Gaussian process based nonlinear latent structure discovery in multivariate spike train data

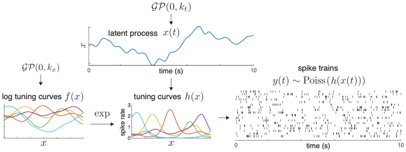
Adam Peterson

UM Dept. Biostatistics

November 26th, 2018

Motivation

- Data: Multi-Electrode Array Observations
- Goal:
    - Estimate Population Process
    - Estimate individual "tuning curves"



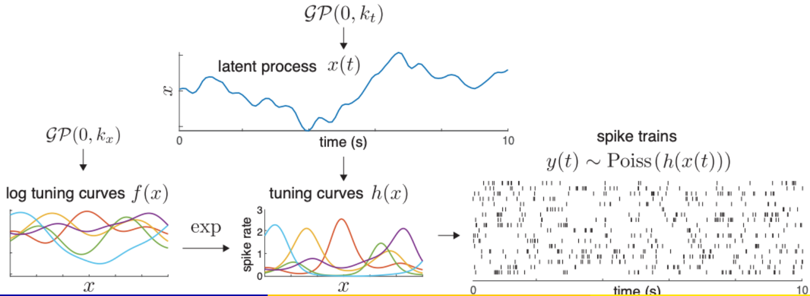
Model and Notation

$$\boldsymbol{Y} \in \mathbb{R}^{N \times T} \qquad i = 1, ..., N; t = 1, ..., T$$

$$\boldsymbol{x}(t) \sim \mathcal{GP}(0, K_t) \qquad \boldsymbol{X} \in \mathbb{R}^{P \times T}$$

$$\boldsymbol{f}_i(\boldsymbol{x}) = \log h(\boldsymbol{x}(t)); \quad \boldsymbol{f}_i(\boldsymbol{x}) \sim \mathcal{GP}(0, k_x)$$

$$y_{i,t} \mid \boldsymbol{f}_i, \boldsymbol{x}_t \overset{iid}{\sim} \text{Poisson}(\exp(\boldsymbol{f}_i(\boldsymbol{x}_t)))$$



Gaussian Process - Reminder

$$\boldsymbol{f}_i \sim \mathcal{GP}(0, k_x) \Rightarrow \boldsymbol{f}_i \overset{iid}{\sim} \mathcal{MVN}(\boldsymbol{0}, \boldsymbol{K}_x)$$

- $\boldsymbol{K}_x$ a covariance matrix that is a result of some covariance function k_x
  - e.g. $k_x(\boldsymbol{x}, \boldsymbol{x}') = \rho e^{-\frac{|\boldsymbol{x}-\boldsymbol{x}'|^2}{2\delta}}$

Inference

$$p(\boldsymbol{Y}, \boldsymbol{F}, \boldsymbol{X}, \theta) = p(\boldsymbol{Y}|\boldsymbol{F})p(\boldsymbol{F}|\boldsymbol{X}, \rho, \delta)p(\boldsymbol{X} \mid r, l)$$
$$= \prod_{i=1}^{N}\prod_{t=1}^{T} p(y_{i,t}|f_{i,t}) \prod_{i=1}^{N} p(\boldsymbol{f}_i \mid \boldsymbol{X}, \rho, \delta) \prod_{j=1}^{P} p(\boldsymbol{x}_j \mid r, l)$$

but,

$$\log p(\boldsymbol{Y}) = \log \int\int p(\boldsymbol{Y}, \boldsymbol{F}, \boldsymbol{X}) d\boldsymbol{X} d\boldsymbol{F}$$
$$= \log \int p(\boldsymbol{Y}|\boldsymbol{F}) \int p(\boldsymbol{F} \mid \boldsymbol{X}) p(\boldsymbol{X}) d\boldsymbol{X} d\boldsymbol{F}$$

nested integral is intractable

Goal and Gameplan

Goal

$$F_{MAP}X_{MAP} = \arg\max_{F,X} p(Y, F)p(F|X)p(X)$$

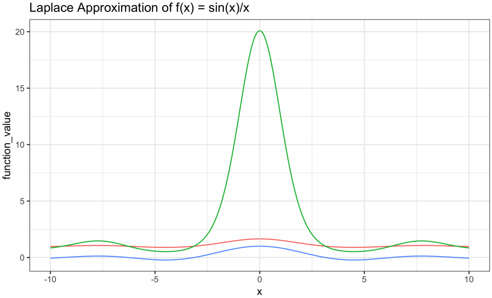
1. Use Laplace's Approximation (LA) to find an approximation to $p(F|Y, X)$
   - We'll talk about what LA is and why it works
2. Use the approximation of to find X_{MAP}
   - Gradient obstacles $\rightarrow$ Third Derivative LA (tLA)
   - tLA computational issues $\rightarrow$ decoupled LA (dLA)
3. dLA algorithm
4. Simulations
5. Real Data

Laplace's Approximation

- Bayesian Central Limit Theorem
- Approximate a function through a Gaussian kernel

$$p(\theta|\boldsymbol{y}) \xrightarrow{d} \mathcal{N}(\hat{\theta}, I(\hat{\theta})^{-1})$$

$$p(\boldsymbol{f}_i|\boldsymbol{y}_i, \boldsymbol{X}) \approx q(\boldsymbol{f}_i|\boldsymbol{y}_i, \boldsymbol{X}) = \mathcal{N}(\hat{\boldsymbol{f}}_i, \boldsymbol{A}^{-1})$$



$$\hat{\boldsymbol{f}}_i = \arg\max_{\hat{\boldsymbol{f}}_i} p(\boldsymbol{f}_i|\boldsymbol{y}_i, \boldsymbol{X})$$

$$\boldsymbol{A} = -\nabla\nabla \log p(f_i|\boldsymbol{y}_i, \boldsymbol{X})_{\boldsymbol{f}_i = \hat{\boldsymbol{f}}_i}$$

Laplace Approximation: Tuning Curves

$$
\begin{aligned}
p(\boldsymbol{f}_i|\boldsymbol{y}_i, \boldsymbol{X}) &= \frac{p(\boldsymbol{y}_i|\boldsymbol{f}_i)p(\boldsymbol{f}_i|\boldsymbol{X})}{p(\boldsymbol{y}_i \mid \boldsymbol{X})} \\
&\propto p(\boldsymbol{y}_i|\boldsymbol{f}_i)p(\boldsymbol{f}_i|\boldsymbol{X}) \\
\Psi(\boldsymbol{f}_i) := \log p(\boldsymbol{f}_i \mid \boldsymbol{y}_i, \boldsymbol{X}) &= \log p(\boldsymbol{y}_i|\boldsymbol{f}_i) + \log p(\boldsymbol{f}_i|\boldsymbol{X}) \\
&= \log p(\boldsymbol{y}_i|\boldsymbol{f}_i) - \frac{1}{2}\boldsymbol{f}_i^T \boldsymbol{K}_x^{-1}\boldsymbol{f}_i - \frac{1}{2}\log \mid \boldsymbol{K}_x \mid
\end{aligned}
$$

(Eqn 10)

F MAP Estimate

$$
\boldsymbol{F}_{MAP} = \arg\max_{\boldsymbol{f_i}} \Psi(\boldsymbol{f}_i) \quad i = 1, ..., N
$$

LA of Marginal Likelihood

$$
\begin{aligned}
p(\boldsymbol{y}_i|\boldsymbol{X}) &= \int p(\boldsymbol{y}_i \mid \boldsymbol{f}_i)p(\boldsymbol{f}_i \mid \boldsymbol{X})d\boldsymbol{f} \\
&\approx q(\boldsymbol{y}_i|\boldsymbol{X}) \\
q(\boldsymbol{y}_i|\boldsymbol{X}) &= \exp(\Psi(\hat{\boldsymbol{f}}_i)) \int \exp\Big\{-\frac{1}{2}(\boldsymbol{f}_i - \hat{\boldsymbol{f}})^T \boldsymbol{A}(\boldsymbol{f}_i - \hat{\boldsymbol{f}})\Big\} d\boldsymbol{f}_i \\
&\qquad\qquad\qquad\qquad \text{(taylor expansion around } \hat{\boldsymbol{f}}\text{)} \\
\Rightarrow \log q(\boldsymbol{y}_i|\boldsymbol{X}) &= \log p(\boldsymbol{y}_i|\hat{\boldsymbol{f}}_i) - \frac{1}{2}\hat{\boldsymbol{f}}_i{}^T \boldsymbol{K}_x^{-1}\hat{\boldsymbol{f}}_i - \frac{1}{2}\log \mid \boldsymbol{I}_T + \boldsymbol{K}_x \boldsymbol{W}_i \mid \\
\boldsymbol{W}_i &:= \nabla\nabla p(\boldsymbol{y}_i \mid \boldsymbol{f}_i)
\end{aligned}
$$

X MAP Estimate

$$
X_{MAP} = \arg\max_{\boldsymbol{X}} \sum_{i=1}^{N} q(\boldsymbol{y}_i|\boldsymbol{X})p(\boldsymbol{X})
$$

Gradient Problems

$$\log q(\boldsymbol{y}_i|\boldsymbol{X}) = \log p(\boldsymbol{y}_i|\hat{\boldsymbol{f}}_i) - \frac{1}{2}\hat{\boldsymbol{f}}_i^{\ T}\boldsymbol{K}_x^{-1}\hat{\boldsymbol{f}}_i - \frac{1}{2}\log|\ \boldsymbol{I}_T + \boldsymbol{K}_x\boldsymbol{W}_i\ | \quad \text{(Eqn 13)}$$

- To maximize, we take the derivative
- But $\hat{\boldsymbol{f}}_i$ $\boldsymbol{K}_x$ are implicit functions of $\boldsymbol{X}$!
  - for different $\boldsymbol{X}$, you get a new mode $\hat{\boldsymbol{f}}_i$
  - No straightforward calculation of full gradient via standard methods
- Solution:
  1. Use chain rule to calculate gradients explicitly - slow
  2. Decouple dependency of $\boldsymbol{f}, \boldsymbol{X}$ into gaussian product

Third Derivative LA

See paper

Decoupled LA

Recall

$$p(\boldsymbol{f}_i|\boldsymbol{y}_i, \boldsymbol{X}) \propto p(\boldsymbol{y}_i|\boldsymbol{f}_i)p(\boldsymbol{f}_i \mid \boldsymbol{X})$$

Approximate $p(\boldsymbol{y}_i|\boldsymbol{f}_i)$ by $q(\boldsymbol{y}_i|\boldsymbol{f}_i) \sim N(\boldsymbol{m}, S)$

$$\Rightarrow \mathcal{N}(\hat{\boldsymbol{f}}_i, \boldsymbol{A}^{-1}) \propto \mathcal{N}(\boldsymbol{m}, \boldsymbol{S})\mathcal{N}(\boldsymbol{0}, \boldsymbol{K}_x)$$

Can re-derive $\boldsymbol{m}, S$

$$\begin{aligned} \boldsymbol{A} &= \boldsymbol{S}^{-1} + \boldsymbol{K}_x^{-1} \quad \hat{\boldsymbol{f}}_i = \boldsymbol{A}^{-1}\boldsymbol{S}^{-1}\boldsymbol{m} \\ \boldsymbol{S} &= (\boldsymbol{A} - \boldsymbol{K}_x^{-1})^{-1} \quad \boldsymbol{m} = \boldsymbol{S}\boldsymbol{A}\hat{\boldsymbol{f}}_i \end{aligned} \qquad \text{(Eqn 18)}$$

Decoupled LA - utility

Now we don't need to worry about $q(\boldsymbol{y}_i \mid \boldsymbol{X})$'s implicit functions of X

Algorithm 1 Decoupled Laplace approximation at iteration k

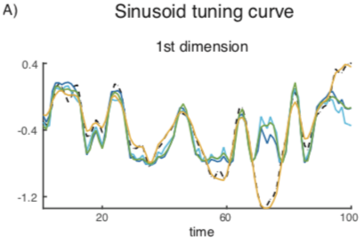
Input: data observation $\mathbf{y}_i$, latent variable $\mathbf{X}^{k-1}$ from iteration $k-1$

1. Compute the new posterior mode $\hat{\mathbf{f}}_i^k$ and the precision matrix A^k by solving (Eq. 10) to obtain $q(\mathbf{f}_i|\mathbf{y}_i, \mathbf{X}^{k-1}) = \mathcal{N}(\hat{\mathbf{f}}_i^k, A^{k^{-1}})$.
2. Derive $\mathbf{m}^k$ and S^k (Eq. 18): $S^k = (A^k - K_x^{-1})^{-1}$, $\mathbf{m}^k = S^k A^k \hat{\mathbf{f}}_i^k$.
3. Fix $\mathbf{m}^k$ and S^k and derive the new mean and covariance for $q(\mathbf{f}_i|\mathbf{y}_i, \mathbf{X}^{k-1})$ as functions of $\mathbf{X}$: $A(\mathbf{X}) = S^{k^{-1}} + K_x(\mathbf{X})^{-1}$, $\hat{\mathbf{f}}_i(\mathbf{X}) = A(\mathbf{X})^{-1} S^{k^{-1}} \mathbf{m}^k = A(\mathbf{X})^{-1} A^k \hat{\mathbf{f}}_i^k$.
4. Since $A = W_i + K_x^{-1}$, we have $W_i = S^{k^{-1}}$, and can obtain the new approximated conditional distribution $q(\mathbf{y}_i|\mathbf{X})$ (Eq. 13) with $\hat{\mathbf{f}}_i$ replaced by $\hat{\mathbf{f}}_i(\mathbf{X})$.
5. Solve $\mathbf{X}^k = \text{argmax}_{\mathbf{X}} \sum_{i=1}^N q(\mathbf{y}_i|\mathbf{X}) p(\mathbf{X})$.

Output: new latent variable $\mathbf{X}^k$

- $\boldsymbol{f}_i | \boldsymbol{y}_i \to \boldsymbol{y}_i \mid \boldsymbol{f}_i \to \boldsymbol{f}_i^*{}_{MAP} \mid \boldsymbol{y}_i^* \to \boldsymbol{y}_i | \boldsymbol{X} \to \boldsymbol{X}_{MAP}$

Simulations



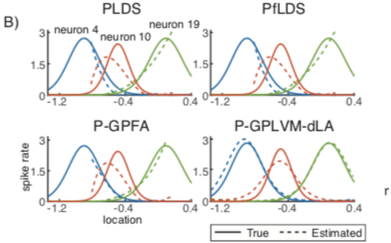
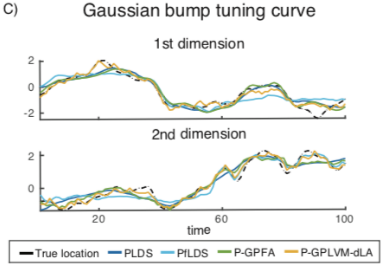


Figure A is the latent process, B the tuning curve

In all simulation studies, we generate 1 single trial per neuron with 20 simulated neurons and 100 time bins for a single experiment. Each experiment is repeated 10 times and results are averaged across 10 repeats.

Simulations II



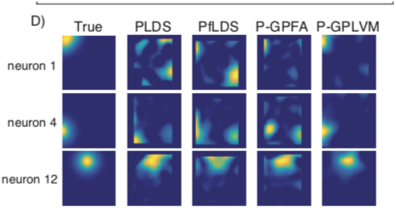
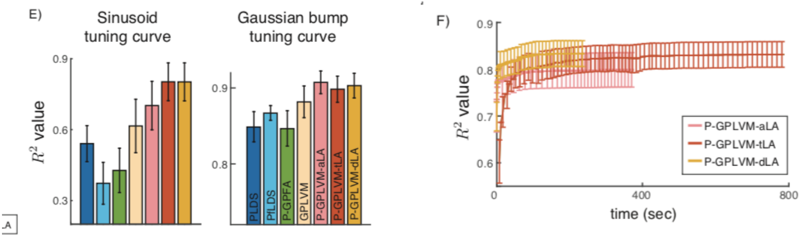


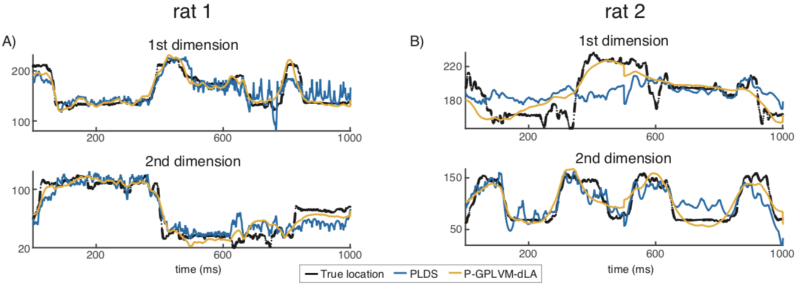
Figure C is the latent process, D the tuning curve

In all simulation studies, we generate 1 single trial per neuron with 20 simulated neurons and 100 time bins for a single experiment. Each experiment is repeated 10 times and results are averaged across 10 repeats.

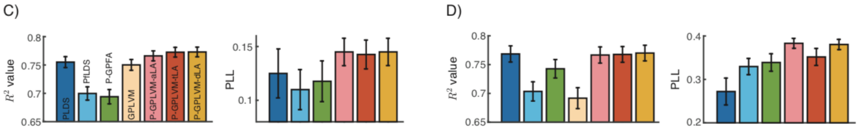
Simulation Results



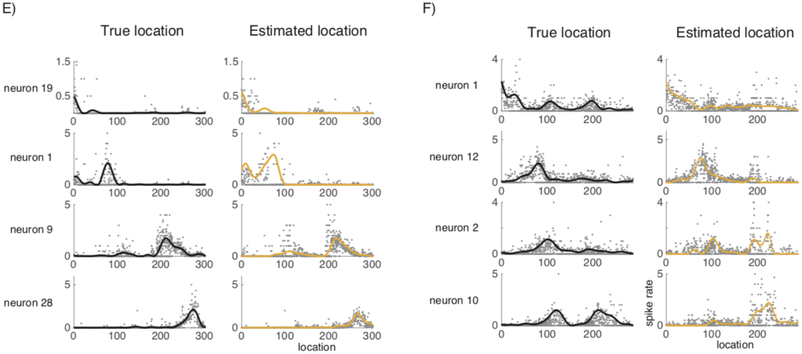
Application: Rat Data I



Application: Rat Data II



Application: Rat Data III



The first column contains the binned spike counts when mapping from the space of x_{true} to the space of $x_{1:G}$. The second column contains the binned spike counts mapped from the space of $x_{P-GPLVM}$ to the space of $x_{1:G}$. The black curves in the first column are achieved by replacing $\hat{x}$ and $\hat{f}$ with x_{true} and y.