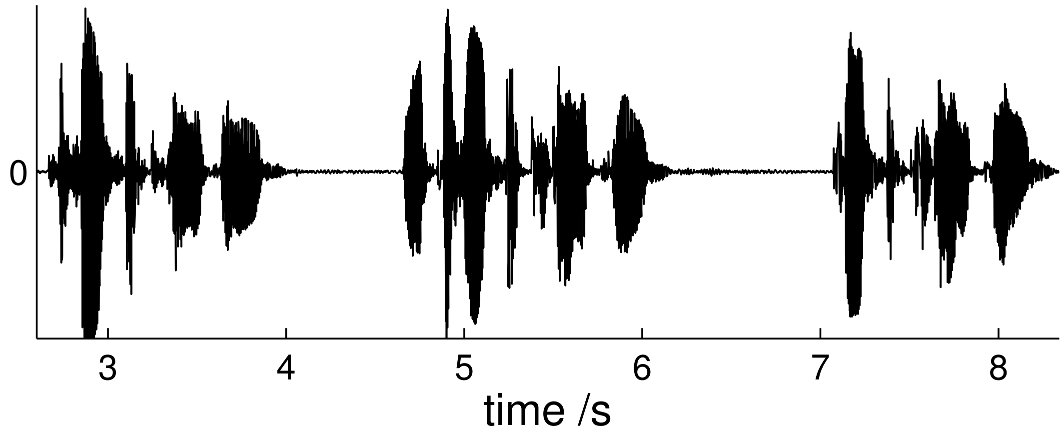
Gaussian Processes: large data and non-linear models

Richard E. Turner
University of Cambridge

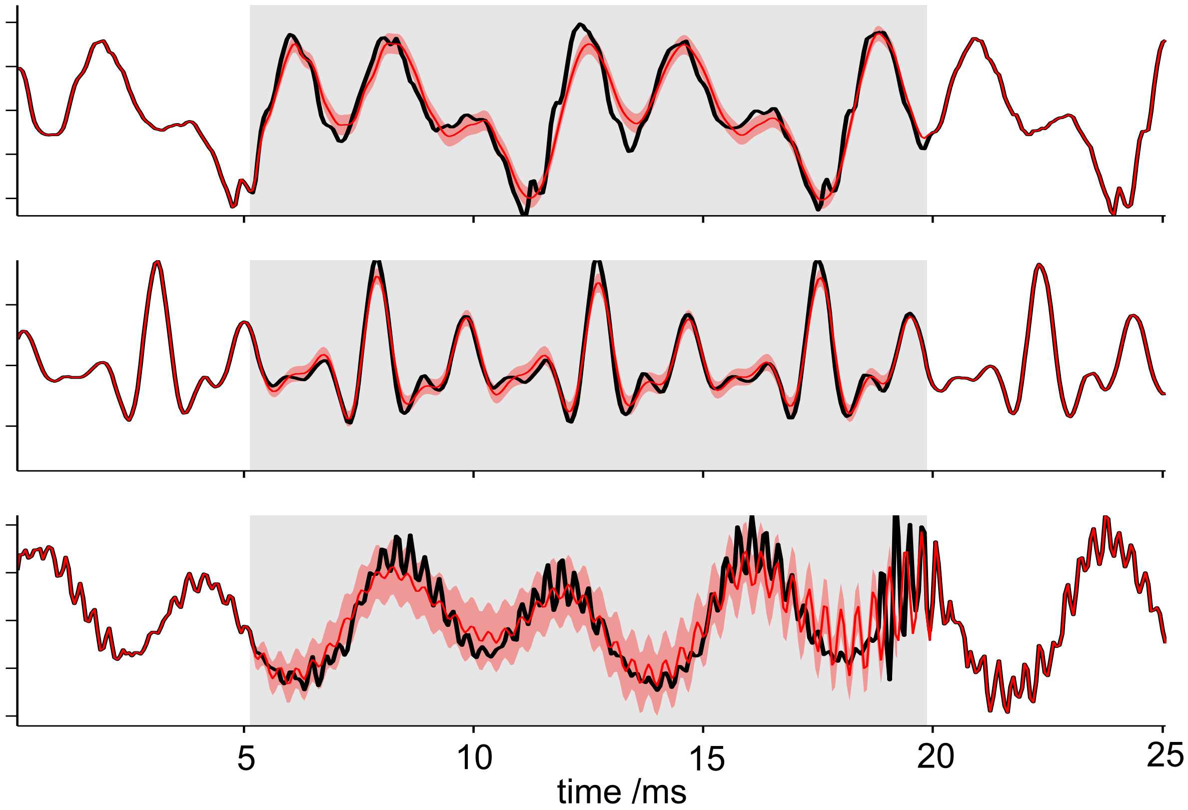
Motivating application 1: Audio modelling

audio
time-series
data



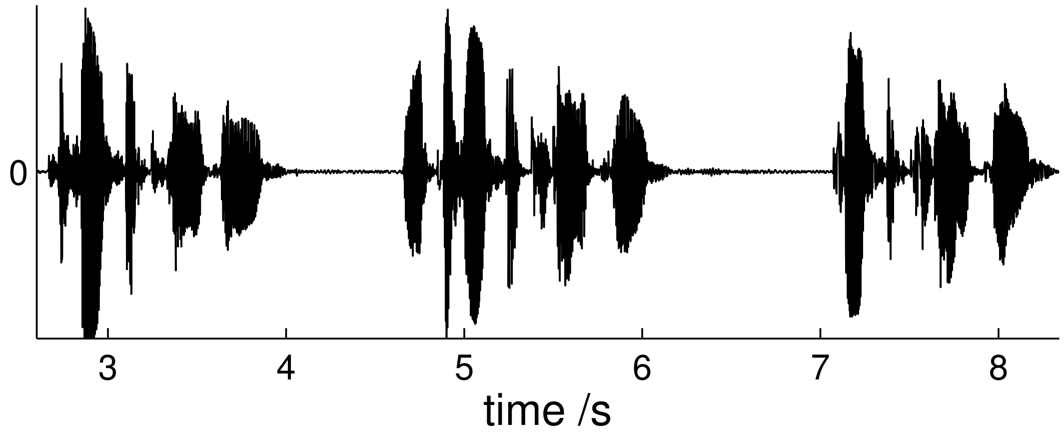
$y(t)$
$T = 10^5\text{-}10^7$
datapoints

reconstruction
using a GP
model



Motivating application 1: Audio modelling

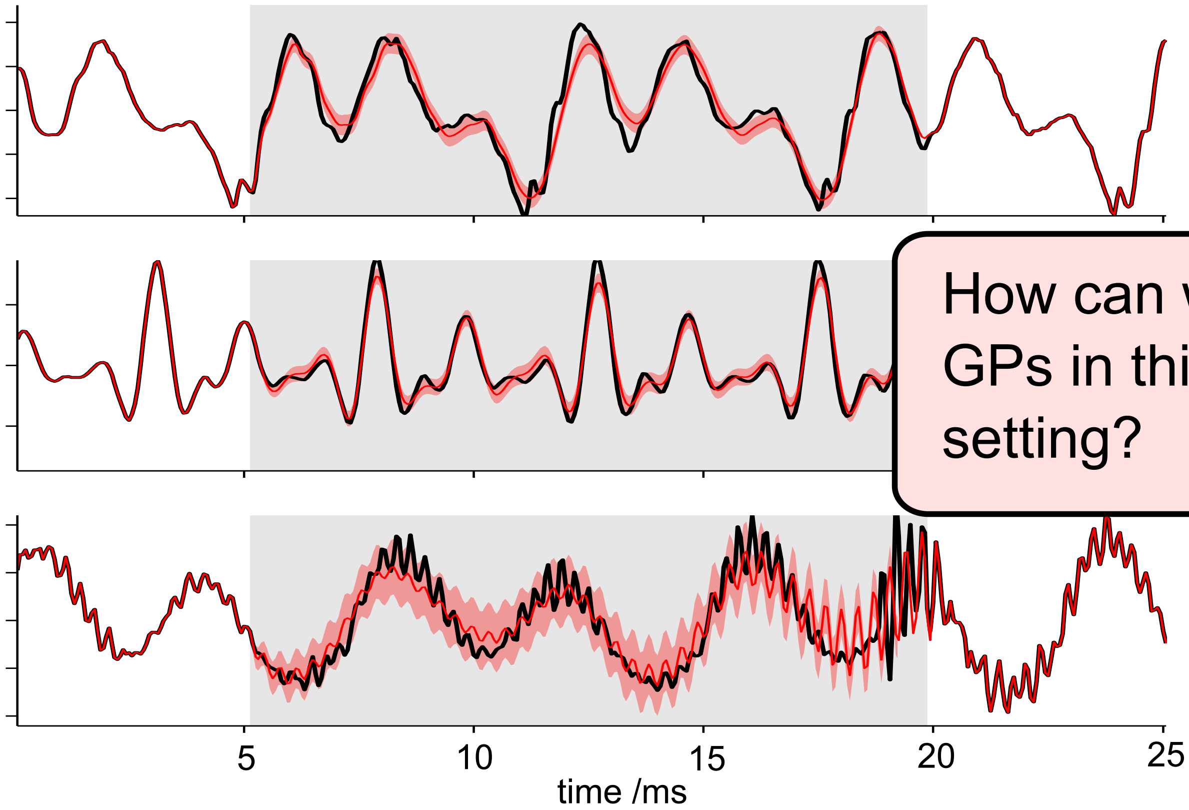
audio
time-series
data



$y(t)$

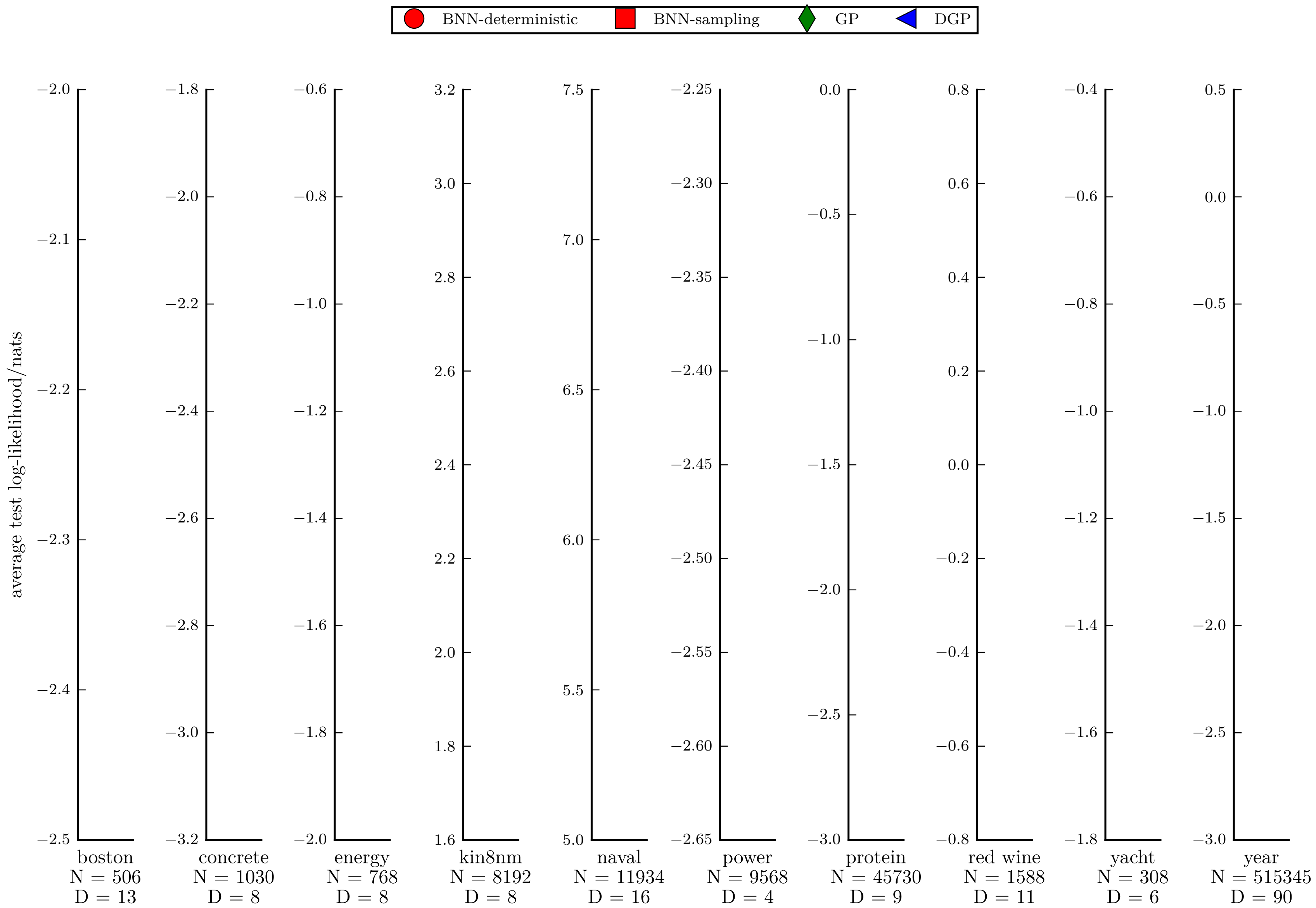
$T = 10^5$-10^7 datapoints

reconstruction
using a GP
model

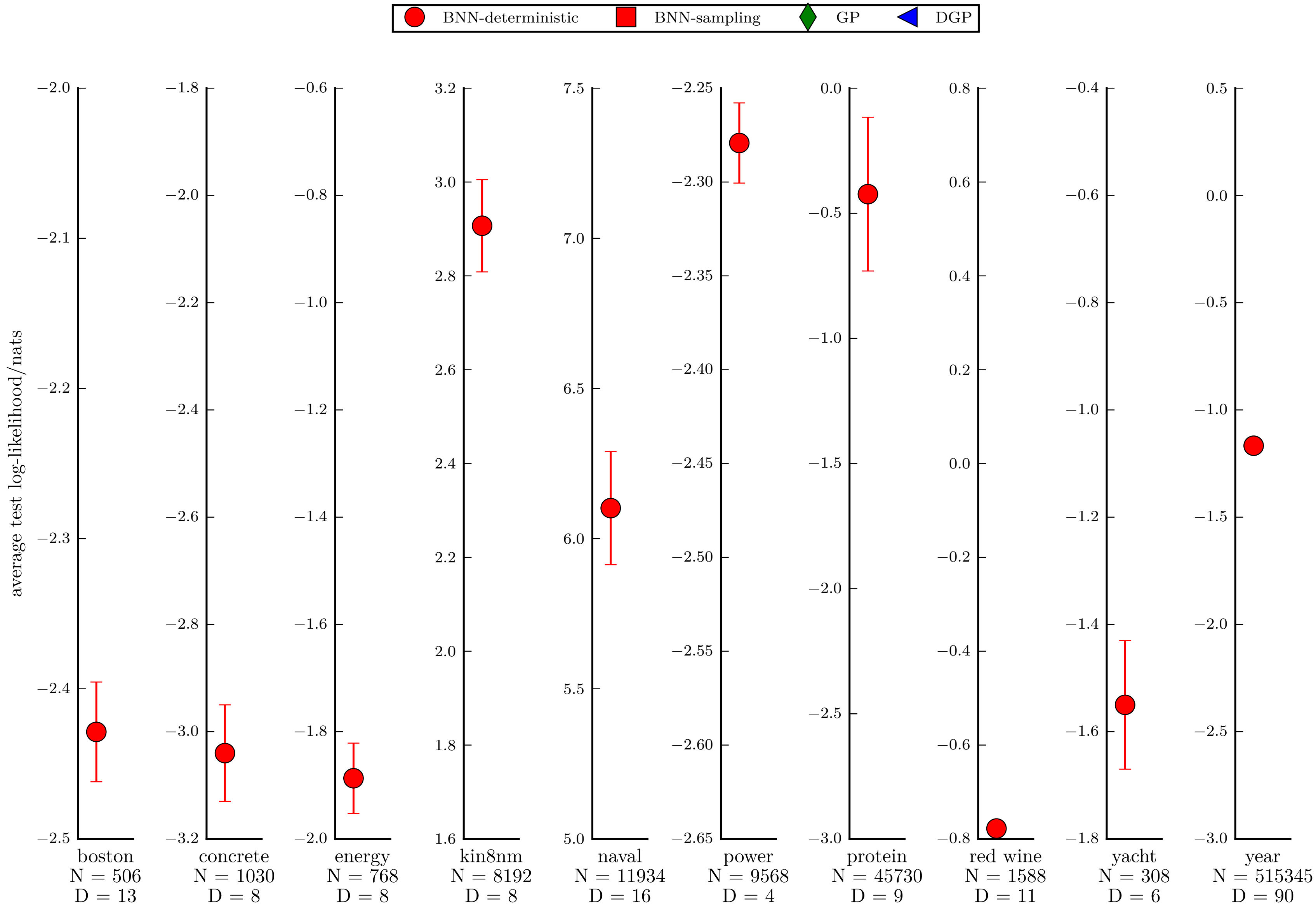


How can we use GPs in this setting?

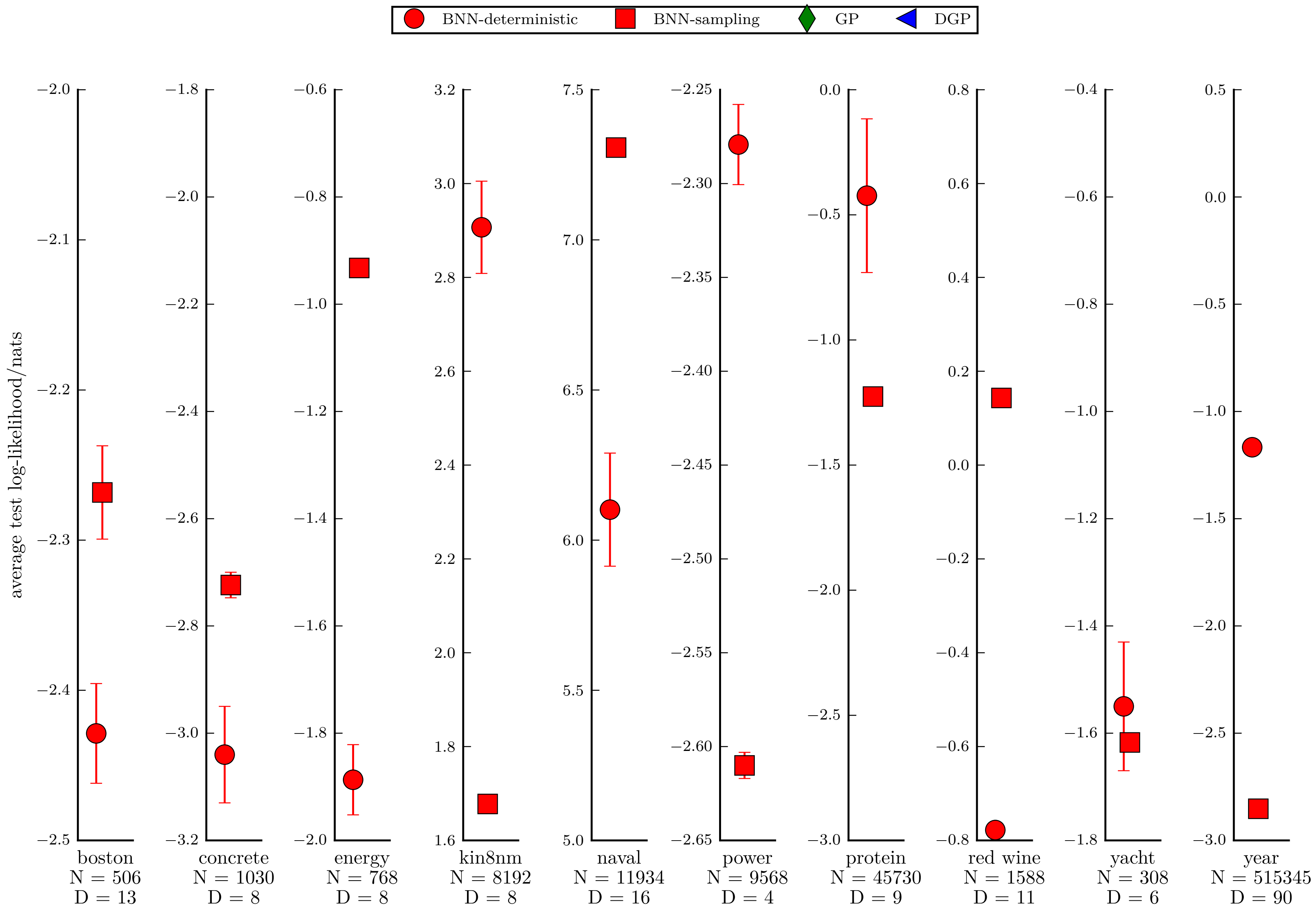
Motivating application 2: non-linear regression



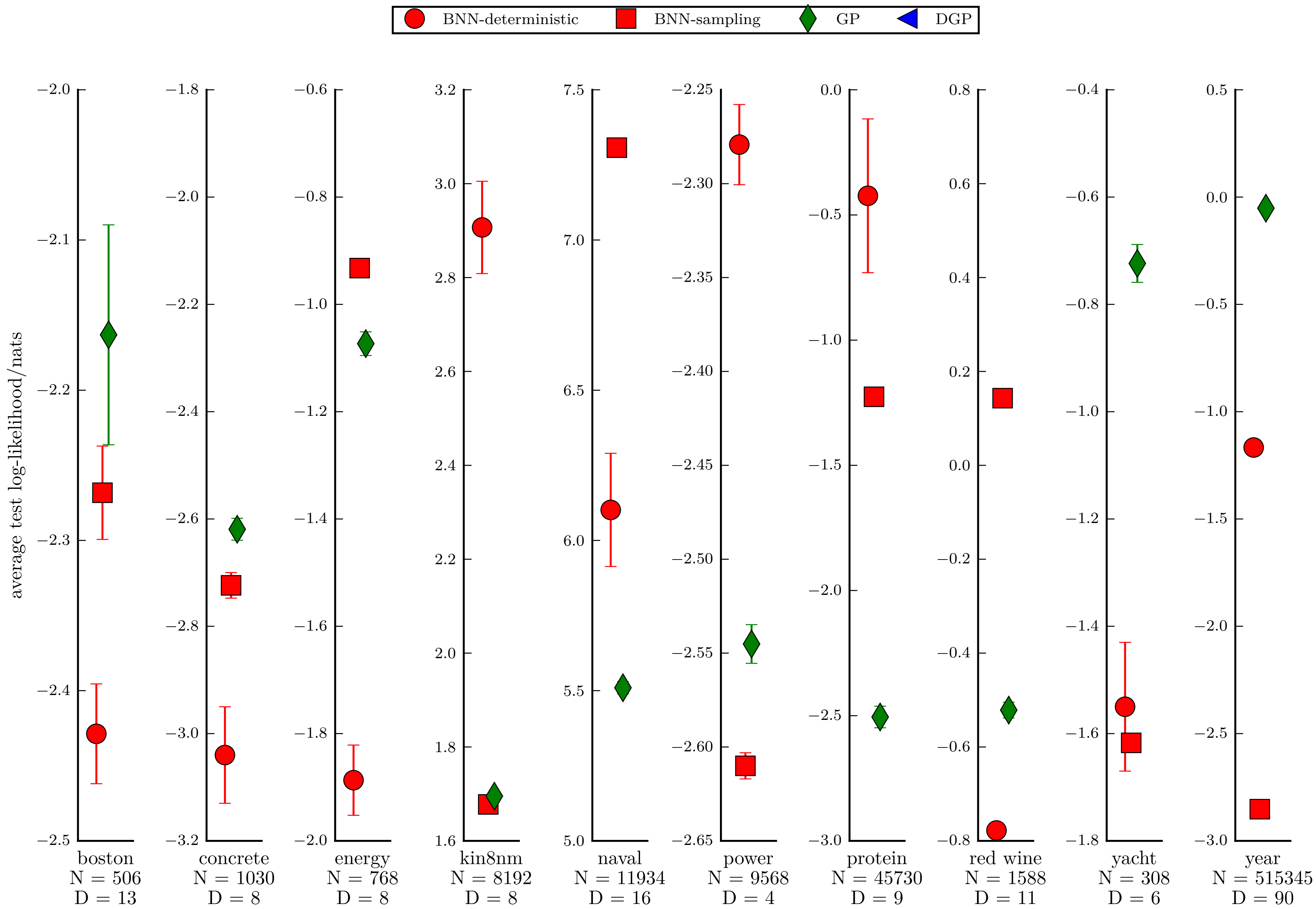
Motivating application 2: non-linear regression



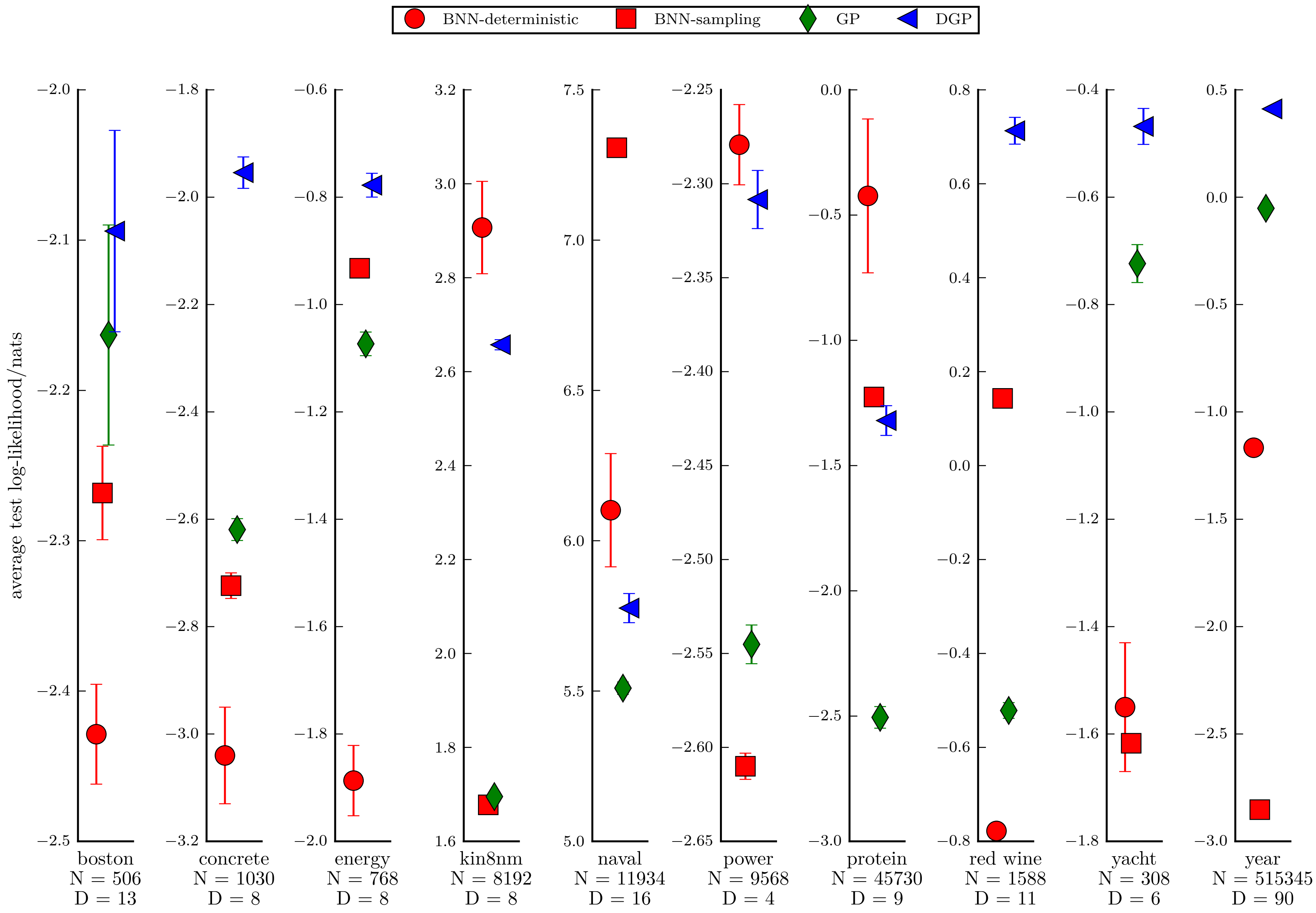
Motivating application 2: non-linear regression



Motivating application 2: non-linear regression



Motivating application 2: non-linear regression



Outline of the tutorial

- **An Introduction to GPs**
  - Mathematical foundations
  - Hyper-parameter learning
  - Covariance functions
  - Multi-dimensional inputs
- **Using GPs: Models, Applications and Connections**
  - Models and more on covariance functions
  - Applications
  - Connections
- **GPs for large data and non-linear models**
  - Scaling through pseudo-data: changing the generative model
  - Scaling through pseudo-data: variational Inference
  - General Approximate inference

GP regression: introducing notation

Q1. What's the formal justification for how we were using GPs for regression?

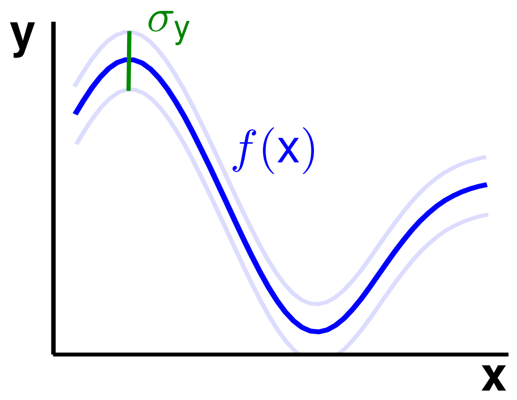
GP regression: introducing notation

Q1. What's the formal justification for how we were using GPs for regression?

generative model (like non-linear regression)

$$\mathsf{y}(\mathsf{x}) = f(\mathsf{x}) + \epsilon\sigma_{\mathsf{y}}$$

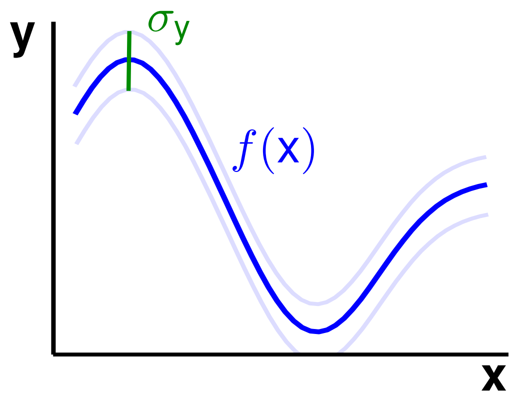
$$p(\epsilon) = \mathcal{N}(\epsilon; 0, 1)$$



GP regression: introducing notation

Q1. What's the formal justification for how we were using GPs for regression?

generative model (like non-linear regression)



$$y(x) = f(x) + \epsilon\sigma_y$$

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

place GP prior over the non-linear function

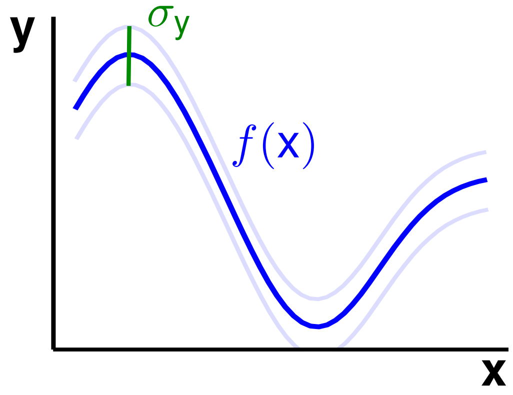
$$p(f(x)|\theta) = \mathcal{GP}(f(x); 0, K_\theta(x, x'))$$

$$K(x, x') = \sigma^2 \exp\left(-\frac{1}{2l^2}(x - x')^2\right)$$ (smoothly wiggling functions expected)

GP regression: introducing notation

Q1. What's the formal justification for how we were using GPs for regression?

generative model (like non-linear regression)



$$y(x) = f(x) + \epsilon\sigma_y$$

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

place GP prior over the non-linear function

$$p(f(x)|\theta) = \mathcal{GP}(f(x); 0, K_\theta(x, x'))$$

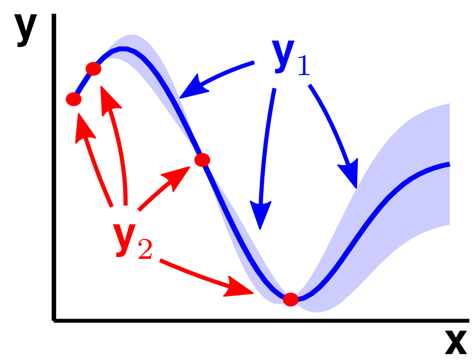
$$K(x, x') = \sigma^2 \exp\left(-\frac{1}{2l^2}(x - x')^2\right)$$ (smoothly wiggling functions expected)

sum of Gaussian variables = Gaussian: induces a GP over $y(x)$

$$p(y(x)|\theta) = \mathcal{GP}(y(x); 0, K_\theta(x, x') + I\sigma_y^2)$$

GP regression: introducing notation

Q4. How do we make predictions?



$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} \mathsf{A} & \mathsf{B} \\ \mathsf{B}^\mathsf{T} & \mathsf{C} \end{bmatrix}\right)$$

$$p(\mathbf{y}_1 | \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)}$$

$$\Longrightarrow \; p(\mathbf{y}_1|\mathbf{y}_2) = \mathcal{N}(\, \mathbf{y}_1 \,;\, \underline{\mathbf{a} + \mathsf{B}\mathsf{C}^{-1}(\mathbf{y}_2 - \mathbf{b})}, \mathsf{A} - \mathsf{B}\mathsf{C}^{-1}\mathsf{B}^\mathsf{T})$$

predictive mean

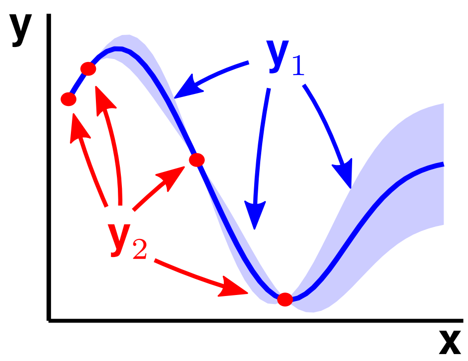
$$\begin{aligned} \mu_{\mathbf{y}_1|\mathbf{y}_2} &= \mathbf{a} + \mathsf{B}\mathsf{C}^{-1}(\mathbf{y}_2 - \mathbf{b}) \\ &= \mathsf{B}\mathsf{C}^{-1}\mathbf{y}_2 \\ &= \mathsf{W}\mathbf{y}_2 \end{aligned}$$

GP regression: introducing notation

Q4. How do we make predictions?

$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} \mathsf{A} & \mathsf{B} \\ \mathsf{B}^\mathsf{T} & \mathsf{C} \end{bmatrix}\right)$$

$$p(\mathbf{y}_1 | \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)}$$



$$\Longrightarrow \quad p(\mathbf{y}_1 | \mathbf{y}_2) = \mathcal{N}(\,\mathbf{y}_1\,;\, \underline{\mathbf{a} + \mathsf{BC}^{-1}(\mathbf{y}_2 - \mathbf{b})}, \mathsf{A} - \mathsf{BC}^{-1}\mathsf{B}^\mathsf{T})$$

predictive mean

$$\mu_{\mathbf{y}_1|\mathbf{y}_2} = \mathbf{a} + \mathsf{BC}^{-1}(\mathbf{y}_2 - \mathbf{b})$$

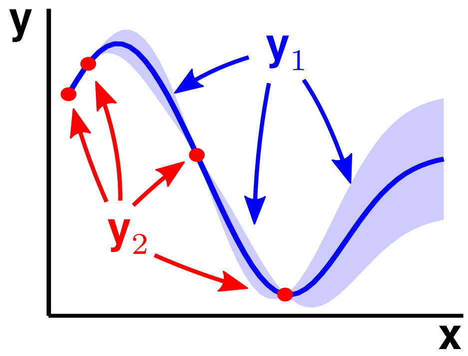
$$= \mathsf{BC}^{-1}\mathbf{y}_2$$

$$= \mathsf{W}\mathbf{y}_2$$

linear in the data

GP regression: introducing notation

Q4. How do we make predictions?



$$p(\mathbf{y}_1, \mathbf{y}_2) = \mathcal{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix} ; \begin{bmatrix} \mathbf{a} \\ \mathbf{b} \end{bmatrix}, \begin{bmatrix} \mathsf{A} & \mathsf{B} \\ \mathsf{B}^\top & \mathsf{C} \end{bmatrix} \right)$$

$$p(\mathbf{y}_1 | \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)}$$

$$\implies p(\mathbf{y}_1 | \mathbf{y}_2) = \mathcal{N}(\mathbf{y}_1 ; \mathbf{a} + \mathsf{B}\mathsf{C}^{-1}(\mathbf{y}_2 - \mathbf{b}), \mathsf{A} - \mathsf{B}\mathsf{C}^{-1}\mathsf{B}^\top)$$

predictive mean

$$\mu_{\mathbf{y}_1|\mathbf{y}_2} = \mathbf{a} + \mathsf{B}\mathsf{C}^{-1}(\mathbf{y}_2 - \mathbf{b})$$

$$= \mathsf{B}\mathsf{C}^{-1}\mathbf{y}_2$$

$$= \mathsf{W}\mathbf{y}_2$$

linear in the data

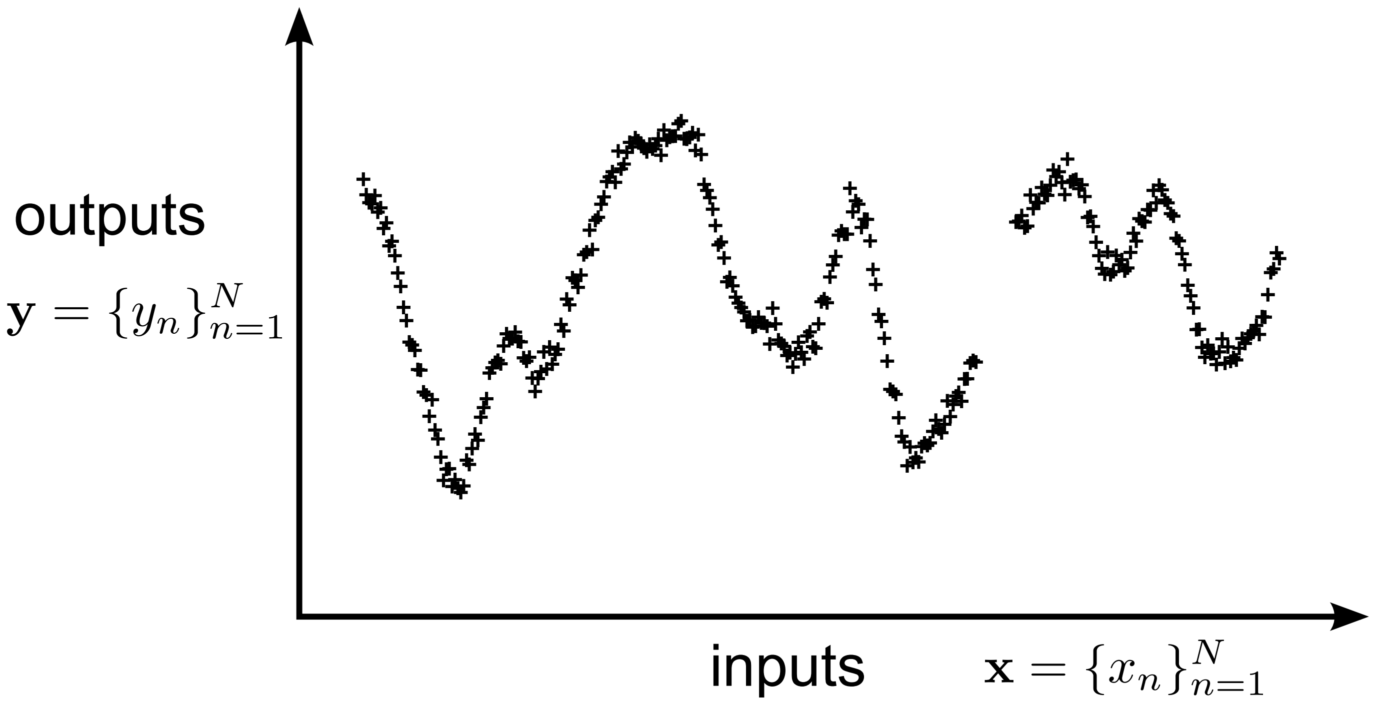
predictive covariance

$$\Sigma_{\mathbf{y}_1|\mathbf{y}_2} = \mathsf{A} - \mathsf{B}\mathsf{C}^{-1}\mathsf{B}^\top$$

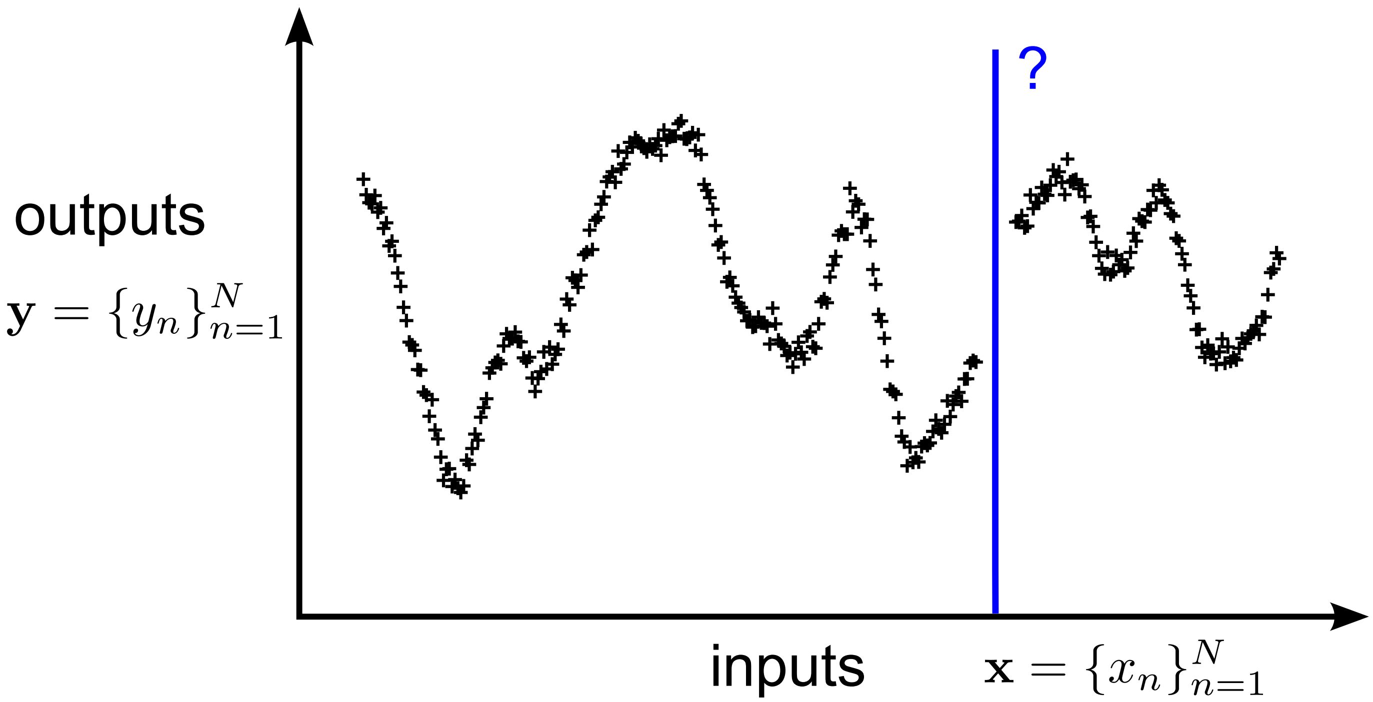
predictive uncertainty = prior uncertainty − reduction in uncertainty

predictions more confident than prior

Motivation: Gaussian Process Regression



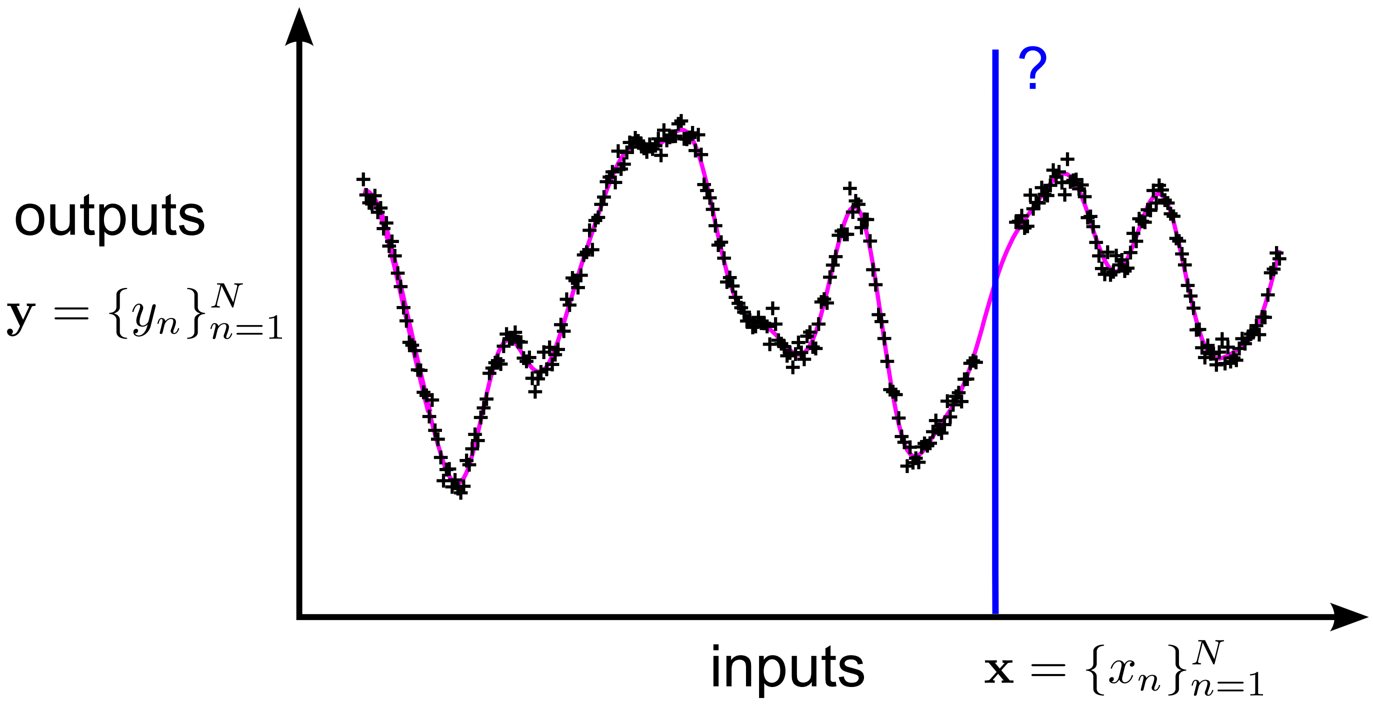
Motivation: Gaussian Process Regression



Motivation: Gaussian Process Regression

$$p(f|\theta) = \mathcal{GP}(f; 0, \mathrm{K}_\theta)$$

$$p(y_n|f, x_n, \theta)$$



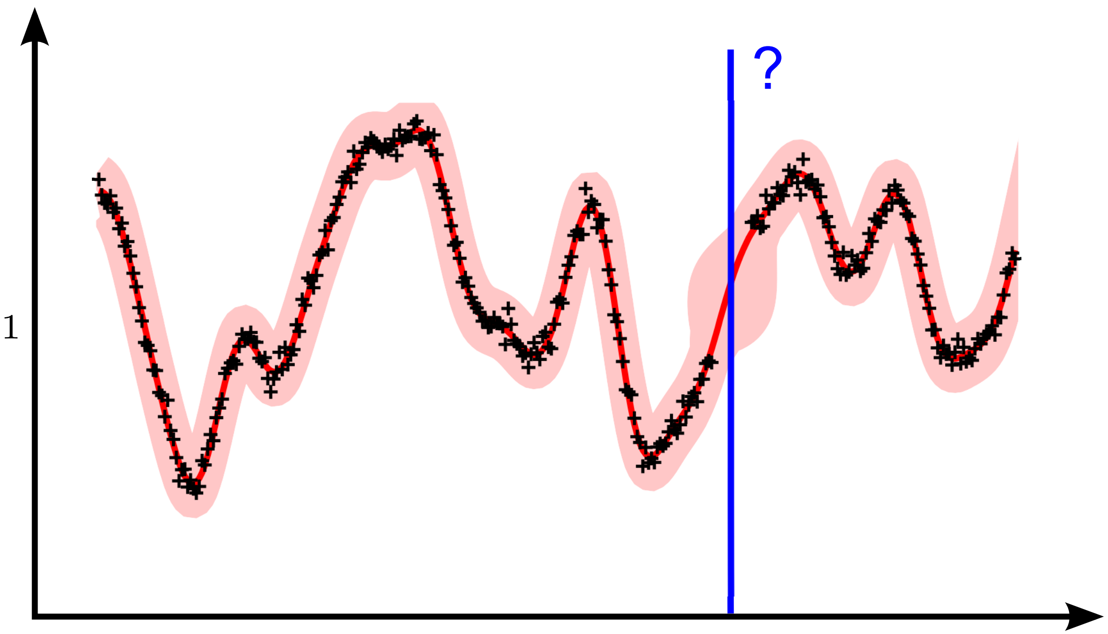
Motivation: Gaussian Process Regression

$$p(f|\theta) = \mathcal{GP}(f; 0, \mathrm{K}_\theta) \qquad \xrightarrow{\text{inference \& learning}} \qquad p(f|\mathbf{y}, \mathbf{x}, \theta)$$

$$p(y_n|f, x_n, \theta) \qquad\qquad\qquad\qquad\qquad p(\mathbf{y}|\mathbf{x}, \theta)$$

outputs

$\mathbf{y} = \{y_n\}_{n=1}^N$



inputs $\quad \mathbf{x} = \{x_n\}_{n=1}^N$

Motivation: Gaussian Process Regression

$$p(f|\theta) = \mathcal{GP}(f; 0, \mathrm{K}_\theta) \qquad p(y_n|f, x_n, \theta)$$

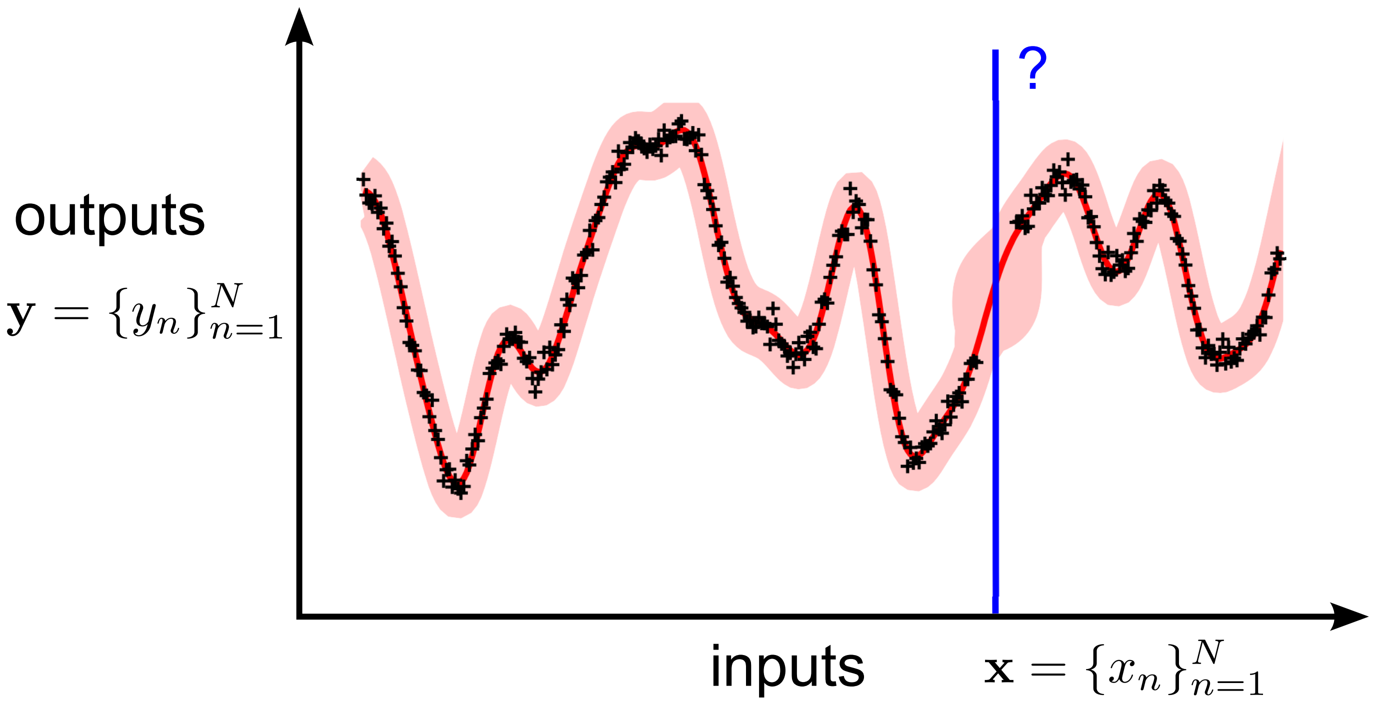
$$\xrightarrow{\text{inference \& learning}}$$

intractabilities

computational $\mathcal{O}(N^3)$

analytic

$$p(f|\mathbf{y}, \mathbf{x}, \theta) \qquad p(\mathbf{y}|\mathbf{x}, \theta)$$



Motivation: Gaussian Process Regression

$p(f|\theta) = \mathcal{GP}(f; 0, \mathrm{K}_\theta)$

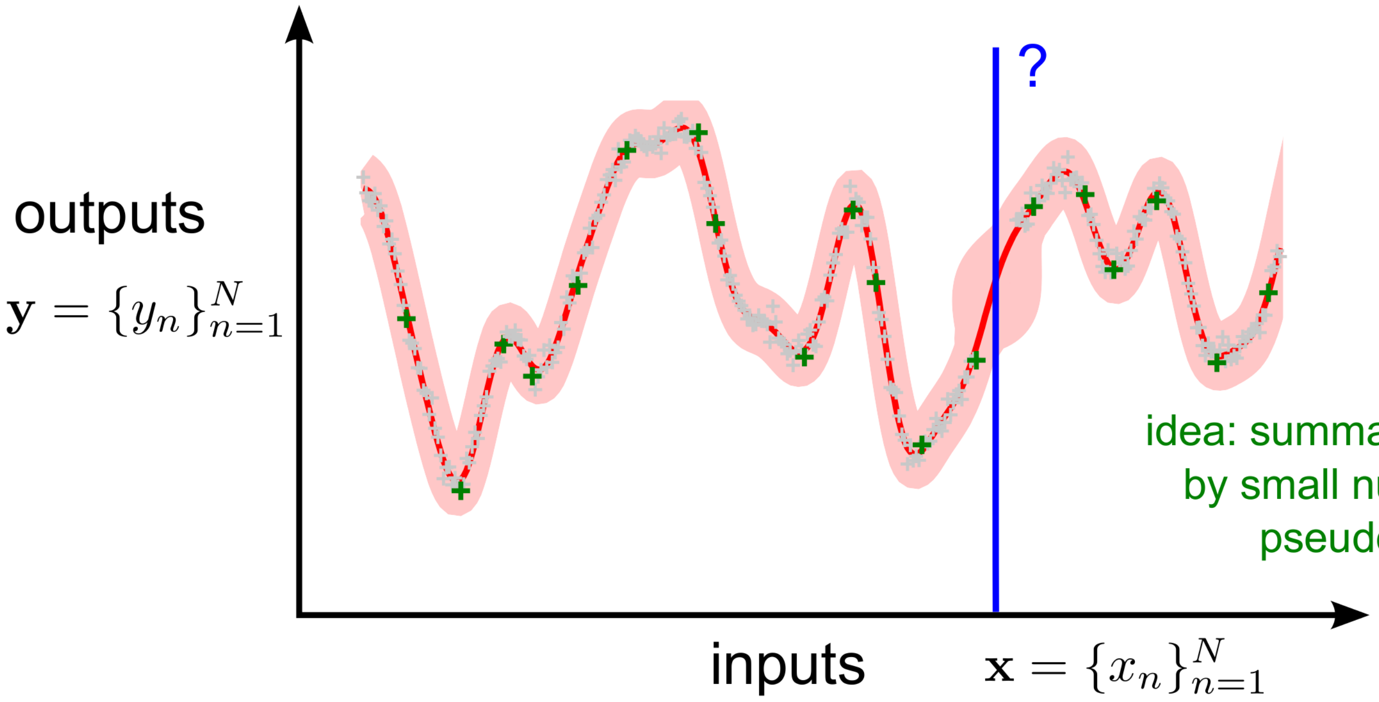
$p(y_n|f, x_n, \theta)$

inference & learning

intractabilities
computational $\mathcal{O}(N^3)$
analytic

$p(f|\mathbf{y}, \mathbf{x}, \theta)$

$p(\mathbf{y}|\mathbf{x}, \theta)$



idea: summarise dataset
by small number (M)
pseudo-data

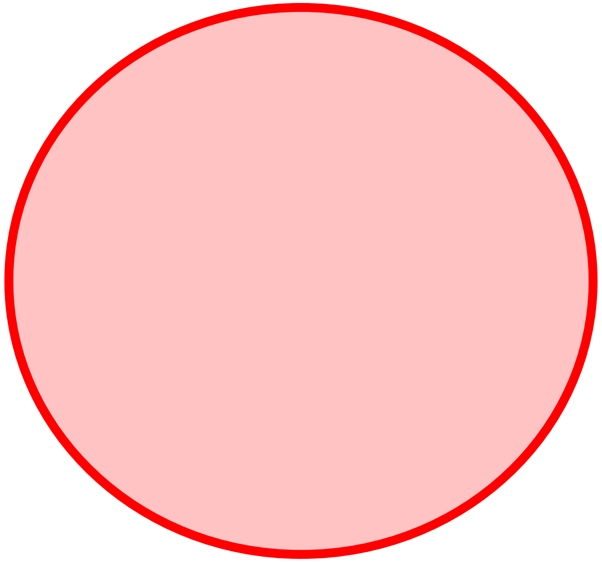
A Brief History of Gaussian Process Approximations

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

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



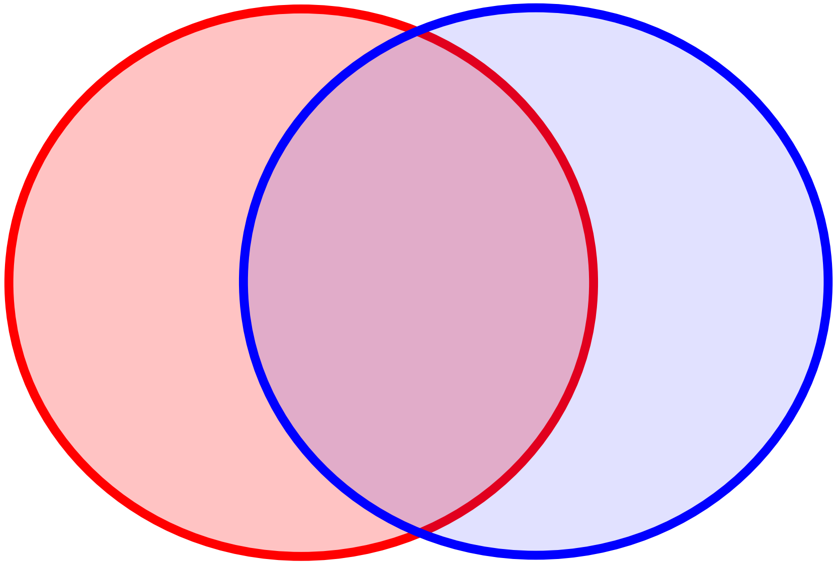
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

methods employing
pseudo-data

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



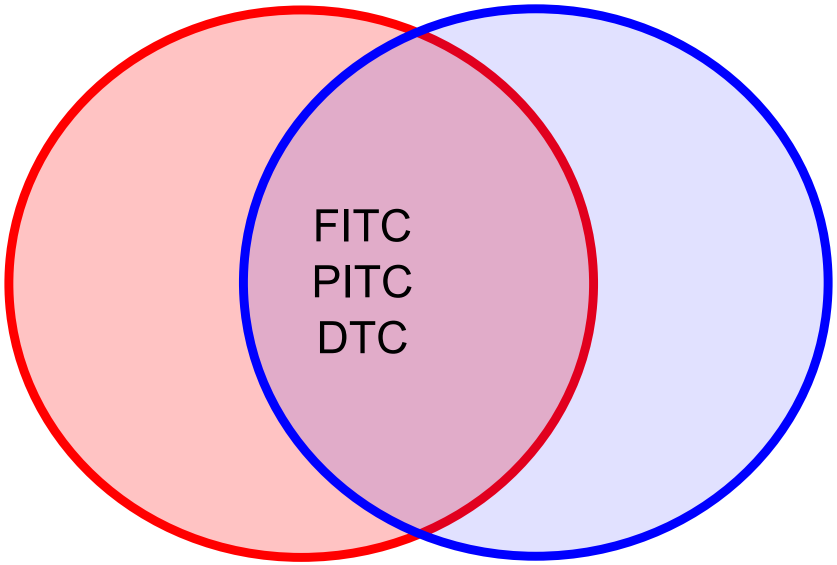
FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

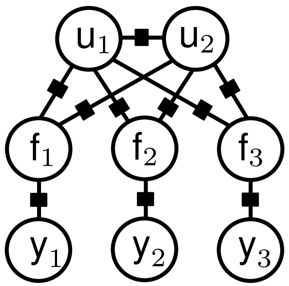
A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

methods employing
pseudo-data

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





FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

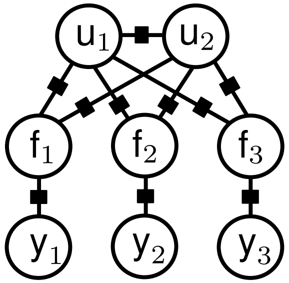
A Brief History of Gaussian Process Approximations

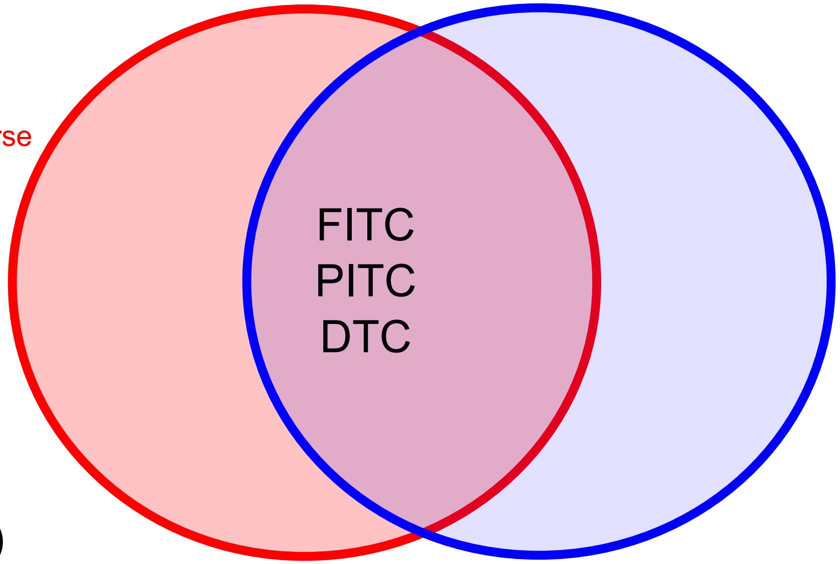
approximate generative model
exact inference

methods employing
pseudo-data

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

A Unifying View of Sparse Approximate Gaussian Process Regression Quinonero-Candela & Rasmussen, 2005 (FITC, PITC, DTC)





FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

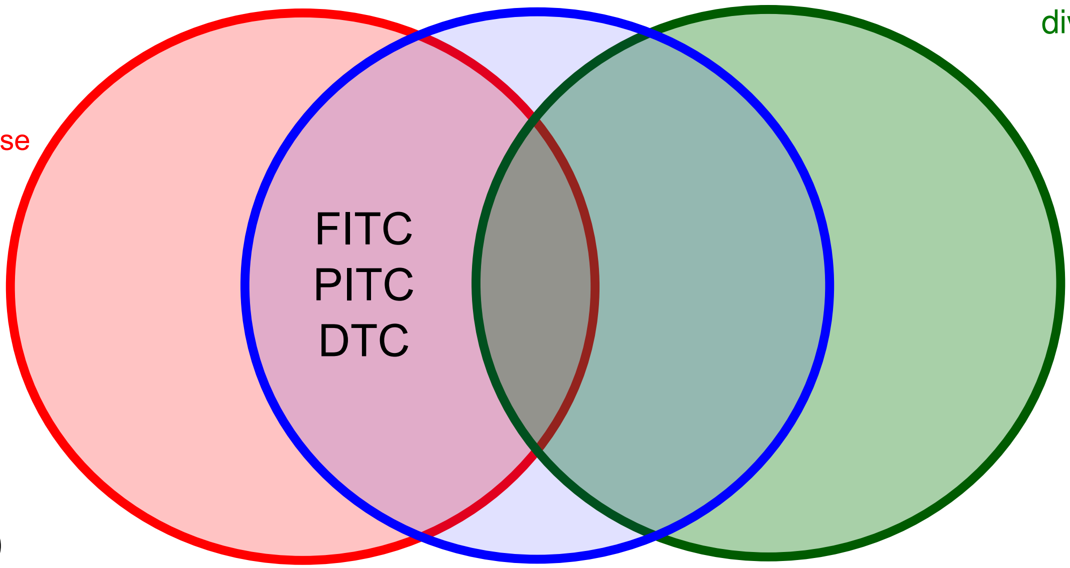
approximate generative model
exact inference

methods employing
pseudo-data

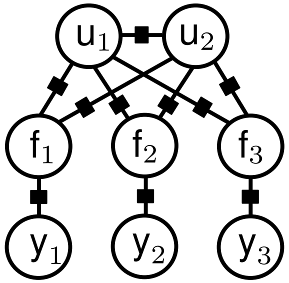
exact generative model
approximate inference

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

A Unifying View of Sparse Approximate Gaussian Process Regression Quinonero-Candela & Rasmussen, 2005 (FITC, PITC, DTC)



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



FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

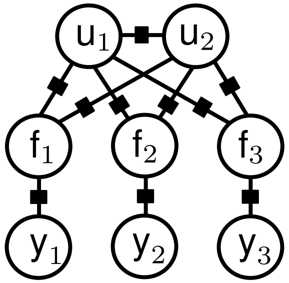
methods employing
pseudo-data

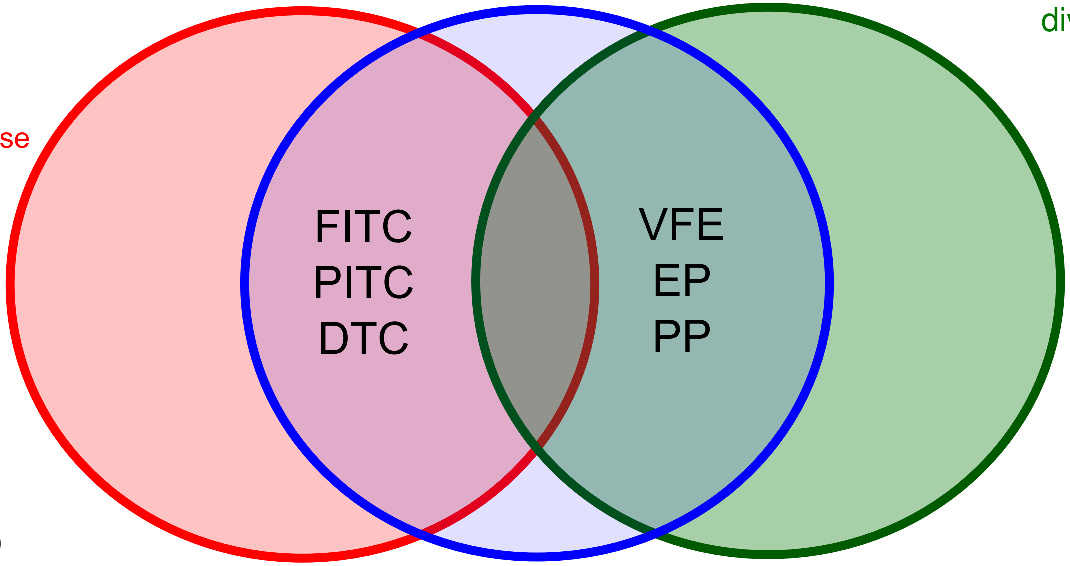
exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression Quinonero-Candela & Rasmussen, 2005 (FITC, PITC, DTC)





FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

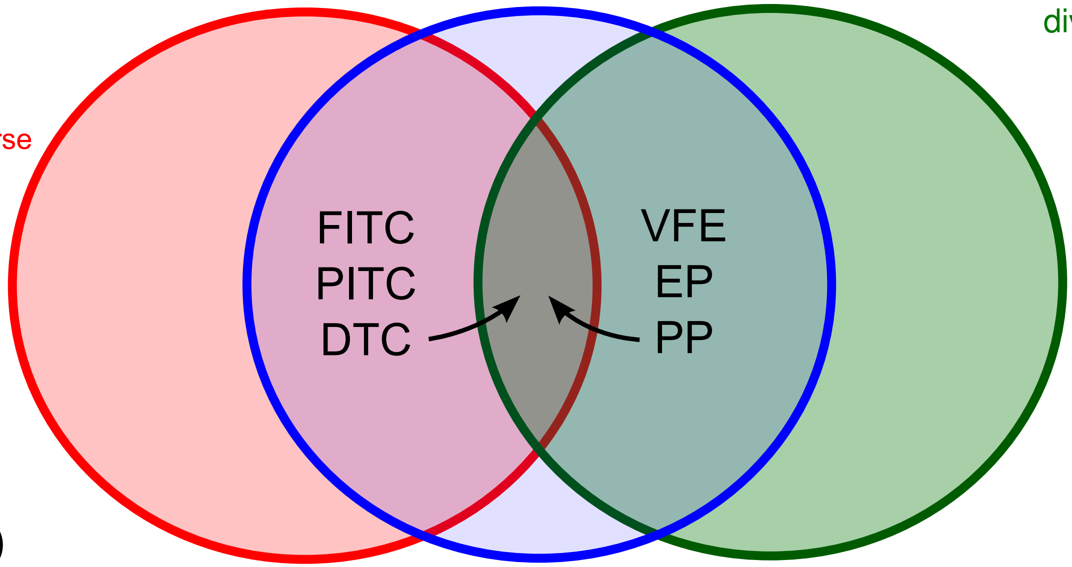
methods employing
pseudo-data

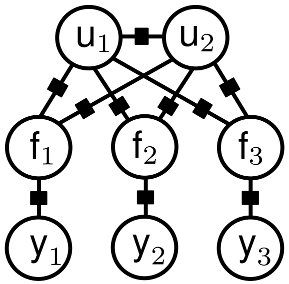
exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)





FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

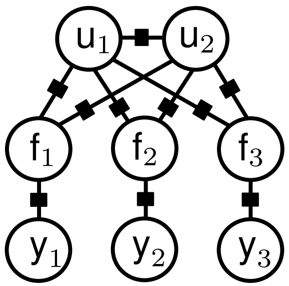
methods employing
pseudo-data

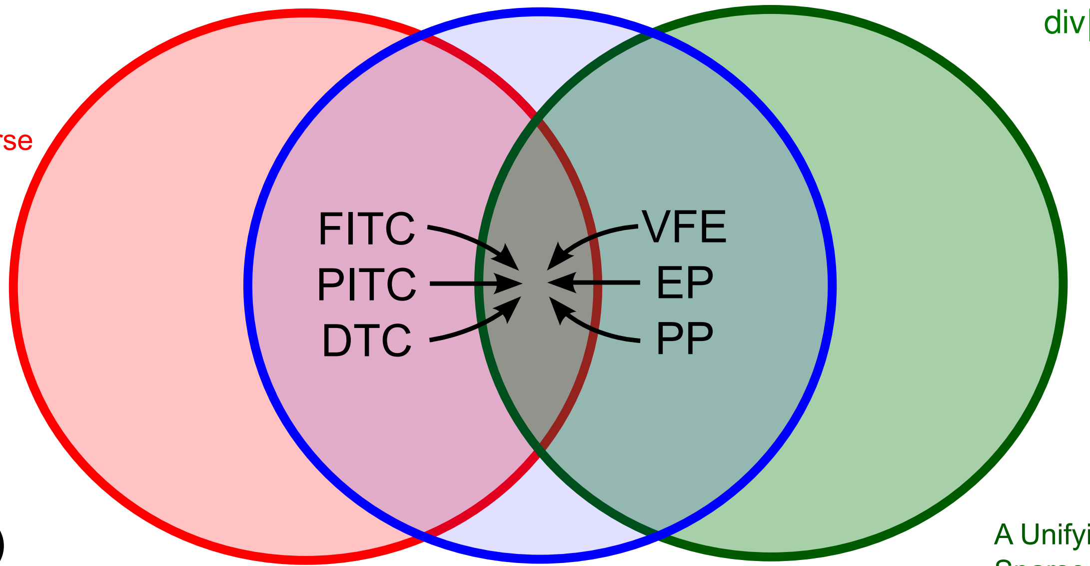
exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)





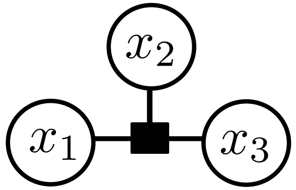
A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

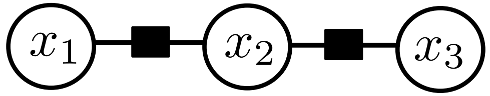
Factor Graphs: reminder (or introduction)

factor graph examples

$$p(x_1, x_2, x_3) = g(x_1, x_2, x_3)$$



$$p(x_1, x_2, x_3) = g_1(x_1, x_2) g_2(x_2, x_3)$$

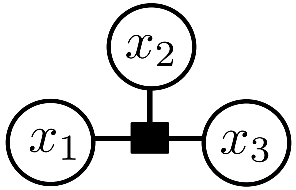


$x_3 \perp x_1 | x_2$

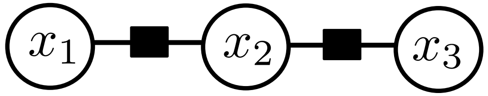
Factor Graphs: reminder (or introduction)

factor graph examples

$$p(x_1, x_2, x_3) = g(x_1, x_2, x_3)$$



$$p(x_1, x_2, x_3) = g_1(x_1, x_2) g_2(x_2, x_3)$$



$$x_3 \perp x_1 | x_2$$

what is the minimal factor graph for this multivariate Gaussian?

$$p(\mathbf{x}|\mu, \Sigma) = \mathcal{N}(\mathbf{x}; \mu, \Sigma)$$

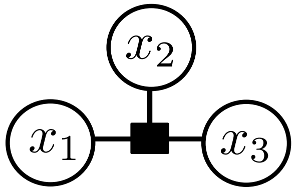
4 dimensional

$$\Sigma = \begin{bmatrix} 1 & 1/2 & 1/2 & 1/4 \\ 1/2 & 5/4 & 1/4 & 1/8 \\ 1/2 & 1/4 & 5/4 & 5/8 \\ 1/4 & 1/8 & 5/8 & 21/16 \end{bmatrix} \quad \Sigma^{-1} = \begin{bmatrix} 1.5 & -1/2 & -1/2 & 0 \\ -1/2 & 1 & 0 & 0 \\ -1/2 & 0 & 5/4 & -1/2 \\ 0 & 0 & -1/2 & 1 \end{bmatrix}$$

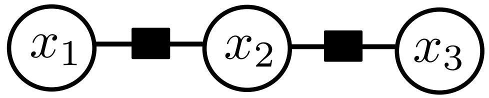
Factor Graphs: reminder (or introduction)

factor graph examples

$$p(x_1, x_2, x_3) = g(x_1, x_2, x_3)$$



$$p(x_1, x_2, x_3) = g_1(x_1, x_2)g_2(x_2, x_3)$$



$x_3 \perp x_1 | x_2$

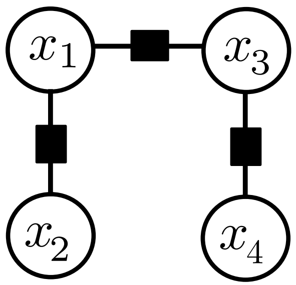
what is the minimal factor graph for this multivariate Gaussian?

$$p(\mathbf{x}|\mu, \Sigma) = \mathcal{N}(\mathbf{x}; \mu, \Sigma)$$

4 dimensional

$$\Sigma = \begin{bmatrix} 1 & 1/2 & 1/2 & 1/4 \\ 1/2 & 5/4 & 1/4 & 1/8 \\ 1/2 & 1/4 & 5/4 & 5/8 \\ 1/4 & 1/8 & 5/8 & 21/16 \end{bmatrix} \quad \Sigma^{-1} = \begin{bmatrix} 1.5 & -1/2 & -1/2 & 0 \\ -1/2 & 1 & 0 & 0 \\ -1/2 & 0 & 5/4 & -1/2 \\ 0 & 0 & -1/2 & 1 \end{bmatrix}$$

solution:



A brief introduction to the Kullback-Leibler divergence

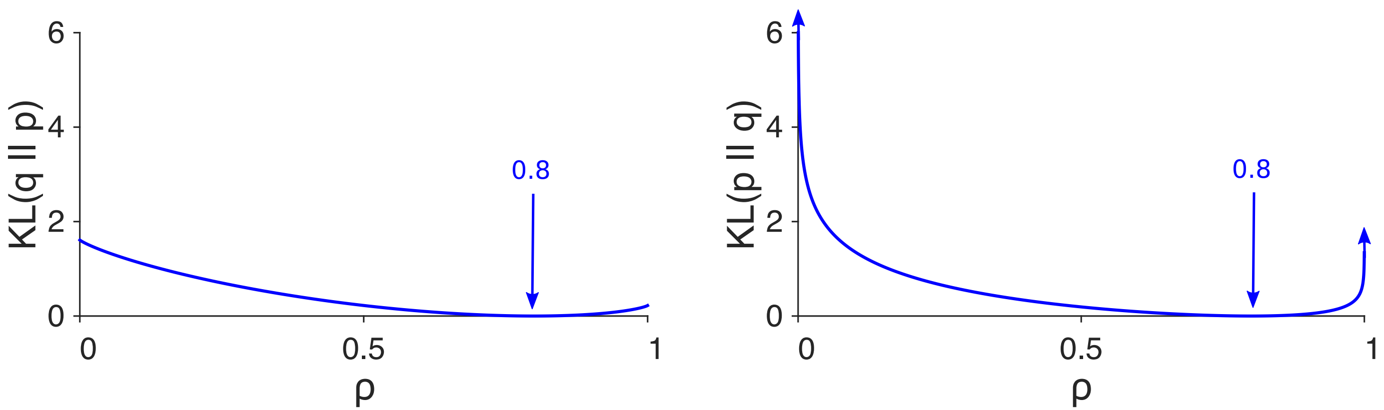
$$\mathcal{KL}(p_1(z)||p_2(z)) = \sum_z p_1(z) \log \frac{p_1(z)}{p_2(z)}$$

Important properties:

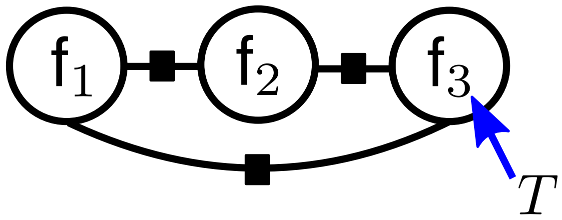
- **Gibb's inequality**: $\mathcal{KL}(p_1(z)||p_2(z)) \geq 0$, equality at $p_1(z) = p_2(z)$
  - proof via Jensen's inequality or differentiation (see slide at end)
- **Non-symmetric**: $\mathcal{KL}(p_1(z)||p_2(z)) \neq \mathcal{KL}(p_2(z)||p_1(z))$
  - hence named *divergence* and not *distance*

Example:

- binary variables $z \in \{0, 1\}$
- $p(z = 1) = 0.8$ and $q(z = 1) = \rho$



Fully independent training conditional (FITC) approximation

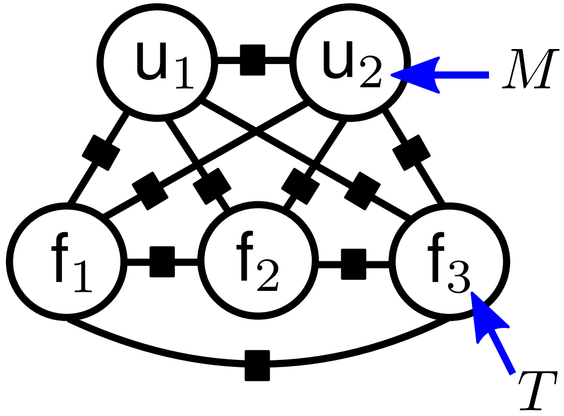


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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

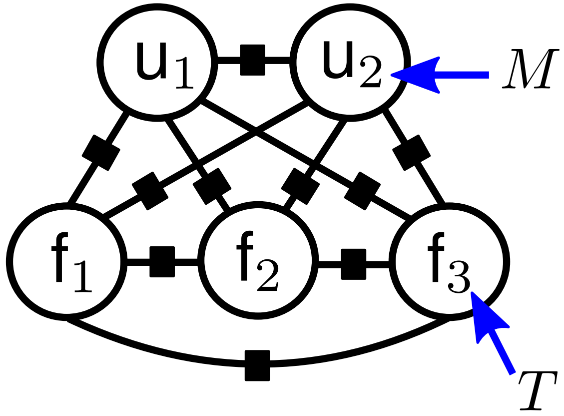


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

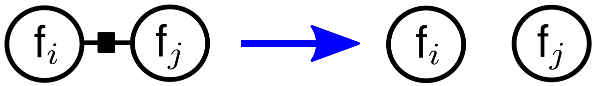
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



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



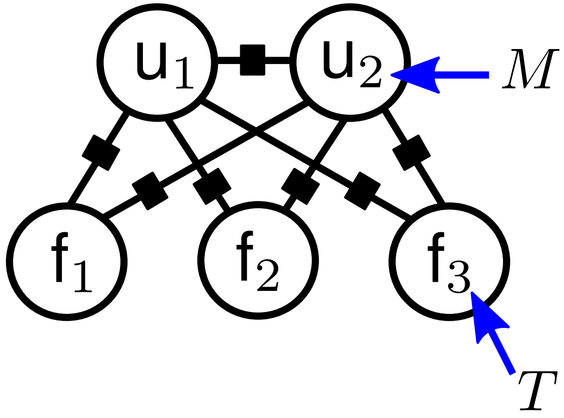
all factors

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

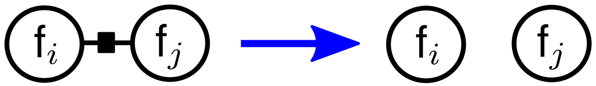
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



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



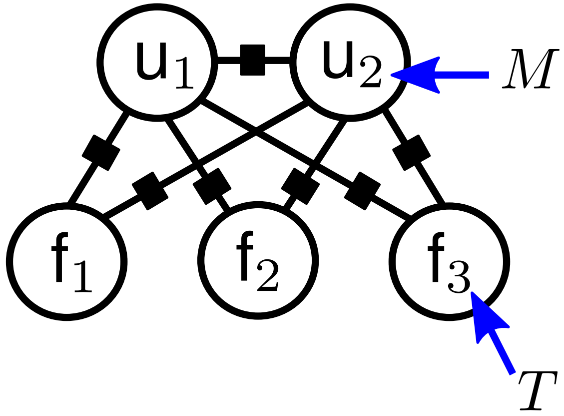
all factors

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

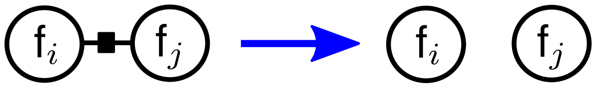
Fully independent training conditional (FITC) approximation

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

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

$$\underset{q(\mathbf{u}),\{q(\mathsf{f}_t|\mathbf{u})\}_{t=1}^{T}}{\arg\min}\ \mathsf{KL}\left(p(\mathbf{f},\mathbf{u})\,||\,q(\mathbf{u})\prod_{t=1}^{T} q(\mathsf{f}_t|\mathbf{u})\right) \implies \begin{array}{c} q(\mathbf{u}) = p(\mathbf{u}) \\ q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u}) \end{array}$$

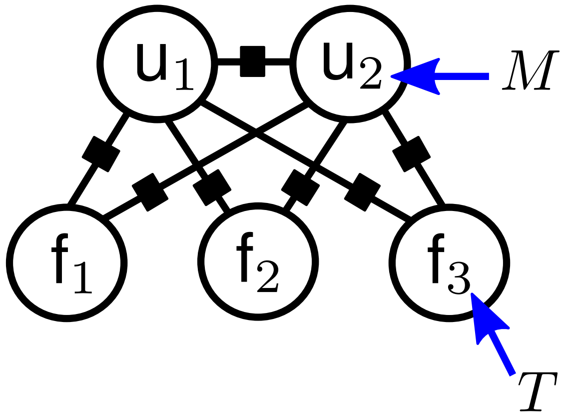
equal to exact conditionals

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

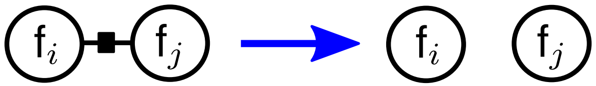
Fully independent training conditional (FITC) approximation

1. augment model with M<T pseudo data

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



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



all factors

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

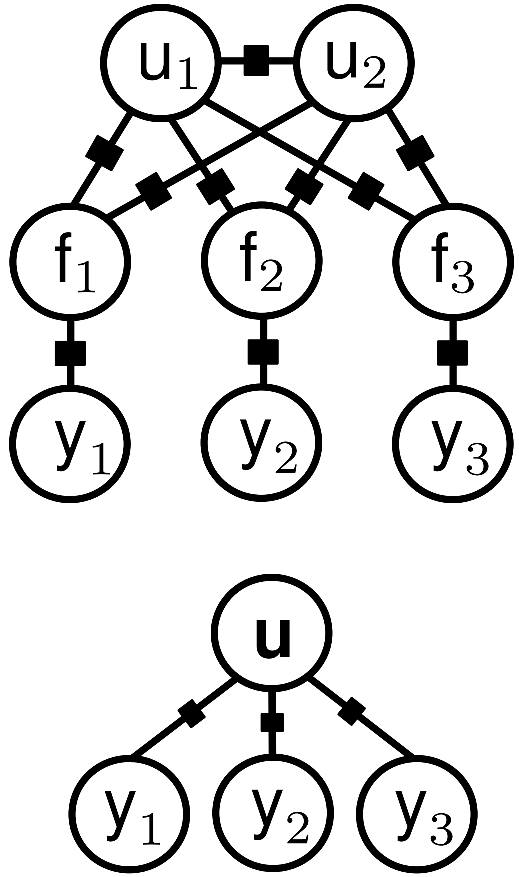
$$\underset{q(\mathbf{u}),\{q(f_t|\mathbf{u})\}_{t=1}^{T}}{\arg\min} \mathrm{KL}\left(p(\mathbf{f},\mathbf{u})\,||\,q(\mathbf{u})\prod_{t=1}^{T} q(f_t|\mathbf{u})\right) \implies \begin{matrix} q(\mathbf{u}) = p(\mathbf{u}) \\ q(f_t|\mathbf{u}) = p(f_t|\mathbf{u}) \end{matrix}$$

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

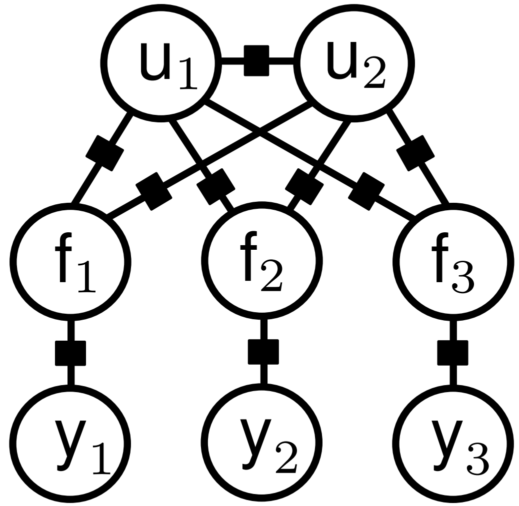


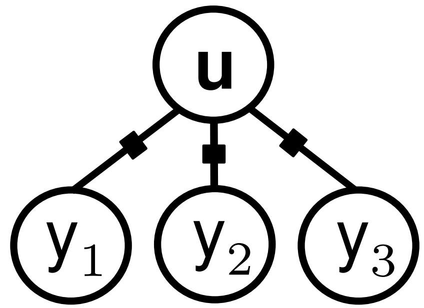
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

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





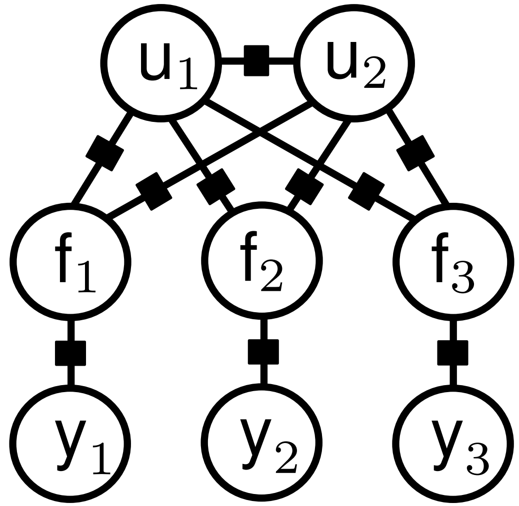
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

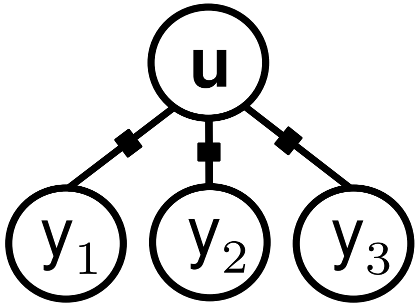
indirect posterior approximation

Fully independent training conditional (FITC) approximation

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

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





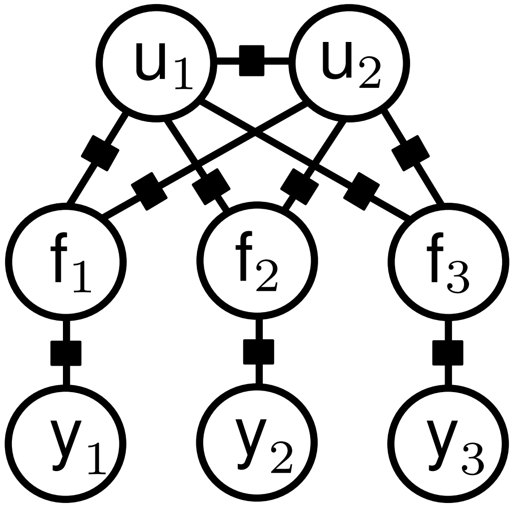
construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

Fully independent training conditional (FITC) approximation

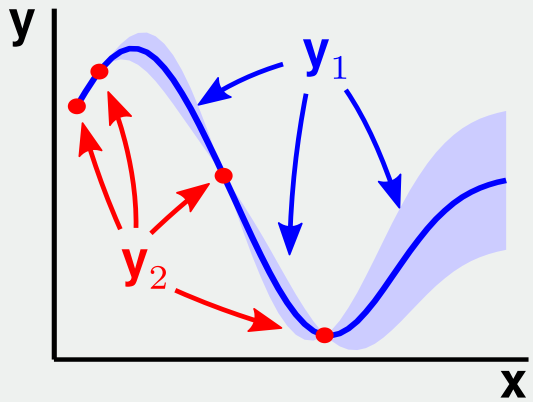
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

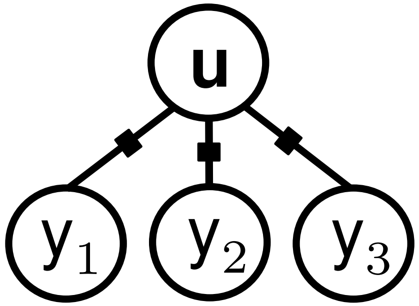
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$



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

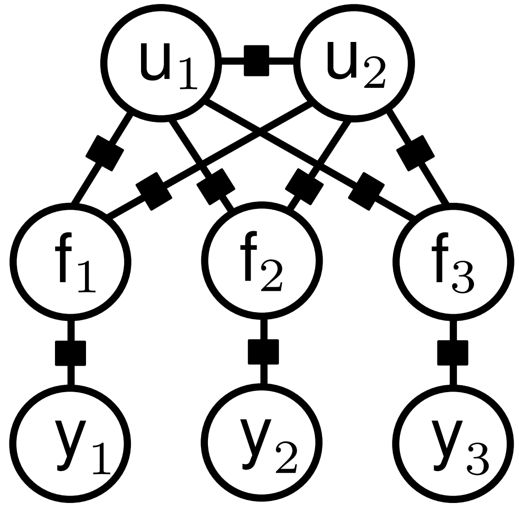
indirect posterior approximation

Fully independent training conditional (FITC) approximation

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

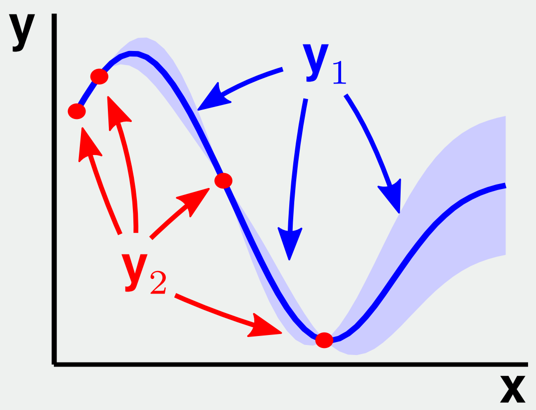
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

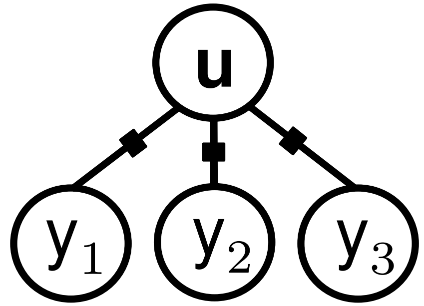
$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t})$$



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

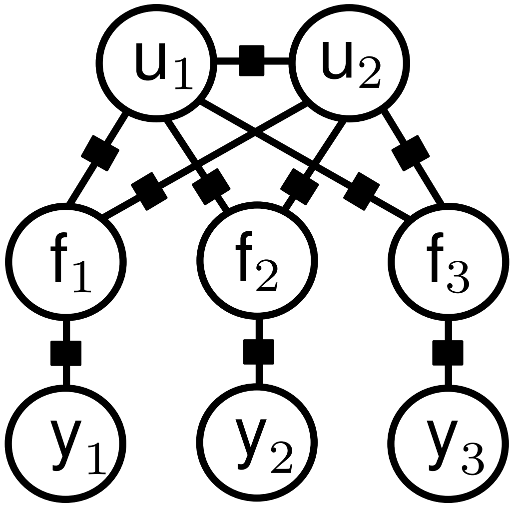
indirect posterior approximation

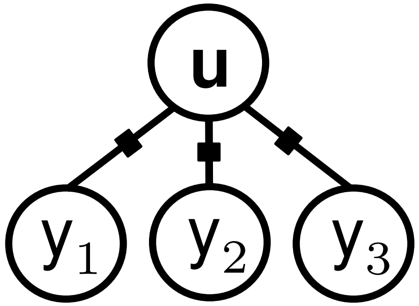
Fully independent training conditional (FITC) approximation

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

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

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





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

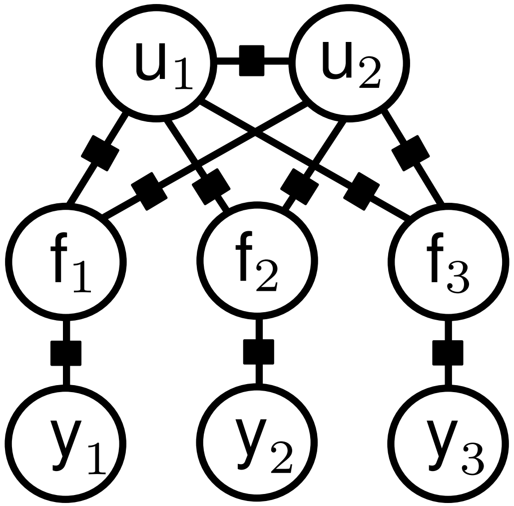
indirect posterior approximation

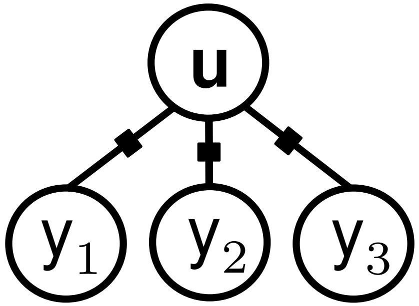
Fully independent training conditional (FITC) approximation

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

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

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





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

indirect posterior approximation

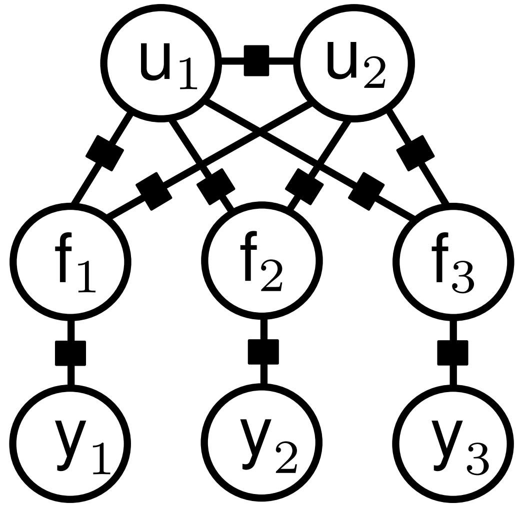
Fully independent training conditional (FITC) approximation

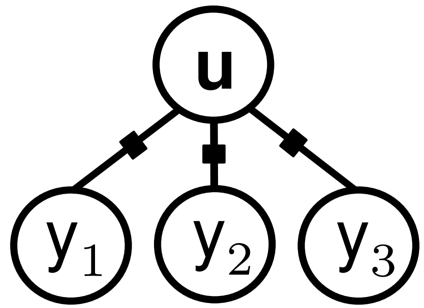
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

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

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

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





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

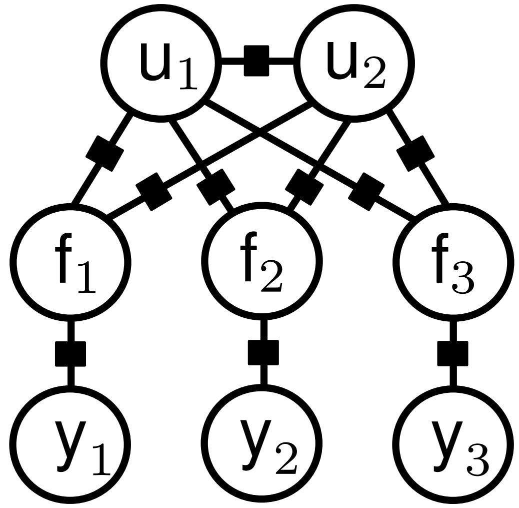
$$q(\mathbf{u}) = p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}})$$

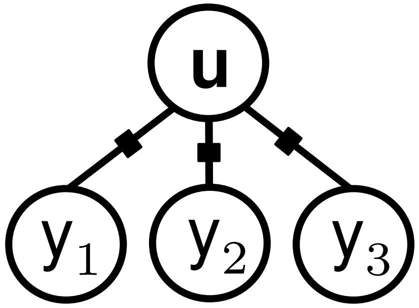
$$q(\mathsf{f}_t|\mathbf{u}) = p(\mathsf{f}_t|\mathbf{u})$$

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

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

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

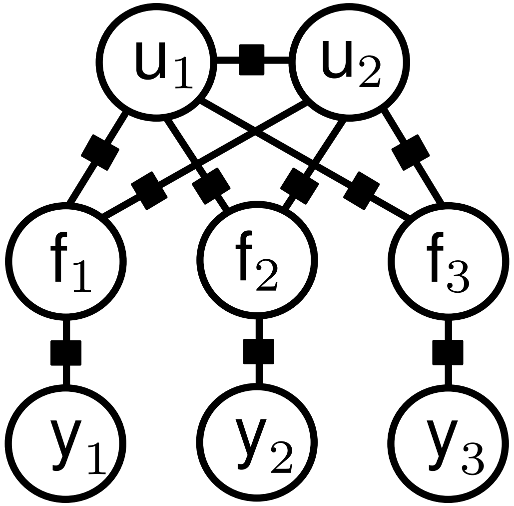
Fully independent training conditional (FITC) approximation

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

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

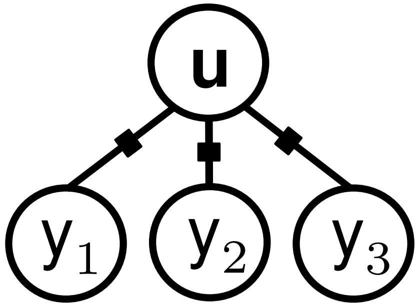
$$= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}})$$

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



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

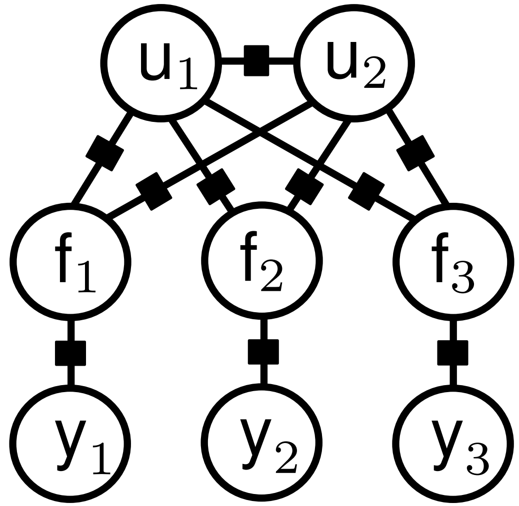
Fully independent training conditional (FITC) approximation

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

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

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

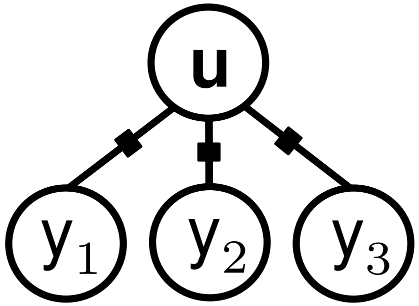
$$q(\mathsf{y}_t|\mathsf{f}_t) = p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)$$



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

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

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

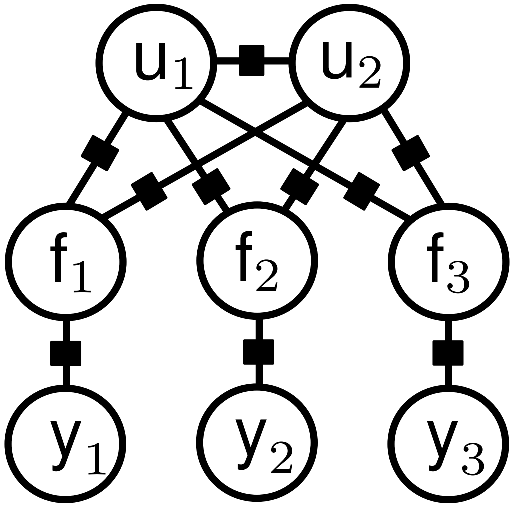


construct new generative model (with pseudo-data)
cheaper to perform exact learning and inference
calibrated to original

indirect posterior approximation

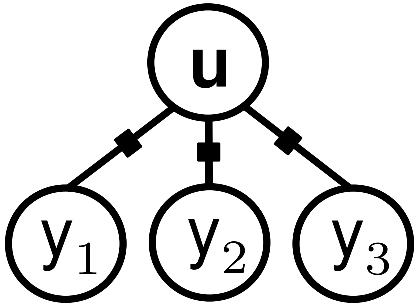
Fully independent training conditional (FITC) approximation

$$
\begin{aligned}
q(\mathbf{u}) &= p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; 0, \mathsf{K}_{\mathsf{uu}}) \\
q(\mathsf{f}_t|\mathbf{u}) &= p(\mathsf{f}_t|\mathbf{u}) \\
&= \mathcal{N}(\mathsf{f}_t; \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathbf{u}, \underbrace{\mathsf{K}_{\mathsf{f}_t\mathsf{f}_t} - \mathsf{K}_{\mathsf{f}_t\mathsf{u}}\mathsf{K}_{\mathsf{uu}}^{-1}\mathsf{K}_{\mathsf{uf}_t}}_{\mathsf{D}_{tt}}) \\
q(\mathsf{y}_t|\mathsf{f}_t) &= p(\mathsf{y}_t|\mathsf{f}_t) = \mathcal{N}(\mathsf{y}_t; \mathsf{f}_t, \sigma_{\mathsf{y}}^2)
\end{aligned}
$$



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

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

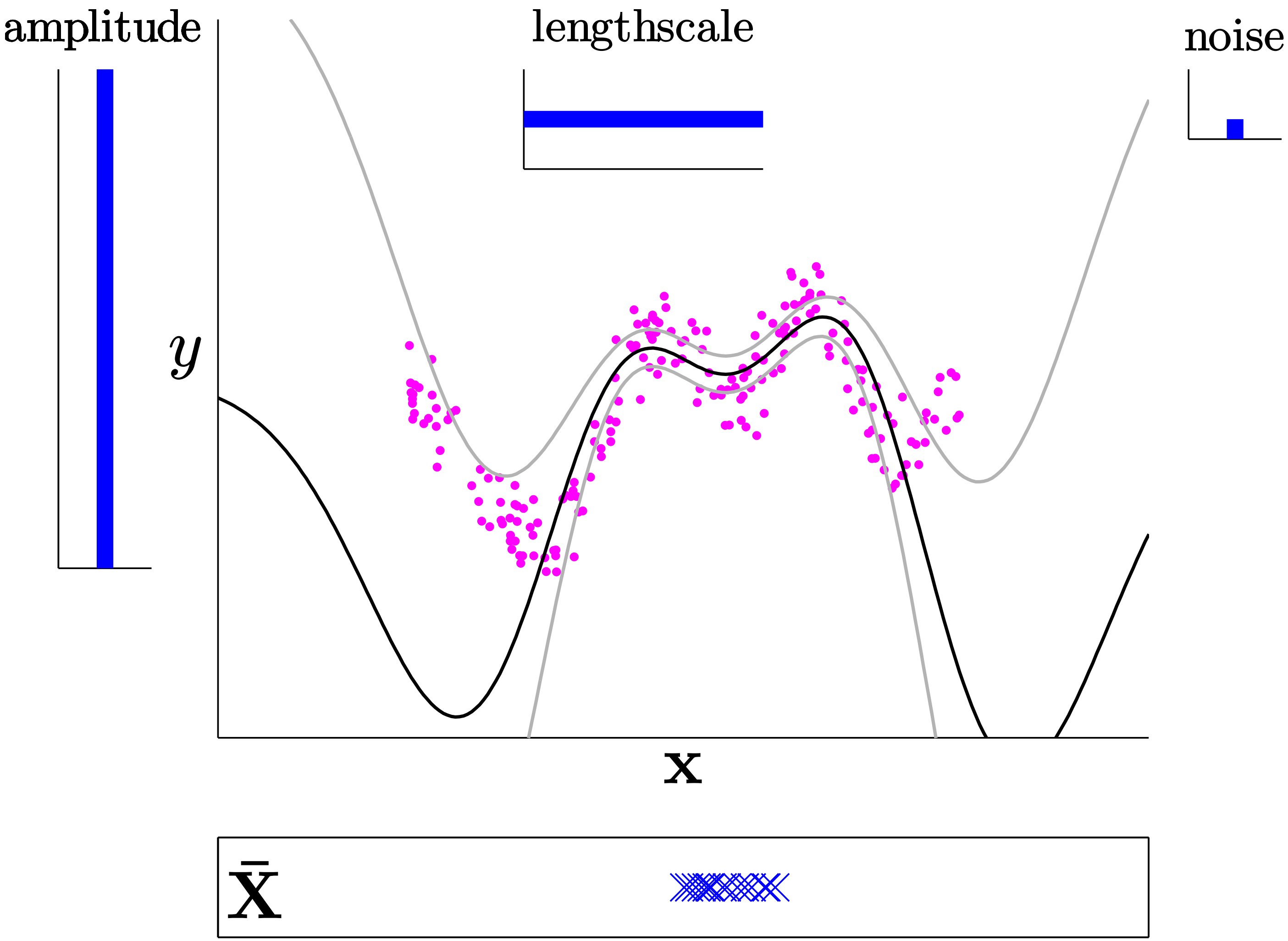


original variances along diagonal: stops variances collapsing

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

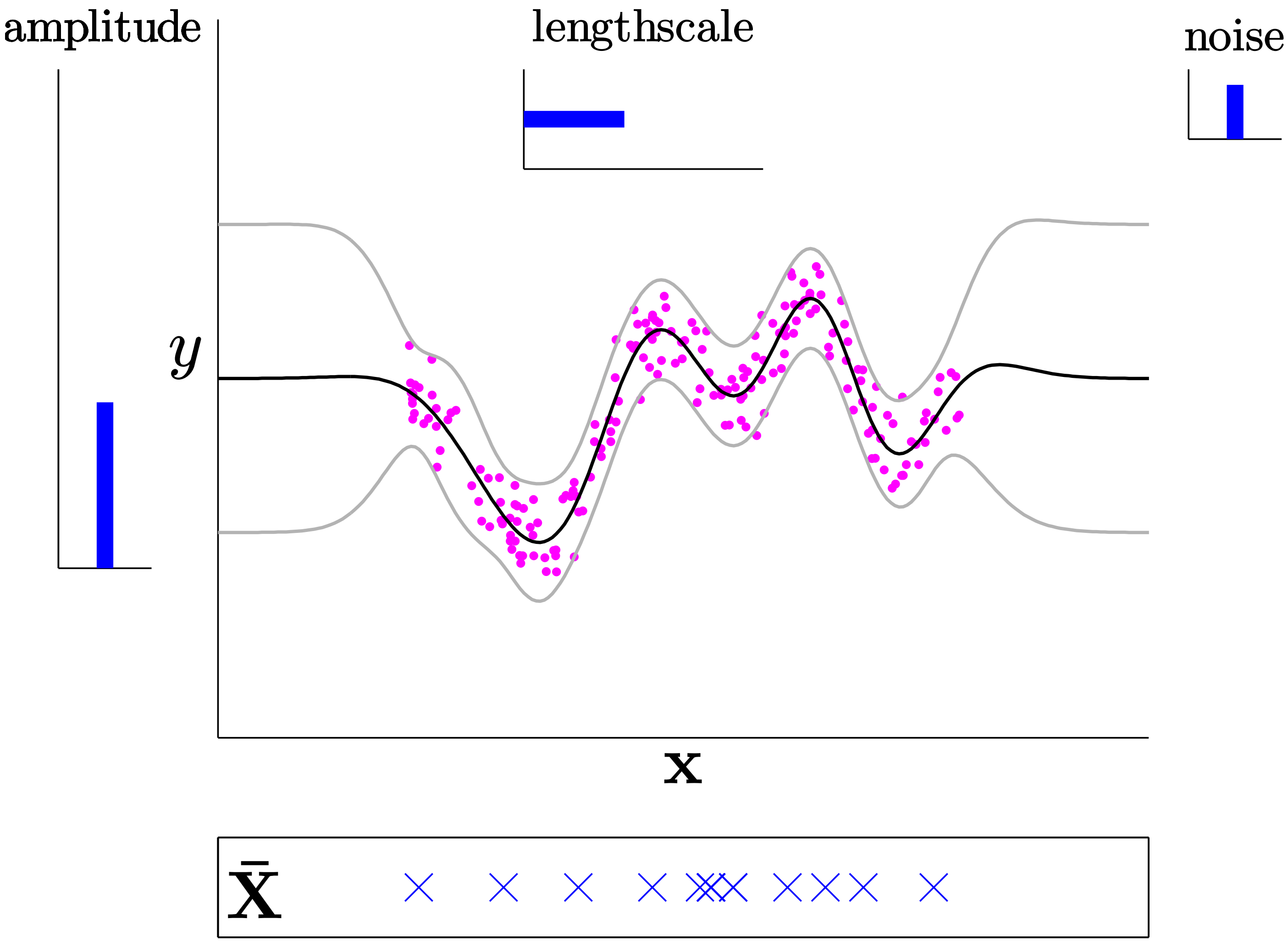
indirect posterior approximation

FITC: Demo (Snelson)



Initialize adversarially: amplitude and lengthscale too big
noise too small
pseudo-inputs bunched up

FITC: Demo (Snelson)



Pseudo-inputs and hyperparameters optimized

Fully independent training conditional (FITC) approximation

- introduces parametric bottleneck into non-parametric model (although in a clever way)
- if I see more data, should I add extra pseudo-data?
  - unnatural from a generative modelling perspective
  - natural from a prediction perspective (posterior gets more complex)

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

- example of **prior approximation**

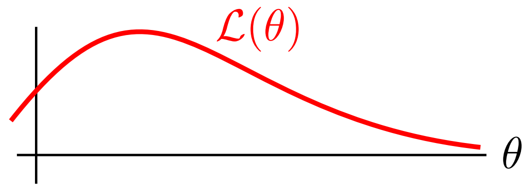
Extensions:

- methods for optimising pseudo-inputs (indirect approximations tend to over-fit)
- partially independent training conditional and tree-structured approximations (see extra slides)

Variational free-energy method (VFE)

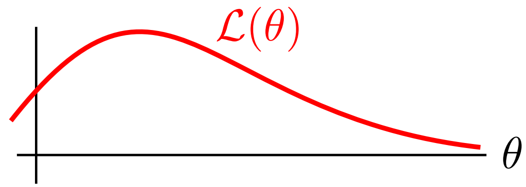
lower bound the likelihood

$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta)$$



Variational free-energy method (VFE)

lower bound the likelihood

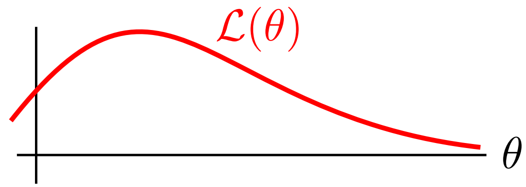


$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)}$$

Variational free-energy method (VFE)

lower bound the likelihood

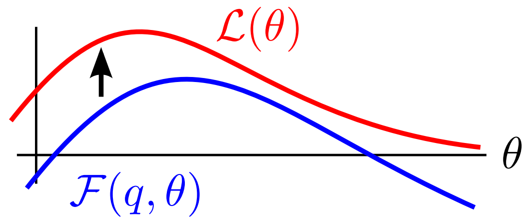


$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)}$$

Variational free-energy method (VFE)

lower bound the likelihood

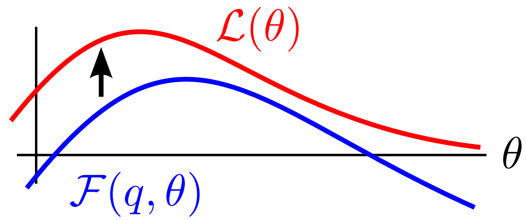


$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathsf{d}f\, p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathsf{d}f\, p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathsf{d}f\, q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

Variational free-energy method (VFE)

lower bound the likelihood



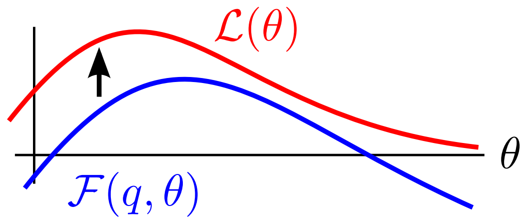
$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

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

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta)$$

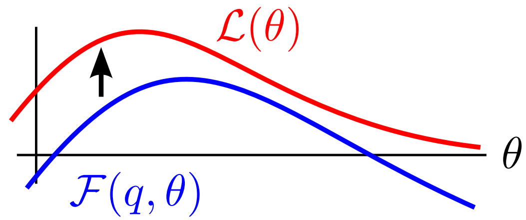
$$= \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(f|\mathbf{y}, \theta) p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(f|\mathbf{y}, \theta) p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

assume approximate posterior factorisation with special form

$$q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u})$$

exact: $q(f_{\neq \mathbf{u}}|\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{y}, \mathbf{u})$

Variational free-energy method (VFE)

$$\mathcal{F}(\theta) = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

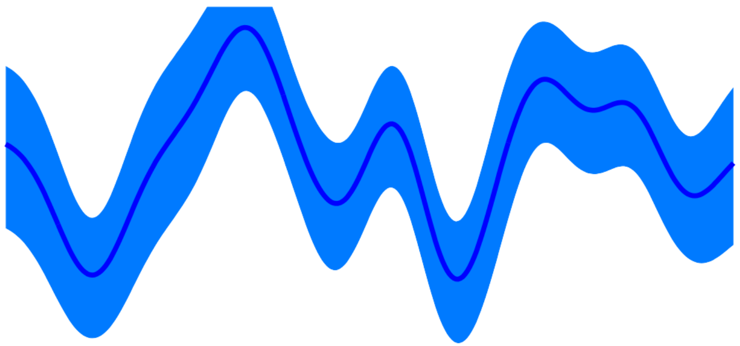
approximate posterior

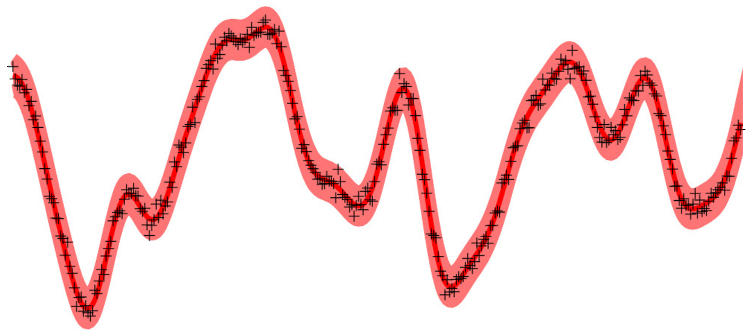
$$q(f) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

true posterior

$$p(f|\mathbf{y})$$

KL





Variational free-energy method (VFE)

$$\mathcal{F}(\theta) \;=\; \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

same form as prediction
from GP-regression

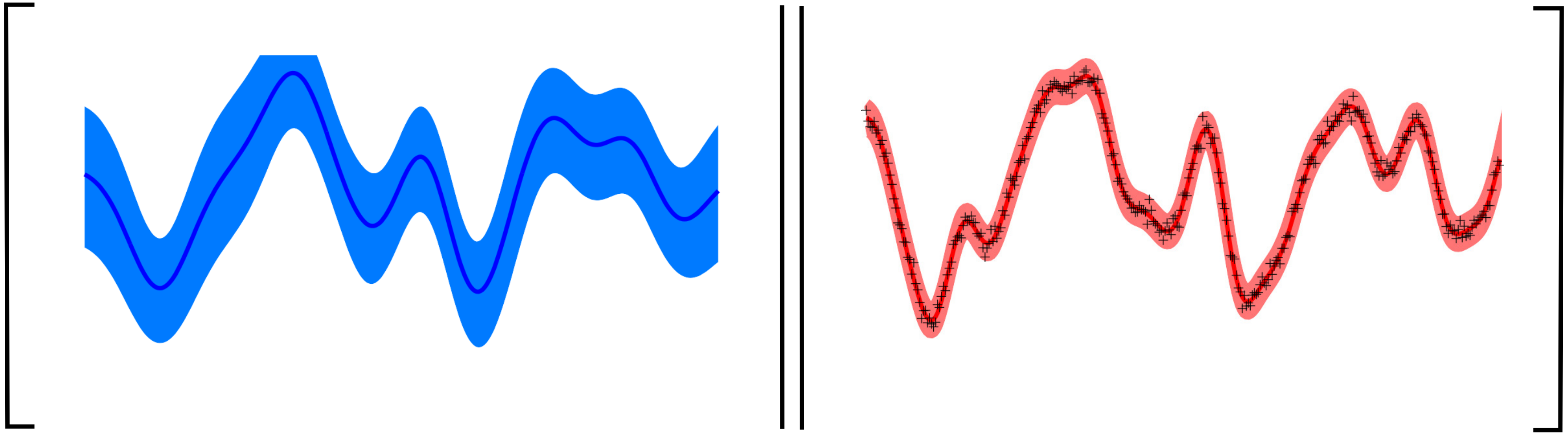
approximate posterior

$$q(f) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

true posterior

$$p(f|\mathbf{y})$$

KL



Variational free-energy method (VFE)

$$\mathcal{F}(\theta) = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

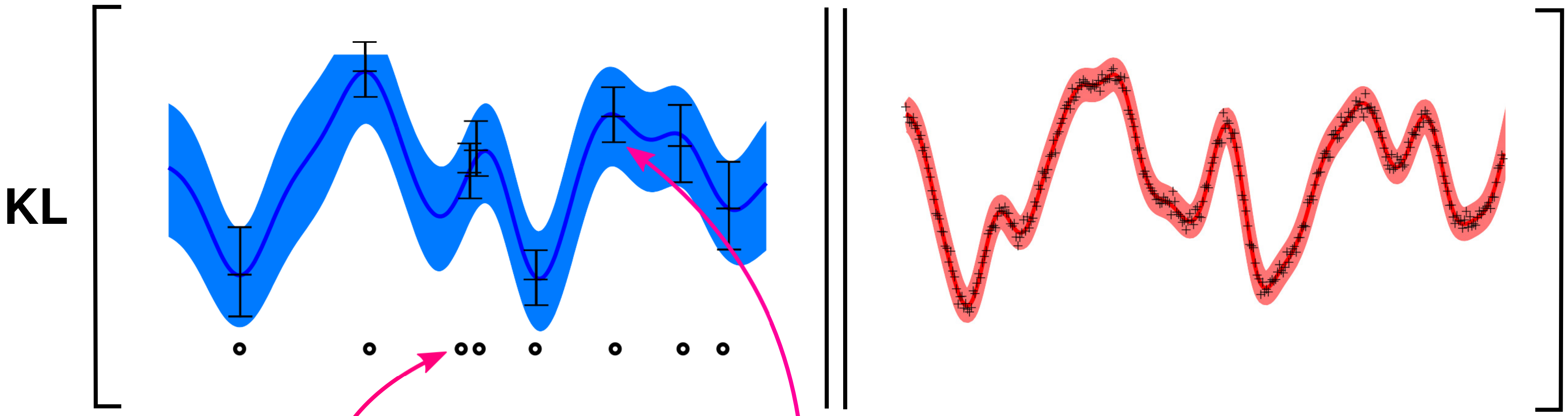
same form as prediction
from GP-regression

approximate posterior

$$q(f) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

true posterior

$$p(f|\mathbf{y})$$



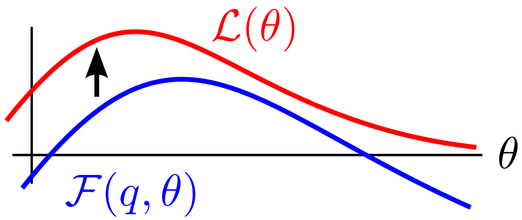
inputs locations of
'pseudo' data

output locations
and covariance
'pseudo' data

optimise variational free-energy wrt to these variational parameters

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(f|\mathbf{y},\theta)p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

assume approximate posterior factorisation with special form

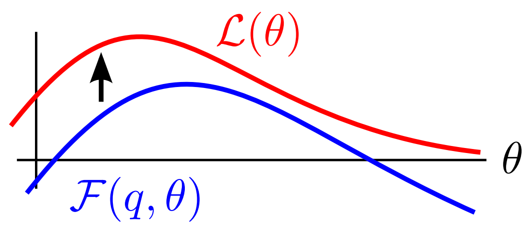
$$q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

predictive from GP regression

exact: $q(f_{\neq \mathbf{u}}|\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{y}, \mathbf{u})$

Variational free-energy method (VFE)

lower bound the likelihood



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

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

$$\mathcal{F}(\theta) = \int \mathbf{d}f \; q(f) \log \frac{p(f|\mathbf{y},\theta)p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

assume approximate posterior factorisation with special form

$$q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

predictive from GP regression

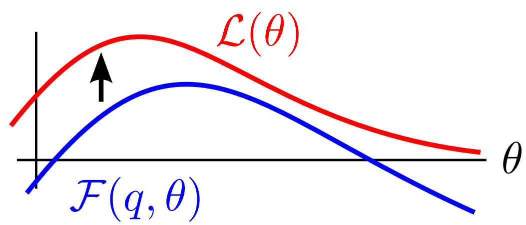
exact: $q(f_{\neq \mathbf{u}}|\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{y}, \mathbf{u})$

plug into Free-energy:

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

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathsf{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

$$\mathcal{F}(\theta) = \int \mathsf{d}f \; q(f) \log \frac{p(f|\mathbf{y},\theta)p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

assume approximate posterior factorisation with special form

$$q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$$

predictive from GP regression

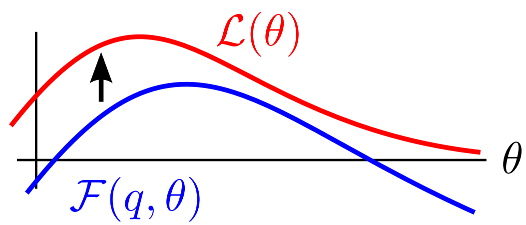
exact: $q(f_{\neq \mathbf{u}}|\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{y}, \mathbf{u})$

plug into Free-energy:

$$\mathcal{F}(\theta) = \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})} = \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f},\theta)p(f_{\neq \mathbf{u}}|\mathbf{u})p(\mathbf{u})}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})}$$

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{L}(\theta) = \log p(\mathbf{y}|\theta) = \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta)$$

$$= \log \int \mathrm{d}f \; p(\mathbf{y}, f|\theta) \frac{q(f)}{q(f)} \geq \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{q(f)} = \mathcal{F}(\theta)$$

$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(f|\mathbf{y}, \theta) p(\mathbf{y}|\theta)}{q(f)} = \log p(\mathbf{y}|\theta) - \mathbf{KL}(q(f)||p(f|\mathbf{y}))$$

KL between stochastic processes

assume approximate posterior factorisation with special form

$$q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u})$$

predictive from GP regression

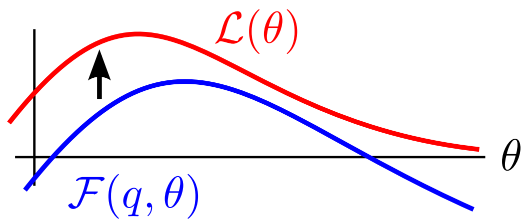
exact: $q(f_{\neq \mathbf{u}}|\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{y}, \mathbf{u})$

plug into Free-energy:

$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u})} = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f}, \theta) \cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})} p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})} q(\mathbf{u})}$$

Variational free-energy method (VFE)

lower bound the likelihood

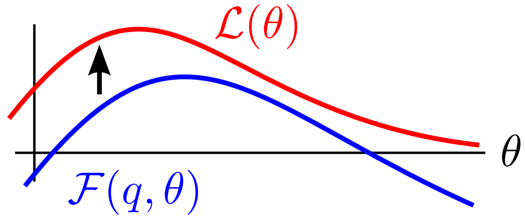


$$\mathcal{F}(\theta) = \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u})} = \int \mathsf{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f}, \theta)\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}q(\mathbf{u})}$$

where $q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u}) q(\mathbf{u})$

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})} = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f},\theta)\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}q(\mathbf{u})}$$

where $q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$

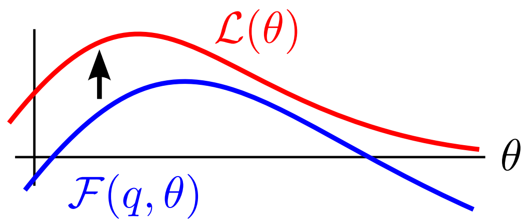
$$\mathcal{F}(\theta) = \langle \log p(\mathbf{y}|\mathbf{f},\theta) \rangle_{q(f)} - \mathbf{KL}(q(\mathbf{u})||p(\mathbf{u}))$$

average of
quadratic form

KL between two
multivariate Gaussians

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})} = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f},\theta)\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}q(\mathbf{u})}$$

where $q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$

$$\mathcal{F}(\theta) = \langle \log p(\mathbf{y}|\mathbf{f},\theta)\rangle_{q(f)} - \mathbf{KL}(q(\mathbf{u})||p(\mathbf{u}))$$

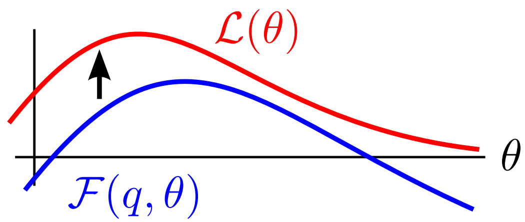
average of quadratic form

KL between two multivariate Gaussians

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

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{F}(\theta) = \int \mathrm{d}f\; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})} = \int \mathrm{d}f\; q(f) \log \frac{p(\mathbf{y}|\mathbf{f},\theta)\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}q(\mathbf{u})}$$

where $q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$

$$\mathcal{F}(\theta) = \langle \log p(\mathbf{y}|\mathbf{f},\theta)\rangle_{q(f)} - \mathbf{KL}(q(\mathbf{u})||p(\mathbf{u}))$$

average of quadratic form

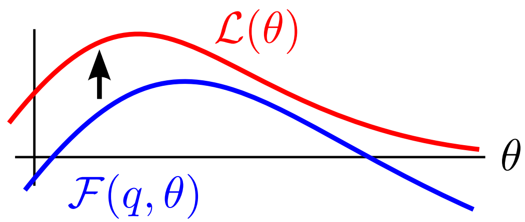
KL between two multivariate Gaussians

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

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

Variational free-energy method (VFE)

lower bound the likelihood



$$\mathcal{F}(\theta) = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}, f|\theta)}{p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})} = \int \mathrm{d}f \; q(f) \log \frac{p(\mathbf{y}|\mathbf{f},\theta)\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}p(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}}|\mathbf{u})}q(\mathbf{u})}$$

where $q(f) = q(\mathbf{u}, f_{\neq \mathbf{u}}) = q(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u}) = p(f_{\neq \mathbf{u}}|\mathbf{u})q(\mathbf{u})$

$$\mathcal{F}(\theta) = \langle \log p(\mathbf{y}|\mathbf{f},\theta)\rangle_{q(f)} - \mathbf{KL}(q(\mathbf{u})||p(\mathbf{u}))$$

average of quadratic form

KL between two multivariate Gaussians

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)

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

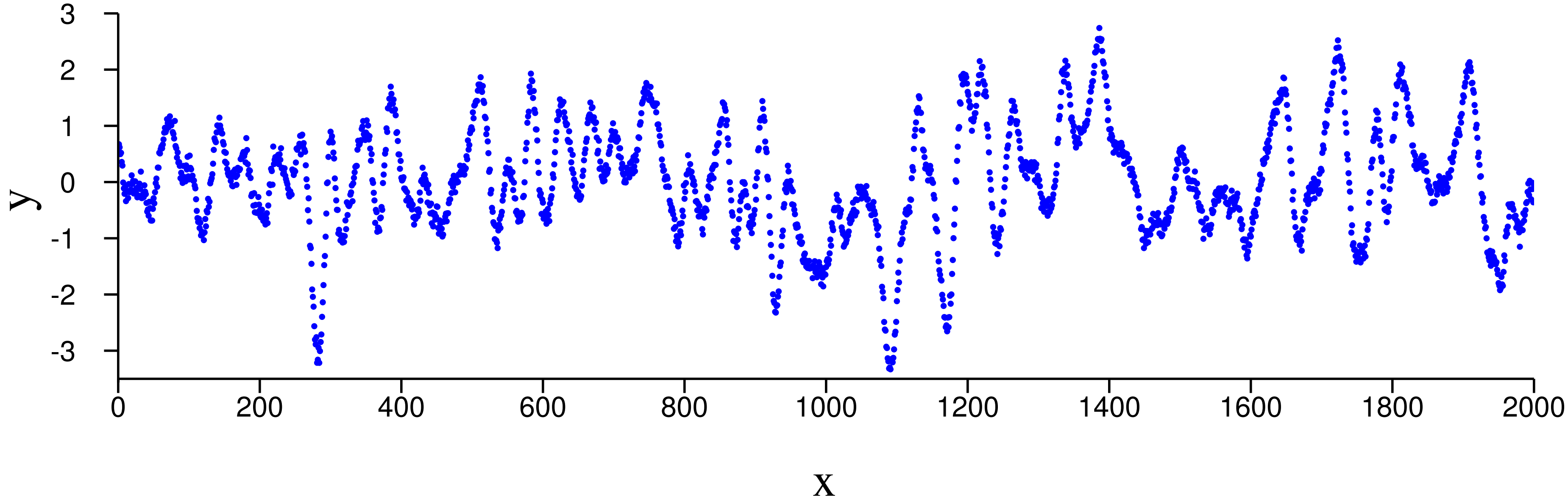
DTC like

uncertainty based correction

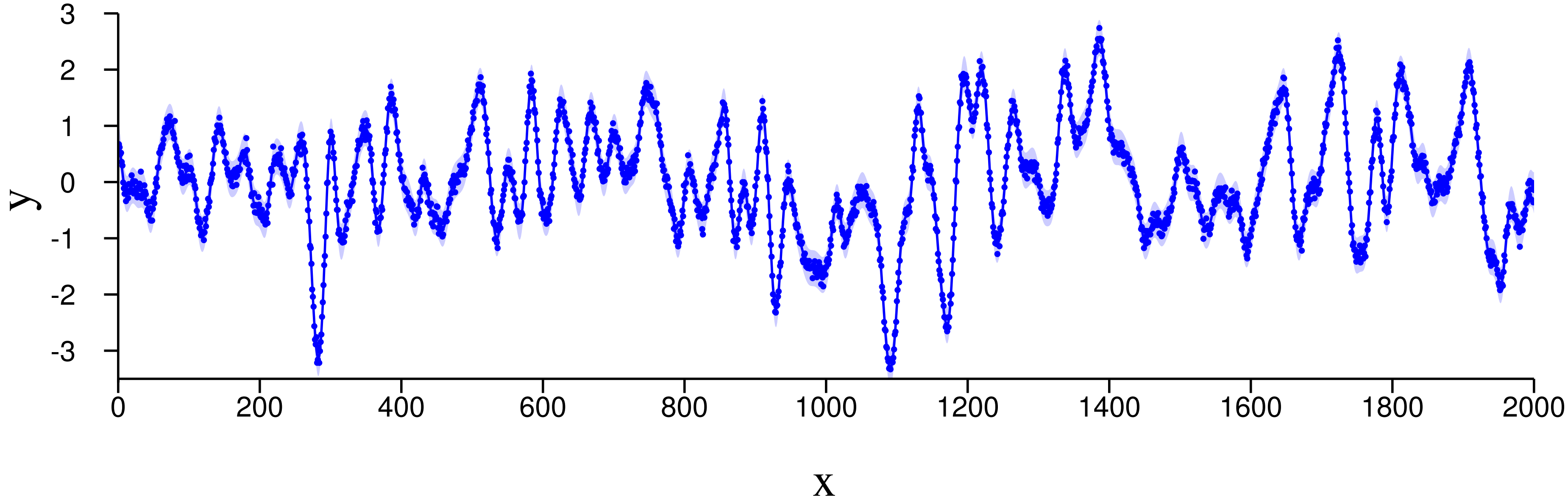
Summary of VFE method

- optimisation pseudo point inputs **better behaved** in VFE methods (direct posterior approximation)
- variational methods known to **underfit** (and have other **biases**)
- **no augmentation required: target is posterior over functions, which includes inducing variables**
  - pseudo-input locations are pure variational parameters (do not parameterise the generative mdoel)
  - coherent way of adding pseudo-data: more complex posteriors require more computational resources (more pseudo-points)
- Curious observation:
  VFE returns better mean estimates
  FITC returns better error-bar estimates
- **how should we select M = number of pseudo-points?**

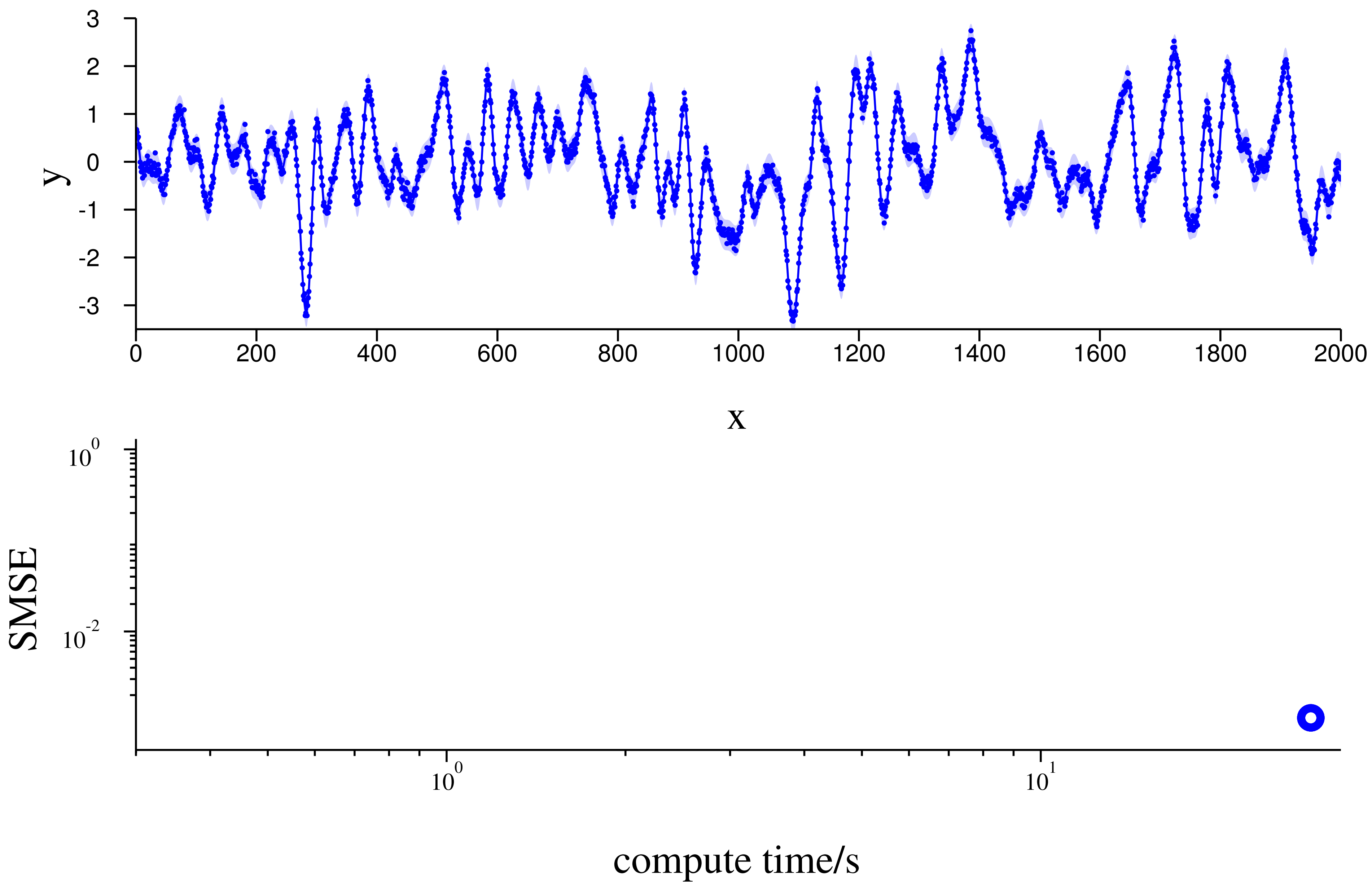
How do we select M = number of pseudo-data?



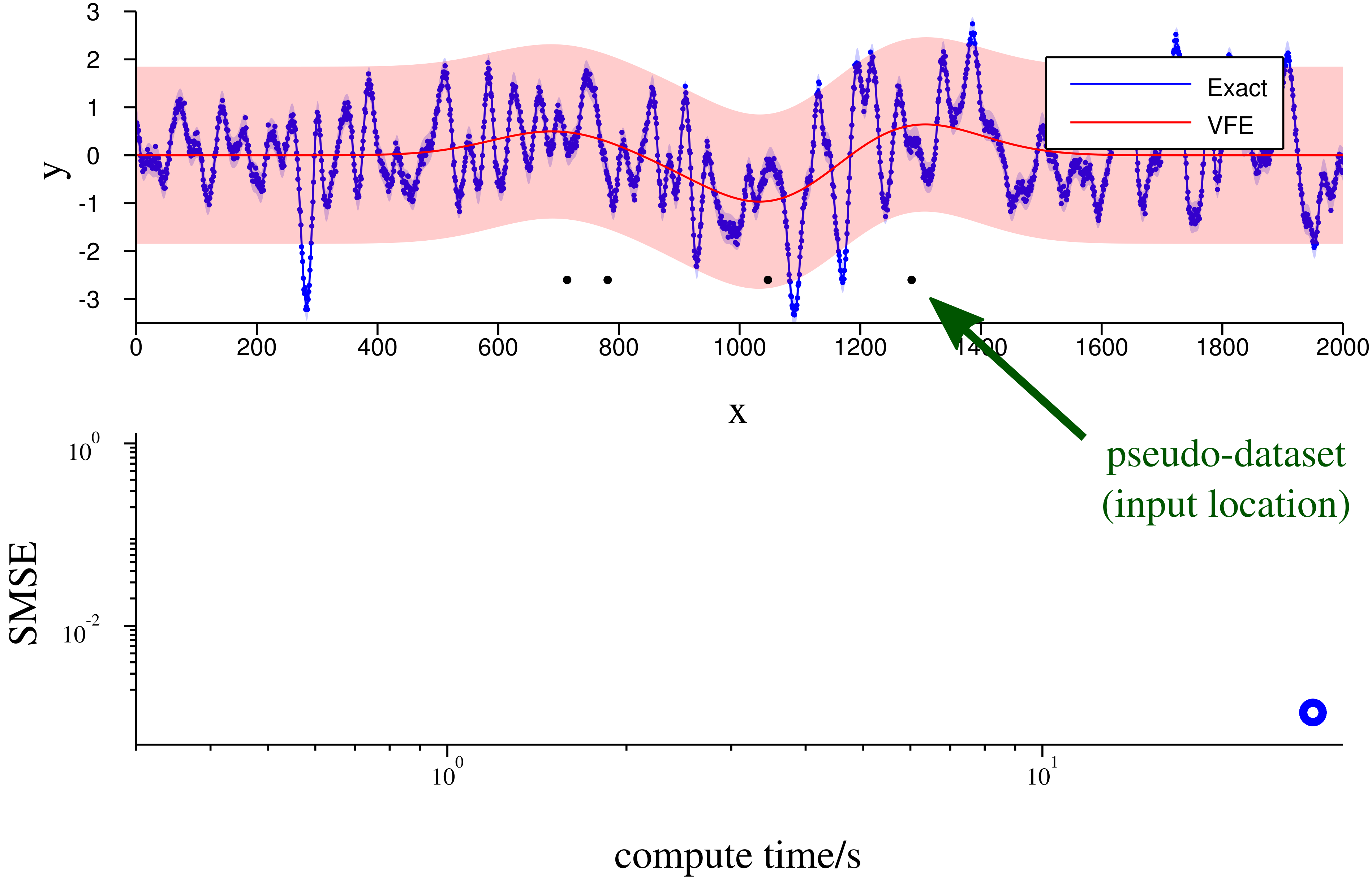
How do we select M = number of pseudo-data?



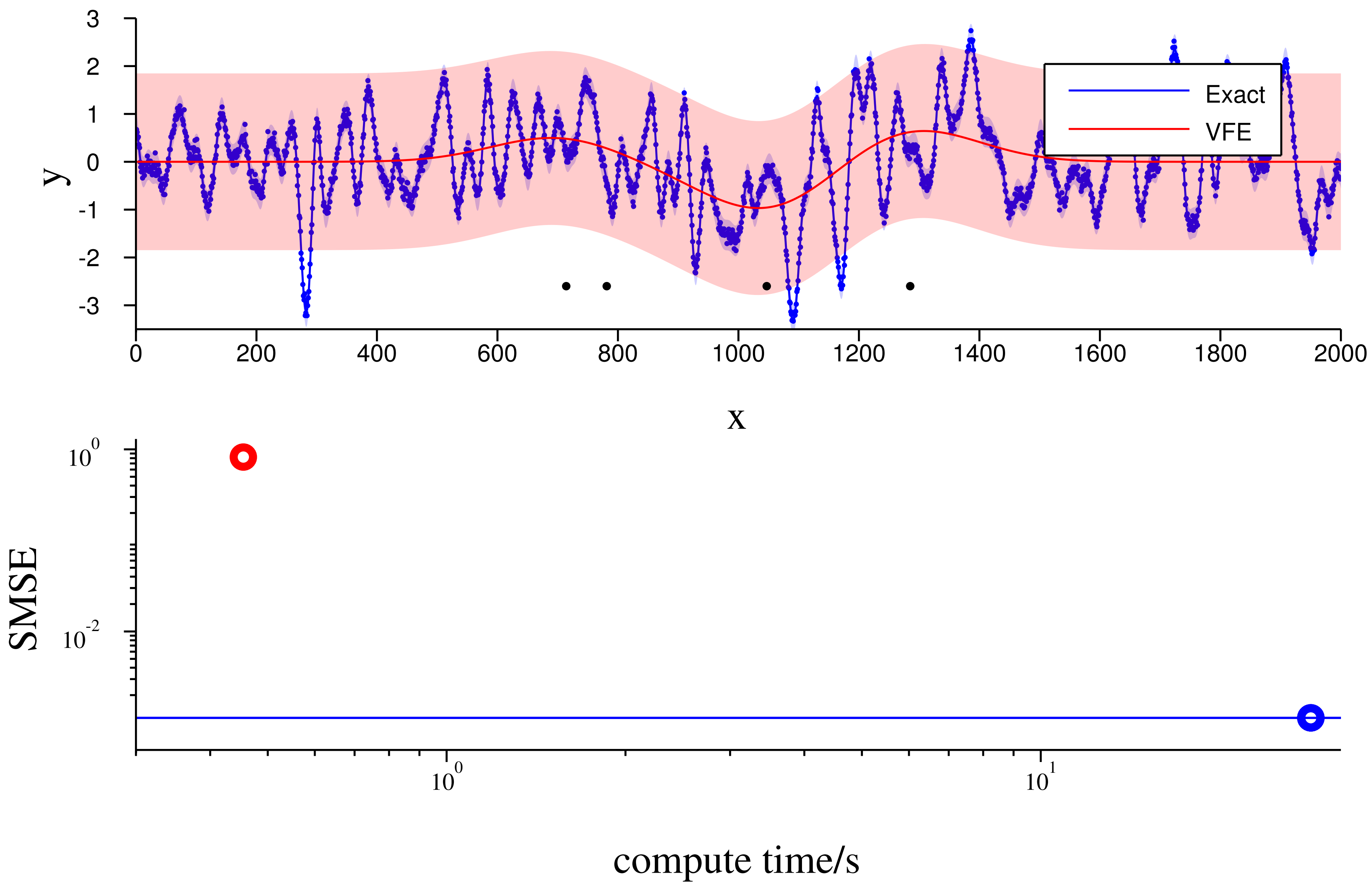
How do we select M = number of pseudo-data?



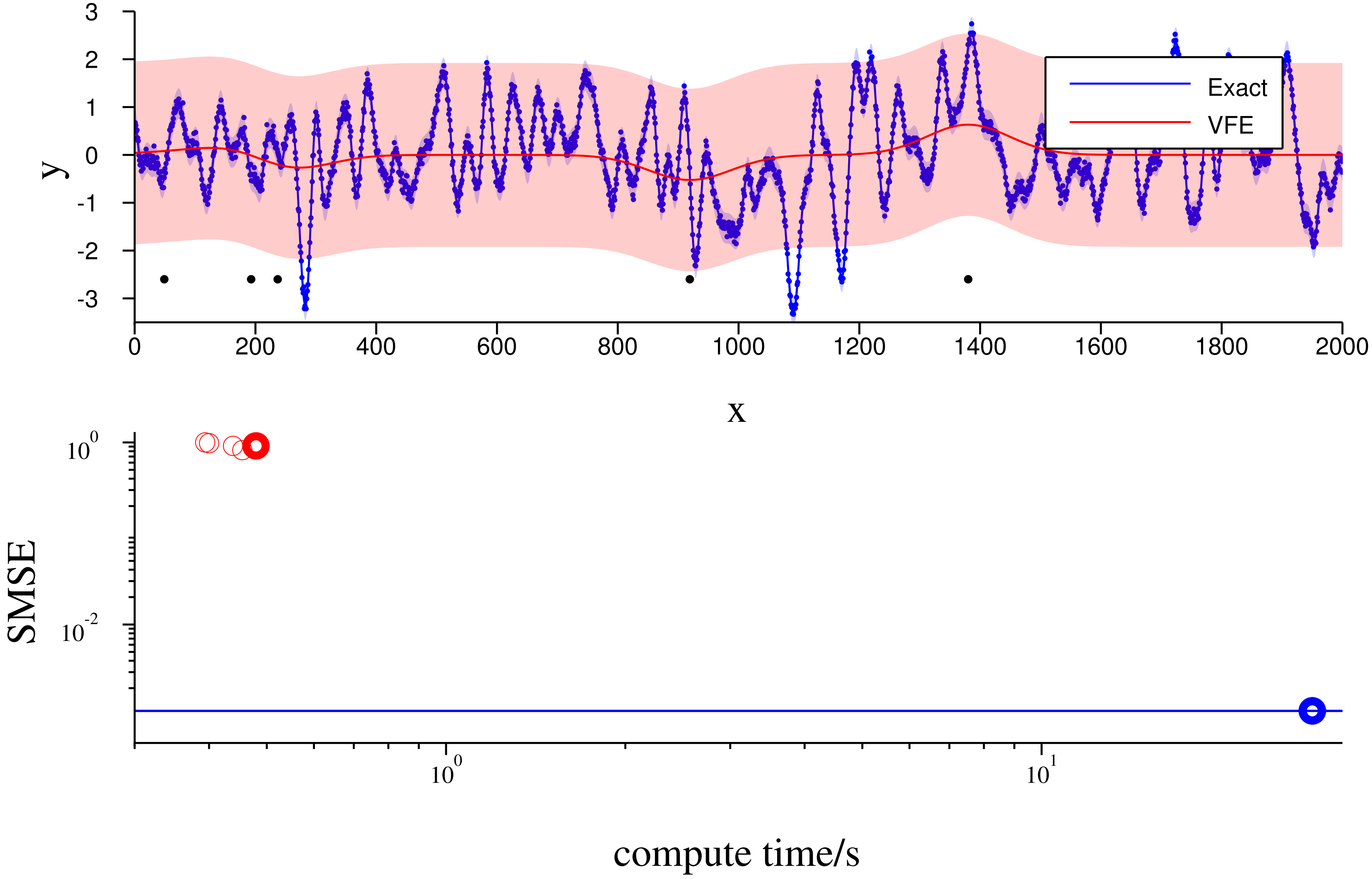
How do we select M = number of pseudo-data?



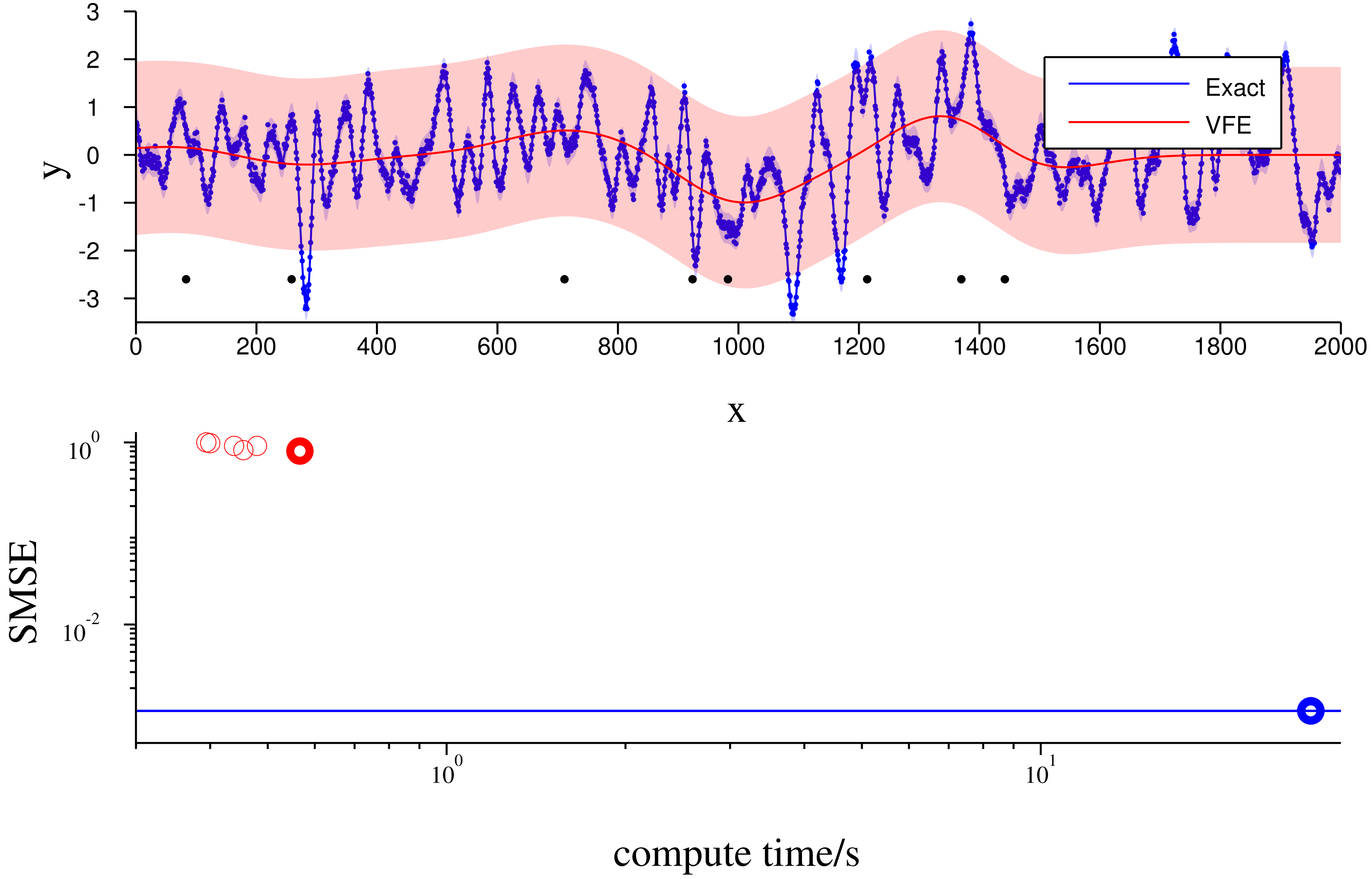
How do we select M = number of pseudo-data?



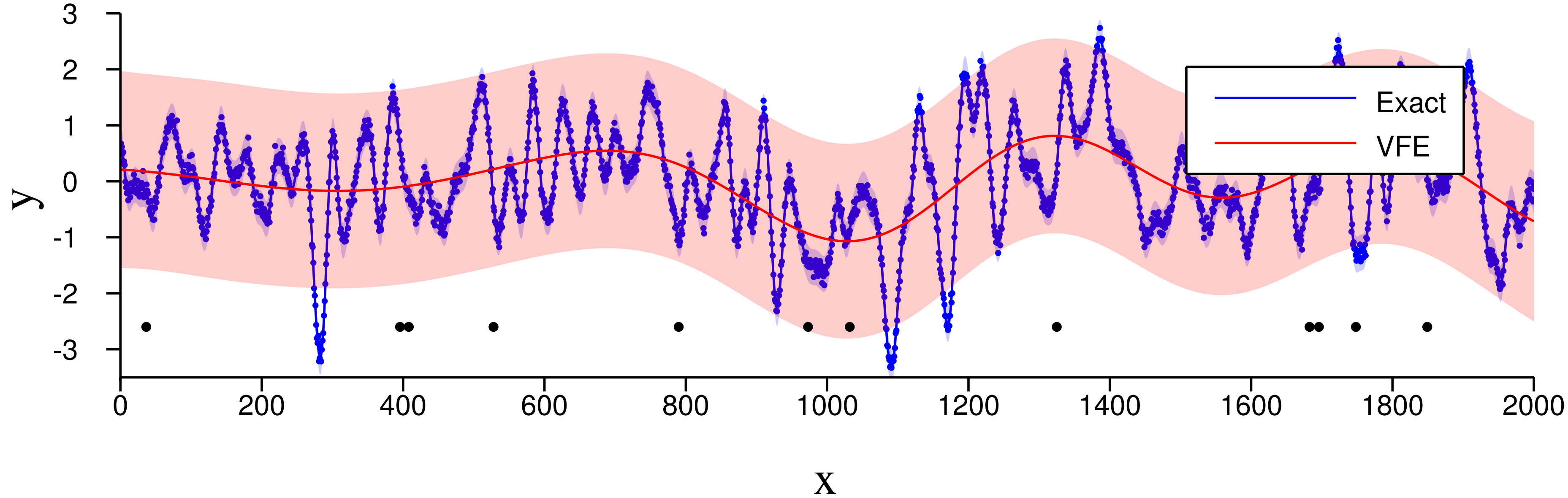
How do we select M = number of pseudo-data?

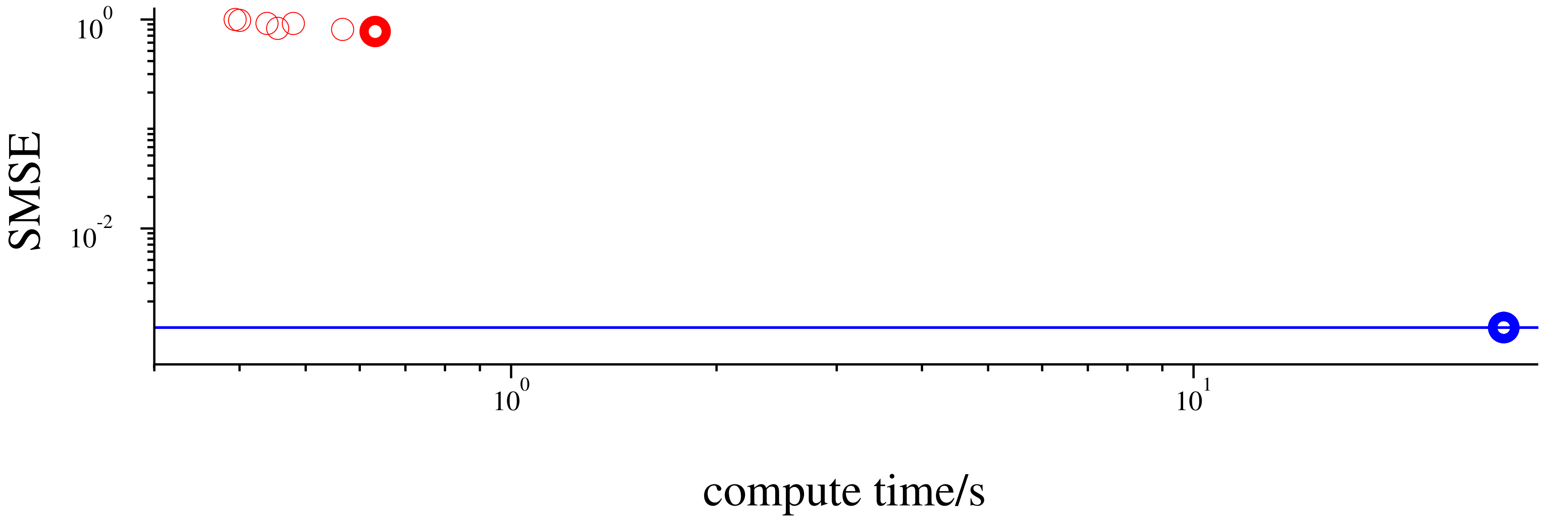


How do we select M = number of pseudo-data?

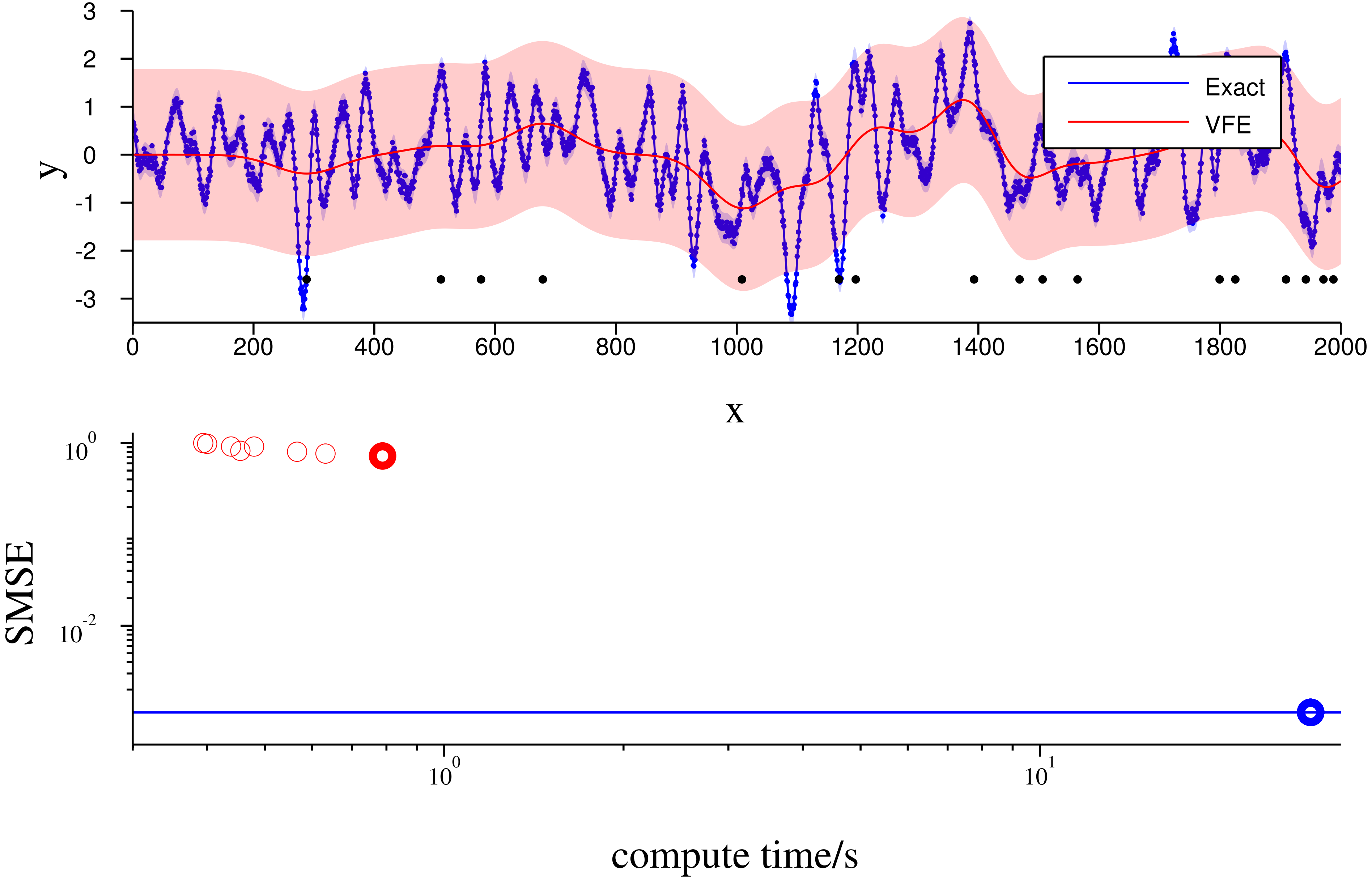


How do we select M = number of pseudo-data?

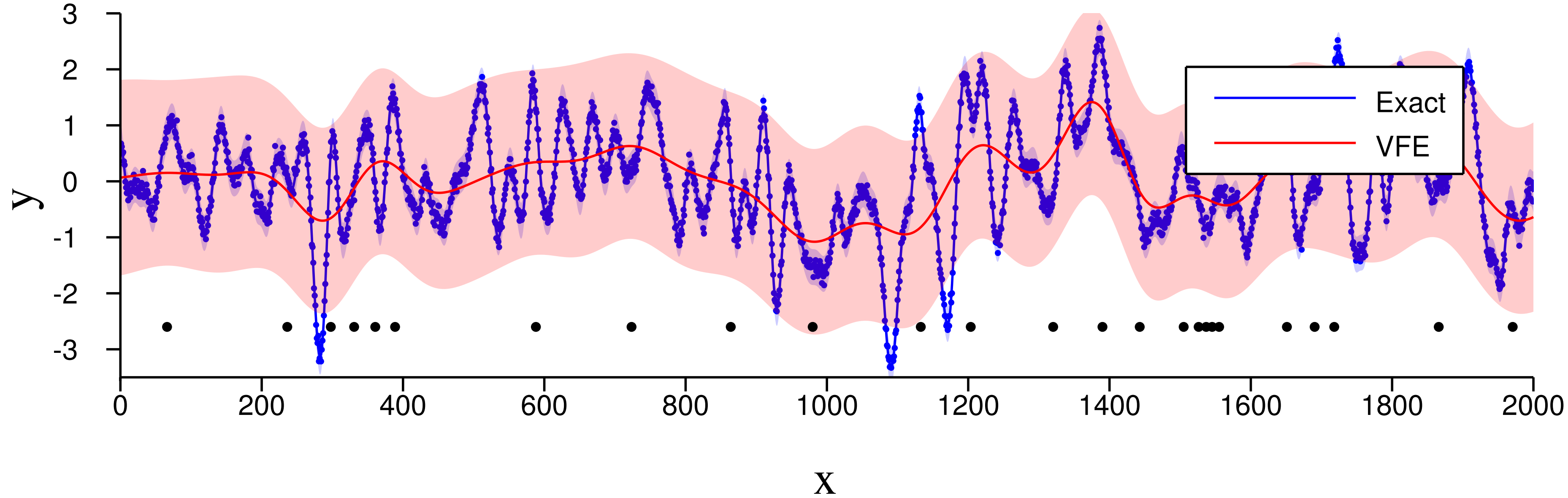


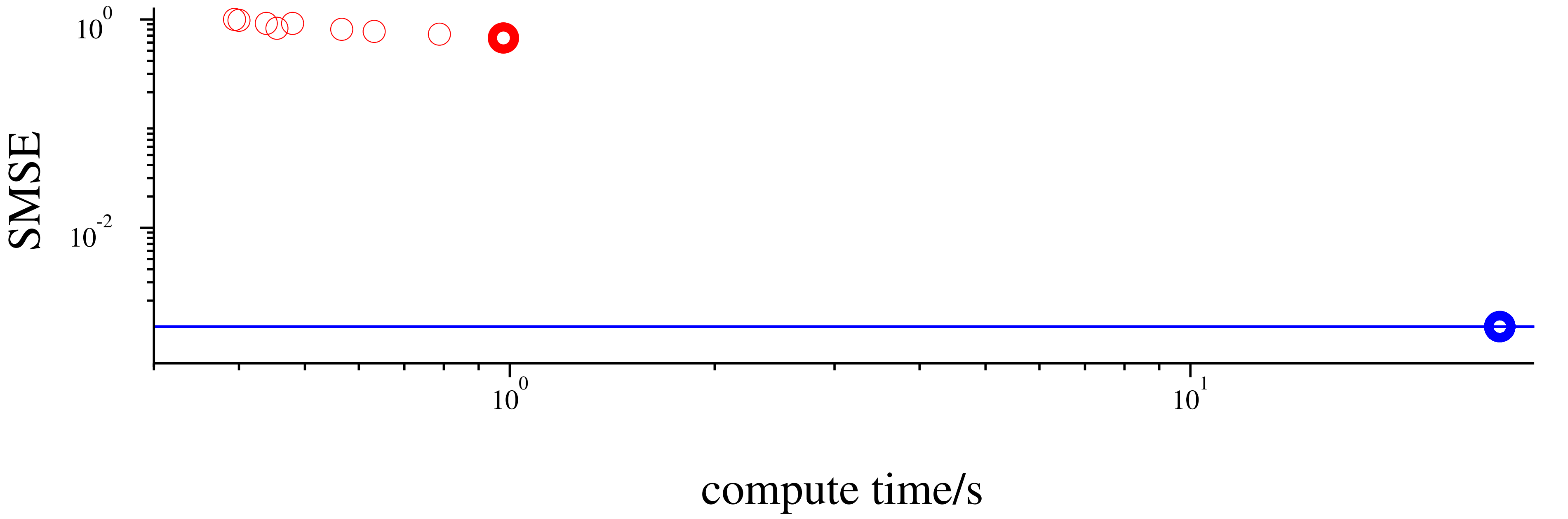


How do we select M = number of pseudo-data?

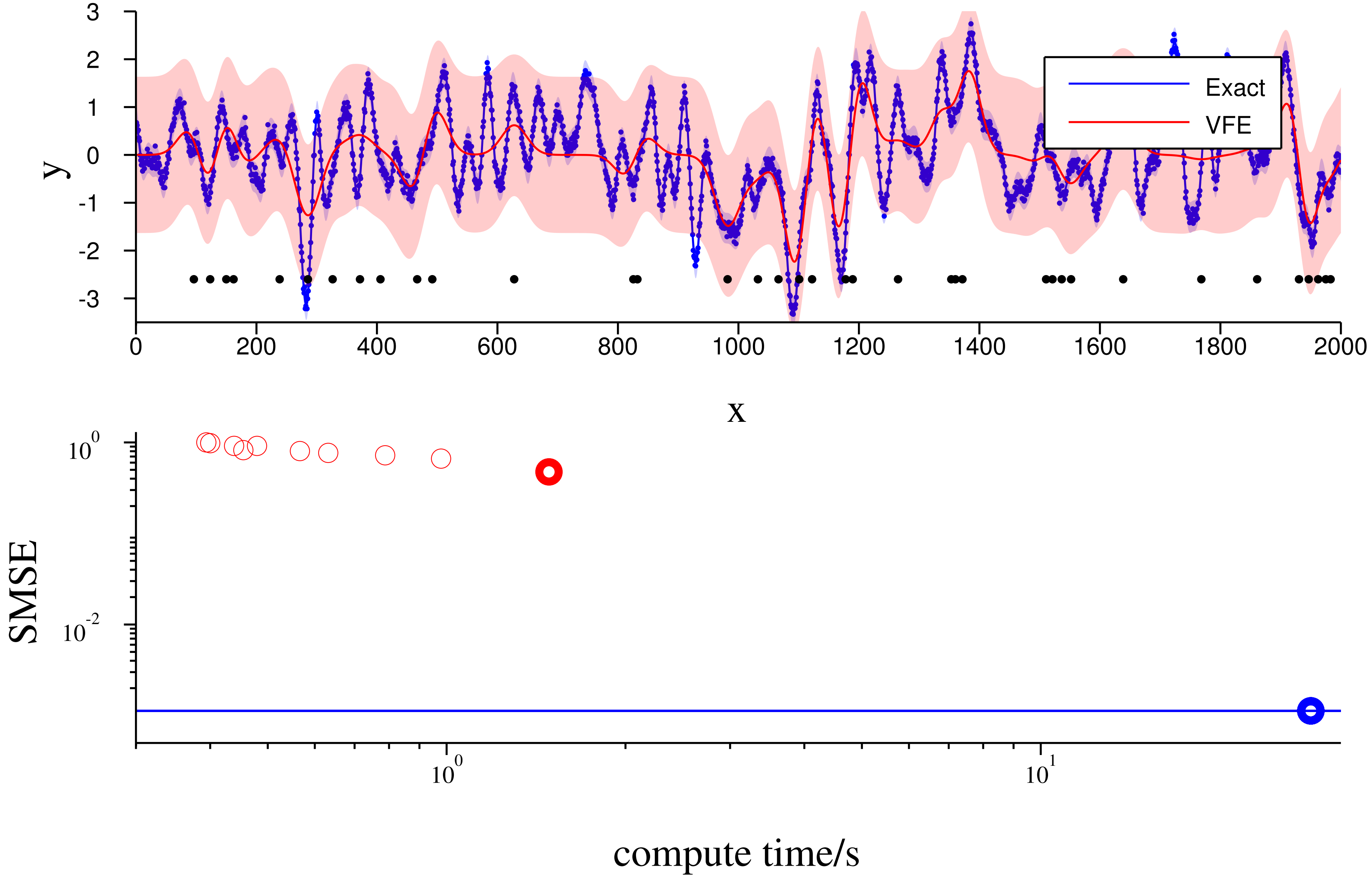


How do we select M = number of pseudo-data?

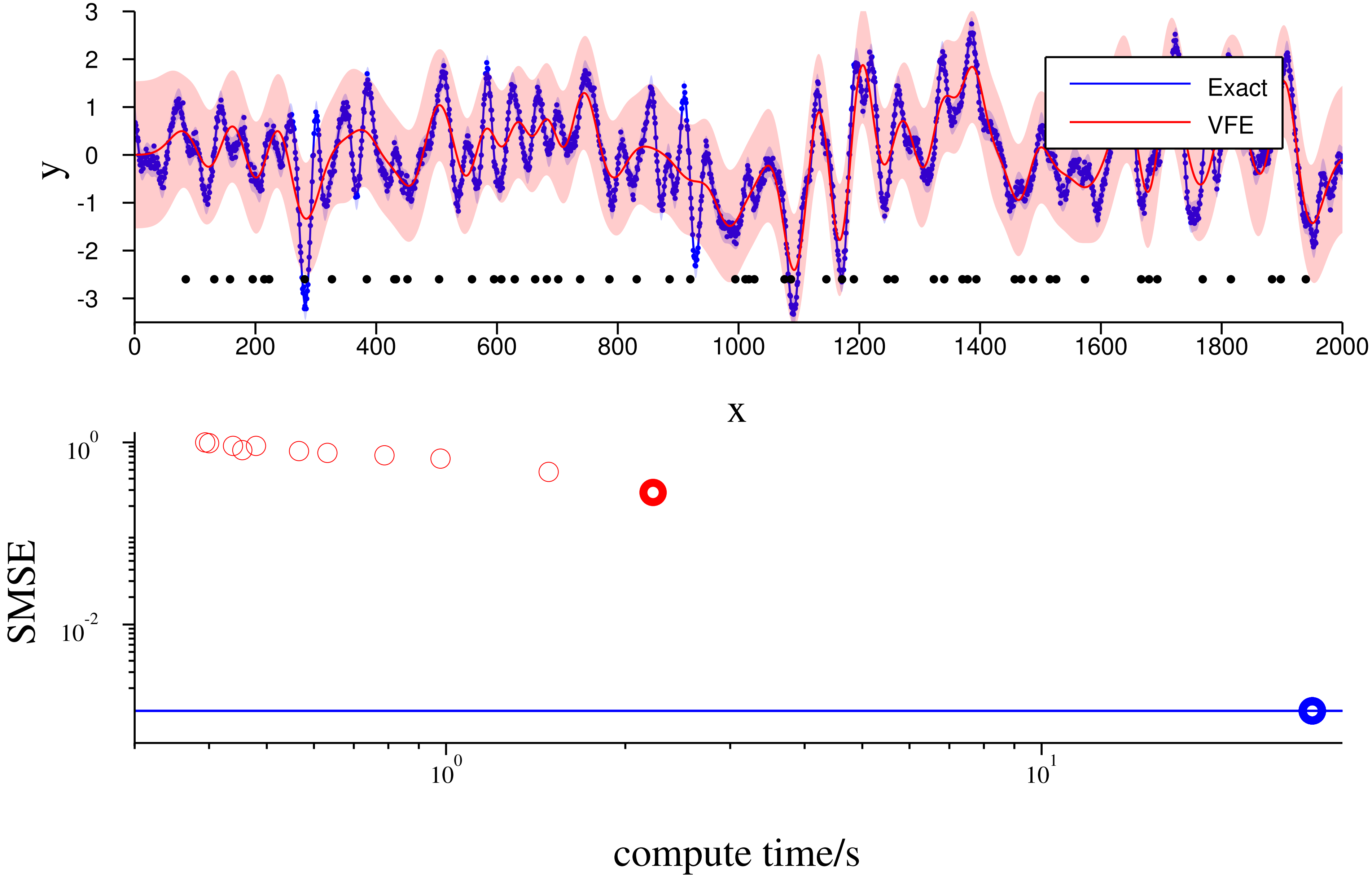




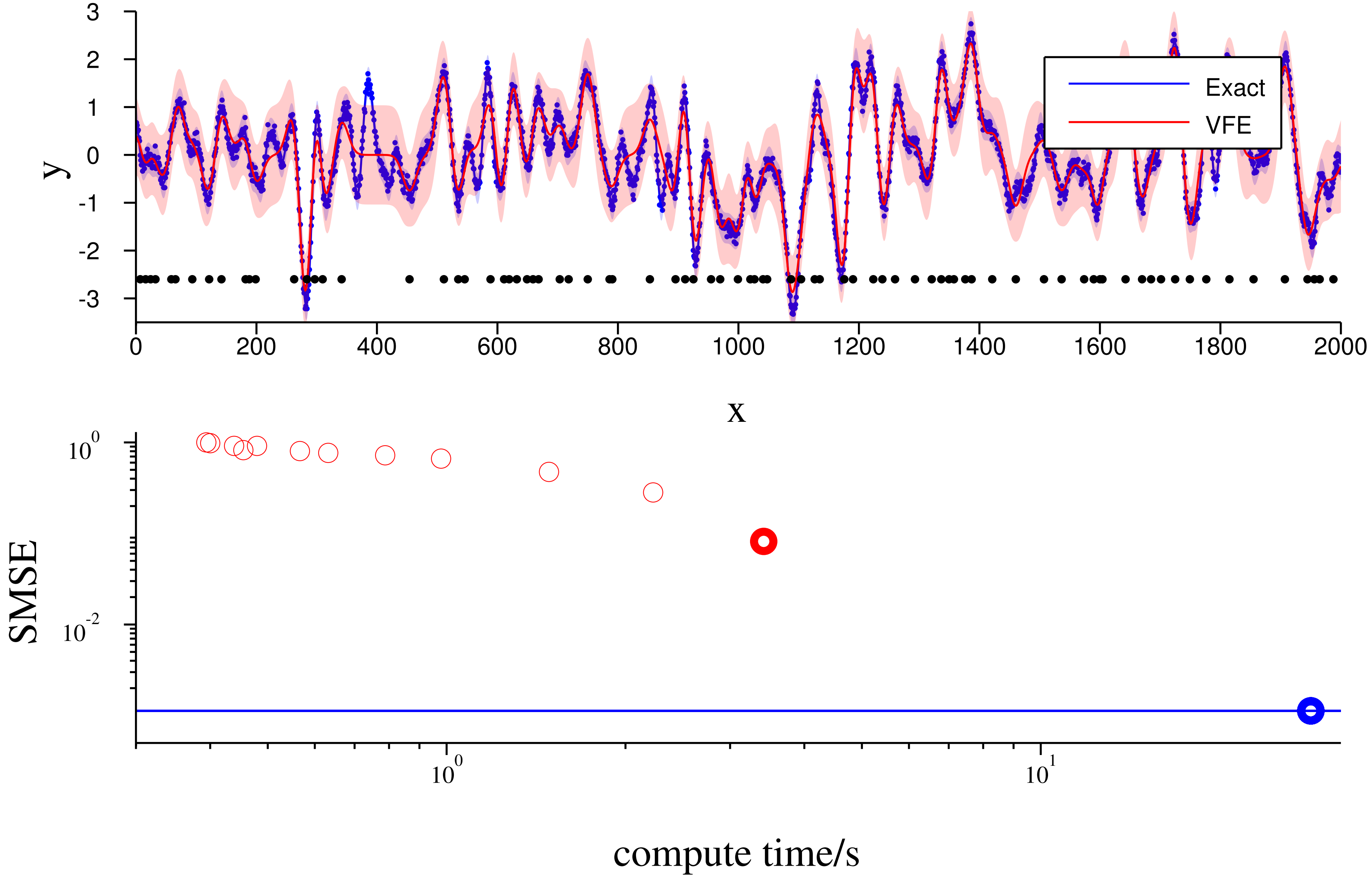
How do we select M = number of pseudo-data?



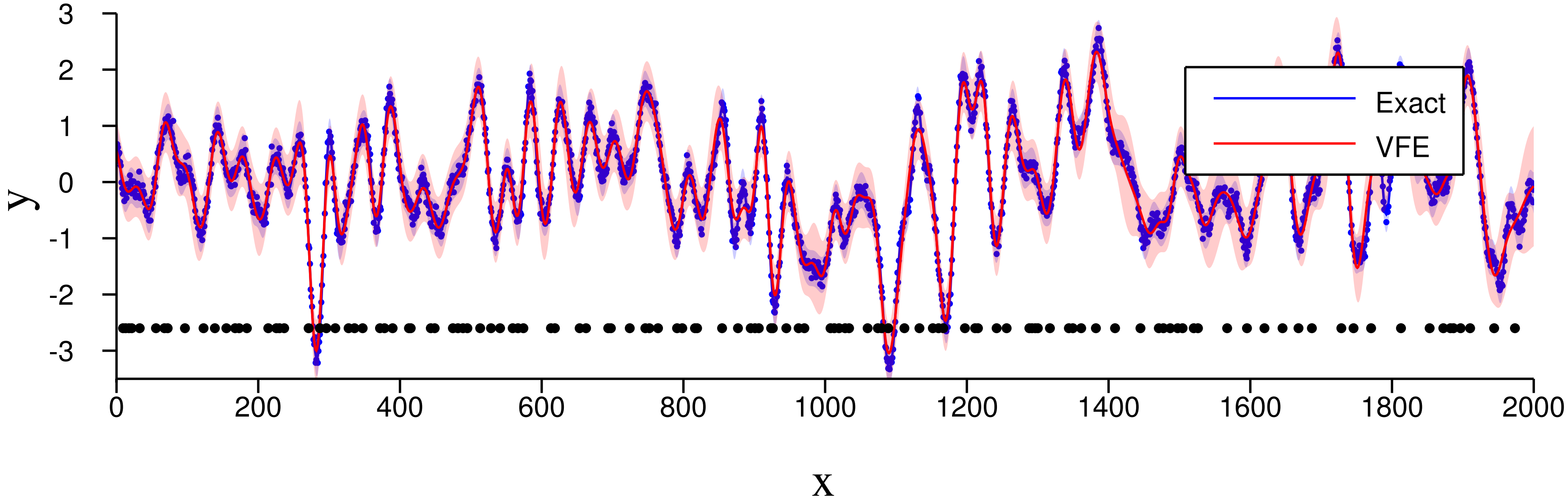
How do we select M = number of pseudo-data?

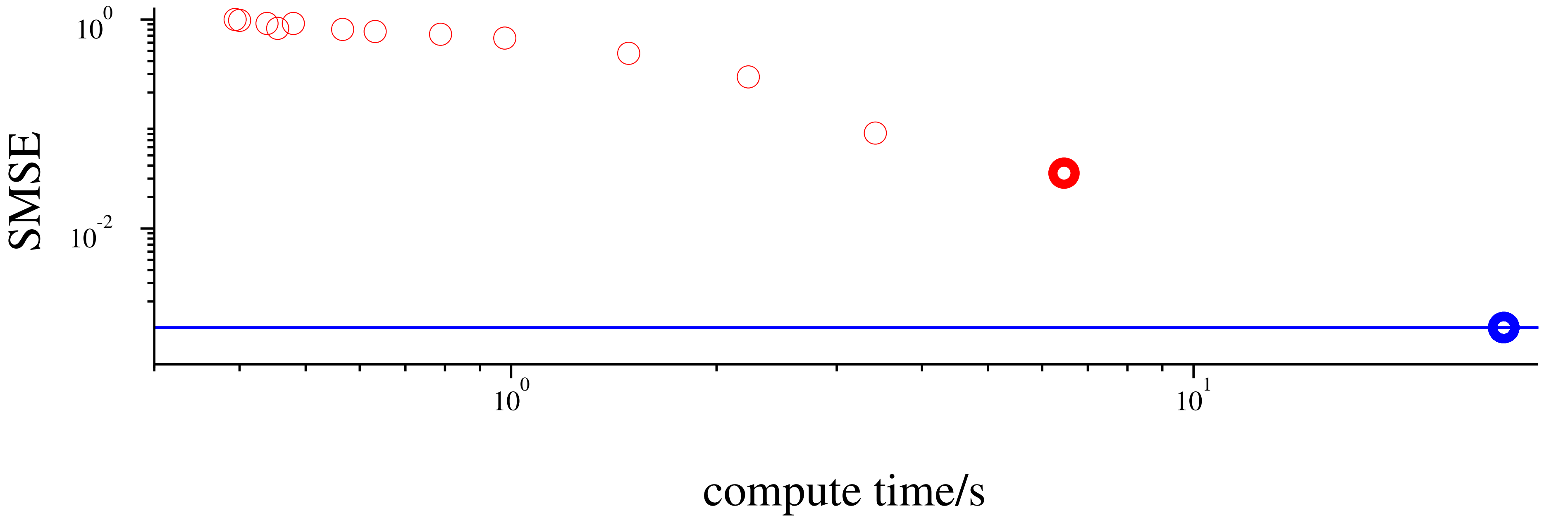


How do we select M = number of pseudo-data?

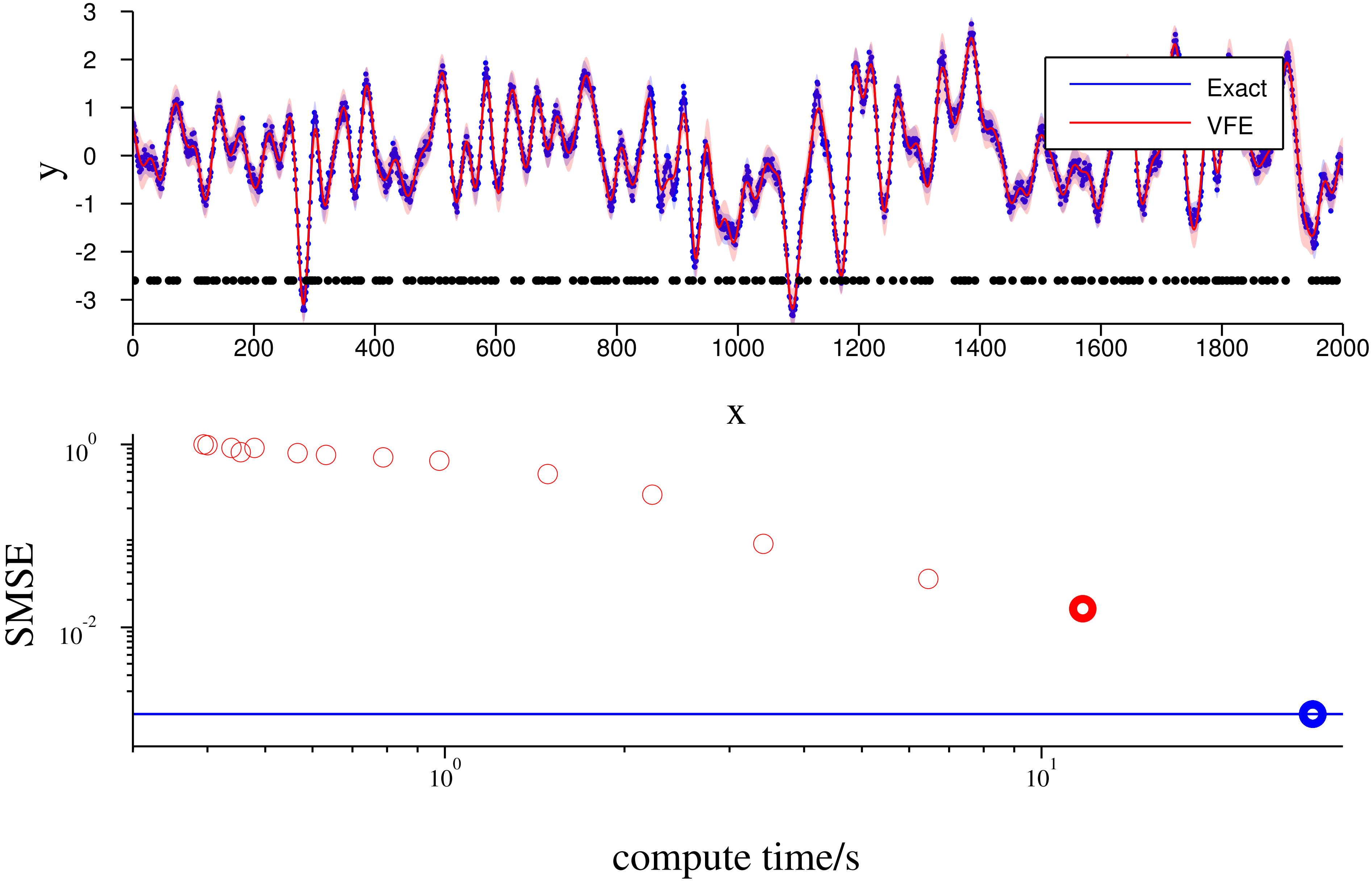


How do we select M = number of pseudo-data?

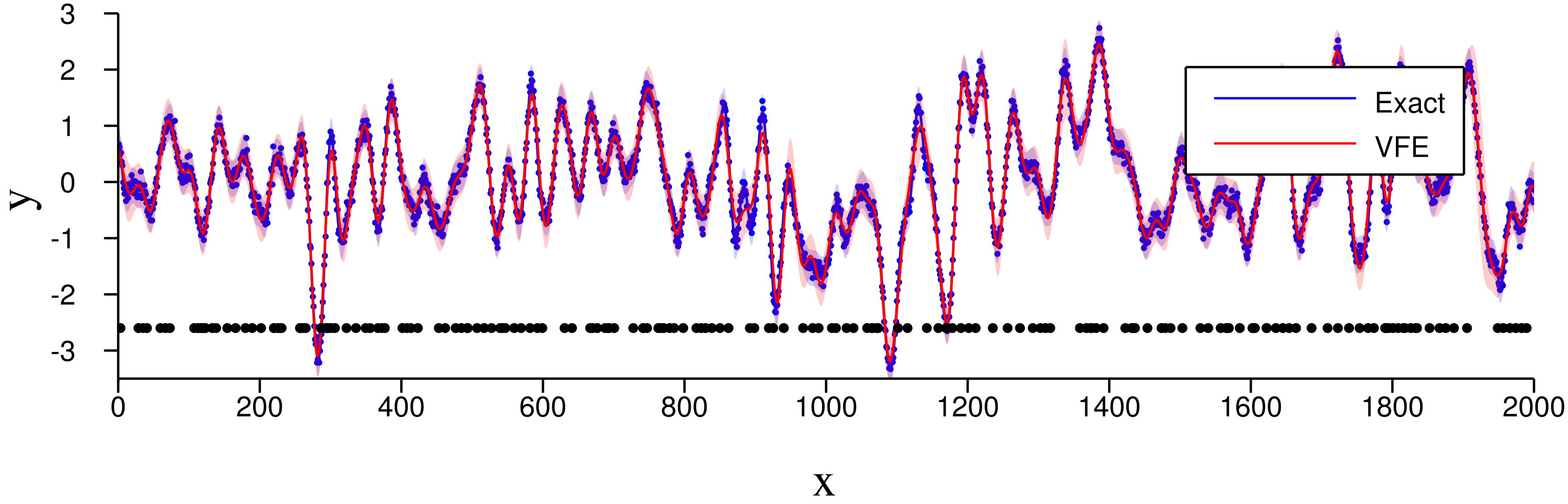


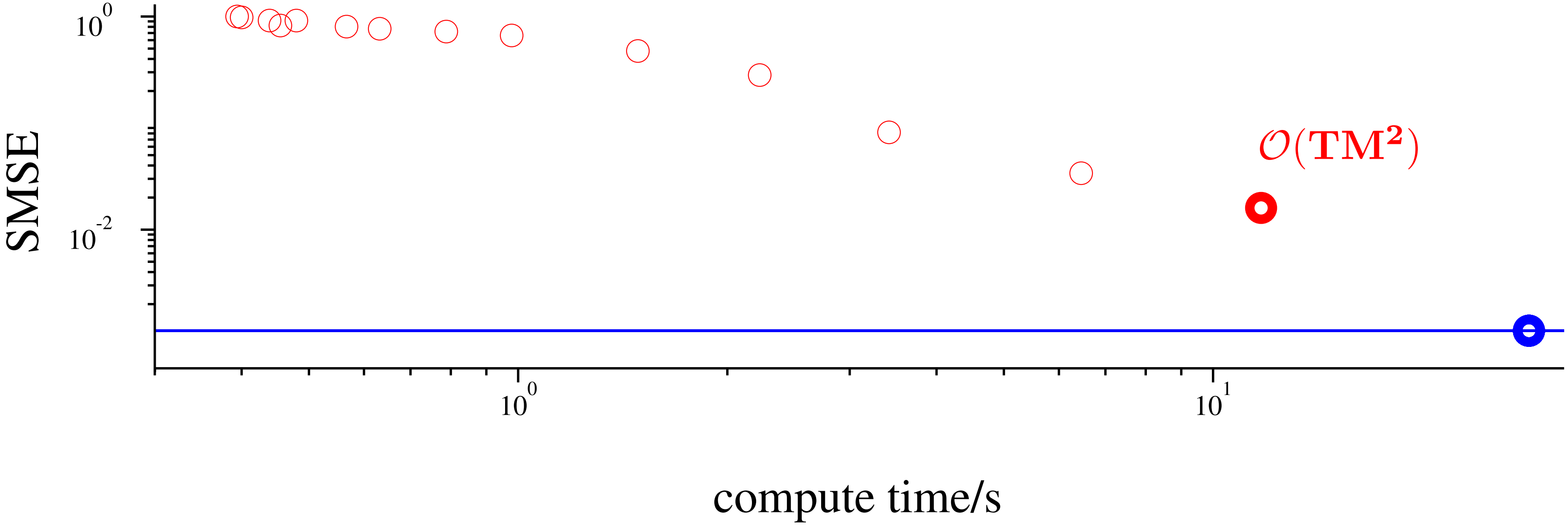


How do we select M = number of pseudo-data?

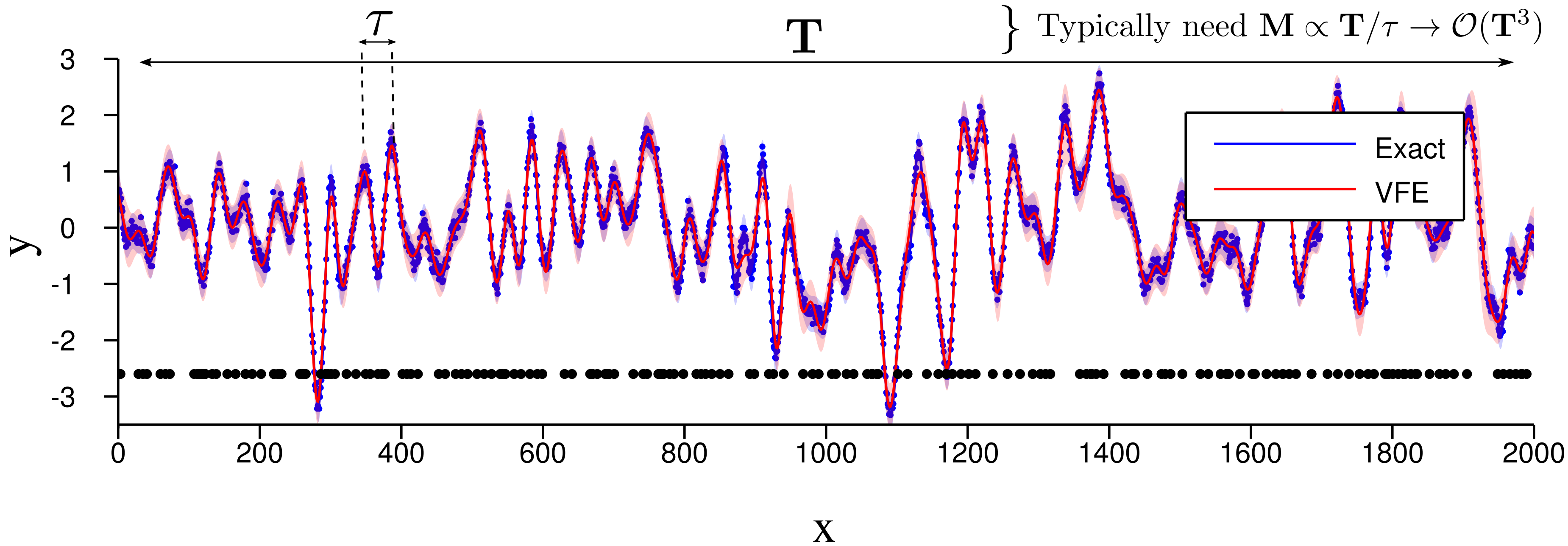


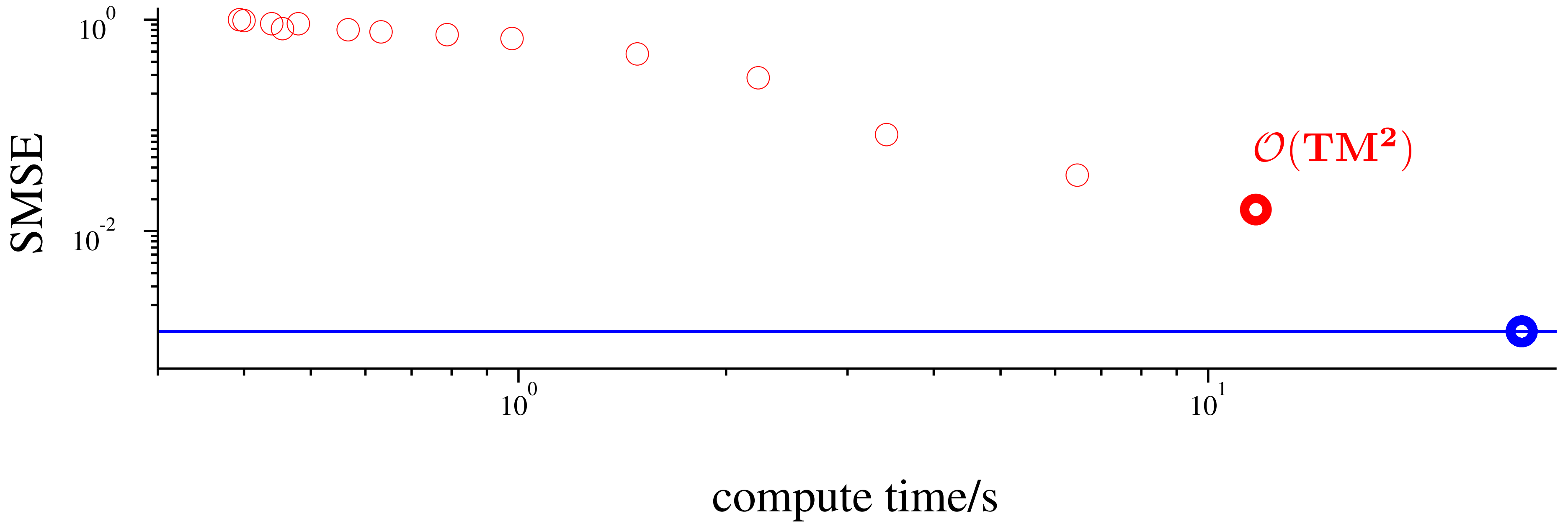
How do we select M = number of pseudo-data?





How do we select M = number of pseudo-data?





Power Expectation Propagation and Gaussian Processes

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

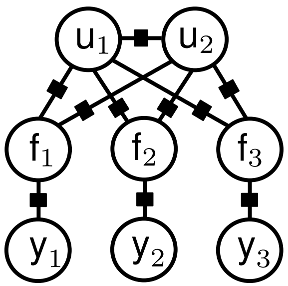
methods employing
pseudo-data

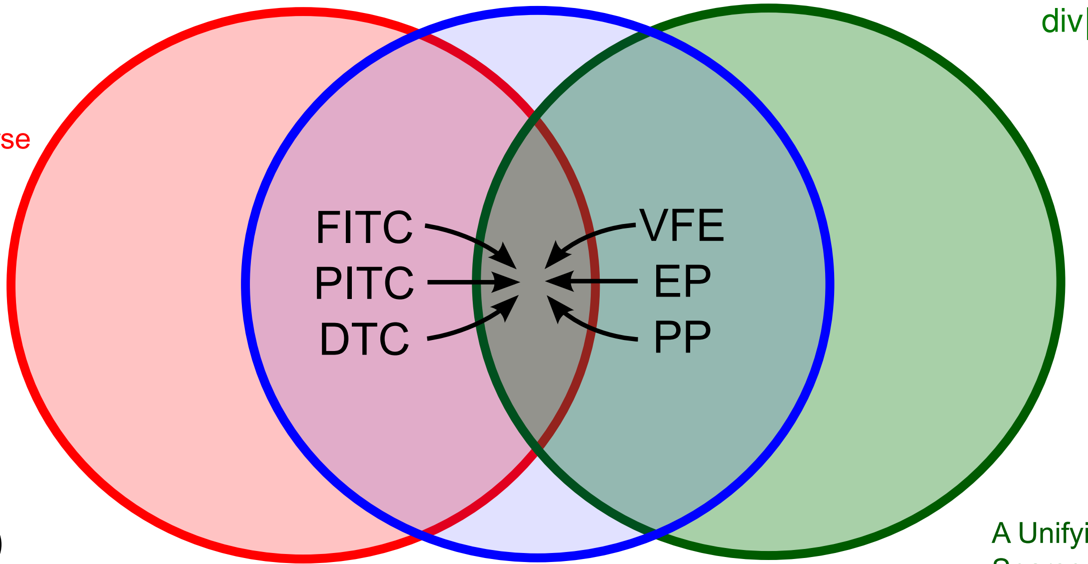
exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)



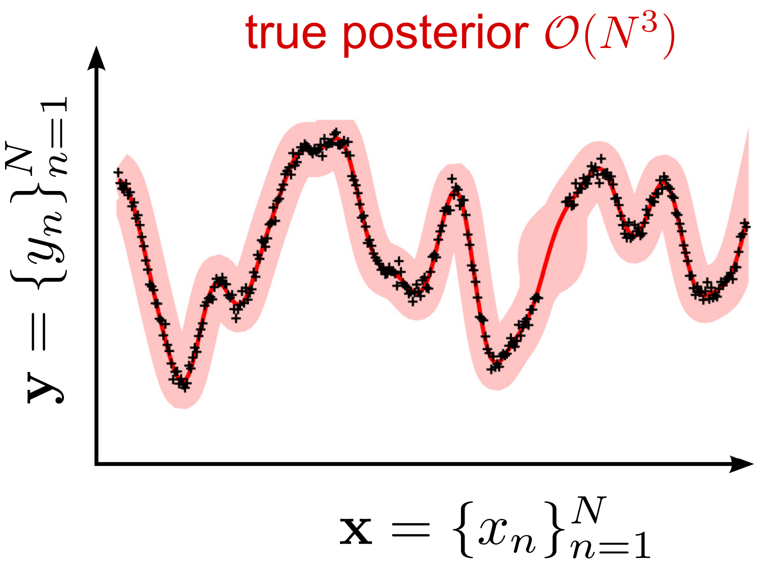


A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

EP pseudo-point approximation

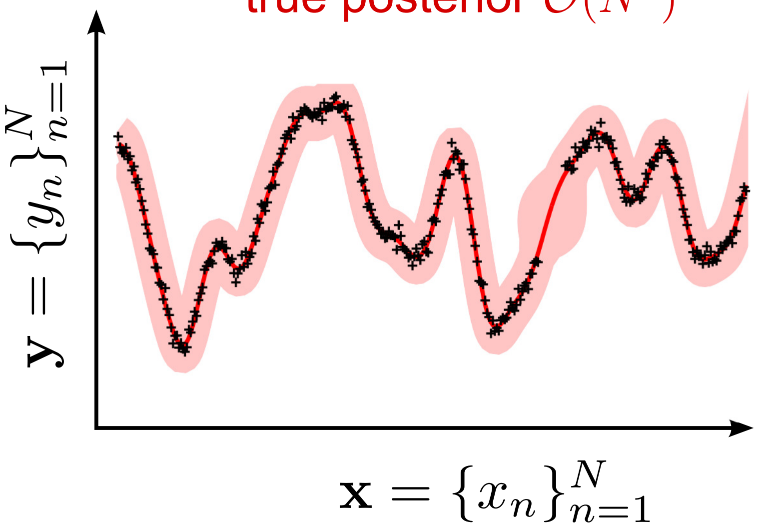
$$p^*(f) = \ p(f, \mathbf{y}|\mathbf{x}, \theta)$$



EP pseudo-point approximation

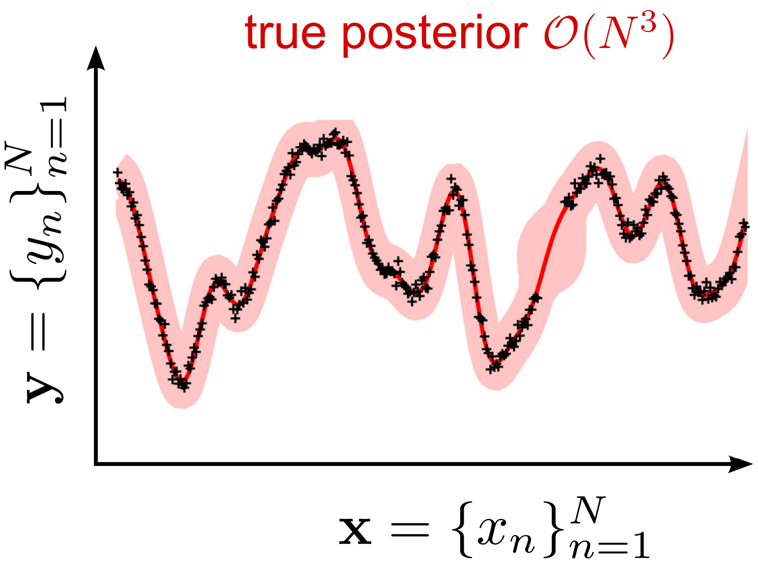
$$p^*(f) = p(f, \mathbf{y}|\mathbf{x}, \theta)$$
$$= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)}$$

true posterior $\mathcal{O}(N^3)$



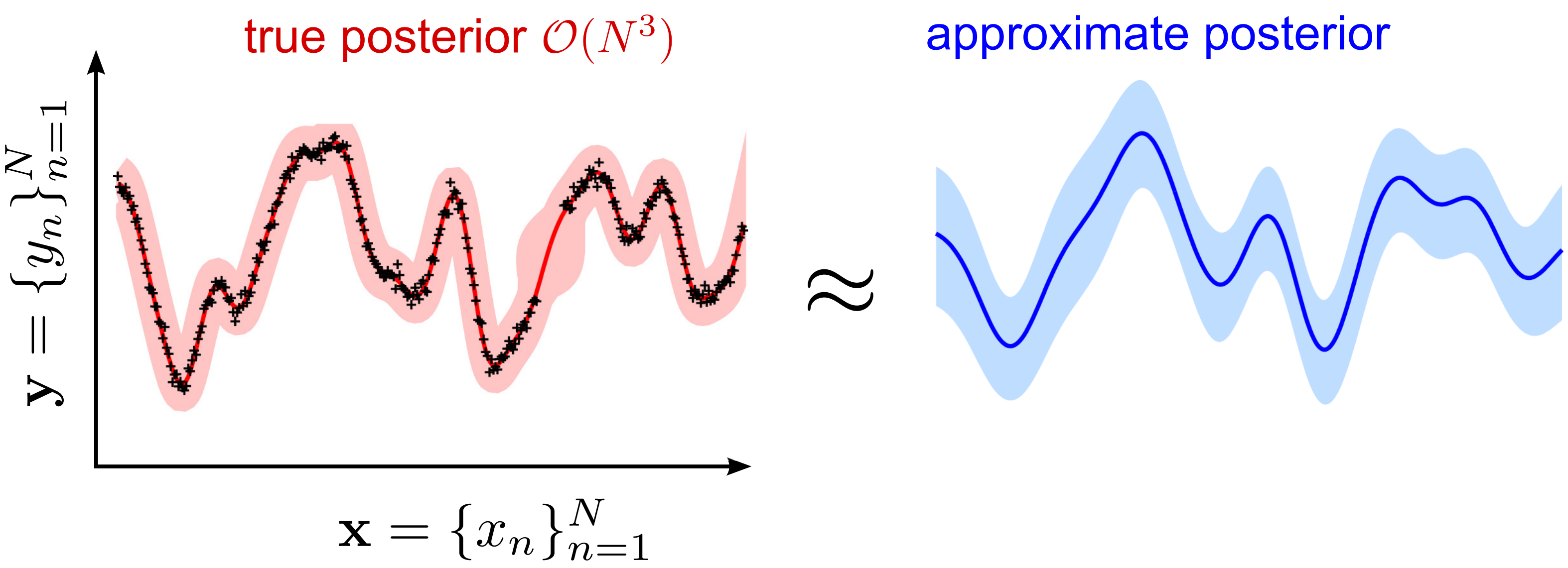
EP pseudo-point approximation

$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y}|\mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \; \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
$$



EP pseudo-point approximation

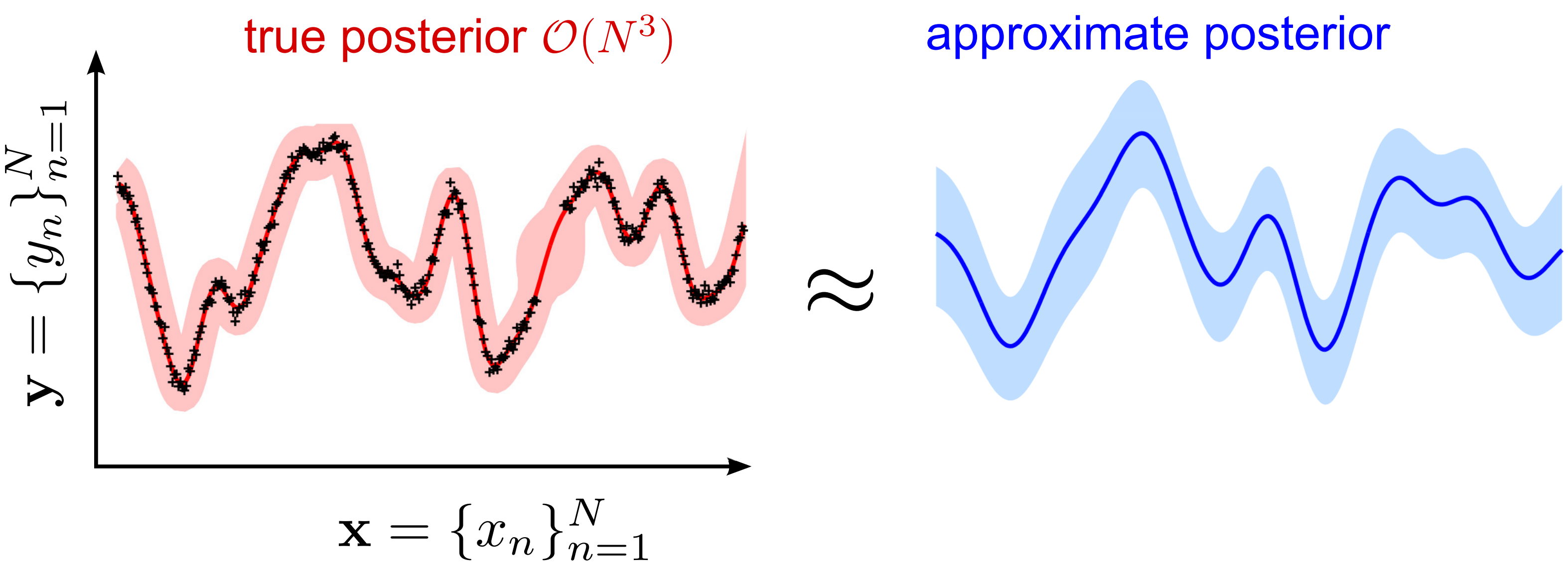
$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y}|\mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \; \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
\qquad
q^*(f) = p(f|\theta) \prod_{n=1}^{N} \underline{t_n(f)}
$$



EP pseudo-point approximation

$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y}|\mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \; \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
$$

$$
\begin{aligned}
q^*(f) &= p(f|\theta) \prod_{n=1}^{N} \underline{t_n(f)} \\
&= \underline{Z_{\mathrm{EP}}} \; \underline{q(f)}
\end{aligned}
$$



EP pseudo-point approximation

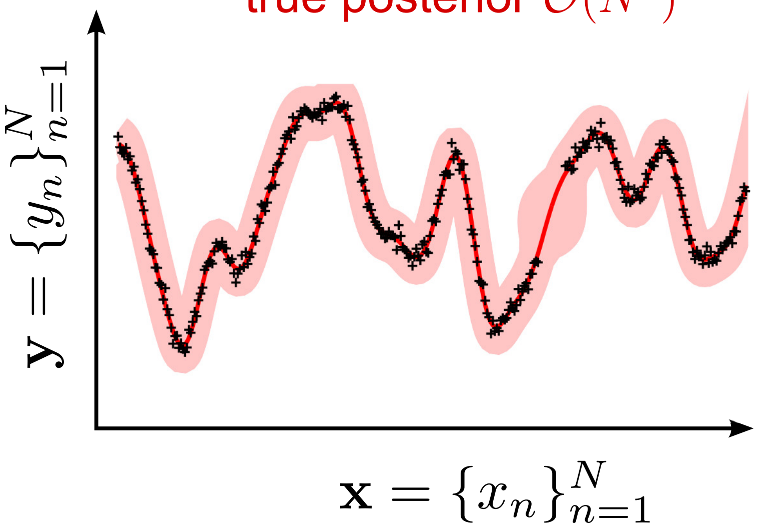
$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y} | \mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n | f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \; \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
$$

$$
\begin{aligned}
q^*(f) &= p(f|\theta) \prod_{n=1}^{N} \underline{t_n(f)} \\
&= \underline{Z_{\mathrm{EP}}} \; \underline{q(f)}
\end{aligned}
$$

$$t_n(f) = \mathcal{N}(\mathbf{u}; \mu_n, \Sigma_n)$$

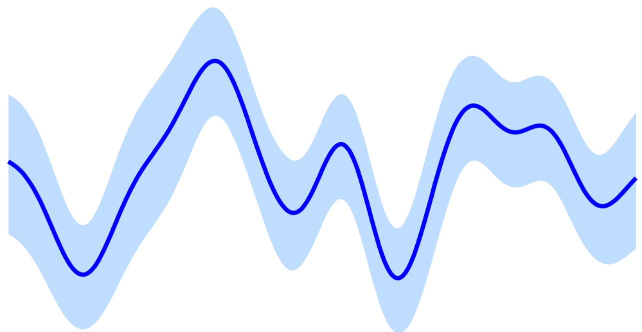
$$\dim(\mathbf{u}) = M \quad f = \{\mathbf{u}, f_{\neq \mathbf{u}}\}$$

true posterior $\mathcal{O}(N^3)$



$\approx$

approximate posterior $\mathcal{O}(NM^2)$



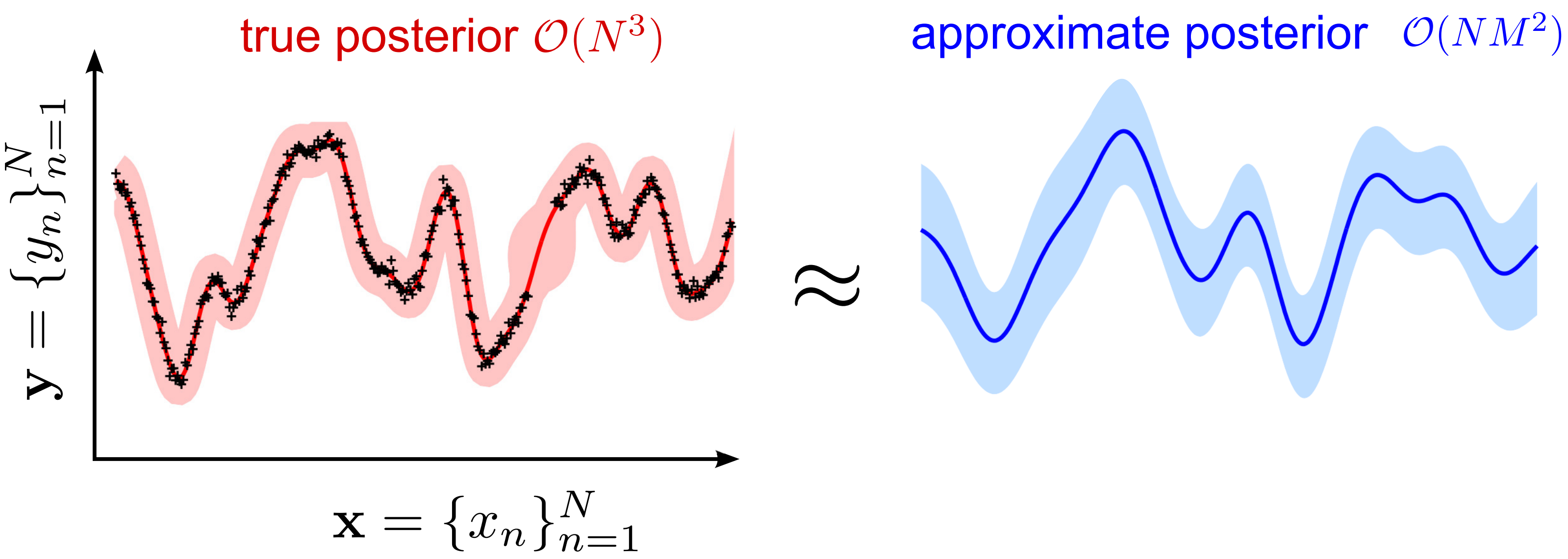
EP pseudo-point approximation

$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y}|\mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \; \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
$$

$$
\begin{aligned}
q^*(f) &= p(f|\theta) p(\tilde{\mathbf{y}}|\mathbf{u}, \tilde{\Sigma}) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{t_n(f)} \\
&= \underline{Z_{\text{EP}}} \; \underline{q(f)}
\end{aligned}
$$

$$t_n(f) = \mathcal{N}(\mathbf{u}; \mu_n, \Sigma_n)$$

$$\dim(\mathbf{u}) = M \quad f = \{\mathbf{u}, f_{\neq \mathbf{u}}\}$$



EP pseudo-point approximation

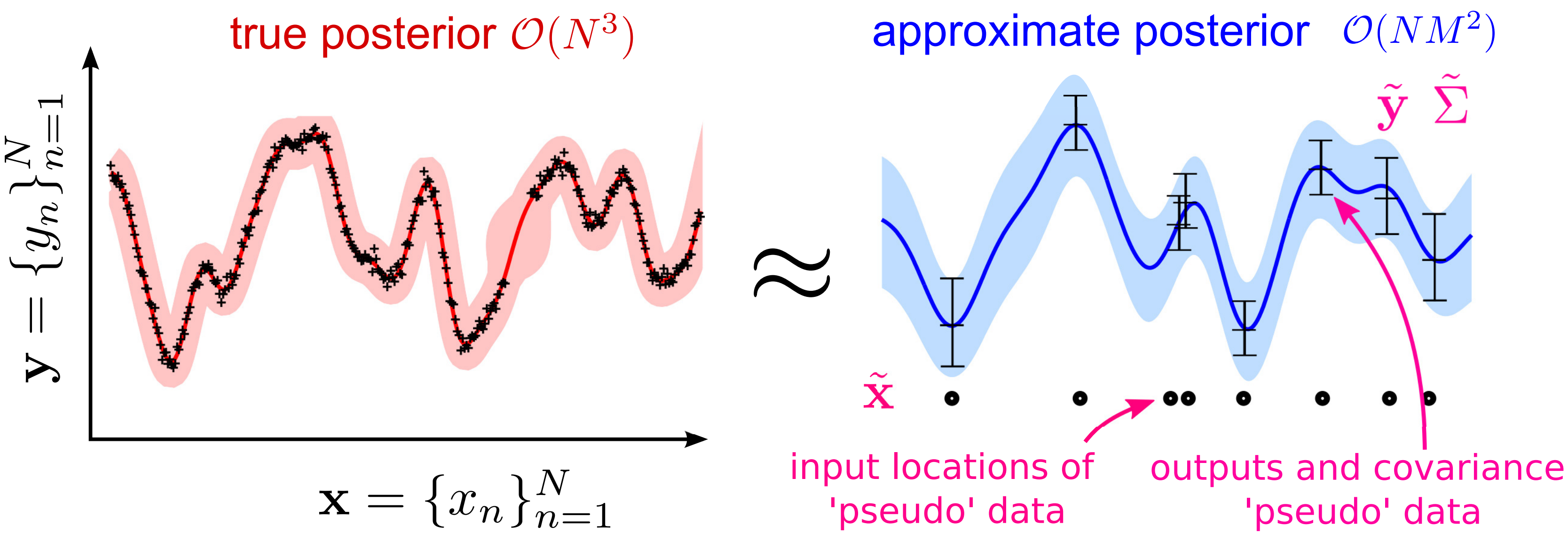
$$
\begin{aligned}
p^*(f) &= p(f, \mathbf{y}|\mathbf{x}, \theta) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{p(y_n|f, x_n, \theta)} \\
&= \underbrace{p(\mathbf{y}|\mathbf{x}, \theta)}_{\text{marginal likelihood}} \ \underbrace{p(f|\mathbf{y}, \mathbf{x}, \theta)}_{\text{posterior}}
\end{aligned}
$$

$$
\begin{aligned}
q^*(f) &= p(f|\theta) p(\tilde{\mathbf{y}}|\mathbf{u}, \tilde{\Sigma}) \\
&= p(f|\theta) \prod_{n=1}^{N} \underline{t_n(f)} \\
&= \underline{Z_{\mathrm{EP}}} \ \underline{q(f)}
\end{aligned}
$$

exact joint of new GP regression model

$$t_n(f) = \mathcal{N}(\mathbf{u}; \mu_n, \Sigma_n)$$

$$\dim(\mathbf{u}) = M \quad f = \{\mathbf{u}, f_{\neq \mathbf{u}}\}$$



EP algorithm

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

2. include

$$p_n^{\text{tilt}}(f) = q^{\backslash n}(f) p(y_n | f, x_n, \theta)$$

tilted

add in one true observation likelihood

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

2. include

$$p_n^{\mathrm{tilt}}(f) = q^{\backslash n}(f)p(y_n|f, x_n, \theta)$$

tilted

add in one true observation likelihood

KL between unnormalised stochastic processes

3. project

$$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \mathrm{KL}\left[p_n^{\mathrm{tilt}}(f)||q^*(f)\right]$$

project onto approximating family

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

2. include

$$p_n^{\text{tilt}}(f) = q^{\backslash n}(f) p(y_n | f, x_n, \theta)$$

tilted

add in one true observation likelihood

KL between unnormalised stochastic processes

3. project

$$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \, \mathrm{KL}\left[p_n^{\text{tilt}}(f) || q^*(f)\right]$$

project onto approximating family

4. update

$$t_n(\mathbf{u}) = \frac{q^*(f)}{q^{\backslash n}(f)}$$

update pseudo-observation likelihood

EP algorithm

1. remove $$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$ take out one pseudo-observation likelihood

cavity

2. include $$p_n^{\text{tilt}}(f) = q^{\backslash n}(f) p(y_n | f, x_n, \theta)$$ add in one true observation likelihood

tilted

KL between unnormalised stochastic processes

3. project $$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \operatorname{KL}\left[p_n^{\text{tilt}}(f) || q^*(f)\right]$$ project onto approximating family

1. minimum: moments matched at pseudo-inputs $\mathcal{O}(NM^2)$
2. Gaussian regression: matches moments everywhere

4. update $$t_n(\mathbf{u}) = \frac{q^*(f)}{q^{\backslash n}(f)}$$ update pseudo-observation likelihood

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

2. include

$$p_n^{\text{tilt}}(f) = q^{\backslash n}(f) p(y_n | f, x_n, \theta)$$

tilted

add in one true observation likelihood

KL between unnormalised stochastic processes

3. project

$$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \operatorname{KL}\left[p_n^{\text{tilt}}(f) || q^*(f)\right]$$

project onto approximating family

1. minimum: moments matched at pseudo-inputs $\mathcal{O}(NM^2)$
2. Gaussian regression: matches moments everywhere

4. update

$$t_n(\mathbf{u}) = \frac{q^*(f)}{q^{\backslash n}(f)}$$
$$= z_n \mathcal{N}(\mathrm{K}_{f_n \mathbf{u}} \mathrm{K}_{\mathbf{uu}}^{-1} \mathbf{u}; g_n, v_n)$$

update pseudo-observation likelihood

rank 1

A Brief History of Gaussian Process Approximations

approximate generative model
exact inference

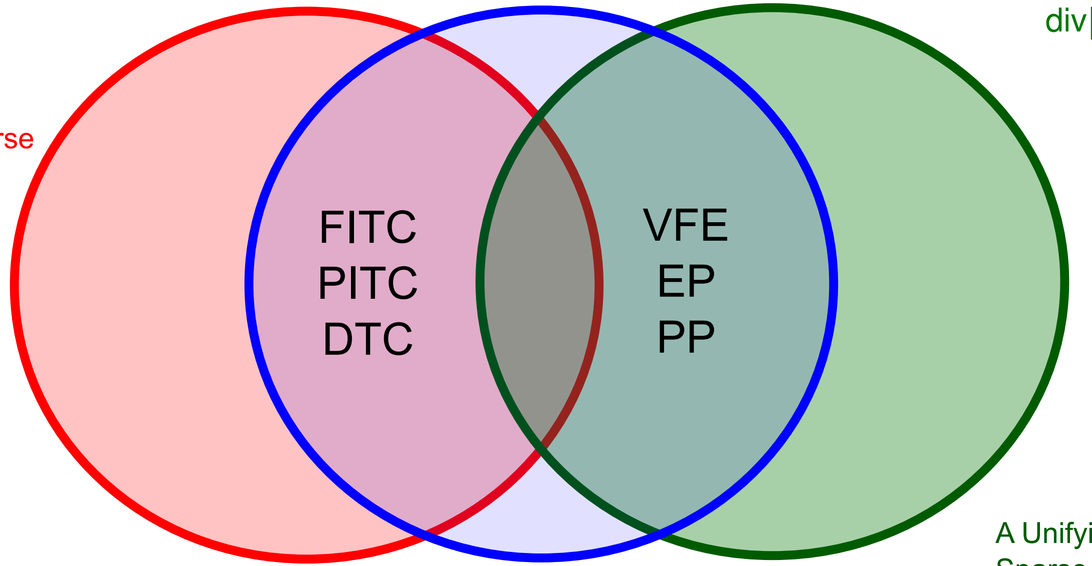
methods employing
pseudo-data

exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)



A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

Fixed points of EP = FITC approximation

approximate generative model
exact inference

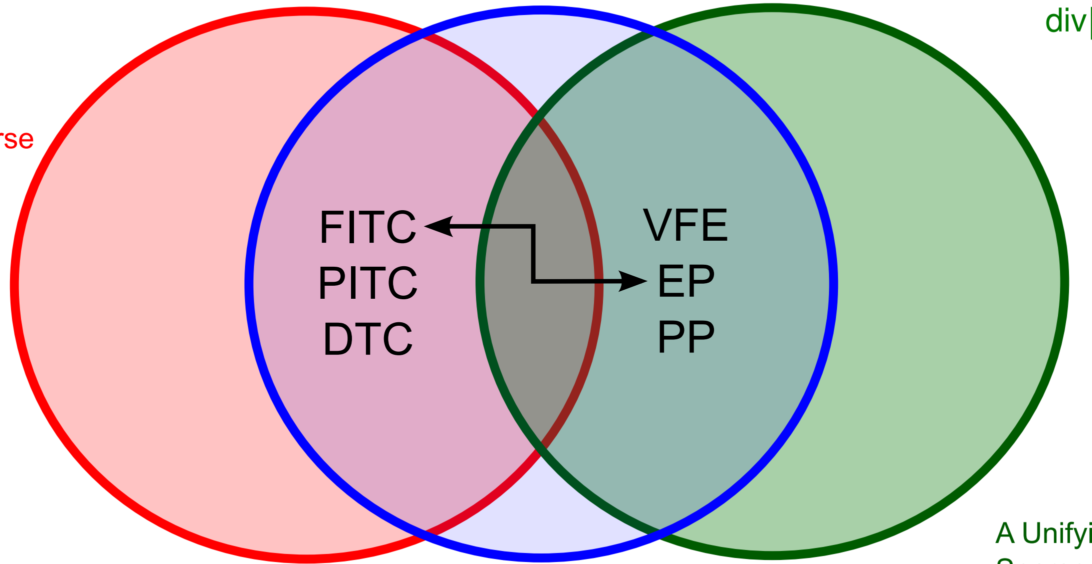
methods employing
pseudo-data

exact generative model
approximate inference

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

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

A Unifying View of Sparse
Approximate Gaussian
Process Regression
Quinonero-Candela &
Rasmussen, 2005
(FITC, PITC, DTC)



A Unifying Framework for
Sparse Gaussian Process
Approximation using
Power Expectation
Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

Fixed points of EP = FITC approximation

approximate generative model
exact inference

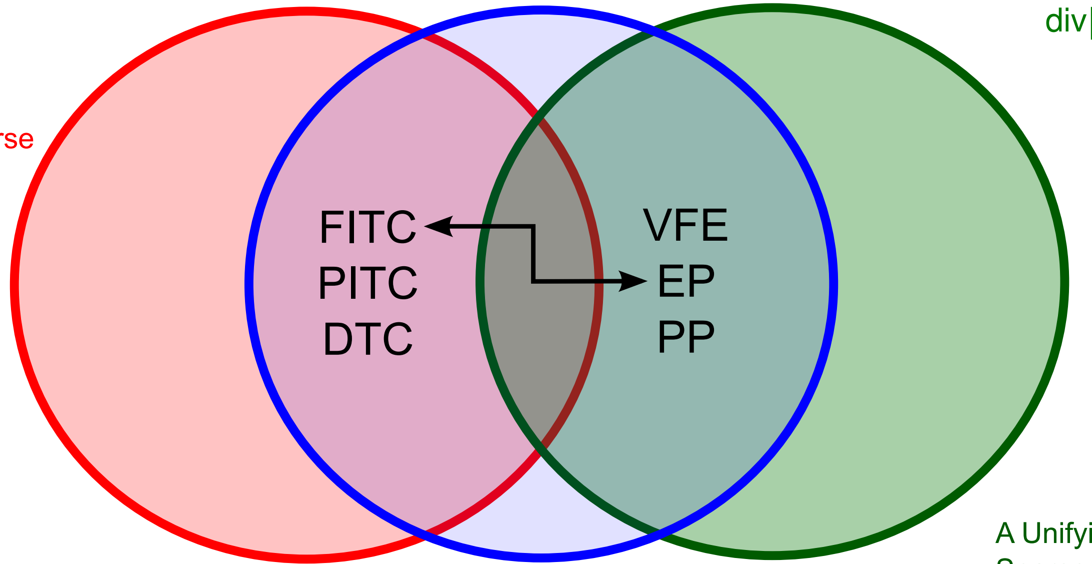
methods employing
pseudo-data

exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)



A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

Fixed points of EP = FITC approximation

approximate generative model
exact inference

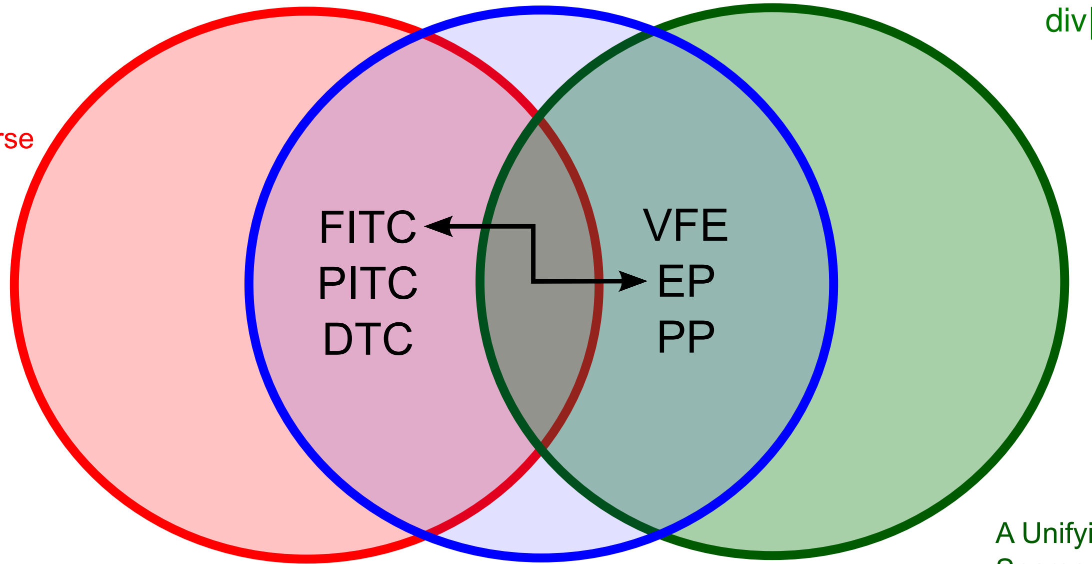
methods employing
pseudo-data

exact generative model
approximate inference

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

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

A Unifying View of Sparse Approximate Gaussian Process Regression
Quinonero-Candela & Rasmussen, 2005
(FITC, PITC, DTC)



A Unifying Framework for Sparse Gaussian Process Approximation using Power Expectation Propagation
Bui, Yan and Turner, 2016
(VFE, EP, FITC, PITC ...)

interpretation resolves issues with FITC:
why does it work so well?
are we allowed to increase M with N

FITC: Snelson et al. “Sparse Gaussian Processes using Pseudo-inputs”
PITC: Snelson et al. “Local and global sparse Gaussian process approximations”
EP: Csato and Opper 2002 / Qi et al. "Sparse-posterior Gaussian Processes for general likelihoods.”
VFE: Titsias “Variational Learning of Inducing Variables in Sparse Gaussian Processes”
DTC / PP: Seeger et al. “Fast Forward Selection to Speed Up Sparse Gaussian Process Regression”

EP algorithm

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})}$$

cavity

take out one pseudo-observation likelihood

2. include

$$p_n^{\mathrm{tilt}}(f) = q^{\backslash n}(f) p(y_n | f, x_n, \theta)$$

tilted

add in one true observation likelihood

KL between unnormalised stochastic processes

3. project

$$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \, \mathrm{KL}\left[p_n^{\mathrm{tilt}}(f) || q^*(f)\right]$$

project onto approximating family

1. minimum: moments matched at pseudo-inputs $\mathcal{O}(NM^2)$
2. Gaussian regression: matches moments everywhere

4. update

$$t_n(\mathbf{u}) = \frac{q^*(f)}{q^{\backslash n}(f)}$$
$$= z_n \mathcal{N}(\mathrm{K}_{f_n \mathbf{u}} \mathrm{K}_{\mathbf{uu}}^{-1} \mathbf{u}; g_n, v_n)$$

update pseudo-observation likelihood

rank 1

Power EP algorithm (as tractable as EP)

1. remove

$$q^{\backslash n}(f) = \frac{q^*(f)}{t_n(\mathbf{u})^{\alpha}}$$

cavity

take out fraction of pseudo-observation likelihood

2. include

$$p_n^{\text{tilt}}(f) = q^{\backslash n}(f)p(y_n|f, x_n, \theta)^{\alpha}$$

tilted

add in fraction of true observation likelihood

KL between unnormalised stochastic processes

3. project

$$q^*(f) = \underset{q^*(f)}{\operatorname{argmin}} \operatorname{KL}\left[p_n^{\text{tilt}}(f)||q^*(f)\right]$$

project onto approximating family

1. minimum: moments matched at pseudo-inputs $\mathcal{O}(NM^2)$
2. Gaussian regression: matches moments everywhere

4. update

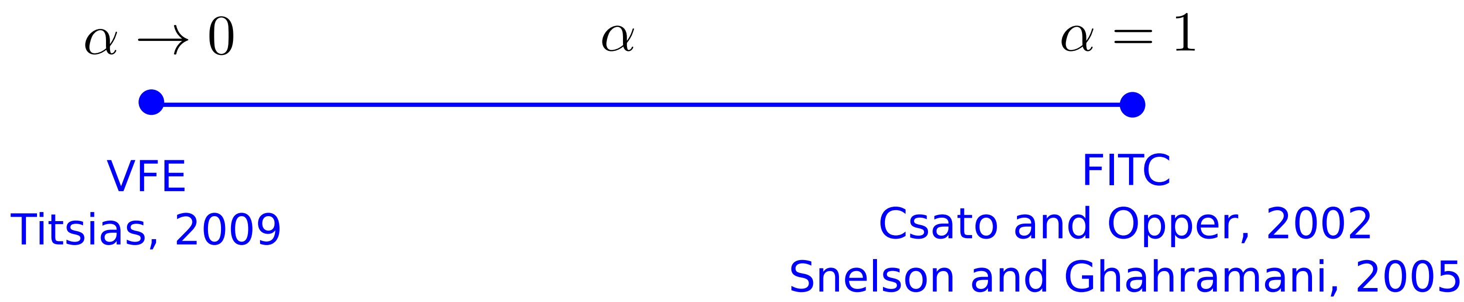
$$t_n(\mathbf{u})^{\alpha} = \frac{q^*(f)}{q^{\backslash n}(f)}$$

$$t_n(\mathbf{u}) = z_n \mathcal{N}(\mathrm{K}_{f_n \mathbf{u}} \mathrm{K}_{\mathbf{uu}}^{-1} \mathbf{u}; g_n, v_n)$$

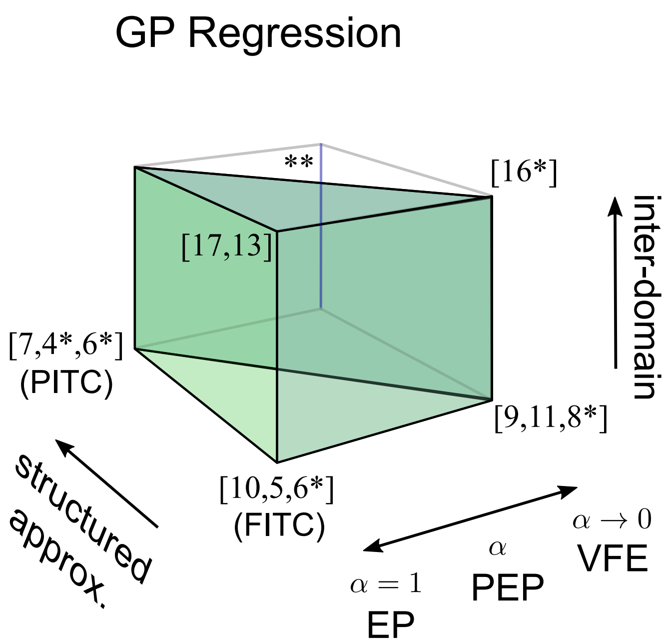
update pseudo-observation likelihood

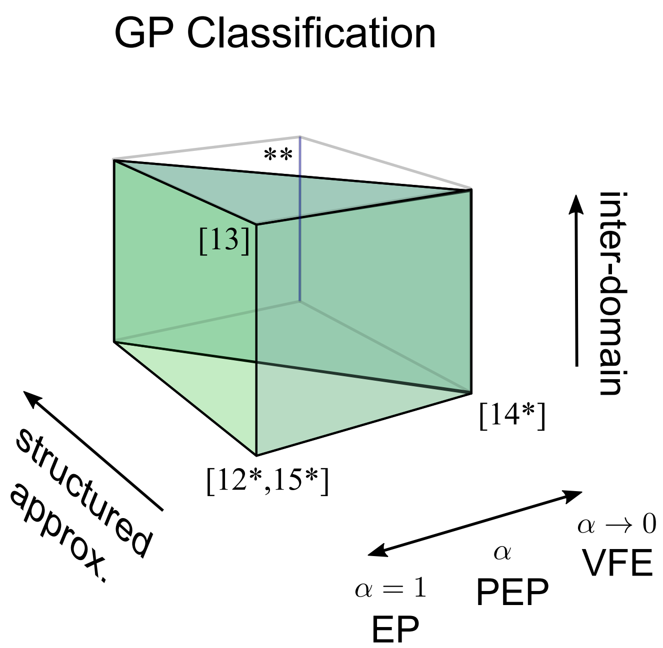
rank 1

Power EP: a unifying framework



Power EP: a unifying framework





[4] Quiñonero-Candela et al. 2005
[5] Snelson et al., 2005
[6] Snelson, 2006
[7] Schwaighofer, 2002

[8] Titsias, 2009
[9] Csató, 2002
[10] Csató et al., 2002
[11] Seeger et al., 2003

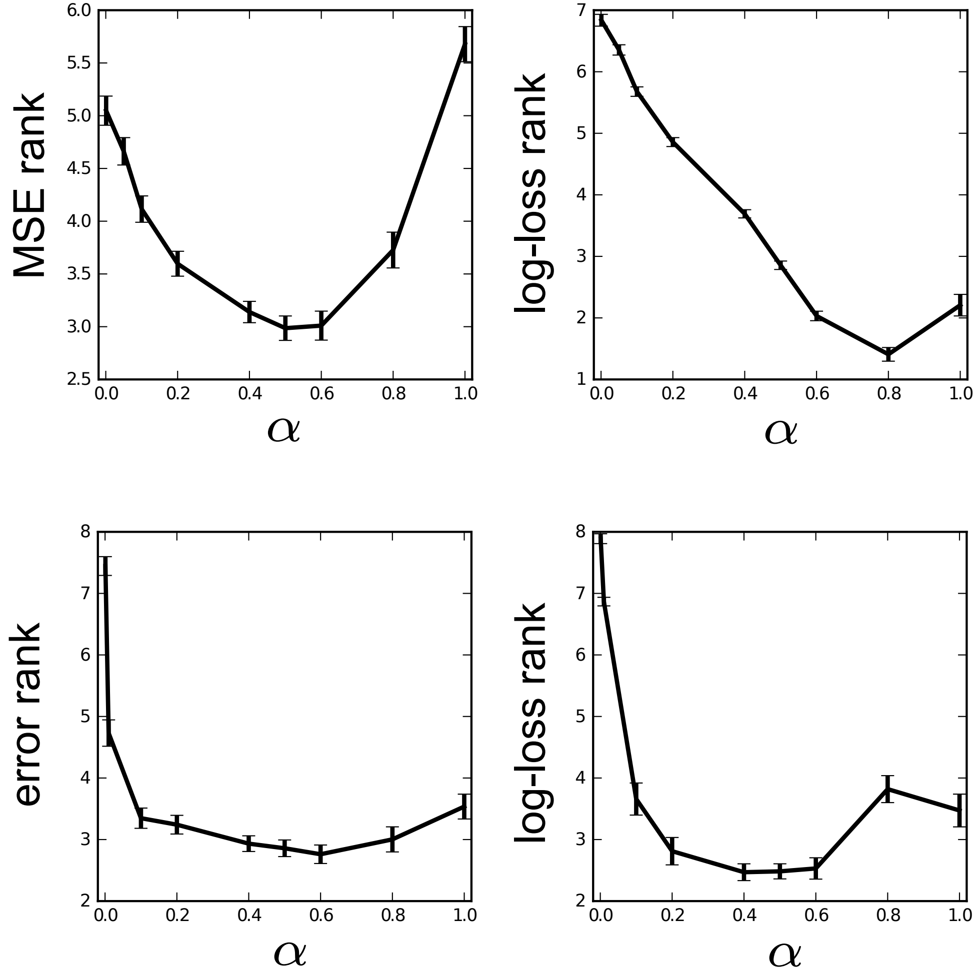
[12] Naish-Guzman et al, 2007
[13] Qi et al., 2010
[14] Hensman et al., 2015
[15] Hernández-Lobato et al., 2016
[16] Matthews et al., 2016
[17] Figueiras-Vidal et al., 2009

* = optimised pseudo-inputs
** = structured versions of VFE recover VFE

How should I set the power parameter α?

8 UCI regression datasets
20 random splits
M = 0 - 200
hypers and inducing
inputs optimised

6 UCI classification datasets
20 random splits
M = 10, 50, 100
hypers and inducing
inputs optimised



α = 0.5 does well on average

References (hyperlinked)

Approximate inference in GPs:

- Sparse Online Gaussian Processes, Csato and Opper, Neural Computation, 2002
- A Unifying View of Sparse Approximate Gaussian Process Regression, Quinonero-Candela and Rasmussen, JMLR, 2005
- Variational Learning of Inducing Variables in Sparse Gaussian Processes Titsias, AIStats, 2009
- On Sparse Variational Methods and the Kullback-Leibler Divergence between Stochastic Processes, Matthews et al., ICML 2016
- A Unifying Framework for Gaussian Process Pseudo-Point Approximations using Power Expectation Propagation, Bui et al., JMLR 2017
- Streaming Sparse Gaussian Process Approximations, Bui et al., NIPS 2017
- Efficient Deterministic Approximate Bayesian Inference for Gaussian Process Models , Bui, thesis, 2018

Deep Gaussian Processes:

- Deep Gaussian Processes for Regression using Approximate Expectation Propagation, Bui et al., ICML 2016
- Doubly Stochastic Variational Inference for Deep Gaussian Processes Salimbeni and Deisenroth, NIPS 2017

Appendix: proof of KL divergence properties

Minimise Kullback Leibler divergence (relative entropy) $\mathcal{KL}(q(x)||p(x))$: add Lagrange multiplier (enforce $q(x)$ normalises), take variational derivatives:

$$\frac{\delta}{\delta q(x)}\Big[\int q(x)\log\frac{q(x)}{p(x)}dx + \lambda(1 - \int q(x)\,dx)\Big] \;=\; \log\frac{q(x)}{p(x)} + 1 - \lambda.$$

Find staionary point by setting the derivative to zero:

$q(x) \;=\; \exp(\lambda{-}1)p(x)$, normalization conditon $\lambda \;=\; 1$, so $q(x) \;=\; p(x)$,

which corresponds to a minimum, since the second derivative is positive:

$$\frac{\delta^2}{\delta q(x)\delta q(x)}\mathcal{KL}(q(x)||p(x)) \;=\; \frac{1}{q(x)} \;>\; 0.$$

The minimum value attained at $q(x) = p(x)$ is $\mathcal{KL}(p(x)||p(x)) = 0$, showing that $\mathcal{KL}(q(x)||p(x))$

- is non-negative and it attains its minimum 0 when $p(x)$ and $q(x)$ are equal