HMM+MCMC for Predicting Stock Return

1 Introduction

Predicting stock returns are of high interests for a lot of reasons. Conventionally, a linear model is used to predict future return based on previous return data (see e.g. sections 3 in [2]). However, we believe such linear models might not be good enough. One thing we can improve is to introduce hidden state into the model using Hidden Markov Model. Such hidden state specifies how good a comany is in one time period. Such information is usually very private (thus hidden). For example, such information could be whether the CEO is healthy or not. There could be enumorous number of such factors. For simplicity, we only consider two hidden states, good state and bad state. When a company is in good state, it will tend to produce higher expected return. We show that MCMC could be used to infer the parameters of this HMM model using the stock return data. We further use that to predict future return and did some benchmarks. During the project, we also tried to improve the performance of MCMC by applying batch update using Forward Backward recursions. It will reduce autocorrelation and accelerate MCMC's convergence. We also deal with the label switching problem, because good state and bad state are like different components in finite mixture models. The following of this writeup is organized as follows: in section 2 we describes our models and MCMC methods to infer the parameters[1]; insection 3 we specify how we get the data and implement the MCMC (such as how do we choose priors and how to switch labels); in section 4 we show the inference results of MCMC; in section 5 we show the prediction results and finally in section 6, we compare our work to others and discuss future research.

2 Methods

2.1 Model

Our model assumes that there are T time periods $t = 1, 2, \dots, T$. A company has a stock return y_t at t time period which is observable. At time period t, there's also a hidden variable $h_t \in \{1, 2\}$ representing whether that company is in a good state or a bad state. The model assumes that y_t is generated from h_t

[1] Most of these are inspired from [1]

by

$$y_t|h_t \sim N(\mu_{h_t}, \sigma_{h_t}) \tag{1}$$

That means, the return is drawn from two normal distributions, a bad distribution for bad state and a good distribution for good state. Presumably, $\mu_2 > \mu_1$, which means that good state tends to get higher expected return.

The h_t satisfies Markov property such that

$$Pr(h_t|h_1^{t-1}) = Pr(h_t|h_{t-1})$$

That means, h_t is conditional independent with $h_1, h_2, \ldots, h_{t-2}$ given h_{t-1}. Therefore the transition probability can be simply characterized as

$Pr(h_t\|h_{t-1})$	$h_t = 1$	$h_t = 2$
$h_{t-1} = 1$	q_1	$1 - q_1$
$h_{t-1} = 2$	q_2	$1 - q_2$

In summary, our model assumes that a company's stock return is a Hidden Markov Model (HMM) of T time periods, where observable data $\boldsymbol{y}$ (stock return) is drawn from normal distributios with mean and standard deviation $\boldsymbol{\mu}, \boldsymbol{\sigma}$ depending on the hidden state $\boldsymbol{h}$. The hidden state transition probability are $\boldsymbol{q}$. Overall, our model has data $\boldsymbol{y}$, hidden variables $\boldsymbol{h}$ and parameters $\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{q}$. In later sections, we also use $\boldsymbol{\theta}$ to denote all parameters $(\boldsymbol{\mu}, \boldsymbol{\sigma}, \boldsymbol{q})$.

2.2 Priors and Full Conditionals for MCMC

To infer the parameters and predict future stock return, we run MCMC with the following priors and full conditionals.

For $\boldsymbol{\mu}, \boldsymbol{\sigma}|y, h, q$, we use the Normal-Gamma as the prior (for $\mu, \phi = 1/\sigma^2$) like what we did in ordinary Gaussian (normal) models:

$$1/\sigma^2 \sim G(\nu_0/2, SS_0/2) \tag{2}$$

$$\mu|\phi \sim N(\mu_0, 1/(\kappa_0\phi)) \tag{3}$$

The full conditionals of $\boldsymbol{\mu}, \boldsymbol{\sigma}$ are also Normal-Gamma. The only difference is that we need to use $\boldsymbol{h}$ to split $\boldsymbol{y}$ into two sets $Y_{h=1}, Y_{h=2}$. Then we use $Y_{h=k}$ to update μ_k, σ_k:

$$\mu_k|\phi_k, Y_{h=k} \sim N\left(\mu_{n,k}, \frac{1}{\kappa_{n,k}\phi_k}\right) \tag{4}$$

$$\phi_k|Y_{h=k} \sim G(\nu_{n,k}/2, SS_{n,k}/2) \tag{5}$$

where

$$\kappa_{n,k} = \kappa_0 + n_k = \kappa_0 + |Y_{h=k}| \tag{6}$$

$$\mu_{n,k} = \frac{\phi_k n_k \overline{Y_{h=k}} + \phi_k \kappa_0 \mu_0}{\phi_k \kappa_{n,k}} \tag{7}$$

$$\nu_n = \nu_0 + n_k \tag{8}$$

$$SS_n = SS_0 + SS_{h=k} + \frac{n_k\kappa_0}{\kappa_{n,k}}(\overline{Y_{h=k}} - \mu_0)^2 \tag{9}$$

All those priors and full conditionals are similar to what's covered in lecture 6.

For $\boldsymbol{q}$, we use Beta as prior

$$q \sim Beta(a_0, b_0) \tag{10}$$

The full conditionals are also Beta distributions. Suppose n_{ij}^t is the number of transitions in the form $h_{t-1} = i, h_t = j$, we have

$$q_1|\boldsymbol{h} \sim Beta(a_0 + n_{11}^t, b_0 + n_{12}^t) \tag{11}$$

$$q_2|\boldsymbol{h} \sim Beta(a_0 + n_{21}^t, b_0 + n_{22}^t) \tag{12}$$

The full conditionals of $\boldsymbol{h}$ can be derived in two ways:

1. **Direct Gibbs (DG)**: In the first approach, we calculate full conditional for each h_t (thus assuming that all other $\boldsymbol{h}_{-t}$ are fixed):

$$Pr(h_t|\boldsymbol{h_{-t}, q, y}) \propto Pr(h_t|h_{t-1}, \boldsymbol{q})Pr(h_{t+1}|h_t, \boldsymbol{q})Pr(y_t|h_t, \mu_{h_t}, \sigma_{h_t}) \tag{13}$$

   where $Pr(h_i|h_{i-1}, \boldsymbol{q}) = q_{h_{i-1}h_i}$ if we set $q_{11} = q_1, q_{12} = 1 - q_1, q_{21} = q_2$ and $q_{22} = 1 - q_2$.

2. **Forward Backward (FB) Recursion**: In the second approach, we calculate full conditionals of the whole $\boldsymbol{h}$ using Forward Backward Recursion algorithms (actually, we will only use forward step here) that are well known in previous HMM research. The full conditionals are (recall that $\boldsymbol{\theta = (q, \mu, \sigma)}$ represents all parameters)

$$\begin{aligned} Pr(\boldsymbol{h}|\boldsymbol{y, \theta}) &= Pr(h_T|y_1^T, \boldsymbol{\theta}) \prod_{t=n-1}^{1} Pr(h_t|h_{t+1}^T, \boldsymbol{y, \theta}) \qquad (14) \\ &= Pr(h_T|y_1^T, \boldsymbol{\theta}) \prod_{t=n-1}^{1} Pr(h_t|h_{t+1}, y_1^{t+1}, \boldsymbol{\theta}) \end{aligned}$$

   where $Pr(h_t|h_{t+1}, y_1^{t+1}, \boldsymbol{\theta}) \propto Pr(h_t, h_{t+1}|y_1^{t+1}, \boldsymbol{\theta})$. Note that $Pr(h_T|y_t^T, \boldsymbol{\theta}), Pr(h_t, h_{t+1}|y_1^{t+1}, \boldsymbol{\theta})$ can be calculated efficiently by Forward Backward Recursion:

$$Pr(h_t, h_{t+1}|y_1^{t+1}, \boldsymbol{\theta}) \propto Pr(h_t|y_1^t, \boldsymbol{\theta})Pr(h_{t+1}|h_t, \boldsymbol{q})Pr(y_{t+1}|h_{t+1}, \boldsymbol{\theta}) \tag{15}$$

$$Pr(h_t|y_1^t, \boldsymbol{\theta}) = \sum_{h_{t-1}} Pr(h_t, h_{t-1}|y_1^t, \boldsymbol{\theta}) \tag{16}$$

3 Implementation

(a) Data

The monthly return of the components of SP100 stock from 2005 to 2011 are obtained from CRSP database. Six of them have missing data over the 72 months so we drop them and the sample size is then 94.

(b) Our Prior

For each stock, when h=1 and 2, return y_{ht} has the following prior distribution

$$y_{ht} \sim Normal(\mu_h, 1/\sqrt{\phi_h})$$
$$\phi_h \sim Gamma(\nu_0/2, SS_0/2)$$
$$\mu_h \sim Normal(\mu_0, 1/(\kappa_0\phi_i)$$

where

$$nu_0 = 10,\ SS_0 = 1,\ \mu_0 = 0,\ \kappa_0 = 10$$

The prior is informative because we have common sense that usually the absolute value of stock monthly return is within 10

For each stock, the transition matrix Q is

$$\begin{pmatrix} q_1 & 1-q_1 \\ q_2 & 1-q_2 \end{pmatrix}$$

where q_1 is the probability of state 1 to state 1 and q_2 is the probability of state 2 to state 1.

$$q_h \sim Beta(a_h, b_h)$$
$$a_1 = a_2 = b_1 = b_2 = 1$$

(c) Label Switching

To avoid label switching problem we use the constraint $mu_1 < mu_2$

A trick here is that when doing Gibbs Sampling, if $mu_1 > mu_2$, besides exchanging μs and ϕs, $[q_1,\ q_2]$ becomes $[1-q_2,\ 1-q_1]$ That is because when there is label switching what we get is indeed the probability of $[2 \to 1,\ 1 \to 1]([q_1,\ q_2])$. So we first get the probability of $[2 \to 2,\ 1 \to 2]([1-q_1,\ 1-q_2])$. Finally it's the probability of $[1 \to 1,\ 2 \to 1]([1-q_2,\ 1-q_1])$.

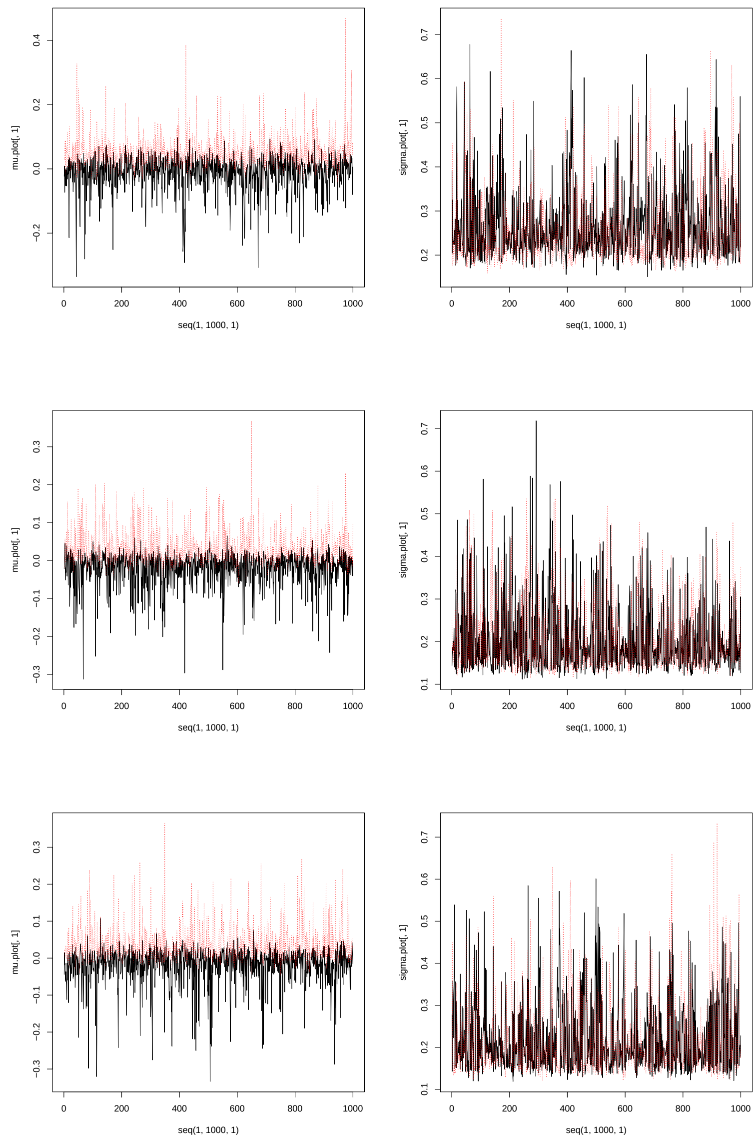
4 Statistical Inference

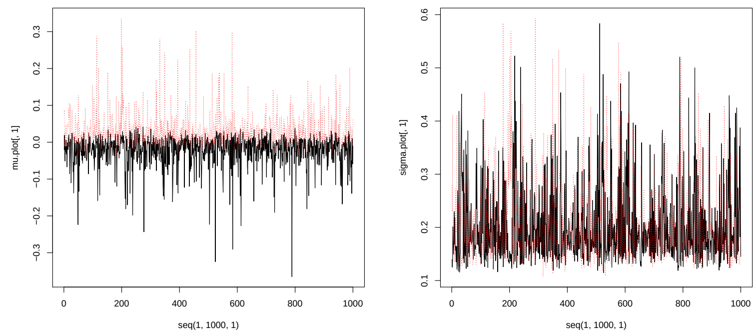
Number of iteration is 1100 and number of burn-in is 100.

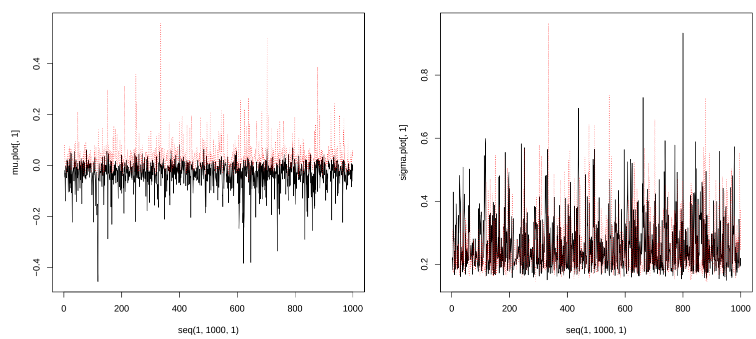
a Traceplot of μ and σ

The following plots show the traceplot of μ and σ for 5 stocks. The training data is from M_1 to M_60.

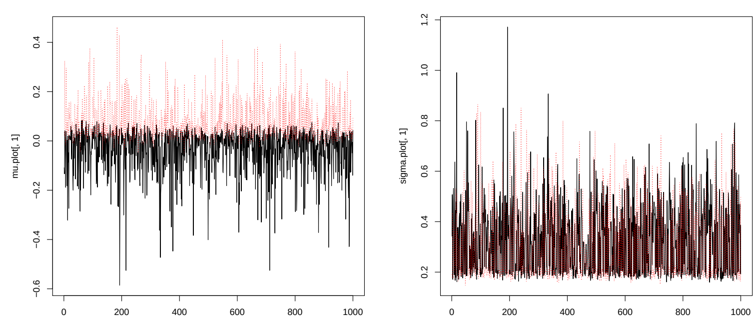
(1) DG method

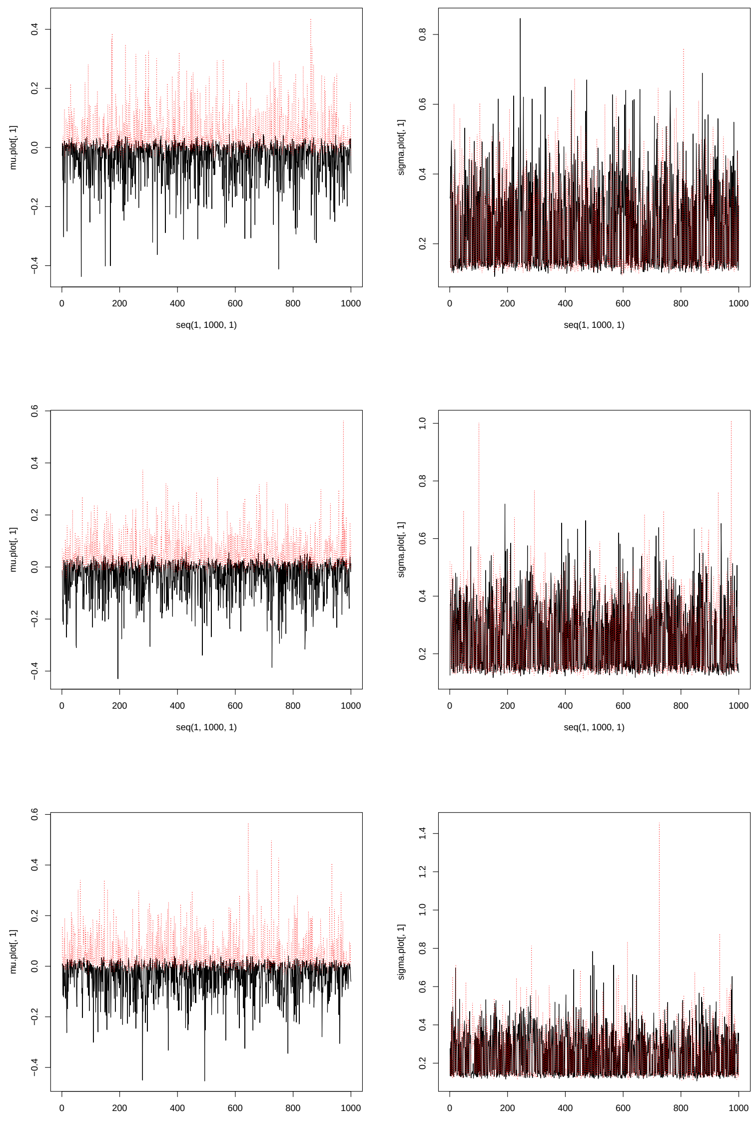

mu.plot[, 1]
sigma.plot[, 1]
seq(1, 1000, 1)
mu.plot[, 1]
sigma.plot[, 1]
seq(1, 1000, 1)
mu.plot[, 1]
sigma.plot[, 1]
seq(1, 1000, 1)

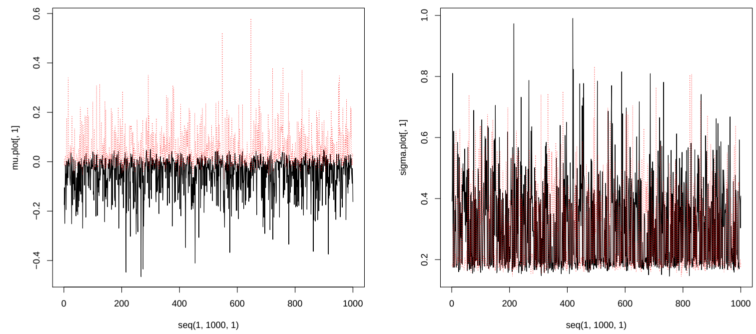

mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)


mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)

(2) FB method


mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)


mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)
mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)
mu.plot[, 1]
seq(1, 1000, 1)
sigma.plot[, 1]
seq(1, 1000, 1)

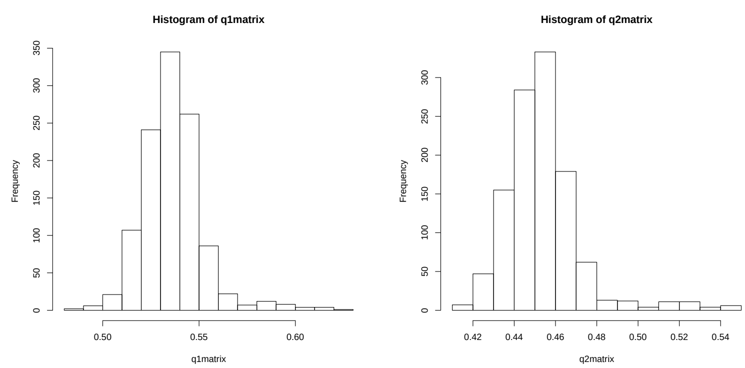


Both methods get converged result, though FB method generates μs with higher variance. There is no much difference between $sigma_1$ and $sigma_2$, consistent with our assumption.

b Transition Matrix We use 94 stocks and 12 different training periods(From M_1 to M_k where $k = 60, \ldots, 71$.We get the averaged Q matrix and histograms of q_1 and q_2 for both methods.

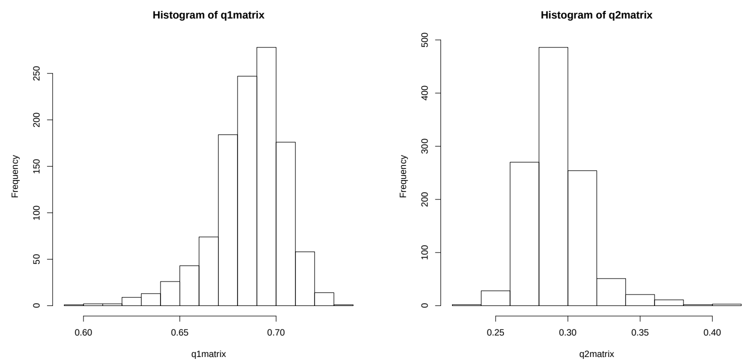
(1) DG Method

$$\begin{pmatrix} 0.54 & 0.46 \\ 0.45 & 0.55 \end{pmatrix}$$



(2) FB Method

$$\begin{pmatrix} 0.69 & 0.31 \\ 0.29 & 0.71 \end{pmatrix}$$



The averaged Q matrix indicates that h has positive autocorrelation, especially when FB method is used. That is to say, today being good means higher probability of tomorrow being good.

5 Prediction

We do not concern very much about the distribution of current return of stocks (Who cares?). We are mostly interested with predicting future returns (Almost everybody does!).

1. training data and prediction

   We use all past data to predict the returns of next month.

   To be specific, firstly, we use the return of all stocks in $M_1, \ldots, M_{60}$ to predict returns in M_61. Then the training data is $M_1, \ldots, M_{61}$ to predict returns in M_{62}. We do it for 12 times.

2. Prediction result

   When compared with the actual returns, the MSE of all stocks in 12 months using both methods are shown below:

Month	61	62	63	64	65	66	67	68	69	70	71	72
DG	0.06	0.05	0.04	0.05	0.05	0.04	0.06	0.07	0.08	0.09	0.06	0.06
FM	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01

   The prediction accuracy is much higher for FB method. The coverage percentage of 95% credible interval is 100% for both methods. A main reason is 95% credible interval is very wide compared with actual returns.

3. Portfolio Construction

   Now we want to study whether we can make money using this model. We pick out 20 stocks that have the highest return and put one in our portfolio and rebalance it monthly. Over 12 months, the yearly return using FB method is 4.29%. For DG method, the return is 3.57%. The return of SP94 (We only keep 94 stocks that have complete data) is 2.88%.

Can we say we beat the market? Yes and No. We need to buy and sell stocks each month. The return is of course higher, but if the transaction fee is considered, Uh...

6 Comparisons and Further research

Stock prices have correlations. When economy is good most stocks get decent returns. To construct a portfolio, both returns and risks should be taken into consideration. A Normal-Inverse-Wishart prior may be more reasonable.

In our research, we use hidden Markov model because our original idea was to predict the return of only a few stocks. Most investors do not have enough money to diversify their stock portfolio so correlation is not very important. When dealing with one stock. We believe that good stocks have are more likely to do well in the future so we introduce the states variables. This is what the Normal-Inverse-Wishart prior does not have.

For further research, if the two can be combined well, the prediction result may be more accurate and more reliable.

References

[1] Eric Jacquier and Nicholas Polson. Bayesian econometrics in finance. 2010.

[2] Scott S. L. Bayesian methods for hidden markov models: Recursive computing in the 21st century. *Journal of the American Statistical Association*, 97:337–351, 2002.