Estimating Factor Models with Hierarchical Bayes

ECE 5412 Course Project, Jinyuan (John) Yu, 05/2022, NetID: jy478

Abstract

The factor model is an important tool for both portfolio managers and researchers in modern finance. In this report, we illustrate a Hierarchical Bayesian approach using Gibbs sampling to estimate linear regression models of stock returns, which can substantially improve the accuracy of parameter estimates using cross-sectional information.

(The source codes: https://github.com/JohnYu0510/ECE5412/tree/main/HierarchicalBayes)

1 Introduction

Factor models have become the standard research framework to study the asset returns in modern finance. These models employ multiple factors in its calculations to explain market phenomena and/or equilibrium asset prices. Usually, the models describe the covariation of asset returns via the following form:

$$y_{jt} = \alpha_j + f_t\beta_j + \epsilon_{jt}, \epsilon_{jt} \sim N(0, v_j)\ (j = 1, \dots, J; t = 1, \dots, T)$$

In the above function, y_{jt} is the stock return of firm j at time t, f_t is a matrix containing predictive signals (e.g., the market index) which are related to the economy, and ϵ_{jt} is a white noise process. The parameter α_j describes the consistent mispricing of the stock that cannot be captured by factors, β_j is a vector of the risk exposures to different factors, and v_j is the idiosyncratic risk.

Portfolio managers aim to estimate (α_j, β_j, v_j) with stock returns and factors data, and use the model's prediction to select portfolios that achieve optimal trade-off between risk and returns. However, traditional portfolio optimization algorithms are quite sensitive to the input parameters, and reducing the estimator error becomes necessary. In this report, a hierarchical factor model with a Bayesian estimation procedure are introduced to improve the accuracy of estimation.

2 The Hierarchical Factor Model

2.1 Model Set-up

Suppose we have a standard-form linear factor model as described above:

$$y_{jt} = \alpha_j + f_t\beta_j + \epsilon_{jt}, \epsilon_{jt} \sim N(0, v_j)\ (j = 1, \dots, J; t = 1, \dots, T)$$

To better characterize the market features, we add hyperparameters in our models to incorporate cross-sectional information. Let $\tau_j = \log(v_j)$, we assume that $(\alpha_j, \beta_j, \tau_j)$ are related to covariates $(z_j^\alpha, z_j^\beta, z_j^\tau)$, which can contain variables like financial ratios, accounting numbers, and sector indicators. The relationship is still linear:

$$\alpha_j = \theta_0' z_j^\alpha + u_{j0}, u_{j0} \sim N(0, \Lambda_0)$$

$$\beta_{jk} = \theta_0' z_j^\beta + u_{jk}, u_{jk} \sim N(0, \Lambda_k)$$

$$\tau_{jk} = \psi' z_j^\tau + w_j, w_j \sim N(0, \delta)$$

Assume the factors follow a stationary joint normal distribution: $f_t \sim N(\mu_f, \Omega_f)$. Based on all these assumptions, the moments of returns that we are interested in can be given by:

$$E(y_{jt}) = \alpha_j + \mu_f'\beta_j; Var(y_{jt}) = \beta_j'\Omega_f\beta_j + v_j; Cov(y_{jt}, y_{kt}) = \beta_j'\Omega_f\beta_k$$

If we can estimate the model above, we can apply these to construct optimal portfolio with the asset returns' moments.

2.2 Prior Distributions on Hyperparameters

To complete the specification of the factor model above, we need to generate an initial belief in the hyperparameters' values to start the Bayesian inference. The hyperparameters here include $\{\theta_k, \Lambda_k\}, \psi, \delta$ for the cross section of companies, and the moments of factors μ_f, Ω_f.

In order to make the posterior distributions depend more on the data, we use comparatively non-informative priors. For $\theta, \psi, \mu_f, \Omega_f^{-1}$, we use $U[-\infty, \infty]$. The priors on the variances Λ, δ are assumed to be very diffuse, which are taken to be $IG(0,0.1)$, where IG denotes the inverse gamma distribution.

The joint posteriors of all the parameters above can hardly be evaluated analytically, so we use the Gibbs sampling, a special case of the Metropolis-Hasting algorithm, to simulate samples of the posteriors. Using this method, we can estimate the parameters numerically.

3 The Markov Chain Monte Carlo Estimation

3.1 The Full Conditional Posterior Distributions

Given the data and priors, the posterior distributions of all the parameters within the model are given in the table below:

Table 1 Posterior Distributions of Parameters

	Priors	Posteriors
μ_f	$U[-\infty, \infty]$	$N(\overline{f_t}, \overline{\Omega_f})$
Ω_f	$\Omega_f^{-1} \sim U[-\infty, \infty]$	$W(\frac{1}{2}\Sigma_t^T(f_t - \mu_f)(f_t - \mu_f)', \frac{1}{T})$
α_j	$N(\theta_0' z_j^\alpha, \Lambda_0)$	$N\left(\Lambda_0^{-1}\theta_0' z_j^\alpha + T v_j^{-1}\overline{(y_{jt} - \beta_j f_t)}, (\Lambda_0^{-1} + T v_j^{-1})^{-1}\right)$
β_j	$k: N(\theta_k' z_j^\beta, \Lambda_k)$	$N(B_j\left(\Lambda^{-1}\theta' z_j^\beta + v_j^{-1} X' y_j^*\right), B_j)$ $B_j = (\Lambda^{-1} + v_j^{-1} X'X)^{-1}, X = f_t, y_j^* = y_{jt} - \alpha$
τ_j	$N(\psi' z_j^\tau, \delta)$	Not of a standard form
θ_0	$U[-\infty, \infty]$	$N((Z^{\alpha\prime}Z^\alpha)^{-1}Z^{\alpha\prime}\alpha, \Lambda_0(Z^{\alpha\prime}Z^\alpha)^{-1})$
θ_k	$U[-\infty, \infty]$	$N((Z^{\beta\prime}Z^\beta)^{-1}Z^{\beta\prime}B_k, \Lambda_j(Z^{\beta\prime}Z^\beta)^{-1})$
ψ	$U[-\infty, \infty]$	$N((Z^{\tau\prime}Z^\tau)^{-1}Z^{\tau\prime}\alpha, \delta(Z^{\tau\prime}Z^\tau)^{-1})$
Λ_0	$IG(0,0.1)$	$IG(0.5 * J, 0.1 + 0.5 * \Sigma_j^p(\alpha_j - \theta_0' z_j^\alpha)^2)$
Λ_k	$IG(0,0.1)$	$IG(0.5 * J, 0.1 + 0.5 * \Sigma_j^p\left(\beta_{jk} - \theta_k' z_j^\beta\right)^2)$
δ	$IG(0,0.1)$	$IG(0.5 * J, 0.1 + 0.5 * \Sigma_j^p(\tau_j - \theta_k' z_j^\tau)^2)$

The posterior of τ_j doesn't not have a close form. The approximation of the posterior can be a normal distribution with variance $var = (0.5 * n + \delta^{-1})^\wedge - 1$, and mean:

$$var * (0.5 * T\, log\left(\frac{S_j}{T}\right) + \delta^{-1}\psi' z_j^\tau)$$

This approximation is biased. The more accurate way is to utilizing an imbedded Metropolis Hasting algorithm here, which uses the above distribution as the proposed distribution. The product of the likelihood can be computed analytically, so this will work. But in our simulation study below, we just approximate it with the normal density to accelerate the computation.

2.4 The Estimation Procedure

Based on the results above, the procedure implementing Gibbs sampling algorithm to estimate the factor model is described below:

1. Initialize the preliminary estimates of $(\alpha_j, \beta_j, \tau_j)$ via least-squares regression.
2. Based on the preliminary estimates of $(\alpha_j, \beta_j, \tau_j)$, obtain preliminary estimates of $\{\theta_k, \Lambda_k\}, \psi, \delta$ also via least-squares.
3. Repeat for G iterations:
(i) Generate samples of μ_f, Ω_f,;
(ii) For each stock, generate $\alpha_j, \beta_j, \tau_j$ from the respective posterior densities;
(iii) Given new samples of $\alpha_j, \beta_j, \tau_j$, generate samples of $\{\theta_k, \Lambda_k\}, \psi, \delta$;
4. The estimate of a particular parameter is obtained by taking the average from the last $G - B$ samples of it. Here, B is the number of initial samples discarded, to ensure the distribution has already converged for estimation.

4 A Numerical Test with Simulation

We now evaluate the performance of the above algorithm with a simulation study. We consider a one-factor model with an intercept. The factor is the market index returns which are simulated from a normal distribution $N(\mu_f, \Omega_f)$, where the mean is 1% and the standard deviation is 4%.

The factor loadings and idiosyncratic risk $\alpha_j, \beta_j, \tau_j$ are simulated based on hyperparameters:$\theta_0 = 0, \theta_1 = 1, \psi = -4.77, \Lambda_0 = 0.7\%^2, \Lambda_1 = 0.025\%^2, \delta = 1.22$. The only covariate (z) to predict $\alpha_j, \beta_j, \tau_j$ is the intercept which always takes 1. We simulated monthly returns of $J = 30$ stocks following the factor model for T=24 observations.

We replicated the simulation tests for $M = 100$ times. At each time, we calculate the mean absolute value (MAE) of the estimates $\alpha_j, \beta_j, \tau_j$ on the cross section. After all the replications, we compute the average of the MAE.

The following table shows the estimation results. We mainly evaluate the performance by comparing it with the least-squares regression.

Table 1 Estimation Results with Simulated Data

MAE	Least Squares	Gibbs Sampling
α_j	1.81%	1.71%
β_j	44.54%	11.72%
v_j	0.35%	0.81%

As is shown above, the Gibbs sampler proposed in this report provided a more accurate estimate for the true estimates of α_j, β_j. In particular, the error of the β_j estimation has been reduced a lot. However, as we are using a biased approximation for estimating v_j, the algorithm didn't outperform the least squares. This can be improved by imbedding the Metropolis-Hasting algorithm I mentioned above.

Reference

[1] Martin R. Young, Peter J. Lenk, (1998) Hierarchical Bayes Methods for Multifactor Model Estimation and Portfolio Selection. Management Science 44(11-part-2):S111-S124.