Numerical Methods in Finance
Problem Set 2

Dr. Hong Lan, PhD
School of Finance, UIBE

December 18, 2017

Instructions: All the questions can be answered in either English or Chinese. This problem set is due on January 03, 2018.

1. **Projection Method with Chebyshev Polynomials** [70 points]

   We have shown in the section that, the Lucas (1978) model is characterized by the following asset pricing equation

$$p_t = \beta \mathbb{E}_t \left[\left(\frac{d_{t+1}}{d_t} \right)^{-\gamma} (p_{t+1} + d_{t+1}) \right] \tag{1}$$

$$d_t = \mu_d + \rho d_{t-1} + \epsilon_t, \quad \epsilon_t \sim N(0, \sigma^2) \tag{2}$$

   Now consider to solve the above asset pricing equation by projecting p_t onto a set of the Chebyshev polynomials, i.e., let

$$p_t = T_n(d_t; a_n) \tag{3}$$

   and to approximate the expectation using as usual the Gauss-Hermite quadratures. The domain of the Chebyshev polynomials is $[-1, 1]$, and we will choose the roots (nodes) of the Chebyshev polynomials that lie in $[-1, 1]$ as our grid points of d_t.

   To construct such a grid, first choose the order n of the Chebyshev polynomials we would like to use as the projection basis, then choose $n+1$ roots to ensure uniqueness. The function `chebnode.m` generates the roots once n is chosen. We also need choose the lower and upper bounds of d_t, denoted as d_{min} and d_{max}. The function `scalup.m` maps the chosen roots that lie in $[-1, 1]$ to the points within $[d_{min}, d_{max}]$. The function `scaldown.m` reverses that mapping by translating the points in $[d_{min}, d_{max}]$ back into the domain $[-1, 1]$.

   The Chebyshev polynomials can be generated via the following recursion

$$T_{n+1}(x) = 2xT_n(x) - T_{n-1}, \quad n = 1, 2, \ldots \tag{4}$$

   The first few Chebyshev polynomials writes

$$T_0(x) = 1 \tag{5}$$

$$T_1(x) = x \tag{6}$$

$$T_2(x) = 2x^2 - 1 \tag{7}$$

$$T_3(x) = 4x^3 - 3x \tag{8}$$

For example, if we decide to use Chebyshev polynomials up to the second order to approximate p_t, then we need use `chebnode.m` to generate three roots $\tilde{d}_1, \tilde{d}_2$ and $\tilde{d}_3 \in [-1, 1]$, the corresponding value of p_t at the three roots, denoted by p_1, p_2 and p_3 are

$$\begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} = \underbrace{\begin{bmatrix} 1 & \tilde{d}_1^2 & 2\tilde{d}_1^2 - 1 \\ 1 & \tilde{d}_2^2 & 2\tilde{d}_2^2 - 1 \\ 1 & \tilde{d}_3^2 & 2\tilde{d}_3^2 - 1 \end{bmatrix}}_{T} \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{9}$$

where the T matrix collects the Chebysheve polynomials up to the second order, evaluated at $\tilde{d}_1, \tilde{d}_2$ and $\tilde{d}_3$, and the function `chebpol.m` generates this T matrix once the roots are provided. The coefficients a_1, a_2 and a_3 are determined by minimizing the sum of the squared difference between the LHS and RHS of the asset pricing equation.

What do you need to do are as follows

(a) [25 points] Fill out lines 16 and 19 in `errfunc.m` with your code, so the program can compute the coefficients of the Chebyshev polynomials for p_t. Then fill out line 67 in `main_projection.m` with your code, so the program can plot p_t.

(b) [20 points] Fill out lines 14 and 17 in `erpfunc.m` with your code, so the program can compute the value of x_t, i.e., the risk premium. Then fill out line 75 in `main_projection.m` with your code, so the program can compute the coefficients of the Chebyshev polynomials for x_t.

(c) [20 points] Fill out line 93 in `main_projection.m` with your code, so the program can compute the simulations of x_t.

(d) [5 points] Comment on the role of parameter γ in your solution.

2. **Importance Sampling in Computing Bayesian Inferences** [30 points]

In Bayesian estimation, we often need to compute the expected value of a function with respect to a probability density, e.g.,

$$\mathbb{E}[g(\theta)|y] = \int g(\theta)p(\theta|y)d\theta \tag{10}$$

where y is the dataset that follows a certain distribution with parameters of that distribution collected in θ. For example, if $y \sim N(\mu, \sigma^2)$, then $\theta = \{\mu, \sigma^2\}$ is the set collecting the mean and variance of the normal distribution. The difficulty that arises in computing (10) is we cannot sample directly from the density $p(\theta|y)$. To solve this problem, we invoke the Bayes' law

$$p(\theta|y) = \frac{p(\theta)p(y|\theta)}{p(y)} \tag{11}$$

where $p(\theta|y)$ is the posterior we want to compute, $p(\theta)$ the prior representing our initial belief of the distribution of θ and $p(y|\theta)$ is, given θ, the likelihood of observing y. $p(y)$ is a marginal density and is a constant with respect to θ. Therefore, we often write

$$p(\theta|y) \propto p(\theta)p(y|\theta) = \tilde{p}(\theta|y) \tag{12}$$

where $\tilde{p}(\theta|y)$ is a kernel of the posterior, i.e., $\tilde{p}(\theta|y)$ is proportional to the posterior up to a multiplicative constant

$$p(\theta|y) = \frac{1}{c_p}\tilde{p}(\theta|y) \tag{13}$$

As $p(\theta|y)$ integrates to one, it follows that

$$c_p = \int \tilde{p}(\theta|y)d\theta \tag{14}$$

The problem (10) now rewrites

$$\mathbb{E}[g(\theta)|y] = \int g(\theta)p(\theta|y)d\theta = \frac{\int g(\theta)\tilde{p}(\theta|y)d\theta}{c_p} = \frac{\int g(\theta)\tilde{p}(\theta|y)d\theta}{\int \tilde{p}(\theta|y)d\theta} \tag{15}$$

Let $q(\theta)$ be the proposal density (the density from which we sample θ's), then (15) rewrites

$$\mathbb{E}[g(\theta)|y] = \frac{\int g(\theta)\tilde{p}(\theta|y)d\theta}{\int \tilde{p}(\theta|y)d\theta} = \frac{\int g(\theta)\frac{\tilde{p}(\theta|y)}{q(\theta)}q(\theta)d\theta}{\int \frac{\tilde{p}(\theta|y)}{q(\theta)}q(\theta)d\theta} = \frac{\int g(\theta)w(\theta)q(\theta)d\theta}{\int w(\theta)q(\theta)d\theta} \tag{16}$$

where

$$w(\theta) = \frac{\tilde{p}(\theta|y)}{q(\theta)} \tag{17}$$

denotes the importance weights. The Monte Carlo estimation of (16) writes

$$I_g = \frac{\frac{1}{n}\sum_{i=1}^{n} g(\theta_i)w(\theta_i)}{\frac{1}{n}\sum_{i=1}^{n} w(\theta_i)} = \frac{\sum_{i=1}^{n} g(\theta_i)w(\theta_i)}{\sum_{i=1}^{n} w(\theta_i)} \tag{18}$$

where θ_i's are sampled from the proposal density $q(\theta)$. Often in practice, we define the standardized weights as follows

$$\omega_i = \frac{w(\theta_i)}{\sum_{i=1}^{n} w(\theta_i)} \tag{19}$$

then the estimator (18) rewrites

$$I_g = \sum_{i=1}^{n} g(\theta_i)\omega_i \tag{20}$$

Given (20), the posterior moments (Bayesian inferences) can be calculated given the functional form of $g(\theta)$. For example, if we want to compute the posterior mean of θ, we can let

$$g(\theta_i) = \theta_i \tag{21}$$

and the posterior mean estimator writes

$$\mu_\theta = \sum_{i=1}^{n} \theta_i \omega_i \tag{22}$$

given (22), the estimator of posterior variance of θ writes

$$\sigma_\theta^2 = \sum_{i=1}^{n} (\theta_i - \mu_\theta)^2 \omega_i \tag{23}$$

and the higher moments can be likewise calculated.

Now we have a dataset y that follows the t distribution, i.e., $y \sim t(\mu, \sigma^2, \nu)$ with μ being the mean, σ^2 variance and ν the degree of freedom. In this case, θ is the set collecting all the three distribution parameters, i.e., $\theta = \{\mu, \sigma^2, \nu\}$. For simplicity, we only compute the posterior mean and variance of μ. For that purpose, we need to build the likelihood function, to choose a prior of μ, to construct the proposal density and to compute the weights. Questions below will guide you through the whole procedure. What you need do is to fill out several lines in `main_bayesian.m` with your code.

(a) [5 points] Evaluate the proposal density of μ:

1. Line 30: Use the maximum likelihood estimation to get the estimate of the t distribution parameters, call them μ_{mle}, σ_{mle}^2 and ν_{mle} and report their values;(Hint: function `llt.m` computes the log likelihood of the t distribution.)
2. Line 47: The previous step would also give you the standard error of μ_{mle}, call it s_μ and report its value (Hint: one way to compute s_μ is to make use of the Hessian matrix from your maximum likelihood estimation!);
3. Line 57: Consider to use a normal proposal density with μ_{mle} and $(3s_\mu)^2$ as its mean and variance, i.e., μ that sampled from this proposal density, call it μ_{prop}, follows the normal distribution $\mu_{prop} \sim N(\mu_{mle}, (3s_\mu)^2)$. Sample a sequence of $\mu_{prop,i}$'s from this proposal density.
4. Line 60: Use the $\mu_{prop,i}$'s you just sampled to compute the (log) proposal density. (Hint: function `lpdfn.m` computes the log density of a normal distribution.)

(b) [5 points] Line 68: Evaluate the prior density: consider to use a *normal* prior with mean μ_0 and variance σ_0^2, i.e., we initially believe that $\mu \sim N(\mu_0, \sigma_0^2)$. Use μ_0, σ_0^2 and the $\mu_{prop,i}$'s you just sampled, compute the (log) prior density.

(c) [5 points] Line 82: Evaluate the likelihood $p(y|\theta)$, note this is the likelihood of the t distribution: as we only want to compute the posterior mean and variance of μ, the likelihood reduces to $p(y, \sigma^2, \nu|\mu)$. Compute the likelihood with y and the $\mu_{prop,i}$'s, using σ_{mle}^2 and ν_{mle} as the variance and degree of freedom for your computation.

(d) [5 points] Line 85: With the prior and the likelihood, following (12) to compute your kernel of the posterior with the $\mu_{prop,i}$'s.

(e) [5 points] Line 91: With the proposal density and the kernel you just computed, following (17) and (19) to compute the weights and the standardized weights.

(f) [5 points] By far the weights and the $\mu_{prop,i}$'s are readily computed, the program will then compute the posterior mean and variance of μ using (22) and (23), and will plot the prior and the posterior kernel. Interpret the plot.

References

LUCAS, R. (1978): “Asset Prices in an Exchange Economy,” Econometrica, 46(6), 1429–45.