Probabilistic Multivariate Time-series Forecasting

Ali Izadi

Problem

- Multivariate time-series: $Y \in N \times T$
- Features: $X \in N \times T \times M$

در زمان t فیچر های X را در اختیار داریم و هدف پیش بینی Y است.

Y از یک فضای نهای تولید شده است. بنابراین یک state space در اختیار داریم.

$$
\begin{aligned}
L_1 &\sim N(\mu_1, \Sigma_1) \\
L_t &= M_t L_{t-1} + \epsilon_t \qquad \epsilon_t \sim N(0, \Sigma_t) \\
Z_t &= N_t L_t + \xi_t \qquad \xi_t \sim N(0, \Gamma_t)
\end{aligned}
$$

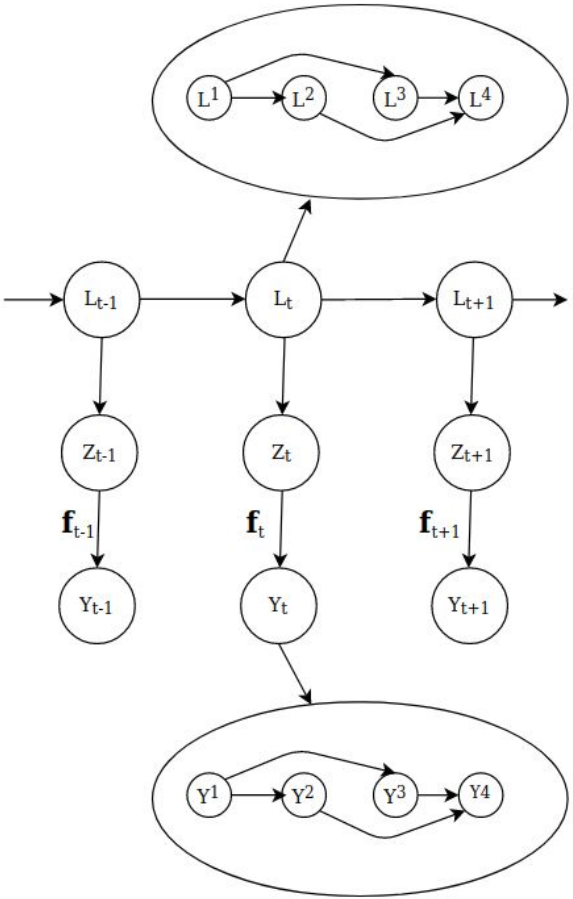
$$
\theta_t = \mu_1, \Sigma_1, \{M_t, \Sigma_t\}_{t=2}^{T}, \{N_t, \Gamma_t\}_{t=1}^{T}
$$

- دیدیم که اگر رابطه بین Z و Y با استفاده از normalizing flow یعنی تابع f غیرخطی شود هم چنان یک state space خطی داریم که تنها Yها توسط وارون تابع f به فرم ساده تری تبدیل میشوند.
- هم چنین پارامترهای تتا توسط یک RNN که از فیچرها به دست می آید در زمان t تخمین زده میشوند.

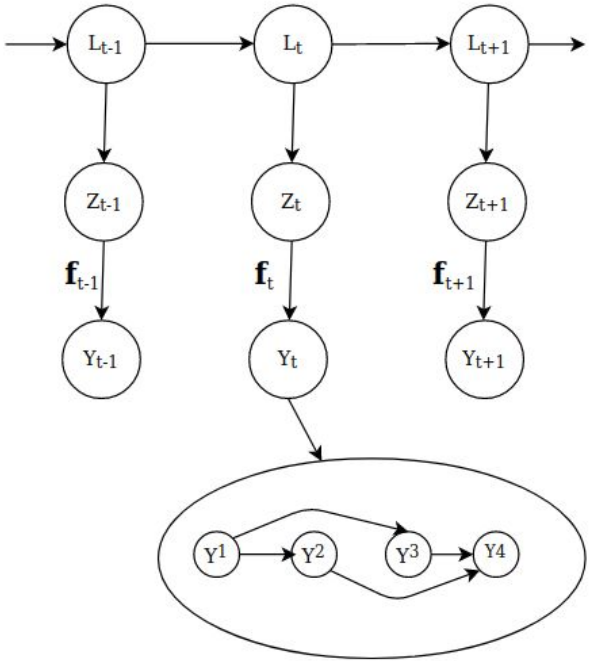
$$
\begin{aligned}
L_1 &\sim N(\mu_1, \Sigma_1) \\
L_t &= M_t L_{t-1} + \epsilon_t \qquad \epsilon_t \sim N(0, \Sigma_t) \\
Y_t &= f_t(N_t L_t + \xi_t) \qquad \xi_t \sim N(0, \Gamma_t)
\end{aligned}
$$

Normalizing Kalman Filter

$$
\begin{aligned}
\theta_t &= \mu_1, \Sigma_1, \{M_t, \Sigma_t\}_{t=2}^{T}, \{N_t, \Gamma_t\}_{t=1}^{T} \\
\theta_t &= \sigma(h_t = \Psi(X_t, h_{t-1}))
\end{aligned}
$$



- دنبال روشی هستیم تا بتواند bayesian network یا dag بین transition بین stateها و هم چنین بین stateها را مشخص کند.
- و همچنان بتوان به از خاصیت kalman filter یعنی فرم بسته filtering و likelihood استفاده کرد.
- روش های موجود یا از قبل bayesian network را در اختیار دارند یا از روش های variational استفاده اند.



- در ابتدا سراغ یادگیری dag بین observationها میرویم.

ایده:
استفاده از ایده مقاله graphical normalizing flow به عنوان یک flow در مقاله normalizing kalman filter تا kalman filter جدید قابلیت یادگیری ساختار در قسمت observation را داشته باشد.

likelihoods

- Normalizing Kalman filter

$$P(Y_{1:T}) = \prod_{t=1}^{T} P_{LGM}(Z_t \mid Z_{1:t-1}) \left|\det J_{g(y_t)}\right|$$

$$where \quad Z_t = g_t(Y_t) \;\; and \, g_t = f_t^{-1}$$

$$and \; P_{LGM} \; denotes \, predictive \, distribution \, of \; kalman \; filter$$

- Adding learnable dag A instead of autoregressive normalizing flow

$$P(Y_{1:T}) = \prod_{t=1}^{T} P_{LGM}(Z_t \mid Z_{1:t-1}) \prod_{i=1}^{N} \left| \frac{\mathrm{d}g^i\left(Y_t^i,\, h^i(Y_t \circ A_{i,:})\right)}{\mathrm{d}Y_t^i} \right|$$

$$where \circ \, denotes \, element - wise \, product \, between \, vector \, Y$$
$$and \, the \, i^{th} \, row \, of \, A$$

How to learn adjacency matrix A?

- Optimization:

$$L(\phi) = P(Y_{1:T}, \phi) = \prod_{t=1}^{T} P_{LGM}(Z_t \mid Z_{1:t-1}) \prod_{i=1}^{N} \left| \frac{\mathrm{d}g^i\left(Y_t^i, h^i(Y_t \circ A_{i,:})\right)}{\mathrm{d}Y_t^i} \right| + \lambda \|A\|$$

$$\max_{\phi}(L(\phi))$$

$$s.t.$$

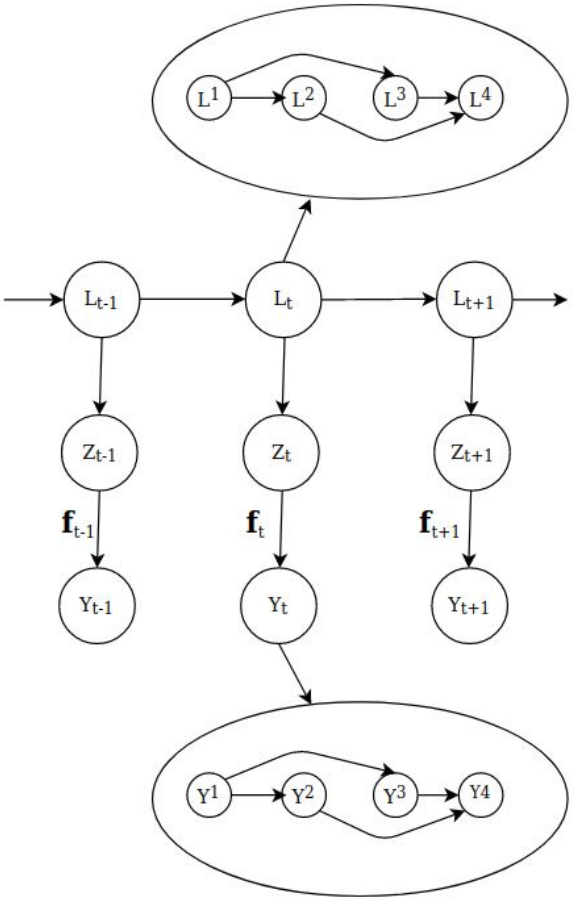
$$h(\phi) = Tr\, e^{A_\phi} - N = 0$$

- Augmented Lagrangian

$$\max_{\phi}(L(\phi, \rho_t, \sigma_t)) = L(\phi) - \rho_t h(\phi) - \frac{\sigma_t}{2} h(\phi)^2$$

$$\rho_{t+1} \leftarrow \rho_t + \sigma_t h(\phi_t^*)$$

$$\sigma_{t+1} \leftarrow \eta\sigma_t \; if \; h(\phi_t^*) > \gamma h\left(\phi_{t-1}^*\right) \; else \; \sigma_t$$



- حال به سراغ اضافه کردن dag بین stateها علاوه بر observationها میرویم.

- ایده:

- NO TEARS: $F(W) = \ell(W; \mathbf{X}) + \lambda \|W\|_1 = \frac{1}{2n} \|\mathbf{X} - \mathbf{X}W\|_F^2 + \lambda \|W\|_1.$

- Same adjacency matrix.

$$L_t = M_t L_{t-1} + \epsilon_t \qquad \epsilon_t \sim N(0, \Sigma_t)$$
$$\hat{L}_t = (M_t L_{t-1})^T W + \epsilon_t$$

$$F(\phi) = \sum_{t=1}^{T} \left(L_t - \hat{L}_t \right) + \lambda ||W||_1$$

$$F(\phi) = \sum_{t=1}^{T} \left(L_t - (M_t L_{t-1})^T W - \epsilon_t \right) + \lambda ||W||_1$$

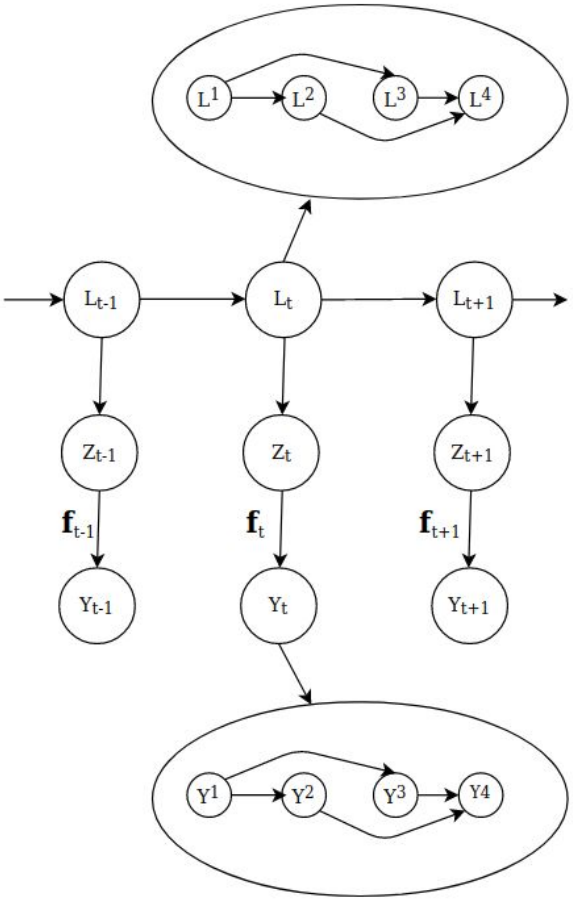
- optimization

$$F(\phi) = \sum_{t=1}^{T} \left(L_t - (M_t L_{t-1})^T W - \epsilon_t \right) + \lambda ||W||_1$$

$$\min_{\phi} (F(\phi))$$

$$s.t.$$

$$h(\phi) = Tr\, e^{W_\phi} - D = 0$$



- پس برای داشتن مدلی مطابق با رو به رو دو optimization خواهیم داشت که میتوان به صورت یک Optimization با دو constraint نوشت که برای این حالت نیز مشابه با زمانی که یک constraint داشتیم روش augmented lagrangian با دو constraint استفاده میشود.

https://en.wikipedia.org/wiki/Augmented_Lagrangian_method

- **Observations dag**

$$L(\phi) = P(Y_{1:T}, \phi) = \prod_{t=1}^{T} P_{LGM}(Z_t \mid Z_{1:t-1}) \prod_{i=1}^{N} \left| \frac{\mathrm{d}g^i\left(Y_t^i, h^i(Y_t \circ A_{i,:})\right)}{\mathrm{d}Y_t^i} \right| + \lambda \|A\|_1$$

$$\max_{\phi}(L(\phi))$$

$$s.t.$$

$$h(\phi) = Tr\, e^{A_\phi} - N = 0$$

- **States dag**

$$F(\phi) = \sum_{t=1}^{T} \left(L_t - (M_t L_{t-1})^T W - \epsilon_t \right) + \lambda ||W||_1$$

$$\min_{\phi}(F(\phi))$$

$$s.t.$$

$$h(\phi) = Tr\, e^{W_\phi} - D = 0$$

- Main optimization

$$LF = L(\phi) - F(\phi)$$
$$\max_{\phi}(LF(\phi))$$
$$s.t.$$
$$h_1(\phi) = Tr\, e^{A_\phi} - N = 0$$
$$h_2(\phi) = Tr\, e^{W_\phi} - D = 0$$

- Augmented Lagrangian

$$\max_{\phi}(LF(\phi, \rho_t, \sigma_t)) = LF(\phi) - \sum_{i=1}^{2}(\rho_t)_i h_i(\phi) - \frac{\sigma_t}{2}\sum_{i=1}^{2} h(\phi)^2$$

$$(\rho_{t+1})_i \leftarrow (\rho_t)_i + \sigma_t h_i(\phi_t^{\cdot})$$

$$\sigma_{t+1} \leftarrow \eta\sigma_t \; if \; h(\phi_t^{\cdot}) > \gamma h\left(\phi_{t-1}^{\cdot}\right) \; else \; \sigma_t$$

- Challenges

1. Graphical Normalizing flow expressiveness. How can we add multiple flows to get more powerful representation like simple normalizing flow.

2. It might help us to introduce new Dag learning approach using the power of normalizing flow.

3. Current dag learning methods have O(d^3) computational complexity instead of one approach which is linear some of its computation is approximation.

4. O(d^3) is challenging for multivariate time-series with more than 1000 dimensions which is not uncommon

Thank you.

aliizadi2030@gmail.com