DYNAMIC MODELING OF SPIKE COUNT DATA WITH CONWAY-MAXWELL POISSON VARIABILITY

BY GANCHAO WEI* AND IAN H. STEVENSON

University of Connecticut, *ganchao.wei@uconn.edu

The abstract should summarize the contents of the paper. It should be clear, descriptive, self-explanatory and not longer than 200 words. It should also be suitable for publication in abstracting services. Formulas should be used as sparingly as possible within the abstract. The abstract should not make reference to results, bibliography or formulas in the body of the paper—it should be self-contained.

This is a sample input file. Comparing it with the output it generates can show you how to produce a simple document of your own.

1. Introduction.

2. Methods.

Here we consider a dynamic GLM with Conway-Maxwell Poisson (CMP) observations to describe time-varying spike counts. We first introduce the model. Although the CMP distribution allows us to flexibly model non-Poisson variability, one major challenge with using this model is that there are not closed-form solutions for the CMP likelihood. Here, we infer fit the model using a global Gaussian approximation, and we discuss several additional technical challenges that arise when using the CMP distribution with a dynamic GLM.

2.1. *Dynamic Conway-Maxwell Poisson Model.*

A count observation y, such as the spike count for a neuron, is assumed to follow the CMP distribution, with parameters λ and ν. The probability mass function (pmf) of CMP distribution is:

$$P(Y=y|\lambda,\nu)=\frac{\lambda^y}{(y!)^\nu}\cdot\frac{1}{Z(\lambda,\nu)} \tag{1}$$

, where $Z(\lambda,\nu)=\sum_{k=0}^{\infty}\frac{\lambda^k}{(k!)^\nu}$ is the normalizing constant. The parameter $\nu\geq 0$ controls different dispersion patterns, i.e. equi- $(\nu=1)$, over- $(0\leq\nu<1)$ or under-dispersion $(\nu>1)$. Three common distributions occur as special cases: 1) the Poisson $(\nu=1)$, 2) the geometric $(\nu=0,\lambda<1)$, and 3) the Bernoulli $(\nu\to\infty)$.

For multiple observations up to T steps, such as simultaneous spike counts from n neurons, denote the counts at time bin t as $\boldsymbol{y}_t=(y_{1t},\ldots,y_{nt})'$, for $t=1,\ldots,T$. The corresponding CMP parameters at t are $\boldsymbol{\lambda}_t=(\lambda_{1t},\ldots,\lambda_{nt})'$ and $\boldsymbol{\nu}_t=(\nu_{1t},\ldots,\nu_{nt})'$. Previous work has examined the CMP-GLM (Chatla and Shmueli (2018); Sellers and Shmueli (2010)), and here we focus on the dynamic version of this GLM. The CMP parameters at t are modeled by two log-linear models, $\log\boldsymbol{\lambda}_t=\boldsymbol{X}_t\boldsymbol{\beta}_t$ and $\log\boldsymbol{\nu}_t=\boldsymbol{G}_t\gamma_t$, with $\boldsymbol{\beta}_t\in\mathbb{R}^p$ and $\gamma_t\in\mathbb{R}^q$, and $\boldsymbol{X}_t$ and $\boldsymbol{G}_t$ denote known predictors. Under the CMP-GLM, the parameters are static. Here, we assume that they progress linearly with a Gaussian noise.

The observations follow conditionally independent CMP distributions, given the state vector $\boldsymbol{\theta}_t=(\boldsymbol{\beta}_t',\gamma_t')'$.

$$\begin{aligned}\boldsymbol{y}_t &\sim CMP(\boldsymbol{\lambda}_t,\boldsymbol{\nu}_t)\\ \log\boldsymbol{\lambda}_t=\boldsymbol{X}_t\boldsymbol{\beta}_t&,\quad \log\boldsymbol{\nu}_t=\boldsymbol{G}_t\gamma_t\end{aligned} \tag{2}$$

Keywords and phrases: First keyword, second keyword.

While the state vector $\boldsymbol{\theta}_t$ evolves linearly with Gaussian noise:

$$\begin{aligned} \boldsymbol{\theta}_1 &\sim N_{p+q}(\boldsymbol{\theta}_0, \boldsymbol{Q}_0) \\ \boldsymbol{\theta}_t|\boldsymbol{\theta}_{t-1} &\sim N_{p+q}(\boldsymbol{F}\boldsymbol{\theta}_{t-1}, \boldsymbol{Q}) \end{aligned} \tag{3}$$

Given the initial state mean $\boldsymbol{\theta}_0$ and covariance $\boldsymbol{Q}_0$, linear dynamics $\boldsymbol{F}$ and process covariance $\boldsymbol{Q}$.

2.1.1. *Inference by Gaussian approximation.* To fit the model to data we need to estimate the time-varying state vector $\boldsymbol{\Theta} = (\boldsymbol{\theta}_1', \ldots, \boldsymbol{\theta}_T') \in \mathbb{R}^{(p+q)T}$. In this section, we first assume $\boldsymbol{F}$ and $\boldsymbol{Q}$ are known. Since the observations are CMP distributed, we cannot estimate $\boldsymbol{\Theta}$ in closed form. Instead, here we approximate it by a multivariate Gaussian distribution, $P(\boldsymbol{\Theta}|\boldsymbol{Y}) \approx N_{(p+q)T}(\boldsymbol{\Theta}|, \boldsymbol{\mu}, \boldsymbol{\Sigma})$, with $\boldsymbol{Y} = (\boldsymbol{y}_1', \ldots, \boldsymbol{y}_T')'$. The parameters of this Gaussian are found by a global Laplace approximation, i.e. $\boldsymbol{\mu} = \arg\max_{\boldsymbol{\Theta}} P(\boldsymbol{\Theta}|\boldsymbol{Y})$ and $\boldsymbol{\Sigma} = -(\nabla\nabla_{\boldsymbol{\Theta}} \log P(\boldsymbol{\Theta}|\boldsymbol{Y})|_{\boldsymbol{\Theta}-\boldsymbol{\mu}})^{-1}$. The log-posterior is given by:

(4)

$$\log P(\boldsymbol{\Theta}|\boldsymbol{Y}) = \sum_{t=1}^{T} l_t - \frac{1}{2}(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0)'\boldsymbol{Q}_0^{-1}(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0) - \frac{1}{2}\sum_{t=2}^{T}(\boldsymbol{\theta}_t - \boldsymbol{F}\boldsymbol{\theta}_{t-1})'\boldsymbol{Q}^{-1}(\boldsymbol{\theta}_t - \boldsymbol{F}\boldsymbol{\theta}_{t-1})$$

$$l_t = l(\boldsymbol{\theta}_t) = \log P(\boldsymbol{y}_t|\boldsymbol{\theta}_t) = \sum_{i=1}^{n} y_{it}\log\lambda_{it} - \nu_{it}y_{it}! - \log Z(\lambda_{it}, \nu_{it})$$

, where $l(\cdot)$is the log-likelihood. The log-posterior is concave (Gupta, Sim and Ong (2014)), and the Markovian structure of the state vector dynamics makes it possible to optimize by Newton-Raphson (NR) in $\mathcal{O}(T)$ time (Paninski et al. (2010)).After the Newton update, we can further quantify the uncertainty for the CMP parameters and the underlying rates, as in Appendix A.

There are several technical challenges involved with performing the Newton update with CMP observations. Firstly, in order to find the gradient and Hessian we need to calculate moments of Y_{it} and $\log Y_{it}!$, which have no closed forms (Shmueli et al. (2005)). We can calculate these moments by truncated summation. However, when $\lambda \geq 2$ and $\nu \leq 1$, truncated summation is computationally costly since we need many steps for accurate approximation. In this case, we approximate the moments using previous (Chatla and Shmueli (2018); Gaunt et al. (2019)) asymptotic results as in Appendix B. A second challenge is that the Hessian is not robust to outliers. Outliers often result in the Hessian being close to singular or even positive-definite. See details in Appendix C. To ensure robustness, we use Fisher scoring where the observed information is replaced by the expected information. Finally, a third challenge is that the Newton updates take a long time to converge if the initial state estimate is far from the maximum of the posterior, especially when T is large. To resolve this issue, we use a smoothing estimate with local Gaussian approximation as a "warm start". Forward filtering for a dynamic Poisson model has been previously described in Eden et al. (2004), and here we implement CMP filtering following the same rationale. Let $\boldsymbol{\theta}_{t|t-1} = E(\boldsymbol{\theta}_t|\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{t-1})$ and $\boldsymbol{\Sigma}_{t|t-1} = Var(\boldsymbol{\theta}_t|\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{t-1})$ be the mean and variance for the one-step prediction density and $\boldsymbol{\theta}_{t|t} = E(\boldsymbol{\theta}_t|\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t)$ and $\boldsymbol{\Sigma}_{t|t} = Var(\boldsymbol{\theta}_t|\boldsymbol{y}_1, \ldots, \boldsymbol{y}_t)$ be mean and variance for the posterior density, then the filtering update for step t is given by

$$\begin{aligned} \boldsymbol{\theta}_{t|t-1} &= \boldsymbol{F}\boldsymbol{\theta}_{t-1|t-1} \\ \boldsymbol{\Sigma}_{t|t-1} &= \boldsymbol{F}\boldsymbol{\Sigma}_{t-1|t-1}\boldsymbol{F}' + \boldsymbol{Q} \\ \boldsymbol{\theta}_{t|t} &= \boldsymbol{\theta}_{t|t-1} + (\boldsymbol{\Sigma}_{t|t})\left[\frac{\partial l_t}{\partial \boldsymbol{\theta}_t}\right]_{\boldsymbol{\theta}_{t|t-1}} \end{aligned} \tag{5}$$

$$(\boldsymbol{\Sigma}_{t|t})^{-1} = (\boldsymbol{\Sigma}_{t|t-1})^{-1} - \left[\frac{\partial^2 l_t}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_t'}\right]_{\boldsymbol{\theta}_{t|t-1}}$$

Here, to again ensure robustness, we use Fisher scoring when updating the state covariance. We then find smoothed estimates using a backward pass (Rauch, Tung and Striebel (1965)). Although doing smoothing is fast, the estimates can be inaccurate, especially when there are large changes in the state vector. In the forward filtering stage, the Gaussian approximation at each step t is conducted locally at the recursive prior $\boldsymbol{\theta}_{t|t-1}$. This will be statistically inefficient when the recursive prior is too far away from the posterior mode, or when there is a large change in the state vector. Moreover, Fisher scoring reduces the efficiency of the smoother even further. The smoother provides reasonable initial estimates, but estimation accuracy is substantially improved by using Newton's method to find the global Laplace approximation for the posterior.

2.1.2. *Estimating process noise.* For the applications to neural data examined here, we assume that $\boldsymbol{F} = \boldsymbol{I}$. However, we still need to estimate the process noise $\boldsymbol{Q}$. When n is small, especially when $n = 1$, different $\boldsymbol{Q}$ values will have a substantial influence on estimation. One possible way to estimate. One possible way to estimate $\boldsymbol{Q}$ is to use an Expectation Maximization (EM) algorithm as in Macke et al. (2011). However, using the Laplace approximation for $\boldsymbol{\Theta}$ during E-step breaks the usual guarantee of non-decreasing likelihoods in EM, and, hence, may lead to divergence. To avoid that, we could sample the posterior directly by MCMC. However, the lack of closed-form moments for the CMP distribution makes sampling computationally intensive. Here, to estimate $\boldsymbol{Q}$ robustly and quickly, we instead assume $\boldsymbol{Q}$ is diagonal and estimate it by maximizing the prediction likelihood in the filtering stage, as in Wei and Stevenson (2021).

2.1.3. *Neural Data.* I prefer to move this section to application part (reason: this is the data we use, but the not the method we propose. If we collect the data by ourself, surely we should write it here. But we are just using it ...)

3. Simulation. TBD

3.1. *Figure 1.* TBD

3.2. *Figure 2.* TBD

4. Application. TBD

4.1. *V1 data.* TBD

4.2. *HC data.* TBD

APPENDIX A: QUANTIFYING UNCERTAINTIES

After convergence, we have an approximation of the log-posterior $P(\boldsymbol{\theta}_t|\boldsymbol{Y}) \approx N(\boldsymbol{\theta}_t|\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$, and we can use this approximation to quantify the uncertainty about the CMP parameters, as well as about the mean rate at each time.

The CMP parameters are log-normal distributed. Let $\boldsymbol{Z}_{it} = \begin{pmatrix} \boldsymbol{x}_{it}' & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{g}_{it}' \end{pmatrix}$, then $(\lambda_{it}, \nu_{it})' = \exp(\boldsymbol{Z}_{it}\boldsymbol{\theta}_t) \sim \text{Lognormal}_2(\boldsymbol{Z}_{it}\boldsymbol{\mu}_t, \boldsymbol{Z}_{it}\boldsymbol{\Sigma}_t\boldsymbol{Z}_{it}')$. Denote the variance of CMP parameters as $\boldsymbol{V}_{it}$. The variance can be easily found in Wiki. If I write it out, I need to define some useless notations. See Word Documents.

The conditional mean firing rate is $\delta_{it} = E(Y_{it})$, whose variance can be calculated by the Delta method:

$$\widehat{Var}(\delta_{it}) = \begin{pmatrix} \frac{\partial \delta_{it}}{\partial \lambda_{it}} & \frac{\partial \delta_{it}}{\partial \nu_{it}} \end{pmatrix} \boldsymbol{V}_{it} \begin{pmatrix} \frac{\partial \delta_{it}}{\partial \lambda_{it}} \\ \frac{\partial \delta_{it}}{\partial \nu_{it}} \end{pmatrix} \tag{6}$$

$$\frac{\partial \delta_{it}}{\partial \lambda_{it}} = \frac{\partial^2 \log Z_{it}}{\partial \log \lambda_{it} \partial \lambda_{it}} = \frac{Var(Y_{it})}{\lambda_{it}} \tag{7}$$

$$\frac{\partial \delta_{it}}{\partial \nu_{it}} = \frac{\partial^2 \log Z_{it}}{\partial \log \lambda_{it} \partial \nu_{it}} = -Cov(Y_{it}, \log Y_{it}!) \tag{8}$$

We can calculate the moments as in Appendix B, or we can use simpler approximations $E(Y) = \lambda^{1/\nu} - \frac{\nu-1}{2\nu}$ when $\nu \leq 1$ or $\lambda > 10^\nu$. Then $\frac{\partial \delta_{it}}{\partial \lambda_{it}} \approx \frac{1}{\nu_{it}} \lambda_{it}^{1/\nu_{it}-1}$ and $\frac{\delta_{it}}{\nu_{it}} \approx -\frac{\lambda_{it}^{1/\nu_{it}} \log \lambda_{it}}{\nu_{it}^2} - \frac{1}{2\nu_{it}^2}$.

APPENDIX B: MOMENTS APPROXIMATION FOR CONWAY-MAXWELL POISSON DISTRIBUTION

To estimate the state-vector for the dynamic CMP model, we need to first and second moments for Y and $\log Y!$. For $Y \sim CMP(\lambda, \nu)$,

$$Z(\lambda, \nu) = \sum_{k=0}^{\infty} \frac{\lambda^k}{(k!)^\nu} \tag{9}$$

$$E(Y) = \frac{\partial \log Z}{\partial \log \lambda} = \frac{1}{Z} \sum_{k=0}^{\infty} \frac{k \lambda^k}{(k!)^\nu}$$

$$Var(Y) = \frac{\partial^2 \log Z}{\partial (\log \lambda)^2} = \frac{1}{Z} \sum_{k=0}^{\infty} \frac{k^2 \lambda^k}{(k!)^\nu} - E^2(Y)$$

$$E(\log Y!) = -\frac{\partial \log Z}{\partial \nu} = \frac{1}{Z} \sum_{k=0}^{\infty} \frac{(\log k!) \lambda^k}{(k!)^\nu}$$

$$Var(\log Y!) = \frac{\partial^2 \log Z}{\partial \nu^2} = \frac{1}{Z} \sum_{k=0}^{\infty} \frac{(\log k!)^2 \lambda^k}{(k!)^\nu} - E^2(\log Y!)$$

$$Cov(Y, \log Y!) = -\frac{\partial^2 \log Z}{\partial \log \lambda \partial \nu} = \frac{1}{Z} \sum_{k=0}^{\infty} \frac{(\log k!) k \lambda^k}{(k!)^\nu} - E(\log Y!) E(Y)$$

Generally, these moments can be calculated by truncated summation.

However, when $\lambda \geq 2$ and $\nu \leq 1$, we need many steps for accurate approximation. In this case, we make use of a previous asymptotic results for efficient calculation. Let $\alpha = \lambda^{1/\nu}, c_1 = \frac{\nu^2-1}{24}$ and $c_2 = \frac{\nu^2-1}{48} + \frac{c_1^2}{2}$,

$$Z(\lambda, \nu) = \frac{e^{\nu\alpha}}{\lambda^{\frac{\nu-1}{2\nu}} (2\pi)^{\frac{\nu-1}{2}} \sqrt{\nu}} (1 + c_1 (\nu\alpha)^{-1} + c_2 (\nu\alpha)^{-2} + \mathcal{O}(\lambda^{-3/\nu})) \tag{10}$$

Then the moments are:

$$E(Y) = \alpha - \frac{\nu-1}{2\nu} - \frac{\nu^2-1}{24\nu^2} \alpha^{-1} - \frac{\nu^2-1}{24\nu^3} \alpha^{-2} + \mathcal{O}(\alpha^{-3}) \tag{11}$$

$$
\begin{aligned}
Var(Y) &= \frac{\alpha}{\nu} + \frac{\nu^2-1}{24\nu^3}\alpha^{-1} + \frac{\nu^2-1}{12\nu^4}\aleph^{-2} + \mathcal{O}(\alpha^{-3}) \\
E(\log Y!) &= \alpha\left(\frac{\log\lambda}{\nu} - 1\right) + \frac{\log\lambda}{2\nu^2} + \frac{1}{2\nu} + \frac{\log 2\pi}{2} \\
&\quad - \frac{\alpha^{-1}}{24}\left(1 + \frac{1}{\nu^2} + \frac{\log\lambda}{\nu} - \frac{\log\lambda}{\nu^3}\right) \\
&\quad - \frac{\alpha^{-2}}{24}\left(\frac{1}{\nu^3} + \frac{\log\lambda}{\nu^2} - \frac{\log\lambda}{\nu^4}\right) + \mathcal{O}(\alpha^{-3}) \\
Var(\log Y!) &= \frac{\alpha(\log\lambda)^2}{\nu^3} + \frac{\log\lambda}{\nu^3} + \frac{1}{2\nu^2} \\
&\quad + \frac{\alpha^{-1}}{24\nu^5}[-2\nu^2 + 4\nu\log\lambda + (-1+\nu^2)(\log\lambda)^2] \\
&\quad + \frac{\alpha^{-2}}{24\nu^6}[-3\nu^2 - 2\nu(-3+\nu^2)\log\lambda + 2(-1+\nu^2)(\log\lambda)^2] + \mathcal{O}(\alpha^{-3}) \\
Cov(Y, \log Y!) &= \frac{\alpha\log\lambda}{\nu^2} + \frac{1}{2\nu^2} + \frac{\alpha^{-1}}{24}\left(\frac{2}{\nu^3} + \frac{\log\lambda}{\nu^2} - \frac{\log\lambda}{\nu^4}\right) \\
&\quad - \frac{1}{24\alpha^2}\left(\frac{1}{\nu^2} - \frac{3}{\nu^4} - \frac{2\log\lambda}{\nu^3} + \frac{2\log\lambda}{\nu^5}\right) + \mathcal{O}(\alpha^{-3})
\end{aligned}
$$

APPENDIX C: GRADIENT AND HESSIAN OF THE LOG-POSTERIOR

We estimate the state vector by maximizing the log-posterior with Newton-Raphson updates. Denote $f = P(\boldsymbol{\Theta}|\boldsymbol{Y})$, the $(k+1)$-th update of NR algorithm is $\boldsymbol{\Theta}^{(k+1)} = \boldsymbol{\Theta}^{(k)} + [\nabla\nabla_{\boldsymbol{\Theta}^{(k)}} f]^{-1}\nabla_{\boldsymbol{\Theta}^{(k)}} f$. The gradient is:

$$
\nabla_{\boldsymbol{\Theta}} f = \left[\left(\frac{\partial f}{\partial\boldsymbol{\theta}_1}\right)', \ldots, \left(\frac{\partial f}{\partial\boldsymbol{\theta}_T}\right)'\right]' \tag{12}
$$

$$
\begin{aligned}
\frac{\partial f}{\partial\boldsymbol{\theta}_1} &= \frac{\partial l_1}{\partial\boldsymbol{\theta}_1} - \boldsymbol{Q}_0^{-1}(\boldsymbol{\theta}_1 - \boldsymbol{\theta}_0) + \boldsymbol{F}'\boldsymbol{Q}^{-1}(\boldsymbol{\theta}_2 - \boldsymbol{F}\boldsymbol{\theta}_1) \\
\frac{\partial f}{\partial\boldsymbol{\theta}_t} &= \frac{\partial l_t}{\partial\boldsymbol{\theta}_t} - \boldsymbol{Q}^{-1}(\boldsymbol{\theta}_t - \boldsymbol{F}\boldsymbol{\theta}_{t-1}) + \boldsymbol{F}'\boldsymbol{Q}^{-1}(\boldsymbol{\theta}_{t+1} - \boldsymbol{F}\boldsymbol{\theta}_t) \\
\frac{\partial f}{\partial\boldsymbol{\theta}_T} &= \frac{\partial l_T}{\partial\boldsymbol{\theta}_T} - \boldsymbol{Q}^{-1}(\boldsymbol{\theta}_T - \boldsymbol{F}\boldsymbol{\theta}_{T-1}) \\
\frac{\partial l_t}{\partial\boldsymbol{\theta}_t} &= \sum_{i=1}^{n}\begin{pmatrix} (y_{it} - E(Y_{it}))\,\boldsymbol{x}_{it} \\ \nu_{it}\,(E(\log Y_{it}!) - \log y_{it}!)\,\boldsymbol{g}_{it} \end{pmatrix}
\end{aligned}
$$

The Hessian is:

$$
\nabla\nabla_{\boldsymbol{\Theta}} f = \begin{pmatrix}
\frac{\partial^2 f}{\partial\boldsymbol{\theta}_1\partial\boldsymbol{\theta}_1'} & \boldsymbol{F}'\boldsymbol{Q}^{-1} & \boldsymbol{0} & \cdots & \boldsymbol{0} \\
\boldsymbol{Q}^{-1}\boldsymbol{F} & \frac{\partial^2 f}{\partial\boldsymbol{\theta}_2\partial\boldsymbol{\theta}_2'} & \boldsymbol{F}'\boldsymbol{Q}^{-1} & \cdots & \vdots \\
\boldsymbol{0} & \boldsymbol{Q}^{-1}\boldsymbol{F} & \frac{\partial^2 f}{\partial\boldsymbol{\theta}_3\partial\boldsymbol{\theta}_3'} & \cdots & \vdots \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
\boldsymbol{0} & \cdots & \cdots & \cdots & \frac{\partial^2 f}{\partial\boldsymbol{\theta}_T\partial\boldsymbol{\theta}_T'}
\end{pmatrix} \tag{13}
$$

$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_1 \partial \boldsymbol{\theta}_1'} = \frac{\partial^2 l_1}{\partial \boldsymbol{\theta}_1 \partial \boldsymbol{\theta}_1'} - \boldsymbol{Q}_0^{-1} - \boldsymbol{F}'\boldsymbol{Q}^{-1}\boldsymbol{F}$$
$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_t'} = \frac{\partial^2 l_t}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_t'} - \boldsymbol{Q}^{-1} - \boldsymbol{F}'\boldsymbol{Q}^{-1}\boldsymbol{F}$$
$$\frac{\partial^2 f}{\partial \boldsymbol{\theta}_T \partial \boldsymbol{\theta}_T'} = \frac{\partial^2 l_T}{\partial \boldsymbol{\theta}_T \partial \boldsymbol{\theta}_T'} - \boldsymbol{Q}^{-1}$$

, where

$$\frac{\partial^2 l_t}{\partial \boldsymbol{\theta}_t \partial \boldsymbol{\theta}_t'} = \sum_{i=1}^{n} \begin{pmatrix} \boldsymbol{A}_{it} & \boldsymbol{B}_{it} \\ \boldsymbol{B}_{it}' & \boldsymbol{C}_{it} \end{pmatrix} \tag{14}$$
$$\boldsymbol{A}_{it} = Var(Y_{it})\boldsymbol{x}_{it}\boldsymbol{x}_{it}'$$
$$\boldsymbol{B}_{it} = -\nu_{it}Cov(Y_{it}, \log Y_{it}!)\boldsymbol{x}_{it}\boldsymbol{g}_{it}'$$
$$\boldsymbol{C}_{it} = \nu_{it}[\nu_{it}Var(\log Y_{it}) - E(\log Y_{it}!) + \log y_{it}!]\boldsymbol{g}_{it}\boldsymbol{g}_{it}'$$

When $\log y_{it}! \ll E(\log Y_{it}!)$, the Hessian may be ill-conditioned or even positive-definite. To ensure the robustness, do Fisher scoring, i.e. replace the observed information $-\nabla\nabla_{\boldsymbol{\Theta}} f$ by the expected information $E(-\nabla\nabla_{\boldsymbol{\Theta}} f)$, so that $\boldsymbol{C}_{it} = \nu_{it}^2 Var(\log Y_{it}!)\boldsymbol{g}_{it}\boldsymbol{g}_{it}'$

Acknowledgments. The authors would like to thank the anonymous referees, an Associate Editor and the Editor for their constructive comments that improved the quality of this paper.

Funding. This material is based upon work supported by the National Science Foundation under Grant No. 1931249.

The first author was supported by NSF Grant DMS-??-??????.

The second author was supported in part by NIH Grant ???????????.

SUPPLEMENTARY MATERIAL

Title of Supplement A
Short description of Supplement A.

REFERENCES

CHATLA, S. B. and SHMUELI, G. (2018). Efficient estimation of COM–Poisson regression and a generalized additive model. *Computational Statistics & Data Analysis* **121** 71–88.

EDEN, U. T., FRANK, L. M., BARBIERI, R., SOLO, V. and BROWN, E. N. (2004). Dynamic Analysis of Neural Encoding by Point Process Adaptive Filtering. *Neural Computation* **16** 971–998.

GAUNT, R. E., IYENGAR, S., OLDE DAALHUIS, A. B., SIMSEK, B. and ROBERT GAUNT, B. E. (2019). An asymptotic expansion for the normalizing constant of the Conway-Maxwell-Poisson distribution. *Ann Inst Stat Math* **71** 163–180.

GUPTA, R. C., SIM, S. Z. and ONG, S. H. (2014). Analysis of discrete data by Conway–Maxwell Poisson distribution. *AStA Advances in Statistical Analysis* **98** 327–343.

MACKE, J. H., BUESING, L., CUNNINGHAM, J. P., YU, B. M., SHENOY, K. V. and SAHANI, M. (2011). Empirical models of spiking in neural populations. *Advances in Neural Information Processing Systems* **24**.

PANINSKI, L., AHMADIAN, Y., FERREIRA, D. G., KOYAMA, S., RAHNAMA RAD, K., VIDNE, M., VOGELSTEIN, J. and WU, W. (2010). A New Look at State-Space Models for Neural Data. *J. Comput. Neurosci.* **29** 107–126.

RAUCH, H. E., TUNG, F. and STRIEBEL, C. T. (1965). Maximum likelihood estimates of linear dynamic systems. *AIAA Journal* **3** 1445–1450.

SELLERS, K. F. and SHMUELI, G. (2010). A flexible regression model for count data. *The Annals of Applied Statistics* **4** 943 – 961.

SHMUELI, G., MINKA, T. P., KADANE, J. B., BORLE, S. and BOATWRIGHT, P. (2005). A useful distribution for fitting discrete data: revival of the Conway–Maxwell–Poisson distribution. *Journal of the Royal Statistical Society: Series C (Applied Statistics)* **54** 127–142.

WEI, G. and STEVENSON, I. H. (2021). Tracking Fast and Slow Changes in Synaptic Weights From Simultaneously Observed Pre- and Postsynaptic Spiking. *Neural computation* **33** 2682–2709.