

Supplementary Material to:

Spectral learning of linear dynamics from generalised-linear
observations with application to neural population data

Lars Buesing*, Jakob H. Macke*,[†], Maneesh Sahani
Gatsby Computational Neuroscience Unit
University College London, London, UK
{lars, jakob, maneesh}@gatsby.ucl.ac.uk

1 Derivation of Moment Conversion for Poisson LDS Model

Here we compute the mean and covariance matrix of \mathbf{z}^\pm from those of \mathbf{y}^\pm under the PLDS model. We introduce the following notation:

$$\begin{aligned}\boldsymbol{\mu} &:= \mathbb{E}[\mathbf{z}^\pm] \\ \Sigma &:= \text{Cov}[\mathbf{z}^\pm] \\ \mathbf{m} &:= \mathbb{E}[\mathbf{y}^\pm] \\ S &:= \text{Cov}[\mathbf{y}^\pm].\end{aligned}$$

Slightly overloading the notation, we denote the elements of \mathbf{z}^\pm and \mathbf{y}^\pm by z_i and y_i respectively for $i = 1, \dots, 2kq$. The observation model can then be written as:

$$y_i \mid z_i \sim \text{Poisson}[\exp(z_i)].$$

For $\mathbf{a} \in \{0, 1\}^{2kq}$ we derive the following expected value, using the fact that all y_1, \dots, y_{2kq} are independent given \mathbf{z}^\pm :

$$\begin{aligned}\mathbb{E}\left[\prod_{i=1}^{2kq} y_i^{a_i}\right] &= \mathbb{E}\left[\mathbb{E}\left[\prod_{i=1}^{2kq} y_i^{a_i} \mid \mathbf{z}^\pm\right]\right] \\ &= \mathbb{E}\left[\prod_{i=1}^{2kq} \mathbb{E}[y_i^{a_i} \mid z_i]\right] \\ &= \mathbb{E}\left[\prod_{i=1}^{2kq} \mathbb{E}[y_i \mid z_i]^{a_i}\right] \\ &= \mathbb{E}\left[\prod_{i=1}^{2kq} \exp(z_i)^{a_i}\right] \\ &= \mathbb{E}[\exp(\mathbf{a}^\top \mathbf{z}^\pm)] \\ &= \exp\left(\frac{1}{2} \mathbf{a}^\top \Sigma \mathbf{a} + \mathbf{a}^\top \boldsymbol{\mu}\right),\end{aligned}$$

where $\mathbb{E}[\cdot \mid \mathbf{z}^\pm]$ denotes the expectation conditioned on \mathbf{z}^\pm . The last step of this derivation involves the Gaussian expected value over the exponential $\exp(\mathbf{a}^\top \mathbf{z}^\pm)$, which is given in eqn. (8) in the Appendix. From this result we can immediately read off the first and the “off-diagonal” second moments of \mathbf{y}^\pm :

$$\mathbb{E}[y_i] = m_i = \exp\left(\frac{1}{2} \Sigma_{ii} + \mu_i\right) \tag{1}$$

$$\begin{aligned}\mathbb{E}[y_i y_j] = S_{ij} + m_i m_j &= \exp\left(\frac{1}{2} \Sigma_{ii} + \mu_i + \frac{1}{2} \Sigma_{jj} + \mu_j + \Sigma_{ij}\right) \\ &= m_i m_j \exp(\Sigma_{ij}).\end{aligned} \tag{2}$$

Solving the last eqn. (2) for Σ_{ij} yields eqn. (8) of the main paper [1].

Now we calculate the “diagonal” second moments:

$$\begin{aligned}
\mathbb{E}[y_i^2] = S_{ii} + m_i^2 &= \mathbb{E}[\mathbb{E}[y_i^2|z_i]] = \mathbb{E}[\text{Var}(y_i|z_i) + \mathbb{E}[y_i|z_i]^2] \\
&= \mathbb{E}[\exp(z_i) + \exp(2z_i)] \\
&= m_i + \exp(2\Sigma_{ii} + 2\mu_i) \\
&= m_i + \exp(\Sigma_{ii}) m_i^2,
\end{aligned} \tag{3}$$

where we used eqn. (8) from the Appendix to compute $\mathbb{E}[\exp(2z_i)]$ and we used eqn. (1) in the last step. Solving the eqn. (3) for Σ_{ii} yields eqn. (7) of the main paper [1]. Plugging the latter into eqn. (1) yields eqn. (6) of the main paper [1].

As an aside, we can write down the covariance Σ in the following more compact way:

$$\Sigma = \log(S + \mathbf{m}\mathbf{m}^\top - \text{diag}(\mathbf{m})) - \log(\mathbf{m}\mathbf{m}^\top), \tag{4}$$

where $\text{diag}(\mathbf{m})$ is the diagonal matrix with m_1, \dots, m_{2kq} on the diagonal.

2 Details to PLDSID with External Inputs

2.1 Moment Conversion

We assume that the external input \mathbf{u} is a Gaussian process. Then $\mathbf{u}^\pm, \mathbf{z}^\pm$ are jointly normal and we redefine $\boldsymbol{\mu}$ and Σ to be their mean and joint covariance matrix respectively:

$$\begin{aligned}
\mathbf{x}_t &:= \begin{pmatrix} \mathbf{u}_t^\pm \\ \mathbf{z}_t^\pm \end{pmatrix} = \begin{pmatrix} \mathbf{u}_t \\ \vdots \\ \mathbf{u}_{t+k-1} \\ \mathbf{u}_{t-1} \\ \vdots \\ \mathbf{u}_{t-k} \\ \mathbf{z}_t \\ \vdots \\ \mathbf{z}_{t+k-1} \\ \mathbf{z}_{t-1} \\ \vdots \\ \mathbf{z}_{t-k} \end{pmatrix} \\
\boldsymbol{\mu} &:= \mathbb{E}[\mathbf{x}] = \mathbb{E}\left[\begin{pmatrix} \mathbf{u}^\pm \\ \mathbf{z}^\pm \end{pmatrix}\right] = \begin{pmatrix} \boldsymbol{\mu}^u \\ \boldsymbol{\mu}^z \end{pmatrix} \\
\Sigma &:= \text{Cov}[\mathbf{x}] = \text{Cov}\left[\begin{pmatrix} \mathbf{u}^\pm \\ \mathbf{z}^\pm \end{pmatrix}\right] = \begin{pmatrix} \Sigma^{uu} & \Sigma^{uz} \\ \Sigma^{zu} & \Sigma^{zz} \end{pmatrix}.
\end{aligned} \tag{5}$$

We again omit the time dependence of expected values because of stationarity. The lower right block Σ^{zz} can be estimated from the data as described in eqn. (4) of the previous section, and Σ^{uu} can be directly estimated as the empirical covariance of the input \mathbf{u}^\pm . We can derive the remaining block $\Sigma^{uz} = \Sigma^{zu\top}$ in the following way. We compute the expectation $\mathbb{E}[y_i u_j]$ using this result given in eqn. (7) in the Appendix:

$$\begin{aligned}
\mathbb{E}[y_i u_j] = \mathbb{E}[\exp(z_i) u_j] &= \exp\left(\frac{1}{2}\Sigma_{ii}^{zz} + \mu_i^z\right) (\mu_j^u + \Sigma_{ji}^{uz}) \\
&= m_i (\mu_j^u + \Sigma_{ji}^{uz}).
\end{aligned}$$

Solving this equation for Σ_{ji}^{uz} results in:

$$\Sigma_{ji}^{uz} = \frac{\text{Cov}[y_i, u_j]}{m_i}. \tag{6}$$

Using eqn. (6) and (4) as well as eqn. (6) of the main paper, we can estimate $\boldsymbol{\mu}$ and Σ from the data.

2.2 Subspace Identification

For subspace identification of driven PLDS models we use the “robust N4SID” algorithm summarized in Fig. 6.1 on p. 169 of [2]. To this end, we define the vector \mathbf{w}_t in the following way:

$$\mathbf{w}_t := \begin{pmatrix} \mathbf{u}_{t-k} \\ \vdots \\ \mathbf{u}_{t+k-1} \\ \mathbf{z}_{t-k} \\ \vdots \\ \mathbf{z}_{t+k-1} \end{pmatrix}.$$

Here we assumed that \mathbf{z}_t and \mathbf{u}_t have mean zero, which can be achieved simply by subtracting their means $\boldsymbol{\mu}^z$ and $\boldsymbol{\mu}^u$. Note that \mathbf{w}_t is given by permuting the rows of \mathbf{x}_t as can be seen from the definition given in eqn. (5). In the original N4SID algorithm it is assumed that we are given observations $\mathbf{w}_1, \dots, \mathbf{w}_j$ which are concatenated into a matrix W (where j is the length of the observed time series):

$$W := j^{-1/2} \begin{pmatrix} \mathbf{w}_1 & \dots & \mathbf{w}_j \end{pmatrix}.$$

The N4SID algorithm proceeds by decomposing W with an RQ-decomposition $W = RQ^\top$, where R is a $(2k(r+q)) \times (2k(r+q))$ lower triangular matrix and Q is a $j \times (2k(r+q))$ matrix with orthonormal columns. In [2] it is shown, that the parameter identification step in the N4SID algorithm only depends on R , and not on Q .

In our setting however, we do not have access to the time series $\mathbf{w}_1, \dots, \mathbf{w}_j$ directly. The moment conversion yields only joint moments $\boldsymbol{\mu}$ and Σ of $\mathbf{x} = ((\mathbf{u}^\pm)^\top, (\mathbf{z}^\pm)^\top)^\top$. However, by permuting rows and columns of Σ we can get an estimate of the covariance $\tilde{\Sigma}$ of \mathbf{w}_t , as \mathbf{w}_t is just a permutation of \mathbf{x} . Furthermore, it is straightforward to show that for infinitely many training examples $j \rightarrow \infty$, R equals the transposed Cholesky factor of the covariance matrix $\tilde{\Sigma}$ of \mathbf{w}_t :

$$\begin{aligned} \tilde{\Sigma} &:= \text{Cov} \left[\begin{pmatrix} \mathbf{u}_{t-k} \\ \vdots \\ \mathbf{u}_{t+k-1} \\ \mathbf{z}_{t-k} \\ \vdots \\ \mathbf{z}_{t+k-1} \end{pmatrix} \right] = \text{Cov}[\mathbf{w}_t] \\ \tilde{\Sigma} &= RR^\top. \end{aligned}$$

Hence, the following procedure yields consistent estimates for parameters of PLDS models driven by Gaussian inputs \mathbf{u}^\pm :

1. Moment conversion: Compute the joint moments $\boldsymbol{\mu}$ and Σ of $\mathbf{u}^\pm, \mathbf{z}^\pm$ from the joint moments of $\mathbf{u}^\pm, \mathbf{y}^\pm$
2. Rearrange entries of Σ to obtain an estimate of the covariance matrix $\tilde{\Sigma}$ of \mathbf{w}
3. Cholesky decomposition of $\tilde{\Sigma} = RR^\top$, where R is lower-triangular
4. Apply standard N4SID algorithm to R

References

- [1] Lars Buesing, Jakob Macke, and Maneesh Sahani. Spectral learning of linear dynamics from generalised-linear observations with application to neural population data. In *Advances in Neural Information Processing Systems* 25.
- [2] P. Van Overschee and B. De Moor. *Subspace Identification for the Linear Systems: Theory – Implementation – Applications*. Boston: Kluwer Academic Publishers, 1996.

Appendix

Let \mathbf{x} be n -dimensional random variable, which is normally distributed with mean $\boldsymbol{\mu}$ and covariance matrix Σ :

$$\mathbf{x} \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma).$$

We state the following expected value, where $f(\mathbf{x})$ is a scalar function of \mathbf{x} and $\mathbf{a} \in \mathbb{R}^n$:

$$\begin{aligned}\mathbb{E}[\exp(\mathbf{a}^\top \mathbf{x})f(\mathbf{x})] &= \int \exp(\mathbf{a}^\top \mathbf{x})f(\mathbf{x})d\mathcal{N}(\mathbf{x}|\boldsymbol{\mu}, \Sigma) \\ &= \exp\left(\frac{1}{2}\mathbf{a}^\top \Sigma \mathbf{a} + \mathbf{a}^\top \boldsymbol{\mu}\right) \int f(\mathbf{x}')d\mathcal{N}(\mathbf{x}'|\boldsymbol{\mu} + \Sigma \mathbf{a}, \Sigma).\end{aligned}\tag{7}$$

In particular, we obtain for $f(\mathbf{x}) = 1$:

$$\mathbb{E}[\exp(\mathbf{a}^\top \mathbf{x})] = \exp\left(\frac{1}{2}\mathbf{a}^\top \Sigma \mathbf{a} + \mathbf{a}^\top \boldsymbol{\mu}\right).\tag{8}$$