it *the indirect inference estimate of* θ.

Formally, the indirect inference estimator, denoted $\hat{\theta}_{TH}$, is defined as follows

$$\hat{\theta}_{TH} = \arg\min_{\theta \in \Theta} d\big(\hat{B}_T, \tilde{B}_H(\theta)\big)$$

where $d\big(\hat{B}_T, \tilde{B}_H(\theta)\big)$ denotes the quadratic distance between $\hat{B}_T$ and $\tilde{B}_H(\theta)$

$$d\big(\hat{B}_T\ ,\ \tilde{B}_H(\theta)\big) = \Big(\hat{B}_T - \tilde{B}_H(\theta)\Big) W \Big(\hat{B}_T - \tilde{B}_H(\theta)\Big)' \tag{11.1}$$

and W is a weighting matrix that gives different weights to each auxiliary statistic.

Lemma 11.1. *Under appropriate regularity conditions, the indirect inference estimator is consistent as the sample size of both observed data and simulated data grows to infinity,*

$$\hat{\theta}_{TH} \xrightarrow{p} 0 \quad as \ \ T \to \infty \ \ and \ \ H \to \infty.$$

Moreover, if H is a multiple of T given by ΔT the indirect inference estimator is asymptotically Gaussian

$$\sqrt{T}(\hat{\theta}_{TH} - \theta_0) \xrightarrow{d} N\left(\mathbf{0}, \left(1 + \frac{1}{\Delta}\right)\Sigma\right)$$

In general, it is worth noting the following:

1. The number of auxiliary statistics has to be equal or larger than the number of parameters in the vector θ. If we have too few *auxiliary statistics* then we will be unable identify θ.

2. As the algorithm tries to minimize the distance in (11.1) by iterating over different values of θ, it is important that the simulations be carried out using the same *seed* value for any θ. Otherwise, even with large H, the estimation error will render the criterion function in (11.1) non smooth, and, as result, it will be difficult to find the optimum value $\hat{\theta}_{TH}$.

3. As sample size increases ($T \to \infty$ and $H \to \infty$), the auxiliary statistics will converge to limit values which ultimately depend only on θ_0 (for the observed data), and θ (for the simulated data),

$$\hat{B}_T \xrightarrow{p} B(\theta_0) \quad \text{and} \quad \tilde{B}_H(\theta) \xrightarrow{p} B(\theta).$$

4. The function $B(\theta)$ that relates the values of the parameter-driven model θ to the properties of the data of interest, is called the *binding function.*

5. When the number of auxiliary statistics is larger than the number of parameters in θ, then different weighting matrices W may lead to different levels of estimation accuracy. In general however, the identity matrix is sufficient for the estimator to be consistent and asymptotically normal

$$W = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & & \vdots & & \vdots \\ 0 & 0 & 0 & \cdots & 1 \end{bmatrix}.$$

6. The asymptotic variance-covariance matrix of the indirect inference estimator shows that its accuracy depends on both T and H. The extra uncertainty introduced by the simulations vanishes as $H \to \infty$.

The indirect inference estimator $\hat{\theta}_{TH}$ is thus the parameter value that makes simulated data as similar as possible to observed data, as judged by the auxiliary statistics $\hat{B}_T$ and $\tilde{B}_H(\theta)$. Figure 11.1 below shows a diagram that illustrates the main elements of the indirect inference method.

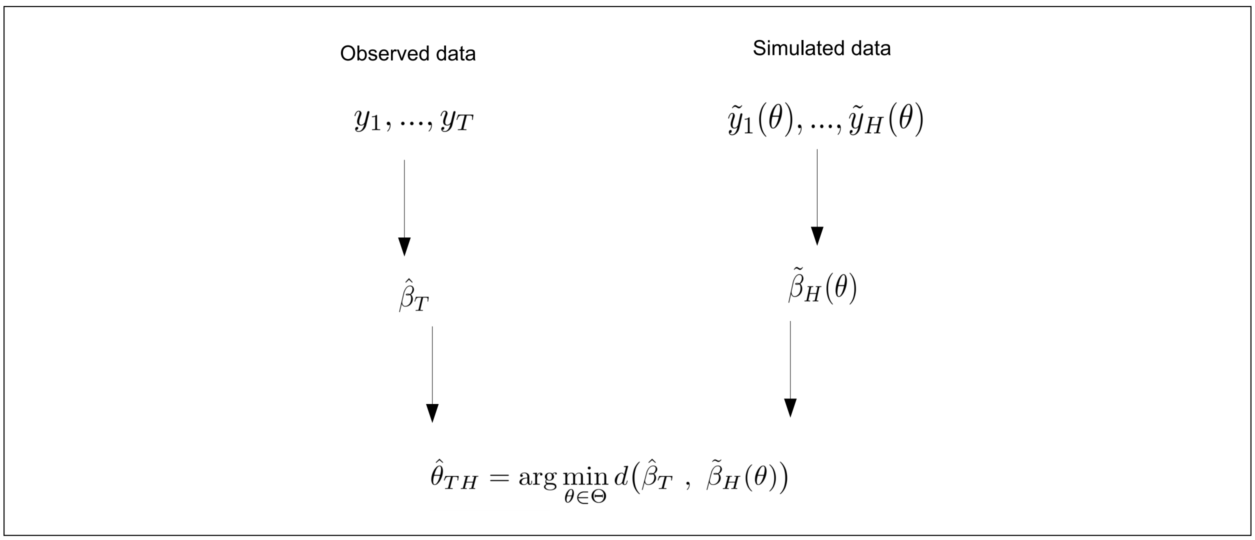

Figure 11.1: Diagram for the indirect inference method.

11.2 Example: Estimating an MA model

Consider the following MA(1) model

$$y_t = \epsilon_t + \phi\epsilon_{t-1}\ , \quad \epsilon_t \sim N(0, \sigma^2)$$

As you may recall from your introductory time series course, the parameters ϕ and σ^2 determine a number of properties for the time-series $\{y_t\}_{t\in\mathbb{Z}}$, including its variance and autocovariance structure. Taking this into account, it seems only natural to use the sample variance and the first-order autocovariance as auxiliary statistics to describe both the observed and simulated data. In particular, we could use

$$\hat{B}_T = \begin{bmatrix} \hat{\gamma}_0 \\ \hat{\gamma}_1 \end{bmatrix} = \begin{bmatrix} (1/T)\sum_{t=1}^{T} y_t^2 \\ (1/T)\sum_{t=2}^{T} y_t y_{t-1} \end{bmatrix}\ , \quad \text{and}$$

$$\tilde{B}_H(\theta) = \begin{bmatrix} \tilde{\gamma}_0(\theta) \\ \tilde{\gamma}_1(\theta) \end{bmatrix} = \begin{bmatrix} (1/H)\sum_{t=1}^{H} \tilde{y}_t^2(\theta) \\ (1/H)\sum_{t=2}^{H} \tilde{y}_t(\theta)\tilde{y}_{t-1}(\theta) \end{bmatrix}\ .$$

In this example, since we have 2 parameters to estimate $\theta = (\phi, \sigma^2)$ and four auxiliary statistics (sample variance, and first-order autocovariance), we can just go ahead and use an algorithm that minimizes the distance between $\hat{B}_T$ and $\tilde{B}_H(\theta)$ using an identity matrix. Figure and show the density function of the indirect inference estimator of the parameters α and θ for different sample sizes T and H. These densities were obtained by means of a Monte Carlo simulation.

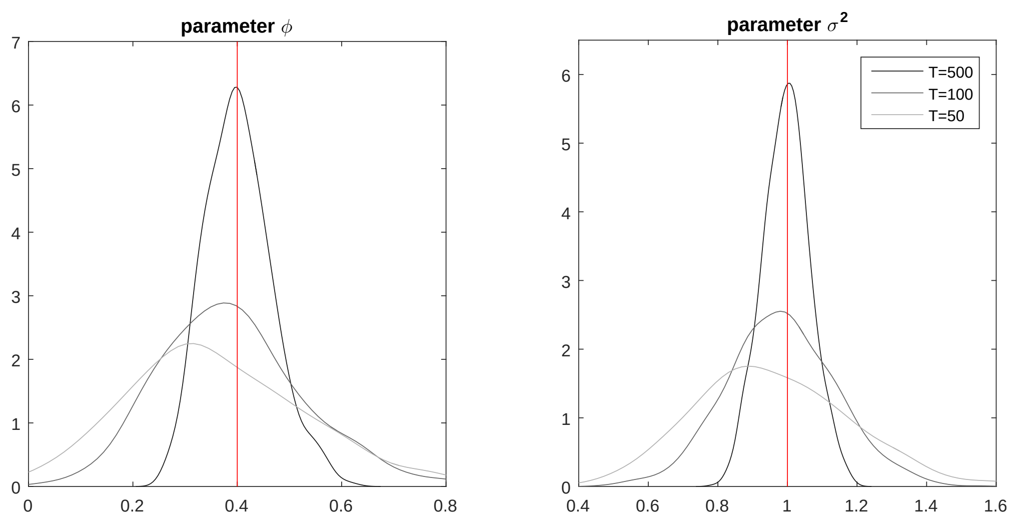

Figure 11.2: Distribution of the indirect inference estimator for different sample sizes T. The length of the simulations is set $H = 20T$.

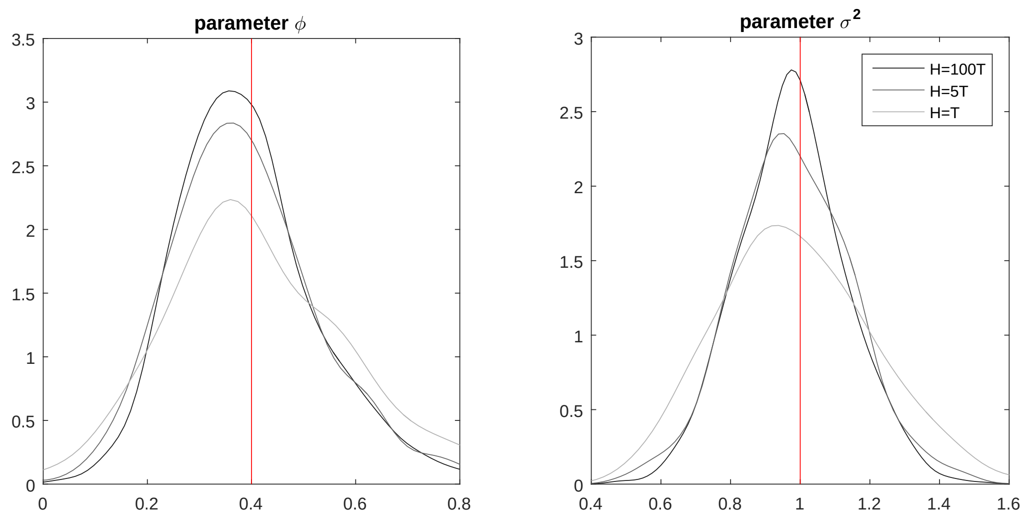

Figure 11.3: Distribution of the indirect inference estimator for different length of the simulated series H. The sample size of the series is set $T = 100$.

Chapter 12

Estimation of the SV model

12.1 Parameter estimation by indirect inference

Consider now the following stochastic volatility model

$$y_t = \sigma_t \varepsilon_t, \;\; \sigma_t^2 = \exp(f_t),$$
$$f_t = \omega + \beta f_{t-1} + \eta_t.$$

In Chapter 9 we have learned that the parameters ω, β and σ_η^2 of an SV model determine certain moments of y_t, like the variance and the kurtosis. Furthermore, we have also seen that these parameters determine the temporal dependence in squared log-returns y_t^2. As such, it is natural that we consider these sample moments as candidates for auxiliary statistics. In particular, we could use

$$\hat{B}_T = \begin{bmatrix} \hat{s}^2 \\ \hat{k}^2 \\ \hat{\gamma}_1^2 \\ \hat{\gamma}_2^2 \end{bmatrix} = \begin{bmatrix} (1/T)\sum_{t=1}^T y_t^2 \\ (1/T)\sum_{t=1}^T y_t^4 \\ (1/T)\sum_{t=2}^T (y_t^2 - \hat{s}^2)(y_{t-1}^2 - \hat{s}^2) \\ (1/T)\sum_{t=3}^T (y_t^2 - \hat{s}^2)(y_{t-2}^2 - \hat{s}^2) \end{bmatrix} \quad \text{and}$$

$$\tilde{B}_H(\theta) = \begin{bmatrix} \tilde{s}^2(\theta) \\ \tilde{k}^2(\theta) \\ \tilde{\gamma}_1^2(\theta) \\ \tilde{\gamma}_2^2(\theta) \end{bmatrix} = \begin{bmatrix} (1/H)\sum_{t=1}^H \tilde{y}_t^2(\theta) \\ (1/H)\sum_{t=1}^H \tilde{y}_t^4(\theta) \\ (1/H)\sum_{t=2}^H \big(\tilde{y}_t^2(\theta) - \tilde{s}^2(\theta)\big)\big(y_{t-1}^2(\theta) - \tilde{s}^2(\theta)\big) \\ (1/H)\sum_{t=3}^H \big(\tilde{y}_t^2(\theta) - \tilde{s}^2(\theta)\big)\big(y_{t-2}^2(\theta) - \tilde{s}^2(\theta)\big) \end{bmatrix}.$$

Note that instead of squared log-returns other suitable transformations could be used as for instance absolute log-returns $|y_t|$.

The parameters of an AR(p) model for squared log-returns y_t^2 constitute a natural alternative to these raw moments of the data. Consider the following AR(p) model,

$$y_t^2 = b_0 + \sum_{i=1}^p b_i y_{t-i}^2 + \epsilon_t \;, \quad \epsilon_t \sim N(0, c^2)$$

The parameters of this AR(p) model describe the autocovariance structure of the squared log-returns, and this determines precisely the moments of y_t described by the auxiliary statistics above. In this setting the auxiliary statistics correspond to the estimated AR(p) parameters $\hat{b}_0, \ldots, \hat{b}_p$ and $\hat{c}^2$ obtained using observed squared log-returns y_t^2, as well as the estimated AR(p) parameters $\tilde{b}_0(\theta), \ldots, \tilde{b}_p(\theta)$ and $\tilde{c}^2(\theta)$ obtained using simulated squared log-returns $\tilde{y}_t^2(\theta)$.

$$\hat{B}_T = \begin{bmatrix} \hat{b}_0 \\ \vdots \\ \hat{b}_p \\ \hat{c}^2 \end{bmatrix} \qquad \text{and} \qquad \tilde{B}_H(\theta) = \begin{bmatrix} \tilde{b}_0(\theta) \\ \vdots \\ \tilde{b}_p(\theta) \\ \tilde{c}^2(\theta) \end{bmatrix}$$

12.2 Estimation of the SV model with R

Function to obtain simulated moments

The first step to estimate the SV model is to write an R function that provides simulated moments for different parameter values θ. Instead of moments, we could also have simulated estimates of the parameters from an auxiliary model. In the following, we consider as auxiliary moments the sample variance, the sample kurtosis and the first-order autocorrelation for absolute log-returns $|y_t|$. Note that squared log-returns y_t^2 could be used as alternative to $|y_t|$. This R function, `sim_m_SV()`, is contained in the R file `sim_m_SV.R`.

The R function `sim_m_SV()`, which code is presented below, takes as input the parameter vector `par` and the simulated errors of both the observation equation and the transition equation, which are labeled `e`. Note that the error matrix `e` should contain two vectors of length H, which is labeled `H`, of $N(0,1)$ random variables. The output of the function is a vector, `output`, that contains the sample moments mentioned above for the SV model generated using the errors `e` and the parameter vector `par`.

```
sim_m_SV <- function(e,par){
```

From the parameter vector `par`, we obtain the parameters of the SV model ω, β and σ_η^2, which are labeled `omega`, `beta` and `sig2f` respectively. Also the length of the simulated series `H` is defined.

```
  omega <- par[1]
  beta <- exp(par[2])/(1+exp(par[2]))
  sig2f <- exp(par[3])

  H <- length(e[,1])
```

Using the error matrix `e` we obtain the error sequences $\{\epsilon_t\}_{t=1}^H$ and $\{\eta_t\}_{t=1}^H$, which are labeled `epsilon` and `eta` respectively. Note that we `eta` is obtained rescaling the vector of errors $e(:,2)$ because the variance of η_t is given by `sig2f`.

```
  epsilon <- e[,1]
  eta <- sqrt(sig2f)*e[,2]
```

As already seen in Chapter 9, the following lines of code generate a series from the SV model. The series is stored in the R vector `x`.

```
  x <- rep(0,H)
  f <- rep(0,H)

  f[1] <- omega/(1-beta)
  x[1] = exp(f[1]/2) * epsilon[1]

  for(t in 2:H){
    f[t] <- omega + beta * f[t-1] + eta[t]
    x[t] <- exp(f[t]/2) * epsilon[t]
  }
```

Finally, we define the vector `output` that contains the sample variance, sample kurtosis and first-order autocorrelation of the absolute values for the generated series `x`.

```
  xa <- abs(x)

  output <- c(var(x),kurtosis(x),cor(xa[2:H],xa[1:(H-1)]))
  return(output)
```

Estimation of the SV model

In the following, we shall see how to perform indirect inference for the SV model using R. The code described below is contained in the R file `estimate_SV_II.R`.

The observed data are contained in the R vector `x`. The first step is to obtain the moments for the real time series, i.e. the sample variance, kurtosis and first-order autocorrelation. These sample moments are stored in the vector `sample_m`.

```
n <- length(x)
xa <- abs(x)
sample_m <- c(var(x), kurtosis(x), cor(xa[2:n],xa[1:(n-1)]))
```

We then choose the length `H` of the simulations that will be used to obtain the simulated moments. In this case we set `H` to be 20 times the sample size T, labeled `n`. We also generate the errors matrix `e` that contains `H`×2 iid normal random draws with mean 0 and variance 1.

```
set.seed(123)
H <- 50*n
epsilon <- rnorm(H)
eta <- rnorm(H)
e <- cbind(epsilon,eta)
```

Here, we set the starting values for the minimization of the distance between the simulated vector of moments and the vector of moments from the real time series. The parameters are transformed with the inverse of the link functions used in `sim_m_SV()`.

```
b <- 0.90
sig2f <- 0.1
omega <- log(var(x))*(1-b)

par_ini <- c(omega,log(b/(1-b)),log(sig2f))
```

Finally, we perform the minimization with respect to `par` of the quadratic distance between the simulated vector of moments `sim_m_SV(e,par)` and the vector of moments `sample_m` from the real series. The indirect inference estimate of the parameter vector θ is then stored in the R vector `theta_hat` after appropriate transformation via link functions.

```
est <- optim(par=par_ini,fn=function(par) mean((sim_m_SV(e,par)-sample_m)^2), method = "BFGS")

omega_hat <- est$par[1]
beta_hat <- exp(est$par[2])/(1+exp(est$par[2]))
sig2f_hat <- exp(est$par[3])

theta_hat <- c(omega_hat,beta_hat,sig2f_hat)
```

12.3 Filtering paths for the SV model

When estimating a parameter-driven model by indirect inference, we obtain parameter estimates $\hat{\theta}_{TH}$, but we do not obtain a *filtered* path for the unobserved time-varying parameter, i.e. an estimate of the conditional variance. For example, in the case of the stochastic volatility model, we would like to obtain a filtered path for the unobserved sequence of variances $\{\sigma_t^2\}_{t=1}^T$.

In the master econometrics you will learn methods like the *Kalman filter* and the *Particle filter* that deliver not only filtered paths of the unobserved time-varying parameter but also log-likelihood values, from which it is possible to obtain maximum likelihood estimates. Those methods are conceptually simple, but often difficult to implement as they pose numerous computational challenges.

In this course, we shall consider a simple filtering method that delivers an approximate maximum likelihood path for the unobserved time-varying parameter. Let us focus first on the stochastic volatility example. Our objective would be to find the path $\{\sigma_t^2\}_{t=1}^T$ that maximizes the joint density of $(y_1, \dots, y_T, \sigma_1^2, \dots, \sigma_T^2)$, which we denote as $p(y_1, \dots, y_T, \sigma_1^2, \dots, \sigma_T^2; \theta)$. The sequence $\{\sigma_t^2\}_{t=1}^T$ that maximizes this joint density can shown to be the mode of the conditional density of $(\sigma_1^2, \dots, \sigma_T^2)$ given $(y_1, \dots, y_T)$. However, the problem is that such maximization problem would be cumbersome. Therefore we consider a sequential maximization procedure to obtain a filtered sequence for $\{\sigma_t^2\}_{t=1}^T$. The idea is that we start with a given value of σ_1^2. Then we can obtain that the joint distribution of y_2 and σ_2^2 given σ_1^2, which is

$$p(y_2, \sigma_2^2|\sigma_1^2) = p(y_2|\sigma_2^2)p(\sigma_2^2|\sigma_1^2).$$

Finally, we can find the value σ_2^2 that maximizes $p(y_2, \sigma_2^2|\sigma_1^2)$. This value corresponds to the mode of the conditional density of σ_2^2 given y_2 and σ_1^2. Once we have obtained σ_2^2, we can then consider σ_2^2 as given and, in the same way as before, obtain σ_3^2 and so on. In general, we can therefore recursively maximize

$$p(y_t, \sigma_t^2|\sigma_{t-1}^2) = p(y_t|\sigma_t^2)p(\sigma_t^2|\sigma_{t-1}^2)$$

over σ_t^2, which is indeed equivalent to maximize the log-density

$$\log p(y_t, \sigma_t^2|\sigma_{t-1}^2) = \log p(y_t|\sigma_t^2) + \log p(\sigma_t^2|\sigma_{t-1}^2).$$

As a result, we obtain that our sequential maximization problem becomes

$$\sigma_t^2 = \arg\max \left\{ \log p(y_t|\sigma_t^2) + \log p(\sigma_t^2|\sigma_{t-1}^2) \right\} \qquad \text{for every } t = 1, ..., T, \tag{12.1}$$

where the initial value σ_1^2 is fixed. For instance σ_1^2 can be set equal to the sample variance.

The conditional densities in the optimization in (12.1) are both available in closed form. In particular, the conditional density $p(y_t|\sigma_t^2)$ is determined by the observation equation

$$y_t = \sigma_t \varepsilon_t$$

Thus we have that the conditional distribution of y_t given the unobserved stochastic variance σ_t^2 is $y_t|\sigma_t^2 \sim N(0, \sigma_t^2)$. As a result the density function $p(y_t|\sigma_t^2)$ is given by

$$p(y_t|\sigma_t^2) = \frac{1}{\sqrt{2\pi\sigma_t^2}} \exp\left(-\frac{y_t^2}{2\sigma_t^2}\right).$$

Similarly, the density function $p(\sigma_t^2|\sigma_{t-1}^2)$ is determined by the transition equation

$$\sigma_t^2 = \exp(f_t), \quad f_t = \omega + \beta f_{t-1} + \eta_t$$

Therefore, we know that σ_t^2 given σ_{t-1}^2 has the following log-Normal distribution

$$\sigma_t^2|\sigma_{t-1}^2 \sim \log\text{-}N(\omega + \beta \log \sigma_{t-1}^2, \sigma_\eta^2).$$

As a result the density function of $\sigma_t^2|\sigma_{t-1}^2$ is given by

$$p(\sigma_t^2|\sigma_{t-1}^2) = \frac{1}{\sigma_t^2\sqrt{2\pi\sigma_\eta^2}} \exp\left(-\frac{(\log\sigma_t^2 - \omega - \beta\log\sigma_{t-1}^2)^2}{2\sigma_\eta^2}\right).$$

For the maximization problem in (12.1), we also note that maximizing with respect to σ_t is the same as maximizing with respect to f_t. Taking into account that $f_t = \log\sigma_t^2$ and plugging-in in (12.1) the corresponding density functions we obtain that the problem in (12.1) is equivalent to

$$f_t = \arg\min\left\{y_t^2\exp(-f_t) + 3f_t + \frac{(f_t - \omega - \beta f_{t-1})^2}{\sigma_\eta^2}\right\} \qquad \text{for every } t = 1, ..., T, \tag{12.2}$$

Then, once we have f_t, σ_t^2 is simply obtained as $\sigma_t^2 = \exp(f_t)$. Figure 12.1 shows the filtered path obtained using a stochastic volatility model for the S&P 500 index.

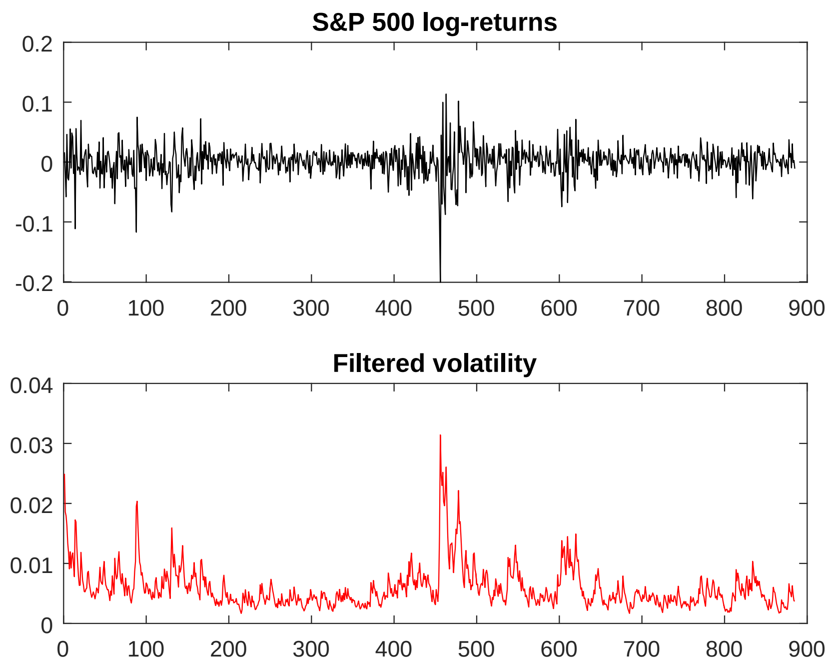

Figure 12.1: Weekly log-returns of S&P 500 and estimated volatility.

12.4 Filtering paths for SV model with R

In this section, we will see how to obtain the filtered volatility of an SV model with R. The first step is to write an R function that evaluates the expression in (12.2). We define the R function `filter_SV()`. This function takes as input y_t, f_t, f_{t-1} and θ, which are labeled `yt`, `ft`, `ft1` and `theta` respectively, and gives as output the value of the expression in (12.2), which is labeled `output`. The following code shows this function that is contained in the R file `filter_SV.m`.

```
filter_SV <- function(yt,ft,ft1,theta){
```

```
  omega <- theta[1]
  beta <- theta[2]
  sig2f <- theta[3]

  output <- yt^2*exp(-ft)+3*ft+(ft-omega-beta*ft1)^2/sig2f
  return(output)
}
```

Finally, the second step is to obtain the filtered f_t for each time period $t = 1, \ldots, T$. To obtain the filtered estimates of f_t we need to use a *for loop* and at each iteration find the value f_t that minimizes the function `filter_SV()`. The initial condition of the filter f_t at time $t = 1$ has to be chosen. In our case we use the logarithm of the sample variance. The R code below calculate the filtered estimate of f_t. You can find this code in the R file `estimate_SV_II.R`. We note that the indirect inference parameter estimate are give and stored in the R vector `theta_hat`. The filtered f_t are labeled `f`.

```
f <- rep(0,n)
f[1] <- log(var(x))

for(t in 2:n){
  ft_ini <- f[t-1]
  f_est <- optim(par=ft_ini ,fn= function(ft) filter_SV(x[t],ft,f[t-1],theta_hat), method = "BFGS")
  f[t] <- f_est$par
}
```