Fast methods for nonparametric estimation of encoding models

Goal: estimate dependence of firing rate on kinematic parameters.

Avoid parametric assumptions: use nonparametric approach.

Avoid oversmoothing: use a penalizer that allows for sharp changes in firing rate

Challenge: computation.

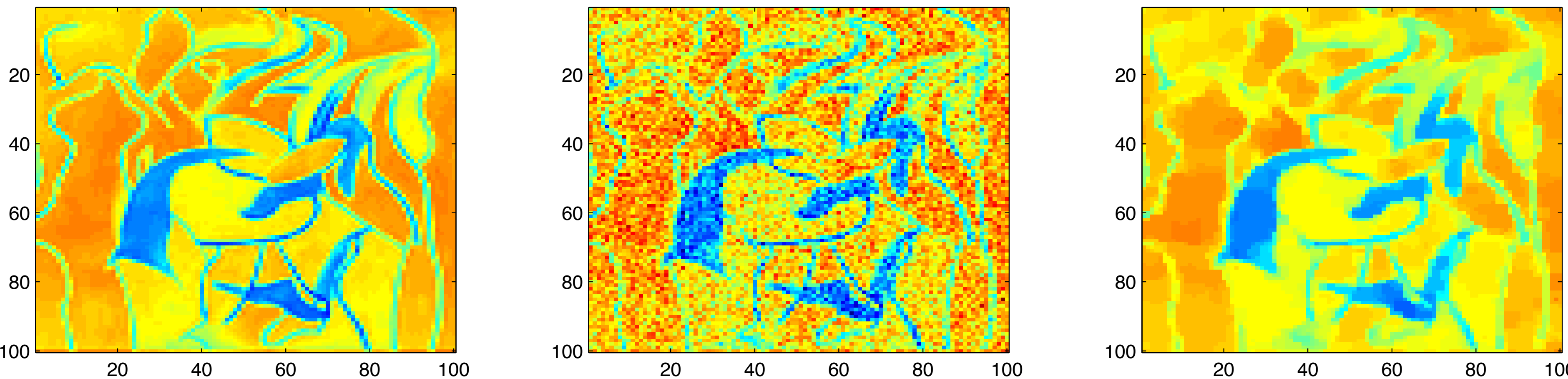
Optimize L(z) + Q(z):

z = firing rate surface

L(z) = data loglikelihood

Q(z) = penalizer (Huber total variation norm)

Solution: use splitting methods to reduce problem to a sequence of HMM-like smoothing problems. Leads to linear-time methods.



Rahnama Rad and Paninski, in progress

Exploiting expected loglikelihoods

$$\begin{aligned} L(\theta) &= \sum_{n=1}^{N} \Big((x_n^T \theta) r_n - G(x_n^T \theta) \Big) + \mathrm{const}(\theta) \\ &\approx \left(\sum_{n=1}^{N} x_n^T r_n \right) \theta - N \mathbf{E}\Big[G(x^T \theta) \Big] \equiv \tilde{L}(\theta) \end{aligned}$$

r: responses; x: kinematic variables; θ: parameter to be estimated

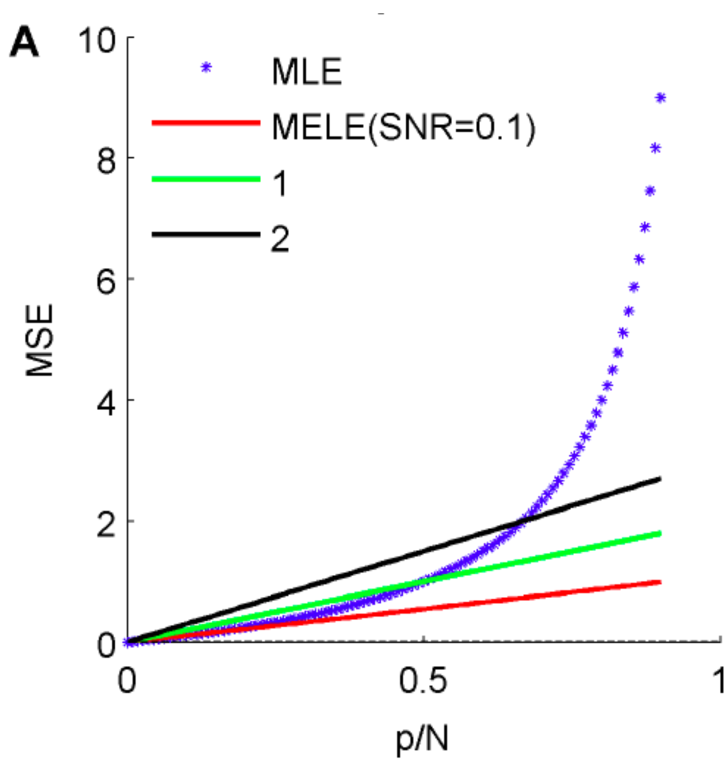
EL can be computed and optimized an order of magnitude faster than LL.

In many cases EL can be optimized analytically; MLE must be computed numerically

In many cases EL-based estimates are more accurate than MLE

Other applications: fast model selection, MCMC sampling

For full details, see Ramirez and Paninski (2012).



Fast high-dimensional state space methods

Standard smoothing methods scale like $O(d^3)$ per timestep (or worse).

New method: $O(d)$. Allows for much richer nonstationary models than previously possible. Main idea: low-rank approximation of posterior state covariance.

Can handle non-smooth priors, likelihoods.

Pnevmatikakis et al (2012): proved convergence, rigorous error bounds.

Exact inference methods in nonstandard state spaces

How do we perform exact inference for time series on manifolds, or more general state spaces (e.g., space of all reachable joint configurations)? Standard methods assume vector state spaces.

Main result: exact inference for priors of form:

$$p(X) \propto \prod_{t=1}^{T-1} \sum_{z_t=1}^{R_t} f_{t,z_t}(x_t) g_{t,z_t}(x_{t+1})$$

See Smith et al (2012) for full details.

Convex methods for state-space identification

Poisson Linear Dynamical System

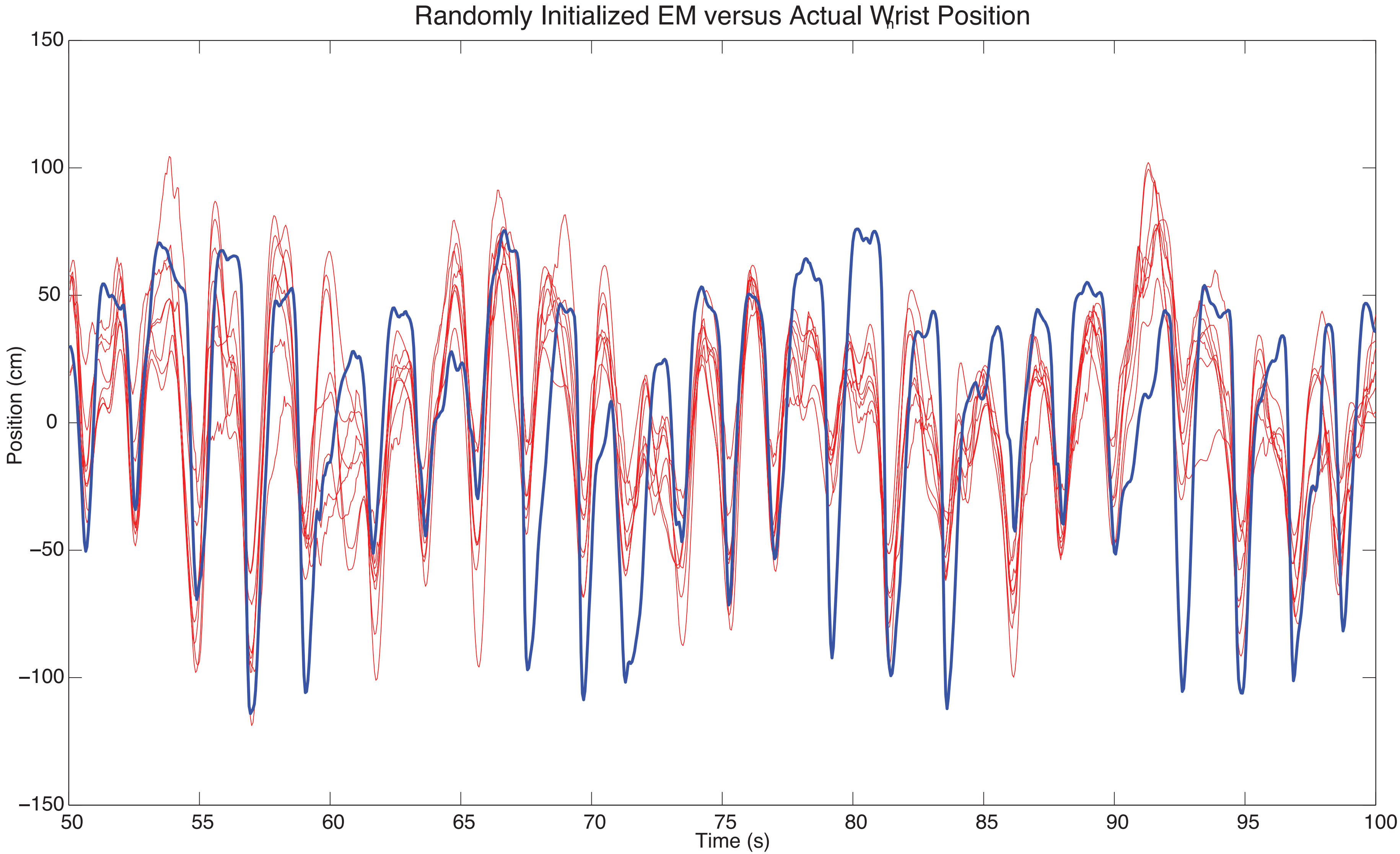
Latent linear dynamical system:

$$\begin{aligned}\vec{x}_t &= A\vec{x}_{t-1} + \vec{\epsilon}_t \\ \vec{\epsilon}_t &\sim \mathcal{N}(0, Q) \\ \vec{x}_t &\in \mathbb{R}^m\end{aligned}$$

Linear-nonlinear-Poisson output:

$$\begin{aligned}\vec{y}_t &= C\vec{x}_t \\ s_{it}|y_{it} &\sim Poiss(exp(y_{it})) \\ \vec{s}_t, \vec{y}_t &\in \mathbb{R}^n\end{aligned}$$

Our approach: estimate $\vec{y}_t$ by *nuclear norm minimization*, recover A and C by *subspace identification*.



Randomly Initialized EM versus Actual Wrist Position
150
100
50
0
−50
−100
−150
Position (cm)
50
55
60
65
70
75
80
85
90
95
100
Time (s)

Noiseless Data

Assume we know $\vec{y}_t$ already, and there is no process noise.

$$\begin{aligned} \vec{y}_t &= C\vec{x}_t \\ \vec{y}_{t+k} &= CA^k\vec{x}_t \end{aligned}$$

$$\begin{pmatrix} \vec{y}_t \\ \vec{y}_{t+1} \\ \vdots \\ \vec{y}_{t+k} \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^k \end{pmatrix} \vec{x}_t$$

$$\begin{pmatrix} \vec{y}_1 & \vec{y}_2 & \cdots & \vec{y}_{T-k+1} \\ \vec{y}_2 & \vec{y}_3 & \cdots & \vec{y}_{T-k+2} \\ \vdots & \vdots & & \vdots \\ \vec{y}_k & \vec{y}_{k+1} & \cdots & \vec{y}_T \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{pmatrix} \begin{pmatrix} \vec{x}_1 \ldots & \vec{x}_{T-k+1} \end{pmatrix}$$

Noiseless Data

$$\begin{pmatrix} \vec{y}_1 & \vec{y}_2 & \cdots & \vec{y}_{T-k+1} \\ \vec{y}_2 & \vec{y}_3 & \cdots & \vec{y}_{T-k+2} \\ \vdots & \vdots & & \vdots \\ \vec{y}_k & \vec{y}_{k+1} & \cdots & \vec{y}_T \end{pmatrix} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{pmatrix} \begin{pmatrix} \vec{x}_1 \ldots & \vec{x}_{T-k+1} \end{pmatrix}$$

Large block-Hankel matrix (nkxT) but only rank m.

$$Y = U\Sigma V^T$$

$$U\Sigma^{1/2} = \begin{pmatrix} C \\ CA \\ \vdots \\ CA^{k-1} \end{pmatrix}$$

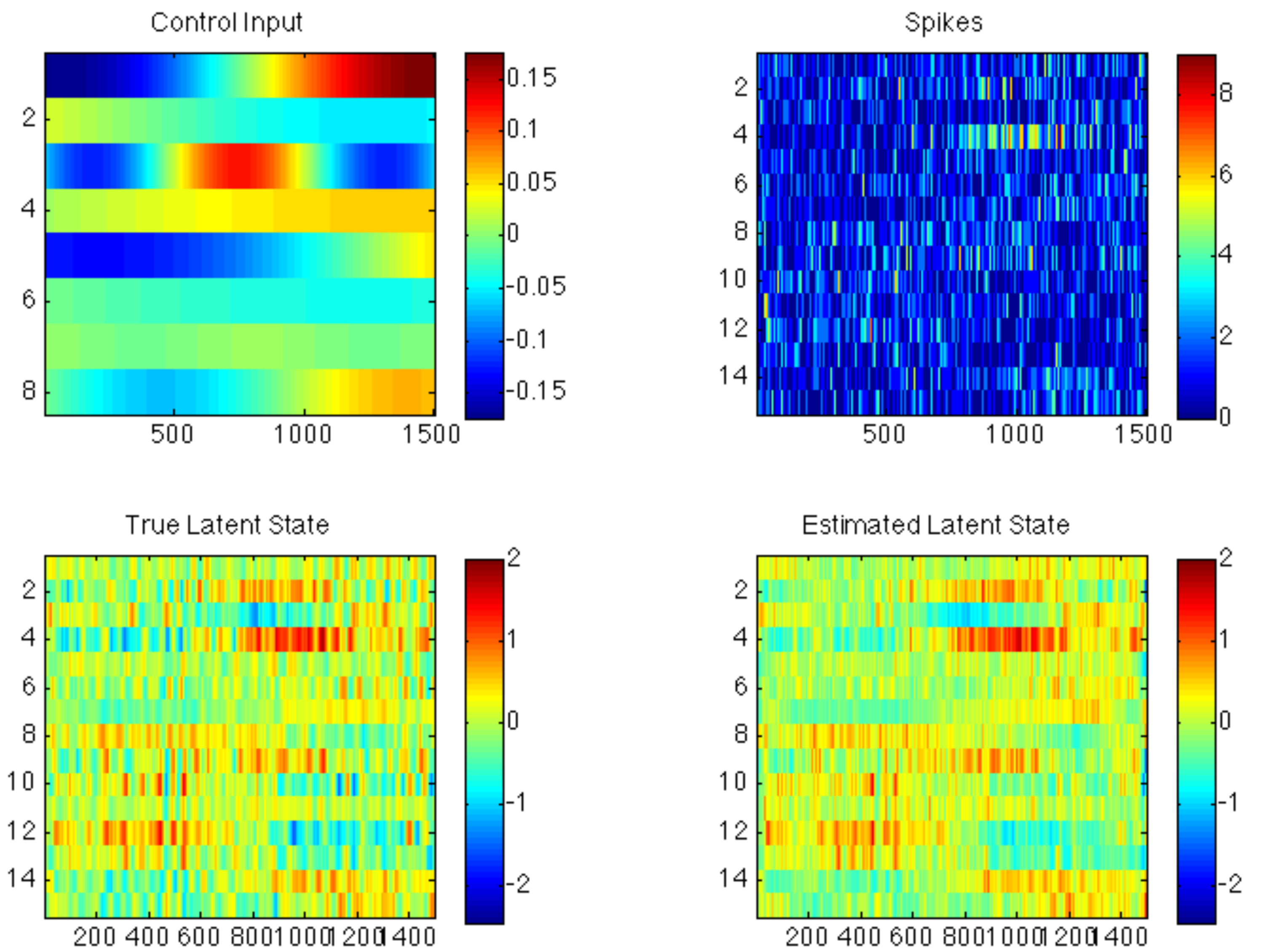
Can reconstruct C and A up to an irrelevant rotation.

Recovering $\vec{y}_t$

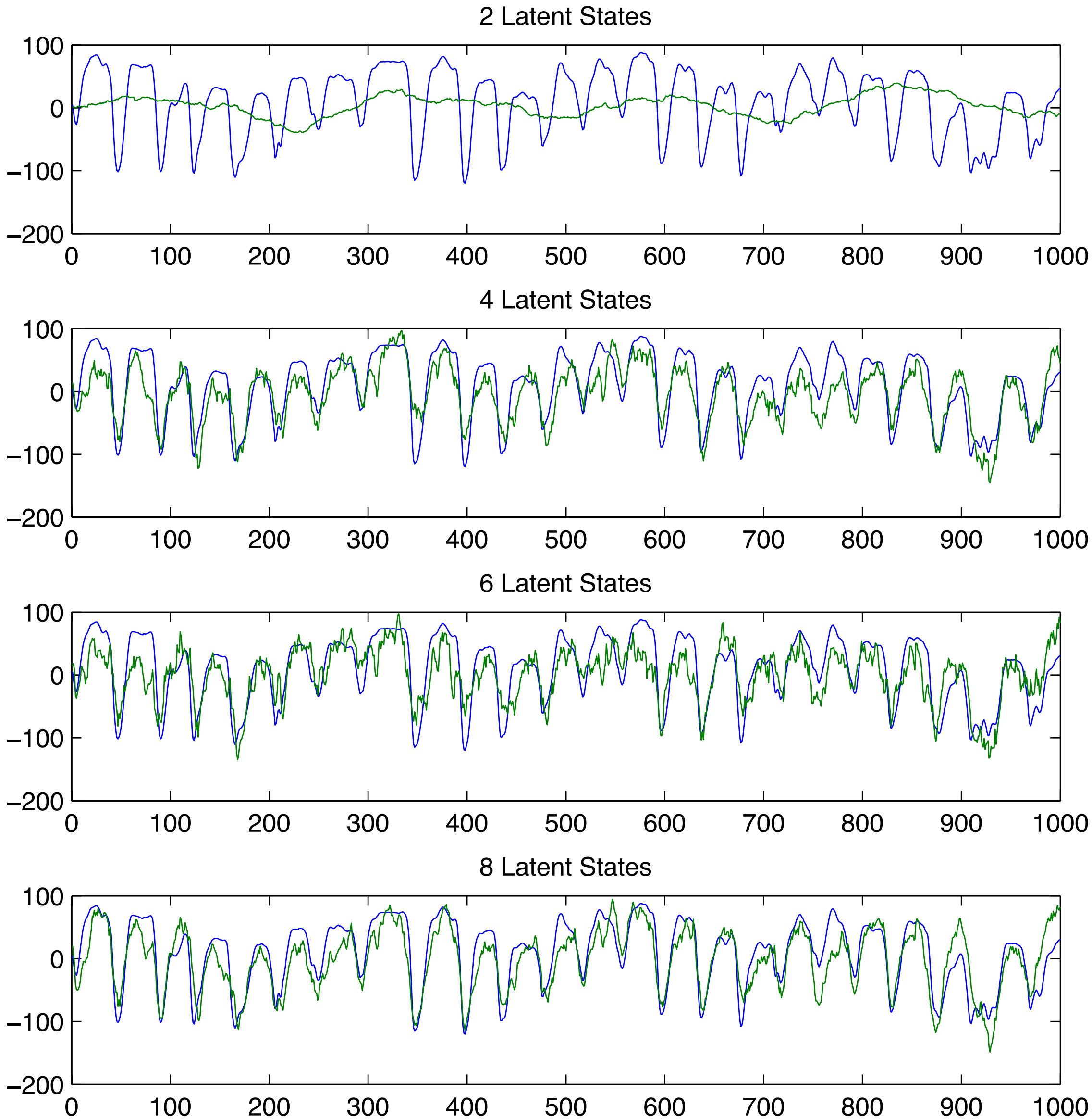
Idea: find estimate $\hat{y}_t$ that looks like it was generated by noiseless system.

$$\min_{\hat{y}} \operatorname{rank}(\hat{Y}) + \lambda \sum_{it} \log \mathrm{p}(s_{it} | \hat{y}_{it})$$

- λ controls amount of regularization.
- Computationally intractable! Try a convex relaxation, the nuclear norm (Fazel et al. [2001], Liu and Vandenberghe [2009])



Simulated example; Pfau, Pnevmatikakis, Paninski, in progress


2 Latent States
4 Latent States
6 Latent States
8 Latent States
100
0
−100
−200
0
100
200
300
400
500
600
700
800
900
1000

Sequential MCMC methods for robust particle filtering

Recall basic recursion:

$$p(q_t, Y_{1:t}) = p(y_t|q_t) \int_{q_{t-1}} p(q_t|q_{t-1}) p(q_{t-1}, Y_{1:t-1}) dq_{t-1}.$$

Particle filter: importance sampling to approximate integral.

Can be highly effective (Doucet et al., 2001).

However, importance sampling is known to be very non-robust in many important cases: if we put the particles in the wrong part of the space, the variance of the importance weights becomes too large and the filter fails.

Robust particle filtering via sequential MCMC

Rewrite basic recursion:

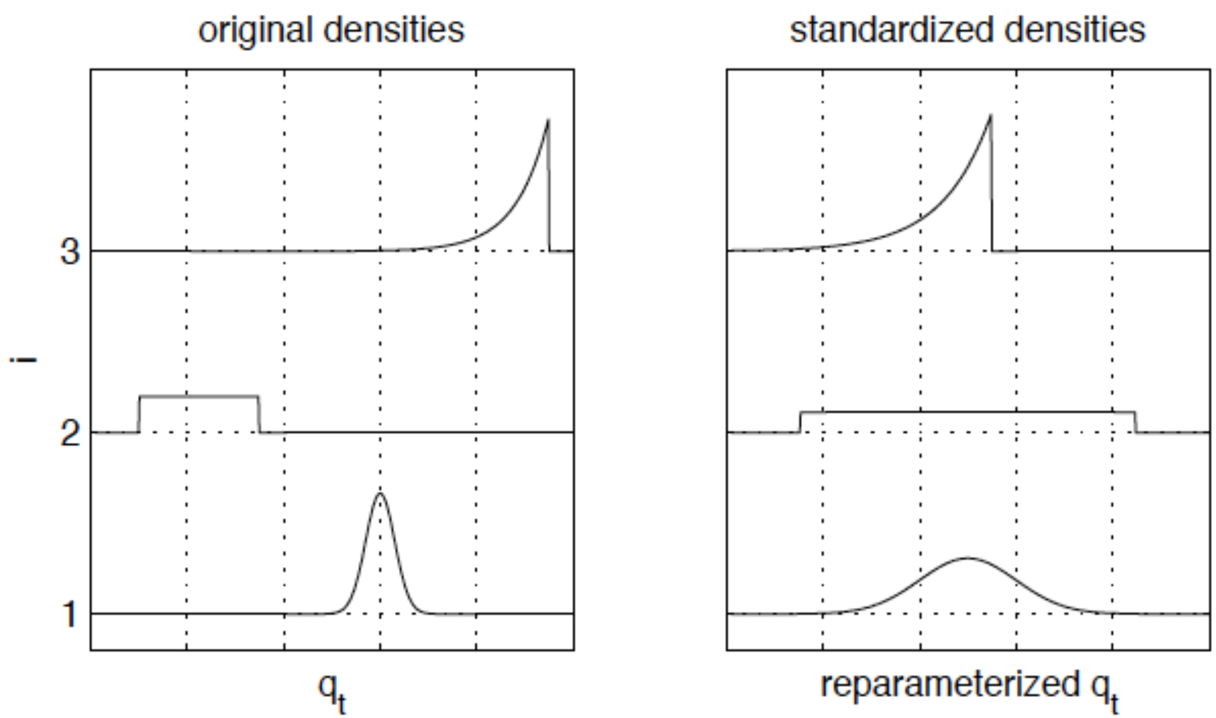
$$p(q_t, Y_{1:t}) = \int_{q_{t-1}} p(y_t|q_t)p(q_t|q_{t-1})p(q_{t-1}, Y_{1:t-1})dq_{t-1}. \quad (1)$$

$$= \int_{q_{t-1}} p(q_t, q_{t-1}, Y_{1:t})dq_{t-1}. \quad (2)$$

Basic idea: use MCMC to sample directly from $p(q_t, q_{t-1}|Y_{1:t})$ — bypass nonrobust particle filter step entirely.

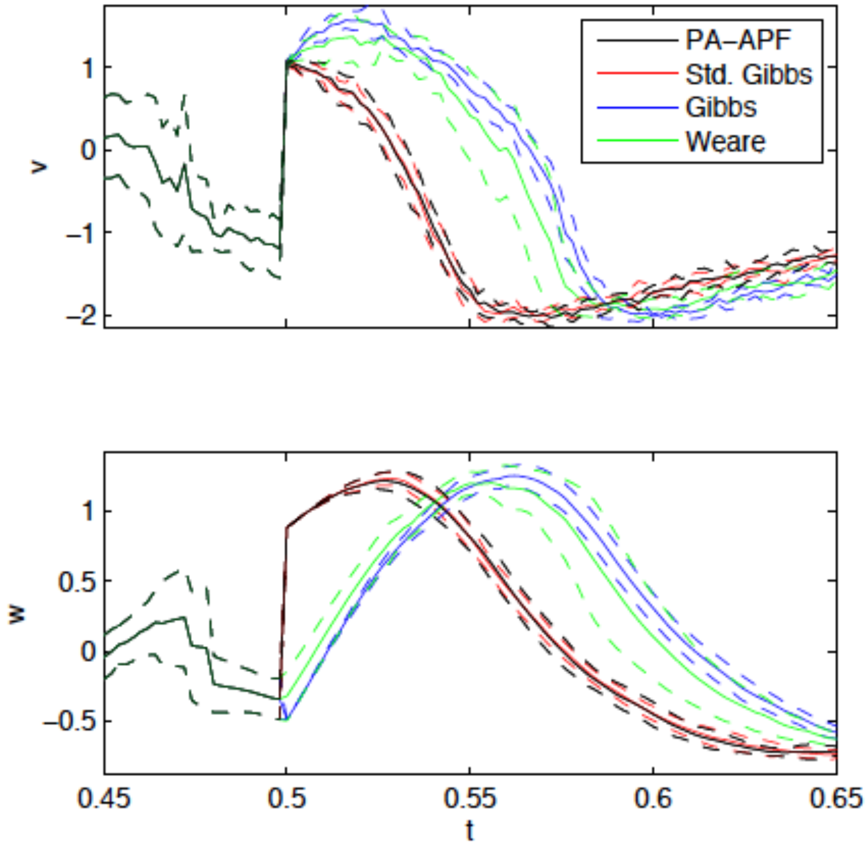
Reparameterized Gibbs sampling

Basic Gibbs sampling on $p(q_t, q_{t-1}|Y_{1:t})$ often doesn't work: mixing is too slow, because $p(q_t|q_{t-1} = q^i_{t-1}, Y_{1:t})$ densities can have minimal overlap for different i.



But Gibbs sampling on $p(q_t, q_{t-1}|Y_{1:t})$ on a reparameterized ("standardized") space often mixes quite efficiently.

Application to Fitzhugh-Nagumo model

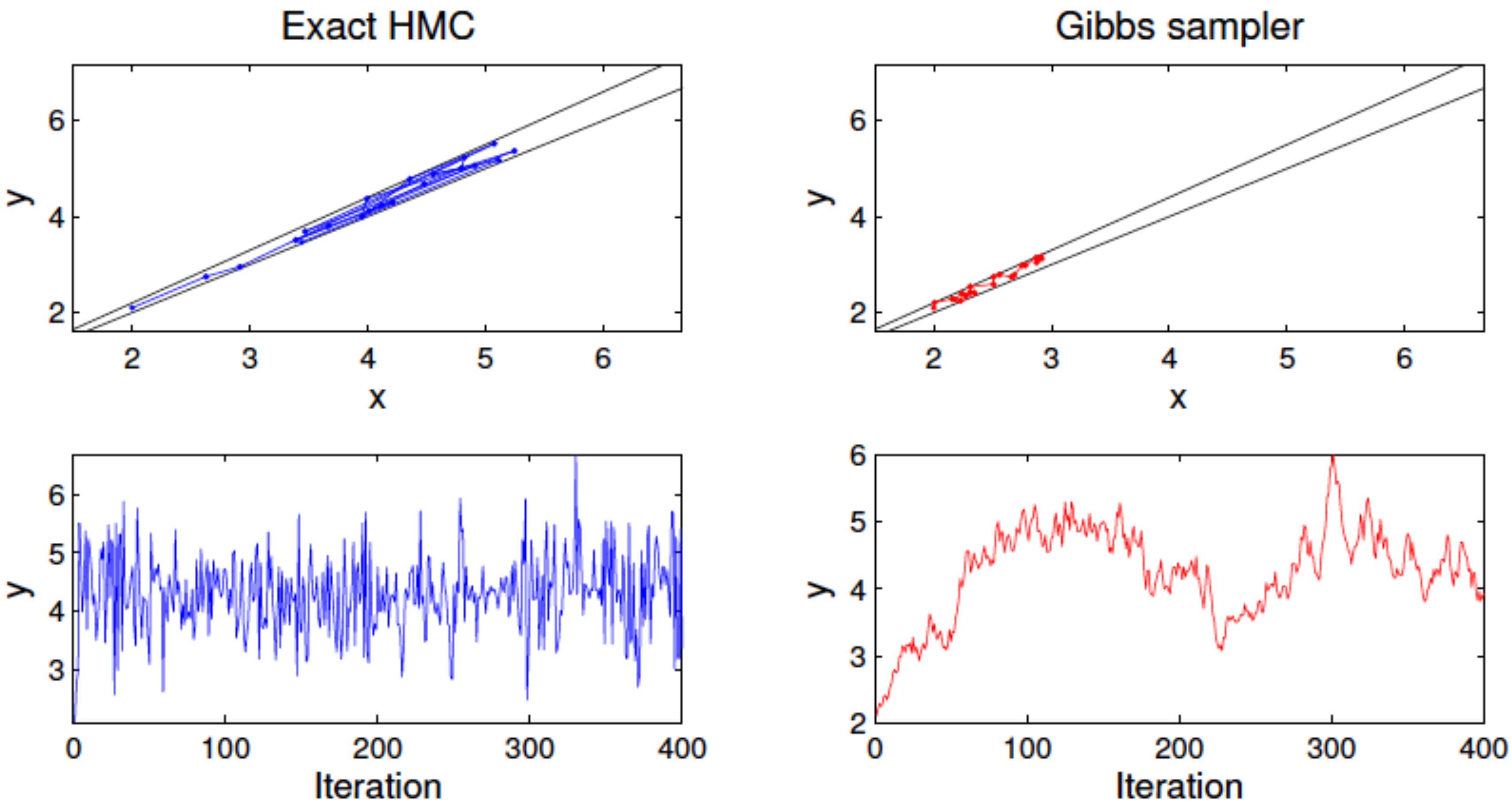


— Spike observed at $t = 0.5$.

— Standard particle filter fails; standardized Gibbs approach works very well.

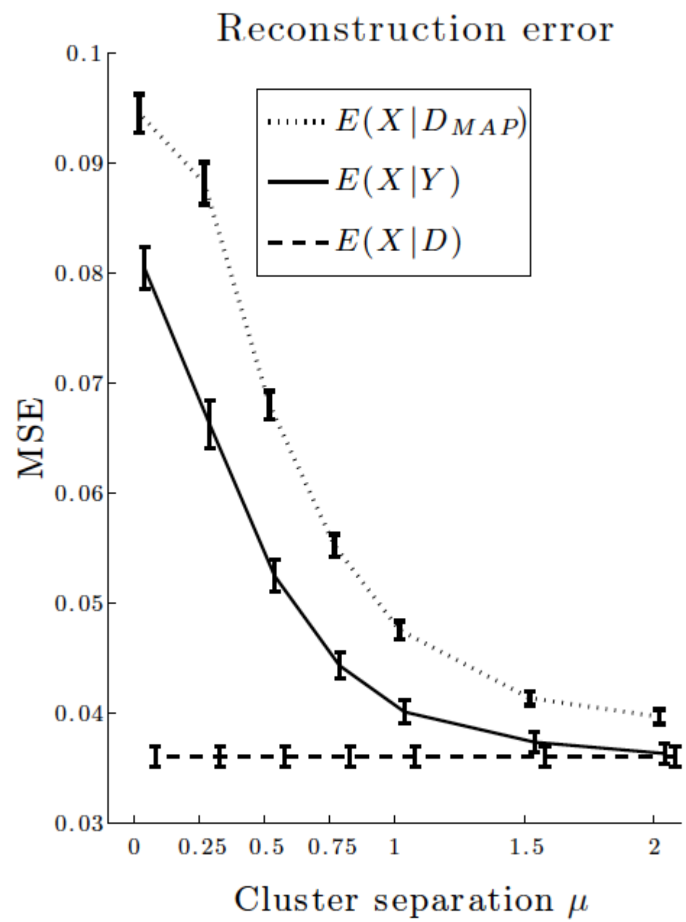
See Paninski et al (2012) for full details.

Fast, robust methods for sampling from truncated Gaussians



See Pakman and Paninski (2012) for full details.

Quantifying information loss due to corrupted or misidentified spike trains



See Smith and Paninski (2012) for full details.