THE UNIVERSITY *of* EDINBURGH

Graph discovery and Bayesian filtering in state-space models

Víctor Elvira
School of Mathematics
University of Edinburgh

Joint work with Émilie Chouzenoux (INRIA, France)

Interfaces between Statistics, ML, & AI (& Signal Processing!)
Centre for Statistics
April 27, 2021

Outline

State-space models (SSMs)

Bayesian filtering and the linear-Gaussian SSM

GraphEM: Graph discovery in linear-Gaussian SSMs

Experimental evaluation

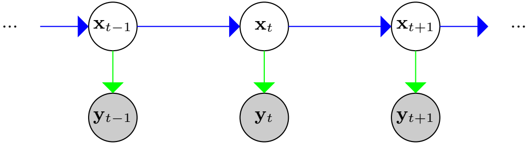
Conclusion

Motivation

- A large class of problems in statistics, machine learning, and signal processing requires **sequential processing of observed data**.
- Examples of **applications**:
  - Geophysical systems (atmosphere, oceans)
  - Robotics
  - Target tracking, positioning, navigation
  - Communications
  - Biomedical signal processing
  - Financial engineering
  - Ecology

Inference in State-Space Models (SSM)

- Let us consider:
  - a set of hidden states $\mathbf{x}_t \in \mathbb{R}^{d_x}$, $t = 1, ..., T$.
  - a set of observations $\mathbf{y}_t \in \mathbb{R}^{d_y}$, $t = 1, ..., T$.
- A SSM is an underlying hidden process of $\mathbf{x}_t$ that evolves and that, partially and noisily, expresses itself through $\mathbf{y}_t$.



- Two ways or describing the system:
  1. *Deterministic* notation:
     - Hidden state $\rightarrow$ $\mathbf{x}_t = g(\mathbf{x}_{t-1}, \mathbf{q}_t)$
     - Observations $\rightarrow$ $\mathbf{y}_t = h(\mathbf{x}_t, \mathbf{r}_t)$

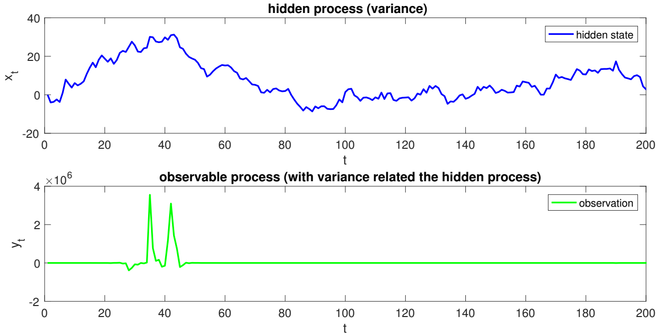
     where $\mathbf{q}_t$ and $\mathbf{r}_t$ are **random** noise vector (with known distributions of $\mathbf{q}_t$ and $\mathbf{r}_t$) and $g(\cdot)$ and $h(\cdot)$ are also known.
  2. *Probabilistic* notation:
     - Hidden state $\rightarrow$ $p(\mathbf{x}_t|\mathbf{x}_{t-1})$
     - Observations $\rightarrow$ $p(\mathbf{y}_t|\mathbf{x}_t)$

Example

- There are two interrelated random processes, one is observed and one is hidden.
  - e.g.,stochastic volatility model, very common in financial engineering

$$
\begin{aligned}
x_t &= 0.999 x_{t-1} + q_t \\
y_t &= e^{\frac{x_t}{2}} r_t,
\end{aligned}
$$

  - with $q_t \sim \mathcal{N}(0,1)$ and $r_t \sim \mathcal{N}(0,1)$
  - **Goal**: estimate the hidden x_t given the observed $y_{1:t}$

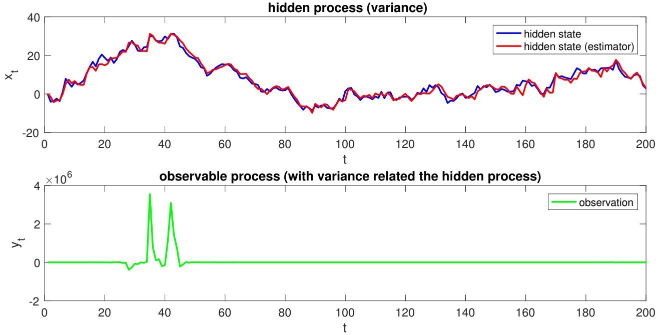


Example

- Consider the following stochastic volatility model, very common in financial engineering

$$\begin{aligned} x_t &= 0.999 x_{t-1} + q_t \\ y_t &= e^{\frac{x_t}{2}} r_t, \end{aligned}$$

  - with $q_t \sim \mathcal{N}(0,1)$ and $r_t \sim \mathcal{N}(0,1)$
  - **Goal**: estimate the hidden x_t given the observed $y_{1:t}$



Outline

The estimation problem

- We sequentially observe observations $\mathbf{y}_t$ related to the hidden state $\mathbf{x}_t$.
- At time t, we have accumulated t observations, $\mathbf{y}_{1:t} \equiv \{\mathbf{y}_1, ..., \mathbf{y}_t\}$.
- Bayesian inference to estimate the unknown states
  - measure of certainty by computing pdfs
- The basic problems:
  - **Filtering**: estimate current state $p(\mathbf{x}_t|\mathbf{y}_{1:t})$
  - **Smoothing**: refine estimate of past states $p(\mathbf{x}_{t-\tau}|\mathbf{y}_{1:t}), \qquad \tau \geq 1$
  - State prediction: predict the future state $p(\mathbf{x}_{t+\tau}|\mathbf{y}_{1:t}), \qquad \tau \geq 1$
  - Observation prediction: predict the future observation $p(\mathbf{y}_{t+\tau}|\mathbf{y}_{1:t}), \qquad \tau \geq 1$
- We will focus on **smoothing** and **filtering** problems
- We want to do it sequentially and efficiently.
  - At time t, we want to *process* only $\mathbf{y}_t$, but not reprocess all $\mathbf{y}_{1:t-1}$ (that were already processed!)

The linear-Gaussian Model

- The linear-Gaussian model is arguably the most relevant SSM:
- *Deterministic* notation:
  - Unobserved state → $\mathbf{x}_t = \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{q}_t$
  - Observations → $\mathbf{y}_t = \boldsymbol{H}_t \mathbf{x}_t + \mathbf{r}_t$

  where $\mathbf{q}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$ and $\mathbf{r}_t \sim \mathcal{N}(0, \mathbf{R}_t)$.
- *Probabilistic* notation:
  - Hidden state → $p(\mathbf{x}_t | \mathbf{x}_{t-1}) \equiv \mathcal{N}(\mathbf{x}_t; \mathbf{A}_t \mathbf{x}_{t-1}, \mathbf{Q}_t)$
  - Observations → $p(\mathbf{y}_t | \mathbf{x}_t) \equiv \mathcal{N}(\mathbf{y}_t; \boldsymbol{H}_t \mathbf{x}_t, \mathbf{R}_t)$
- **Kalman filter**: obtains the filtering pdfs $p(\mathbf{x}_t | \mathbf{y}_{1:t})$, at each t
  - Gaussian pdfs, with means and covariances matrices are calculated at each t
  - Efficient processing of $\mathbf{y}_t$, obtaining $p(\mathbf{x}_t | \mathbf{y}_{1:t})$ from $p(\mathbf{x}_{t-1} | \mathbf{y}_{1:t-1})$ (intermediate $p(\mathbf{x}_t | \mathbf{y}_{1:t-1})$ result)
- **Rauch-Tung-Striebel (RTS) smoother**: obtains the smoothing distribution $p(\mathbf{x}_{1:T} | \mathbf{y}_{1:T})$, i.e., posterior of the whole trajectory
  - requires a backwards reprocessing, refining the Kalman estimates

Kalman Filter: prediction step

1. **Prediction** step (marginalization of Gaussian):

$$p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1})d\mathbf{x}_{t-1}$$

- Suppose that filtered distribution at $t-1$ is Gaussian $p(\mathbf{x}_{t-1}|\mathbf{y}_{1:t-1}) \equiv \mathcal{N}(\mathbf{m}_{t-1}, \mathbf{P}_{t-1})$.
- Predictive distribution is also Gaussian $p(\mathbf{x}_t|\mathbf{y}_{1:t-1}) \equiv \mathcal{N}(\mathbf{m}_t^-, \mathbf{P}_t^-)$
  - Mean: $\mathbf{m}_t^- = \mathbf{A}_t\mathbf{m}_{t-1}$
  - Variance: $\mathbf{P}_t^- = \mathbf{A}_t\mathbf{P}_{t-1}\mathbf{A}_t^T + \mathbf{Q}_t$

- Interpretation:
  - The mean is projected through matrix $\mathbf{A}_t$
  - The **uncertainty** is propagated too through $\mathbf{A}_t$, plus the variance of the process noise

Kalman Filter: update step

2. **Update** step (product of Gaussians):

$$p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \frac{p(\mathbf{y}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{y}_{1:t-1})}{p(\mathbf{y}_t|\mathbf{y}_{1:t-1})}$$

- The filtered distribution at time t is also Gaussian $p(\mathbf{x}_t|\mathbf{y}_{1:t}) \equiv \mathcal{N}(\mathbf{m}_t, \mathbf{P}_t)$
  - Mean: $\mathbf{m}_t = \mathbf{m}_t^- + \boldsymbol{K}_t \left(\mathbf{y}_t - \boldsymbol{H}_t \mathbf{m}_t^-\right)$
  - Variance: $\mathbf{P}_t = (\boldsymbol{I} - \boldsymbol{K}_t \boldsymbol{H}_t)\, \mathbf{P}_t^-$

where $\boldsymbol{K}_t = \mathbf{P}_t^- \boldsymbol{H}_t^T \left(\boldsymbol{H}_t \mathbf{P}_t^- \boldsymbol{H}_t^T + \mathbf{R}_t\right)^{-1}$ is the optimal Kalman gain.

- Interpretation:
  - The mean is corrected w.r.t. the predictive in the direction of the residual/error.
  - The variance is propagated by $\boldsymbol{H}_t$ and divided by the covariance of the residual/error.

Kalman summary and RTS smoother

Kalman filter

- Initialize: $\mathbf{m}_0$, $\mathbf{P}_0$
- For $t = 1, \ldots, T$

  Predict stage:

$$\mathbf{x}_t^- = \mathbf{A}_t \mathbf{m}_{t-1}$$
$$\mathbf{P}_t^- = \mathbf{A}_t \mathbf{P}_{t-1} \mathbf{A}_t^\top + \mathbf{Q}_t$$

  Update stage:

$$\mathbf{z}_t = \mathbf{y}_t - \mathbf{H}_t \mathbf{x}_t^-$$
$$\mathbf{S}_t = \mathbf{H} \mathbf{P}_t^- \mathbf{H}_t^\top + \mathbf{R}_t$$
$$\mathbf{K}_t = \mathbf{P}_t^- \mathbf{H}_t^\top \mathbf{S}_t^{-1}$$
$$\mathbf{m}_t = \mathbf{x}_t^- + \mathbf{K}_t \mathbf{z}_t$$
$$\mathbf{P}_t = \mathbf{P}_t^- - \mathbf{K}_t \mathbf{S}_t \mathbf{K}_t^\top$$

RTS smoother

- For $t = T, \ldots, 1$

  Smoothing stage:

$$\mathbf{x}_{t+1}^- = \mathbf{A}_t \mathbf{m}_t$$
$$\mathbf{P}_{t+1}^- = \mathbf{A}_t \mathbf{P}_t \mathbf{A}_t^\top + \mathbf{Q}_t$$
$$\mathbf{G}_t = \mathbf{P}_t \mathbf{A}_t^\top (\mathbf{P}_{t+1}^-)^{-1}$$
$$\mathbf{m}_t^s = \mathbf{m}_t + \mathbf{G}_t(\mathbf{m}_{t+1}^s - \mathbf{x}_{t+1}^-)$$
$$\mathbf{P}_t^s = \mathbf{P}_t + \mathbf{G}_t(\mathbf{P}_{t+1}^s - \mathbf{P}_{t+1}^-)\mathbf{G}_t^\top$$

✓ Filtering distribution: $p(\mathbf{x}_t|\mathbf{y}_{1:t}) = \mathcal{N}(\mathbf{x}_t; \mathbf{m}_t, \mathbf{P}_t)$

✓ Smoothing distribution: $p(\mathbf{x}_t|\mathbf{y}_{1:T}) = \mathcal{N}(\mathbf{x}_t; \mathbf{m}_t^s, \mathbf{P}_t^s)$

✗ How to proceed if some model parameters are unknown ?

Outline

- Recall the linear-Gaussian system:
  - Unobserved state → $\mathbf{x}_t = \mathbf{A}_t \mathbf{x}_{t-1} + \mathbf{q}_t$
  - Observations → $\mathbf{y}_t = \boldsymbol{H}_t \mathbf{x}_t + \mathbf{r}_t$

  where $\mathbf{q}_t \sim \mathcal{N}(0, \mathbf{Q}_t)$ and $\mathbf{r}_t \sim \mathcal{N}(0, \mathbf{R}_t)$.
- In practice, most of these parameters are **unknown**: $\mathbf{A}_t$, $\mathbf{H}_t$, $\mathbf{Q}_t$, $\mathbf{R}_t$.
  - A common assumption is that they are static, i.e., $\mathbf{A}$, $\mathbf{H}$, $\mathbf{Q}$, $\mathbf{R}$.
- The most challenging parameter to estimate (but also interesting) is $\mathbf{A}$:
  - **Graph discovery perspective**: $\mathbf{x}_t \in \mathbb{R}^{N_x}$ contains N_x unidimensional time-series, each of them acquired in a node of a graph (with N_x total nodes)
  - The elements $a_{i,j}$ of $\mathbf{A}$ represents, the linear effect of node j at time $t-1$ in the update of the signal of node i at time t:

$$x_{t,i} = \sum_{j=1}^{N_x} a_{i,j} x_{t-1,j} + q_{t,i} \tag{1}$$

- GraphEM: An expectation-maximization (EM) method within Kalman filters for the estimation of $\mathbf{A}$ (along with the hidden states).[1]

[1] E. Chouzenoux and V. Elvira. "GraphEM: EM algorithm for blind Kalman filtering under graphical sparsity constraints". In: *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE. 2020, pp. 5840–5844.

GraphEM in a nutshell

- **Goal:** Find $\mathbf{A}^*$ that maximizes $p(\mathbf{A}|\mathbf{y}_{1:T}) \propto p(\mathbf{A})p(\mathbf{y}_{1:T}|\mathbf{A})$, i.e., the MAP estimate of $\mathbf{A}$
  - Equivalent to minimizing $\varphi_T(\mathbf{A}) = -\log p(\mathbf{A}) - \log p(\mathbf{y}_{1:T}|\mathbf{A})$.
  - **Challenge**: evaluating $p(\mathbf{y}_{1:T}|\mathbf{A})$ (or $\varphi_T(\mathbf{A})$) requires to run Kalman filter:

$$\varphi_T(\mathbf{A}) = -\log p(\mathbf{A}) + \sum_{t=1}^{T} \frac{1}{2} \log |2\pi \mathbf{S}_t(\mathbf{A})| + \frac{1}{2} \mathbf{z}_t(\mathbf{A})^\top \mathbf{S}_t(\mathbf{A})^{-1} \mathbf{z}_t(\mathbf{A}) \quad (2)$$

    - Non tractable minimization.

- EM strategy: Minimize a sequence of tractable approximations of φ_T satisfying a majorizing property.

- Lasso regularization (prior): In order to limit the degrees of freedom in the parametric model, we choose the prior to promote a sparse matrix $\mathbf{A}$.

$$(\forall \mathbf{A} \in \mathbb{R}^{N_x \times N_x}) \quad -\log p(\mathbf{A}) \equiv \varphi_0(\mathbf{A}) = \gamma \|\mathbf{A}\|_1, \qquad \gamma > 0.$$

Expression of EM steps

• Majorizing approximation (E-step): Run the Kalman filter/RTS smoother by setting the state matrix to $\mathbf{A}'$ and define

$$\mathbf{\Sigma} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{P}_t^s + \mathbf{m}_t^s(\mathbf{m}_t^s)^\top,$$

$$\mathbf{\Phi} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{P}_{t-1}^s + \mathbf{m}_{t-1}^s(\mathbf{m}_{t-1}^s)^\top$$

$$\mathbf{C} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{P}_t^s\mathbf{G}_{t-1}^\top + \mathbf{m}_t^s(\mathbf{m}_{t-1}^s)^\top.$$

Then, as a consequence of, we can build

$$\mathcal{Q}(\mathbf{A};\mathbf{A}') = \frac{T}{2}\mathrm{tr}\left(\mathbf{Q}^{-1}(\mathbf{\Sigma} - \mathbf{C}\mathbf{A}^\top - \mathbf{A}\mathbf{C}^\top + \mathbf{A}\mathbf{\Phi}\mathbf{A}^\top)\right) + \varphi_0(\mathbf{A}) + \mathcal{C},$$

such that, for every $\mathbf{A} \in \mathbb{R}^{N_x \times N_x}$:

$$\mathcal{Q}(\mathbf{A};\mathbf{A}') \geq \varphi_T(\mathbf{A}), \qquad \text{and} \qquad \mathcal{Q}(\mathbf{A}';\mathbf{A}') = \varphi_T(\mathbf{A}').$$

• Upper bound optimization (M-step): The M-step consists in searching for a minimizer of $\mathcal{Q}(\mathbf{A};\mathbf{A}')$ with respect to $\mathbf{A}$ ($\mathbf{A}'$ being fixed).

Computation of the M-step

- Minimization problem:

$$\text{argmin}_{\mathbf{A}} \underbrace{\mathcal{Q}(\mathbf{A};\mathbf{A}')}_{f(\mathbf{A})} = \text{argmin}_{\mathbf{A}} \underbrace{\frac{T}{2}\text{tr}\left(\mathbf{Q}^{-1}(\boldsymbol{\Sigma} - \mathbf{C}\mathbf{A}^\top - \mathbf{A}\mathbf{C}^\top + \mathbf{A}\boldsymbol{\Phi}\mathbf{A}^\top)\right)}_{f_1(\mathbf{A})=\text{upper bound of } -log\,(p(\mathbf{y}_{1:T}|\mathbf{A}))} + \underbrace{\gamma\|\mathbf{A}\|_1}_{\substack{f_2(\mathbf{A})=-\log p(\mathbf{A})\\ \text{(prior)}}}$$

  - Convex non-smooth minimization problem

- Proximal splitting approach: The proximity operator of $f : \mathbb{R}^{N_x\times N_x} \to \mathbb{R}$ is defined[2]

$$\text{prox}_f(\widetilde{\mathbf{A}}) = \text{argmin}_{\mathbf{A}} \left(f(\mathbf{A}) + \frac{1}{2}\|\mathbf{A} - \widetilde{\mathbf{A}}\|_F^2 \right).$$

Douglas-Rachford algorithm

- Set $\mathbf{Z}_0 \in \mathbb{R}^{N_x\times N_x}$ and $\theta \in (0,2)$.
- For $n = 1, 2, \ldots$

$$\begin{aligned} \mathbf{A}_n &= \text{prox}_{\theta f_2}(\mathbf{Z}_n) \\ \mathbf{V}_n &= \text{prox}_{\theta f_1}(2\mathbf{A}_n - \mathbf{Z}_n) \\ \mathbf{Z}_{n+1} &= \mathbf{Z}_n + \theta(\mathbf{V}_n - \mathbf{A}_n) \end{aligned}$$

✓ $\{\mathbf{A}_n\}_{n\in\mathbb{N}}$ guaranteed to converge to a minimizer of $\mathcal{Q}(\mathbf{A};\mathbf{A}') = f_1 + f_2$

✓ Both involved proximity operators have closed form solution.

[2] P.L. Combettes and JC. Pesquet. "Proximal Splitting Methods in Signal Processing." In: *Fixed-Point Algorithms for Inverse Problems in Science and Engineering* 49 (2011), pp. 185–212.

GraphEM algorithm

GraphEM algorithm

- Initialization of $\mathbf{A}^{(0)}$.
- For $i = 1, 2, \ldots$

E-step Run the Kalman filter and RTS smoother by setting $\mathbf{A}' := \mathbf{A}^{(i-1)}$ and construct $\mathcal{Q}(\mathbf{A}; \mathbf{A}^{(i-1)})$.

M-step Update $\mathbf{A}^{(i)} = \text{argmin}_{\mathbf{A}} \left(\mathcal{Q}(\mathbf{A}; \mathbf{A}^{(i-1)})\right)$ using Douglas-Rachford algorithm.

✓ Flexible approach, valid as long as the proximity operator of f_2 is available.

✓ sound convergence properties of the EM algorithm

- monotonical decrease and convergence of $\{\varphi_T(\mathbf{A}^{(i)})\}_{i\in\mathbb{N}}$ can be shown.

Outline

Data description and numerical settings

- Four synthetic datasets with $\mathbf{H} = \mathbf{Id}$ and block-diagonal matrix $\mathbf{A}$, composed with b blocks of size $(b_j)_{1 \leq j \leq b}$, so that $N_y = N_x = \sum_{j=1}^{b} b_j$. We set $T = 10^3$, $\mathbf{Q} = \sigma_{\mathbf{Q}}^2 \mathbf{Id}$, $\mathbf{R} = \sigma_{\mathbf{R}}^2 \mathbf{Id}$, $\mathbf{P}_0 = \sigma_{\mathbf{P}}^2 \mathbf{Id}$.

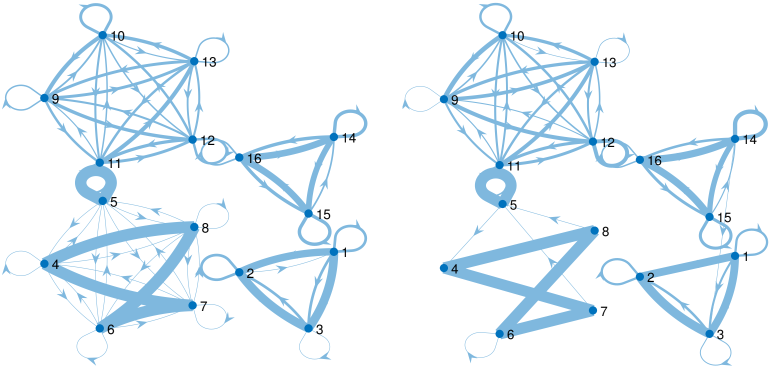
Dataset	N_x	$(b_j)_{1\leq j\leq b}$	$(\sigma_{\mathbf{Q}}, \sigma_{\mathbf{R}}, \sigma_{\mathbf{P}})$
A	9	$(3,3,3)$	$(10^{-1}, 10^{-1}, 10^{-4})$
B	9	$(3,3,3)$	$(1, 1, 10^{-4})$
C	16	$(3,5,5,3)$	$(10^{-1}, 10^{-1}, 10^{-4})$
D	16	$(3,5,5,3)$	$(1, 1, 10^{-4})$

- GraphEM is compared with:
  - Maximum likelihood EM (MLEM)[3]
  - Granger-causality approaches: pairwise Granger Causality (PGC) and conditional Granger Causality (CGC)[4]

[3] S. Sarkka. *Bayesian Filtering and Smoothing*. Ed. by Cambridge University Press. 3rd ed. 2013.

[4] D. Luengo et al. "Hierarchical algorithms for causality retrieval in atrial fibrillation intracavitary electrograms". In: *IEEE journal of biomedical and health informatics* 23.1 (2018), pp. 143–155.

Experimental results



True graph (left) and GraphEM estimate (right) for dataset C.

Experimental results

	method	RMSE	accur.	prec.	recall	spec.	F1
A	GraphEM	0.081	0.9104	0.9880	0.7407	0.9952	**0.8463**
	MLEM	0.149	0.3333	0.3333	1	0	0.5
	PGC	-	0.8765	0.9474	0.6667	0.9815	0.7826
	CGC	-	0.8765	1	0.6293	1	0.7727
B	GraphEM	0.082	0.9113	0.9914	0.7407	0.9967	**0.8477**
	MLEM	0.148	0.3333	0.3333	1	0	0.5
	PGC	-	0.8889	1	0.6667	1	0.8
	CGC	-	0.8889	1	0.6667	1	0.8
C	GraphEM	0.120	0.9231	0.9401	0.77	0.9785	**0.8427**
	MLEM	0.238	0.2656	0.2656	1	0	0.4198
	PGC	-	0.9023	0.9778	0.6471	0.9949	0.7788
	CGC	-	0.8555	0.9697	0.4706	0.9949	0.6337
D	GraphEM	0.121	0.9247	0.9601	0.7547	0.9862	**0.8421**
	MLEM	0.239	0.2656	0.2656	1	0	0.4198
	PGC	-	0.8906	0.9	0.6618	0.9734	0.7627
	CGC	-	0.8477	0.9394	0.4559	0.9894	0.6139

Outline

Conclusions

GraphEM algorithm:

- ✓ Interpretation of hidden states as a (causal) directed graph
- ✓ Lasso penalization to promote sparsity
  - common in complex systems
  - reduces the implicit dimension
- ✓ EM-based method with proximal splitting M-step
  - sound convergence guarantees
- ✓ Good numerical performance compared to several techniques

Thank you for your attention!

E. Chouzenoux and V. Elvira, "GraphEM: EM algorithm for blind Kalman filtering under graphical sparsity constraints," IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pp. 5840-5844, 2020.