Sparse Linear Dynamical System with Its Application in Multivariate Clinical Time Series

Zitao Liu
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15213
ztliu@cs.pitt.edu

Milos Hauskrecht
Department of Computer Science
University of Pittsburgh
Pittsburgh, PA 15213
milos@cs.pitt.edu

Abstract

Linear Dynamical System (LDS) is an elegant mathematical framework for modeling and learning multivariate time series. However, in general, it is difficult to set the dimension of its hidden state space. A small number of hidden states may not be able to model the complexities of a time series, while a large number of hidden states can lead to overfitting. In this paper, we study methods that impose an ℓ_1 regularization on the transition matrix of an LDS model to alleviate the problem of choosing the optimal number of hidden states. We incorporate a generalized gradient descent method into the Maximum a Posteriori (MAP) framework and use Expectation Maximization (EM) to iteratively achieve sparsity on the transition matrix of an LDS model. We show that our Sparse Linear Dynamical System (SLDS) improves the predictive performance when compared to ordinary LDS on a multivariate clinical time series dataset.

1 Introduction

Developing accurate models of dynamical systems is critical for their successful applications in outcome prediction, decision support, and optimal control. A large spectrum of models have been developed and successfully applied for these purposes in the literature [3, 11, 20]. In this paper we focus on a popular model for time series analysis: the Linear Dynamical System (LDS) [15] and its application to clinical time series [18, 19]. We aim to develop a method to learn an LDS that performs better on future value predictions when learned from a small amount of complex multivariate time series dataset.

LDS is a widely used model for time series analysis of real-valued sequences. The model is Markovian and assumes the dynamic behaviour of the system is captured well using a small set of real-valued hidden-state variables and linear-state transitions corrupted by a Gaussian noise. The observations in LDS, similarly to hidden states, are real-valued. Briefly, the observations at time t are linear combinations of hidden state values for the same time. While in some LDS applications the model parameters are known a priori, in the majority of real-world applications the model parameters are unknown, and we need learn them from data that consists of observation sequences we assume were generated by the LDS model. While this can be done using standard LDS learning approaches, the problem of learning an LDS model gives rise to numerous important questions: Given the multivariate observation sequences, how many hidden states are needed to represent the system dynamics well? Moreover, since transition and observation matrices depend on the number of hidden states, how do we prevent the overfit of the model parameters when the number of examples is small?

In this work we address the above issues by presenting a method based on the sparse representation of LDS (SLDS) that is able to adjust (depending on the observation sequences in the data) the number of hidden states and at the same time prevent the overfit of the model. Our approach builds upon the probabilistic formulation of the LDS model, and casts the optimization of its parameters as

1

a maximum a posteriori (MAP) estimate, where the choice of the parameter priors biases the model towards sparse solutions.

Our SLDS approach is distinctly different from previous work [4, 6, 9]. [4] formulates the traditional Kalman filter as a one-step update optimization procedure and incorporates sparsity constraints to achieve sparsity in the hidden states. [9] trains an LDS for each training example and tries to find a sparse linear combination of coefficients in order to combine the ensemble of models. Neither [4] nor [9] directly achieve sparsity on the parameters of the LDS, and furthermore, the performance of their resulting models still depends on the optimal number of hidden states. [6] introduces a Bayesian nonparametric approach to the identification of observation-only linear systems, where no hidden states are involved. The underlining assumption is that the observations are obtained from linear combinations of previous observations and some system inputs, which may be too restrictive to model complex multivariate time series and makes the model more sensitive to noisy observations and outliers.

We test our sparse solution on the problem of modeling the dynamics of sequences of laboratory test results. We show that it improves the learning of the LDS model and leads to better accuracy in predicting future time-series values.

Our paper is organized as follows. In Section 2 we review the basics of the linear dynamical system. In Section 3 we describe SLDS – our method of sparsifying the LDS parameters. Inference and learning details of SLDS are explained in Section 3. Experimental results that compare SLDS method to ordinary LDS are presented in Section 4. In Section 5, we summarize the work and outline possible future extensions.

2 Linear Dynamical System (LDS)

The Linear Dynamical System (LDS) is a real-valued time series model that represents observation sequences indirectly with the help of hidden states. Let $\{z_t\}$, $\{y_t\}$ define sequences of hidden states and observations respectively. The LDS models the dynamics of these sequences in terms of the state transition probability $p(z_t|z_{t-1})$, and state-observation probability $p(y_t|z_t)$. These probabilities are modeled using the following equations:

$$z_t = Az_{t-1} + e_t; \quad y_t = Cz_t + v_t, \tag{1}$$

where y_t is a $d \times 1$ observation vector made at (current time) t, and z_t an $l \times 1$ hidden states vector. The transitions among the current and previous hidden states are linear and captured in terms of an $l \times l$ transition matrix A. The stochastic component of the transition, e_t, is modeled by a zero-mean Gaussian noise $e_t \sim \mathcal{N}(0, Q)$ with an $l \times 1$ zero mean and an $l \times l$ covariance matrix Q. The observations sequence is derived from the hidden states sequence. The dependencies in between the two are linear and modeled using a $d \times l$ emission matrix C. A zero mean Gaussian noise $v_t \sim \mathcal{N}(0, R)$ models the stochastic relation in between the states and observation. In addition to A, C, Q, R, the LDS is defined by the initial state distribution for z_1 with mean $\boldsymbol{\pi}_1$ and covariance matrix V_1, $z_1 \sim \mathcal{N}(\boldsymbol{\pi_1}, V_1)$. The complete set of the LDS parameters is $\Omega = \{A, C, Q, R, \boldsymbol{\pi_1}, V_1\}$. The parameters of the LDS model can be learned using either the Expectation-Maximization (EM) algorithm [8] or spectral learning algorithms [16, 21].

3 Sparse Linear Dynamical System (SLDS)

In this section, we propose a sparse representation of LDS that is able to adjust the number of hidden states and at the same time prevents the overfit of the model. More specifically, we impose ℓ_1 regularizers on every element of the transition matrix A_{ij}, which leads to zero entries in the transition matrix A. The zero entries in the transition matrix of LDS indeed reduce the actual number of parameters of LDS, sparsify the hidden states, and avoid the overfitting problem from the real data, even if we set the number of hidden states originally picked is too large.

To achieve sparsity on the transition matrix, we introduce a Laplacian prior to each element of A, A_{ij}, since Laplacian priors are equivalent to ℓ_1 regularizations [5, 10, 24]. In general, the Laplacian distribution has the following form: $p(x|\mu, \lambda) = \frac{1}{2\lambda} \exp(-\frac{|x-\mu|}{\lambda})$, $\lambda \geq 0$ where μ is the location parameter and λ is the scale parameter. Here, we assume every element A_{ij} is independent to each other and has the following Laplacian density ($\mu = 0$ and $\lambda = 1/\beta$), $p(A_{ij}|\beta) = \frac{\beta}{2} \exp(-\beta|A_{ij}|)$.

Hence, the prior probability for A is $p(A|\beta) = \prod_{i=1}^{l} \prod_{j=1}^{l} p(A_{ij}|\beta)$ and the log joint distribution for SLDS is:

$$\log p(\mathbf{z}, \mathbf{y}, A) = \log p(A) + \log p(z_1) + \sum_{t=2}^{T} \log p(z_t|z_{t-1}, A) + \sum_{t=1}^{T} \log p(y_t|z_t) \tag{2}$$

where T is the observation sequence length.

3.1 Learning

In this section we develop an EM algorithm for the MAP estimation of the SLDS. Let $\hat{z}_{t|T} \equiv \mathbb{E}[z_t|\mathbf{y}]$, $M_{t|T} \equiv \mathbb{E}[z_t z_t'|\mathbf{y}]$, $M_{t,t-1|T} \equiv \mathbb{E}[z_t z_{t-1}'|\mathbf{y}]$ and define the $\mathcal{Q}$ function as $\mathcal{Q} = \mathbb{E}_{\mathbf{z}}\Big[\log p(\mathbf{z}, \mathbf{y}, A)|\Omega\Big]$, where

$$\mathcal{Q} = \mathbb{E}_{\mathbf{z}}\Big[\log p(z_1)\Big] + \log p(A) + \mathbb{E}_{\mathbf{z}}\Big[\sum_{t=2}^{T} \log p(z_t|z_{t-1}, A)\Big] + \mathbb{E}_{\mathbf{z}}\Big[\sum_{t=1}^{T} \log p(y_t|z_t)\Big] \tag{3}$$

In the E-step (Inference), we follow the backward algorithm in [8] to compute $\mathbb{E}[z_t|\mathbf{y}]$, $\mathbb{E}[z_t z_t'|\mathbf{y}]$ and $\mathbb{E}[z_t z_{t-1}'|\mathbf{y}]$, which are sufficient statistics of the expected log likelihood. In the M-step (Learning), we try to find Ω that maximizes the likelihood lower bound $\mathcal{Q}$. In the following, we derive the M-step for gradient based optimization of the parameters Ω. We omit the explicit conditioning on Ω for notational brevity. Since the $\mathcal{Q}$ function is non-differentiable with respect to A, but differentiable with respect to all the other variables (C, R, Q, π_1, V_1), we separate the optimization into two parts.

Optimization of A. In each iteration in the M-step, we need to maximize $\mathbb{E}_{\mathbf{z}}\Big[\sum_{t=2}^{T} \log p(z_t|z_{t-1}, A)\Big] + \log p(A)$ with respect to A, which is equivalent to minimizing a function $f(A)$ that

$$f(A) = \underbrace{\frac{1}{2}\sum_{t=2}^{T} \mathbb{E}_{\mathbf{z}}\Big[(z_t - Az_{t-1})'Q^{-1}(z_t - Az_{t-1})\Big]}_{g(A)} + \underbrace{\beta||A||_1}_{h(A)} \tag{4}$$

where $||A||_1$ is the ℓ_1 norm on every element of matrix A, $||A||_1 = \sum_{i=1}^{l} \sum_{j=1}^{l} ||A_{ij}||_1$.

As we can see $f(A)$ is convex but non-differentiable and we can easily decompose $f(A)$ into two parts: $f(A) = g(A) + h(A)$, as shown in eq.(4). Since $g(A)$ is differentiable, we can adopt the generalized gradient descent algorithm to minimize $f(A)$. The update rule is: $A^{(k+1)} = \text{prox}_{\alpha_k}(A^{(k)} - \alpha_k \nabla g(A^{(k)}))$ where α_k is the step size at iteration k and the proximal function $\text{prox}_{\alpha_k}(A)$ is defined as the soft-thresholding function $S_{\beta\alpha_k}(A)$

$$[S_{\beta\alpha_k}(A)]_{ij} = \begin{cases} A_{ij} - \beta\alpha_k & \text{if } A_{ij} > \beta\alpha_k \\ 0 & \text{if } -\beta\alpha_k \leq A_{ij} \leq \beta\alpha_k \\ A_{ij} + \beta\alpha_k & \text{if } A_{ij} < -\beta\alpha_k \end{cases}$$

Theorem 1. *Generalized gradient descent with a fixed step size $\alpha \leq 1/(||Q^{-1}||_F \cdot ||\sum_{t=2}^{T} M_{t-1|T}||_F)$ for minimize eq.(4) has convergence rate $O(1/k)$, where k is the number of iterations.*

Proof. $g(A)$ is differentiable with respect to A, and its gradient is $\nabla g(A) = Q^{-1}(A\sum_{t=2}^{T} M_{t-1|T} - \sum_{t=2}^{T} M_{t,t-1|T})$. Using simple algebraic manipulation we arrive at $|| \nabla g(X) - \nabla g(Y)||_F = ||Q^{-1}(X - Y)\sum_{t=2}^{T} M_{t-1|T}||_F \leq ||Q^{-1}||_F \cdot ||\sum_{t=2}^{T} M_{t-1|T}||_F \cdot ||X - Y||_F$ where $|| \cdot ||_F$ is the Frobenius norm and the inequality holds because of the sub-multiplicative property of Frobenius norm. Since we know from eq.(4), $f(A) = g(A) + \beta||A||_1$, and $g(A)$ has Lipschitz continuous gradient with constant $||Q^{-1}||_F \cdot ||\sum_{t=2}^{T} M_{t-1|T}||_F$, according to [7, 22], $f(A^{(k)}) - f(A^*) \leq ||A^{(0)} - A^*||_F^2/2\alpha k$, where $A^{(0)}$ is the initial value and A^* is the optimal value for A; k is the number of iterations. $\square$

Theorem 1 gives us a simple way to set the step size during the generalized gradient updates and also guarantees the fast convergence rate.

Optimization of $\Omega \backslash A = \{C, R, Q, \pi_1, V_1\}$. Each of these parameters is estimated similarly to the approach in [8] by taking the corresponding derivative of the eq.(3), setting it to zero, and by solving it analytically. Update rules for $\Omega \backslash A = \{C, R, Q, \pi_1, V_1\}$ are shown in Algorithm 1.

The M-Step for optimizing Ω is summarized in Algorithm 1 in Appendix.

4 Experiments

We test our approach on time series data obtained from electronic health records of 4,486 post-surgical cardiac patients stored in PCP database [2, 13, 12, 23]. To test the performance of our prediction model, we randomly select 600 patients that have at least 10 *Complete Blood Count* (CBC) tests [1] ordered during their hospitalizations. The three tests used in this experiment are Mean Corpuscular Hemoglobin Concentration (MCHC), Mean Corpuscular Hemoglobin (MCH) and Mean Corpuscular Volume (MCV). These time series data are noisy, their signals fluctuate in time, and observations are obtained with varied time-interval period.

In order to get regularly sampled multivariate time series dataset, we apply an 8-hour discretization on our original multivariate time series dataset and use linear interpolation to fill the missing gaps from discretization. We compare our sparse LDS (SLDS) with ordinary LDS (OLDS) on the above multivariate time series dataset.

To evaluate the performance of our SLDS approach we split our time series for 600 patients into the training and testing sets, such that 50/100 times series form the training data, and 500 are used for testing.

Evaluation Metric. Our objective is to test the predictive performance of our approach by its ability to predict the future value of an observation for a patient for some future time *t* given a sequence of patient's past observations. We judge the quality of the prediction using the Average Mean Absolute Error (AMAE) on multiple test data predictions. More specifically, the AMAE is defined as follows:

$$AMAE = m^{-1}n^{-1} \sum_{i=1}^{m} \sum_{j=1}^{n} |y_{i,j} - \hat{y}_{i,j}| \tag{5}$$

where $y_{i,j}$ is the *j*th true observation from time series *i*, $\hat{y}_{i,j}$ is the corresponding predicted value of $y_{i,j}$. *m* is the number of time series and *n* is the length of each time series.

To conduct the evaluation, we use the test dataset to generate various prediction tasks as follows. For each patient *p* and complete time series *i* for that patient, we calculate the number of observations n_i^p in that time series *i*. We use n_i^p to generate different pairs of indices (ψ, ϕ) for that patient, such that $1 \leq \psi < \phi \leq n_i^p$, where ψ is the index of the last observation assumed to be seen, and ϕ is the index of the observation we would like to predict. By adding time stamp reading to each index, the two indices help us define all possible prediction tasks that we can formulate on that time series. For each time series *i* from patient *p*, we proceed by randomly picking 5 different pairs of indices (or 5 different prediction tasks) for the total of 2500 predictions tasks (500 x 5 = 2500). For each method, we repeat this random sampling predictions 10 times and we use the Average Mean Absolute Error (AMAE) on these tasks to judge the quality of test predictions. The prediction results are shown in Table 1, Figure 1 and Figure 2.

From Figure 1 and Figure 2, we can see that OLDS achieves its lowest prediction error *AMAE* when the number of hidden states is 5. By varying the number of states, the errors for OLDS first improve (till the optimal number of states is reached) and then increase when the number of hidden states exceeds the optimal point. This clearly shows the overfitting problem. The SLDS performs similarly; its performance first impoves and after that it deteriorates. However, its errors deteriorate at slower pace which shows it is more robust to the overfitting problem. Comparing the two methods, the SLDS always outperfroms the OLDS, indicating that the additional sparsity term included in optimization helps it to better fit the underlying structure of the transition matrix.

5 Conclusion

In this paper, we have presented a sparse linear dynamical system (SLDS) for multivariate time series predictions. Comparing with the traditional linear state-space systems, SLDS model tries

---

[1]CBC panel is used as a broad screening test to check for such disorders as anemia, infection, and other diseases.

Table 1: Average mean absolute error for OLDS and SLDS with different hidden states sizes.

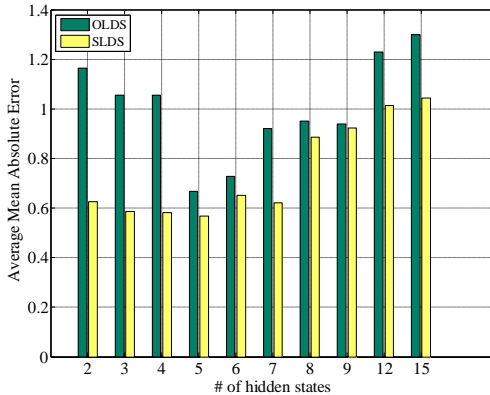| # of states	2	3	4	5	6	7	8	9	12	15
OLDS(50)	1.1653	1.0553	1.0561	0.6667	0.7268	0.9209	0.9502	0.9402	1.2291	1.2993
SLDS(50)	0.6256	0.5858	0.5821	0.5684	0.6506	0.6200	0.8854	0.9236	1.0134	1.0430
OLDS(100)	1.1527	1.0427	1.0039	0.6406	0.7153	0.8364	0.9210	0.9327	1.1427	1.2427
SLDS(100)	0.5709	0.5429	0.5889	0.6379	0.6897	0.6949	0.7643	0.7811	0.7874	0.8309

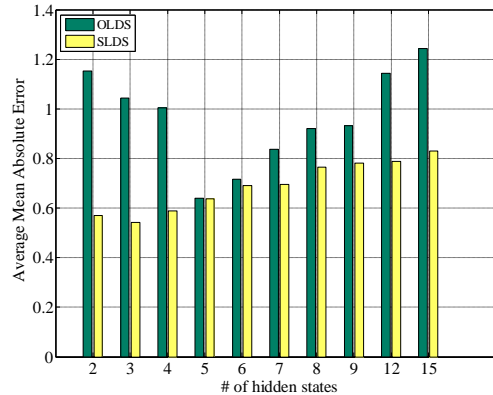Figure 1: AMAE on 50 training examples.

Figure 2: AMAE on 100 training examples.

to (1) prevent the overfitting problem and (2) represent additional structure in the transition matrix. Experimental results on real world clinical data from electronic health records systems demonstrated that this novel model achieves errors that is statistically significantly lower than errors of ordinary linear dynamical system. We would like to note that the results presented in this work are preliminary and include only three time series. Further investigation of more complex higher dimensional time-series data is needed and will be conducted in the future. In addition, we would like to study group lasso regularization techniques which we believe would be able to better control the dimensionality of the hidden state space. Finally, we plan to study extensions of our model to switching-state and controlled dynamical systems [14, 17].

References

[1] Iyad Batal, Dmitriy Fradkin, James Harrison, Fabian Moerchen, and Milos Hauskrecht. Mining recent temporal patterns for event detection in multivariate time series data. In *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 280–288. ACM, 2012.

[2] Iyad Batal, Hamed Valizadegan, Gregory F Cooper, and Milos Hauskrecht. A pattern mining approach for classifying multivariate temporal data. In *Bioinformatics and Biomedicine (BIBM), 2011 IEEE International Conference on*, pages 358–365. IEEE, 2011.

[3] Iyad Batal, Hamed Valizadegan, Gregory F Cooper, and Milos Hauskrecht. A temporal pattern mining approach for classifying electronic health record data. *Transactions on Intelligent Systems and Technology, Special Issue on Health Informatics*, 2013.

[4] Adam Charles, Muhammad Salman Asif, Justin Romberg, and Christopher Rozell. Sparsity penalties in dynamical system estimation. In *Information Sciences and Systems (CISS), 2011 45th Annual Conference on*, pages 1–6. IEEE, 2011.

[5] Scott Shaobing Chen, David L Donoho, and Michael A Saunders. Atomic decomposition by basis pursuit. *SIAM journal on scientific computing*, 20(1):33–61, 1998.

[6] Alessandro Chiuso and Gianluigi Pillonetto. Learning sparse dynamic linear systems using stable spline kernels and exponential hyperpriors. In *NIPS*, pages 397–405, 2010.

[7] Massimo Fornasier and Holger Rauhut. Iterative thresholding algorithms. *Applied and Computational Harmonic Analysis*, 25(2):187–208, 2008.

[8] Zoubin Ghahramani and Geoffrey E Hinton. Parameter estimation for linear dynamical systems. Technical report, Technical Report CRG-TR-96-2, University of Totronto, 1996.

[9] Bernard Ghanem and Narendra Ahuja. Sparse coding of linear dynamical systems with an application to dynamic texture recognition. In *Pattern Recognition (ICPR), 2010 20th International Conference on*, pages 987–990. IEEE, 2010.

[10] Yue Guan and Jennifer G Dy. Sparse probabilistic principal component analysis. In *Proceedings of AISTATS*, volume 5, pages 185–192, 2009.

[11] James Douglas Hamilton. *Time series analysis*, volume 2. Cambridge Univ Press, 1994.

[12] M. Hauskrecht, M. Valko, I. Batal, G. Clermont, S. Visweswaran, and G.F. Cooper. Conditional outlier detection for clinical alerting. In *AMIA Annual Symposium Proceedings*, volume 2010, page 286. American Medical Informatics Association, 2010.

[13] Milos Hauskrecht, Iyad Batal, Michal Valko, Shyam Visweswaran, Gregory F Cooper, and Gilles Clermont. Outlier detection for patient monitoring and alerting. *Journal of Biomedical Informatics*, 2012.

[14] Milos Hauskrecht and Hamish Fraser. Modeling treatment of ischemic heart disease with partially observable markov decision processes. *Proceedings of the AMIA Symposium*, page 538, 1998.

[15] Rudolph Emil Kalman et al. A new approach to linear filtering and prediction problems. *Journal of basic Engineering*, 82(1):35–45, 1960.

[16] T. Katayama. *Subspace methods for system identification*. Springer, 2005.

[17] Branislav Kveton and Milos Hauskrecht. Solving factored mdps with exponential-family transition models. In *ICAPS*, pages 114–120, 2006.

[18] Zitao Liu and Milos Hauskrecht. Clinical time series prediction with a hierarchical dynamical system. In *Artificial Intelligence in Medicine*, pages 227–237. Springer, 2013.

[19] Zitao Liu, Lei Wu, and Milos Hauskrecht. Modeling clinical time series using gaussian process sequences. In *SIAM International Conference on Data Mining (SDM)*, pages 623–631, 2013.

[20] Lennart Ljung and Torkel Glad. Modeling of dynamic systems. 1994.

[21] P.V. Overschee, BLD Moor, D.A. Hensher, J.M. Rose, W.H. Greene, K. Train, W. Greene, E. Krause, J. Gere, and R. Hibbeler. *Subspace Identification for the Linear Systems: Theory–Implementation*. Boston: Kluwer AcademicPublishers, 1996.

[22] NZ Shor. The rate of convergence of the generalized gradient descent method. *Cybernetics and Systems Analysis*, 4(3):79–80, 1968.

[23] Michal Valko and Milos Hauskrecht. Feature importance analysis for patient management decisions. In *MEDINFO*, 2010.

[24] Peter M Williams. Bayesian regularization and pruning using a laplace prior. *Neural computation*, 7(1):117–143, 1995.

Appendix

Algorithm 1 EM: M-step for the $(k+1)$th iteration. (We omit the explicit superscript $(k+1)$ for notational brevity.)

INPUT:
- Observation sequence y_ts, $t = 1, \ldots, T$.
- Sufficient statistics $\hat{z}_{t|T}$, $M_{t|T}$, $M_{t,t-1|T}$, $i = 1, \ldots, T$ from the $(k+1)$th iteration in E-Step.

PROCEDURE:

1: Update $\Omega \backslash A$: $C = (\sum_{t=1}^{T} y_t \hat{z}_{t|T}^{'})(\sum_{t=1}^{T} M_{t|T})^{-1}$, $\quad Q = \frac{1}{T-1}(\sum_{t=2}^{T} M_{t|T} - A \sum_{t=2}^{T} M_{t-1,t|T})$, $R = \frac{1}{T}\sum_{t=1}^{T}(y_t y_t^{'} - C\hat{z}_{t|T} y_t^{'})$, $\pi_1 = \hat{z}_{1|T}$, $V_1 = M_{1|T} - \hat{z}_{1|T}\hat{z}_{1|T}^{'}$.

2: Initialize A, $A = (\sum_{t=2}^{T} M_{t,t-1|T})(\sum_{t=2}^{T} M_{t-1|T})^{-1}$.

3: **repeat**

4:     Compute the fixed step size α, $\alpha = 1/(||Q^{-1}||_F \cdot ||\sum_{t=2}^{T} M_{t-1|T}||_F)$.

5:     Compute gradient of $g(A)$, $\bigtriangledown g(A) = Q^{-1}A\sum_{t=2}^{T} M_{t-1|T} - Q^{-1}\sum_{t=2}^{T} M_{t,t-1|T}$.

6:     Update A, $A = S_{\beta\alpha}(A - t \bigtriangledown g(A))$.

7: **until** Convergence

OUTPUT: $\Omega^{(k+1)} = \{A, C, Q, R, \pi_1, V_1\}$.