Tree-structured Gaussian Process Approximations

Richard Turner and Thang Bui

May 21, 2015

Gaussian processes for time-series

$$\mathsf{x_t} = \lambda \mathsf{x}_{t-1} + \sigma_{\mathsf{x}} \eta_t$$
$$\eta_t \sim \mathcal{N}(0, 1)$$

Gaussian processes for time-series

$$\mathsf{x_t} = \lambda \mathsf{x}_{t-1} + \sigma_{\mathsf{x}} \eta_t$$
$$\eta_t \sim \mathcal{N}(0,1)$$

$$\mathsf{x_t} = f(\mathsf{x}_{t-1}) + \sigma_{\mathsf{x}} \eta_t$$
$$f(\mathsf{x}) \sim \mathcal{GP}(\mu(\mathsf{x}), K(\mathsf{x}, \mathsf{x}'))$$

$\mathsf{x}_{1:T}$ non-Gaussian

discrete-time

Gaussian processes for time-series

$$\mathsf{x_t} = \lambda \mathsf{x}_{t-1} + \sigma_{\mathsf{x}} \eta_t$$
$$\eta_t \sim \mathcal{N}(0, 1)$$

$$\mathsf{x_t} = f(\mathsf{x}_{t-1}) + \sigma_{\mathsf{x}} \eta_t$$
$$f(\mathsf{x}) \sim \mathcal{GP}(\mu(\mathsf{x}), K(\mathsf{x}, \mathsf{x}'))$$

$\mathsf{x}_{1:T}$ non-Gaussian
discrete-time

$$\mathsf{x}(t) \sim \mathcal{GP}(\mu(t), K(t, t'))$$

$\mathsf{x}_{1:T}$ Gaussian
continuous-time

Gaussian processes for time-series

$$x_t = \lambda x_{t-1} + \sigma_x \eta_t$$
$$\eta_t \sim \mathcal{N}(0,1)$$

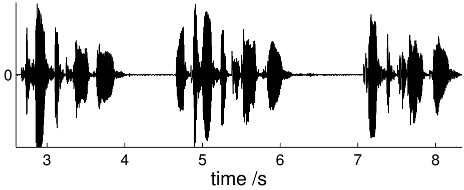
$$x_t = f(x_{t-1}) + \sigma_x \eta_t$$
$$f(x) \sim \mathcal{GP}(\mu(x), K(x, x'))$$

$x_{1:T}$ non-Gaussian
discrete-time

$$x(t) \sim \mathcal{GP}(\mu(t), K(t, t'))$$

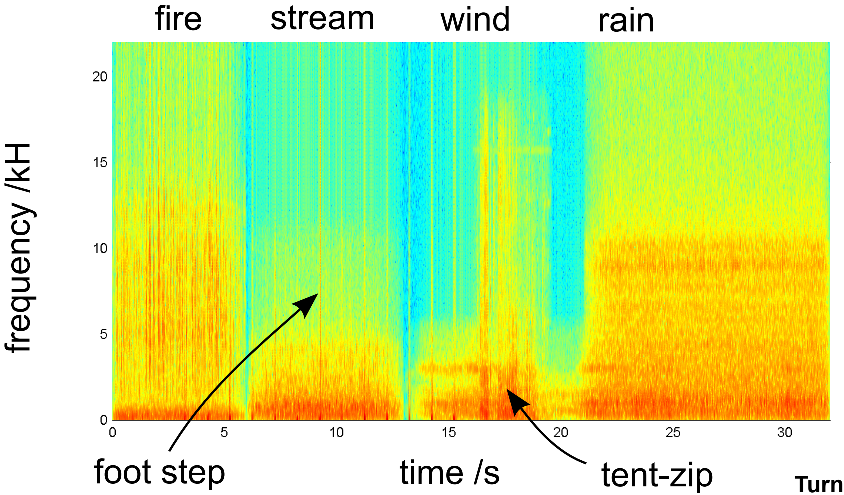
$x_{1:T}$ Gaussian
continuous-time

audio
time-series
data



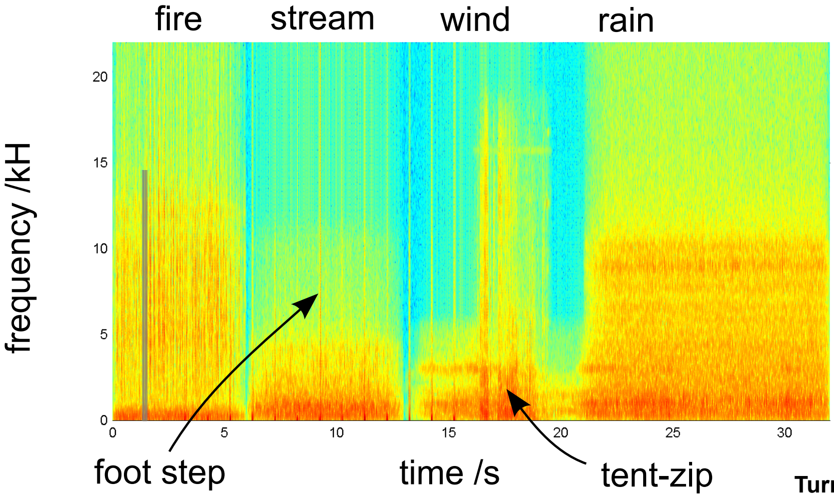
T = 10^5-10^7
datapoints

Audio texture modelling



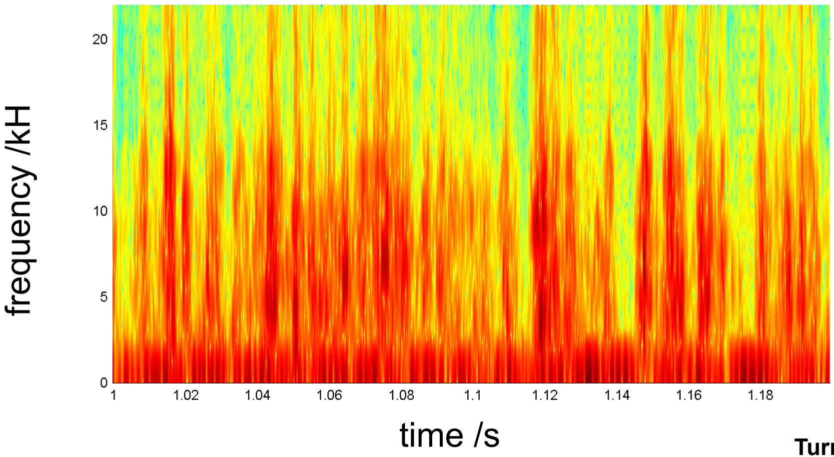
Turner, 2010

Audio texture modelling



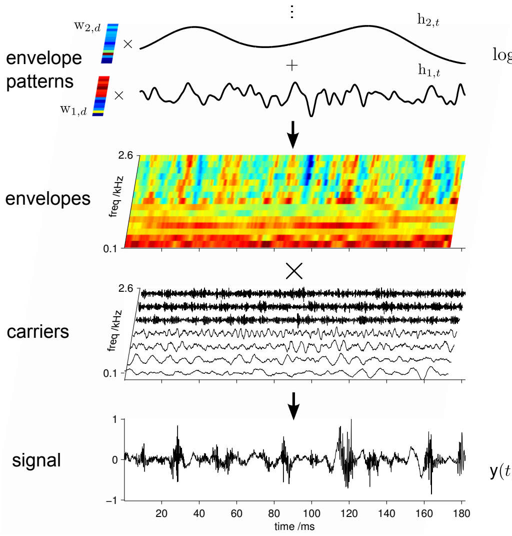
Turner, 2010

Audio texture modelling



Turner, 2010

Audio texture modelling



mean spectrum

$$\log h_{l,t} \sim \mathsf{GP}\left(\mu_k, \;\; 0 \;\; f\right)$$

= slow Gaussian process

$$a_{d,t} = \sum_{l=1}^{L} h_{l,t} w_{l,d}$$

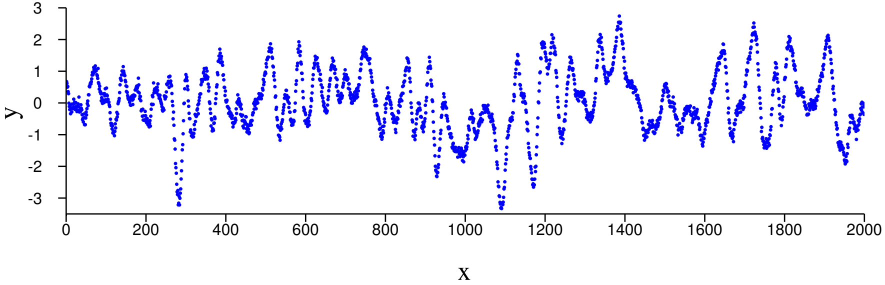
mean spectrum

$$c_d(t) \sim \mathsf{GP}\left(0, \;\; 0 \;\; f\right)$$

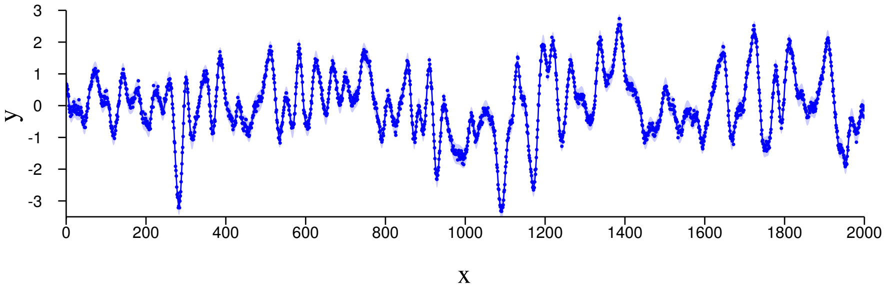
= bandpass Gaussian noise

$$y(t) = \sum_{d=1}^{D} \Re\left(x_d(t)\right) a_d(t)$$

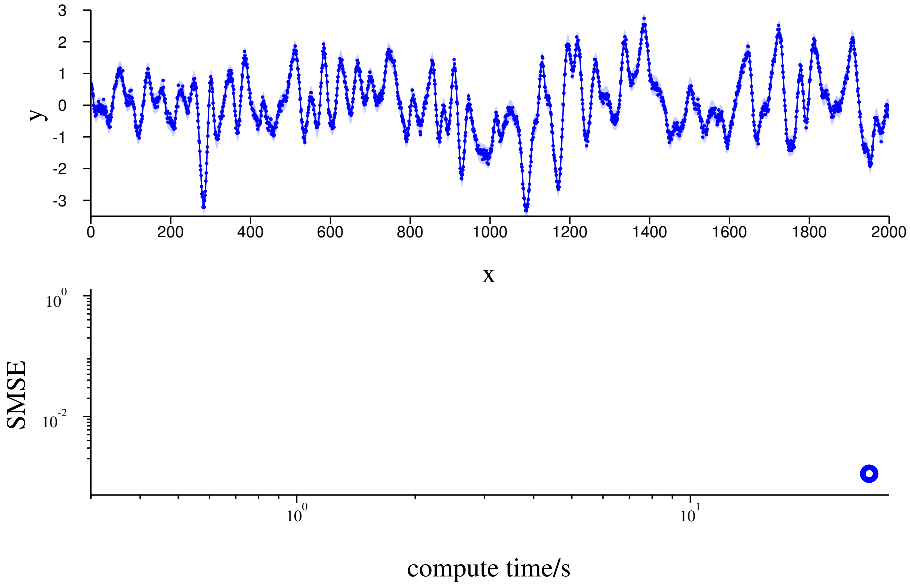
Many GP approximations are poor for time-series



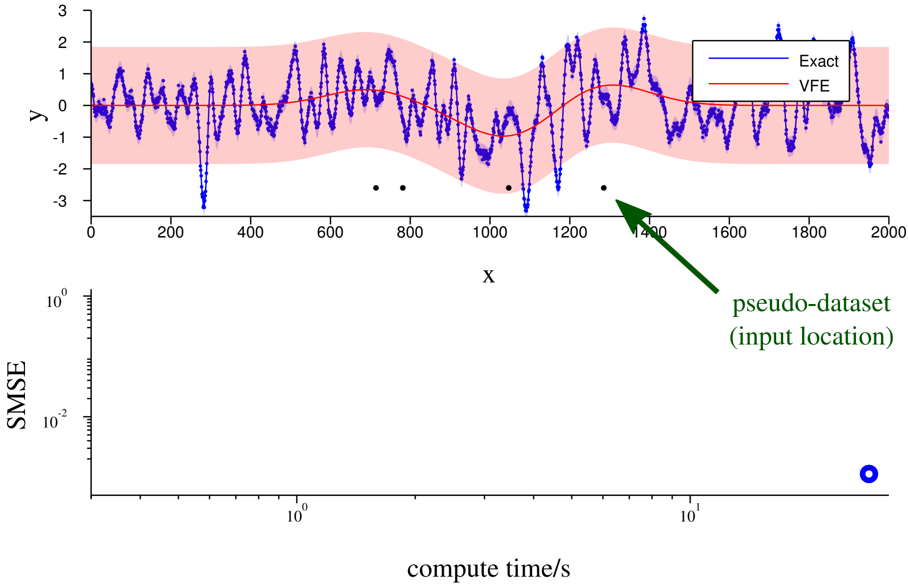
Many GP approximations are poor for time-series



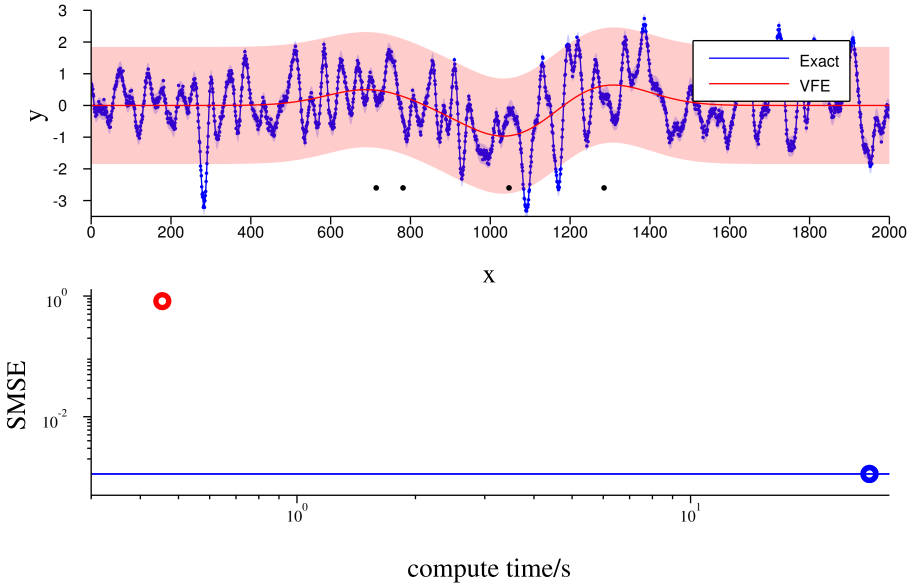
Many GP approximations are poor for time-series



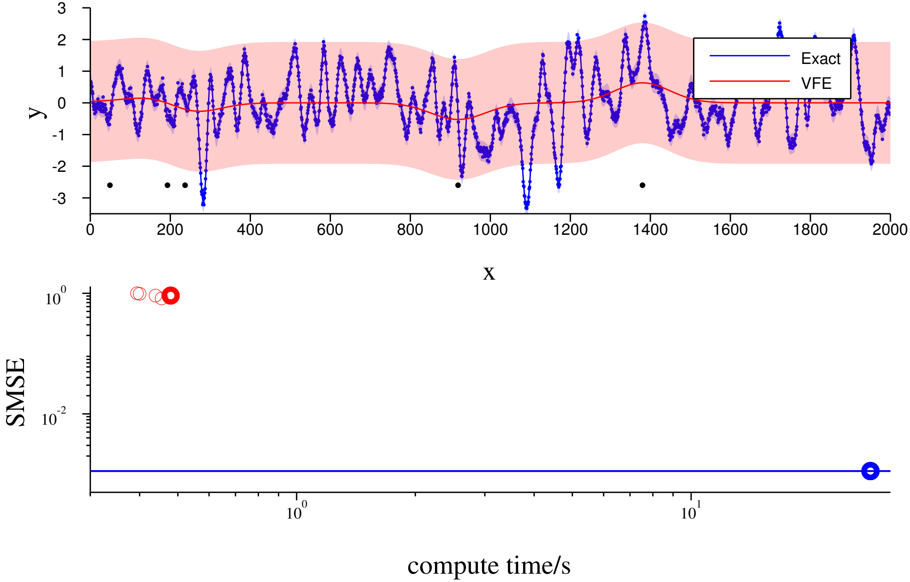
Many GP approximations are poor for time-series



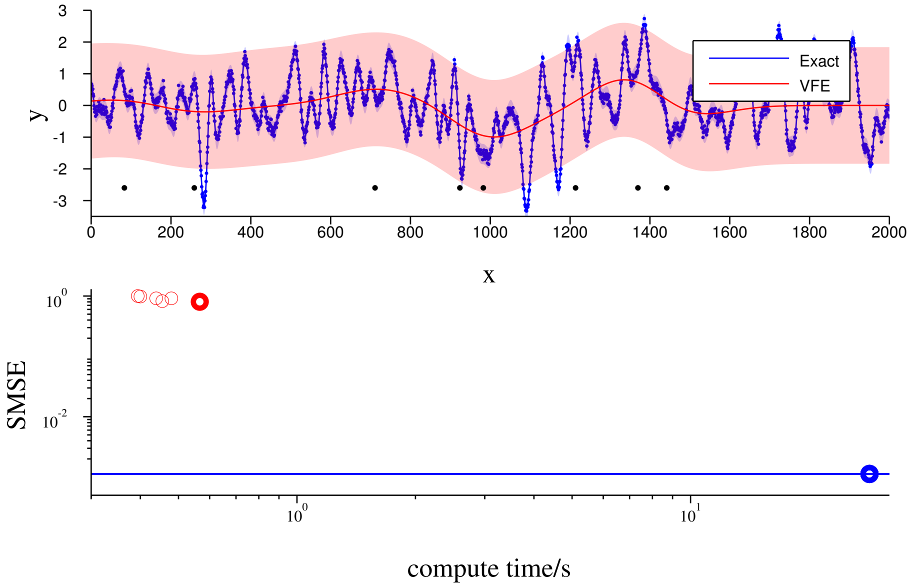
Many GP approximations are poor for time-series



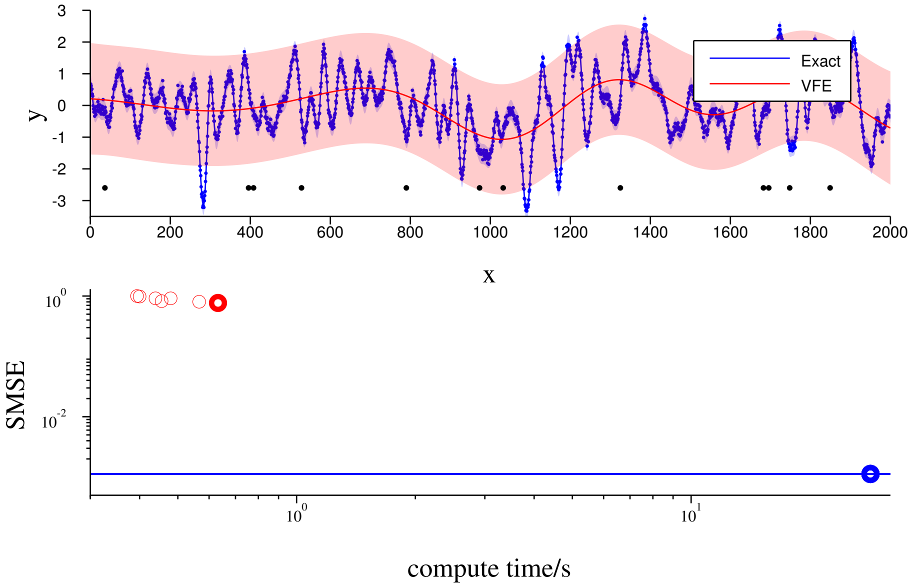
Many GP approximations are poor for time-series



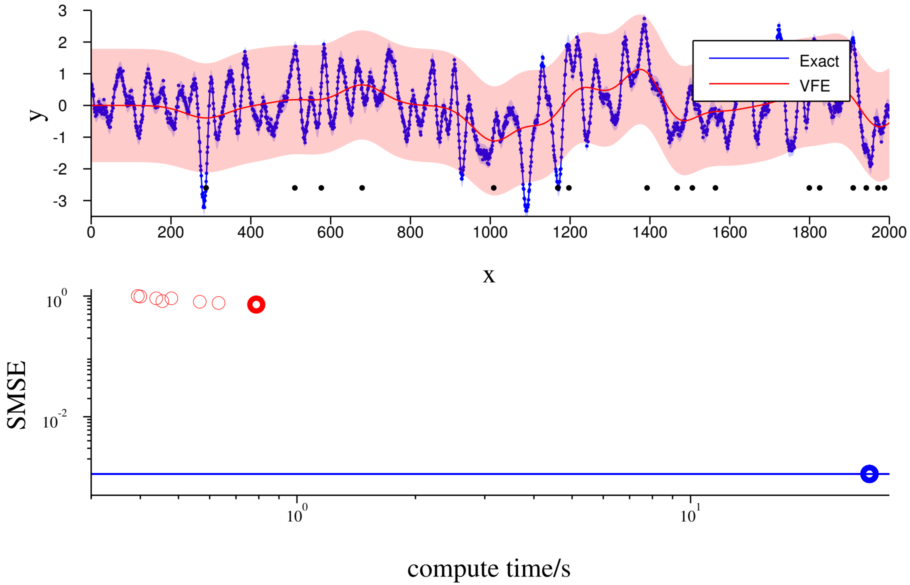
Many GP approximations are poor for time-series



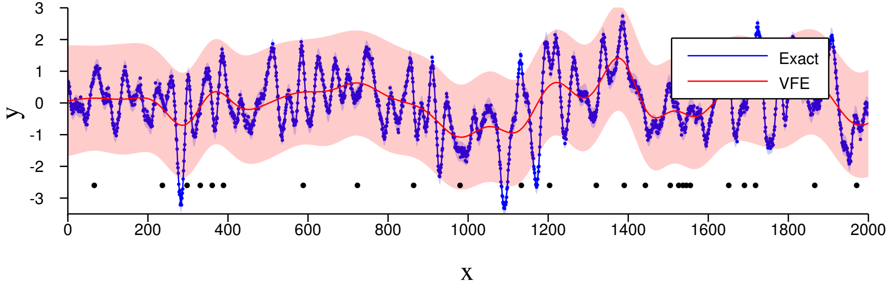
Many GP approximations are poor for time-series

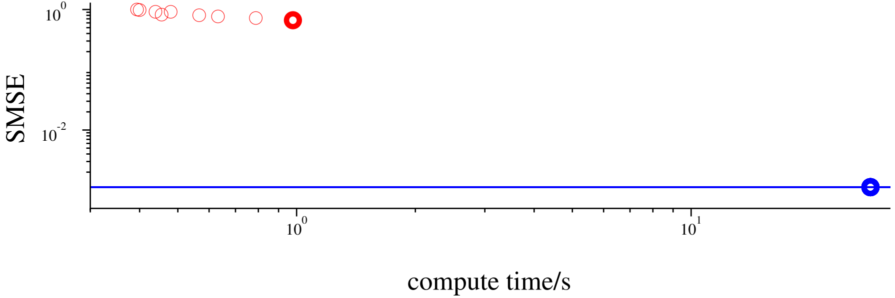


Many GP approximations are poor for time-series

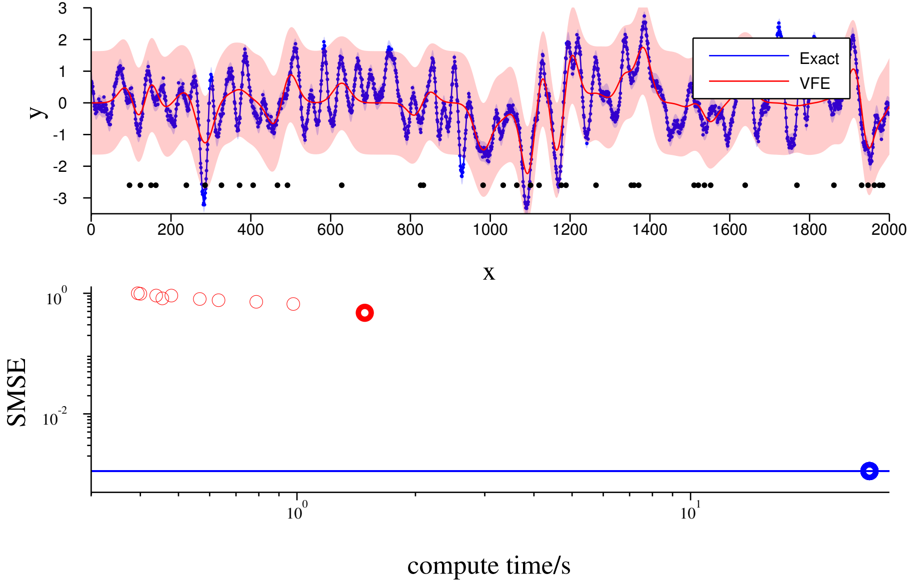


Many GP approximations are poor for time-series

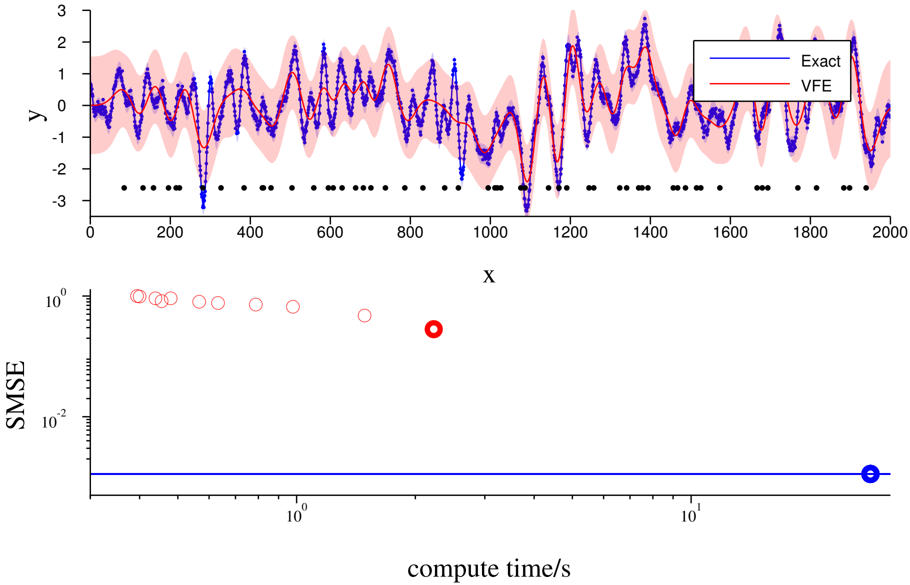




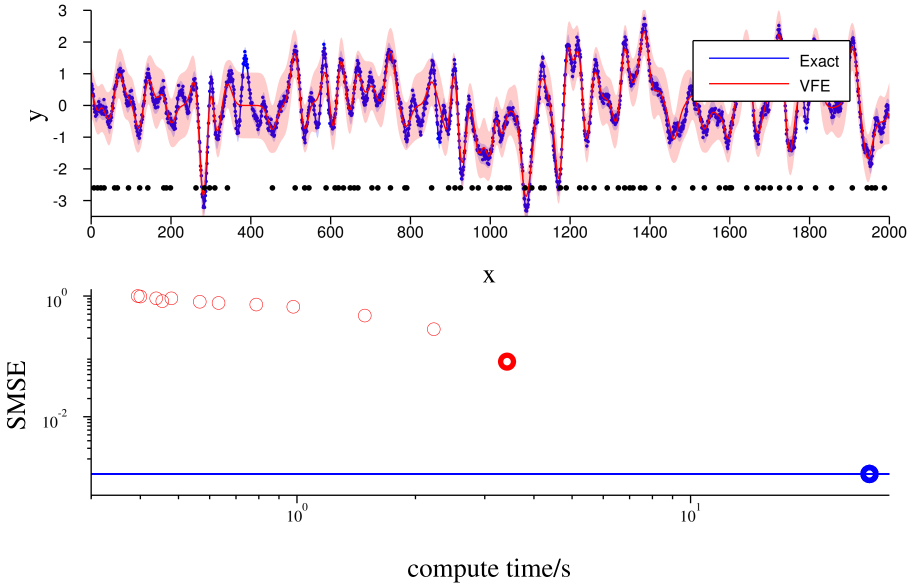
Many GP approximations are poor for time-series



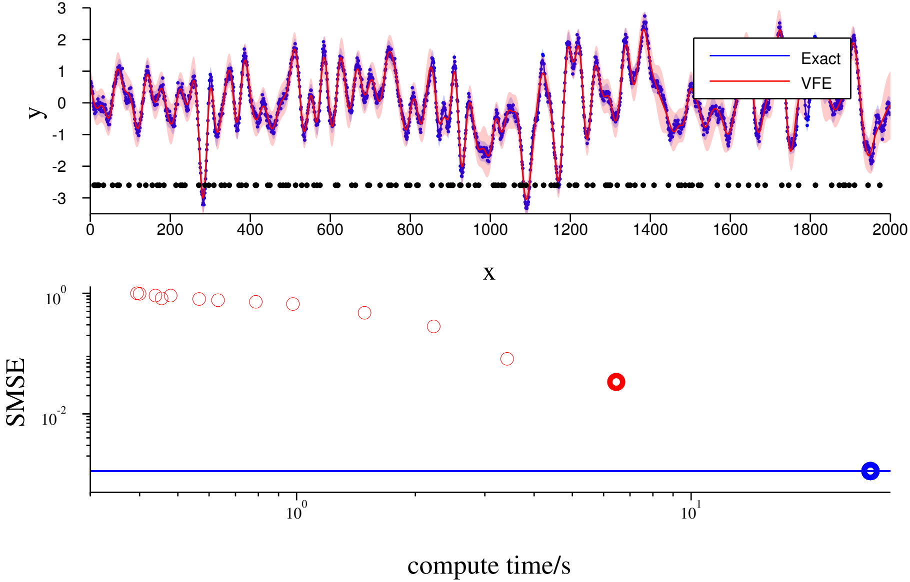
Many GP approximations are poor for time-series



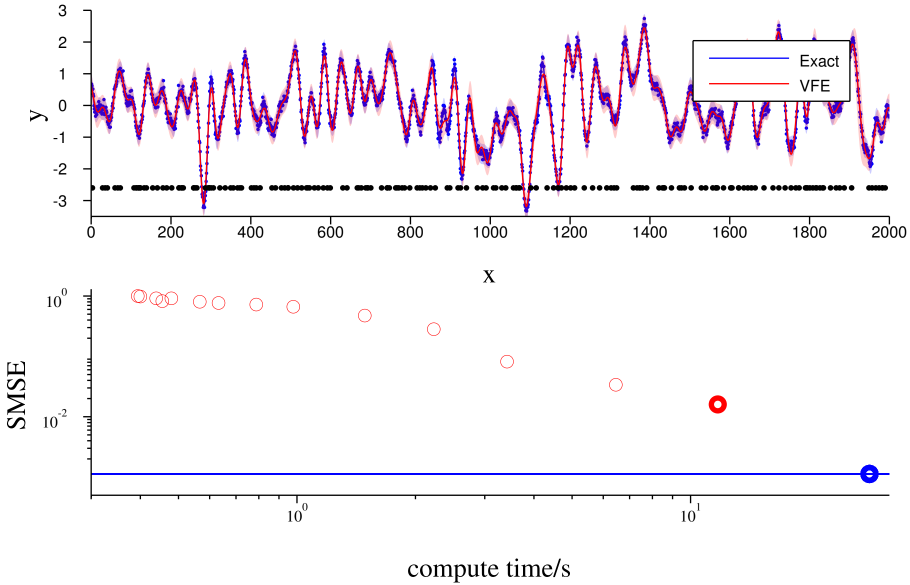
Many GP approximations are poor for time-series



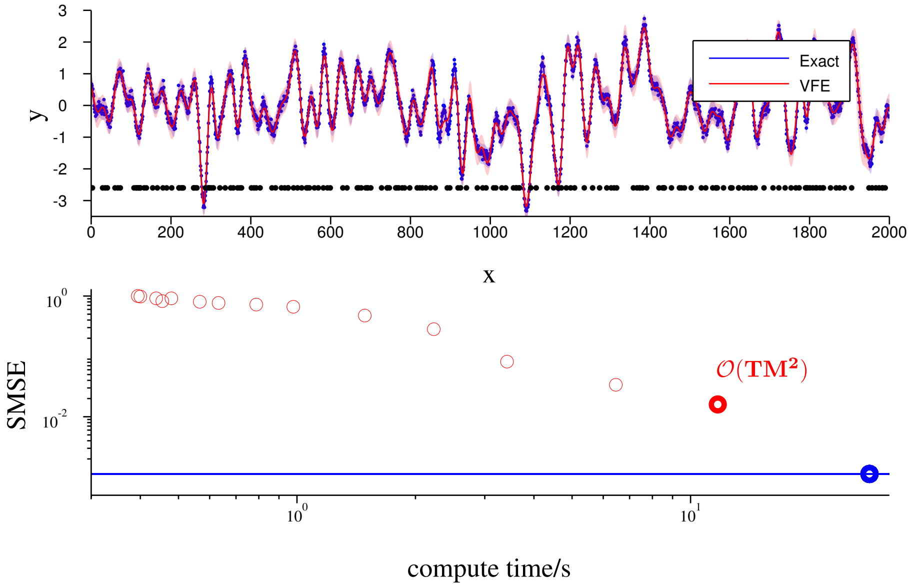
Many GP approximations are poor for time-series



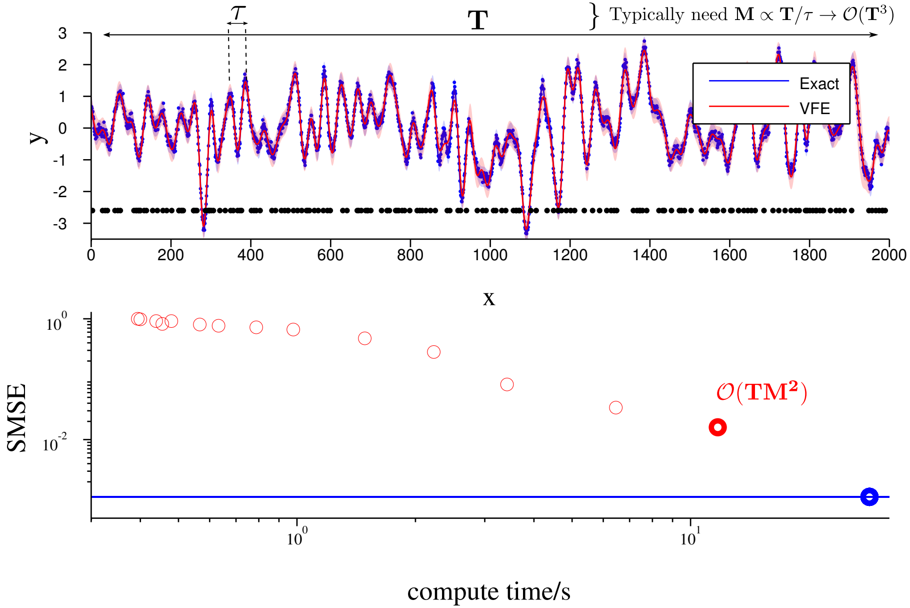
Many GP approximations are poor for time-series



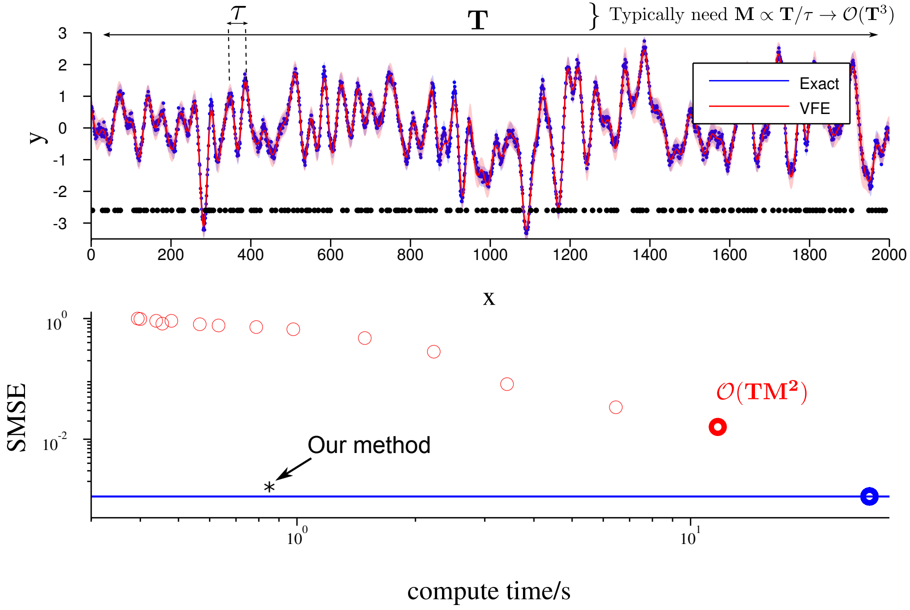
Many GP approximations are poor for time-series



Many GP approximations are poor for time-series



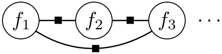
Many GP approximations are poor for time-series



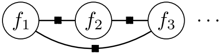
The Fully Independent Training Conditional (FITC) approximation

(Snelson and Ghahramani 2006)

The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)



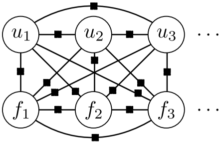
The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)



1. **Augment the model with inducing points** $\{x_m, u_m\}_{m=1}^{M}$

The Fully Independent Training Conditional (FITC) approximation

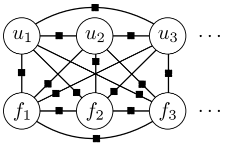
(Snelson and Ghahramani 2006)



1. **Augment the model with inducing points** $\{x_m, u_m\}_{m=1}^{M}$

The Fully Independent Training Conditional (FITC) approximation

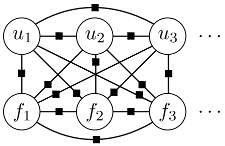
(Snelson and Ghahramani 2006)



1. **Augment the model with inducing points** $\{x_m, u_m\}_{m=1}^{M}$

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{f} \\ \mathbf{u}\end{bmatrix}; \begin{bmatrix}\mathbf{0} \\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K_{ff}} & \mathbf{K_{fu}} \\ \mathbf{K_{uf}} & \mathbf{K_{uu}}\end{bmatrix}\right)$$

The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)

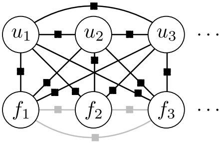


1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{u} \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K_{ff}} & \mathbf{K_{fu}} \\ \mathbf{K_{uf}} & \mathbf{K_{uu}} \end{bmatrix} \right)$$

2. **Remove *all* direct dependencies between function values $\mathbf{f}$, i.e. assume $f_i \perp\!\!\!\perp f_j | \mathbf{u}$, $\forall i, j$**

The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)

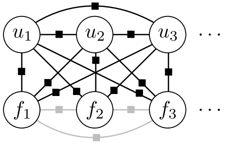


1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{f}\\ \mathbf{u}\end{bmatrix}; \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K_{ff}} & \mathbf{K_{fu}}\\ \mathbf{K_{uf}} & \mathbf{K_{uu}}\end{bmatrix}\right)$$

2. **Remove *all* direct dependencies between function values $\mathbf{f}$, i.e. assume $f_i \perp\!\!\!\perp f_j | \mathbf{u}$, $\forall i, j$**

The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)



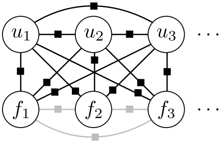
1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{u} \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \mathbf{K_{ff}} & \mathbf{K_{fu}} \\ \mathbf{K_{uf}} & \mathbf{K_{uu}} \end{bmatrix} \right)$$

2. Remove *all* direct dependencies between function values $\mathbf{f}$, i.e. assume $f_i \perp\!\!\!\perp f_j | \mathbf{u}$, $\forall i, j$
3. **Calibrate model using a forward KL divergence**

$$\underset{q(\mathbf{u}), \{q(f_i|\mathbf{u})\}_{i=1}^{N}}{\arg\min} \ \mathrm{KL}(p(\mathbf{f}, \mathbf{u}) || q(\mathbf{u}) \prod_{n=1}^{N} q(f_i | \mathbf{u}))$$

The Fully Independent Training Conditional (FITC) approximation
(Snelson and Ghahramani 2006)



1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$

$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix}\mathbf{f}\\ \mathbf{u}\end{bmatrix}; \begin{bmatrix}\mathbf{0}\\ \mathbf{0}\end{bmatrix}, \begin{bmatrix}\mathbf{K_{ff}} & \mathbf{K_{fu}}\\ \mathbf{K_{uf}} & \mathbf{K_{uu}}\end{bmatrix}\right)$$

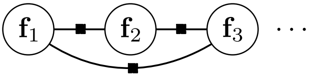
2. Remove *all* direct dependencies between function values $\mathbf{f}$, i.e. assume $f_i \perp\!\!\!\perp f_j|\mathbf{u}$, $\forall i, j$
3. **Calibrate model using a forward KL divergence**

$$\underset{q(\mathbf{u}),\{q(f_i|\mathbf{u})\}_{i=1}^{N}}{\arg\min} \; \mathrm{KL}(p(\mathbf{f}, \mathbf{u})||q(\mathbf{u})\prod_{n=1}^{N} q(f_i|\mathbf{u}))$$

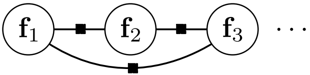
$$\Rightarrow q(\mathbf{u}) = p(\mathbf{u}) \;\;,\;\; q(f_i|\mathbf{u}) = p(f_i|\mathbf{u})$$

The chain-structured pseudo point approximation

The chain-structured pseudo point approximation

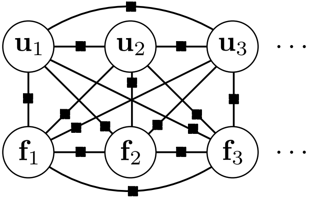


The chain-structured pseudo point approximation



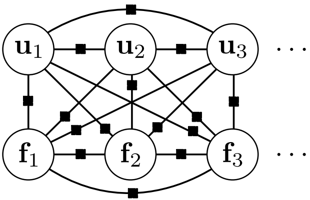
1. **Augment the model with inducing points** $\{x_m, u_m\}_{m=1}^{M}$

The chain-structured pseudo point approximation



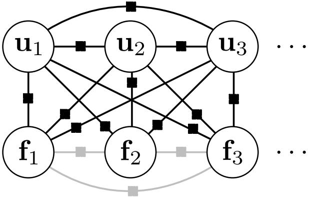
1. **Augment the model with inducing points** $\{x_m, u_m\}_{m=1}^{M}$

The chain-structured pseudo point approximation



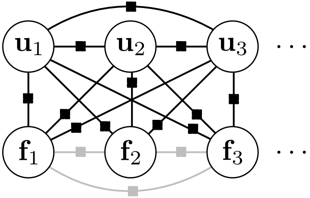
1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. **Remove *some* direct dependencies between function values $\mathbf{f}$,**

The chain-structured pseudo point approximation



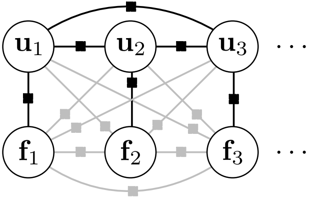
1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^M$
2. **Remove *some* direct dependencies between function values $\mathbf{f}$,**

The chain-structured pseudo point approximation



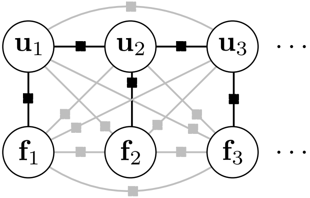
1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. Remove *some* direct dependencies between function values $\mathbf{f}$, **and assume a chain structure on inducing points $\mathbf{u}$**

The chain-structured pseudo point approximation



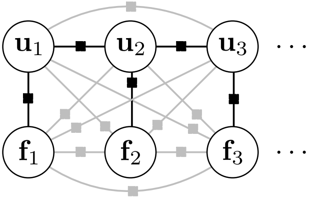
1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. Remove *some* direct dependencies between function values $\mathbf{f}$, **and assume a chain structure on inducing points $\mathbf{u}$**

The chain-structured pseudo point approximation



1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. Remove *some* direct dependencies between function values $\mathbf{f}$, **and assume a chain structure on inducing points $\mathbf{u}$**

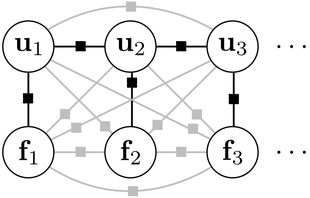
The chain-structured pseudo point approximation



1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. Remove *some* direct dependencies between function values $\mathbf{f}$, and assume a chain structure on inducing points $\mathbf{u}$
3. **Calibrate model using a forward KL divergence**

$$\underset{\{q(\mathbf{u}_k|\mathbf{u}_{k-1}),q(\mathbf{f}_k|\mathbf{u}_k)\}_{k=1}^{K}}{\arg\min} \mathrm{KL}(p(\mathbf{f},\mathbf{u})|| \prod_k q(\mathbf{u}_k|\mathbf{u}_{k-1})q(\mathbf{f}_k|\mathbf{u}_k))$$

The chain-structured pseudo point approximation

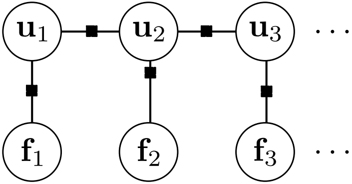


1. Augment the model with inducing points $\{x_m, u_m\}_{m=1}^{M}$
2. Remove *some* direct dependencies between function values $\mathbf{f}$, and assume a chain structure on inducing points $\mathbf{u}$
3. **Calibrate model using a forward KL divergence**

$$\underset{\{q(\mathbf{u}_k|\mathbf{u}_{k-1}), q(\mathbf{f}_k|\mathbf{u}_k)\}_{k=1}^{K}}{\arg\min} \mathrm{KL}(p(\mathbf{f}, \mathbf{u}) || \prod_k q(\mathbf{u}_k|\mathbf{u}_{k-1}) q(\mathbf{f}_k|\mathbf{u}_k))$$

$$\Rightarrow q(\mathbf{u}_k|\mathbf{u}_{k-1}) = p(\mathbf{u}_k|\mathbf{u}_{k-1}) \;\; , \;\; q(\mathbf{f}_k|\mathbf{u}_k) = p(\mathbf{f}_k|\mathbf{u}_k)$$

The chain-structured pseudo point approximation



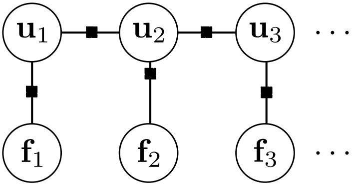
The chain-structured pseudo point approximation

New generative model:

$$q(\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{u}_k|\mathbf{u}_{k-1}),$$

$$q(\mathbf{f}|\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{f}_k|\mathbf{u}_k),$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n; f_n, \sigma_n^2).$$



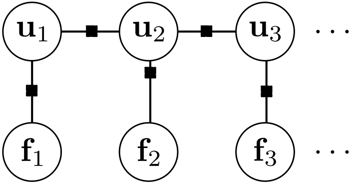
The chain-structured pseudo point approximation

New generative model:

$$q(\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{u}_k|\mathbf{u}_{k-1}),$$

$$q(\mathbf{f}|\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{f}_k|\mathbf{u}_k),$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n; f_n, \sigma_n^2).$$



where

$$q(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = p(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = \mathcal{N}(\mathbf{u}_{\mathrm{B}_k}; \mathbf{A}_k\mathbf{u}_{\mathrm{B}_{k-1}}, \mathbf{Q}_k),$$
$$q(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = p(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = \mathcal{N}(\mathbf{f}_{\mathrm{B}_k}; \mathbf{C}_k\mathbf{u}_{\mathrm{B}_k}, \mathbf{R}_k).$$

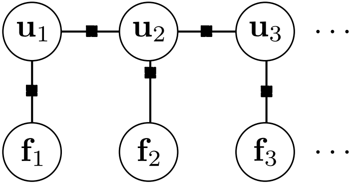
The chain-structured pseudo point approximation

New generative model:

$$q(\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{u}_k|\mathbf{u}_{k-1}),$$

$$q(\mathbf{f}|\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{f}_k|\mathbf{u}_k),$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n; f_n, \sigma_n^2).$$



where

$$q(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = p(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = \mathcal{N}(\mathbf{u}_{\mathrm{B}_k}; \mathbf{A}_k\mathbf{u}_{\mathrm{B}_{k-1}}, \mathbf{Q}_k),$$
$$q(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = p(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = \mathcal{N}(\mathbf{f}_{\mathrm{B}_k}; \mathbf{C}_k\mathbf{u}_{\mathrm{B}_k}, \mathbf{R}_k).$$

- This is a Linear Dynamical System with a *strange* parameterisation!

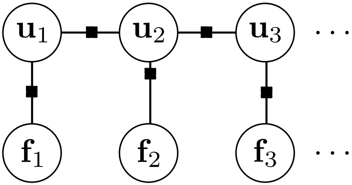
The chain-structured pseudo point approximation

New generative model:

$$q(\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{u}_k|\mathbf{u}_{k-1}),$$

$$q(\mathbf{f}|\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{f}_k|\mathbf{u}_k),$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n; f_n, \sigma_n^2).$$



where

$$q(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = p(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = \mathcal{N}(\mathbf{u}_{\mathrm{B}_k}; \mathbf{A}_k \mathbf{u}_{\mathrm{B}_{k-1}}, \mathbf{Q}_k),$$

$$q(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = p(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = \mathcal{N}(\mathbf{f}_{\mathrm{B}_k}; \mathbf{C}_k \mathbf{u}_{\mathrm{B}_k}, \mathbf{R}_k).$$

- This is a Linear Dynamical System with a *strange* parameterisation!
- Inference using Kalman smoothing algorithm

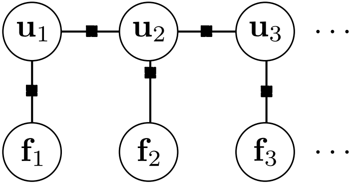
The chain-structured pseudo point approximation

New generative model:

$$q(\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{u}_k|\mathbf{u}_{k-1}),$$

$$q(\mathbf{f}|\mathbf{u}) = \prod_{k=1}^{K} q(\mathbf{f}_k|\mathbf{u}_k),$$

$$p(\mathbf{y}|\mathbf{f}) = \prod_{n=1}^{N} p(y_n; f_n, \sigma_n^2).$$

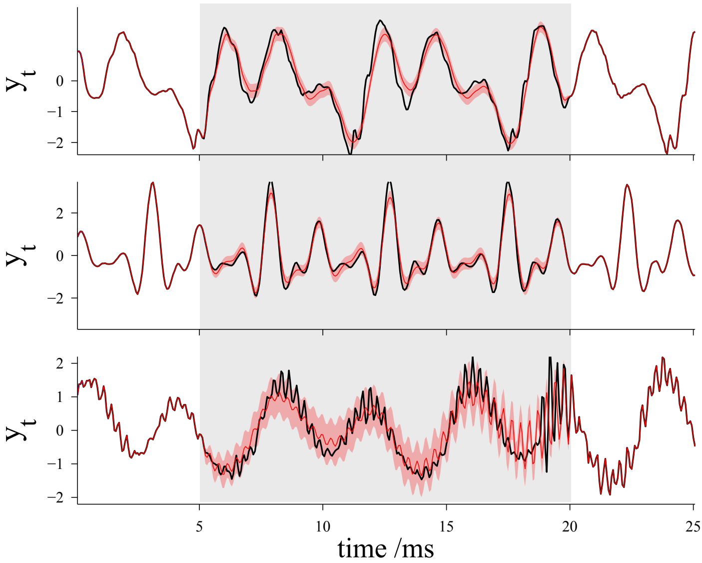


where

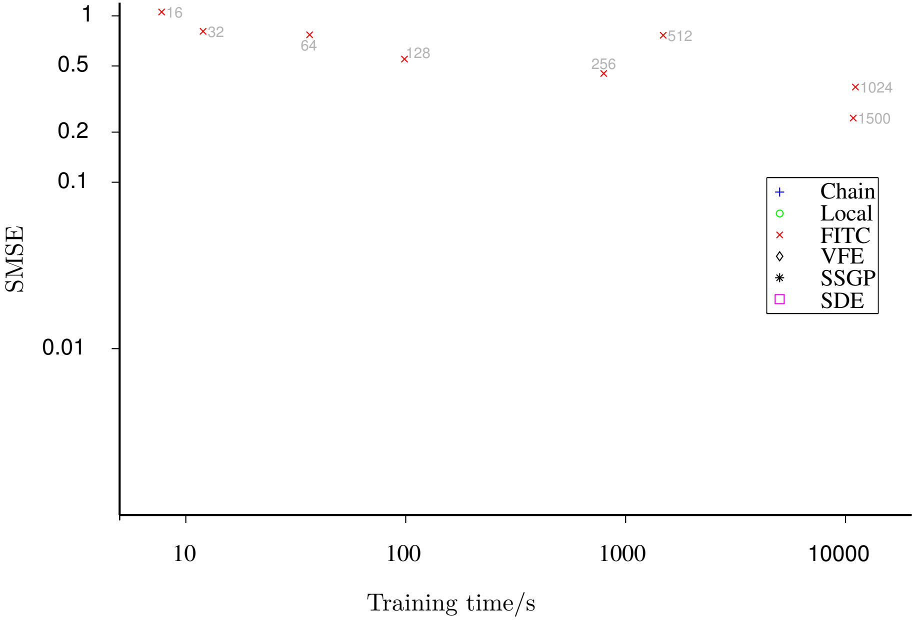
$$q(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = p(\mathbf{u}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_{k-1}}) = \mathcal{N}(\mathbf{u}_{\mathrm{B}_k}; \mathbf{A}_k\mathbf{u}_{\mathrm{B}_{k-1}}, \mathbf{Q}_k),$$
$$q(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = p(\mathbf{f}_{\mathrm{B}_k}|\mathbf{u}_{\mathrm{B}_k}) = \mathcal{N}(\mathbf{f}_{\mathrm{B}_k}; \mathbf{C}_k\mathbf{u}_{\mathrm{B}_k}, \mathbf{R}_k).$$

- This is a Linear Dynamical System with a *strange* parameterisation!
- Inference using Kalman smoothing algorithm
- Complexity: $\mathcal{O}(TD^2)$, D: average number of observations per block

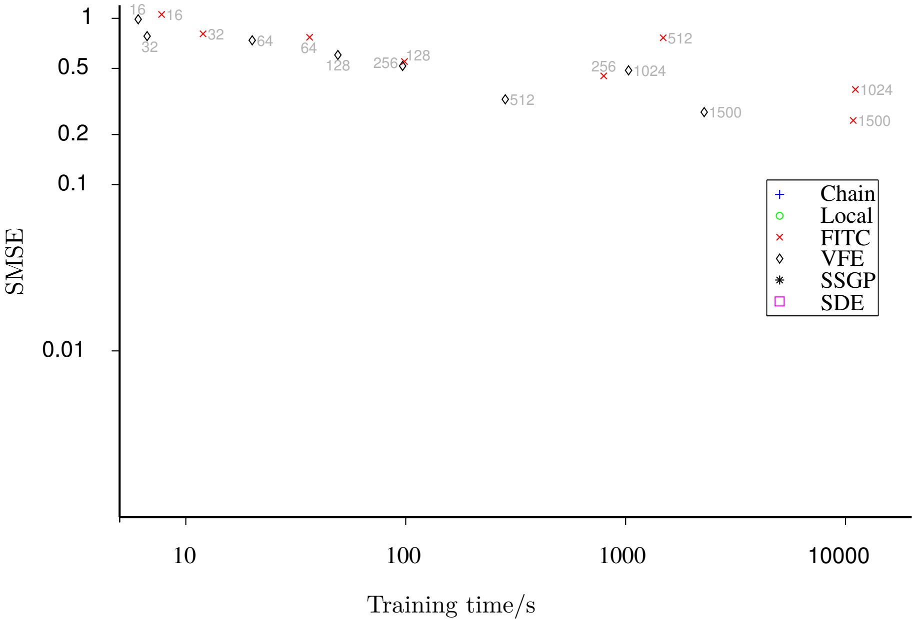
Results: Audio missing data imputation



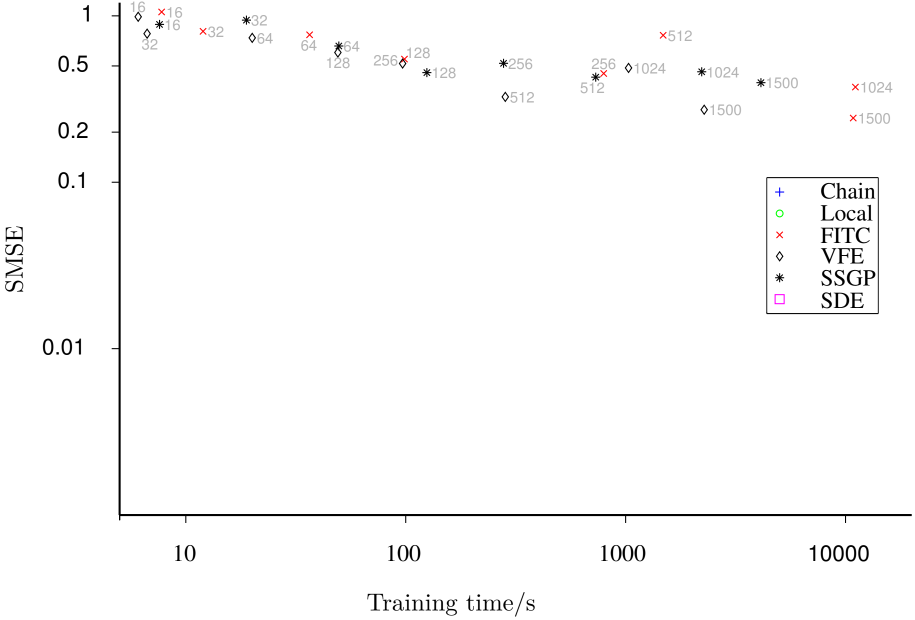
Results: Speed accuracy trade-off



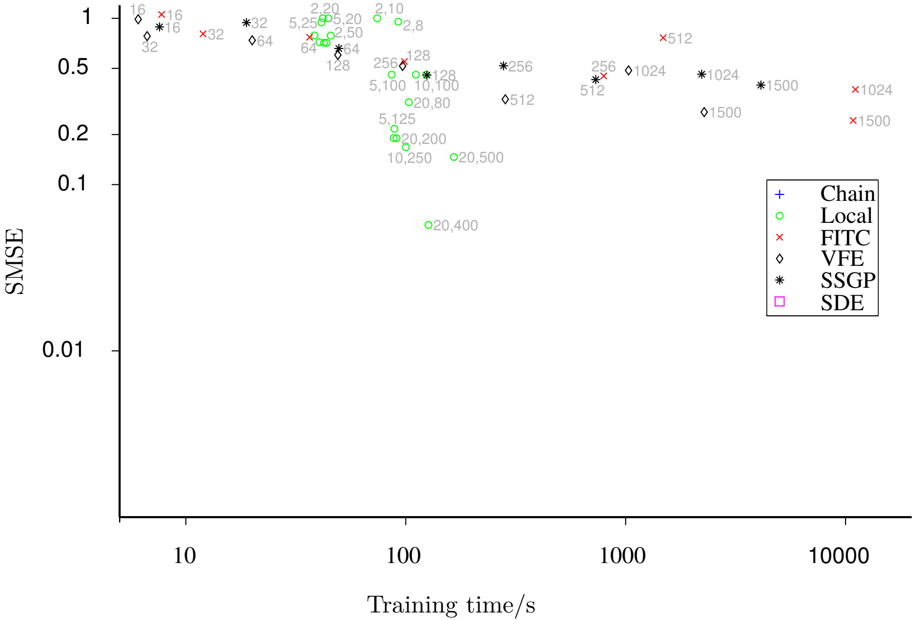
Results: Speed accuracy trade-off



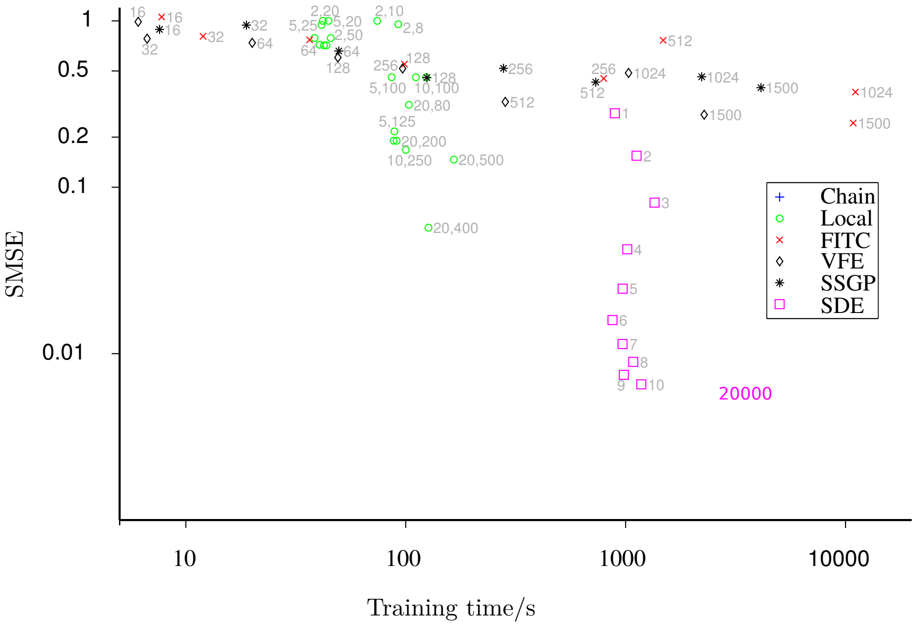
Results: Speed accuracy trade-off



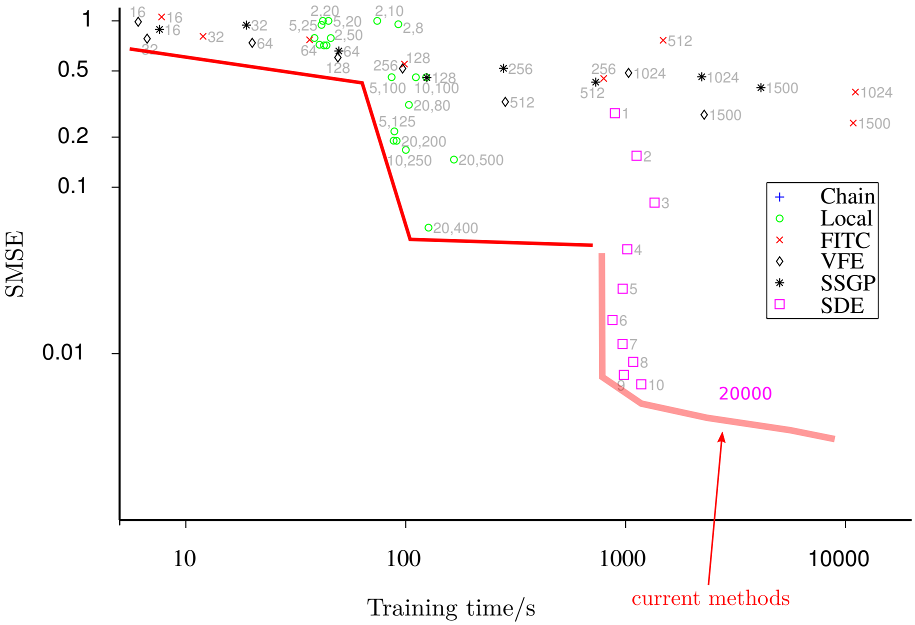
Results: Speed accuracy trade-off



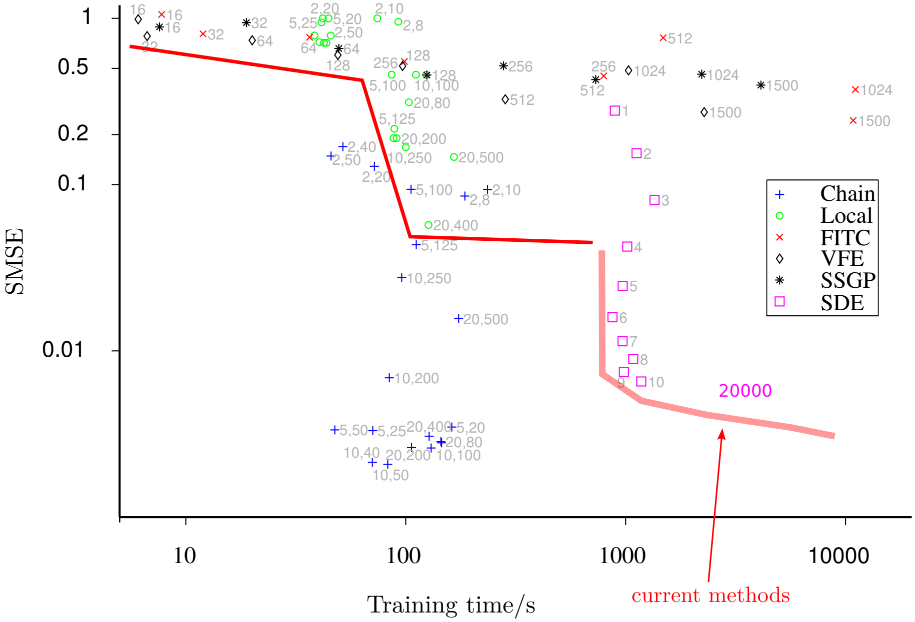
Results: Speed accuracy trade-off



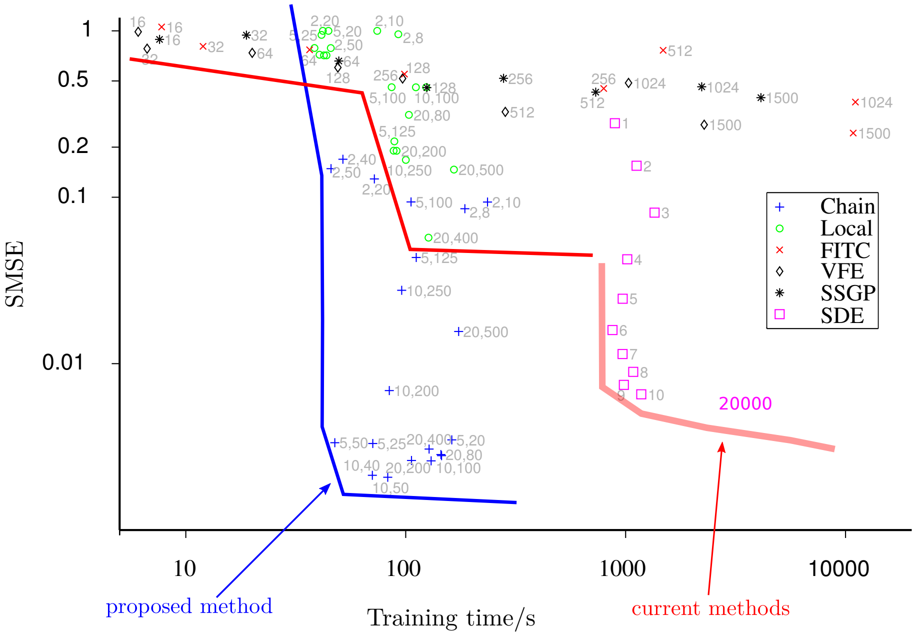
Results: Speed accuracy trade-off



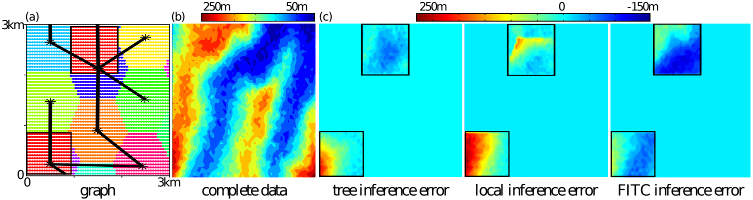
Results: Speed accuracy trade-off



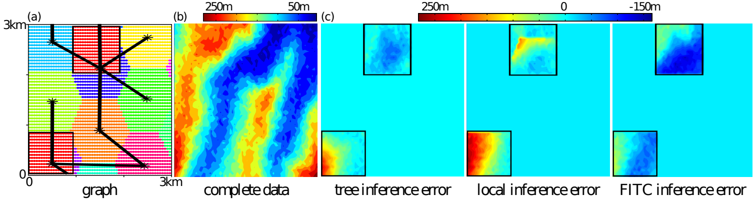
Results: Speed accuracy trade-off

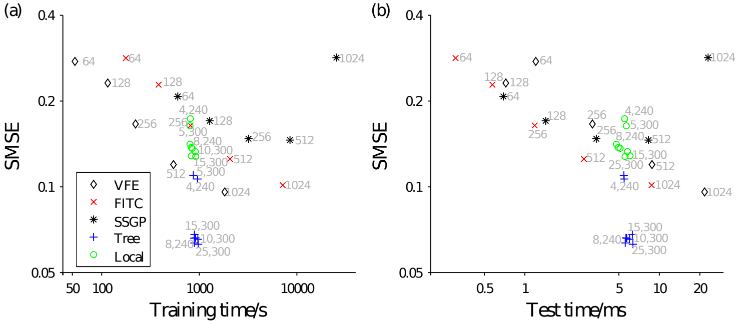


Results: 2D spatial dataset (tree structured)



Results: 2D spatial dataset (tree structured)





Summary

- pseudo-dataset approximation methods **must grow in size with the length of the time-series**
- **simple extension to FITC** (or PITC) that imposes tree-structured conditional dependencies
- fast inference by the **up-down** algorithm

Open questions and current work

- indirect approximation method
  - involves exact inference in an approximate model
  - can we use similar ideas for direct approximation of the true posterior?
- connections between GPs and time-frequency analysis
  - multi-rate filters and striding as variational free-energy + FFT based approximations
  - rediscover Nyquist in the context of limits on GP approximation accuracy