Applied Artificial Intelligence Institute (A2I2)
Deakin University
March 2, 2025

Problem Notes

Long Van Tran

s224930257@deakin.edu.au

1 Problem Statement

In a conventional setting, we have a time-series dataset $\{\mathbf{X}^{(m)}\}_{m=1}^{M}$,

$$\mathbf{X}^{(m)} = \left[\mathbf{X}_1^{(m)}, \mathbf{X}_2^{(m)}, \ldots, \mathbf{X}_T^{(m)}\right],$$

where $\mathbf{X}_i^{(m)}$ can be vectors ($\mathbf{X}_i^{(m)} \in \mathbf{R}^d$) or matrices ($\mathbf{X}_i^{(m)} \in \mathbf{R}^{t\times d}$), depending on the unit of time being considered here is a single time step or a period of t steps. Here we consider the latter matrix case. Each of these trajectories describe patient records across T time periods (e.g., patient measurements throughout T hours).

However, due to various practical reasons (such as data anonymization, incomplete data logs, etc.), the temporal ordering (i.e. the time dimension) is lost. Furthermore, the trajectory data is also missing, with the assumption that for each of T time period, there is always at least one trajectory with data present. Thus, instead of a temporally-labeled complete dataset $\{\mathbf{X}^{(m)}\}_{m=1}^{M}$, we have a dataset of M temporally-unordered partially-completed trajectories.

We assume that the data is governed by a time-homogeneous linear additive noise stochastic differential equation (SDE), which has the form:

$$\mathrm{d}\mathbf{X}_t = \mathbf{A}\mathbf{X}_t\mathrm{d}t \ + \ \mathbf{G}\mathrm{d}\mathbf{W}_t,$$

with unknown drift-diffusion parameters $\mathbf{A} \in \mathbf{R}^{d\times d}$ and $\mathbf{G} \in \mathbf{R}^{d\times m}$. $\mathbf{W}$ is an m-dimensional Brownian motion. Our aim is to:

1. Reconstruct missing segments into full (or partially completed) trajectories $\{\widetilde{\mathbf{X}}^{(m)}\}$. This aim is intermediate.

2. Estimate $\mathbf{A}$ and $\mathbf{G}$ by maximum likelihood, while enforcing a Directed Acyclic Graph (DAG) constraint on the global ordering of the reconstructed segments (thus avoiding "cyclic" order). This is our final and most important aim.

2 Proposed Method

We proposed a solution that works as follows:

2.1. Sorting Data by Variance

- Empirically evaluate the variance of each observed segment of all our trajectories.
- Sort segments in ascending order of variance, with the assumption that we are dealing with a diverging SDE and that a diverging SDE usually has increasing variance over (hidden) time. This can be done in a similar manner for converging SDEs.

2.2. Iterative Two-Step Scheme

First, we randomly initialize the parameters $\mathbf{A}^{(0)}$ and $\mathbf{G}^{(0)}$. Then, we follow a two-step iterative scheme as follows:

(a) **Update Previously-Sorted Segments**
Keeping $\mathbf{A}^{(n)}$ and $\mathbf{G}^{(n)}$ at iteration n fixed, rearrange the previously sorted segments for each trajectory to maximize the log-likelihood minus a DAG penalty Ω_{DAG}:

$$\{\widetilde{\mathbf{X}}^{(m)}\} = \arg \max_{\{\widetilde{\mathbf{X}}^{(m)}\}} \sum_{m=1}^{M} \ln p\big(\widetilde{\mathbf{X}}^{(m)} \mid \mathbf{A}^{(n)}, \mathbf{G}^{(n)}\big) \;-\; \Omega_{\mathrm{DAG}}\big(\{\widetilde{\mathbf{X}}^{(m)}\}\big).$$

A well-known continuous DAG-penalty comes from the NOTEARS approach by Zheng et al., 2018 [2]:

$$h(\mathbf{A}) = \operatorname{trace}(e^{\mathbf{A}\circ\mathbf{A}}) - d,$$

$$\Omega_{\mathrm{DAG}}\big(\{\widetilde{\mathbf{X}}^{(m)}\}\big) = \alpha(h(\mathbf{A})),$$

where d is the number of variables (i.e. the size of the square matrix $\mathbf{A}$) and α is a regularization hyper-parameter. It is proved in [2] that

$h(\mathbf{A}) = 0$ if and only if $\mathbf{A}$, our adjacency matrix, corresponds to a DAG (no directed cycles). If cycles exist, $h(\mathbf{A}) > 0$. Hence, adding this term as a penalty encourages $\mathbf{A}$ to remain acyclic.

(b) **Update SDE Parameters**
With the newly completed trajectories fixed, re-estimate the SDE parameters:

$$(\mathbf{A}^{(n+1)}, \mathbf{G}^{(n+1)}) = \arg\max_{\mathbf{A},\mathbf{G}} \sum_{m=1}^{M} \ln p\big(\widetilde{\mathbf{X}}^{(m)} \mid \mathbf{A}^{(n)}, \mathbf{G}^{(n)}\big).$$

This step can be done using the parameter estimation framework named APPEX introduced by Guan et al., 2024 [1].

3 Notes

Some notes to keep in mind:

- Proving convergence of the iterative scheme.
- Leveraging the identifiability conditions presented in [1].

References

[1] Vincent Guan, Joseph Janssen, Hossein Rahmani, Andrew Warren, Stephen Zhang, Elina Robeva, Geoffrey Schiebinger, Identifying drift, diffusion, and causal structure from temporal snapshots, 2024, available from: `https://arxiv.org/abs/2410.22729`.

[2] Xun Zheng, Bryon Aragam, Pradeep Ravikumar, Eric P. Xing, Dags with no tears: Continuous optimization for structure learning, 2018, available from: `https://arxiv.org/abs/1803.01422`.