Lecture Notes: part 3

Financial Econometrics
2024-2025

Francisco Blasques
and
Paolo Gorgi

Contents

Part III

Dynamic regression models

Chapter 13

Dynamic regression models

Until now we have focused essentially on models with time-varying conditional variances and covariances. Parameter-driven and observation-driven models can however describe a wide range of other phenomena. In this chapter, we shall study regression models with time-varying regression coefficient. We will see that time varying regression is useful to derive a more flexible extension of the well-known Capital Asset Pricing Model (CAPM).

13.1 Observation-driven dynamic regression

Consider the following linear Gaussian regression model that describes the relation between a dependent variable y_t and an exogenous variable x_t

$$y_t = \beta x_t + \varepsilon_t \ , \quad \varepsilon_t \sim NID(0, \sigma^2), \tag{13.1}$$

where $\mathbb{E}(\varepsilon_t|x_t) = 0$. You have surely learned how to estimate and analyze this type of regression model in your introductory econometrics courses. The population regression coefficient β is the covariance between x_t and y_t divided by the variance of x_t, namely,

$$\beta = \frac{\mathbb{C}ov(y_t, x_t)}{\mathbb{V}ar(x_t)} = \frac{\sigma_{xy}}{\sigma_x^2}.$$

The regression coefficient β can be estimated by OLS and an estimate of the conditional expectation $\mathbb{E}(y_t|x_t)$

$$\hat{y}_t = \hat{\beta} x_t,$$

where $\hat{\beta}$ is the OLS estimate of β. The linear regression model is a powerful tool widely used in empirical analyses. When dealing with time series data, the relationship between the variables may be changing over time. These situations can be described by allowing the coefficient β to be constant over time.

As we have seen for the GARCH model, the specification of the time-varying parameter of an observation-driven model is based on past observations. For general time-varying parameter models, the Generalized Autoregressive Score (GAS) framework of Creal et al. (2013) provides a powerful approach to specify observation-driven models. The idea is to define the time-varying parameter as an autoregressive process with innovation given by the score of the predictive log-likelihood. The GAS approach provides us a specification of the time varying β_t. Note that we are not going to discuss the details of the GAS framework in this course, you will encounter more of these models in the Econometrics Master. The *observation-driven regression* model is given by

$$y_t = \beta_t x_t + \varepsilon_t,$$

$$\beta_t = \omega + \phi \beta_{t-1} + \alpha (y_{t-1} - \beta_{t-1} x_{t-1}) x_{t-1},$$

where $\{\varepsilon_t\}_{t\in\mathbb{Z}}$ is a $NID(0, \sigma^2)$ sequence independent of $\{x_t\}_{t\in\mathbb{Z}}$, and ω, ϕ, α are parameters that determine the dynamic properties of β_t. We can see that the time-varying regression coefficient β_t depends only on past values of y_t and x_t. The regression coefficient β_t is updated using $(y_{t-1} - \beta_{t-1}x_{t-1})x_{t-1} = \varepsilon_{t-1}x_{t-1}$, which can be seen as an estimate of the covariance between the error term ε_{t-1} and the regressor x_{t-1}.

The conditional expectation of y_t is given by $\mathbb{E}(y_t|Y^{t-1}, X^t) = \beta_t x_t$. Note that we condition on past values of y_t, $Y^{t-1} = \{y_{t-1}, y_{t-2}\dots\}$, and on the current and past values of x_t, $X^t = \{x_t, x_{t-1}\dots\}$. Instead, we have that the conditional variance is constant and given by $\mathbb{V}ar(y_t|Y^{t-1}, X^t) = \sigma^2$. Finally, the conditional distribution of y_t is $y_t|(Y^{t-1}, X^t) \sim N(\beta_t x_t, \sigma^2)$.

13.1.1 Maximum likelihood estimation

We now focus on the estimation of the parameter vector $\theta = (\omega, \phi, \alpha, \sigma^2)^\top$. We can estimate θ by Maximum Likelihood. The likelihood function is in closed form. In particular, the conditional density function of y_t is given by

$$p(y_t|Y^{t-1}, X^t) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(y_t - \beta_t x_t)^2}{2\sigma^2}\right).$$

Therefore, we can write the log-likelihood function as the sum of the conditional log-densities. The log-likelihood function is

$$L_T(\theta) = \sum_{i=2}^{T} -\frac{1}{2}\left(\log\sigma^2 + \frac{(y_t - \beta_t x_t)^2}{\sigma^2}\right),$$

where the time-varying coefficient β_t is obtained using the updating equation

$$\beta_t = \omega + \phi\beta_{t-1} + \alpha(y_{t-1} - \beta_{t-1}x_{t-1})x_{t-1}.$$

The initialization β_1 can be set equal to the unconditional expectation of β_t, i.e. $\beta_1 = \omega/(1-\phi)$. The ML estimator is defined as

$$\hat{\theta}_T = \arg\max_{\theta\in\Theta} L_T(\theta).$$

13.1.2 ML estimation with R

In this section, we see how to estimate the observation-driven regression model by maximum likelihood using R. More specifically, we are only going to discuss how to write the log-likelihood function because the optimization is then equivalent to what we have seen for GARCH models in the first part of the course. The R file `estimate_OD_reg.R` contains the code to optimize the log-likelihood function. The log-likelihood function is labeled `llik_OD_regression(,)` and it is contained in the file `llik_OD_regression.R`. We need to create an R function that takes as argument the observed time series y_t and x_t, labeled `y` and `x`, and a parameter vector, labeled `par`, and gives as output the average log-likelihood value, labeled `llik`.

The first line of code defines the name of the function, the input and the output.

```
llik_OD_regression <- function(y,x,par){
```

The time series length is obtained and each parameter value is set equal to an element of the input parameter vector `par` using appropriate link functions.

```
  n <- length(x)
  omega <- par[1]
  phi <- exp(par[2])/(1+exp(par[2]))
  alpha <- exp(par[3])
  s2 <- exp(par[4])
```

Now, the time-varying β_t is initialized using its unconditional expectation.

```
beta <- rep(0,n)
beta[1] <- omega/(1-phi)
```

A *for loop* allows us to obtain recursively the time-varying β_t using the updating equation of the observation-driven regression model.

```
for(t in 2:n){
  beta[t] <- omega+phi*beta[t-1]+alpha*(y[t-1]-beta[t-1]*x[t-1])*x[t-1]
}
```

Finally, the average log-likelihood is computed.

```
l <- -(1/2)*log(s2)-(1/2)*(y-beta*x)^2/s2
llik <- mean(l)
return(llik)
```

The full code to obtain the log-likelihood function is given below.

```
llik_OD_regression <- function(y,x,par){
  n <- length(x)
  omega <- par[1]
  phi <- exp(par[2])/(1+exp(par[2]))
  alpha <- exp(par[3])
  s2 <- exp(par[4])

  beta <- rep(0,n)
  beta[1] <- omega/(1-phi)
  for(t in 2:n){
    beta[t] <- omega+phi*beta[t-1]+alpha*(y[t-1]-beta[t-1]*x[t-1])*x[t-1]
  }

  l <- -(1/2)*log(s2)-(1/2)*(y-beta*x)^2/s2
  llik <- mean(l)
  return(llik)
}
```

13.1.3 Estimating β_t with R

Once we have the ML estimate $\hat{\theta}_T$, we can obtain the filtered coefficient β_t by running the updating equation evaluated at $\hat{\theta}_T$. The R code to obtain the filtered coefficient β_t is

```
n <- length(xt)
beta <- rep(0,n)
beta[1] <- omega_hat/(1-phi_hat)

for(t in 2:n){
  beta[t] <- omega_hat+phi_hat*beta[t-1]+alpha_hat*(yt[t-1]-beta[t-1]*xt[t-1])*xt[t-1];
}
```

Figure 13.1 below shows the time series and the estimated β_t.

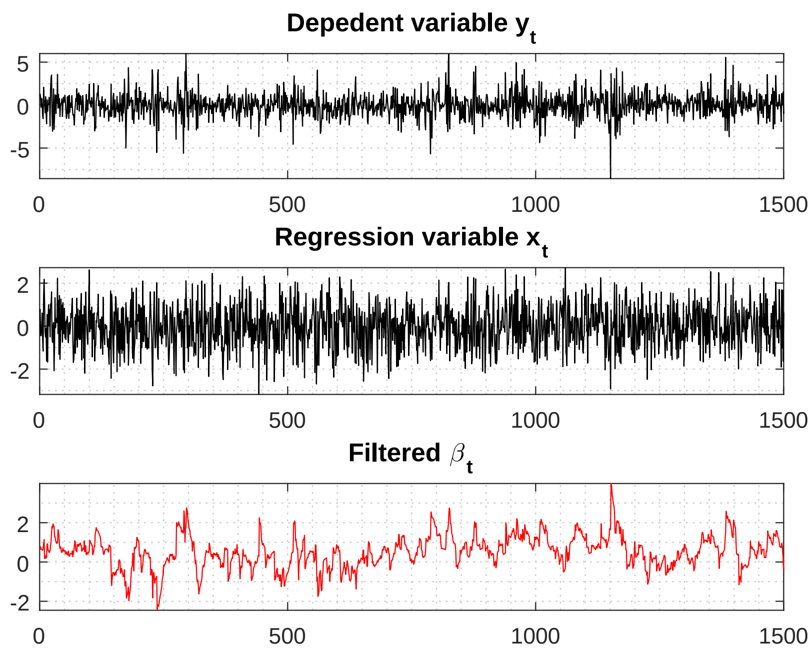

Figure 13.1: Time series of y_t and x_t and estimated β_t.

13.2 Parameter-driven dynamic regression

An alternative approach to the previous section is to consider a parameter-driven regression model. The idea is to specify β_t as an unobserved autoregressive Gaussian process. The *parameter-driven regression* model is given by

$$\begin{aligned} y_t &= \beta_t x_t + \varepsilon_t \ , & \varepsilon_t &\sim N(0, \sigma_\varepsilon^2) \\ \beta_t &= \alpha_0 + \alpha_1 \beta_{t-1} + \eta_t \ , & \eta_t &\sim N(0, \sigma_\eta^2) \end{aligned}$$

Here the relationship between y_t and x_t is allowed to change at each point in time and the dynamic of this variation is described by the transition equation $\beta_t = \alpha_0 + \alpha_1 \beta_{t-1} + \eta_t$. Indeed the dynamic coefficient β_t is a stochastic unobserved process, which is not constant even conditioning on the entire past of y_t and x_t. For the estimation of the model, it can be shown that a closed-form of the likelihood function can be derived using the Kalman Filter. We are not going to discuss Kalman Filtering in this course and you will study it in the Econometrics Master. Alternatively, as discussed for the SV model, indirect inference is a viable way for estimation. In the following, we will see estimation of the *parameter-driven regression* by indirect inference.

13.2.1 Dynamic regression estimation by indirect inference

In practice, we observe the sequences $\{y_t\}_{t=1}^T$ and $\{x_t\}_{t=1}^T$. The estimation of the dynamic regression model is very similar as for the SV model but we need to take care of some additional issues. First note that we are not specifying any model for x_t and therefore, when we generate data for y_t, we need to keep in mind that we have to use the observed x_t and we cannot generate the x_t. This leads to the fact that the maximum length of a generated series of y_t is T because the length of x_t is T as well.

Fortunately, what we can do is to choose the size H of the simulated paths to be a multiple of T, let's say $H = M \times T$ where $M \in \mathbb{N}$. Then, we can generate M series of length T for y_t and for each of these series obtain several auxiliary statistics $\tilde{B}_i(\theta)$, $i = 1, \dots, M$. Then we can average these statistics and obtain the *auxiliary statistic* $\tilde{B}_H(\theta) = \frac{1}{M} \sum_{i=1}^M \tilde{B}_i(\theta)$. Note that the total length of the simulations for the *auxiliary statistic* $\tilde{B}_H(\theta)$ is H because each $\tilde{B}_i(\theta)$ is obtained from simulations of length T and we have M of these, thus $H = M \times T$ in total. Once we have our *auxiliary statistic* $\tilde{B}_H(\theta)$, we can just proceed with indirect inference as discussed in the previous chapter.

Finally, the last issue we need to address is what auxiliary statistics are a reasonable choice for our dynamic regression model. As noted before many different options are available but the accuracy of the indirect inference estimator may be strongly affected by a poor choice. We propose the following set of auxiliary statistics that seems to provide rather accurate results. The auxiliary statistic $\hat{B}_T$ for the observed data can be obtained as follows. First we run a static regression, as the one described in (13.1), and obtain the OLS estimate $\hat{\beta}$. Second we calculate the residuals of this regression as $\hat{e}_t = y_t - \hat{\beta} x_t$. We then obtain the series $\{\hat{e}_t x_t\}_{t=1}^T$ and estimate the autocovariance function for a certain number p of lags, namely $\hat{\gamma}_l = T^{-1} \sum_{i=1}^T \hat{e}_t x_t \hat{e}_{t-l} x_{t-l}$ for $l = 0, 1, \dots, p$. We define the auxiliary statistic from the sample

$$\hat{B}_T = \begin{bmatrix} \hat{\beta} \\ \hat{\gamma}_0 \\ \vdots \\ \hat{\gamma}_p \end{bmatrix} .$$

In an equivalent way, we can also get the auxiliary statistic $\tilde{B}_H(\theta)$ from the simulated data. In particular, we have M generated series of length T, $\{\tilde{y}_{i,t}(\theta)\}_{t=1}^T$, $i = 1, \dots, M$, and for each of these series we obtain the OLS regression coefficient $\tilde{\beta}_i(\theta)$. We then obtain the residuals as $\tilde{e}_{i,t}(\theta) = \tilde{y}_{i,t} - \tilde{\beta}_i(\theta) x_t$ and the series $\{\tilde{e}_{i,t}(\theta) x_t\}_{t=1}^i$ for $i = 0, \dots, M$. Finally, we calculate the autocovariance function for these series as $\tilde{\gamma}_{l,i}(\theta) = T^{-1} \sum_{t=1}^T \tilde{e}_{i,t}(\theta) x_t \tilde{e}_{i,t-l}(\theta) x_{t-l}$ for $l = 0, \dots, p$ and $i = 1, \dots, M$. We define the auxiliary statistic from the simulated data as

$$\tilde{B}_H(\theta) = \frac{1}{M}\begin{bmatrix} \sum_{i=1}^{M} \tilde{\beta}_i(\theta) \\ \sum_{i=1}^{M} \tilde{\gamma}_{0,i}(\theta) \\ \vdots \\ \sum_{i=1}^{M} \tilde{\gamma}_{p,i}(\theta) \end{bmatrix}.$$

13.2.2 Estimation of dynamic regression with R

Function to obtain simulated moments

The first step to estimate the dynamic regression 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 the auxiliary moments proposed in the previous section. The R function `sim_m_REG()` is contained in the R file `sim_m_REG.R`.

The R function `sim_m_REG()`, which code is presented below, takes as input a parameter vector, labeled `par`, the observed regressor $\{x_t\}_{t=1}^T$, which are labeled `x`, 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 $H = M \times T$, 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 in the previous section for the dynamic regression model generated using the errors `e`, the regressor `x` and the parameter vector `par`.

```
sim_m_REG <- function(e,x,par){
```

From the parameter vector `theta`, we obtain the parameters of the dynamic regression model α_0, α_1, σ_η^2 and σ_ε^2, which are labeled `a0`, `a1`, `s_eta` and `s_eps` respectively. Also the length of the simulated series `H` is defined, which is `M` times `T`.

```
  a0 <- par[1]
  a1 <- exp(par[2])/(1+exp(par[2]))
  s_eta <- exp(par[3])
  s_eps <- exp(par[4])

  output <- 0

  n <- length(x)
  H <- length(e[,1])
  M <- H/n
```

Using the error matrix `e` we obtain the error sequences $\{\varepsilon_t\}_{t=1}^H$ and $\{\eta_t\}_{t=1}^H$, which are labeled `eps` and `eta` respectively. Note that `eps` and `eta` are obtained rescaling the vector of errors `e[,1]` and `e[,2]` because the variances of ε_t and η_t are given by `s_eps` and `s_eta` respectively.

```
  eta <- sqrt(s_eta)*e[,1]
  eps <- sqrt(s_eps)*e[,2]
```

Now we need to remember that $\{x_t\}_{t=1}^T$ is given and not generated and therefore, as discussed before, we have to generate and obtain moments for `M` series of length `n`. In particular, we use a *for loop* with `M` iterations where, at each iteration, we generate a series of y_t of length `n`. The generated series is stored in the R vector `y`. This is exactly what the following code does. Note that the elements in the error vectors `eps` and `eta` need to be carefully selected at each iteration of the loop. This because `eps` and `eta` have length `H` and at the first iteration we need the first `n` elements, at the second we need the elements from `n`+1 to 2`n` and so on.

```
  for(m in 1:M){

    b <- rep(0,n)
    b[1] <- a0/(1-a1)

    for(t in 2:n){
      b[t] <- a0+a1*b[t-1]+eta[(m-1)*n+t]
    }

    y <- b*x+eps[((m-1)*n+1):(m*n)]
```

Finally, we obtain the moments for the simulated series as discussed in the previous section. We compute the regression coefficient between the simulated `y` and `x` (remember `x` is not simulated) and we store it in the R object `hb`. We then obtain the residuals of this regression, which are labeled `yr` and obtain the sequence $\{\tilde{y}_t(\theta)x_t\}_{t=1}^{T}$, which is labeled `xy`. Next, we obtain the autocovariance function for `xy`. Note that in this case we consider 15 lags, i.e. $p = 15$. Finally, we store the resulting simulated auxiliary statistic in the vector `output` by take the average over `M`, as discussed in the previous section.

```
    hb <- cov(y,x)/var(x)
    yr <- y-hb*x
    xy <- yr*x
    acvfxy <- acf(xy, lag.max=15, type ="covariance", plot=F)$acf[-1]

    output <- output+c(var(yr),hb,acvfxy)/M
  }
   return(output)
}
```

Estimation of the dynamic regression model

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

The observed data $\{y_t\}_{t=1}^{T}$ and $\{x_t\}_{t=1}^{T}$ are contained in the R vectors `y` and `x` respectively. The first step is to obtain the auxiliary statistic for the real time series, i.e. $\hat{B}_T$. In particular, the auxiliary statistics are those discussed in the previous section and they are stored in the vector `sample_R`. Note that the number of lags of the autocovariance of `xy` must be the same as considered in the simulations. In this case $p = 15$.

```
hb <- cov(yt,xt)/var(xt)
yr <- yt-hb*xt

xy <- yr*xt
acvfxy <- acf(xy, lag.max=15, type ="covariance", plot=F)$acf[-1]

sample_m <- c(var(yr),hb,acvfxy)
```

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

```
n <- length(xt)
M <- 20
H <- M*n
```

```
set.seed(123)
eta <- rnorm(H)
eps <- rnorm(H)
e <- cbind(eta,eps)
```

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.

```
a1 <- 0.95
a0 <- cov(xt,yt)/var(xt)*(1-a1)
s_eta <- 0.2
s_eps <- var(yr)

par_ini <- c(a0, log(a1/(1-a1)), log(s_eta), log(s_eps))
```

Finally, we perform the minimization with respect to `par` of the quadratic distance between the simulated vector of auxiliary statistics `sim_m_REG(e,par)` and the vector of auxiliary statistics `sample_m` from the real series. The indirect inference estimate of the parameter vector θ is then stored in the R vector `theta_hat`.

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

a0_hat <- est$par[1]
a1_hat <- exp(est$par[2])/(1+exp(est$par[2]))
s_eta <- exp(est$par[3])
s_eps <- exp(est$par[4])
theta_hat <- c(a0_hat,a1_hat,s_eta,s_eps)
```

13.3 Dynamic Capital Asset Pricing Model

The capital asset pricing model (CAPM) provides a stylized description of how financial markets price securities. The CAPM provides a description of expected return on financial investments that is based on a quantification of risk. The CAPM assumes that there are no sources of risk except the systematic market risk as the idiosyncratic risk of a specific asset can be eliminated through diversification. In financial markets dominated by risk-averse investors, higher-risk securities are priced to yield higher expected returns than lower-risk securities. The expected return of an asset depends on the exposition to the market risk: higher risk, higher returns. This positive relationship between market risk exposition and return is described by the following relation

$$R_i = r^f + \beta_i(R^m - r^f),$$

where R_i is the expected return of asset i, β_i is the beta asset i, R^m is the expected return of the market, and r^f is the risk free rate. The β_i coefficient is a measure of systematic risk for stock i. It describes the exposition of the asset to the market risk. A stock with a beta equal to 1 is as risky as the market and therefore it has the same expected return. A stock with $\beta_i < 1$ is less risky than the market and it has a lower expected return. In contrast, a stock with $\beta_i > 1$ is more risky than the market and therefore it has an higher expected return than the market.

The CAPM can be expressed through a regression model

$$r_{i,t} = r^f + \beta_i(r_t^m - r^f) + \varepsilon_{i,t}$$

where $r_{i,t}$ is the return of stock i at time t, r^f is the risk-free rate, r_t^m is the market return, $\varepsilon_{i,t}$ is an idiosyncratic error (i.e. specific to stock i). From this regression models, we can see that the risk (variance) of asset i can be decomposed as follows:

$$\mathbb{V}ar(r_{i,t}) = \beta_i^2 \mathbb{V}ar(r_t^m) + \mathbb{V}ar(\varepsilon_{i,t}),$$

where $\mathbb{V}ar(r_t^m)$ is the systematic risk (market risk) and $\mathbb{V}ar(\varepsilon_{i,t})$ is the idiosyncratic risk (diversifiable risk). Only exposition to systematic risk gives higher returns as idiosyncratic risk can be eliminated through diversification. In order to estimate the CAPM model, in practice, the market returns r_t^m are measured through the use of a broad market index, such as the S&P 500, and the risk free returns r^f are measured using some short-term treasury bills (r^f may be time varying r_t^f). The estimate of the stock's beta β_i can be obtained by OLS regressing the excess of returns of the asset $r_{i,t} - r^f$ on the excess of returns of the market $r_t^m - r^f$.

The CAPM model assumes that the β_i of a stock does not change over time. However, this assumption is somewhat restrictive and often rejected by the data. The standard CAPM can be extended by introducing a time-varying $\beta_{i,t}$. We can consider our *observation-driven regression* model and obtain the following dynamic CAPM regression

$$r_{i,t} = r^f + \beta_{i,t}(r_t^m - r^f) + \varepsilon_{i,t},$$

$$\beta_{i,t} = \omega + \phi\beta_{i,t-1} + \alpha(r_{i,t-1} - r^f - \beta_{i,t-1}(r_{t-1}^m - r^f))(r_{i,t-1} - r^f),$$

where $\{\varepsilon_t\}_{t\in\mathbb{Z}}$ a $NID(0, \sigma^2)$. The time-varying betas allows the model to describe changes in the exposition of a stock to the market risk over time. The specification of $\beta_{i,t}$ may also be based on the *parameter-driven regression* model.

An alternative way is to obtain a time-varying beta indirectly from a bivariate GARCH model. In particular, in the standard CAPM, the beta is given by $\beta_i = \frac{\mathbb{C}ov(r_t^m, r_{i,t})}{\mathbb{V}ar(r_t^m)} = \frac{\sigma_{m,i}}{\sigma_m^2}$. Therefore, we can estimate the time varying covariance matrix from a bivariate GARCH with the excess of returns of the asset $r_{i,t} - r^f$ and the excess of returns of the market $r_t^m - r^f$. The time-varying beta is then obtained from the conditional covariance $\sigma_{m,i,t}$ and the conditional variance $\sigma_{m,t}^2$, that is, $\beta_{i,t} = \frac{\sigma_{m,i,t}}{\sigma_{m,t}^2}$.