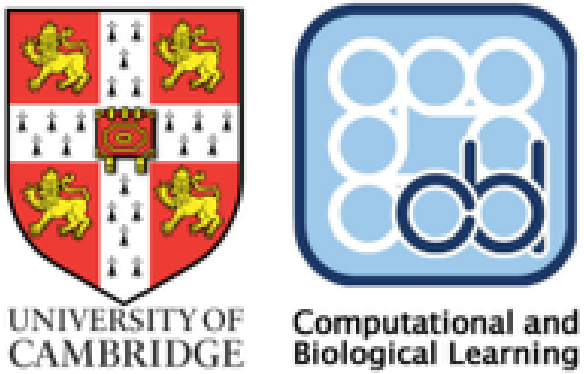
Gaussian processes approximations for time series

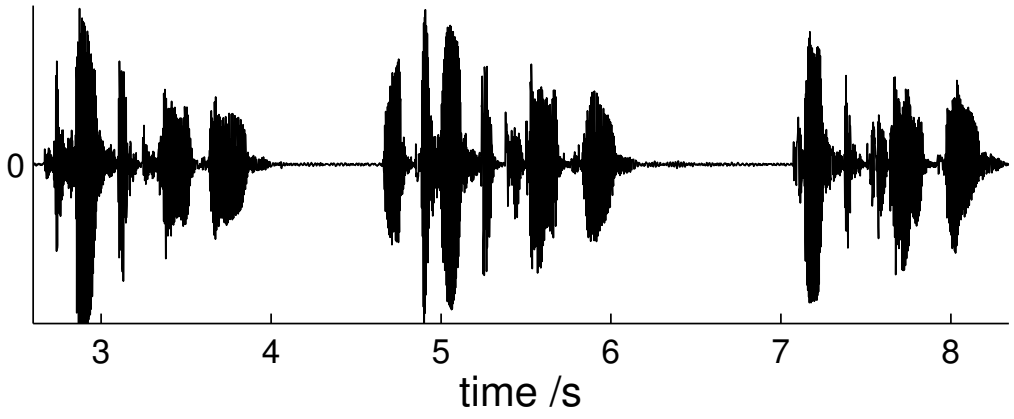
Dr. Richard E. Turner (ret26@cam.ac.uk)

Computational and Biological Learning Lab, Department of Engineering,
University of Cambridge



Using Gaussian processes for time-series modelling

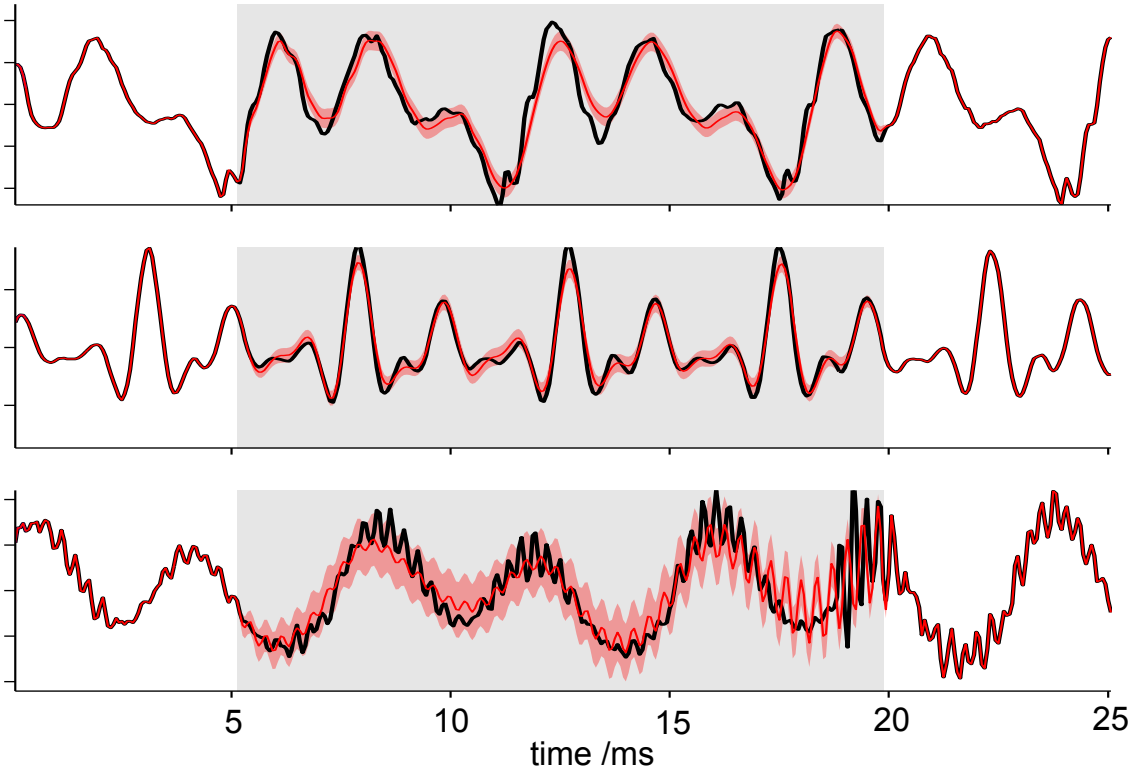
audio time-series data



$y(t)$

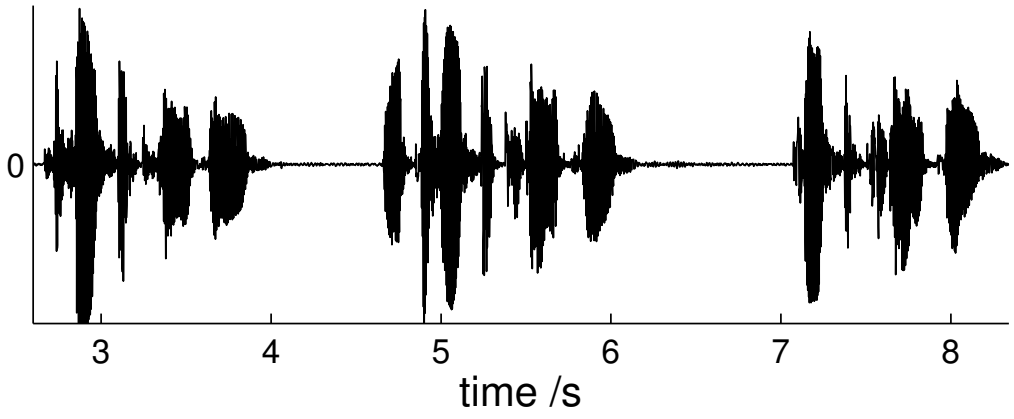
$T = 10^5\text{-}10^7$ datapoints

reconstruction using a GP model



Using Gaussian processes for time-series modelling

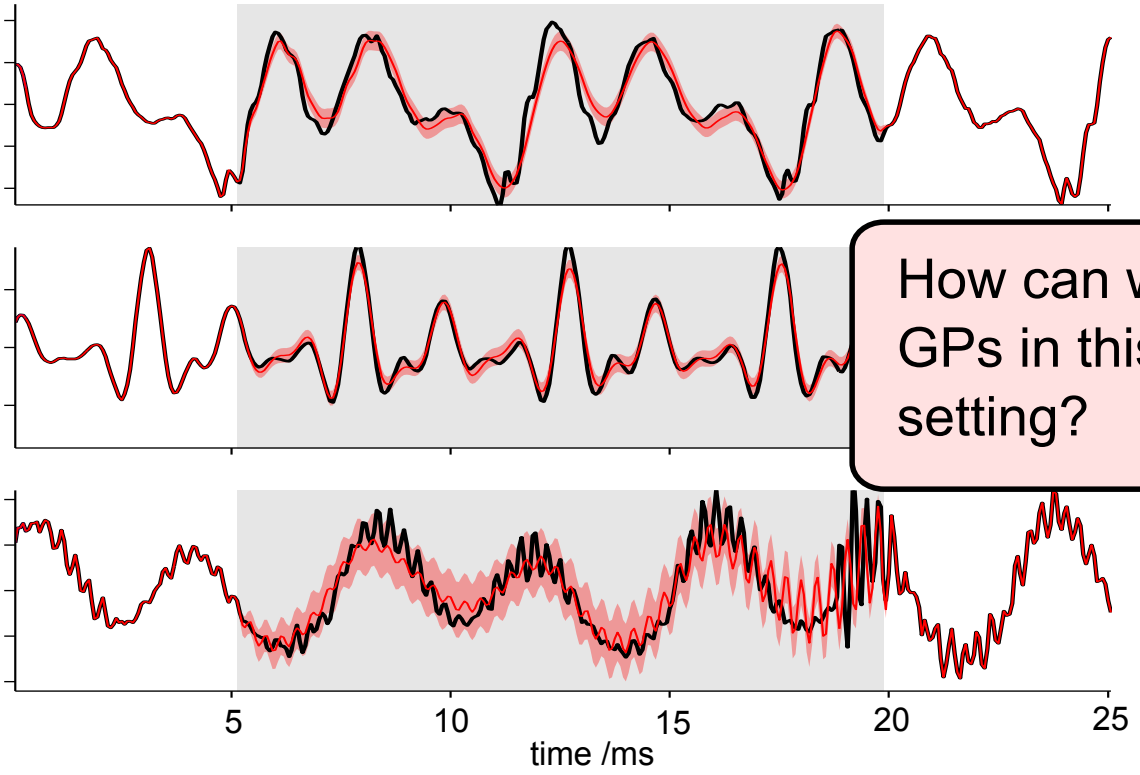
audio time-series data



$y(t)$

$T = 10^5\text{-}10^7$ datapoints

reconstruction using a GP model



How can we use GPs in this setting?

GP regression: introducing notation

Generative model (like non-linear regression)

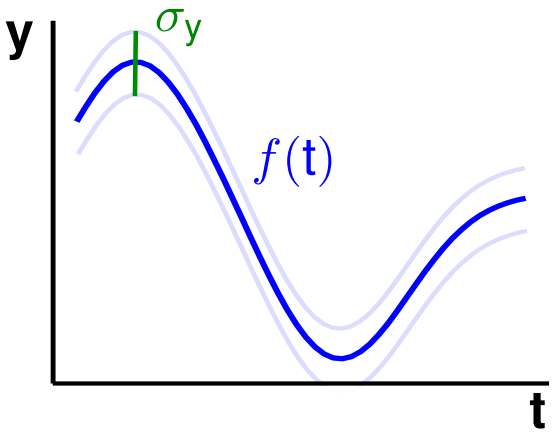
$$y(t) = f(t) + \epsilon\sigma_y \quad \text{(independent Gaussian noise)}$$

$$p(\epsilon) = \mathcal{N}(0, 1)$$

place GP prior over the non-linear function

$$p(f(t)|\theta) = \mathcal{GP}(0, K(t, t'))$$

$$K(t, t') = \sigma^2 \cos(\omega(t - t')) \exp\left(-\frac{1}{2l^2}(t - t')^2\right)$$



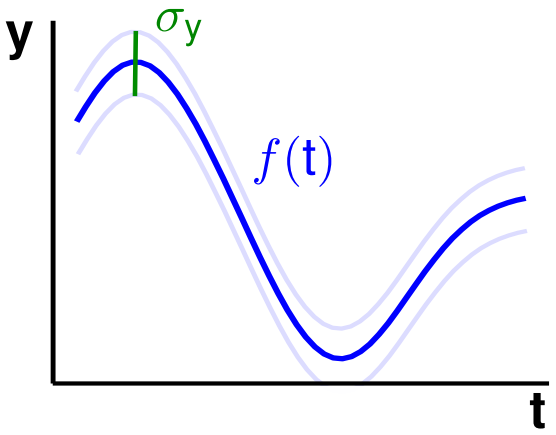
typical time-series covariance
sinusoids with SE envelopes
power in Gaussian subband

GP regression: introducing notation

Generative model (like non-linear regression)

$\mathsf{y}(\mathsf{t}) = f(\mathsf{t}) + \epsilon\sigma_{\mathsf{y}}$ (independent Gaussian noise)

$p(\epsilon) = \mathcal{N}(0, 1)$



place GP prior over the non-linear function

$p(f(\mathsf{t})|\theta) = \mathcal{GP}(0, \mathsf{K}(\mathsf{t}, \mathsf{t}'))$

$$\mathsf{K}(\mathsf{t}, \mathsf{t}') = \sigma^2 \cos(\omega(\mathsf{t} - \mathsf{t}')) \exp\left(-\frac{1}{2l^2}(\mathsf{t} - \mathsf{t}')^2\right)$$

typical time-series covariance
sinusoids with SE envelopes
power in Gaussian subband

another popular class of GP time-series models

$$\mathsf{x}_{\mathsf{t}} = g(\mathsf{x}_{t-1}) + \sigma_{\mathsf{x}}\eta_t$$

$$\mathsf{y}_t = f(\mathsf{x}_t) + \sigma_{\mathbf{y}}\epsilon_t$$

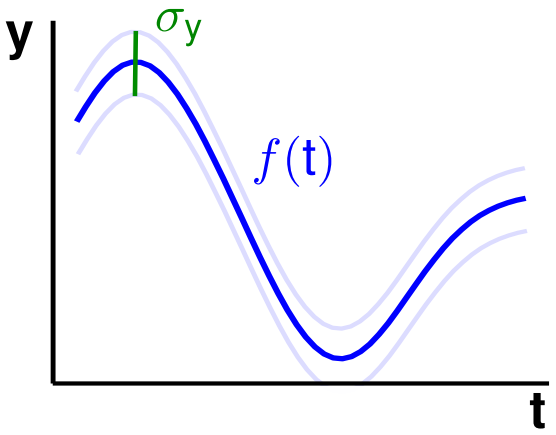
$$f(\mathsf{x}), g(\mathsf{x}) \sim \mathcal{GP}(0, \mathsf{K})$$

GP regression: introducing notation

Generative model (like non-linear regression)

$\mathsf{y}(\mathsf{t}) = f(\mathsf{t}) + \epsilon\sigma_{\mathsf{y}}$ (independent Gaussian noise)

$p(\epsilon) = \mathcal{N}(0, 1)$



place GP prior over the non-linear function

$$p(f(\mathsf{t})|\theta) = \mathcal{GP}(0, \mathsf{K}(\mathsf{t}, \mathsf{t}'))$$

$$\mathsf{K}(\mathsf{t}, \mathsf{t}') = \sigma^2 \cos(\omega(\mathsf{t} - \mathsf{t}')) \exp\left(-\frac{1}{2l^2}(\mathsf{t} - \mathsf{t}')^2\right)$$

typical time-series covariance
sinusoids with SE envelopes
power in Gaussian subband

sum of two Gaussians is a Gaussian,
$\Longrightarrow$ induces GP over $\mathsf{y}(\mathsf{t})$

$$p(\mathsf{y}(\mathsf{t})|\theta) = \mathcal{GP}(0, \underbrace{\mathsf{K}(\mathsf{t}, \mathsf{t}') + \mathrm{I}\sigma_{\mathsf{y}}^2}_{\Sigma(\mathsf{t}, \mathsf{t}')})$$

another popular class of GP time-series models

$$\mathsf{x}_{\mathsf{t}} = g(\mathsf{x}_{t-1}) + \sigma_{\mathsf{x}}\eta_t$$

$$\mathsf{y}_t = f(\mathsf{x}_t) + \sigma_{\mathbf{y}}\epsilon_t$$

$$f(\mathsf{x}), g(\mathsf{x}) \sim \mathcal{GP}(0, \mathsf{K})$$

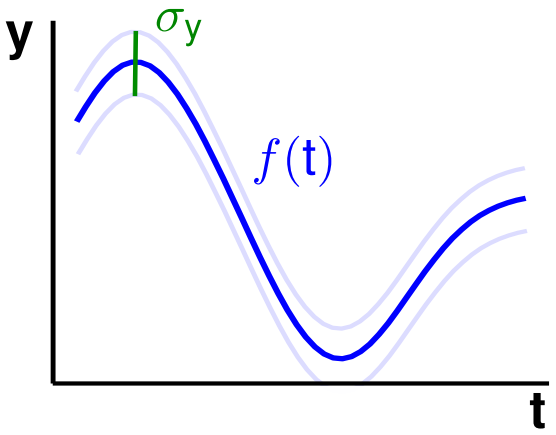
non-Gaussian distribution on $\mathsf{y}(\mathsf{t})$

GP regression: introducing notation

Generative model (like non-linear regression)

$$\mathsf{y}(\mathsf{t}) = f(\mathsf{t}) + \epsilon\sigma_{\mathsf{y}} \quad \text{(independent Gaussian noise)}$$

$$p(\epsilon) = \mathcal{N}(0, 1)$$



place GP prior over the non-linear function

$$p(f(\mathsf{t})|\theta) = \mathcal{GP}(0, \mathsf{K}(\mathsf{t}, \mathsf{t}'))$$

$$\mathsf{K}(\mathsf{t}, \mathsf{t}') = \sigma^2 \cos(\omega(\mathsf{t} - \mathsf{t}')) \exp\left(-\frac{1}{2l^2}(\mathsf{t} - \mathsf{t}')^2\right)$$

typical time-series covariance
sinusoids with SE envelopes
power in Gaussian subband

sum of two Gaussians is a Gaussian,
$\Longrightarrow$ induces GP over $\mathsf{y}(\mathsf{t})$

$$p(\mathsf{y}(\mathsf{t})|\theta) = \mathcal{GP}(0, \underbrace{\mathsf{K}(\mathsf{t}, \mathsf{t}') + \mathrm{I}\sigma_{\mathsf{y}}^2}_{\Sigma(\mathsf{t}, \mathsf{t}')})$$

How do we make predictions?
How do we learn hyper-parameters?

another popular class of GP time-series models

$$\mathsf{x}_\mathsf{t} = g(\mathsf{x}_{t-1}) + \sigma_\mathsf{x}\eta_t$$

$$\mathsf{y}_t = f(\mathsf{x}_t) + \sigma_\mathbf{y}\epsilon_t$$

$$f(\mathsf{x}), g(\mathsf{x}) \sim \mathcal{GP}(0, \mathsf{K})$$

non-Gaussian distribution on $\mathsf{y}(\mathsf{t})$

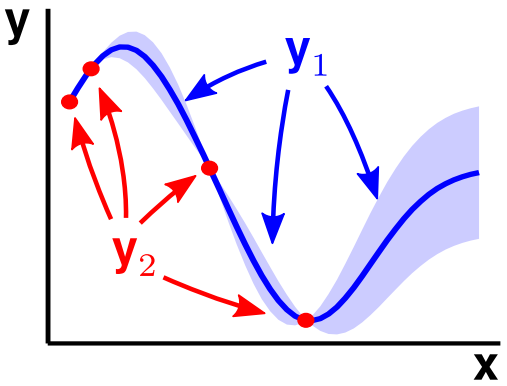
GP regression: introducing notation

How do we make predictions?

$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

$$p(\mathbf{y}_1|\mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} \qquad p(\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_2; \mathbf{0}, \Sigma_{22})$$

$$\implies p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \Sigma_{12}\Sigma_{22}^{-1}\mathbf{y}_2, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\intercal})$$

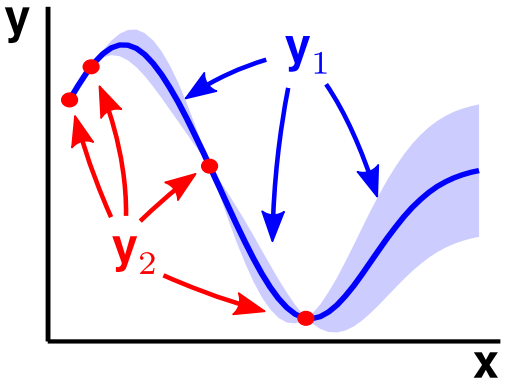


GP regression: introducing notation

How do we make predictions?

$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

$$p(\mathbf{y}_1|\mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} \qquad p(\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_2; \mathbf{0}, \Sigma_{22})$$



$$\implies p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \Sigma_{12}\Sigma_{22}^{-1}\mathbf{y}_2, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\intercal})$$

How do we learn hyper-parameters?

$$p(\theta|\mathbf{y}_{1:N}) = \frac{p(\mathbf{y}_{1:N}|\theta)p(\theta)}{p(\mathbf{y}_{1:N})} \qquad \text{(Bayes' Rule)}$$

$p(\mathbf{y}_{1:N}|\theta)$ = likelihood of the parameters

= how well did θ predict the data we observed

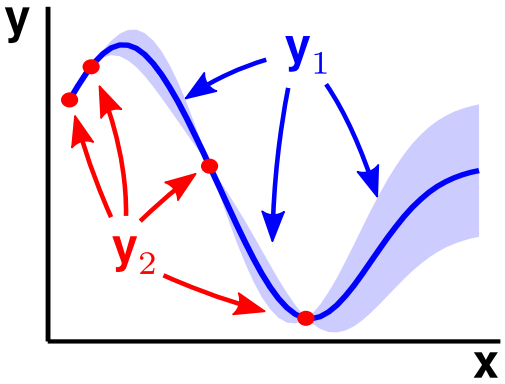
$$p(\mathbf{y}_{1:N}|\theta) = \frac{1}{\det(2\pi\Sigma(\theta))^{-1/2}} \exp\left(-\frac{1}{2}\mathbf{y}_{1:N}^{\intercal}\Sigma^{-1}(\theta)\mathbf{y}_{1:N}\right)$$

GP regression: introducing notation

How do we make predictions?

$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \end{bmatrix}, \begin{bmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{bmatrix}\right)$$

$$p(\mathbf{y}_1|\mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} \qquad p(\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_2; \mathbf{0}, \Sigma_{22})$$



$$\implies p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \Sigma_{12}\Sigma_{22}^{-1}\mathbf{y}_2, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\intercal})$$

require matrix inversion (Cholesky)

$\Rightarrow$ O(1000) datapoints

How do we learn hyper-parameters?

$$p(\theta|\mathbf{y}_{1:N}) = \frac{p(\mathbf{y}_{1:N}|\theta)p(\theta)}{p(\mathbf{y}_{1:N})} \qquad \text{(Bayes' Rule)}$$

$p(\mathbf{y}_{1:N}|\theta)$ = likelihood of the parameters

= how well did θ predict the data we observed

$$p(\mathbf{y}_{1:N}|\theta) = \frac{1}{\det(2\pi\Sigma(\theta))^{-1/2}} \exp\left(-\frac{1}{2}\mathbf{y}_{1:N}^{\intercal}\Sigma^{-1}(\theta)\mathbf{y}_{1:N}\right)$$

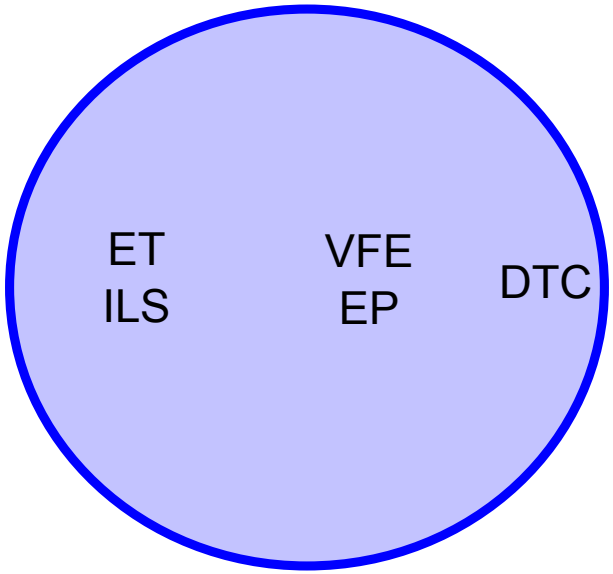
Taxonomy of Gaussian process approximations

DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”

Taxonomy of Gaussian process approximations

direct posterior approximation

$\text{div}[p(\mathbf{f}|\mathbf{y})||q(\mathbf{f})]$



DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”

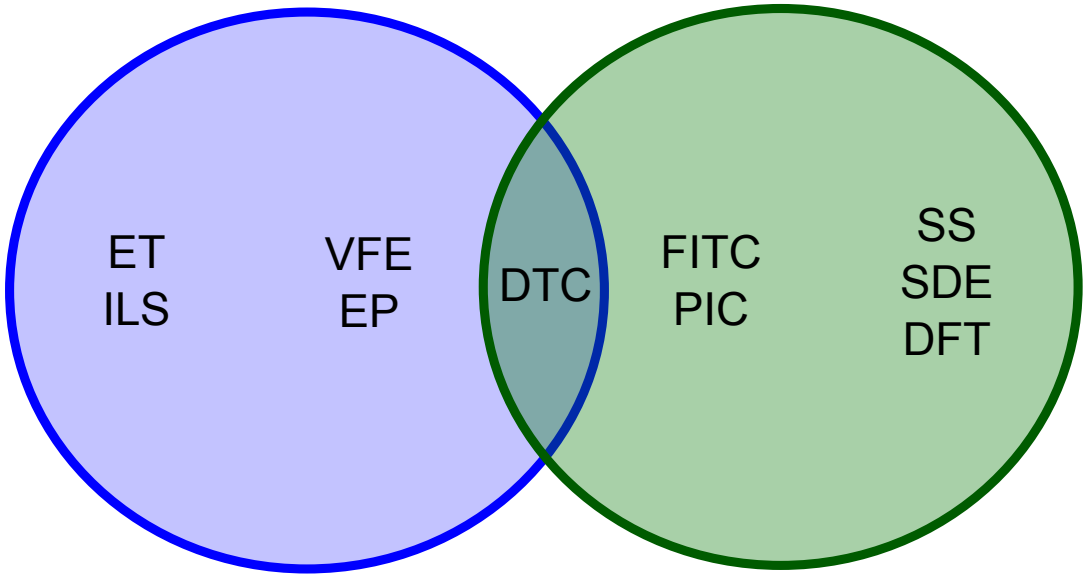
Taxonomy of Gaussian process approximations

direct posterior approximation

$\text{div}[p(\mathbf{f}|\mathbf{y})||q(\mathbf{f})]$

indirect posterior approximation

$\text{div}[p(\mathbf{f},\mathbf{y})||q(\mathbf{f},\mathbf{y})]$



DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”

Taxonomy of Gaussian process approximations

direct posterior approximation

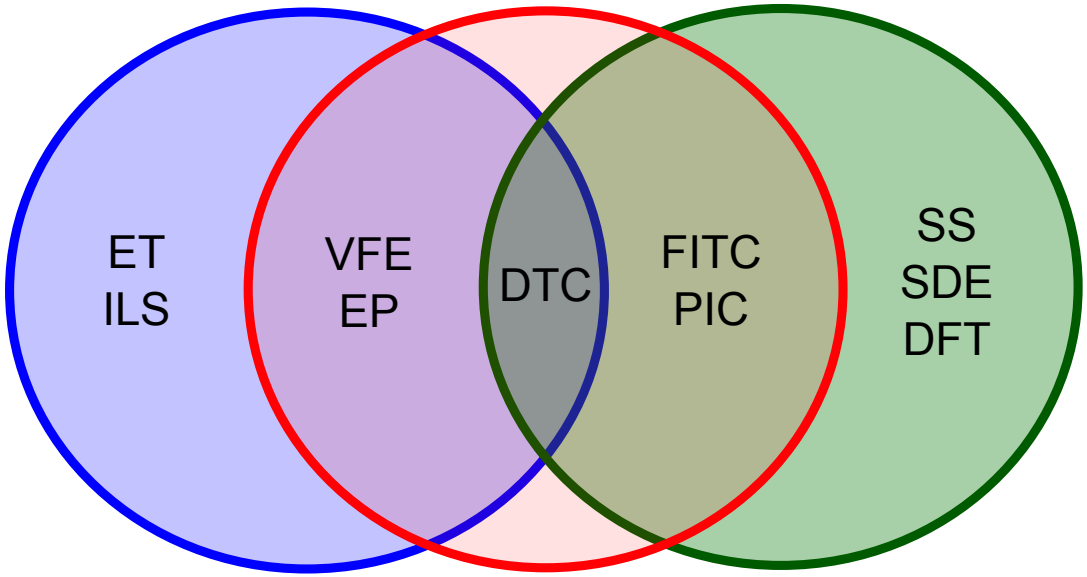
$\text{div}[p(\mathbf{f}|\mathbf{y})||q(\mathbf{f})]$

methods employing pseudo-data

$\mathbf{u}$

indirect posterior approximation

$\text{div}[p(\mathbf{f},\mathbf{y})||q(\mathbf{f},\mathbf{y})]$



DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”

Taxonomy of Gaussian process approximations

direct posterior approximation

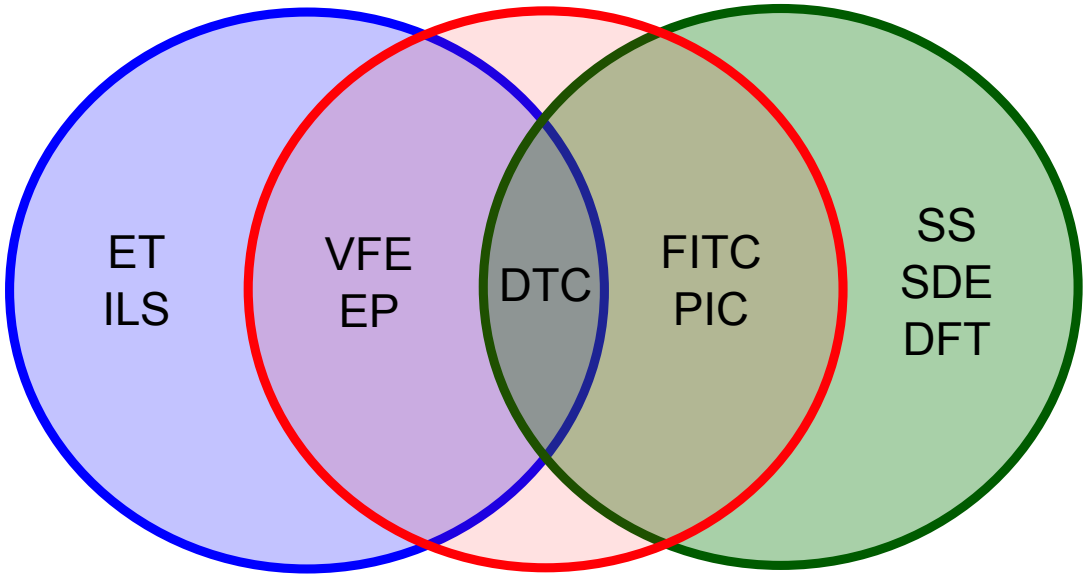
$\text{div}[p(\mathbf{f}|\mathbf{y})||q(\mathbf{f})]$

methods employing pseudo-data

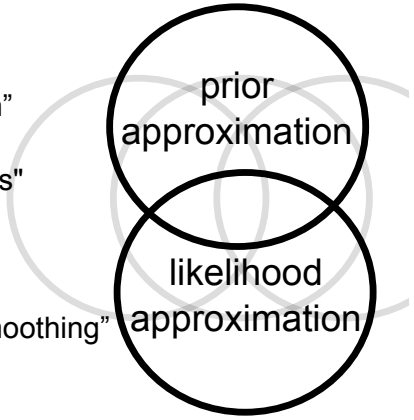
$\mathbf{u}$

indirect posterior approximation

$\text{div}[p(\mathbf{f},\mathbf{y})||q(\mathbf{f},\mathbf{y})]$



DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”



Taxonomy of Gaussian process approximations

direct posterior approximation

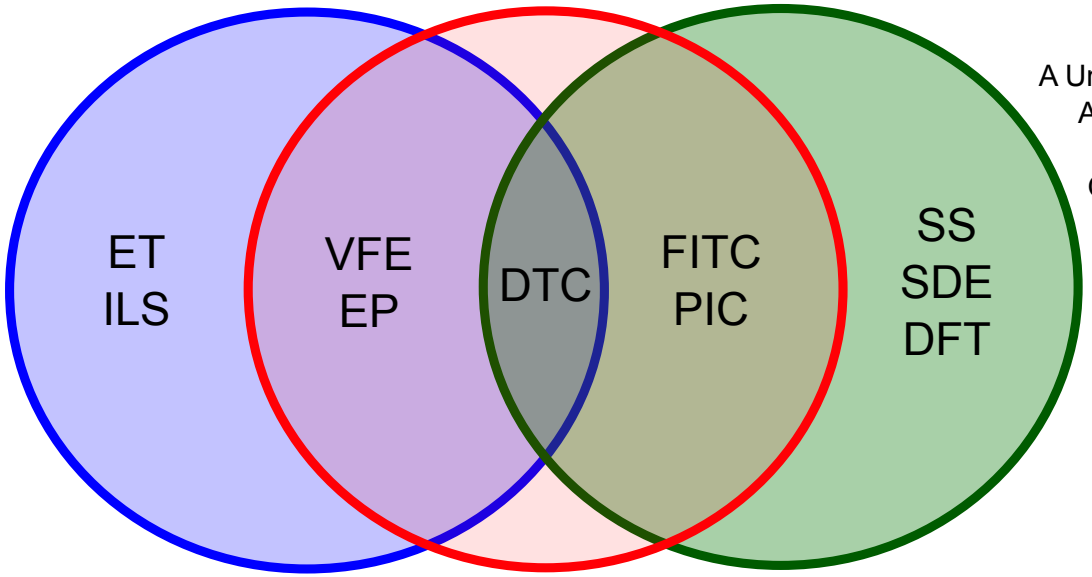
$\text{div}[p(\mathbf{f}|\mathbf{y})||q(\mathbf{f})]$

methods employing pseudo-data

$\mathbf{u}$

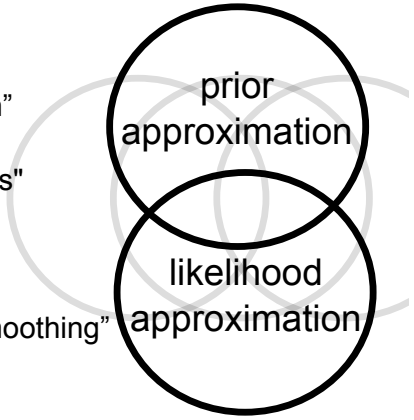
indirect posterior approximation

$\text{div}[p(\mathbf{f},\mathbf{y})||q(\mathbf{f},\mathbf{y})]$

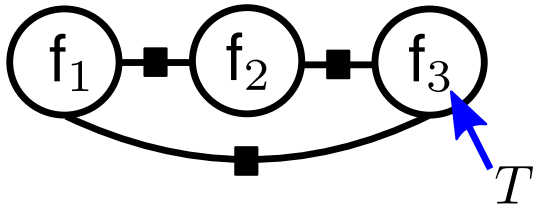


A Unifying View of Sparse Approximate Gaussian Process Regression Quinonero-Candela & Rasmussen, 2005

DFT: Turner."Statistical models for natural sounds"
DTC: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”
EP: Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
ET: Sudderth et al. "Embedded Trees: Estimation of Gaussian Processes on Graphs with Cycles"
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
ISL: Gibbs et al. "Efficient implementation of Gaussian processes"
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
SDE: Sarkka et al. “Spatiotemporal Learning via Infinite Dimensional Bayesian Filtering and Smoothing”
SS: Lazaro-Gredilla et al. “Sparse spectrum Gaussian process regression”.
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”



Fully independent training conditional (FITC) approximation

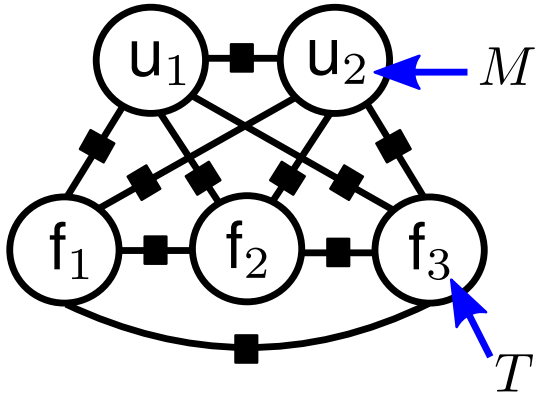


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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

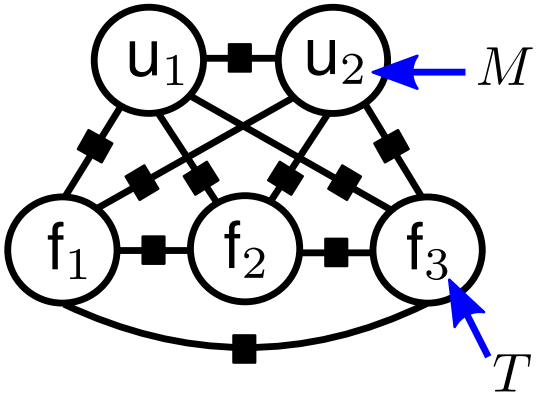


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

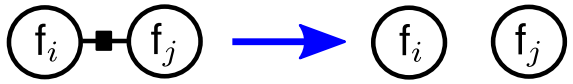
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



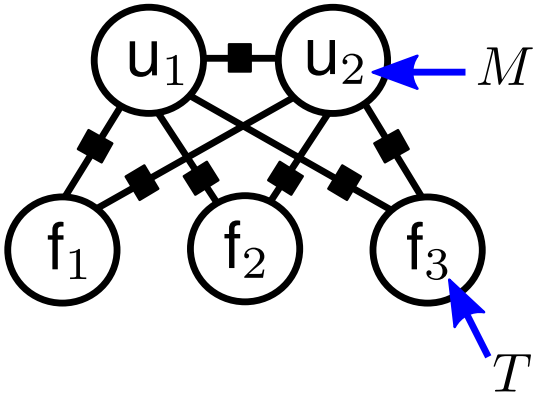
all factors

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

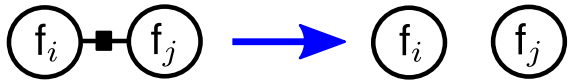
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



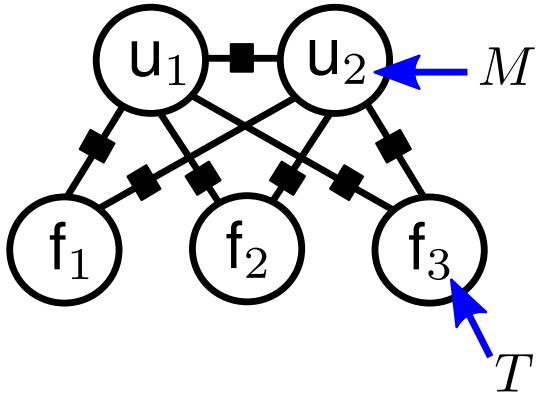
all factors

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

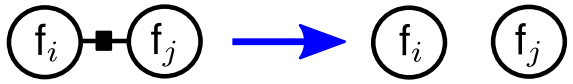
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



all factors

3. calibrate model
   (e.g. using KL divergence, many choices)

$$\underset{q(\mathbf{u}),\{q(\mathbf{f}_t|\mathbf{u})\}_{t=1}^{T}}{\arg\min}\ \mathrm{KL}(p(\mathbf{f},\mathbf{u})||q(\mathbf{u})\prod_{t=1}^{T}q(\mathbf{f}_t|\mathbf{u})) \implies \begin{aligned} q(\mathbf{u}) &= p(\mathbf{u})\\ q(\mathbf{f}_t|\mathbf{u}) &= p(\mathbf{f}_t|\mathbf{u})\end{aligned}$$

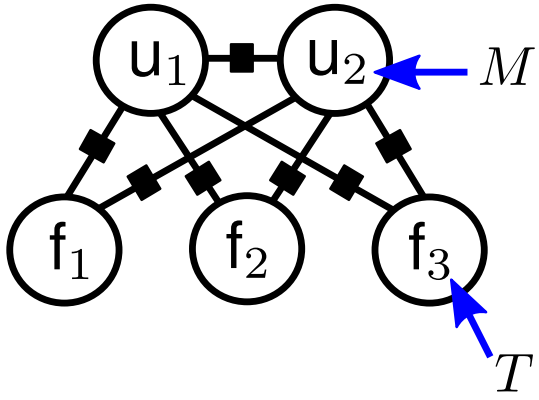
equal to exact conditionals

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

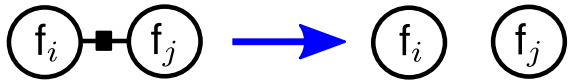
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



all factors

3. calibrate model
   (e.g. using KL divergence, many choices)

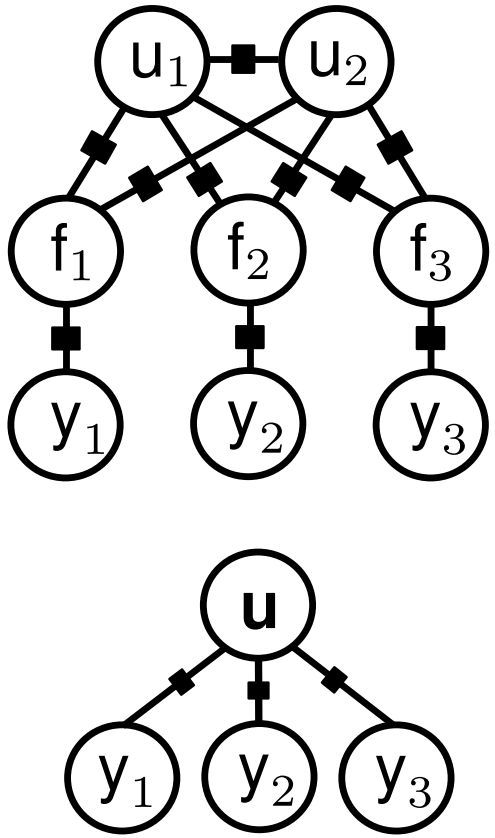
$$\underset{q(\mathbf{u}),\{q(\mathbf{f}_t|\mathbf{u})\}_{t=1}^{T}}{\arg\min} \mathrm{KL}(p(\mathbf{f},\mathbf{u})||q(\mathbf{u})\prod_{t=1}^{T} q(\mathbf{f}_t|\mathbf{u})) \implies \begin{aligned} q(\mathbf{u}) &= p(\mathbf{u}) \\ q(\mathbf{f}_t|\mathbf{u}) &= p(\mathbf{f}_t|\mathbf{u}) \end{aligned}$$

equal to exact conditionals

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

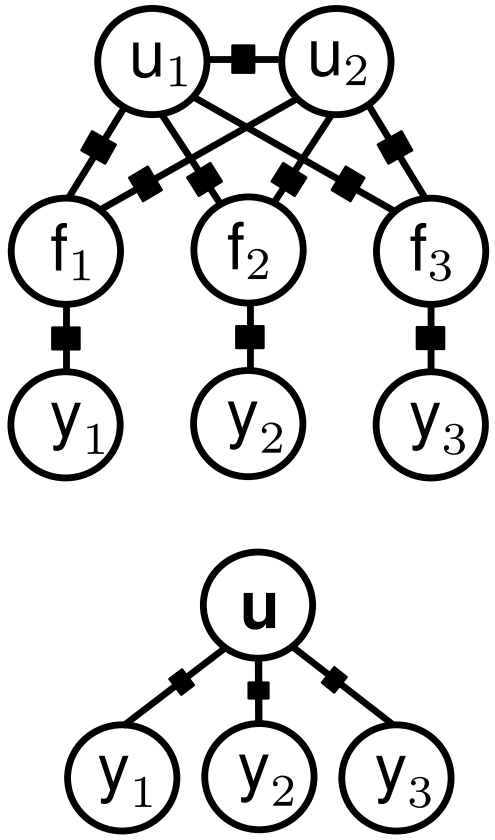


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$



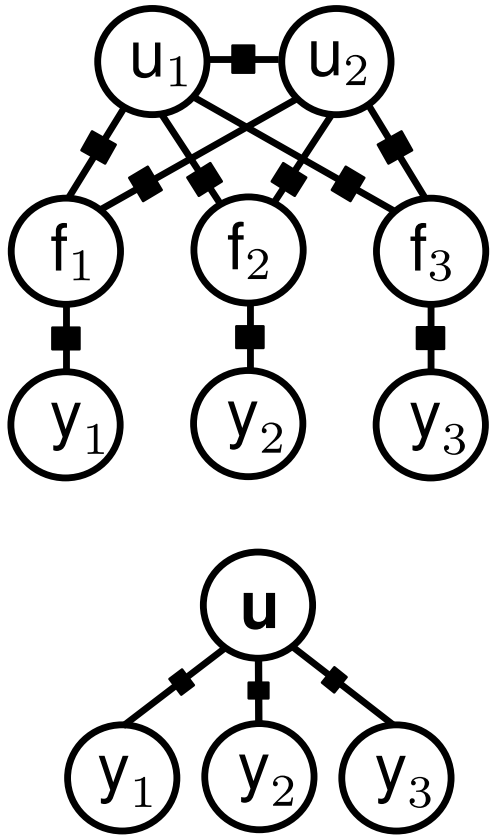
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$



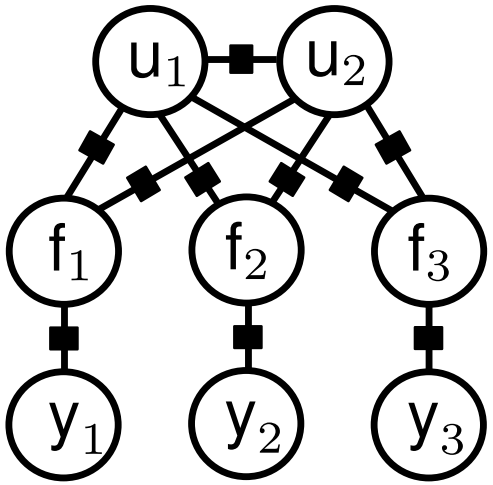
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

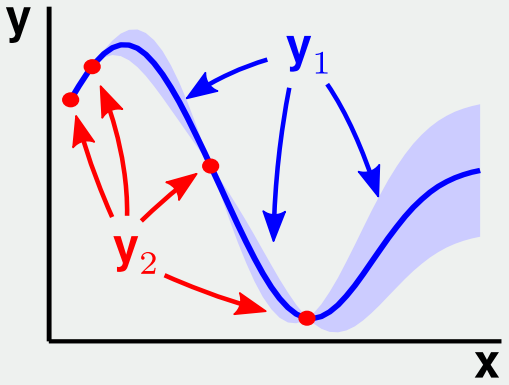
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

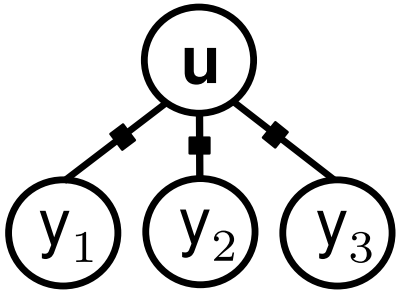
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$



How do we make predictions?

$$p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \Sigma_{12}\Sigma_{22}^{-1}\mathbf{y}_2, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\mathsf{T}})$$





construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

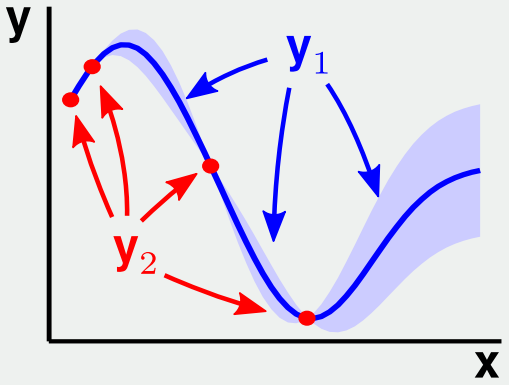
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

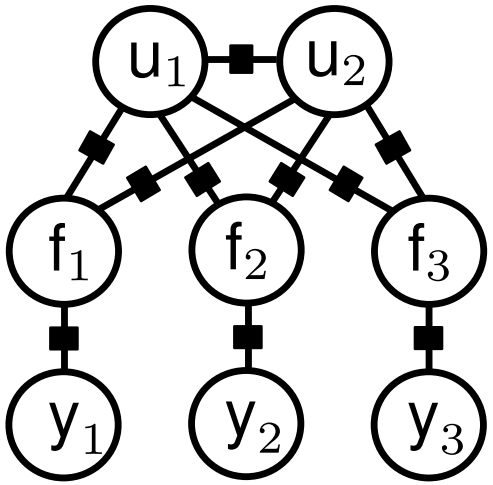
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

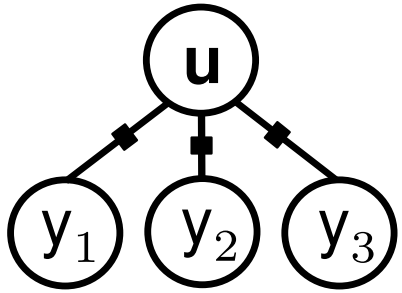
$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t})$$

How do we make predictions?

$$p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1; \Sigma_{12}\Sigma_{22}^{-1}\mathbf{y}_2, \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{12}^{\mathsf{T}})$$







construct new generative model (with pseudo-data)

cheaper to perform exact learning and inference

calibrated to original

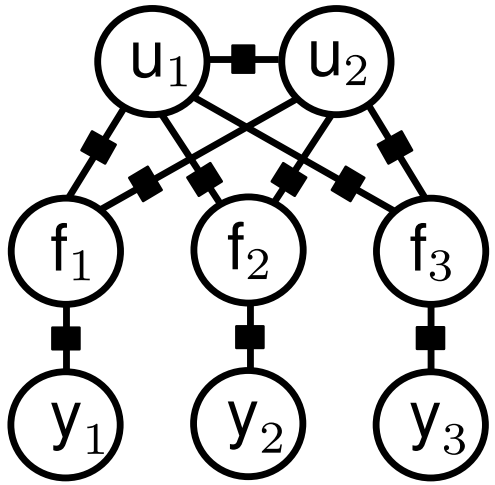
indirect posterior approximation

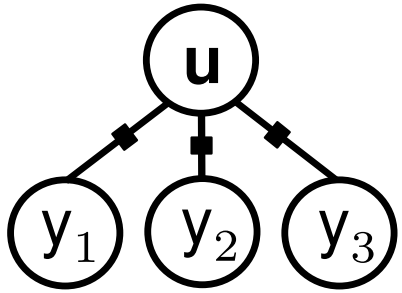
Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{u}\mathsf{f}_t})$$





construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

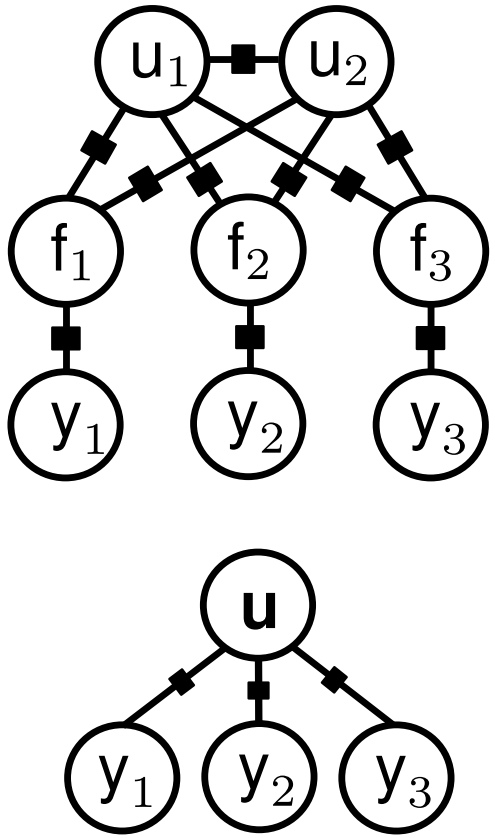
indirect posterior approximation

Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$



construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

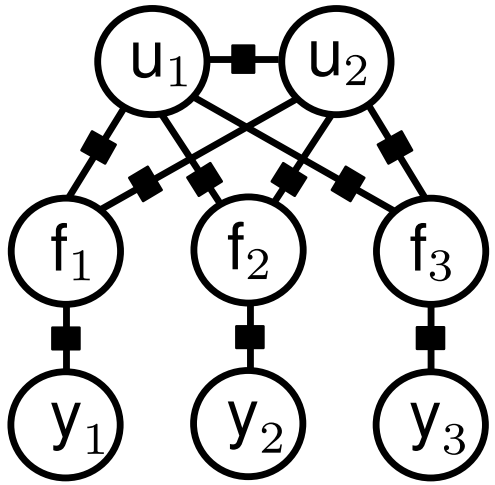
Fully independent training conditional (FITC) approximation

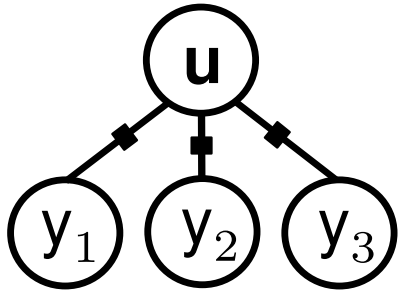
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$





construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

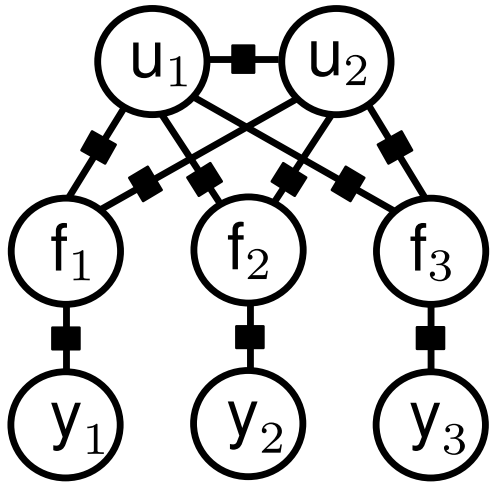
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

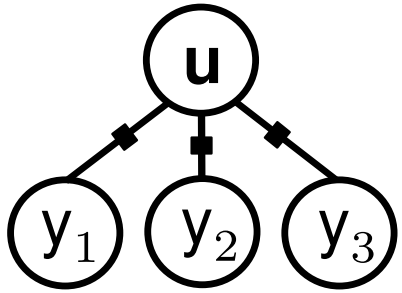
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$

cost of computing likelihood is $\mathcal{O}(TM^2)$





construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

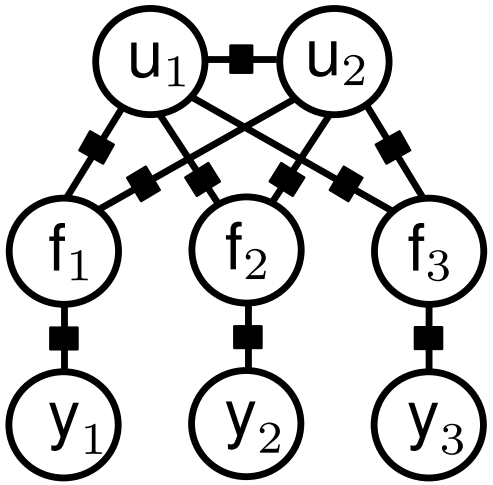
Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

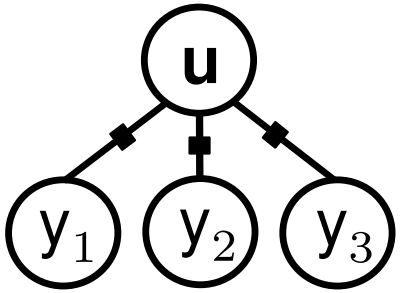
$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$



cost of computing likelihood is $\mathcal{O}(TM^2)$

$$p(\mathbf{y}_t|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$



construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

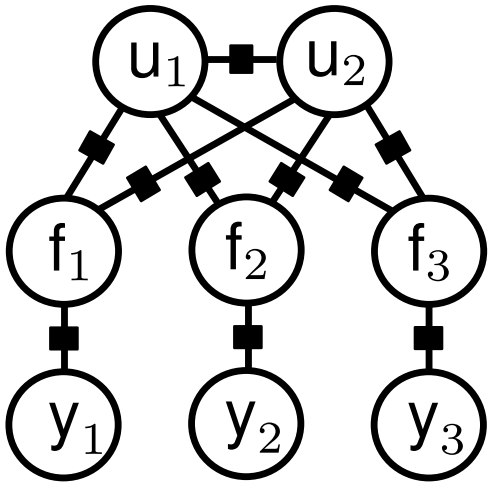
Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

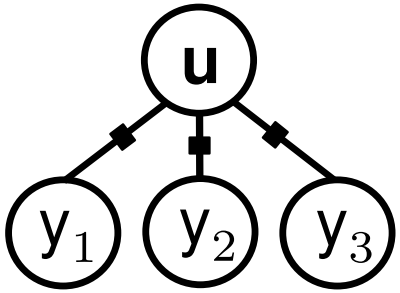
$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$



cost of computing likelihood is $\mathcal{O}(TM^2)$

$$p(\mathbf{y}_t|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

$$= \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$



construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

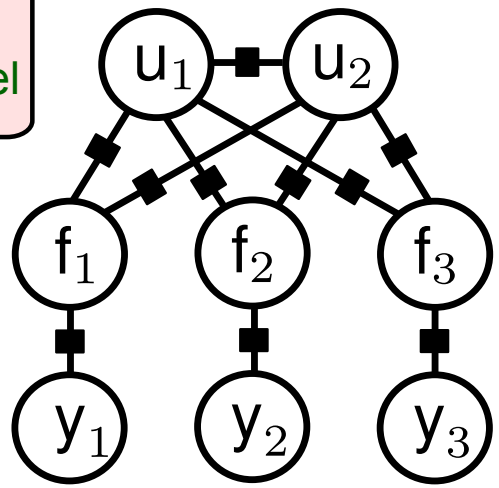
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

Joaquin's talk: finite linear model

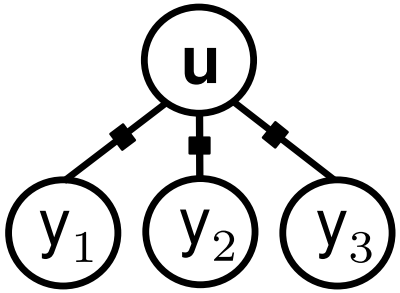
$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$



cost of computing likelihood is $\mathcal{O}(TM^2)$

$$p(\mathbf{y}_t|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

$$= \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$



construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

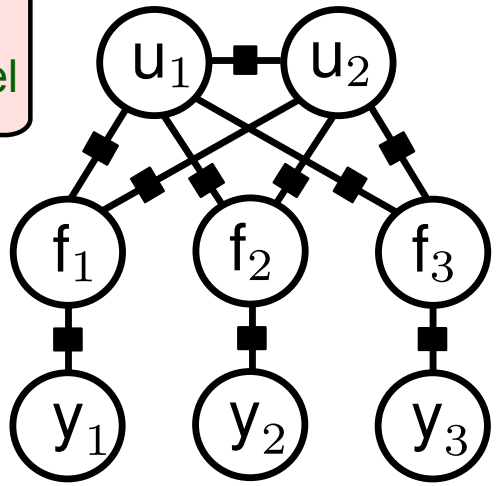
Fully independent training conditional (FITC) approximation

$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

Joaquin's talk: finite linear model

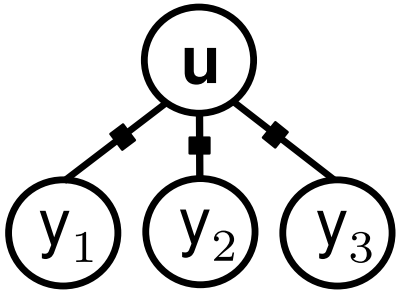


$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$

cost of computing likelihood is $\mathcal{O}(TM^2)$

$$p(\mathbf{y}_t|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

$$= \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$



original variances along diagonal: stops variances collapsing

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

- parametric (although cleverly so)
- if I see more data, should I add extra pseudo-data?

Fully independent training conditional (FITC) approximation

- parametric (although cleverly so)
- if I see more data, should I add extra pseudo-data?
  - unnatural from a generative modelling perspective
  - natural from a prediction perspective (need greater complexity/bandwidth)

$\Longrightarrow$ **lost elegant separation of model, inference and approximation**

Fully independent training conditional (FITC) approximation

- parametric (although cleverly so)
- if I see more data, should I add extra pseudo-data?
  - unnatural from a generative modelling perspective
  - natural from a prediction perspective (need greater complexity/bandwidth)

$\Longrightarrow$ **lost elegant separation of model, inference and approximation**

- example of **prior approximation**

Fully independent training conditional (FITC) approximation

- parametric (although cleverly so)
- if I see more data, should I add extra pseudo-data?
  - unnatural from a generative modelling perspective
  - natural from a prediction perspective (need greater complexity/bandwidth)

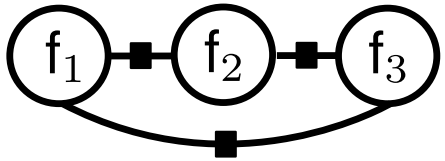
$\Longrightarrow$ **lost elegant separation of model, inference and approximation**

- example of **prior approximation**

Extensions:

- methods for optimising pseudo-inputs (indirect approximations tend to over-fit)
- partially independent training conditional...

Partially independent training conditional (PITC) approx.



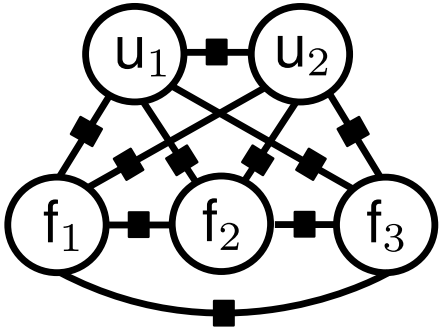
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Partially independent training conditional (PITC) approx.

1. augment model with M<T pseudo data

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



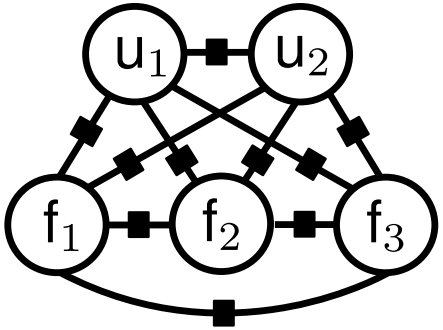
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

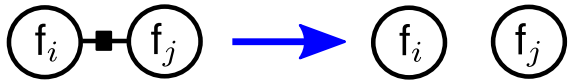
Partially independent training conditional (PITC) approx.

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



between blocks

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

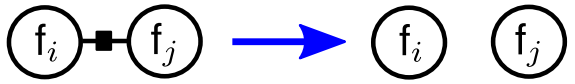
indirect posterior approximation

Partially independent training conditional (PITC) approx.

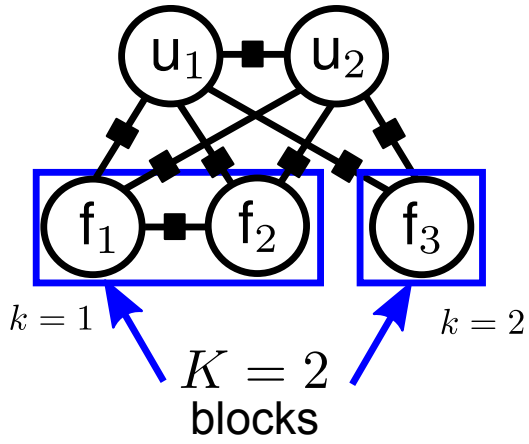
1. augment model with M<T pseudo data

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

2. remove some of the dependencies
   (results in simpler model)



between blocks



$$\mathbf{f}_1 = \begin{bmatrix} \mathsf{f}_1 \\ \mathsf{f}_2 \end{bmatrix} \quad \mathbf{f}_2 = \mathsf{f}_3$$

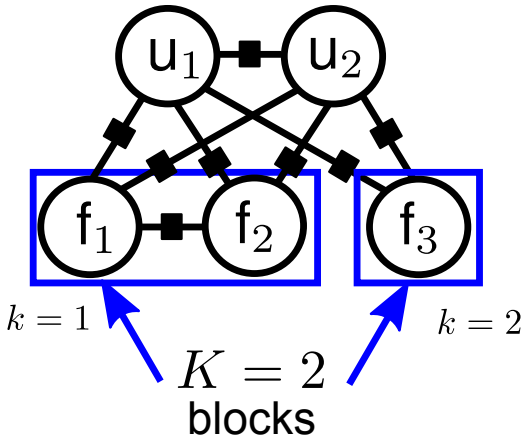
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

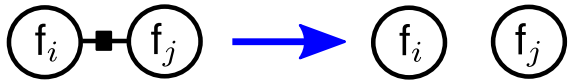
Partially independent training conditional (PITC) approx.

1. augment model with M<T pseudo data

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



2. remove some of the dependencies
   (results in simpler model)



between blocks

$$\mathbf{f}_1 = \begin{bmatrix}\mathbf{f}_1\\ \mathbf{f}_2\end{bmatrix} \quad \mathbf{f}_2 = \mathbf{f}_3$$

3. calibrate model
   (e.g. using KL divergence, many choices)

$$\underset{q(\mathbf{u}),\{q(\mathbf{f}_k|\mathbf{u})\}_{k=1}^K}{\arg\min} \mathrm{KL}(p(\mathbf{f},\mathbf{u})||q(\mathbf{u})\prod_{k=1}^K q(\mathbf{f}_k|\mathbf{u})) \Longrightarrow \begin{aligned} q(\mathbf{u}) &= p(\mathbf{u}) \\ q(\mathbf{f}_k|\mathbf{u}) &= p(\mathbf{f}_k|\mathbf{u}) \end{aligned}$$

equal to exact conditionals

construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Deterministic training conditional (DTC)

- minimise variational KL between two posterior distributions
  - direct posterior approximation
  - likelihood approximation

$$\arg\min_{q(\mathbf{y}|\mathbf{u})} \mathrm{KL}\left(\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})q(\mathbf{y}|\mathbf{u})||\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})p(\mathbf{y}|\mathbf{f})\right) \quad \text{such that} \quad \int \mathrm{d}\mathbf{y}\; q(\mathbf{y}|\mathbf{u}) = 1$$

Deterministic training conditional (DTC)

- minimise variational KL between two posterior distributions
  - direct posterior approximation
  - likelihood approximation

$$\arg\min_{q(\mathbf{y}|\mathbf{u})} \mathrm{KL}\left(\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})q(\mathbf{y}|\mathbf{u})||\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})p(\mathbf{y}|\mathbf{f})\right) \quad \text{such that} \quad \int \mathrm{d}\mathbf{y}\; q(\mathbf{y}|\mathbf{u}) = 1$$

$$\implies q(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I})$$

Deterministic training conditional (DTC)

- minimise variational KL between two posterior distributions
  - direct posterior approximation
  - likelihood approximation

$$\arg\min_{q(\mathbf{y}|\mathbf{u})} \mathrm{KL}\left(\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})q(\mathbf{y}|\mathbf{u})||\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})p(\mathbf{y}|\mathbf{f})\right) \quad \text{such that} \quad \int \mathrm{d}\mathbf{y}\; q(\mathbf{y}|\mathbf{u}) = 1$$

$$\implies q(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I})$$

can this approximation be categorised in another way?

Deterministic training conditional (DTC)

- minimise variational KL between two posterior distributions
  - direct posterior approximation
  - likelihood approximation

$$\arg\min_{q(\mathbf{y}|\mathbf{u})} \mathrm{KL}\left(\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})q(\mathbf{y}|\mathbf{u}) || \frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})p(\mathbf{y}|\mathbf{f})\right) \quad \text{such that} \quad \int \mathrm{d}\mathbf{y}\; q(\mathbf{y}|\mathbf{u}) = 1$$

$$\implies q(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I})$$

- probabilistic PCA to FITC's factor analysis

$$\text{DTC}: \; p(\mathbf{y}|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

$$\text{FITC}: \; p(\mathbf{y}|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

Deterministic training conditional (DTC)

- minimise variational KL between two posterior distributions
  - direct posterior approximation
  - likelihood approximation

$$\arg\min_{q(\mathbf{y}|\mathbf{u})} \mathrm{KL}\left(\frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})q(\mathbf{y}|\mathbf{u}) || \frac{1}{Z}p(\mathbf{u})p(\mathbf{f}|\mathbf{u})p(\mathbf{y}|\mathbf{f})\right) \quad \text{such that} \quad \int \mathrm{d}\mathbf{y}\; q(\mathbf{y}|\mathbf{u}) = 1$$

$$\implies q(\mathbf{y}|\mathbf{u}) = \mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I})$$

- probabilistic PCA to FITC's factor analysis

$$\text{DTC}: \; p(\mathbf{y}|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

$$\text{FITC}: \; p(\mathbf{y}|\theta) = \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}} + \mathsf{D} + \sigma_{\mathsf{y}}^2\mathsf{I})$$

- blurred division between direct/indirect and likelihood/prior approximation

Variational free-energy method (VFE)

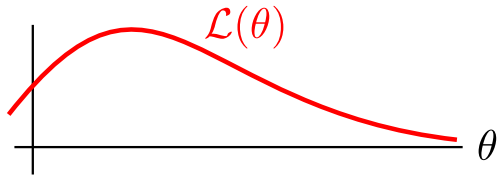
augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood

$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta)$

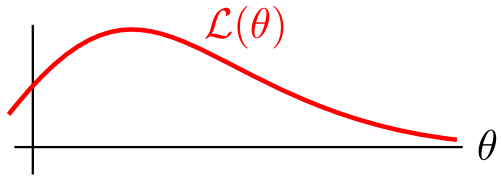


Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood

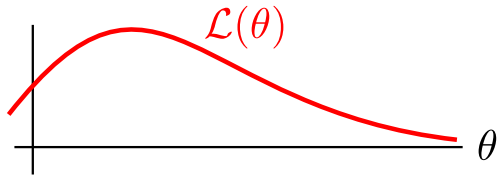
$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$



Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



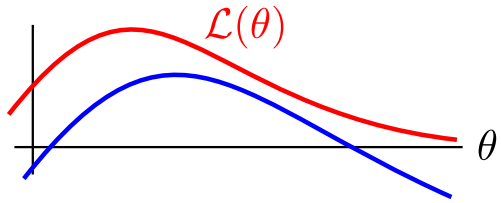
$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})}$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



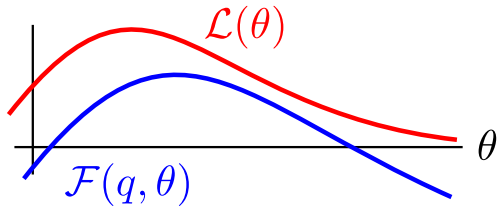
$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\ \mathbf{du}\ q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})}$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



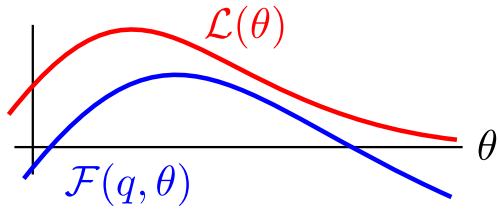
$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\ \mathbf{du}\ p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\ \mathbf{du}\ q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

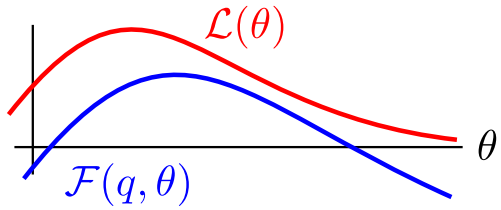
assume approximate posterior factorisation with special form

$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

assume approximate posterior factorisation with special form

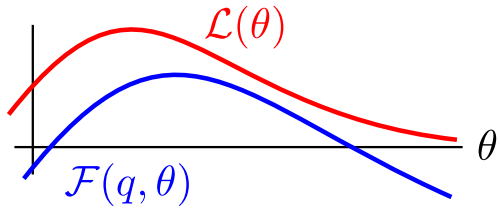
$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

$$\mathcal{F}(q, \theta) = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})} = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\mathbf{u})p(\mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})}$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

assume approximate posterior factorisation with special form

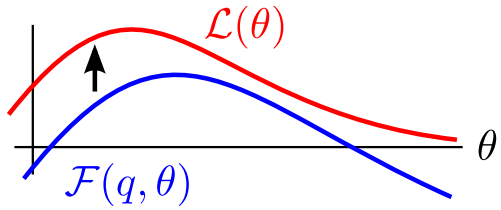
$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

$$\mathcal{F}(q, \theta) = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})} = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})\cancel{p(\mathbf{f}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(\mathbf{f}|\mathbf{u})}q(\mathbf{u})}$$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

assume approximate posterior factorisation with special form

$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

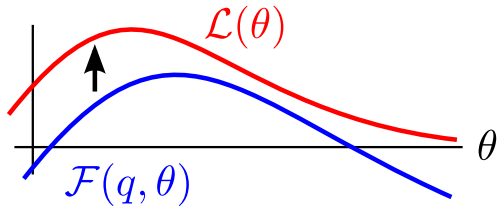
$$\mathcal{F}(q, \theta) = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})} = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})\cancel{p(\mathbf{f}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(\mathbf{f}|\mathbf{u})}q(\mathbf{u})}$$

make bound as tight as possible: $q^*(\mathbf{u}) = \arg\max_{q(\mathbf{u})} \mathcal{F}(q, \theta)$

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

assume approximate posterior factorisation with special form

$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

$$\mathcal{F}(q, \theta) = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})} = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})\cancel{p(\mathbf{f}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(\mathbf{f}|\mathbf{u})}q(\mathbf{u})}$$

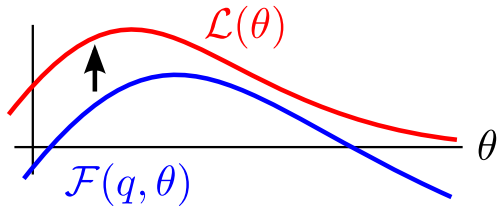
make bound as tight as possible: $q^*(\mathbf{u}) = \arg\max_{q(\mathbf{u})} \mathcal{F}(q, \theta)$

$q^*(\mathbf{u}) \propto p(\mathbf{u})\mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I})$ (DTC)

Variational free-energy method (VFE)

augment the model with pseudo-data: $p(\mathbf{y}, \mathbf{f}, \mathbf{u}|\theta) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, \mathbf{u})$

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u})$$

$$= \log \int \mathbf{df}\, \mathbf{du}\; p(\mathbf{y}, \mathbf{f}, \mathbf{u}) \frac{q(\mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} \geq \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{q(\mathbf{f}, \mathbf{u})} = \mathcal{F}(q, \theta)$$

assume approximate posterior factorisation with special form

$$q(\mathbf{f}, \mathbf{u}) = q(\mathbf{f}|\mathbf{u})q(\mathbf{u}) = p(\mathbf{f}|\mathbf{u})q(\mathbf{u}) \qquad \text{(exact } q(\mathbf{f}|\mathbf{u}) = p(\mathbf{f}|\mathbf{y})\text{)}$$

$$\mathcal{F}(q, \theta) = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}, \mathbf{f}, \mathbf{u})}{p(\mathbf{f}|\mathbf{u})q(\mathbf{u})} = \int \mathbf{df}\, \mathbf{du}\; q(\mathbf{f}, \mathbf{u}) \log \frac{p(\mathbf{y}|\mathbf{f})\cancel{p(\mathbf{f}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(\mathbf{f}|\mathbf{u})}q(\mathbf{u})}$$

make bound as tight as possible: $q^*(\mathbf{u}) = \arg\max_{q(\mathbf{u})} \mathcal{F}(q, \theta)$

$$q^*(\mathbf{u}) \propto p(\mathbf{u})\mathcal{N}(\mathbf{y}; \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \sigma_{\mathsf{y}}^2\mathsf{I}) \quad \text{(DTC)}$$

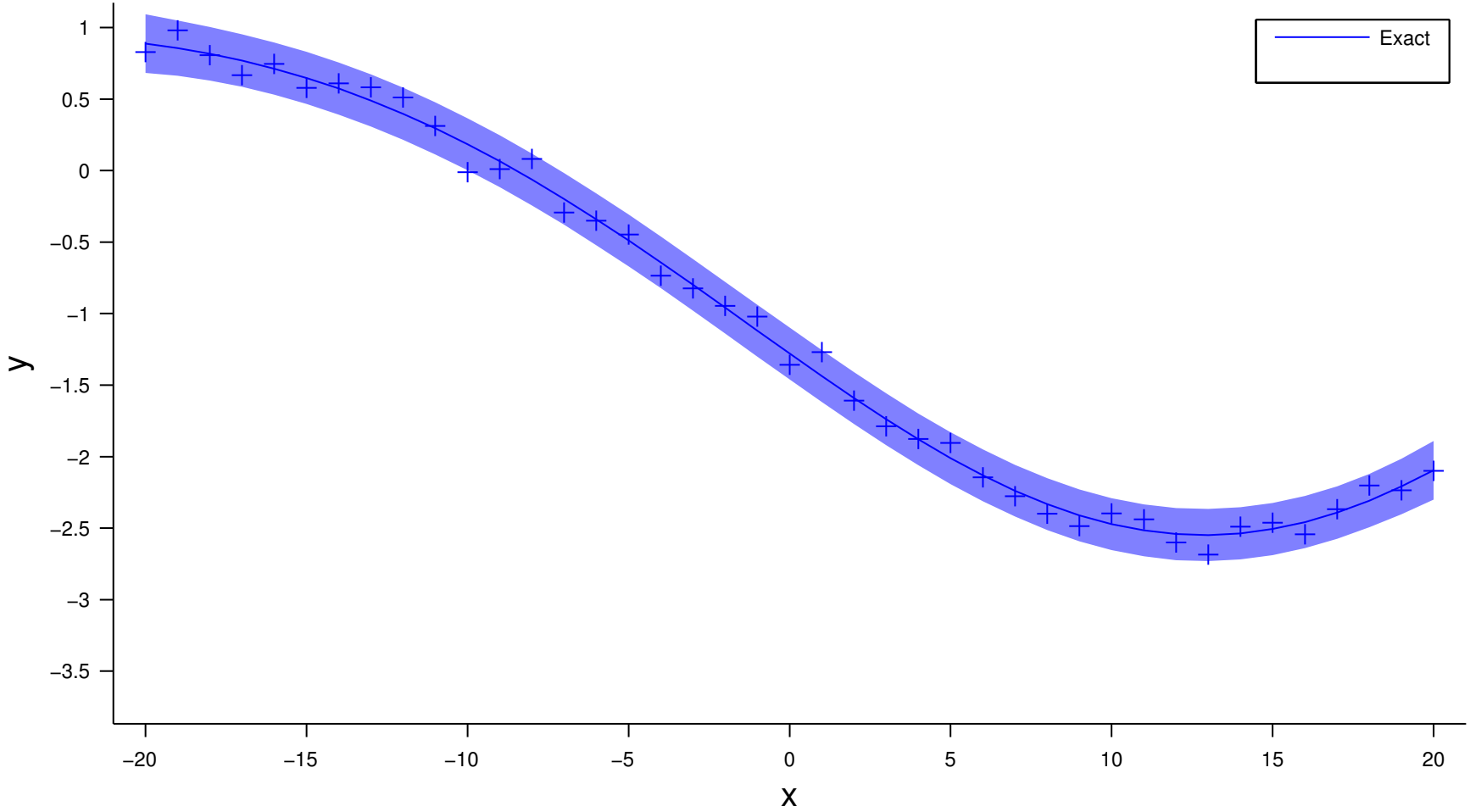
$$\mathcal{F}(q^*, \theta) = \log \mathcal{N}(\mathbf{y}; \mathbf{0}, \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}}, \sigma_{\mathsf{y}}^2\mathsf{I}) - \frac{1}{2\sigma_{\mathsf{y}}^2}\mathsf{trace}(\mathsf{K}_{\mathsf{ff}} - \mathsf{K}_{\mathsf{fu}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}})$$

DTC like — uncertainty based correction

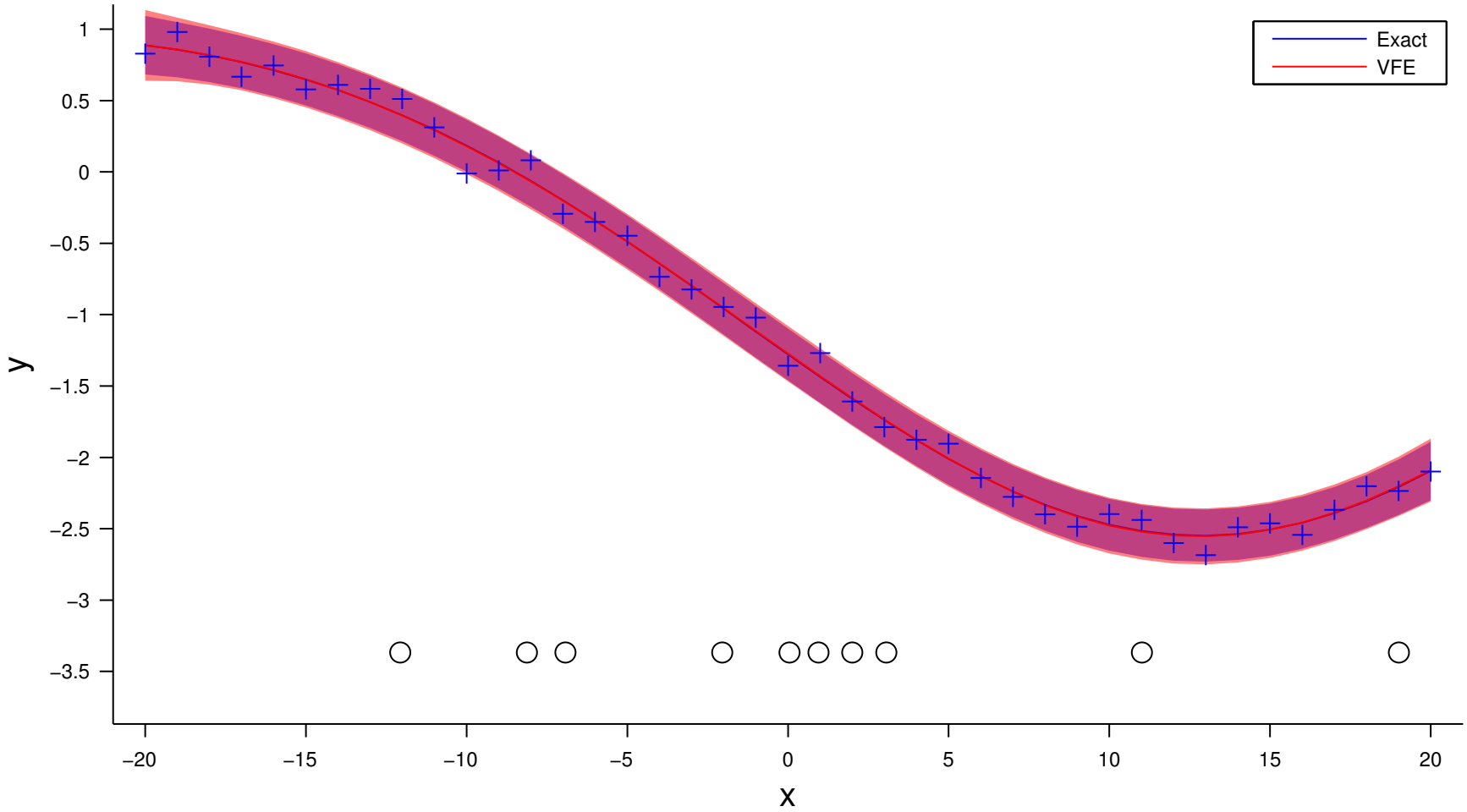
Summary of DTC/VFE methods

- optimisation pseudo point inputs **much less prone to over-fitting** in VFE methods (direct posterior approximation)
- variational methods known to **underfit** (and have othe **biases**)
- like FITC, VFE added **u** to **generative model** – not merely parameters of the approximation
  - approximation will 'waste' some representational power capturing the distribution over the pseudo-data
  - coherrent way of adding pseudo-data in light of more data?
  - perhaps a more pure approach uses pseudo-data exclusively to parameterise the posterior: $q(\mathbf{f}) = \frac{1}{Z} p(\mathbf{f}) p(\tilde{\mathbf{y}}|\mathbf{f})$ (e.g. Qi et al.)
- **how do these methods perform for time-series data-sets?**

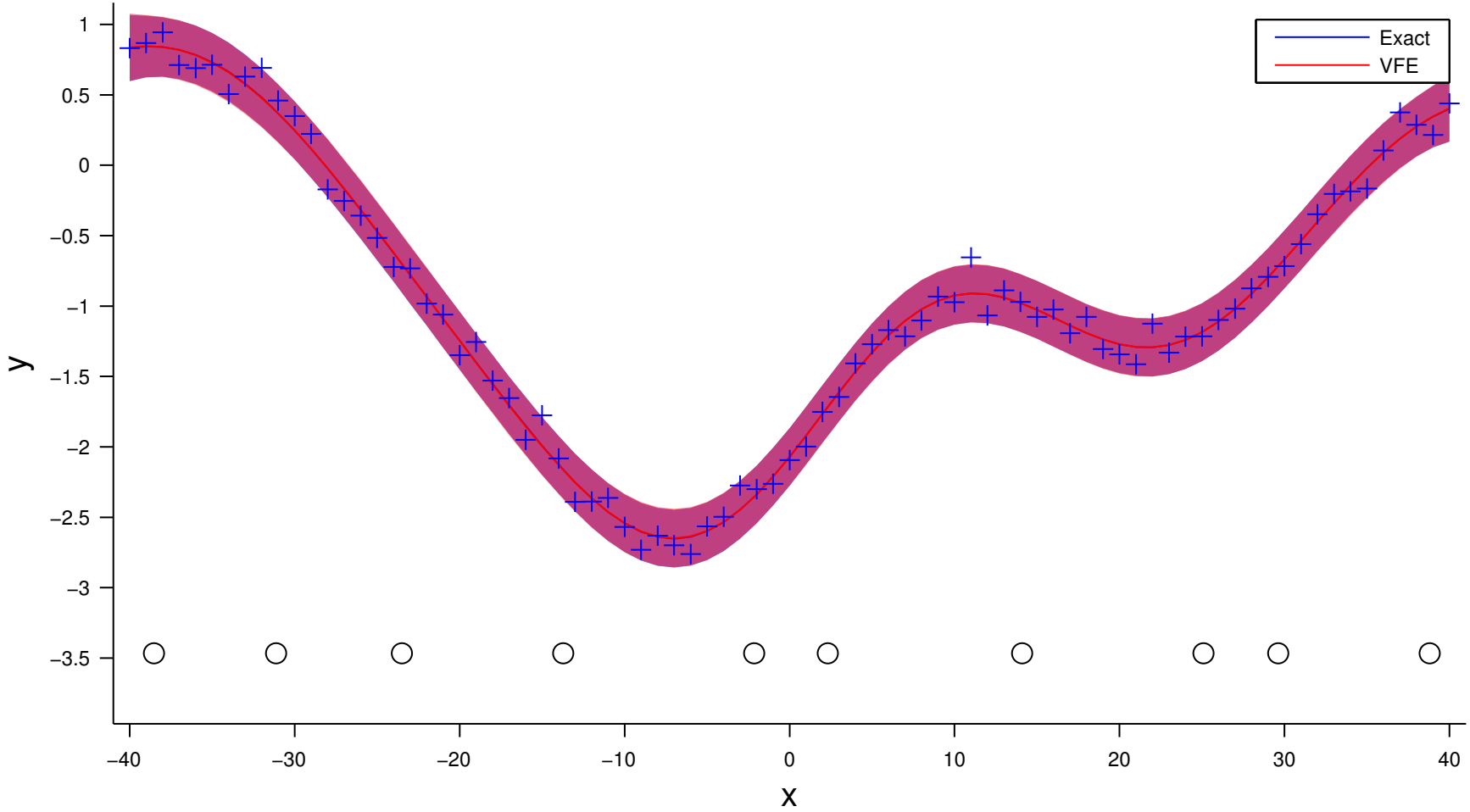
How well do current pseudo data-point approximation methods perform on time-series?



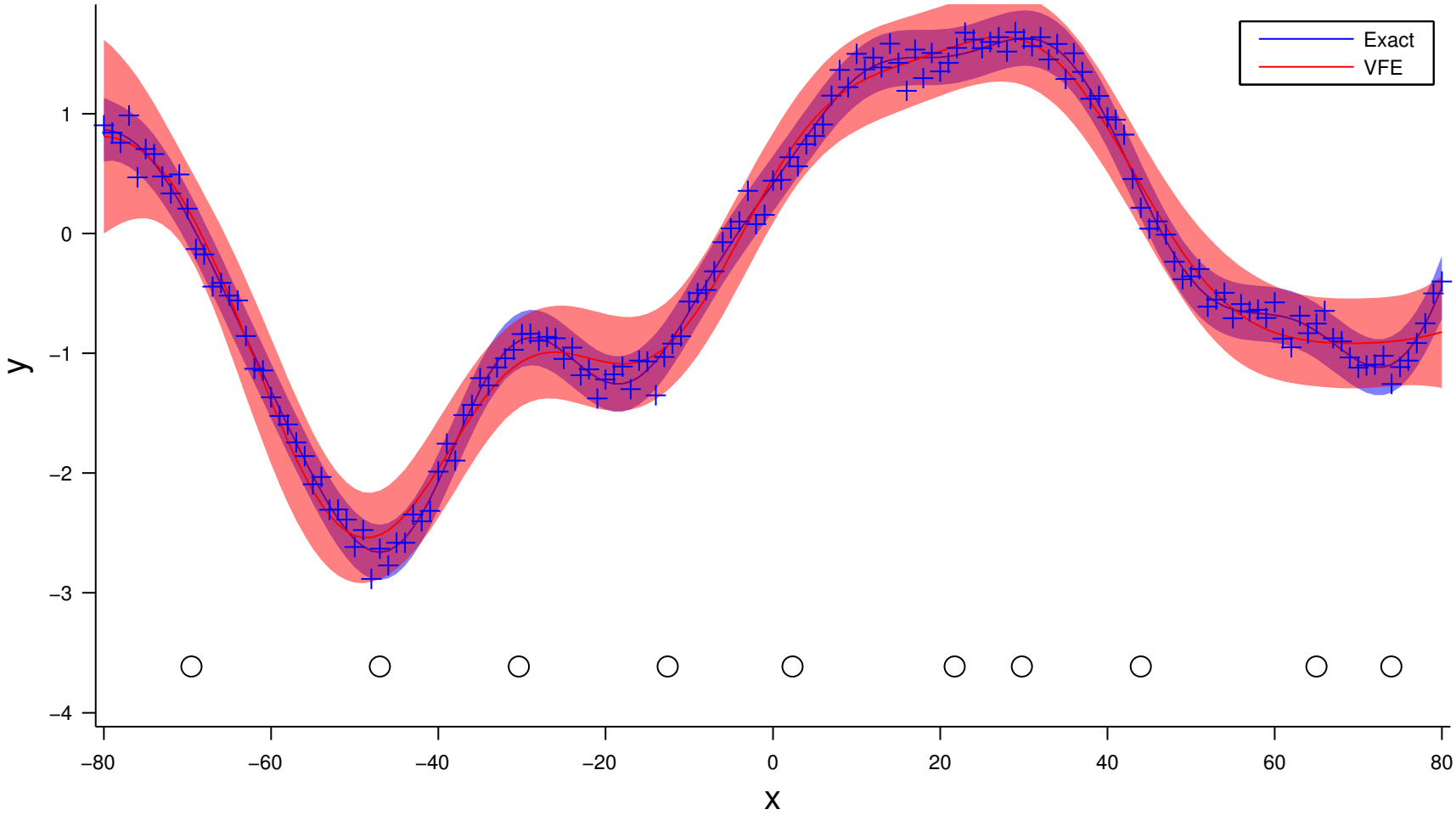
How well do current pseudo data-point approximation methods perform on time-series?



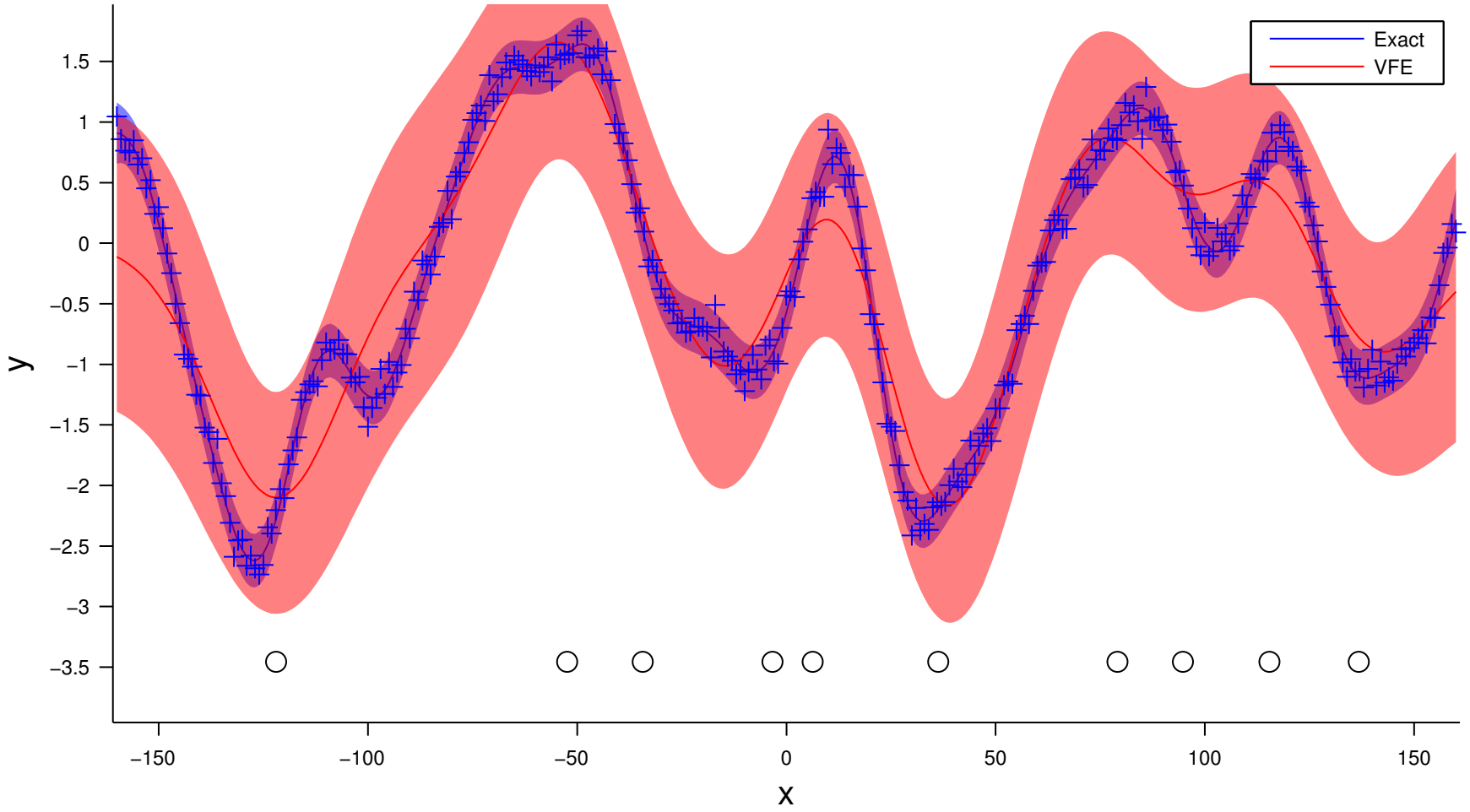
How well do current pseudo data-point approximation methods perform on time-series?



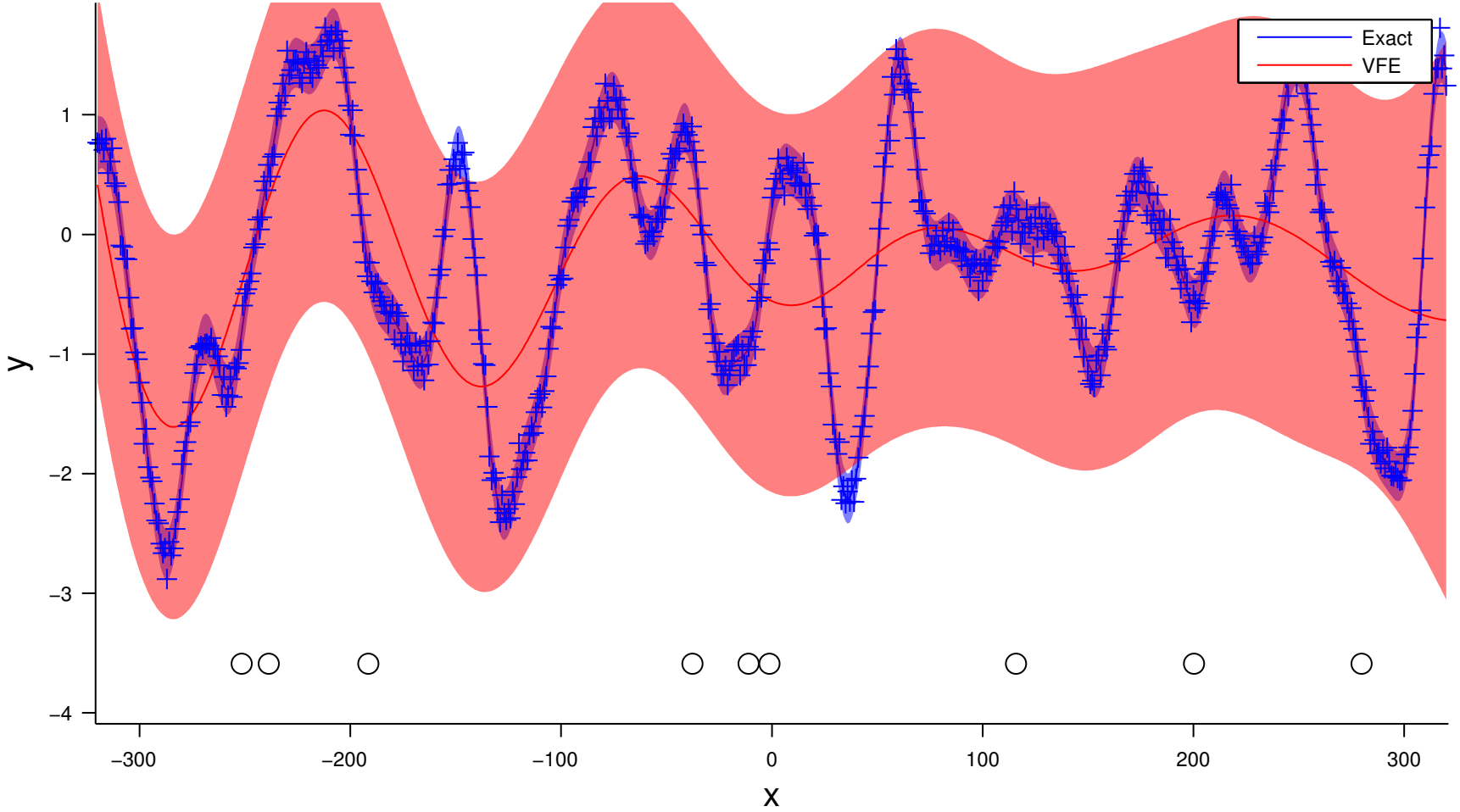
How well do current pseudo data-point approximation methods perform on time-series?



How well do current pseudo data-point approximation methods perform on time-series?



How well do current pseudo data-point approximation methods perform on time-series?



How well do current pseudo data-point approximation methods perform on time-series?

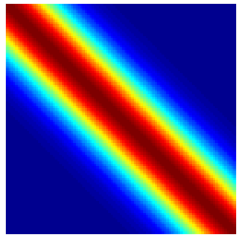
- effect of each pseudo-datapoint is **local**
- but computations involving them are **global**
- back of the envelope calculation:
  - M = number pseudo-points, T = data-range
  - l = range of the (shortest) dependencies in the posterior
  - $M \geq \frac{T}{l}$
  - e.g. audio $l = 10$, $T = 10^5 \implies M \geq 10^4$

$\implies$ require large numbers of pseudo-points, but methods are $\mathcal{O}(NM^2)$

$\implies$ **many applications are out of reach**

Using the discrete Fourier transform (DFT) to accelerate GPs

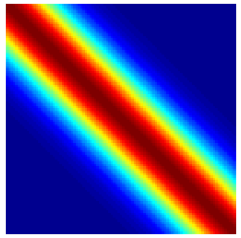
stationary covariance functions $\mathsf{K}(t, t') = \mathsf{k}(t - t')$



Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathrm{K}(t, t') = \mathrm{k}(t - t')$

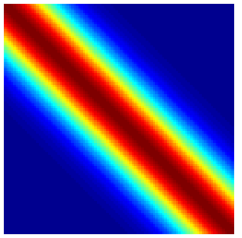
- eigen-functions are sines and cosines



Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t, t') = \mathsf{k}(t - t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu \; \gamma(\nu) \exp(2\pi i \tau \nu)$$

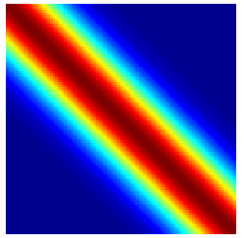
Fourier transform

power spectral density

Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t, t') = \mathsf{k}(t - t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu \; \gamma(\nu) \exp(2\pi i \tau \nu)$$

Fourier transform

power spectral density

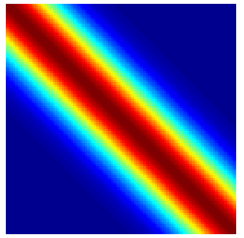
for **regularly sampled data**, leads to approximations based on the FFT

$$\Longrightarrow \; \mathcal{O}(T \log T)$$

Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t, t') = \mathsf{k}(t - t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu \; \gamma(\nu) \exp(2\pi i \tau \nu)$$

← Fourier transform

power spectral density

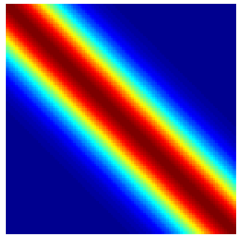
for **regularly sampled data**, leads to approximations based on the FFT

$$\mathsf{K}_{t,t'} \approx \sum_{k=1}^{T} \mathsf{FT}^{-1}_{t,k} \gamma_{d,k} \mathsf{FT}_{k,t'} \quad \Longrightarrow \; \mathcal{O}(T \log T)$$

Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t,t') = \mathsf{k}(t-t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu\ \gamma(\nu) \exp(2\pi i \tau \nu)$$

← Fourier transform

↑ power spectral density

for **regularly sampled data**, leads to approximations based on the FFT

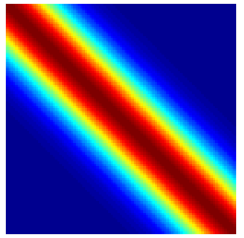
$$\mathsf{K}_{t,t'} \approx \sum_{k=1}^{T} \mathsf{FT}^{-1}_{t,k} \gamma_{d,k} \mathsf{FT}_{k,t'} \qquad \Longrightarrow\ \mathcal{O}(T \log T)$$

$$\mathsf{FT}_{k,t} = e^{-2\pi i (k-1)(t-1)/T}$$

Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t,t') = \mathsf{k}(t-t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu\ \gamma(\nu) \exp(2\pi i \tau \nu)$$

← Fourier transform

power spectral density

for **regularly sampled data**, leads to approximations based on the FFT

$$\mathsf{K}_{t,t'} \approx \sum_{k=1}^{T} \mathsf{FT}^{-1}_{t,k} \gamma_{d,k} \mathsf{FT}_{k,t'} \qquad \Longrightarrow\ \mathcal{O}(T \log T)$$

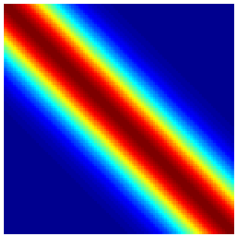
$$\mathsf{FT}_{k,t} = e^{-2\pi i (k-1)(t-1)/T}$$

$$p(\mathsf{y}_{1:T}|\theta) \propto \prod_{k=1}^{T} \gamma_k^{-1/2}(\theta) \exp\left(-\frac{1}{2} |\tilde{y}_k|^2 / \gamma_k\right)$$

Using the discrete Fourier transform (DFT) to accelerate GPs

stationary covariance functions $\mathsf{K}(t,t') = \mathsf{k}(t-t')$

- eigen-functions are sines and cosines



Bochner's theorem/Wiener Khintchine Theorem

$$\mathsf{k}(\tau) = \int \mathrm{d}\nu\, \gamma(\nu) \exp(2\pi i \tau \nu)$$

← Fourier transform

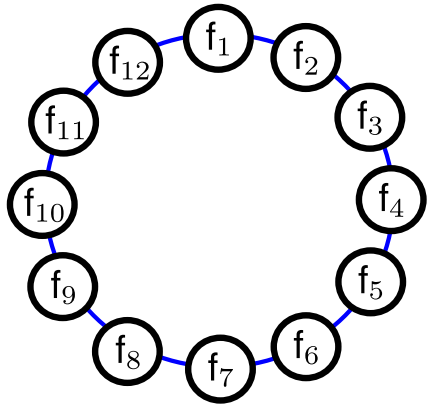
↑ power spectral density

for **regularly sampled data**, leads to approximations based on the FFT

$$\mathsf{K}_{t,t'} \approx \sum_{k=1}^{T} \mathsf{FT}^{-1}_{t,k} \gamma_{d,k} \mathsf{FT}_{k,t'} \quad \Longrightarrow \mathcal{O}(T \log T)$$

$$\mathsf{FT}_{k,t} = e^{-2\pi i (k-1)(t-1)/T}$$

$$p(\mathsf{y}_{1:T}|\theta) \propto \prod_{k=1}^{T} \gamma_k^{-1/2}(\theta) \exp\left(-\frac{1}{2} |\tilde{y}_k|^2 / \gamma_k\right)$$



alternative view: new model with ring topography + exact inference

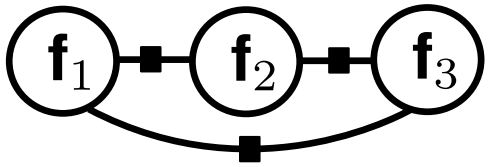
Stochastic differential equations (SDE)

- core idea: **convert into a classical smoothing problem**
- linear, time-invariant stochastic differential equation:

$$\frac{\mathrm{d}^M}{\mathrm{d}t^M} f(t) + c_{M-1} \frac{\mathrm{d}^{M-1}}{\mathrm{d}t^{M-1}} f(t) + \ldots + c_0 f(t) = w(t)$$

- defines a (stationary) GP covariance function (filters the white noise)
- **procedure:**
  - convert GP covariance function to above form (spectrum matching procedure), truncate derivative expansion (if necessary)
  - implement inference using Rauch–Tung–Striebel smoothing (truncation sets dimensionality of state space, $\tilde{M}$)
- **Linear complexity in time** (cubic in size of state-space, finding LDS parameters costly cf learning): $\mathcal{O}(T\tilde{M}^3)$

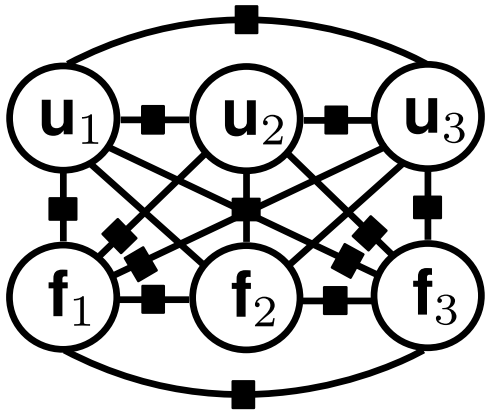
Chain structured indirect approximation



Chain structured indirect approximation

1. augment model with M<T pseudo data

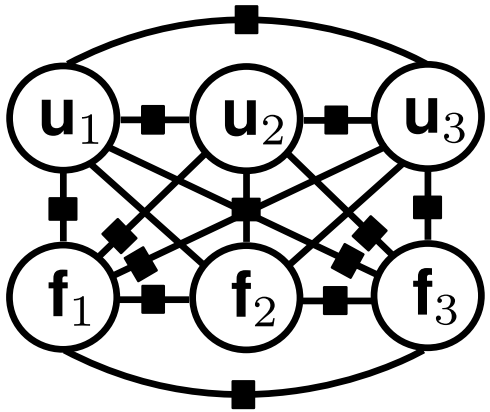
$$p(\mathbf{f}, \mathbf{u}) = \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ \mathbf{u} \end{bmatrix}; \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \begin{bmatrix} \mathsf{K}_{\mathbf{ff}} & \mathsf{K}_{\mathbf{fu}} \\ \mathsf{K}_{\mathbf{uf}} & \mathsf{K}_{\mathbf{uu}} \end{bmatrix}\right)$$



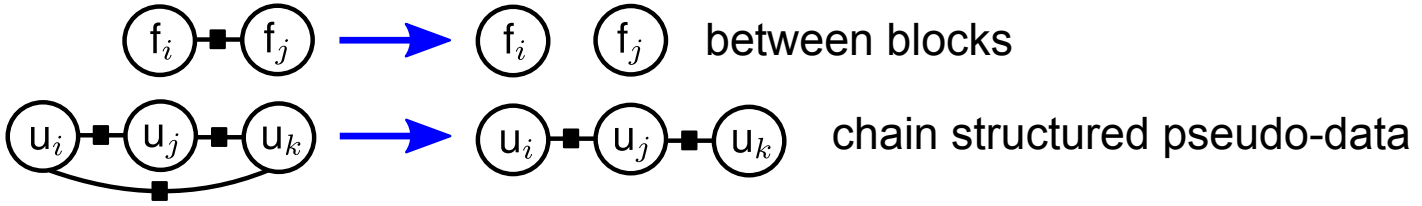
Chain structured indirect approximation

1. augment model with M<T pseudo data

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



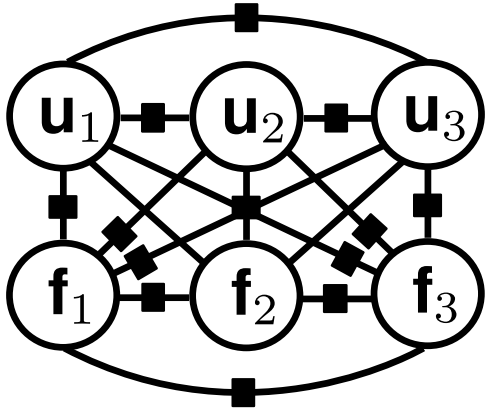
2. remove some of the dependencies
   (results in simpler model)



Chain structured indirect approximation

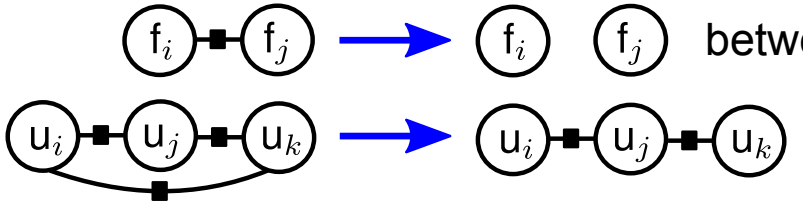
1. augment model with M<T pseudo data

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



$K = 3$ blocks

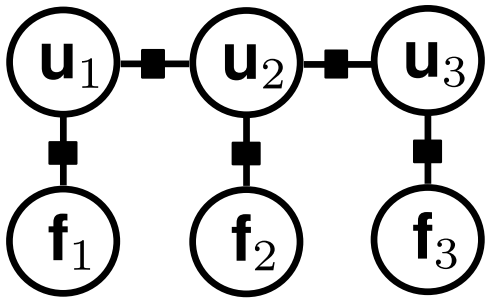
2. remove some of the dependencies
   (results in simpler model)



Chain structured indirect approximation

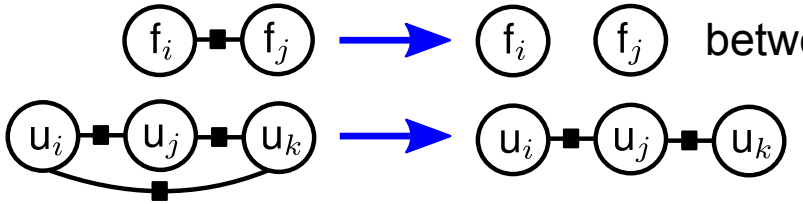
1. augment model with M<T pseudo data

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



$K = 3$ blocks

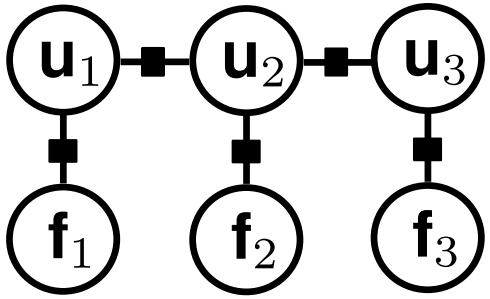
2. remove some of the dependencies
(results in simpler model)



Chain structured indirect approximation

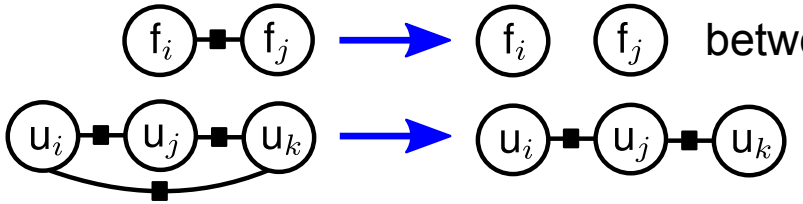
1. augment model with M<T pseudo data

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



$K = 3$ blocks

2. remove some of the dependencies
   (results in simpler model)



between blocks

chain structured pseudo-data

3. calibrate model
   (e.g. using KL divergence, many choices)

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

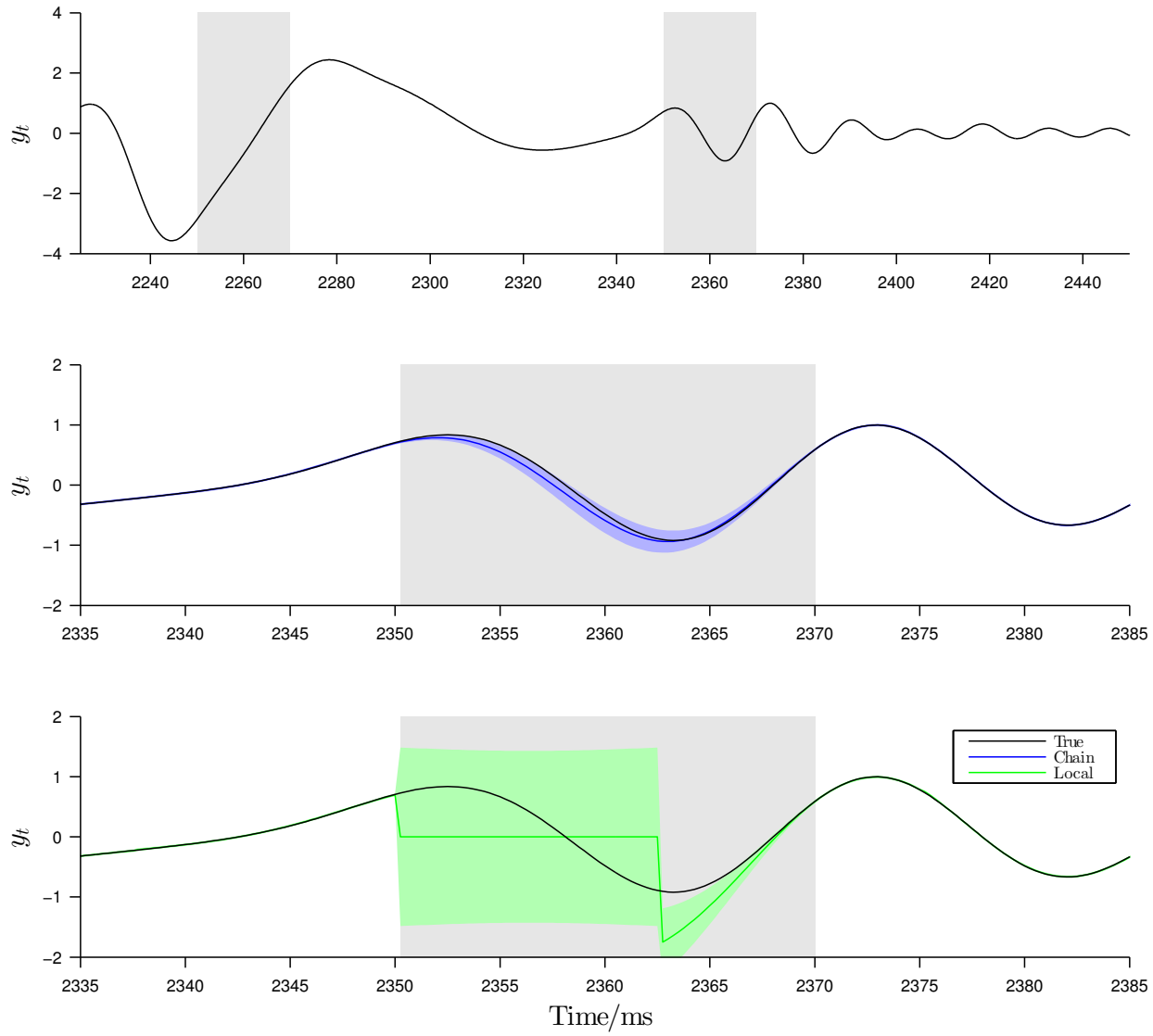
$$\Longrightarrow \quad \begin{aligned} q(\mathbf{u}_k|\mathbf{u}_{k-1}) &= p(\mathbf{u}_k|\mathbf{u}_{k-1})\\ q(\mathbf{f}_k|\mathbf{u}_k) &= p(\mathbf{f}_k|\mathbf{u}_k)\end{aligned}$$

equal to exact conditionals

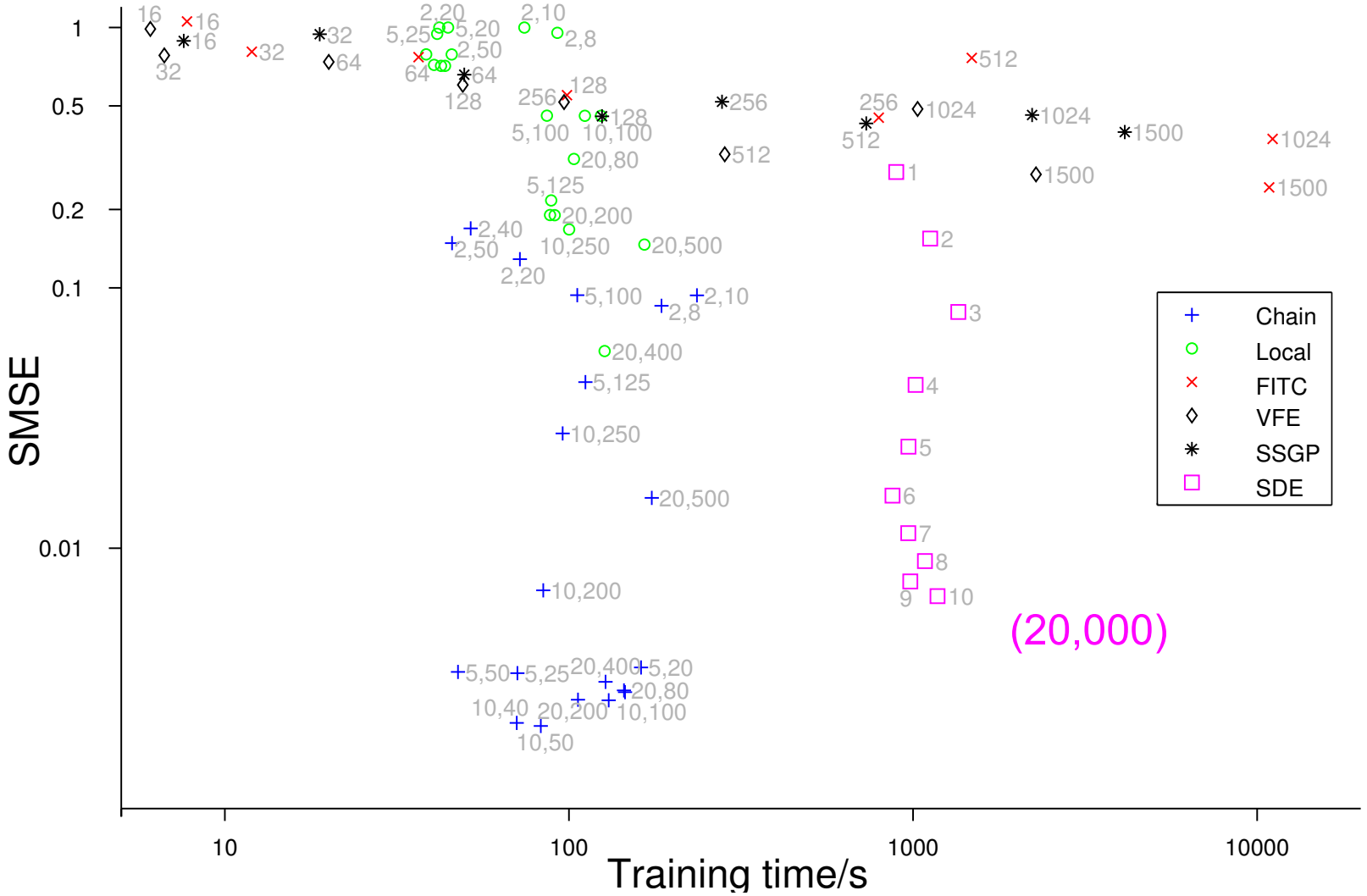
Chain structured indirect approximation

- **cost of inference is linear in T** (Rauch–Tung–Striebel smoothing)
- PITC, FITC and local versions are special cases (more edges deleted)
- sensible choices for the blocking and numbers of pseudo-points required

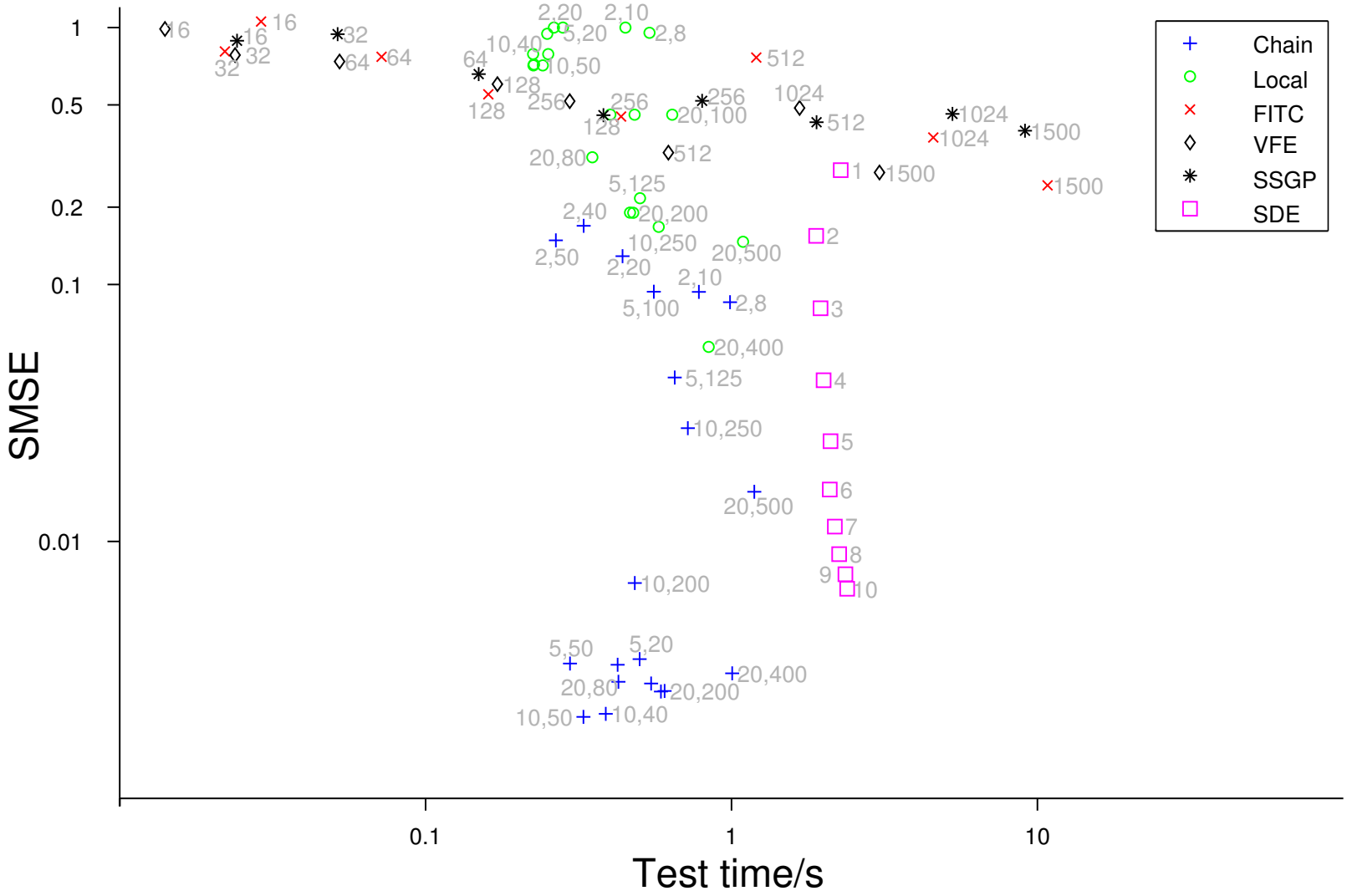
Comparisons: training time (T=50,000)



Comparisons: training time (T=50,000)



Comparisons: testing time (T=50,000)



Summary

- **taxonomy of GP approximations** (direct/indirect, w/ or w/o pseudo-data, likelihood/prior)
- **standard pseudo datapoint based approximations do not work well for long time-series**
- **chain structured variant performs better** (outperforming other time-series GP approximations)
- **opportunity**: direct approximations in which pseudo-points only appear in the recognition model