Supplementary Materials to Accompany: Estimation and Selection for High-Order Markov Chains with Bayesian Mixture Transition Distribution Models

Matthew Heiner*

Department of Statistics, Brigham Young University
and
Athanasios Kottas†
Department of Statistics, University of California, Santa Cruz

September 7, 2021

S1 Marginal distributions

We report the marginal distributions of observations associated with the Dirichlet and SBM priors for probability vectors given originally in Heiner et al. (2019). These distributions can be useful for computing Bayes factors in addition to facilitating the MCMC algorithms described in Appendix S2.

Consider a length-N sequence of independent random variables $\{x_t\} \in \{1, \ldots, J\}^N$ with common distribution $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_J)$. Given $\boldsymbol{\theta}$, the probability of the sequence is $\prod_t \theta_{x_t} = \theta_1^{n_1} \cdots \theta_J^{n_J}$ where the sufficient statistics in $\boldsymbol{n} = (n_1, \ldots, n_J)$ count the occurrences of each category.

*M. Heiner (heiner@stat.byu.edu) is Assistant Professor, Department of Statistics, Brigham Young University.

†A. Kottas (thanos@soe.ucsc.edu) is Professor, Department of Statistics, University of California, Santa Cruz.

If $\boldsymbol{\theta}$ follows a Dirichlet distribution with shape parameter vector $\boldsymbol{\alpha}$, then the marginal (prior predictive) distribution of $\{x_t\}$ is given by

$$p(\{x_t\}) = \int p(\{x_t\} \mid \boldsymbol{\theta})\, p(\boldsymbol{\theta})\, \mathrm{d}\boldsymbol{\theta} = \frac{\Gamma(\sum_j \alpha_j)}{\prod_j \Gamma(\alpha_j)} \frac{\prod_j \Gamma(\alpha_j + n_j)}{\Gamma(\sum_j \alpha_j + n_j)} = \frac{\mathrm{MVB}(\boldsymbol{\alpha} + \boldsymbol{n})}{\mathrm{MVB}(\boldsymbol{\alpha})}\,, \tag{S1}$$

where $\mathrm{MVB}(\cdot)$ denotes the multivariate beta function.

Now suppose $\boldsymbol{\theta}$ follows the SBM distribution with parameters $\{\pi_{1,j}\}$, $\{\pi_{3,j}\}$, η, $\{\gamma_j\}$, and $\{\delta_j\}$. Let

$$g_j(a_j, b_j, \boldsymbol{n}) \equiv \frac{\Gamma(a_j + b_j)}{\Gamma(a_j^* + b_j^*)} \frac{\Gamma(a_j^*)}{\Gamma(a_j)} \frac{\Gamma(b_j^*)}{\Gamma(b_j)}\,,$$

with $a_j^* \equiv a_j + n_j$, and $b_j^* \equiv b_j + \sum_{h=j+1}^{J} n_h$. Then the marginal distribution of $\{x_t\}$ has probability mass function

$$p(\{x_t\}) = \prod_{j=1}^{J-1} [\pi_{1,j}\, g_j(1, \eta, \boldsymbol{n}) + \pi_{2,j}\, g_j(\gamma_j, \delta_j, \boldsymbol{n}) + \pi_{3,j}\, g_j(\eta, 1, \boldsymbol{n})]\,. \tag{S2}$$

S2 MCMC algorithm details

Following the hierarchical MMTD model, the joint posterior distribution of all unknown parameters is given up to proportionality:

$$\begin{aligned} p\big(\{\zeta_t\}_{t=L+1}^{T}, &\boldsymbol{\Lambda}, \{\boldsymbol{\lambda}^{(r)}\}_{r=1}^{R}, \{\boldsymbol{\mathcal{Q}}^{(r)}\}_{r=0}^{R}, \mid \{s_t\}_{t=1}^{T}\big) \propto \\ &p(\boldsymbol{\Lambda})\, p(\boldsymbol{\mathcal{Q}}^{(0)}) \prod_{r=1}^{R} \left[p(\boldsymbol{\lambda}^{(r)}) \prod_{j=1}^{K^r} p\left((\boldsymbol{\mathcal{Q}}^{(r)})_{\cdot,j} \right) \right] \times \\ &\prod_{t=L+1}^{T} \left[p\left(\zeta_t \mid \boldsymbol{\Lambda}, \{\boldsymbol{\lambda}^{(r)}\}_{r=1}^{R}\right)\, p\left(s_t \mid \zeta_t, \{\boldsymbol{\mathcal{Q}}^{(r)}\}_{r=0}^{R}, \{s_{t-\ell}\}_{\ell=1}^{L}\right) \right]\,, \end{aligned} \tag{S3}$$

where $(\boldsymbol{\mathcal{Q}}^{(r)})_{\cdot,j}$ denotes column j from a matricized version of $\boldsymbol{\mathcal{Q}}^{(r)}$.

We use a collapsed Gibbs sampler that cycles through conditional updates of $\{\zeta_t\}$, $\boldsymbol{\Lambda}$ and each $\boldsymbol{\lambda}^{(r)}$, with all $\{\boldsymbol{\mathcal{Q}}^{(r)}\}$ marginalized out of the posterior. It is then straightforward to draw each $\boldsymbol{\mathcal{Q}}^{(r)}$ from the full conditional distributions given below.

In what follows, let $Z(\zeta)$ and $\boldsymbol{z}(\zeta)$ map ζ to its corresponding Z and $\boldsymbol{z}$ respectively. Also, let $\varrho_r(\boldsymbol{s})$ be a unique map from each possible length-r vector of lagged states $\boldsymbol{s} \in \{1, \ldots, K\}^r$ to the corresponding column index of the flattened (matricized) $\boldsymbol{\mathcal{Q}}^{(r)}$. Further, let $\boldsymbol{s}_{t-1}(\boldsymbol{z})$ be a function accepting a lag configuration $\boldsymbol{z}$ and returning the values of the states at those selected lags from the vector $(s_{t-1}, s_{t-2}, \ldots, s_{t-L})$. For example, if $\boldsymbol{z}_t = (2, 5)$, then $\boldsymbol{s}_{t-1}(\boldsymbol{z}_t)$ will return the vector (s_{t-2}, s_{t-5}).

For each $r = 1, \ldots, R$, let $\boldsymbol{N}^{(r)}$ be a matrix containing transition counts for which the (k, j) entry is the cardinality of $\{t : Z(\zeta_t) = r \text{ and } \varrho_r(\boldsymbol{s}_{t-1}(\boldsymbol{z}(\zeta_t))) = j \text{ and } s_t = k\}$. Also let the kth entry of vector $\boldsymbol{N}^{(0)}$ be the cardinality of $\{t : Z(\zeta_t) = 0 \text{ and } s_t = k\}$. Integrating all $\{\boldsymbol{\mathcal{Q}}^{(r)}\}$ from the full joint posterior proportional to (S3) yields

$$\begin{aligned} p\left(\{\zeta_t\}, \boldsymbol{\Lambda}, \{\boldsymbol{\lambda}^{(r)}\} \mid \{s_t\}\right) \propto \mathrm{SBM}(\boldsymbol{\Lambda}) \prod_r \left[\mathrm{SDM}(\boldsymbol{\lambda}^{(r)})\right] \prod_t \left[\Lambda_{Z(\zeta_t)}\, \lambda_{(\boldsymbol{z}(\zeta_t))}^{(Z(\zeta_t))}\right] \times \\ p\left(\boldsymbol{N}^{(0)} \mid \{\zeta_t\}, \{s_t\}\right) \prod_{r=1}^{R} \left[\prod_{j=1}^{K^r} p\left((\boldsymbol{N}^{(r)})_{\cdot,j} \mid \{\zeta_t\}, \{s_t\}\right)\right], \end{aligned} \tag{S4}$$

where

$$p\left((\boldsymbol{N}^{(r)})_{\cdot,j} \mid \{\zeta_t\}, \{s_t\}\right) = \frac{\Gamma(\sum_k (\boldsymbol{\alpha}_{Q^{(r)}})_k)}{\prod_k \Gamma((\boldsymbol{\alpha}_{Q^{(r)}})_k)} \frac{\prod_k \Gamma((\boldsymbol{\alpha}_{Q^{(r)}})_k + (\boldsymbol{N}^{(r)})_{k,j})}{\Gamma(\sum_k (\boldsymbol{\alpha}_{Q^{(r)}})_k + (\boldsymbol{N}^{(r)})_{k,j})}, \tag{S5}$$

and $p\left(\boldsymbol{N}^{(0)} \mid \{\zeta_t\}, \{s_t\}\right)$ takes the same Dirichlet-multinomial form.

The $\{\zeta_t\}$ are updated individually using the collapsed conditional probability mass function given by

$$\begin{aligned} p\left(\zeta_t \mid \cdots, -\{\boldsymbol{\mathcal{Q}}^{(r)}\}\right) \propto \Lambda_{Z(\zeta_t)}\, \lambda_{(\boldsymbol{z}(\zeta_t))}^{(Z(\zeta_t))} p\left(\boldsymbol{N}^{(0)} \mid \{\zeta_t\}, \{s_t\}\right) \times \\ \prod_{r=1}^{R} \left[\prod_{j=1}^{K^r} p\left((\boldsymbol{N}^{(r)})_{\cdot,j} \mid \{\zeta_t\}, \{s_t\}\right)\right], \end{aligned} \tag{S6}$$

where we modify $\{\boldsymbol{N}^{(r)}\}$ to reflect each possible $\zeta_t \in \left\{0, 1, \ldots, \left[\binom{L}{1} + \binom{L}{2} + \ldots + \binom{L}{R}\right]\right\}$. Instead of sampling from (S6) directly, we Metropolize the update by first drawing a candidate with probability mass proportional to (S6), excluding the current state. The Metropolis acceptance ratio is then the sum over all probabilities excluding the current state, divided

by the sum over all full conditional probabilities excluding the candidate state (Liu, 1996; Robert and Casella, 2004, p. 394).

The remaining full conditional distributions do not change with marginalization:

- $p(\boldsymbol{\Lambda} \mid \cdots) \propto p(\boldsymbol{\Lambda}) \prod_t p(\zeta_t \mid \boldsymbol{\Lambda}, \{\boldsymbol{\lambda}^{(r)}\}) \propto \mathrm{SBM}(\boldsymbol{\Lambda}; \boldsymbol{\pi}_1, \boldsymbol{\pi}_3, \eta, \boldsymbol{\gamma}, \boldsymbol{\delta}) \prod_t \Lambda_{Z(\zeta_t)}$, a conjugate SBM-multinomial update using the counts of $Z(\zeta_t)$ in each of $\{0, 1, \ldots, R\}$. See Section 2.2 of Heiner et al. (2019) for a sampling strategy based on the stick-breaking construction, noting that in the MMTD model application, π_1 and π_3 change with index $r = 0, 1, \ldots, R$.

- $p(\boldsymbol{\lambda}^{(r)} \mid \cdots) \propto p(\boldsymbol{\lambda}^{(r)}) \prod_t p(\zeta_t \mid \boldsymbol{\Lambda}, \{\boldsymbol{\lambda}^{(r)}\}) \propto \mathrm{SDM}(\boldsymbol{\lambda}^{(r)}; \boldsymbol{\alpha}_\lambda^{(r)}, \beta_\lambda^{(r)}) \prod_{t:Z(\zeta_t)=r} \lambda^{(r)}_{(\boldsymbol{z}(\zeta_t))}$ independently for $r \in \{1, \ldots, R\}$. Here, $\boldsymbol{\lambda}^{(r)}$ is indexed by the $\binom{L}{r}$ possible sets of lags. This is a conjugate SDM-multinomial update using the counts of the $\binom{L}{r}$ unique lag configurations $\boldsymbol{z}_t$ within level r. The full conditional is a SDM distribution with $\beta_\lambda^{(r)}$ and with the multinomial counts added to $\boldsymbol{\alpha}_\lambda^{(r)}$, analogous to Dirichlet full conditionals.

- $p(\boldsymbol{\mathcal{Q}}^{(0)} \mid \cdots) \propto p(\boldsymbol{\mathcal{Q}}^{(0)}) \prod_{t:Z(\zeta_t)=0} p\left(s_t \mid \zeta_t, \{\boldsymbol{\mathcal{Q}}^{(r)}\}_{r=0}^L, \{s_{t-\ell}\}_{\ell=1}^L\right)$
  $= \mathrm{Dir}(\boldsymbol{\mathcal{Q}}^{(0)} \mid \boldsymbol{\alpha}_{Q^{(0)}}) \prod_{t:Z(\zeta_t)=0} (\boldsymbol{\mathcal{Q}}^{(0)})_{s_t}$, a standard conjugate Dirichlet-multinomial update using the counts collected in $\boldsymbol{N}^{(0)}$. The full conditional distribution is then $\mathrm{Dir}(\boldsymbol{\alpha}_{Q^{(0)}} + \boldsymbol{N}^{(0)})$.

- $p\left((\boldsymbol{\mathcal{Q}}^{(r)})_{\cdot,j} \mid \cdots\right) \propto p\left((\boldsymbol{\mathcal{Q}}^{(r)})_{\cdot,j}\right) \prod_{\{t:Z(\zeta_t)=r \text{ and } \varrho_r(\boldsymbol{s}_{t-1}(\boldsymbol{z}(\zeta_t)))=j\}} \times$
  $p\left(s_t \mid \zeta_t, \{\boldsymbol{\mathcal{Q}}^{(r)}\}_{r=0}^R, \{s_{t-\ell}\}_{\ell=1}^L\right)$
  $\propto \mathrm{Dir}\left((\boldsymbol{\mathcal{Q}}^{(r)})_{\cdot,j} \mid \boldsymbol{\alpha}_{Q^{(r)}}\right) \prod_{\{t:Z(\zeta_t)=r \text{ and } \varrho_r(\boldsymbol{s}_{t-1}(\boldsymbol{z}(\zeta_t)))=j\}} (\boldsymbol{\mathcal{Q}}^{(r)})_{s_t,j}$,
  independently for $r = 1, \ldots, R$, and $j = 1, \ldots, K^r$. Again, this is a standard conjugate Dirichlet-multinomial update using the transition counts collected in $(\boldsymbol{N}^{(r)})_{\cdot,j}$. The full conditional distribution is then $\mathrm{Dir}(\boldsymbol{\alpha}_{Q^{(r)}} + (\boldsymbol{N}^{(r)})_{\cdot,j})$.

S3 Details for simulation studies

To evaluate estimation of transition probabilities, we fit each model using a subset of training samples and compared point estimates of the transition distributions to the true transition distributions at each validation time. Specifically, for validation time point t', each model produced a vector $\hat{\boldsymbol{p}}_{t'}$ to estimate each $p_{t'}^{(k)} = \Pr(s_{t'} = k \mid s_{t'-1}, \ldots, s_{t'-L}) = (\boldsymbol{\Omega})_{k, s_{t'-1}, \ldots, s_{t'-L}}$, for $k = 1, \ldots, K$. In Bayesian models, the point estimate is the Monte Carlo-computed posterior mean of $\hat{\boldsymbol{p}}_{t'}$. In non-Bayesian models, $\hat{\boldsymbol{p}}_{t'}$ is computed from the optimized model fit. For each validation time point, we computed the $\mathcal{L}_1$ loss given by $L_{t'} = \sum_{k=1}^{K} |\hat{p}_{t'}^{(k)} - p_{t'}^{(k)}|$. The reported loss metric for model comparison is $100 \times \sum_{t'} L_{t'}/(KT')$, i.e., 100 times the mean $\mathcal{L}_1$ loss across the $T' = 1{,}000$ validation points.

S3.1 Simulation 2 results

The mixture decomposition profile for Simulation 2 given in Table S1 (left) reveals full, fifth-order dynamics in which all lags interact. At most one subset of lags carries appreciable weight. The full generating model has 32 free parameters.

All models were fit to the first $T = 100$, $T = 200$, and $T = 500$ training samples from Simulation 2. Here, we assume that the modeler is considering a horizon of seven lags, which we use where possible to promote equitable comparisons. Results of the mean $\mathcal{L}_1$ loss across the 1,000 validation points are given in Table S1 (right).

S3.1.1 Sample size 100

The VLMC model performs well for all sample sizes in this scenario, and best among all methods for $T = 100$. This is mildly surprising, given that the generating tensor is not sparse, exhibiting full, fifth-order dependence. In the $T = 100$ case, higher-level interactions are not estimable in the multinomial logit model. Models with lower-level interactions perform comparably across model classes. Again, regularization preserves integrity of over-specified models.

MMTD models tend to mix with lag 2 (and occasionally lag 4) in the first level. They tend not to support higher levels, and fail to coalesce around a best two, three, or four-lag

Table S1: Simulation 2 (fifth-order, binary chain). (Left) Mixture decomposition profile for the simulation-generating tensor. (Right) Results for transition probability estimation under various models/settings using sample sizes $T = 100$, $T = 200$, and $T = 500$. The lowest mean loss is highlighted with bold font.

Level	Wt. ($\tilde{\lambda}$)	Lag config. ($\mathcal{C}$)
0	.066	
1	.026	5
2	.043	2, 4
2	.040	1, 2
3	.027	1, 2, 4
3	.009	3, 4, 5
4	.155	2, 3, 4, 5
4	.014	1, 2, 4, 5
4	.003	1, 3, 4, 5
4	.002	1, 2, 3, 4
5	.616	1, 2, 3, 4, 5
Lag	**Weight (total contrib.)**	
1	.702	
2	.897	
3	.784	
4	.868	
5	.823	

Model	$100 \times$ Mean $\mathcal{L}_1$ loss		
	$T = 100$	$T = 200$	$T = 500$
VLMC	**20.51**	15.46	12.12
LogitMC(7, 1)	24.30	20.03	18.53
LogitMC(7, 2)	26.15	16.26	14.66
LogitMC(7, 4)	n/a	n/a	n/a
LogitMC(7, 1, lso)	24.26	20.28	18.57
LogitMC(7, 2, lso)	24.51	15.78	14.65
LogitMC(7, 4, lso)	23.38	15.18	10.12
LogitMC(7, 7, lso)	22.22	**13.66**	10.28
MMTD(7, 1)	24.09	23.91	21.70
MMTD(7, 2)	23.71	20.84	19.51
MMTD(7, 4)	23.29	15.79	15.28
MMTD(7, 7)	23.32	13.70	**7.39**

configuration. It appears that the prior structure holds against a weak signal, as the sample size is not sufficiently large to estimate fifth-order dynamics.

S3.1.2 Sample size 200

The larger sample size helps the models reveal higher-order dynamics. The VLMC model is again competitive, and generally outperforms multinomial logit models. Interactions significantly improve logit models, which marginally outperform their MMTD counterparts in this scenario. Low-level MMTD models are unable to leverage increased sample size to the extent that the other models can.

The MMTD(7, 2) model leverages its mixture structure to include lags 1 and 4 on the first level and lag configuration (3, 5) on the second level, yielding an effective order between 3 and 4. The MMTD(7, 4) model mixes primarily over lag configurations (1, 2, 3, 5), (1, 2, 3, 6), (1, 2, 3), and (3, 5). The over-specified MMTD(7, 7) model clearly identifies

the fifth level and correct lag configuration, and mixes with a fourth-level configuration (1, 2, 3, 5).

S3.1.3 Sample size 500

The 32 free parameters are more easily identified from the series with 500 samples. Here, the VLMC model is less competitive, but still performs well. Again, interaction significantly improves multinomial logit model fits, while low-level mixing in MMTD models fails to approximate the dependence structure.

As in the $T = 200$ case, the MMTD(7, 2) model mixes over non-overlapping lag configurations, leveraging the MTD structure in a manner consistent with the model's second intended use. However, it does not yield strong performance gains in this scenario.

The over-specified MMTD(7, 7) model again clearly identifies the true structure, and in this case performs best. Replicate runs of the MMTD(7, 4) model yield losses well below the reported 15.28 (12.12 and 9.73), indicating that MCMC chains are exploring distinct posterior modes. For this reason, we recommend always running multiple parallel chains.

S4 Multiple-chain MCMC analysis

Either model misspecification or high-order dynamics observed in small samples tends to exacerbate mixing challenges in the MMTD model. This is evident in both the seizure and pink salmon analyses, but far more pronounced with the seizure time series. Although the current MCMC algorithm does not generally explore sufficiently rapidly to reliably estimate relative heights of posterior modes, one could employ strategies for combining non-mixing chains. See, for example, Yao et al. (2020) when prediction (or overall transition probability estimation) is the primary goal. Mixing and identifiability issues are more problematic if the goal is inference for effective order or lag relevance. Here, we suggest two straightforward and practical approaches, each employing our decomposition algorithm, that could help in extracting useful information from multiple MCMC runs.

The first approach attempts to de-noise the decomposition results presented in the analyses by replacing posterior draws for component $\boldsymbol{\mathcal{Q}}$ tensors with full-conditional point

summaries (i.e., means). This requires that either the decomposition is calculated during MCMC, or draws for the ζ_t allocation variables are stored for post-processing. This strategy enhances the differences between MCMC and decomposition estimates in the pink salmon analysis, and makes the lag-inclusion index more pronounced. If estimated high-level interactions are superfluous, this approach may reduce, but does not eliminate posterior multimodality.

In the case that multiple chains explore distinct modes, or a single chain visits multiple modes, one strategy is to concatenate these chains and separate the modes with a clustering algorithm. In the case of relatively few modes, one may proceed to summarize estimates within each mode, perhaps by applying the decomposition to a mode-representing point estimate of the full transition tensor, or examining configuration weights $\boldsymbol{\lambda}^{(r)}$ directly. Mode separation may occur at the $\boldsymbol{\Lambda}$ or $\boldsymbol{\lambda}^{(r)}$ levels. We typically start by attempting a k-means clustering algorithm on traces of $\boldsymbol{\Lambda}$. Clustering on traces of log-likelihood and lag-inclusion weights proved ineffective in the seizure analysis.

References

Heiner, M., Kottas, A., and Munch, S. (2019), "Structured priors for sparse probability vectors with application to model selection in Markov chains," *Statistics and Computing*, 29, 1077–1093.

Liu, J. S. (1996), "Peskun's theorem and a modified discrete-state Gibbs sampler." *Biometrika*, 83.

Robert, C. P. and Casella, G. (2004), *Monte Carlo Statistical Methods*, Springer-Verlag New York, 2nd edition.

Yao, Y., Vehtari, A., and Gelman, A. (2020), "Stacking for Non-mixing Bayesian Computations: The Curse and Blessing of Multimodal Posteriors," *arXiv preprint arXiv:2006.12335*.