Statistical inefficiency of Markov model count matrices

Frank Noé

22. September 2015

We consider the problem of inferring a Markov model with transition matrix $\hat{P} \in \mathbb{R}^{n\times n}$ from a discrete-state time series $X = (x_1, ..., x_T)$, where T is the number of time steps. The posterior probability of a Markov chain with transition matrix P given X is:

$$\mathbb{P}(P|X) \propto \mathbb{P}(P)\,\mathbb{P}(X|P)$$

where $\mathbb{P}(P)$ is the transition matrix prior and $\mathbb{P}(X|P)$ is the likelihood, i.e. the probability that the sequence X was generated by transition matrix P. If X has been generated by a Markov chain, then the count matrix $C(\tau) = (c_{ij}(\tau)) \in \mathbb{R}^{n\times n}$ at lag time τ with elements

$$c_{ij}(\tau) = \sum_{t=1}^{T} 1_i(x_t)\, 1_j(x_{t+\tau}) \tag{1}$$

where $1_i(x)$ is the characteristic function with:

$$1_i(x) = \begin{cases} 1 & x = i \\ 0 & \text{else.} \end{cases}$$

is a sufficient statistics for inferring the transition probabilities at lag time τ. For now, let's assume $\tau = 1$ because X is a Markovian sequence. The likelihood of transition matrix $P = P(\tau)$ is then given by

$$\mathbb{P}(X|P) \propto \prod_{i,j=1}^{n} p_{ij}^{c_{ij}}. \tag{2}$$

It is well-known that the maximum likelihood estimator $\hat{P} = \arg\max_P \mathbb{P}(X|P)$, i.e. the matrix P that maximizes the likelihood under the constraint that P is a transition matrix, is given by:

$$\hat{p}_{ij} = \frac{c_{ij}}{c_i},$$

where we have defined the row sum $c_i = \sum_j c_{ij}$. The mean and the variance of p_{ij} under the likelihood (2) are

$$\begin{aligned} \bar{p}_{ij} &= \frac{c_{ij}+1}{c_i+n} \\ \mathrm{Var}[p_{ij}] &= \frac{\bar{p}_{ij}(1-\bar{p}_{ij})}{(c_i+n+1)}, \end{aligned} \tag{3}$$

which asymptotically (for $T \to \infty$) decreases proportionally to n^{-1} as usual for Monte Carlo methods. If a uniform prior is used these expressions are also identical to the posterior mean and posterior variance. The latter is important to assess the uncertainty of estimation. More generally, one can sample (2) by generating Dirichlet-distributed random variables for the independent row distributions $p(c_{i1}, ..., c_{in}|p_{i1}, ..., p_{in}) = \prod_j p_{ij}^{c_{ij}}$ and then compute sample distributions of arbitrary functions of P, such as the distribution of eigenvalues or other quantities of interest.

The same approach of maximum likelihood estimation and Bayesian estimation can be followed when additional constraints on P are made, such as detailed balance [4, 9, 8, 1, 17] and detailed balance with respect to a fixed stationary distribution [9, 16, 17]. These aspects have been extensively discussed in the field of Markov models of molecular kinetics from molecular dynamics (MD) simulations [14, 15, 3, 11, 2]. In this field, the choice of suitable prior has also been discussed.

In this note, our main question is: how should we infer Markov chains if X is not Markovian? This is the realm of Markov models, i.e. now the transition matrix $P(\tau)$ has a finite systematic error that will not vanish as a function of trajectory length. For MD simulation, and for many other simulations or experiments of physical systems, it is now well understood that this systematic error can be controlled by choosing a sufficiently large τ [13, 5, 10]. It is much less understood, however, how the count matrix $C(\tau)$ should be obtained at a given τ such that the statistical error, e.g. (3), can be correctly estimated. This is precisely the question addressed in this paper.

Transition counting mode

Consider these three options for the transition counting mode:

1. Sample count: perform one count per τ, thus generating $\lfloor T/\tau \rfloor$ transition counts:

$$c_{ij}^{\mathrm{sample}}(\tau) = \sum_{k=0}^{\lfloor T/\tau \rfloor - 1} 1_i(x_{k\tau}) 1_j(x_{k\tau+\tau})$$

2. Sliding window count: use all possible $T - \tau$ transition counts as given in (1).

3. Scaled sliding window count: Use the sliding window counts $c_{ij}(\tau)$ but scale them with some factor $I_{ij} \leq 1$ called statistical inefficiency. A choice that has been previously suggested is $I_{ij} = \tau^{-1}$.

All three choices converge to the same maximum likelihood estimator in the limit of good statistics ($T \to \infty$). However, the width of the likelihood and thus the size of standard deviations and confidence intervals, are vastly different under these different choices.

For Markov models from MD simulations, the standard approach was so far to use sample counts (choice 1) [6, 9, 11]. This choice is based on the argument that when the sequence X appears approximately Markovian at lag time τ, then transition counts are approximately independent at this lag time. However this approach is statistically inefficient: If S is also Markovian for shorter lag times than τ, then we are using less information than we could. Even if S only becomes Markovian at lag times of τ or longer, transitions such as $1 \to \tau + 1$ and $\tau/2 \to \tau + \tau/2$ are usually only partially correlated, such that discarding the second transition is also not fully exploiting the data. In practical MD simulations, the lag times required such that a Markov model is a good approximation need to be quite long (often in the the range of nanoseconds), such that subsampling the data at τ will create severe problems with data and connectivity loss.

Using the sliding window approach (choice 2) exploits all data, but harvests too many counts as transitions $t \to t + \tau$ and $t + 1 \to t + \tau + 1$ are generally not independent for non-Markovian data. Therefore the error estimates will be too small with this approach.

The scaled sliding window count (choice 3) offers a solution to the problems faced by choices 1 and 2. We always obtain the count matrix c_{ij} in a sliding window mode. We formally correct the overcounting problem by introducing a statistical inefficiency $I_{ij}(\tau)$ for every count at a given lag time, such that $c_{ij}^{\mathrm{eff}}(\tau) = I_{ij}(\tau)\, c_{ij}(\tau)$ is the effective number of counts. The determination of statistical inefficiencies for univariate signals is well established [7]. Determining $s_{ij}(\tau)$ for transition count matrices is an open problem and a first approach is made in this paper. Herein, we make an approach to to determine statistical inefficiencies by row, $I_i(\tau) = I_{ij}(\tau)$ for all j, resulting in the likelihood:

$$\mathbb{P}(C|P) \propto \prod_i \left(\prod_j p_{ij}^{c_{ij}} \right)^{I_i} \tag{4}$$

If we can determine a global statistical inefficiency that is constant for all states, $I(\tau) = I_{ij}(\tau)$ for all i, j, we obtain:

$$\mathbb{P}(C|P) \propto \left(\prod_{t=1}^{\tau} \prod_{i,\,j} p_{ij}^{c_{ij}^{(t)}} \right)^{I} = \prod_{i,\,j} p_{ij}^{c_{ij}^{\mathrm{eff}}}. \tag{5}$$

where $c_{ij}^{(t)}$ is the count matrix for the t-shifted subsequence, i.e. $\{s_t,\, s_{t+\tau},\, ...\}$. For the choice $I = \tau^{-1}$ (see above) the likelihood is given by a geometric mean of the likelihoods at t-shifted subsequences and equivalently as the likelihood of the arithmetic mean count matrix, $c_{ij}^{\mathrm{eff}} = c_{ij}/\tau$. For different choices of I we can interpret the likelihood as a weighted geometric mean of shifted sequences.

Conditional statistical inefficiencies

Let us consider again the discrete trajectory $\{x_t\}$. We can write:

$$p_{ij}(\tau) = \frac{c_{ij}}{c_i} = \frac{1}{c_i} \sum_t 1_i(x_t)\, 1_j(x_{t+\tau})$$

Now we filter our sequence into n sequences of target states with the same starting state i:

$$Y^{(i)} = (x_{t+\tau} | x_t = i)_{t=1,\,...,\,T-\tau}.$$

For example, if our sequence is $X = (1, 0, 0, 0, 0, 1, 1, 1, 0, 0)$, then, at lag time $\tau = 1$ we arrive at the sequences:

$$\begin{aligned} Y^{(0)} &= (0, 0, 0, 1, 0) \\ Y^{(1)} &= (0, 1, 1, 0) \end{aligned}$$

The sequence $Y^{(i)}$ has c_i elements. The transition probability is now given as:

$$p_{ij}(\tau) = \frac{1}{c_i} \sum_t 1_j(y_t^{(i)}). \tag{6}$$

Using $Y^{(1)}$ and $Y^{(2)}$ from in the above example in (6), we get

$$P(\tau = 1) = \begin{bmatrix} 0.8 & 0.2 \\ 0.5 & 0.5 \end{bmatrix}$$

which is indeed identical to the maximum likelihood estimator computed from X directly. Thus, we have transformed our data to a family of signals $1_j(y_t^{(i)})$ from which our quantity of interest, P, can be calculated as an ordinary arithmetic average of the signal. This makes the framework of statistical inefficiency [7] available to estimate the effective number of counts in the denominators, $c_i^{\mathrm{eff}} = c_i I_i$. This framework is briefly described in the Appendix. First, we compute the damped autocorrelation time $\Delta_{i,j}$ of each signal $1_j(y_t^{(i)})$ as:

$$\Delta_{ij} = \frac{1}{2} + \sum_{t=2}^{N} A_{ij}(t) \left(1 - \frac{t}{N} \right)$$

where $A_{ij}(t)$ is the normalized autocorrelation function of the sequence $1_j(y_t^{(i)})$. In practice we compute Δ_{ij} by only computing the sum until $A_{ij}(t)$ first passes through 0, in order to avoid integrating noise at large values of t. The statistical inefficiency of each signal is now given by $(2\Delta_{ij})^{-1}$. However, we won't apply the effective counts to individual transitions but rather to individual rows. This is because (1) the estimate of an individual Δ_{ij} itself involves significant statistical error and is thus better averaged over multiple transition pairs, and (2) we want to be able to apply Eq. (6) and must therefore take the effective number of counts per row:

$$c_i^{\mathrm{eff}} = \sum_j \frac{c_{ij}}{2\Delta_{ij}}$$

The row-wise statistical inefficiency is

$$I_i = \frac{c_i^{\text{eff}}}{c_i}.$$

Finally, we rescale the count matrix to obtain the effective count matrix C^{eff} as:

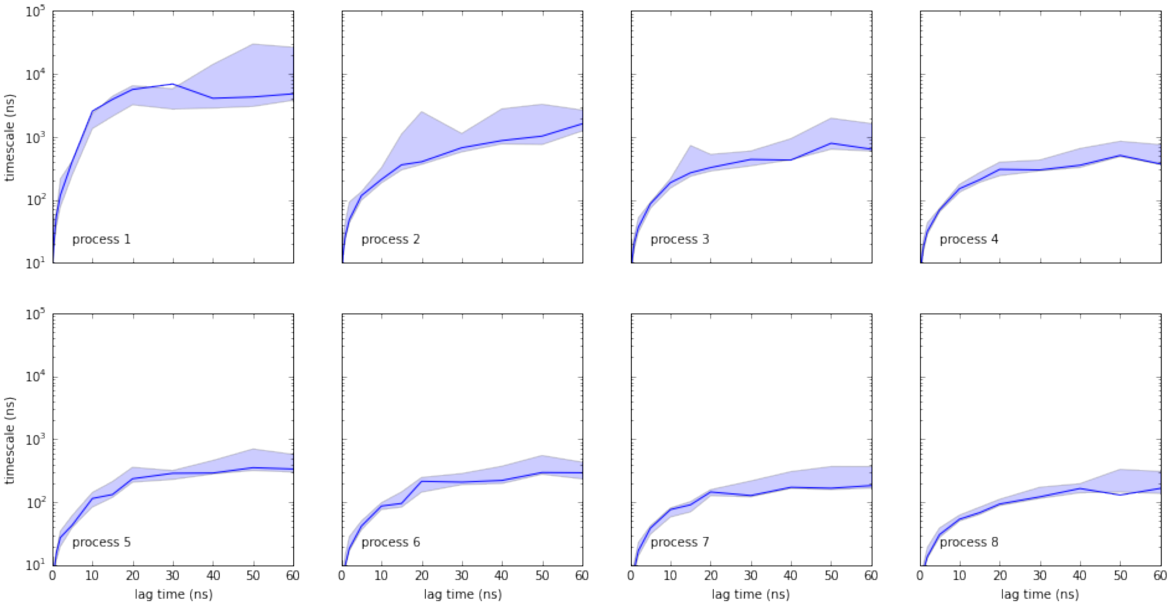
$$c_{ij}^{\text{eff}} = I_i c_{ij}.$$

The most significant approximation made by the present approach is that we only use correlation times of the individual signals $1_j(y_t^{(i)})$ and neglect correlations between these signals. This approximation is currently made for both computational and conceptual reasons.

Now we use the transition counts c_{ij}^{eff} and pass the result to the estimator (MLE or Bayesian).

Illustration

As an illustration consider the calculation of relaxation times for a Dynamin protein domain motion computed from 40 trajectories of 100 ns, as described in [12]. In that publication, errors of similar magnitude have been computed by bootstrapping trajectories. Shown below are the eight longest relaxation timescales computed from a Markov model estimated at a lag time between 0 and 60 nanoseconds (details of Markov model construction in [12]). Note that a Bayesian approach to sampling errors in that data has as previously been unfeasible as the question of transition counting had not been addressed. The two traditional counting modes described above both fail - the sliding window approach vastly overcounts transitions and thus error bars are very tightly around the maximum likelihood estimate, while the lag-sampling approach loses almost all of the data (retaining 120 out of 40000 time steps at a lag time of 50 ns), with the consequence of losing connectivity and rendering any estimation impossible. The effective count matrix estimation described herein for the first time allows us to estimate error bars of reasonable size for such data using a Bayesian approach. Shown below are 95% error intervals that surround the maximum likelihood estimator (solid) for most estimates - both the maximum likelihood estimator and the error bars have been computed using the effective count matrix estimated at each lag time. The error intervals are obtained from running 250 samples with 20 intervening steps using the reversible transition matrix sampler described in [17] and implemented in pyEMMA - www.pyemma.org.



Discussion

The method proposed here presents a first step towards effective count matrix estimation for non-Markovian discrete time series. Addressing this problem is essential for being able to estimate correct error bars of Markov model, and it is thus surprising that it has not yet been addressed at all while a significant number of papers have been published on how to estimate error bars *assuming* that a matrix of statistically independent transition counts is at hand.

We are fully aware that the proposed method is only a first approach to a problem and still has significant deficiencies. For this reason we present the approach in the form of a preprint and hope that it will stimulate scientific discussion in the community, and eventually lead to a more refined method. In particular, the current method does not take cross-correlations between different transition pairs into account, and therefore probably estimates the effective count matrix inaccurately. We have initial data suggesting that the errors estimated by our method are somewhat (roughly a factor of 1.5 to 2) underestimated, but this depends on the system studied and we have therefore not shown such data here.

Acknowledgements

We thank Hao Wu, Fabian Paul and Benjamin Trendelkamp-Schroer (all FU Berlin) for valuable discussions. F.N. was funded in part by ERC starting grant “pcCell”.

Appendix A: Estimators and autocorrelation times

Here we give a short derivation of statistical inefficiencies, following the presentation in [7].

Suppose we have a data sequence $\{x_t\}$ (discrete or continuous) of length N, e.g. generated by molecular dynamics or Monte Carlo. An estimator for the expectation value is the arithmetic mean:

$$\bar{x} = \frac{1}{N}\sum_{t=1}^{N} x_t \tag{7}$$

If the samples are taken such that they are all uncorrelated, all N samples are effective in reducing the uncertainty of the estimator. The variance of the estimator would then be:

$$\sigma_{\bar{x}}^2 = \frac{\sigma_x^2}{N}$$

where σ_x^2 is the variance of the signal.

However, if the sequence $\{x_t\}$ is time-correlated, the variance of the estimator depends on a smaller number of effective counts and therefore the uncertainty diminishes slower. Using (7), we can express the variance of the mean as follows:

$$\begin{aligned}
\sigma_{\bar{x}}^2 &= \langle \bar{x}^2 \rangle - \langle \bar{x} \rangle^2 \\
&= \left\langle \left(\frac{1}{N}\sum_{t=1}^{N} x_t \right)^2 \right\rangle - \left\langle \frac{1}{N}\sum_{t=1}^{N} x_t \right\rangle^2 \\
&= \frac{1}{N^2}\left\langle \sum_{s,t=1}^{N} x_s x_t \right\rangle - \frac{1}{N^2}\left\langle \sum_{s=1}^{N} x_t \right\rangle \left\langle \sum_{t=1}^{N} x_t \right\rangle \\
&= \frac{1}{N^2}\sum_{s,t=1}^{N} \langle x_s x_t \rangle - \frac{1}{N^2}\sum_{s,t=1}^{N} \langle x_s \rangle \langle x_t \rangle
\end{aligned}$$

Collecting diagonal and offdiagonal terms yields:

$$\begin{aligned}
\sigma_{\bar{x}}^2 &= \frac{1}{N^2}\sum_{s=1}^{N} (\langle x_s^2 \rangle - \langle x_s \rangle^2) + \frac{1}{N^2}\sum_{s \neq t} (\langle x_s x_t \rangle - \langle x_s \rangle \langle x_t \rangle) \\
&= \frac{1}{N}\left[\sigma_x^2 + \frac{2}{N}\sum_{s=1}^{N}\sum_{t=s+1}^{N} (\langle x_s x_t \rangle - \langle x_s \rangle \langle x_t \rangle) \right]
\end{aligned}$$

where we have used that the first term is equal to the data variance, and in the second term we have used the $s \leftrightarrow t$ symmetry.

Now we use time invariance of the expectation value, i.e. $\langle x_t \rangle = \langle x_{t+k} \rangle$ and $\langle x_t x_{t+k} \rangle = \langle x_1 x_{1+k} \rangle$, and write the second term as:

$$\begin{aligned}
S &= \sum_{s=1}^{N} \sum_{t=s+1}^{N} \left(\langle x_s x_t \rangle - \langle x_s \rangle \langle x_t \rangle \right) \\
&= \sum_{t=2}^{N} (\langle x_1 x_t \rangle - \langle x_1 \rangle \langle x_t \rangle) + \sum_{t=3}^{N} (\langle x_2 x_t \rangle - \langle x_2 \rangle \langle x_t \rangle) \\
&\quad +\ldots + (\langle x_{N-1} x_N \rangle - \langle x_{N-1} \rangle \langle x_N \rangle) \\
&= \sum_{t=2}^{N} (\langle x_1 x_t \rangle - \langle x_1 \rangle \langle x_t \rangle) + \sum_{t=2}^{N-1} (\langle x_1 x_t \rangle - \langle x_1 \rangle \langle x_t \rangle) \\
&\quad +\ldots + (\langle x_1 x_2 \rangle - \langle x_1 \rangle \langle x_2 \rangle) \\
&= \sum_{t=1}^{N} (N-t)(\langle x_1 x_t \rangle - \langle x_1 \rangle \langle x_t \rangle).
\end{aligned}$$

Resubstituting this expression allows us to write the variance of the estimator as:

$$\sigma_{\bar{x}}^2 = \frac{1}{N} \left[\sigma_x^2 + 2 \sum_{t=1}^{N} (\langle x_1 x_t \rangle - \langle x_1 \rangle \langle x_t \rangle) \left(1 - \frac{t}{N} \right) \right]$$

Factoring out the variance yields:

$$\sigma_{\bar{x}}^2 = \frac{\sigma_x^2}{N} \left[1 + 2 \sum_{t=2}^{N} A(t) \left(1 - \frac{t}{N} \right) \right]$$

where $A(t)$ is the normalized autocorrelation function $A(0) = 1$:

$$A(t) = \frac{\langle x_s x_{s+t} \rangle - \langle x_t \rangle^2}{\langle x_t^2 \rangle - \langle x_t \rangle^2}.$$

The damped autocorrelation time is defined by:

$$\tau_d = \frac{1}{2} + \sum_{t=2}^{N} A(t) \left(1 - \frac{t}{N} \right)$$

Yielding the variance of the mean:

$$\sigma_{\bar{x}}^2 = \frac{\sigma_x^2}{N_{\text{eff}}}$$

with the effective sample count:

$$N_{\text{eff}} = \frac{N}{2\tau_d}$$

The factor $1/2\tau_d$ is called statistical inefficiency.

References

[1] G. R. Bowman, K. A. Beauchamp, G. Boxer, and V. S. Pande. Progress and challenges in the automated construction of Markov state models for full protein systems. *J. Chem. Phys.*, 131:124101, 2009.

[2] G. R. Bowman, V. S. Pande, and F. Noé, editors. *An Introduction to Markov State Models and Their Application to Long Timescale Molecular Simulation.*, volume 797 of *Advances in Experimental Medicine and Biology*. Springer Heidelberg, 2014.

[3] N. V. Buchete and G. Hummer. Coarse Master Equations for Peptide Folding Dynamics. *J. Phys. Chem. B*, 112:6057–6069, 2008.

[4] P. Diaconis and S. W. W. Rolles. Bayesian analysis for reversible markov chains. *Ann. Statist.*, 34:1270–1292, 2006.

[5] N. Djurdjevac, M. Sarich, and C. Schütte. Estimating the eigenvalue error of Markov State Models. *Multiscale Model. Simul.*, 10:61–81, 2012.

[6] N. S. Hinrichs and V. S. Pande. Calculation of the distribution of eigenvalues and eigenvectors in Markovian state models for molecular dynamics. *J. Chem. Phys.*, 126:244101, 2007.

[7] W. Janke. Statistical analysis of simulations: Data correlations and error estimation. In A. Muramatsu J. Grotendorst, D. Marx, editor, *Quantum Simulations of Complex Many-Body Systems: From Theory to Algorithms, Lecture Notes*, volume NIC Series, Vol. 10,, pages 423–445. John von Neumann Institute for Computing, Jülich, 2002.

[8] P. Metzner, F. Noé, and C. Schütte. Estimation of transition matrix distributions by monte carlo sampling. *Phys. Rev. E*, 80:021106, 2009.

[9] F. Noé. Probability Distributions of Molecular Observables computed from Markov Models. *J. Chem. Phys.*, 128:244103, 2008.

[10] J.-H. Prinz, J. D. Chodera, and F. Noé. Spectral rate theory for two-state kinetics. *Phys. Rev. X*, 4:011020, 2014.

[11] J.-H. Prinz, H. Wu, M. Sarich, B. Keller, M. Senne, M. Held, J. D. Chodera, C. Schütte, and F. Noé. Markov models of molecular kinetics: Generation and validation. *J. Chem. Phys.*, 134:174105, 2011.

[12] T. F. Reubold, K. Faelber, N. Plattner, Y. Posor, K. Branz, U. Curth, J. Schlegel, R. Anand, D. Manstein, F. Noé, V. Haucke, O. Daumke, and S. Eschenburg. Crystal structure of the dynamin tetramer. *Nature (in press)*, 2015.

[13] M. Sarich, F. Noé, and C. Schütte. On the approximation quality of markov state models. *SIAM Multiscale Model. Simul.*, 8:1154–1177, 2010.

[14] C. Schütte, A. Fischer, W. Huisinga, and P. Deuflhard. A Direct Approach to Conformational Dynamics based on Hybrid Monte Carlo. *J. Comput. Phys.*, 151:146–168, 1999.

[15] W. C. Swope, J. W. Pitera, and F. Suits. Describing protein folding kinetics by molecular dynamics simulations: 1. Theory. *J. Phys. Chem. B*, 108:6571–6581, 2004.

[16] B. Trendelkamp-Schroer and F. Noé. Efficient bayesian estimation of markov model transition matrices with given stationary distribution. *J. Phys. Chem.*, 138:164113., 2013.

[17] B. Trendelkamp-Schroer, H. Wu, F. Paul, and F. Noé. Estimation and uncertainty of reversible markov models. *in preparation.*