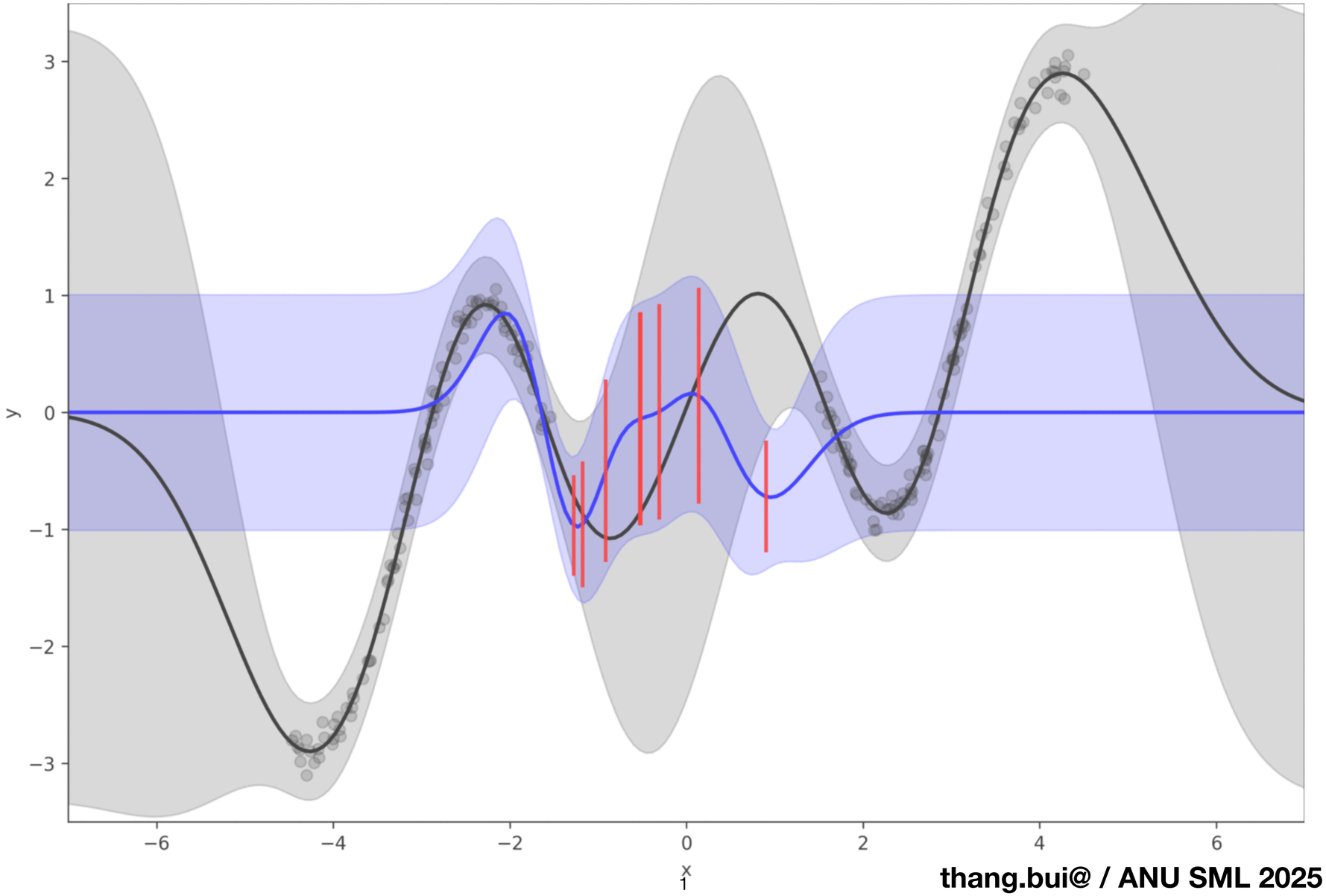
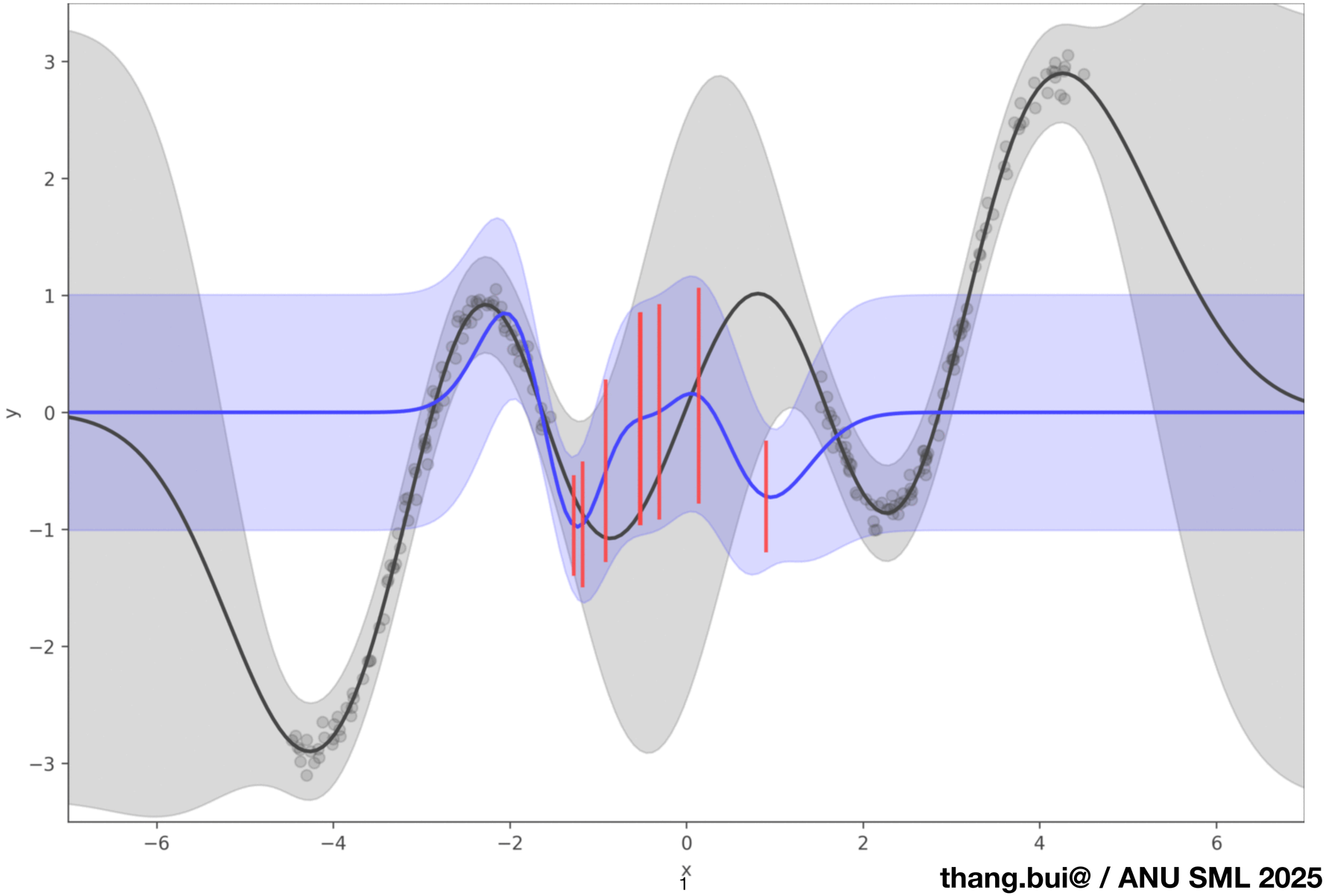
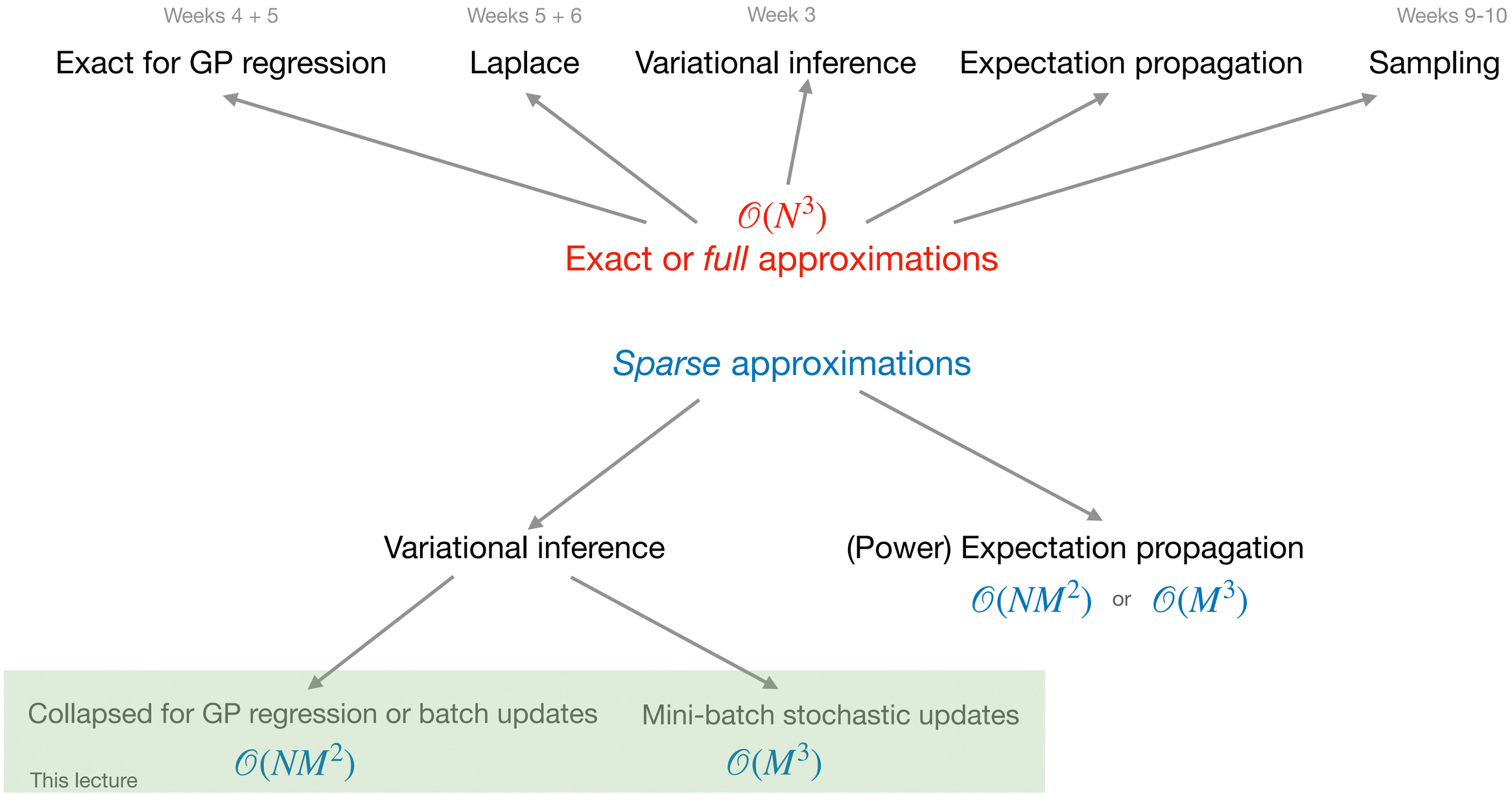
Gaussian process approximations



Gaussian process approximations



Big picture


Weeks 4 + 5
Exact for GP regression
Weeks 5 + 6
Laplace
Week 3
Variational inference
Expectation propagation
Weeks 9-10
Sampling
$\mathcal{O}(N^3)$
Exact or *full* approximations
Sparse approximations
Variational inference
(Power) Expectation propagation
$\mathcal{O}(NM^2)$ or $\mathcal{O}(M^3)$
Collapsed for GP regression or batch updates
$\mathcal{O}(NM^2)$
Mini-batch stochastic updates
$\mathcal{O}(M^3)$
This lecture

Gaussian process refresher

A *Gaussian process* is a collection of random variables, any finite number of which have a joint *Gaussian distribution*

Gaussian process refresher

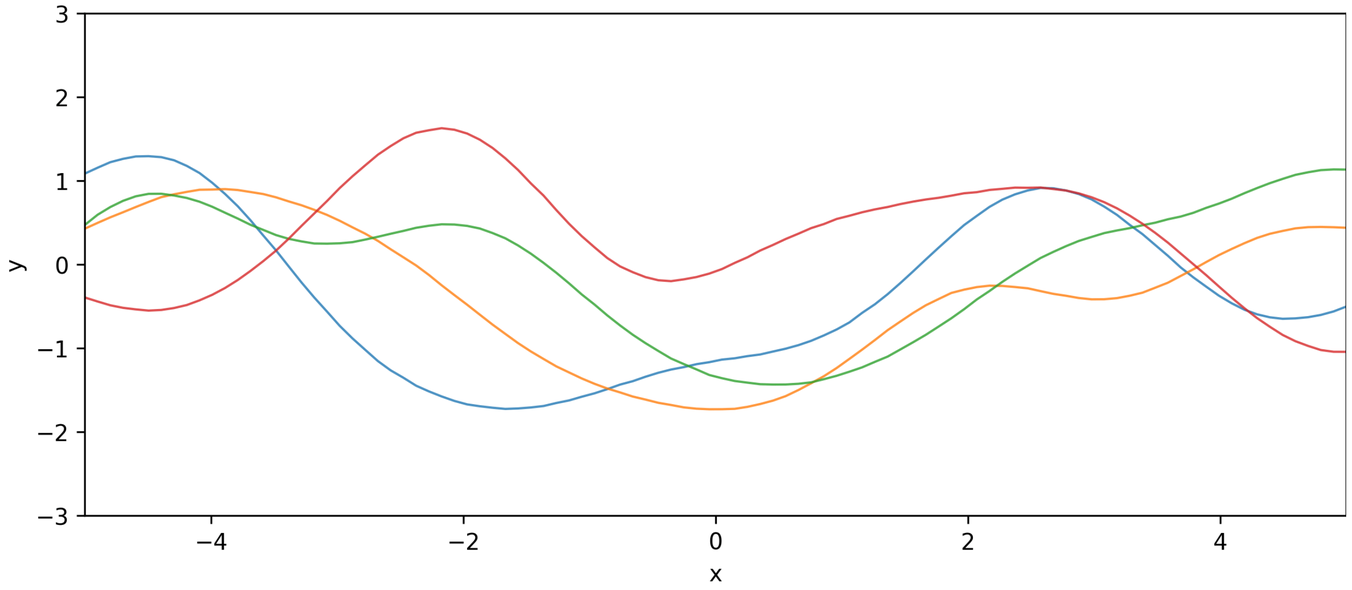
A *Gaussian process* is a collection of random variables, any finite number of which have a joint *Gaussian distribution*

Multivariate Gaussian distribution

- $\mathbf{f} \sim \mathrm{N}(\mu, \Sigma)$
- Mean μ
- Covariance matrix Σ

Gaussian process

- $f(x) \sim \mathrm{GP}(m(x), k(x, x'))$
- Mean function $m(x)$
- Covariance func. $k(x, x')$



Gaussian process refresher

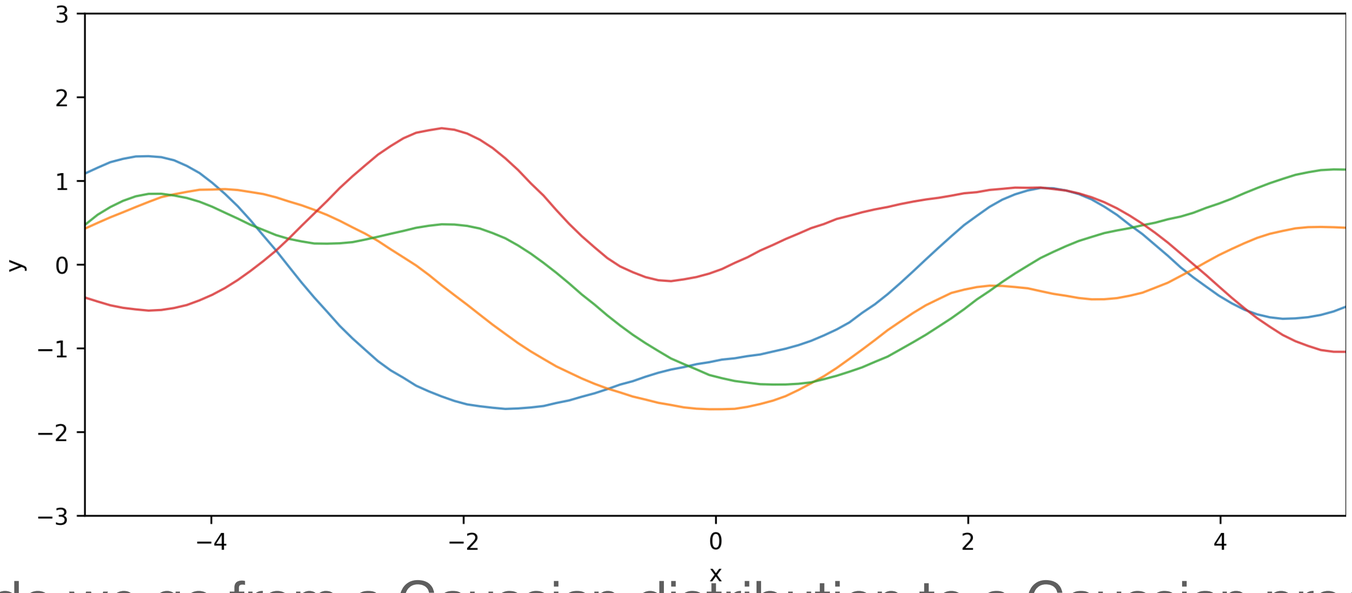
A *Gaussian process* is a collection of random variables, any finite number of which have a joint *Gaussian distribution*

Multivariate Gaussian distribution

- $\mathbf{f} \sim \mathrm{N}(\mu, \Sigma)$
- Mean μ
- Covariance matrix Σ

Gaussian process

- $f(x) \sim \mathrm{GP}(m(x), k(x, x'))$
- Mean function $m(x)$
- Covariance func. $k(x, x')$



How do we go from a Gaussian distribution to a Gaussian process?

Gaussian process regression

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $$p(\mathbf{y}_1, \mathbf{y_2}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$$

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y_2}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

observed

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y_2}) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

observed

Conditioning on the observed variable

$$p(\mathbf{y}_1 \,|\, \mathbf{y_2}) = \frac{p(\mathbf{y}_1, \mathbf{y_2})}{p(\mathbf{y_2})} = \mathrm{N}\left(\mathbf{y}_1; \mathbf{m}_1 + \mathbf{K}_{12}\mathbf{K}_{22}^{-1}(\mathbf{y}_2 - \mathbf{m}_2), \mathbf{K}_{11} - \mathbf{K}_{12}\mathbf{K}_{22}^{-1}\mathbf{K}_{21}\right)$$

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y}_2) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

observed

Conditioning on the observed variable

$$p(\mathbf{y}_1 \mid \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} = \mathrm{N}\left(\mathbf{y}_1; \mathbf{m}_1 + \mathbf{K}_{12}\mathbf{K}_{22}^{-1}(\mathbf{y}_2 - \mathbf{m}_2), \mathbf{K}_{11} - \mathbf{K}_{12}\mathbf{K}_{22}^{-1}\mathbf{K}_{21}\right)$$

prior

uncertainty reduction

Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y}_2) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

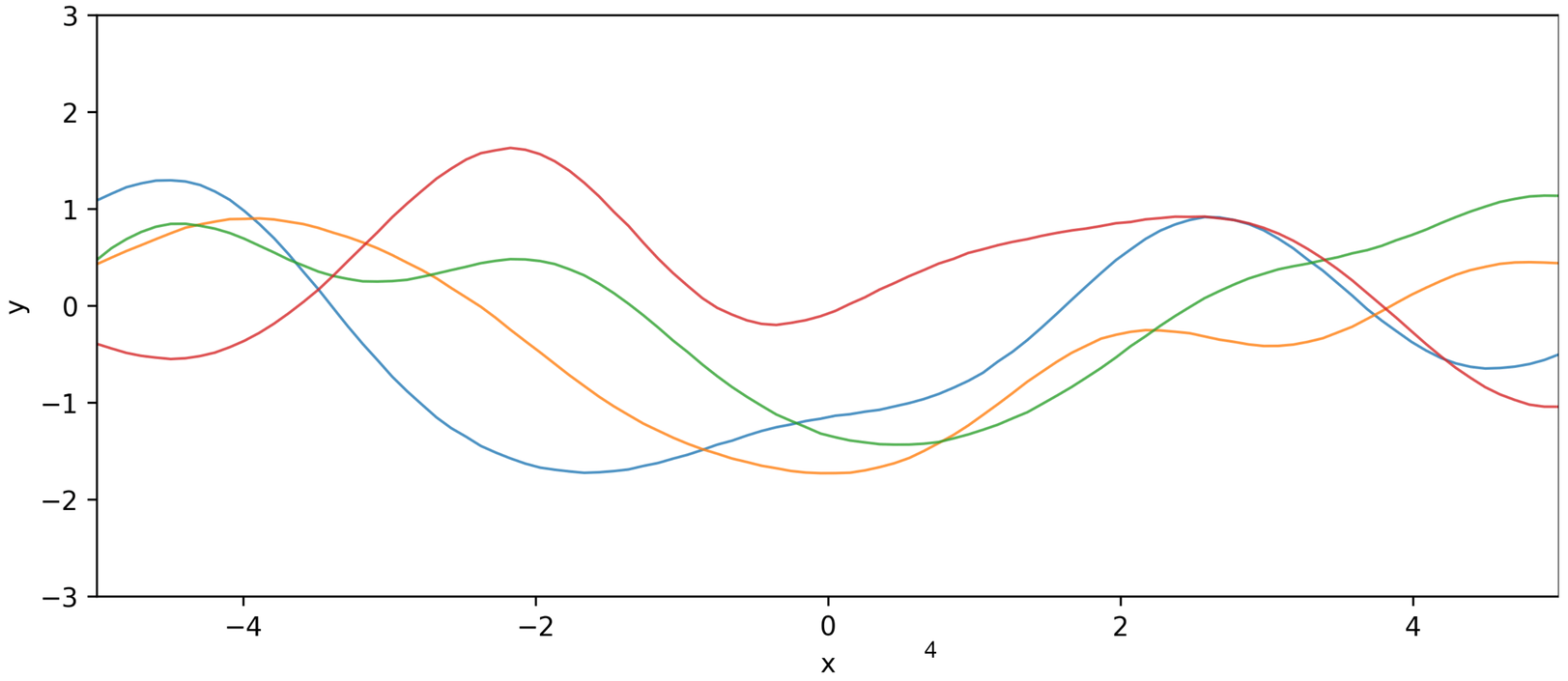
observed

Conditioning on the observed variable

$$p(\mathbf{y}_1 \mid \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} = \mathrm{N}\left(\mathbf{y}_1; \mathbf{m}_1 + \mathbf{K}_{12}\mathbf{K}_{22}^{-1}(\mathbf{y}_2 - \mathbf{m}_2), \mathbf{K}_{11} - \mathbf{K}_{12}\mathbf{K}_{22}^{-1}\mathbf{K}_{21}\right)$$

prior

uncertainty reduction



Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y}_2) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

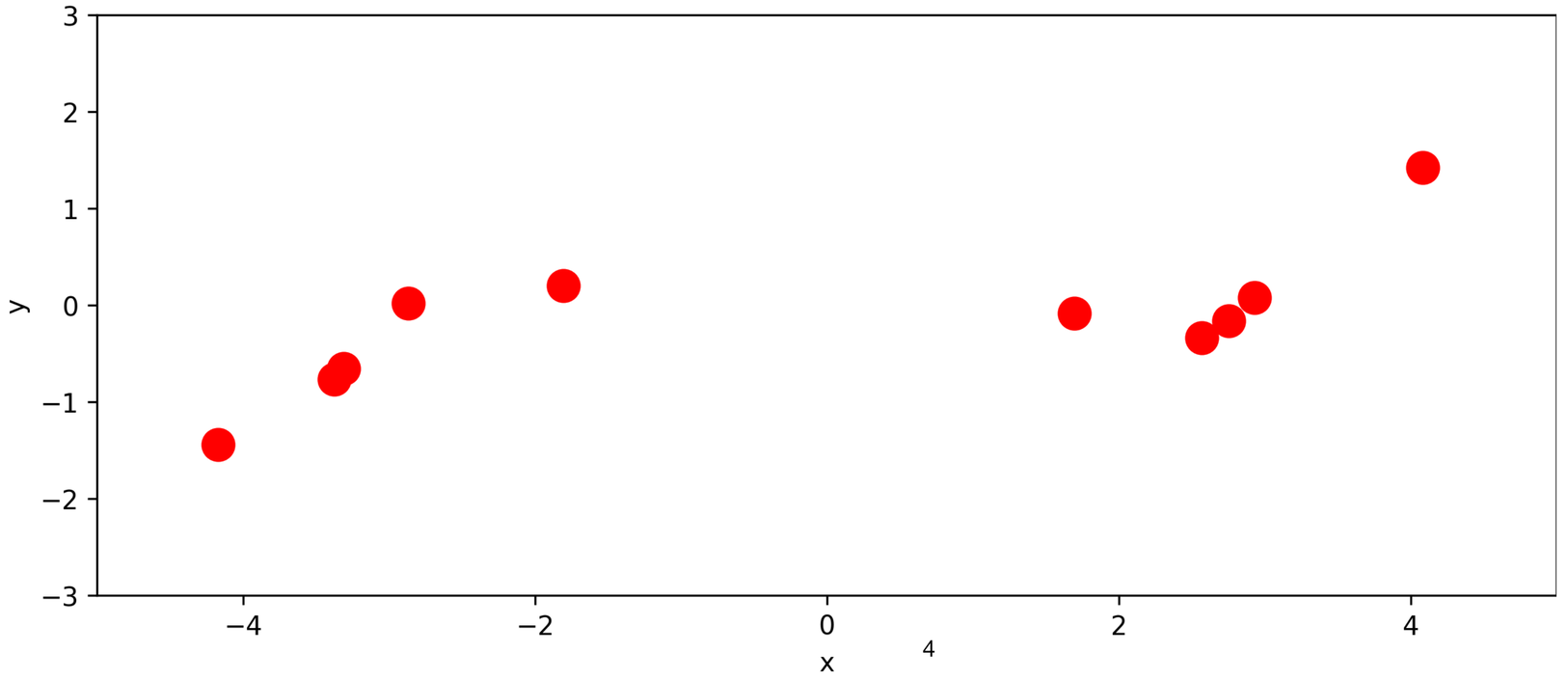
observed

Conditioning on the observed variable

$$p(\mathbf{y}_1 \mid \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} = \mathrm{N}\left(\mathbf{y}_1; \mathbf{m}_1 + \mathbf{K}_{12}\mathbf{K}_{22}^{-1}(\mathbf{y}_2 - \mathbf{m}_2), \mathbf{K}_{11} - \mathbf{K}_{12}\mathbf{K}_{22}^{-1}\mathbf{K}_{21}\right)$$

prior

uncertainty reduction



Gaussian process regression

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

$$f \sim \mathrm{GP}(m(x), k(x, x'))$$

$$\epsilon \sim \mathrm{N}(0,1)$$

Joint distribution $p(\mathbf{y}_1, \mathbf{y}_2) = \mathrm{N}\left(\begin{bmatrix} \mathbf{y}_1 \\ \mathbf{y}_2 \end{bmatrix}; \begin{bmatrix} \mathbf{m}_1 \\ \mathbf{m}_2 \end{bmatrix}, \begin{bmatrix} \mathbf{K}_{11} & \mathbf{K}_{12} \\ \mathbf{K}_{21} & \mathbf{K}_{22} \end{bmatrix} \right)$

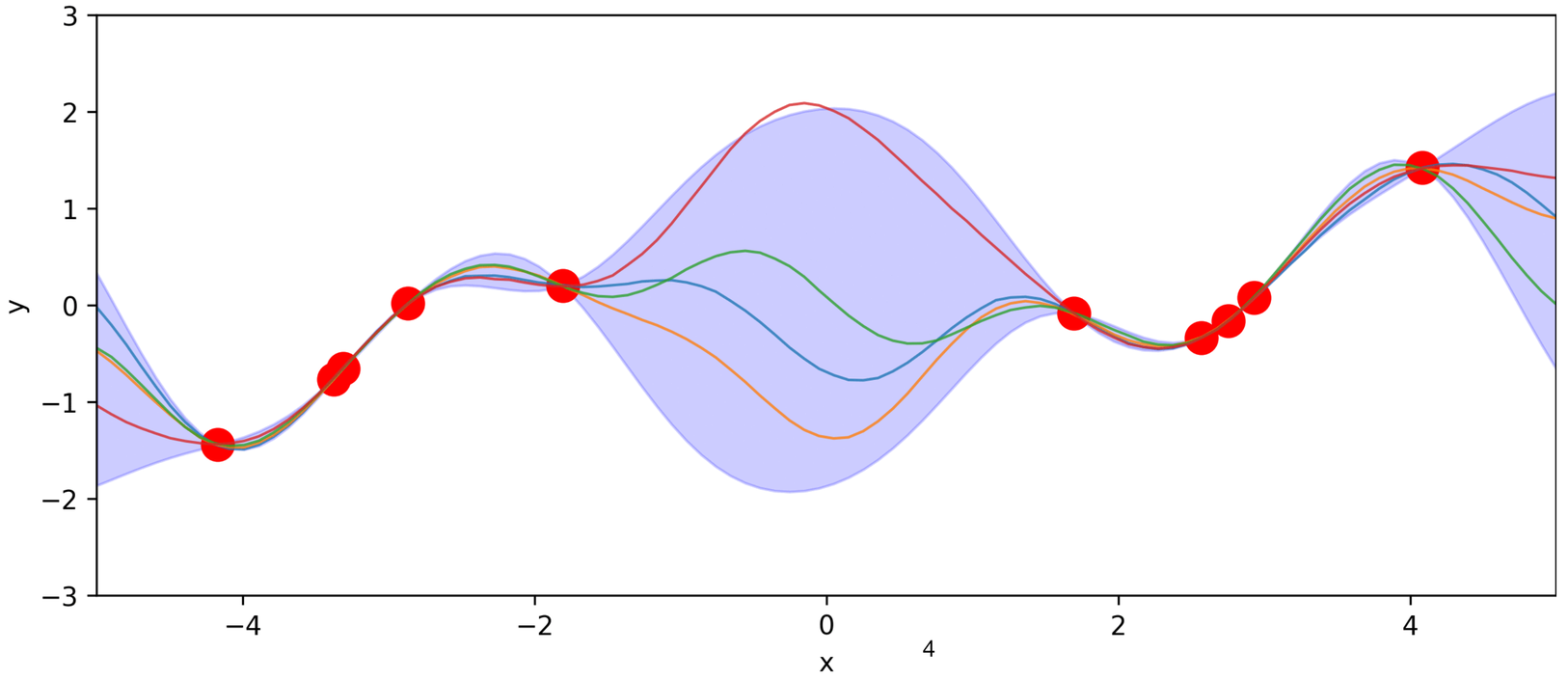
observed

Conditioning on the observed variable

$$p(\mathbf{y}_1 \mid \mathbf{y}_2) = \frac{p(\mathbf{y}_1, \mathbf{y}_2)}{p(\mathbf{y}_2)} = \mathrm{N}\left(\mathbf{y}_1; \mathbf{m}_1 + \mathbf{K}_{12}\mathbf{K}_{22}^{-1}(\mathbf{y}_2 - \mathbf{m}_2), \mathbf{K}_{11} - \mathbf{K}_{12}\mathbf{K}_{22}^{-1}\mathbf{K}_{21}\right)$$

prior

uncertainty reduction



Algorithm

$$p(y^* \mid \mathbf{x}^*, \mathbf{X}, \mathbf{y}) = \mathcal{N}(y^*; m(x^*), \sigma^2(x^*))$$

$$m(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{X})\,(k(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathrm{I})^{-1}\,\mathbf{y} \qquad \mathcal{O}(N^3)$$

$$\sigma^2(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{x}^*) - k(\mathbf{x}^*, \mathbf{X})\,(k(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathrm{I})^{-1}\,k(\mathbf{X}, \mathbf{x}^*)$$

$N^3/3$ operations

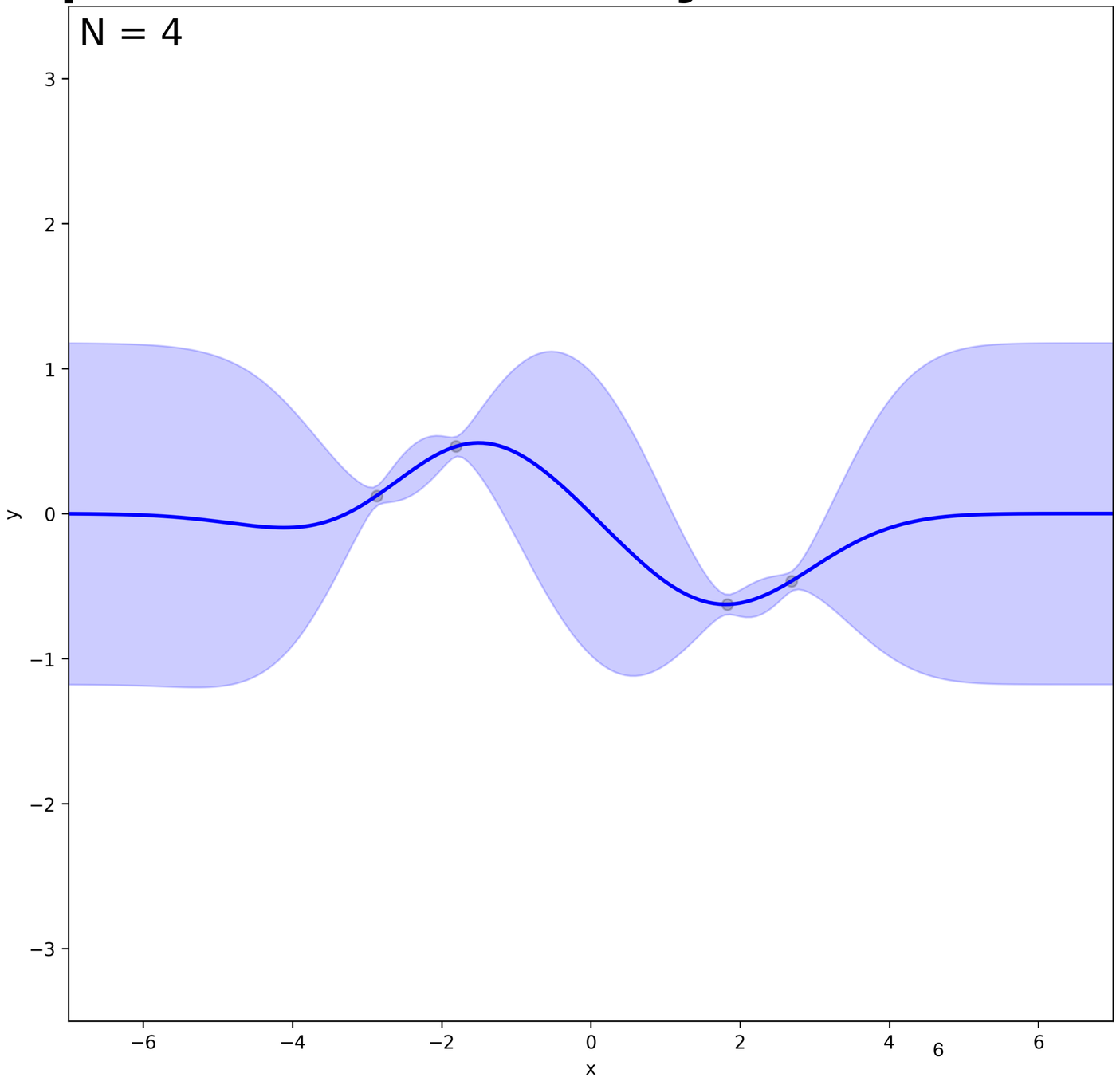
- $L = \text{Cholesky}\,(k(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathrm{I})$
- $\boldsymbol{\alpha} = \mathbf{L}^{\mathbf{T}} \backslash (\mathbf{L} \backslash \mathbf{y})$
- $m(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{X}) \cdot \boldsymbol{\alpha}$
- $\mathbf{v} = \mathbf{L} \backslash k(\mathbf{X}, \mathbf{x}^*)$
- $\sigma^2(\mathbf{x}^*) = k(\mathbf{x}^*, \mathbf{x}^*) - \mathbf{v}^{\mathbf{T}}\mathbf{v}$

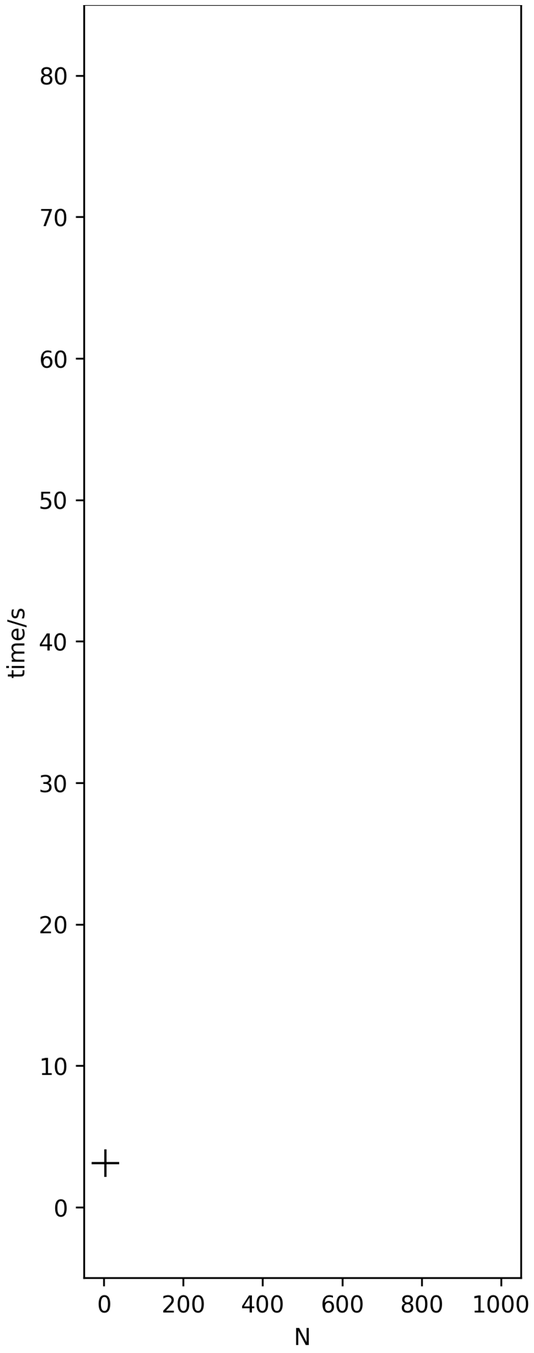
$$k(\mathbf{X}, \mathbf{X}) + \sigma^2 \mathrm{I} = \mathbf{L} \cdot \mathbf{L}^T$$

$\mathbf{L}$ lower-triangular

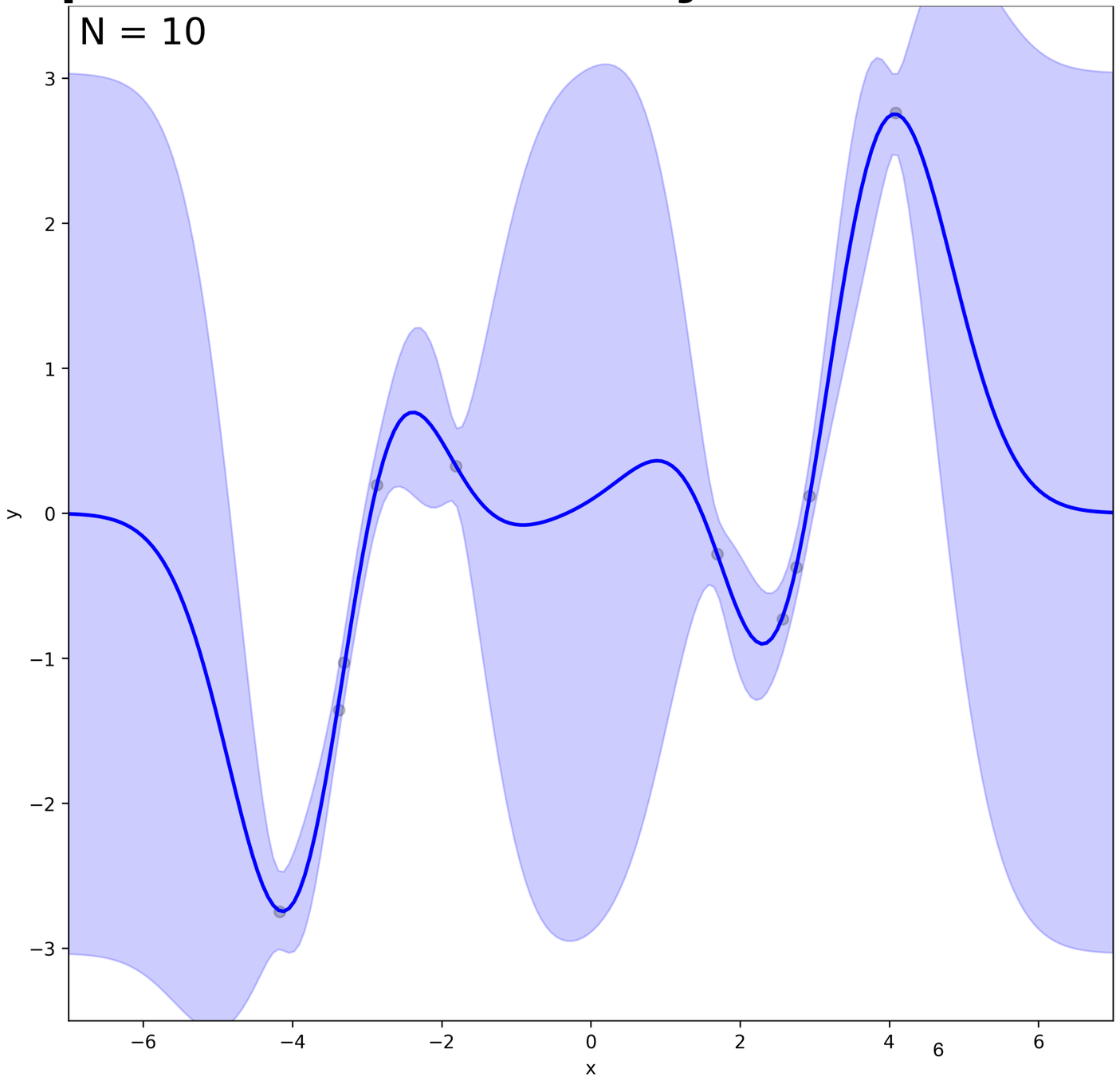
$$\mathbf{A}\mathbf{x} = \mathbf{b} \Rightarrow \mathbf{x} = \mathbf{A} \backslash \mathbf{b}$$

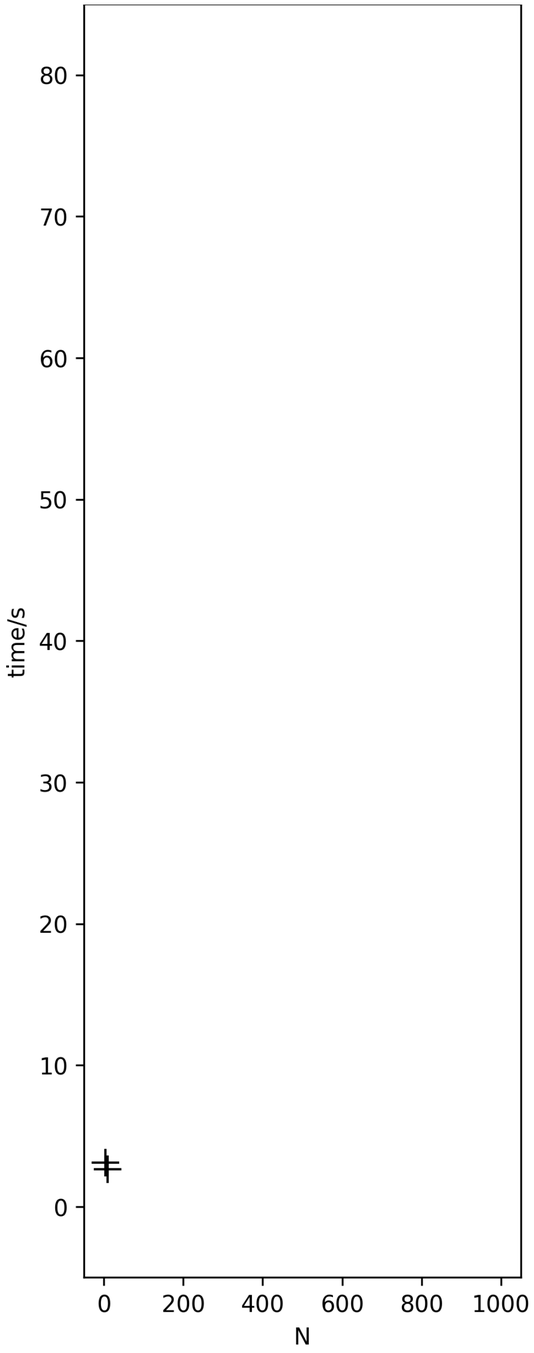
Computational intractability



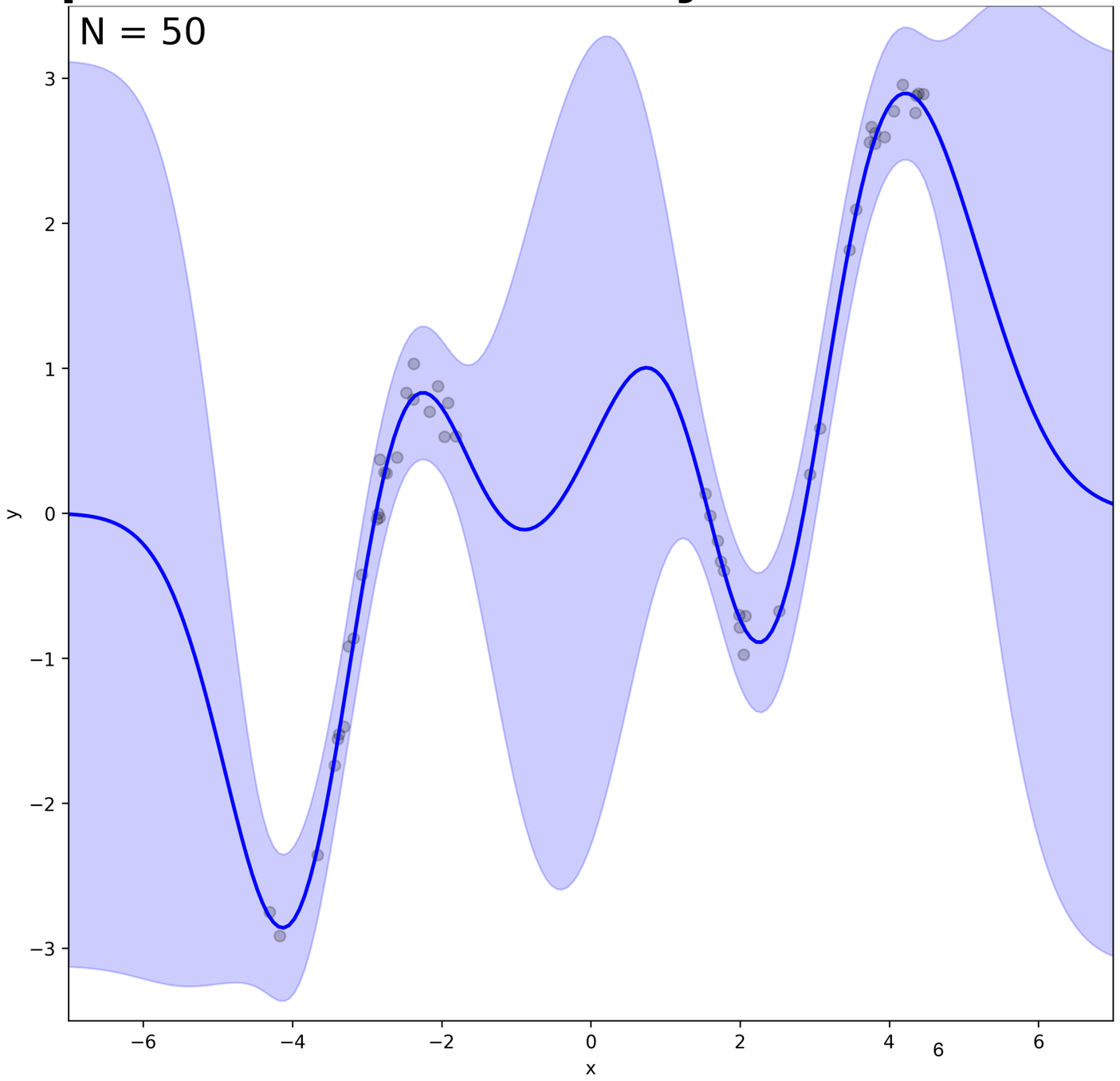


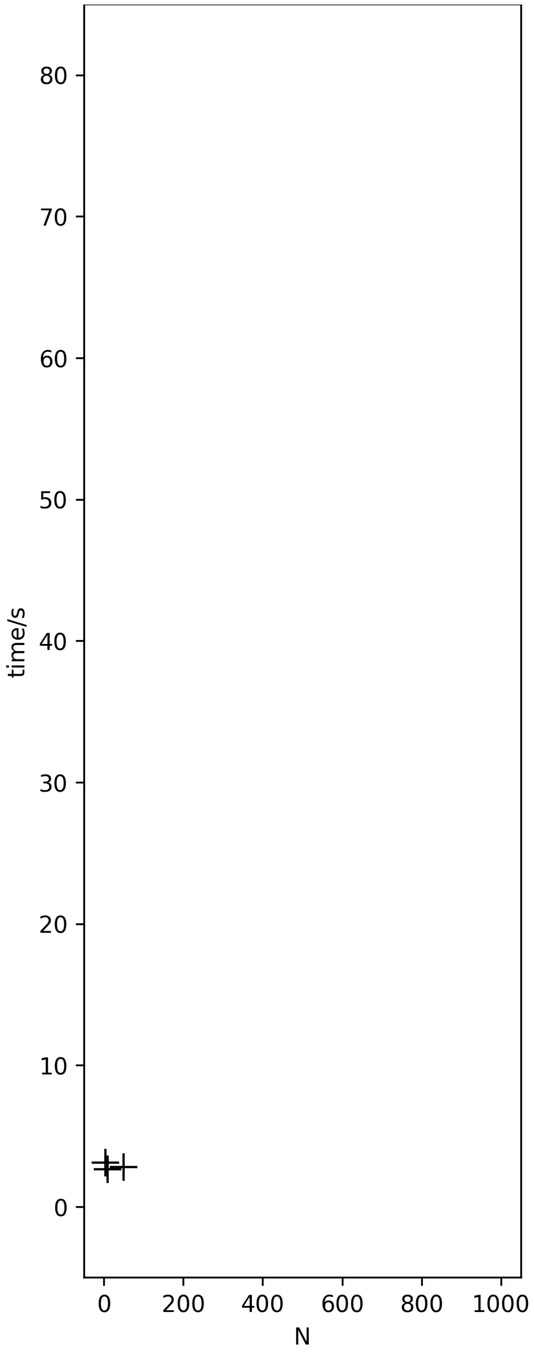
Computational intractability



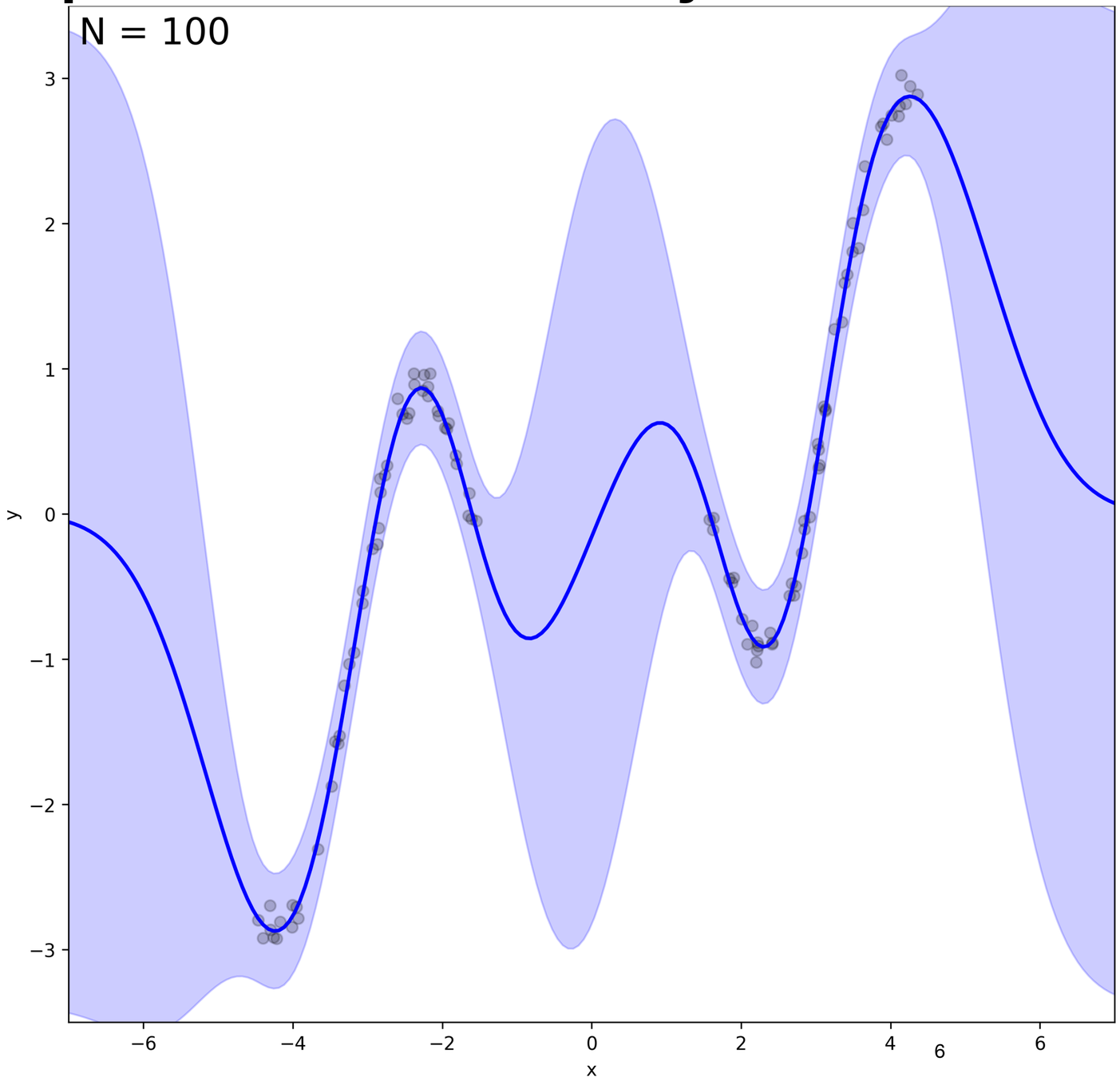


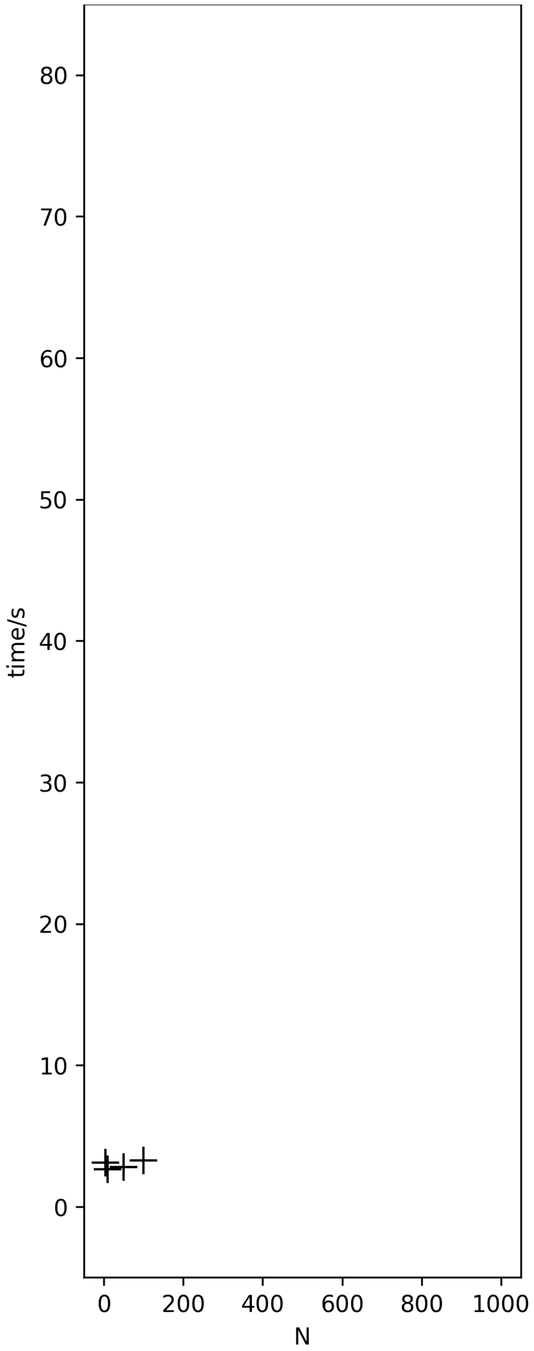
Computational intractability



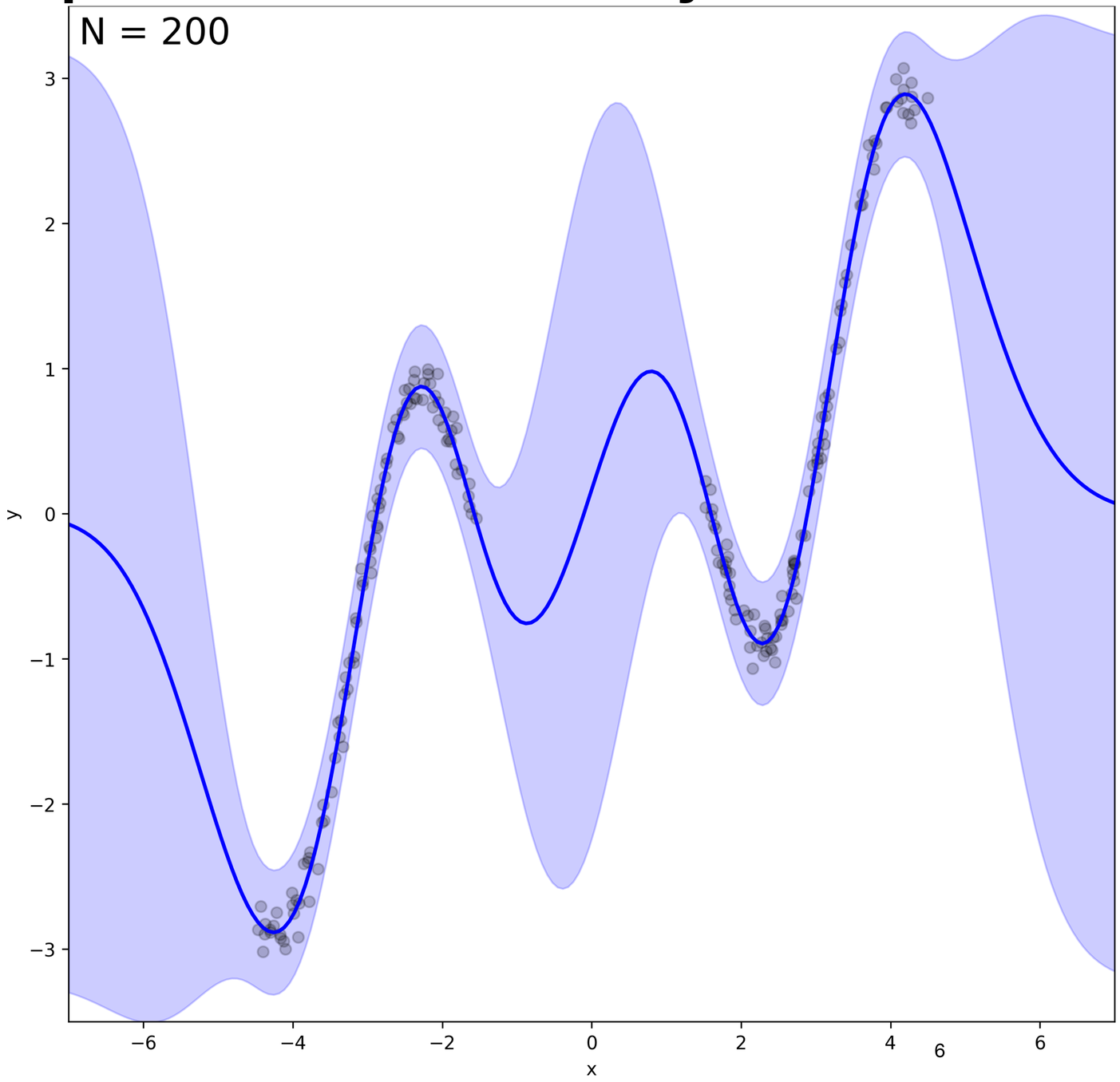


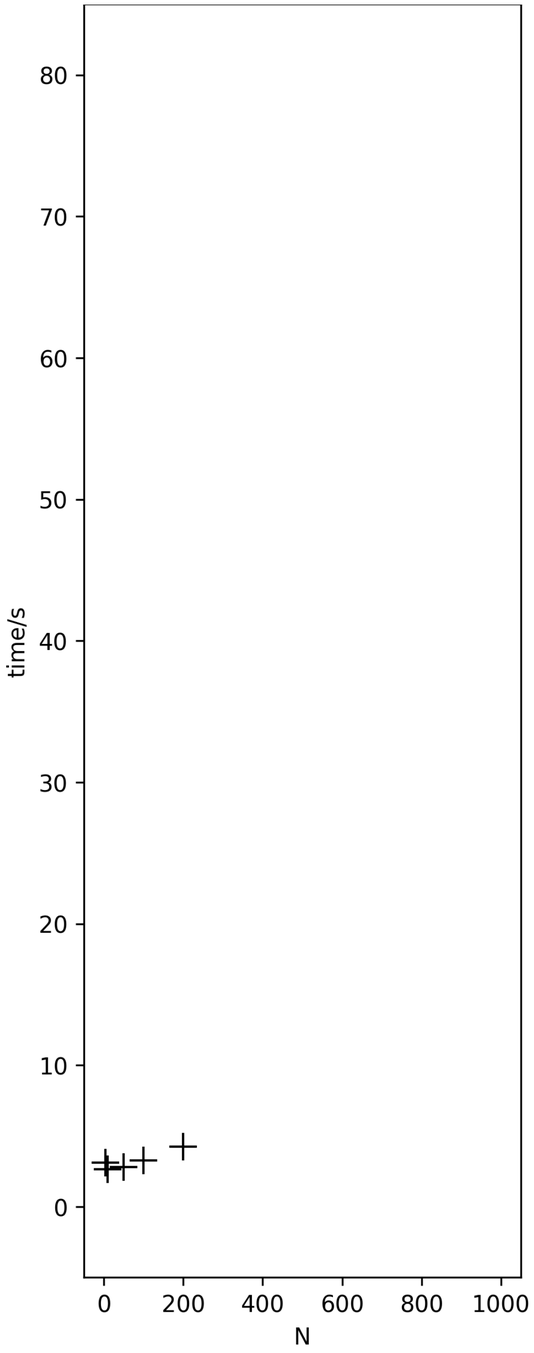
Computational intractability



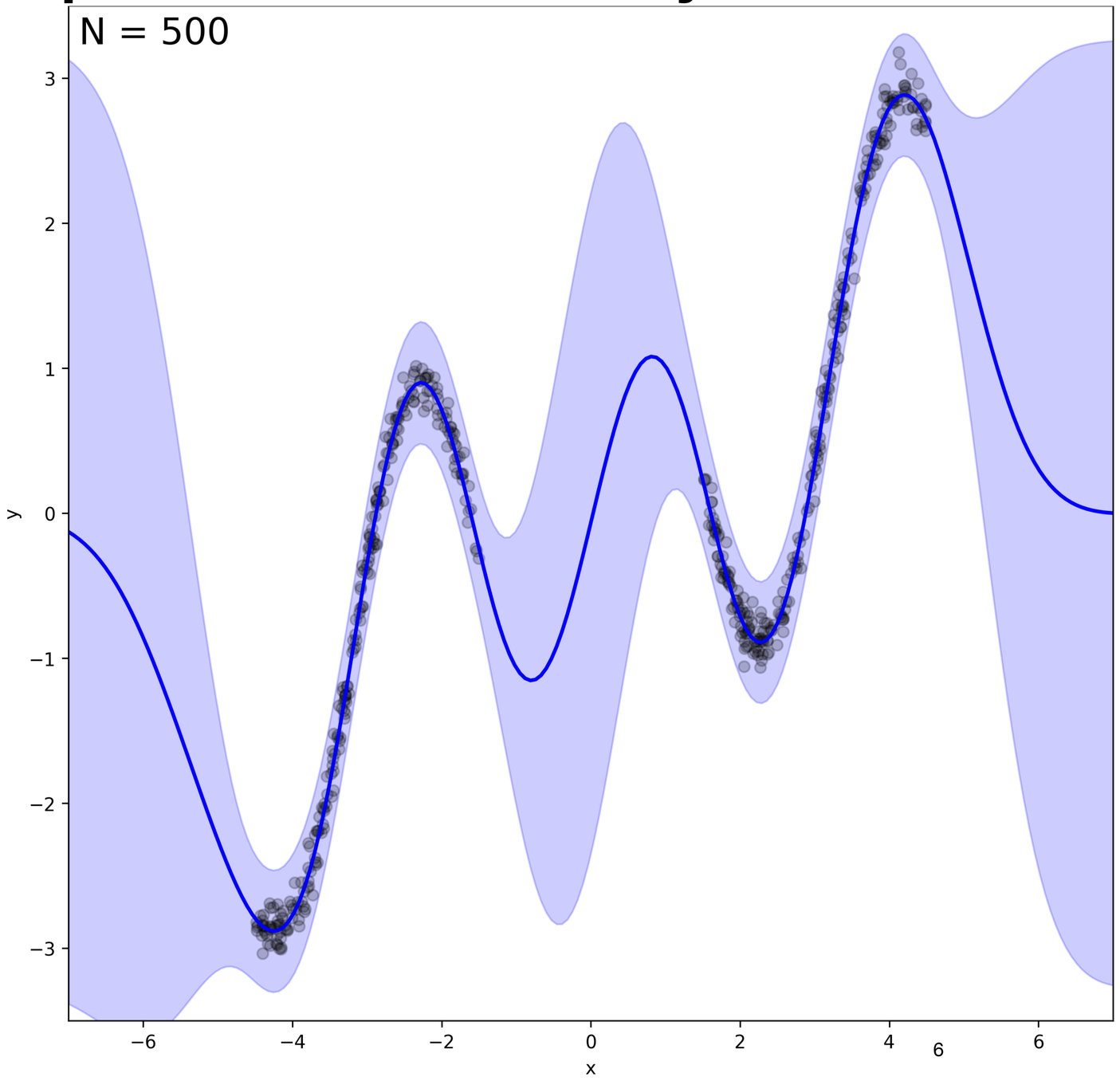


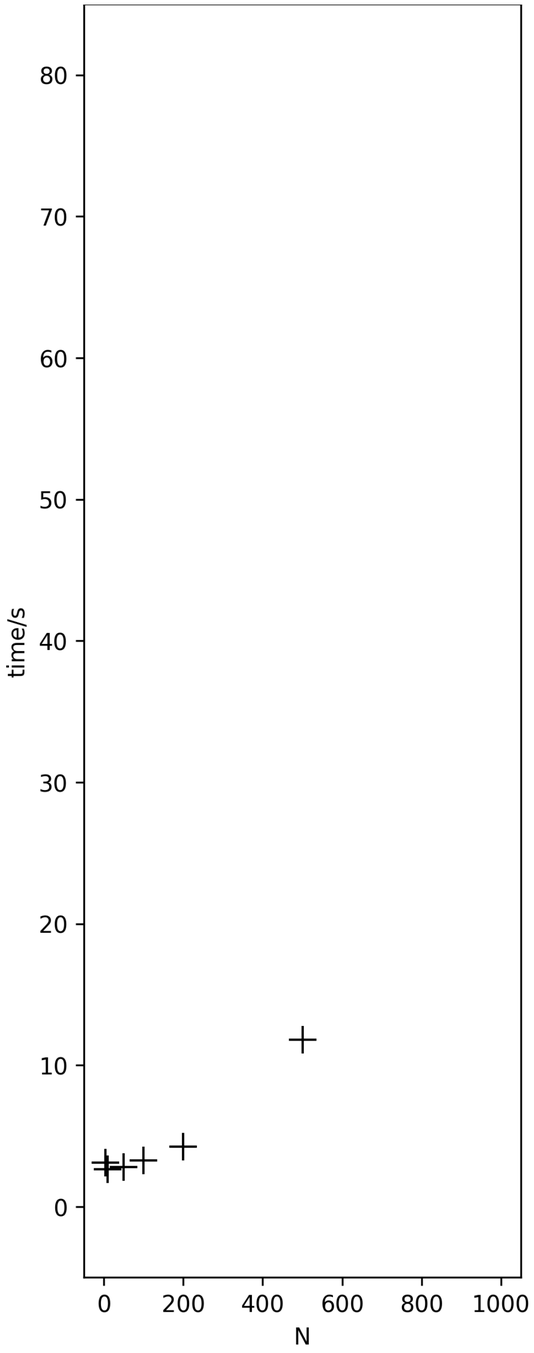
Computational intractability



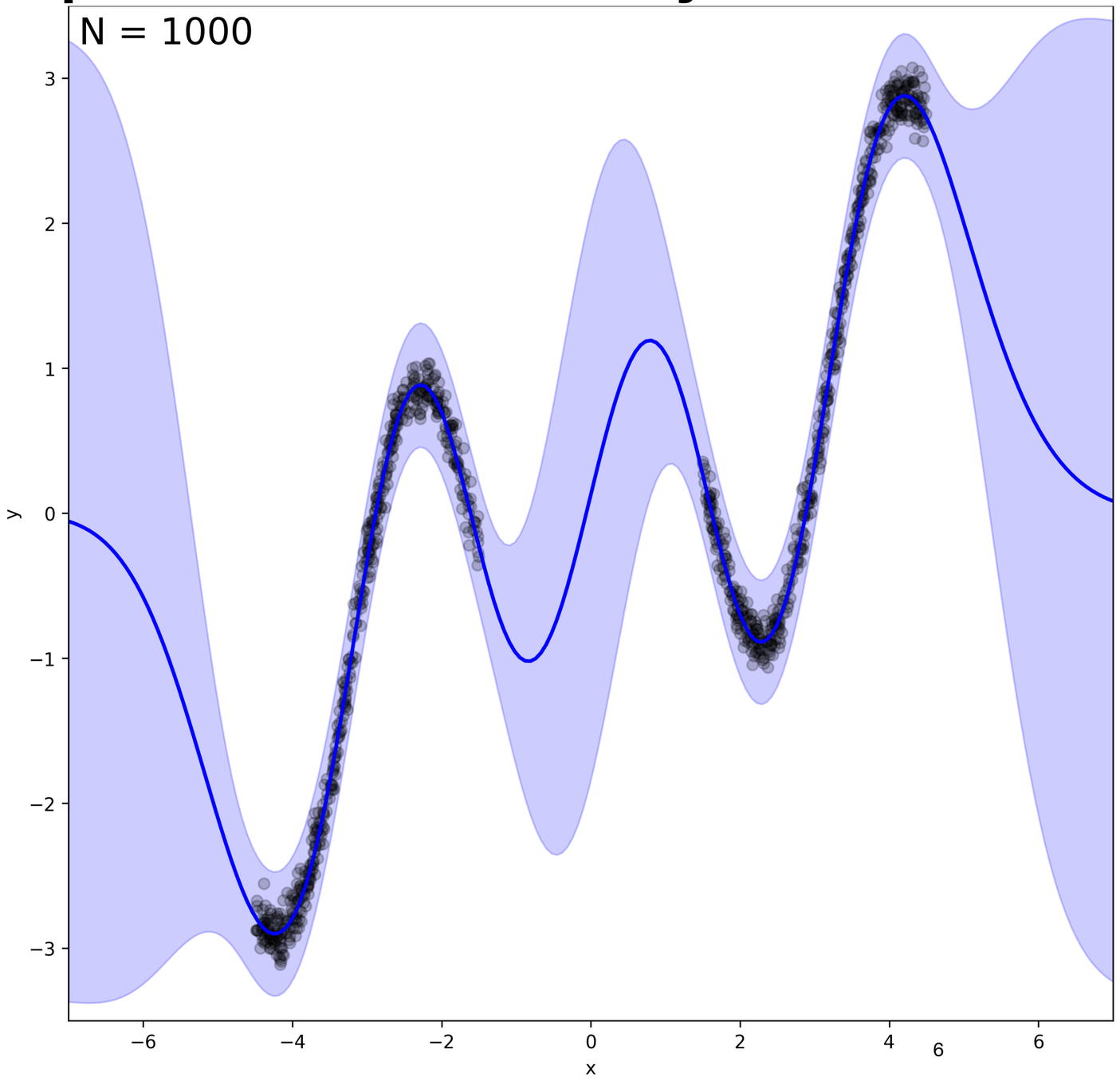


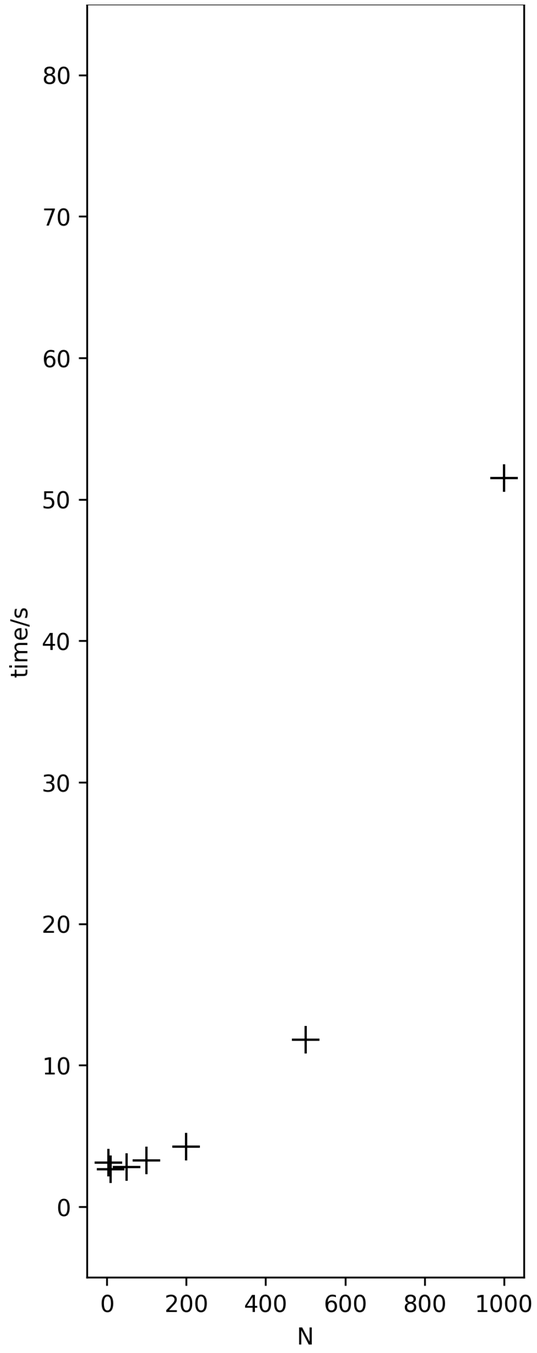
Computational intractability



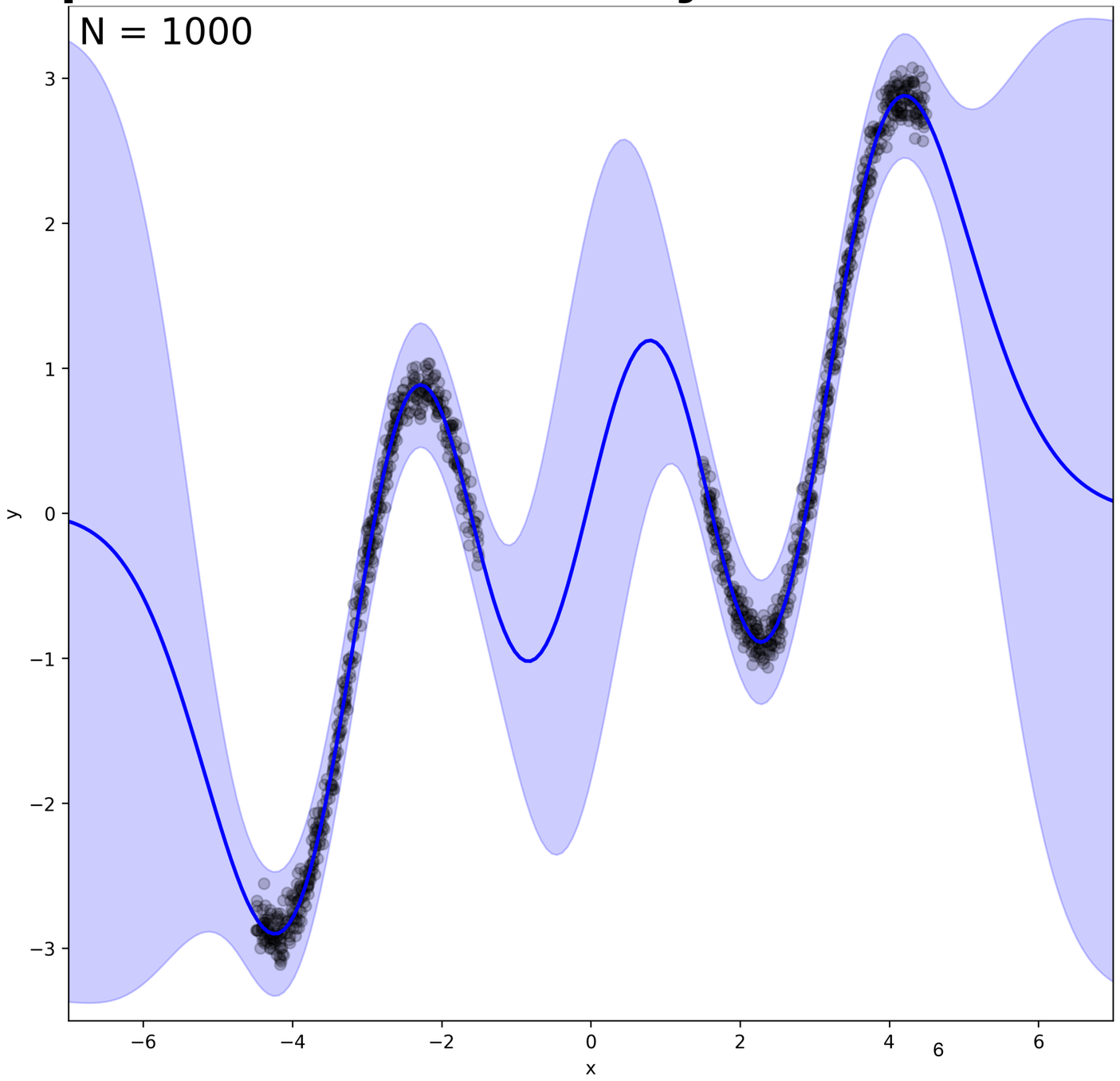


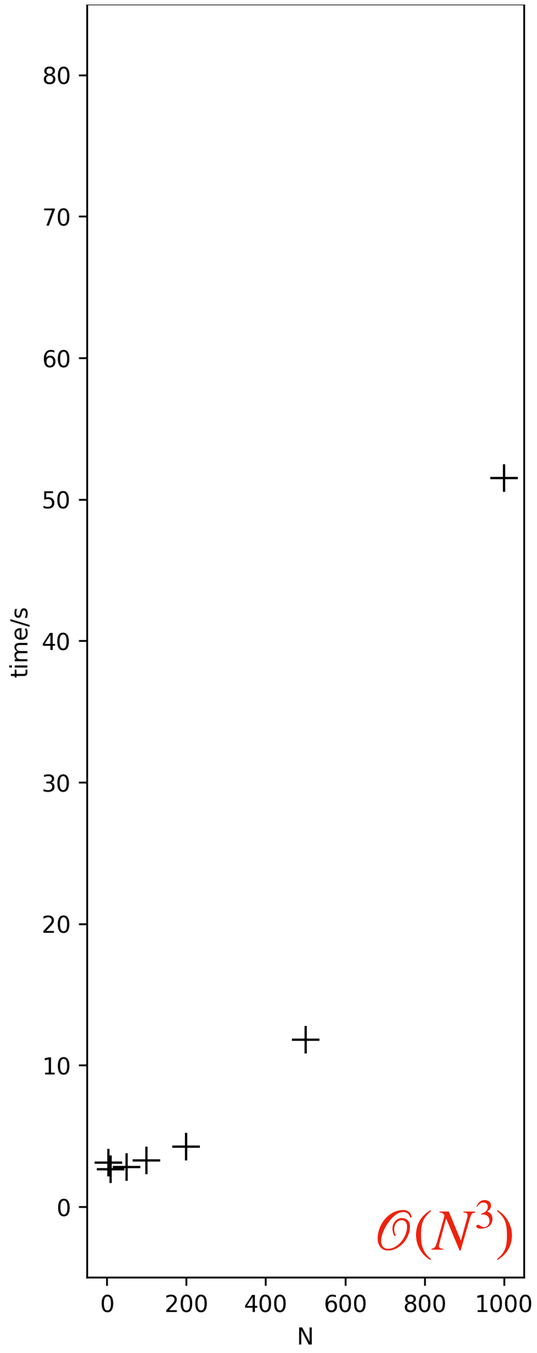
Computational intractability



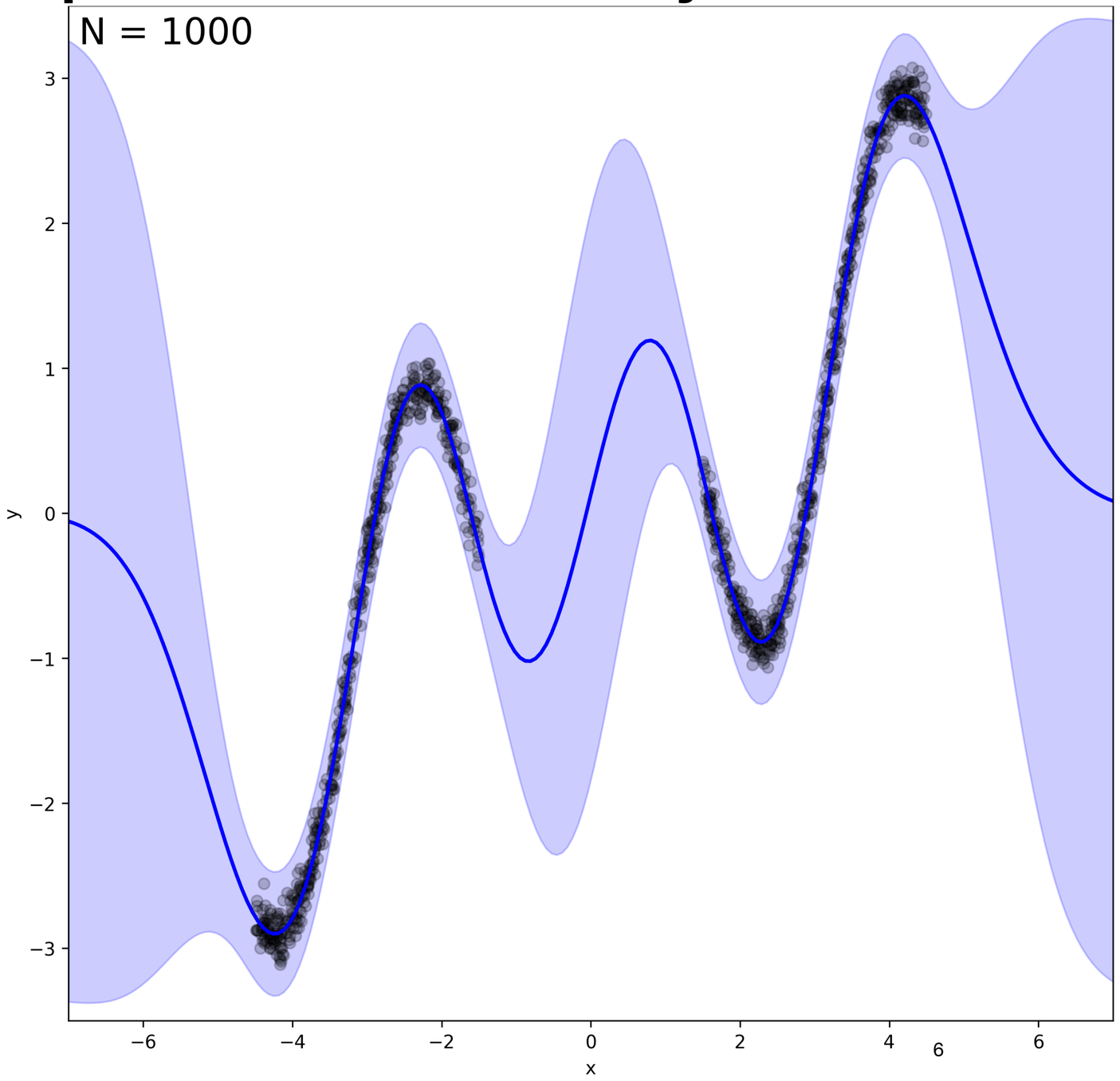


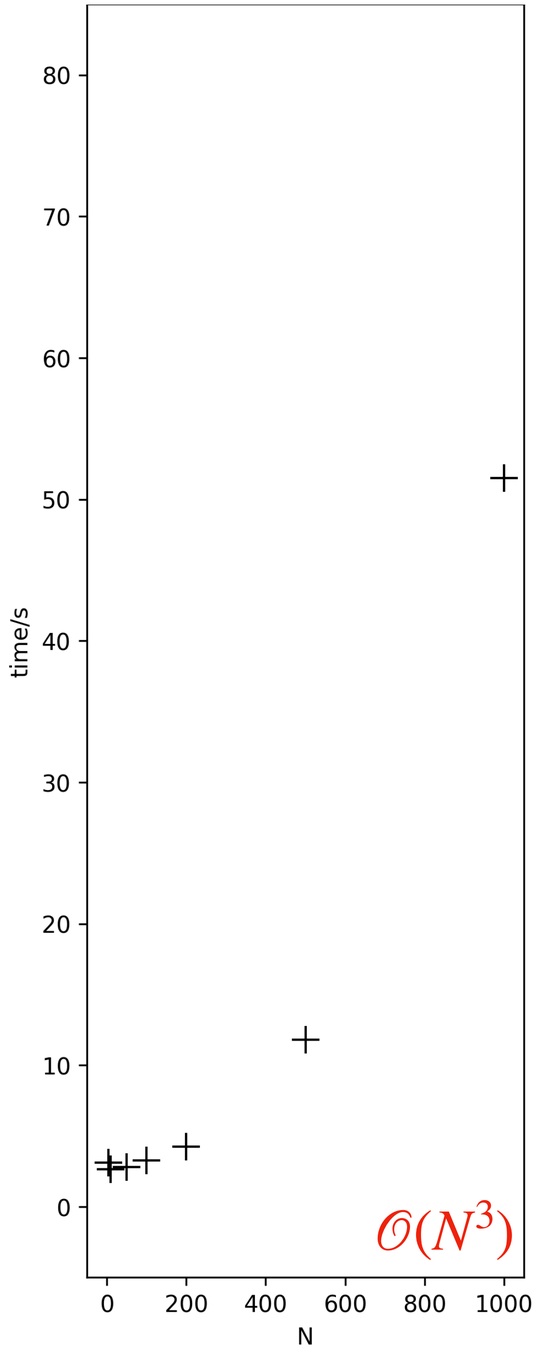
Computational intractability





Computational intractability





N = 1e6
Time = 1500 years!

Approximation using pseudo-data or inducing points

Approximation using pseudo-data or inducing points

Exact posterior

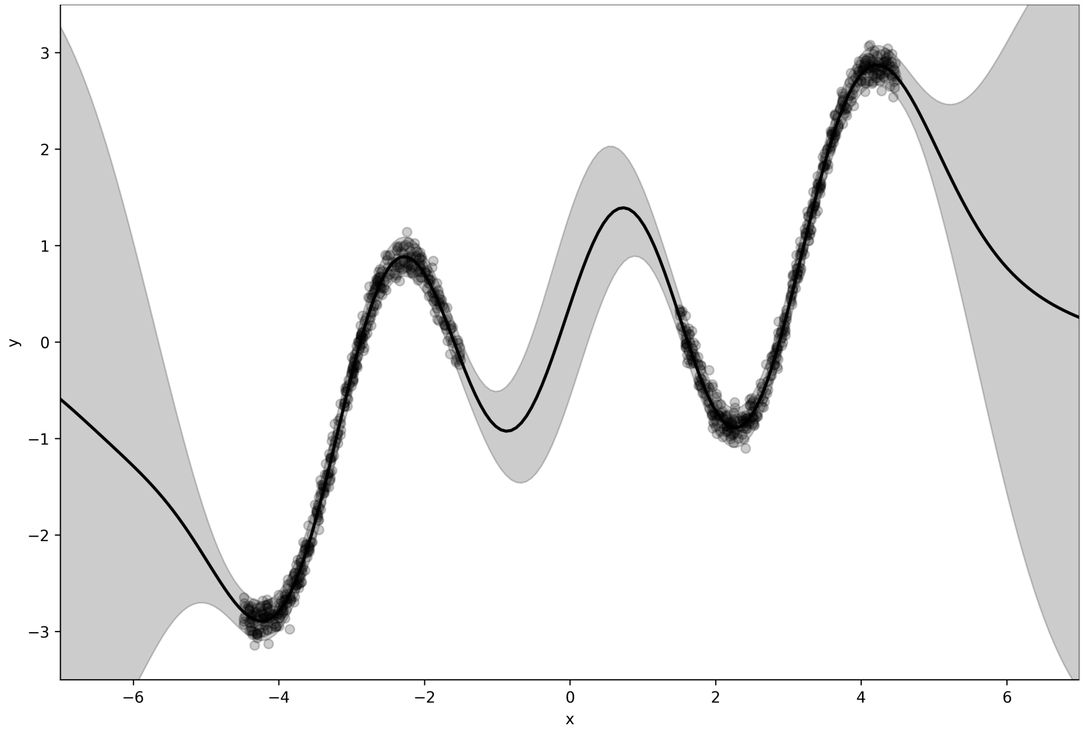
Prior Likelihood

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Marginal likelihood

for hyperparameter optimisation aka finding theta

$\mathcal{O}(N^3)$ where $N = \dim(\mathbf{x}, \mathbf{y})$



Approximation using pseudo-data or inducing points

Approximate posterior

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

$$f = \{\mathbf{u}, f_{\neq \mathbf{u}}\} \qquad q(\mathbf{u}) = \mathrm{N}(\mathbf{u}; \mu, \Sigma)$$

Exact posterior

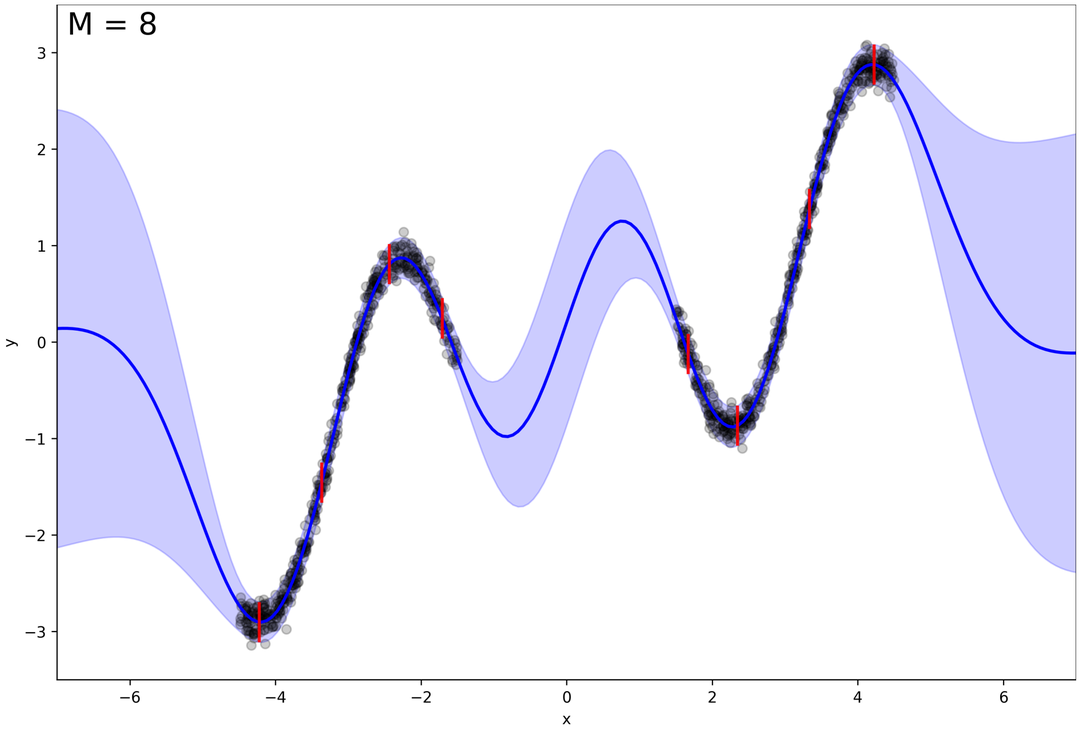
$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Prior Likelihood

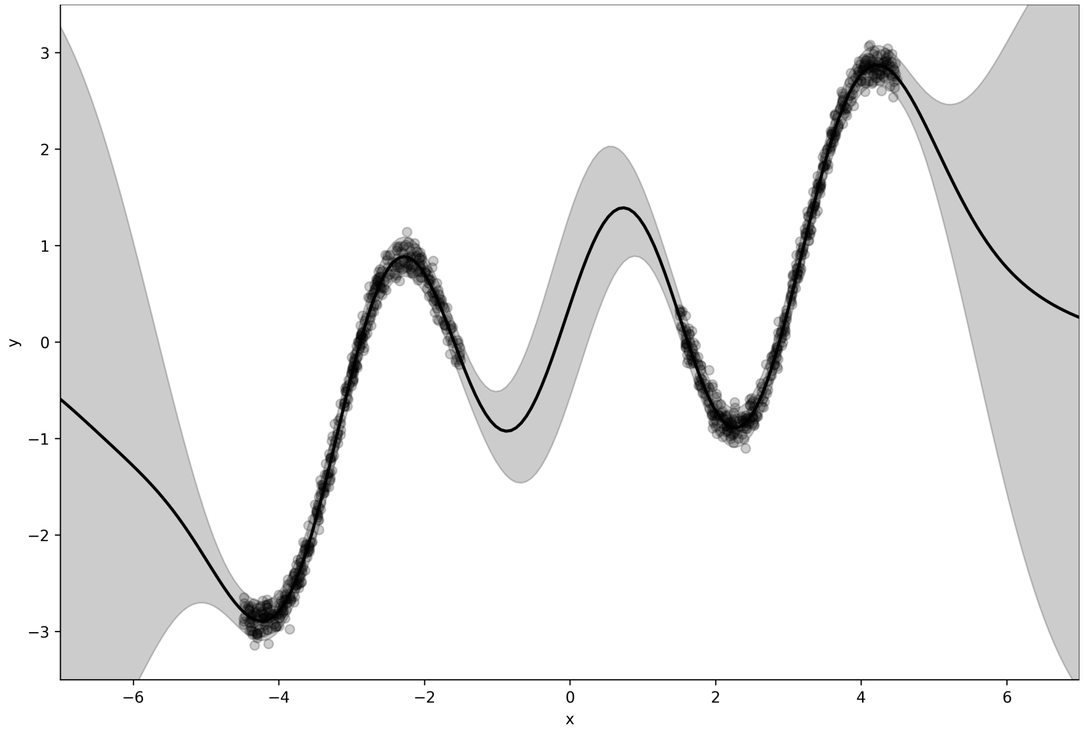
Marginal likelihood

for hyperparameter optimisation aka finding theta

$\mathcal{O}(NM^2)$ where $M = \dim(\mathbf{u}) \ll N$



$\mathcal{O}(N^3)$ where $N = \dim(\mathbf{x}, \mathbf{y})$



Approximation using pseudo-data or inducing points

Approximate posterior

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

$$f = \{\mathbf{u}, f_{\neq \mathbf{u}}\} \qquad q(\mathbf{u}) = \mathrm{N}(\mathbf{u}; \mu, \Sigma)$$

Exact posterior

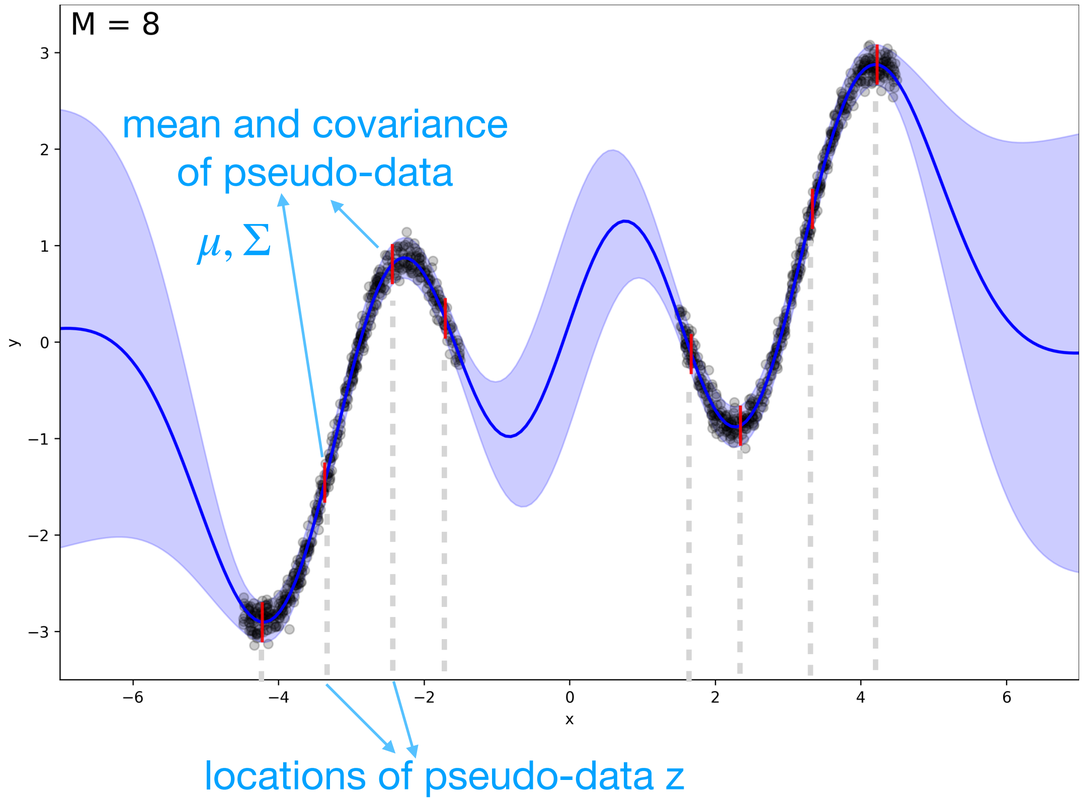
$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Prior Likelihood

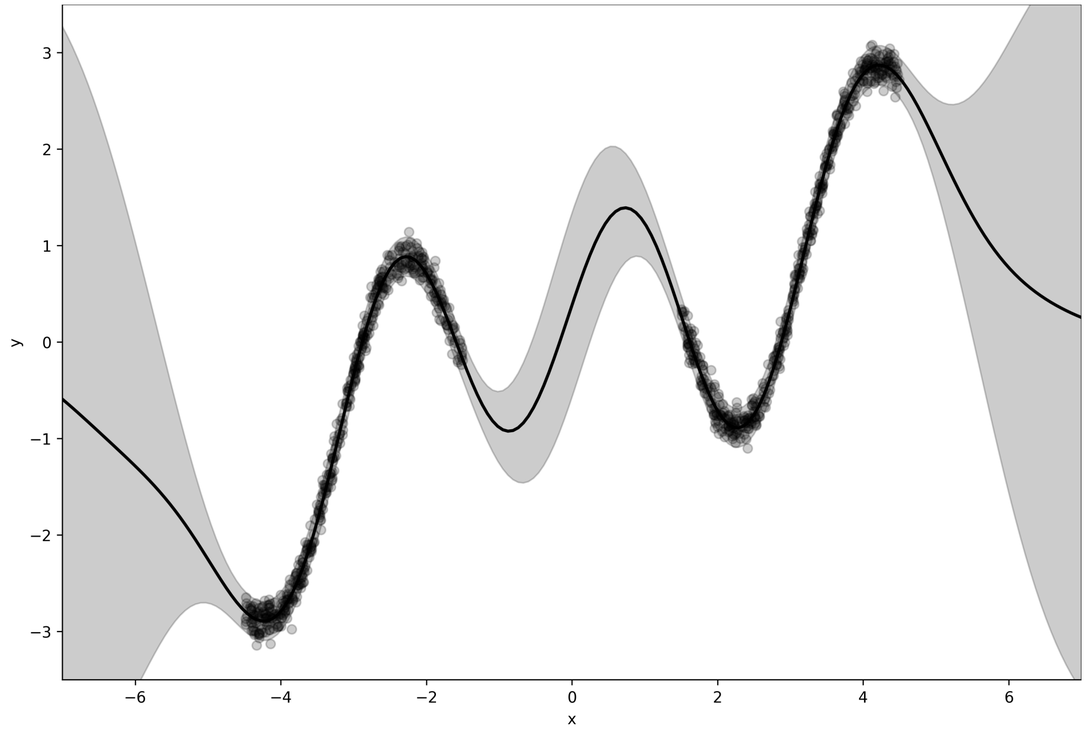
Marginal likelihood

for hyperparameter optimisation aka finding theta

$\mathcal{O}(NM^2)$ where $M = \dim(\mathbf{u}) \ll N$



$\mathcal{O}(N^3)$ where $N = \dim(\mathbf{x}, \mathbf{y})$



Approximation using pseudo-data or inducing points

Approximate posterior

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

$$f = \{\mathbf{u}, f_{\neq \mathbf{u}}\} \qquad q(\mathbf{u}) = \mathrm{N}(\mathbf{u}; \mu, \Sigma)$$

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Prior Likelihood

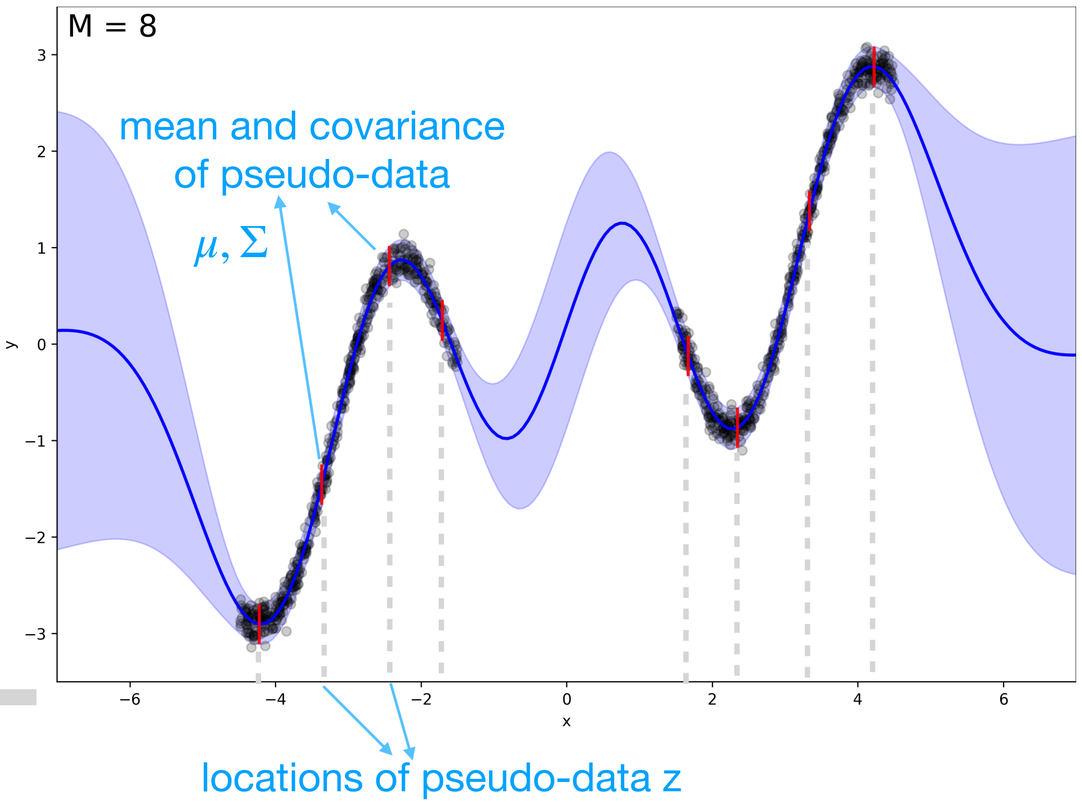
Marginal likelihood

for hyperparameter optimisation aka finding theta

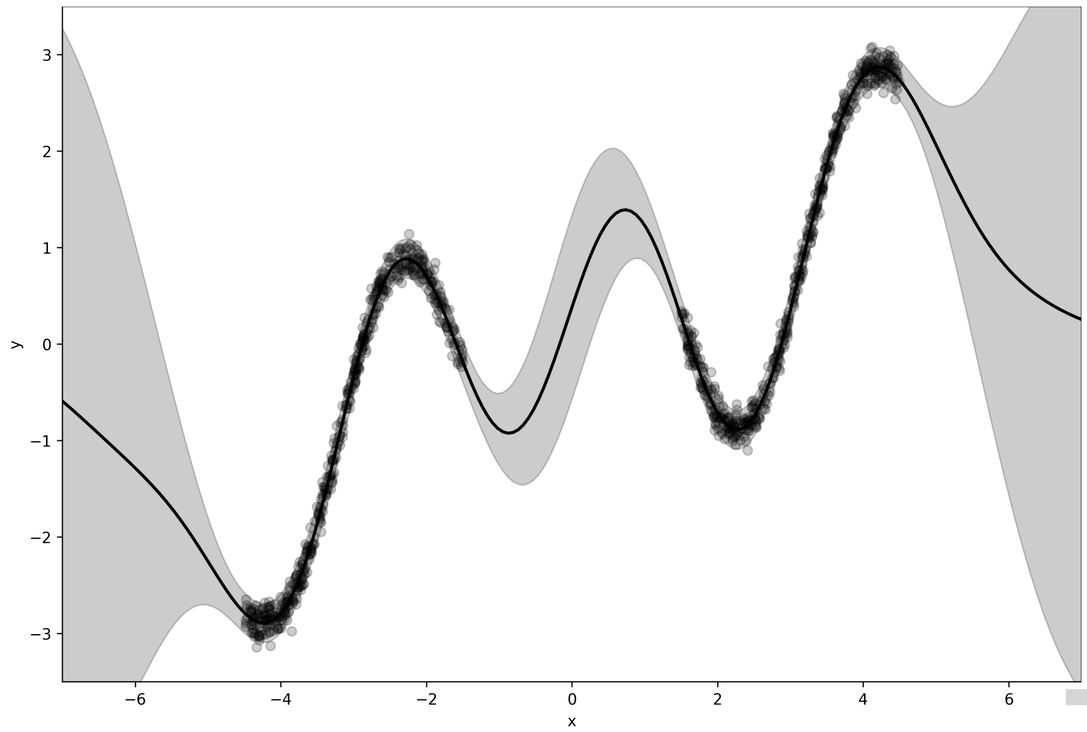
KL

variational inference

$\mathcal{O}(NM^2)$ where $M = \dim(\mathbf{u}) \ll N$



$\mathcal{O}(N^3)$ where $N = \dim(\mathbf{x}, \mathbf{y})$



Approximation using pseudo-data or inducing points

Approximate posterior

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

$$f = \{\mathbf{u}, f_{\neq \mathbf{u}}\} \qquad q(\mathbf{u}) = \mathrm{N}(\mathbf{u}; \mu, \Sigma)$$

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Prior Likelihood

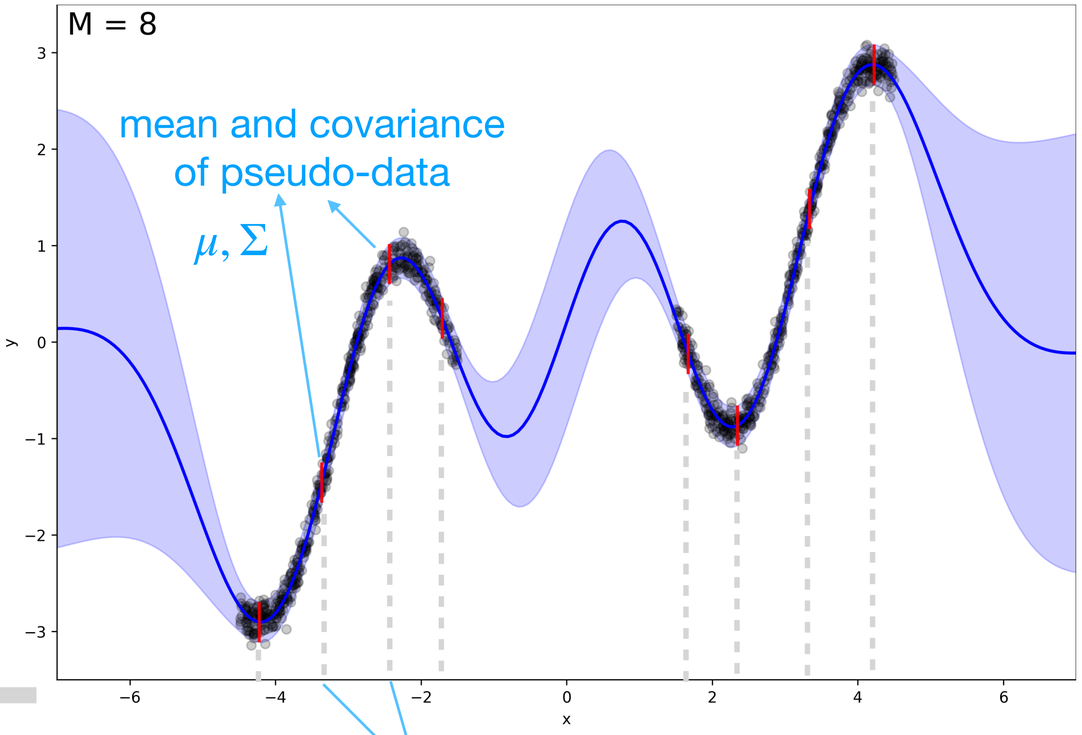
Marginal likelihood

for hyperparameter optimisation aka finding theta

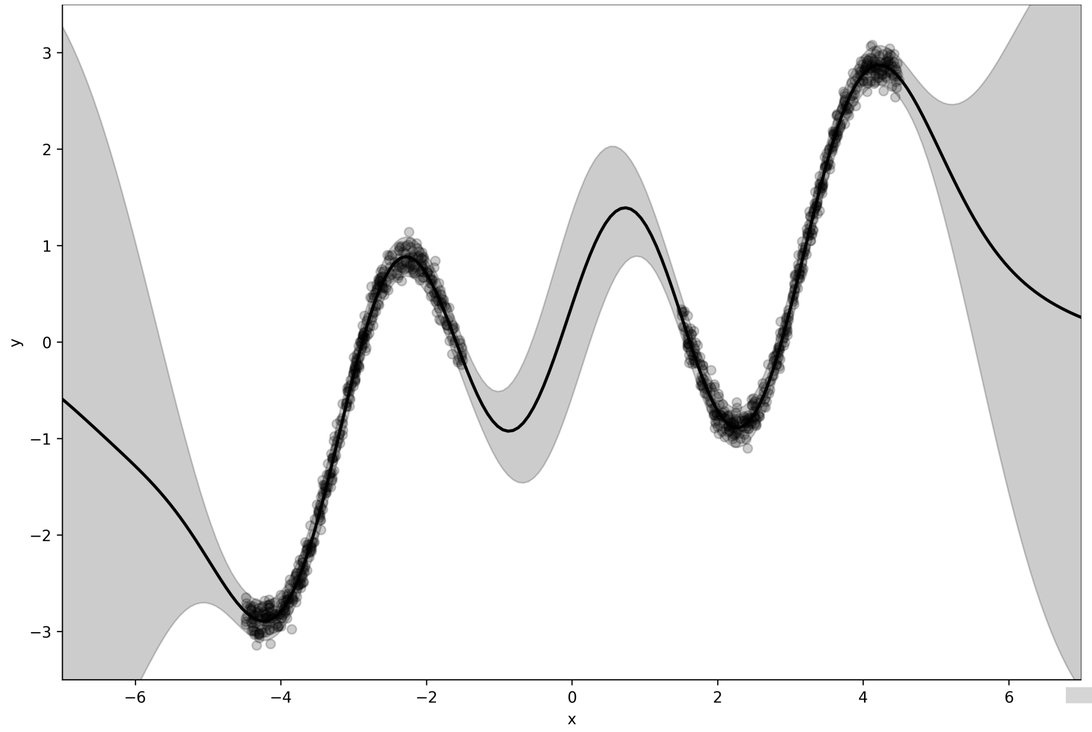
KL

variational inference

$\mathcal{O}(NM^2)$ where $M = \dim(\mathbf{u}) \ll N$



$\mathcal{O}(N^3)$ where $N = \dim(\mathbf{x}, \mathbf{y})$



Variational inference

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) - \mathcal{F}(q(\mathbf{u}), \theta)$$

F = negative of the integral term

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{\cancel{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta)} q(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta)} p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) - \mathcal{F}(q(\mathbf{u}), \theta)$$

F = negative of the integral term

minimising KL(approx post||exact post) $\equiv$ maximising $\mathcal{F}$

as $\log p(\mathbf{y} \mid \mathbf{x}, \theta)$ does not depend on q

$\mathcal{F}$: variational lower bound or negative variational free-energy

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{\cancel{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta)} q(\mathbf{u})}{\cancel{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta)} p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) - \mathcal{F}(q(\mathbf{u}), \theta)$$

F = negative of the integral term

minimising KL(approx post$\mid\mid$exact post) $\equiv$ maximising $\mathcal{F}$

as $\log p(\mathbf{y} \mid \mathbf{x}, \theta)$ does not depend on q

$\mathcal{F}$: variational lower bound or negative variational free-energy

$$\mathcal{F}(q(\mathbf{u}), \theta) = \int_f q(f \mid \theta) \log \frac{p(\mathbf{u} \mid \theta)}{q(\mathbf{u})} + \int_f q(f \mid \theta) \log p(\mathbf{y} \mid f, \mathbf{x}, \theta)$$

conditional term cancelled out, split the log term

Variational inference

Approximate posterior

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

Exact posterior

$$p(f \mid \mathbf{x}, \mathbf{y}, \theta) = \frac{p(f \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}{p(\mathbf{y} \mid \mathbf{x}, \theta)}$$

$$\mathrm{KL}(q(f \mid \theta) \parallel p(f \mid \mathbf{x}, \mathbf{y}, \theta)) = \int_f q(f \mid \theta) \log \frac{q(f \mid \theta)}{p(f \mid \mathbf{x}, \mathbf{y}, \theta)}$$

Kullback-Leibler divergence definition

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) + \int_f q(f \mid \theta) \log \frac{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) q(\mathbf{u})}{p(f_{\neq \mathbf{u}} \mid \mathbf{u}, \theta) p(\mathbf{u} \mid \theta) p(\mathbf{y} \mid f, \mathbf{x}, \theta)}$$

sub in q and p

noting $p(f) = p(f_{\neq \mathbf{u}} \mid \mathbf{u}) p(\mathbf{u})$

$$= \log p(\mathbf{y} \mid \mathbf{x}, \theta) - \mathcal{F}(q(\mathbf{u}), \theta)$$

F = negative of the integral term

minimising KL(approx post$\mid\mid$exact post) $\equiv$ maximising $\mathcal{F}$

as $\log p(\mathbf{y} \mid \mathbf{x}, \theta)$ does not depend on q

$\mathcal{F}$: variational lower bound or negative variational free-energy

$$\mathcal{F}(q(\mathbf{u}), \theta) = \int_f q(f \mid \theta) \log \frac{p(\mathbf{u} \mid \theta)}{q(\mathbf{u})} + \int_f q(f \mid \theta) \log p(\mathbf{y} \mid f, \mathbf{x}, \theta)$$

conditional term cancelled out, split the log term

$$= -\mathrm{KL}(q(\mathbf{u}) \parallel p(\mathbf{u} \mid \theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n) \mid \theta) \log p(y_n \mid f(\mathbf{x}_n), \theta)$$

since $p(\mathbf{y} \mid f, \mathbf{x}, \theta) = \prod_{n=1}^{N} p(y_n \mid f(\mathbf{x}_n), \theta)$ as in GPR and GPC

Computational complexity

Computational complexity

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\operatorname{KL}(q(\mathbf{u}) \parallel p(\mathbf{u} \mid \theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n) \mid \theta) \log p(y_n \mid f(\mathbf{x}_n), \theta)$$

Computational complexity

KL term 1

Expected log likelihood term 2

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

Computational complexity

KL term **1** Expected log likelihood term **2**

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

For general likelihood

2 can be estimated by Monte Carlo with reparameterisation trick or quadrature $\int_{f_n} \approx \sum_{f_{n,k}}$, where $f_{n,k} \sim q(f_n)$

Cost: $\mathcal{O}(M^3)$ for **1** $\mathcal{O}(M^2)$ per data point for **2** Total cost: $\mathcal{O}(NM^2 + M^3)$

Computational complexity

KL term **1** Expected log likelihood term **2**

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

For general likelihood

2 can be estimated by Monte Carlo with reparameterisation trick or quadrature $\int_{f_n} \approx \sum_{f_{n,k}}$, where $f_{n,k} \sim q(f_n)$

Cost: $\mathcal{O}(M^3)$ for **1** $\mathcal{O}(M^2)$ per data point for **2** Total cost: $\mathcal{O}(NM^2)$ if $M \ll N$

Computational complexity

KL term **1** Expected log likelihood term **2**

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

For general likelihood

2 can be estimated by Monte Carlo with reparameterisation trick or quadrature $\int_{f_n} \approx \sum_{f_{n,k}}$, where $f_{n,k} \sim q(f_n)$

Cost: $\mathcal{O}(M^3)$ for **1** $\mathcal{O}(M^2)$ per data point for **2** Total cost: $\mathcal{O}(NM^2)$ if $M \ll N$

Minibatch stochastic approximation $\sum_{n=1}^{N} \approx \frac{N}{B} \sum_{b=1}^{B}$ with batch size $B \ll N$ Total cost: $\mathcal{O}(BM^2 + M^3)$

Hensman et al 2013

Computational complexity

KL term **1** Expected log likelihood term **2**

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

For general likelihood

2 can be estimated by Monte Carlo with reparameterisation trick or quadrature $\int_{f_n} \approx \sum_{f_{n,k}}$, where $f_{n,k} \sim q(f_n)$

Cost: $\mathcal{O}(M^3)$ for **1** $\mathcal{O}(M^2)$ per data point for **2** Total cost: $\mathcal{O}(NM^2)$ if $M \ll N$

Minibatch stochastic approximation $\sum_{n=1}^{N} \approx \frac{N}{B} \sum_{b=1}^{B}$ with batch size $B \ll N$ Total cost: $\mathcal{O}(M^3)$ if $B \approx M$

Hensman et al 2013

Computational complexity

KL term 1 Expected log likelihood term 2

$$\mathcal{F}(q(\mathbf{u}), \theta) = -\,\mathrm{KL}(q(\mathbf{u}) \,\|\, p(\mathbf{u}\,|\,\theta)) + \sum_{n=1}^{N} \int_{f_n} q(f(\mathbf{x}_n)\,|\,\theta)\ \log p(y_n\,|\,f(\mathbf{x}_n), \theta)$$

For general likelihood

2 can be estimated by Monte Carlo with reparameterisation trick or quadrature $\int_{f_n} \approx \sum_{f_{n,k}}$, where $f_{n,k} \sim q(f_n)$

Cost: $\mathcal{O}(M^3)$ for 1 $\mathcal{O}(M^2)$ per data point for 2 Total cost: $\mathcal{O}(NM^2)$ if $M \ll N$

Minibatch stochastic approximation

$$\sum_{n=1}^{N} \approx \frac{N}{B} \sum_{b=1}^{B}$$ with batch size $B \ll N$ Total cost: $\mathcal{O}(M^3)$ if $B \approx M$

Hensman et al 2013

Collapsed bound for regression with Gaussian likelihood

Titsias 2009

optimal $q(\mathbf{u}) \propto p(\mathbf{u}\,|\,\theta)\mathrm{N}(\mathbf{y}; \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{u}, \sigma_y^2\mathrm{I})$

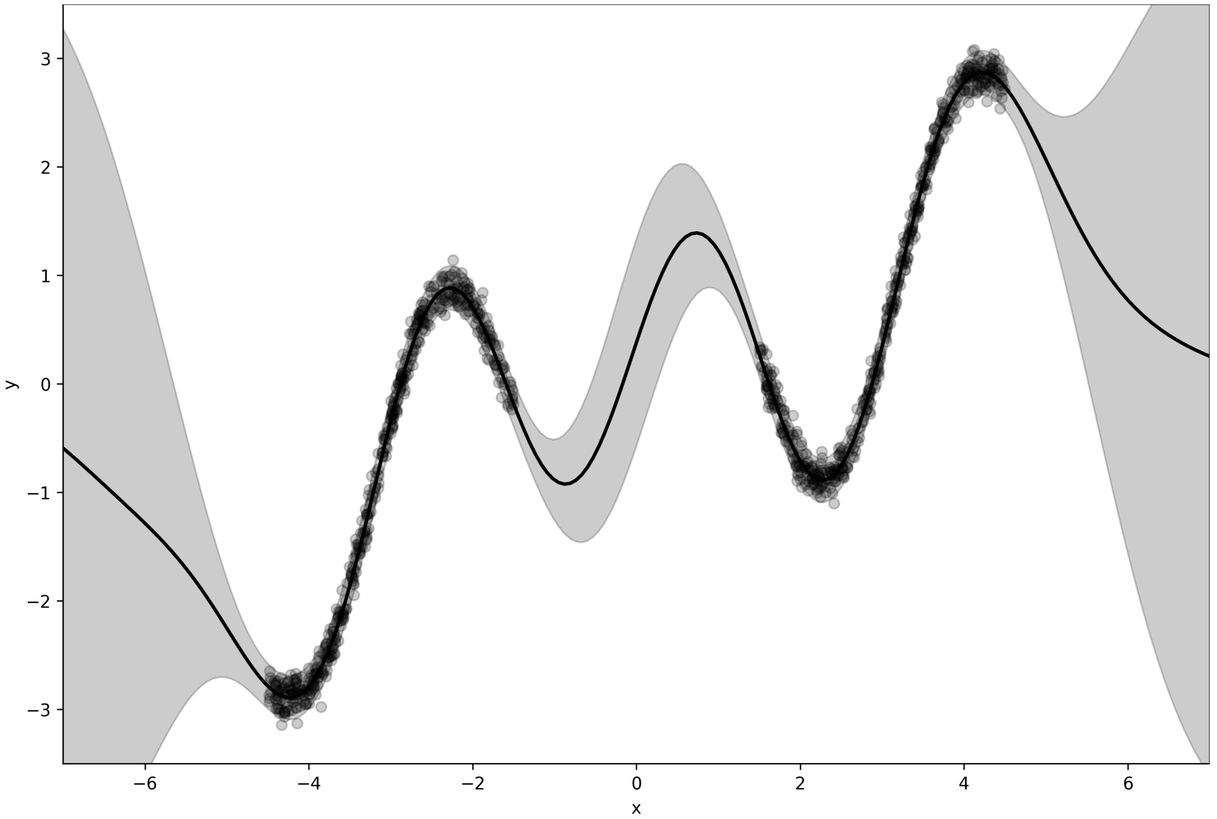
$$\mathcal{F}(\theta) = \log \mathrm{N}(\mathbf{y}; \mathbf{0}, \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{K_{uf}} + \sigma_y^2\mathrm{I}) - \frac{1}{2\sigma_y^2}\mathrm{trace}(\mathbf{K_{ff}} - \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\mathbf{K_{uf}})$$

Total cost: $\mathcal{O}(NM^2)$

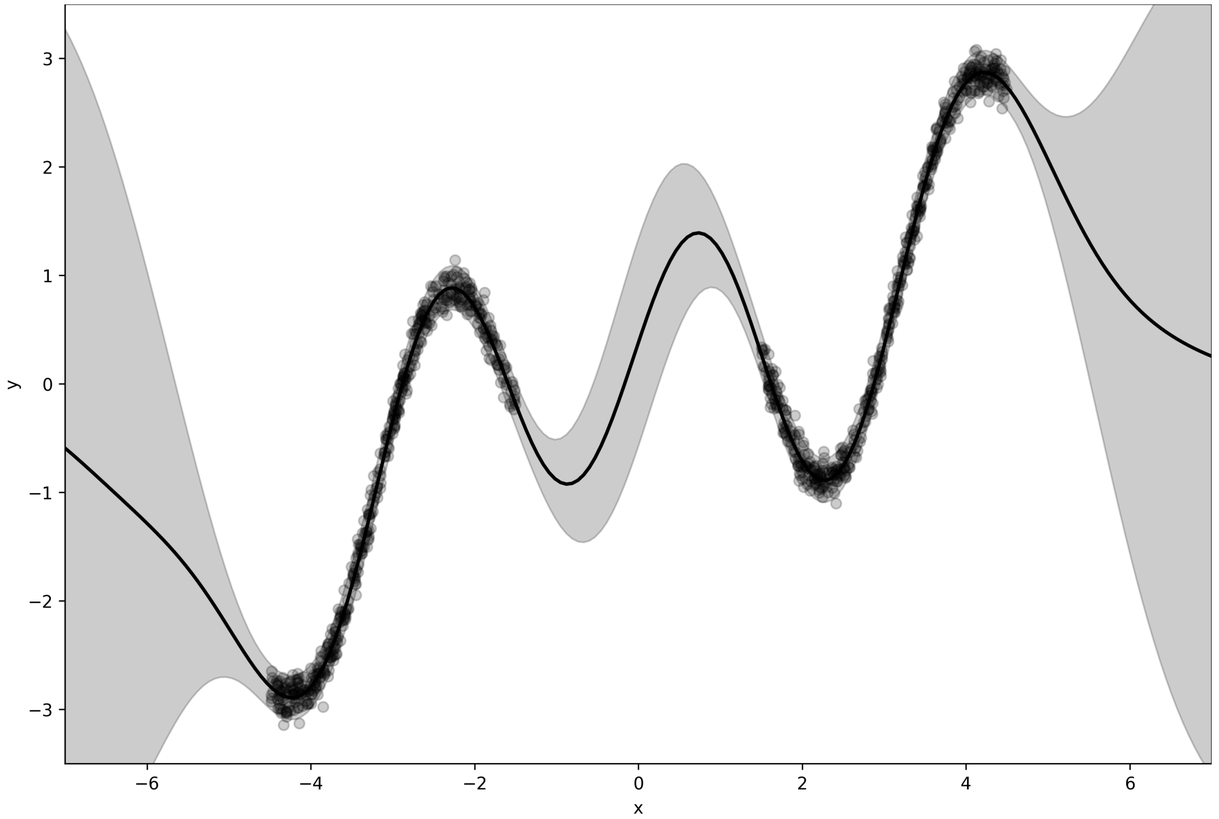
Titsias 2009, Variational Learning of Inducing Variables in Sparse Gaussian Processes, AISTATS
Hensman et al 2013, Gaussian processes for big data, UAI

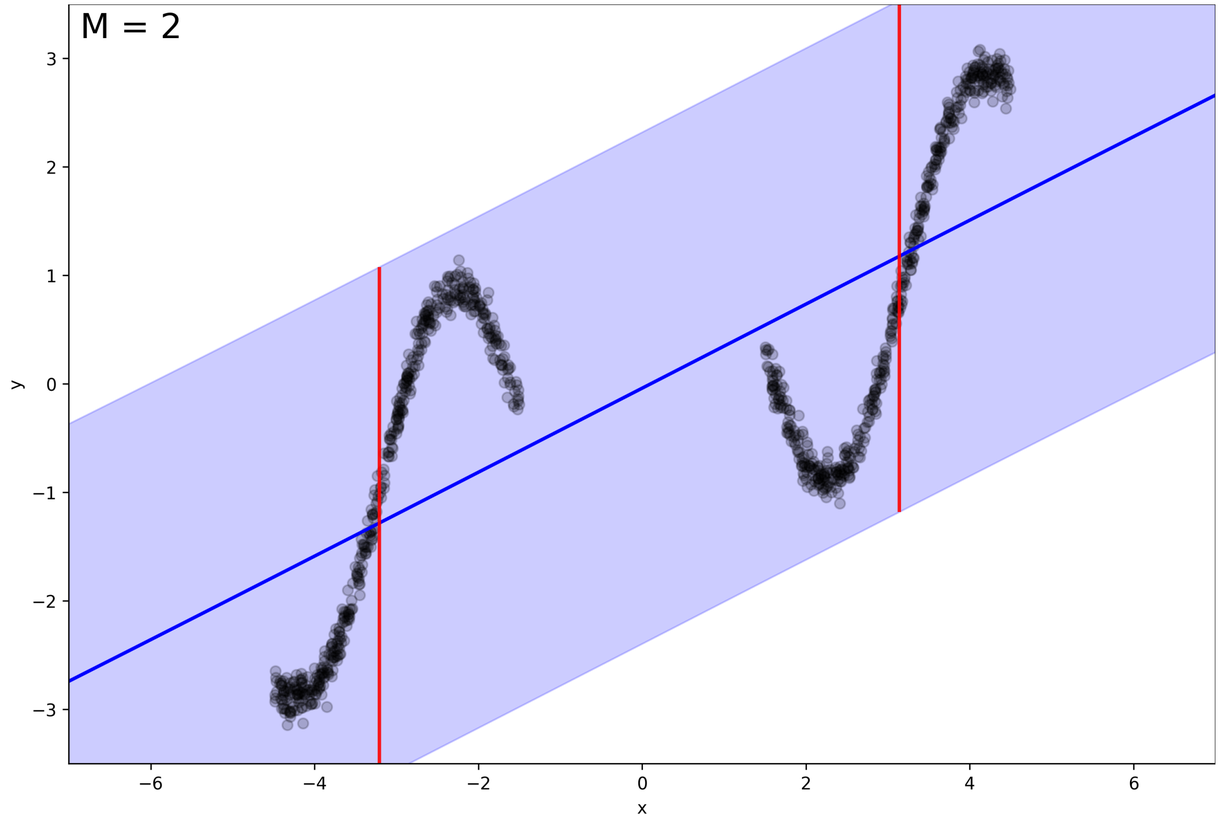
How many pseudo-data points is enough?

How many pseudo-data points is enough?

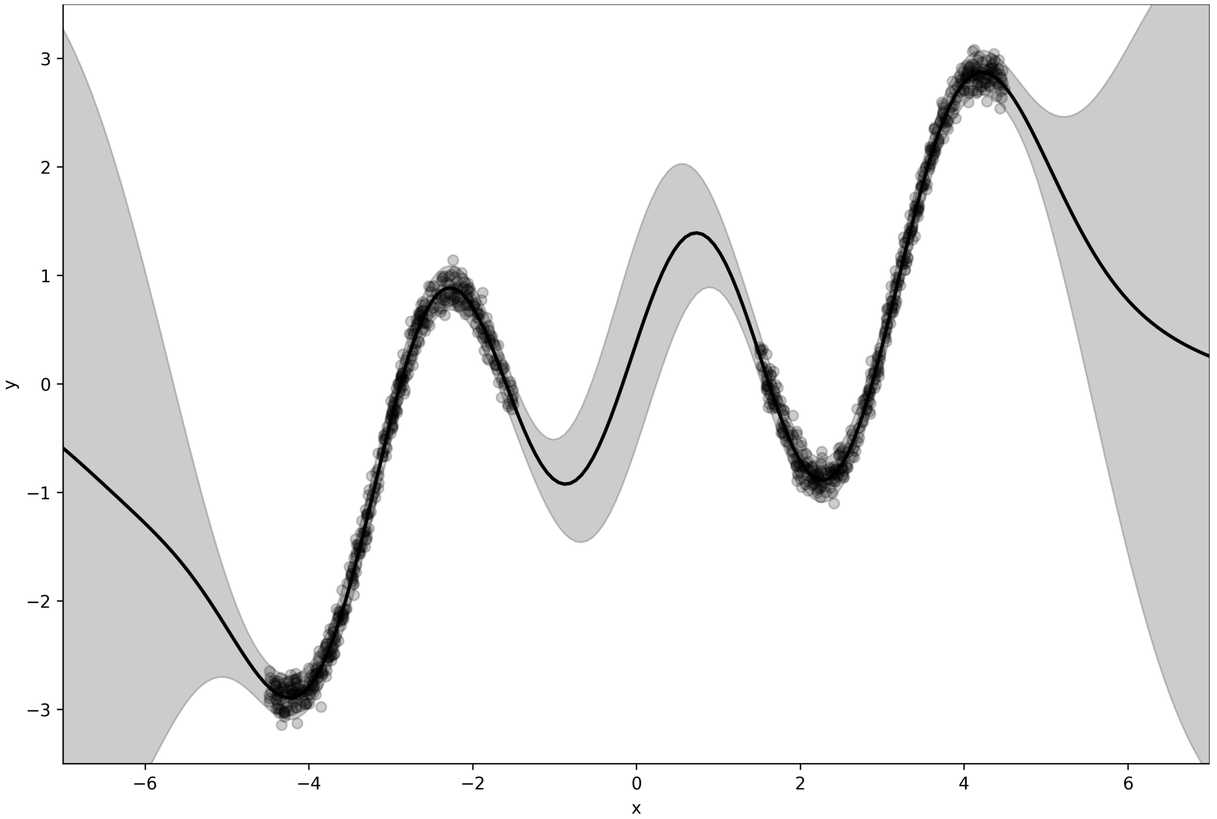


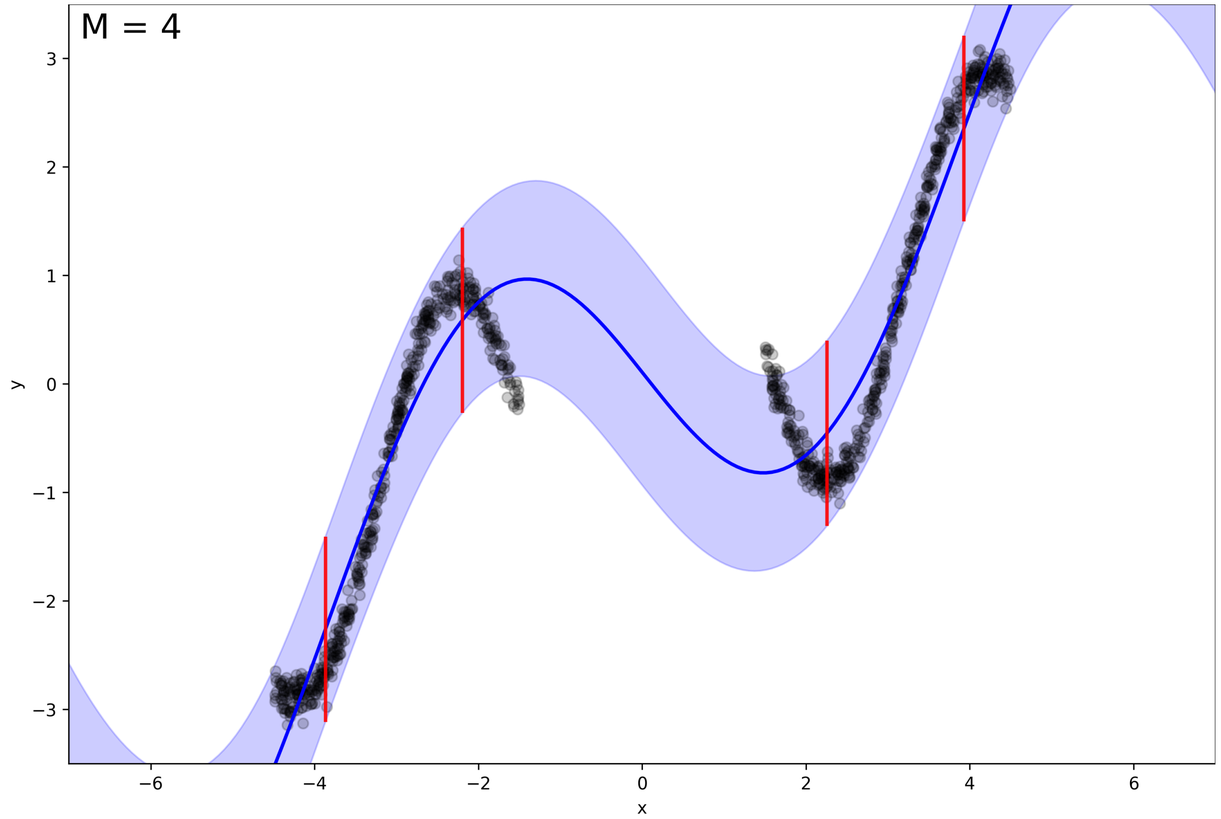
How many pseudo-data points is enough?



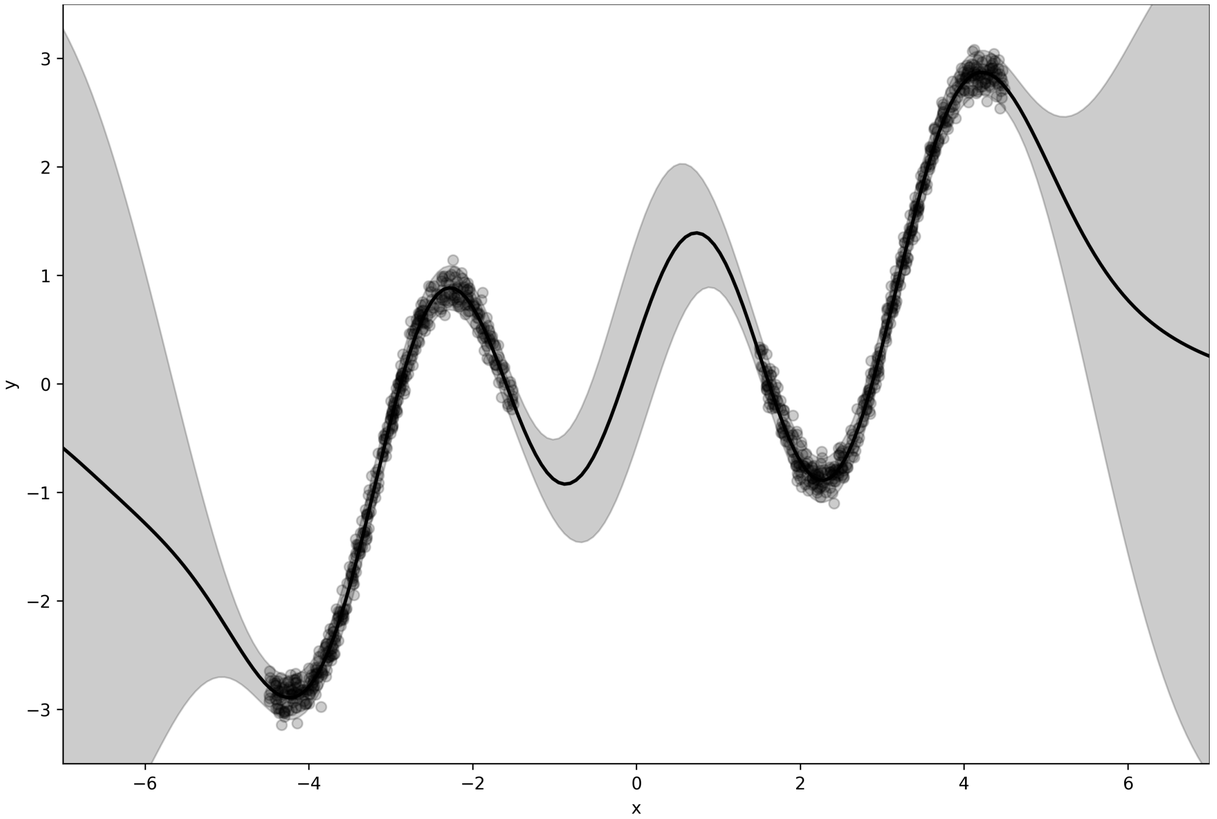


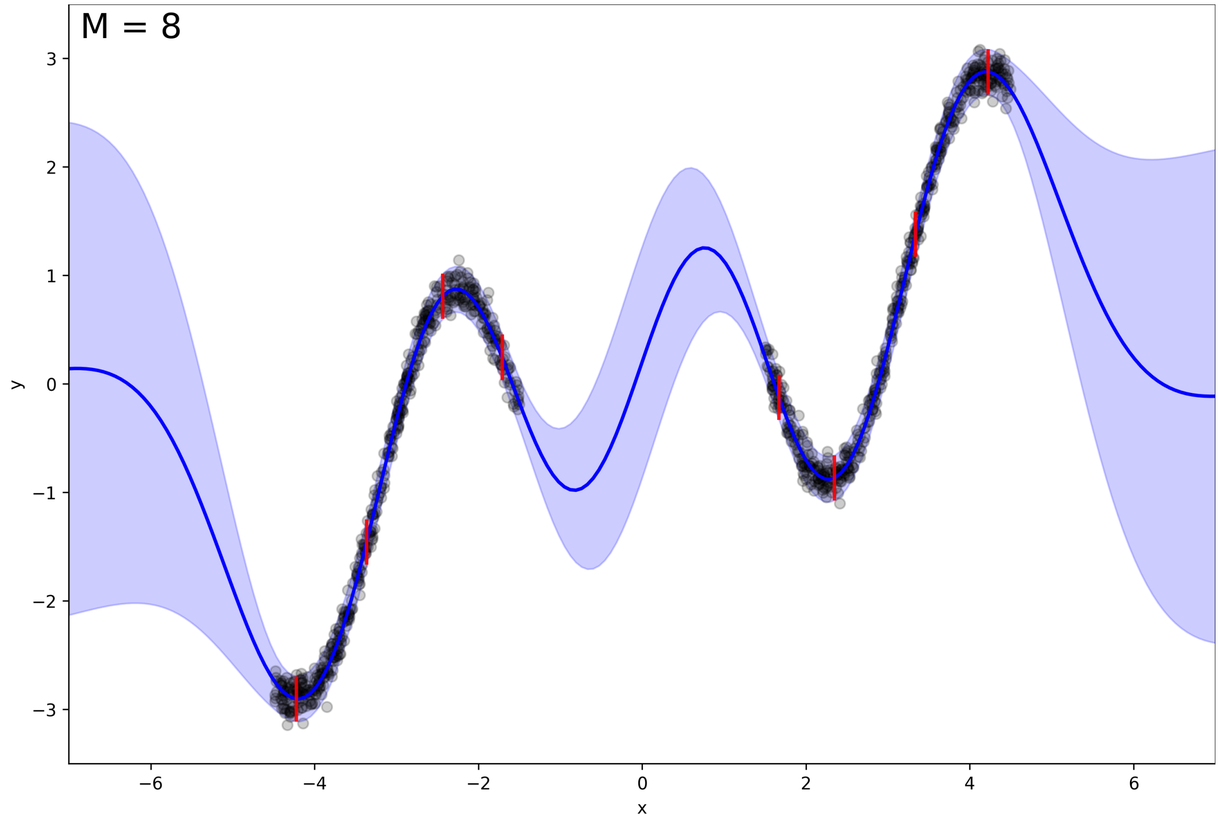
How many pseudo-data points is enough?



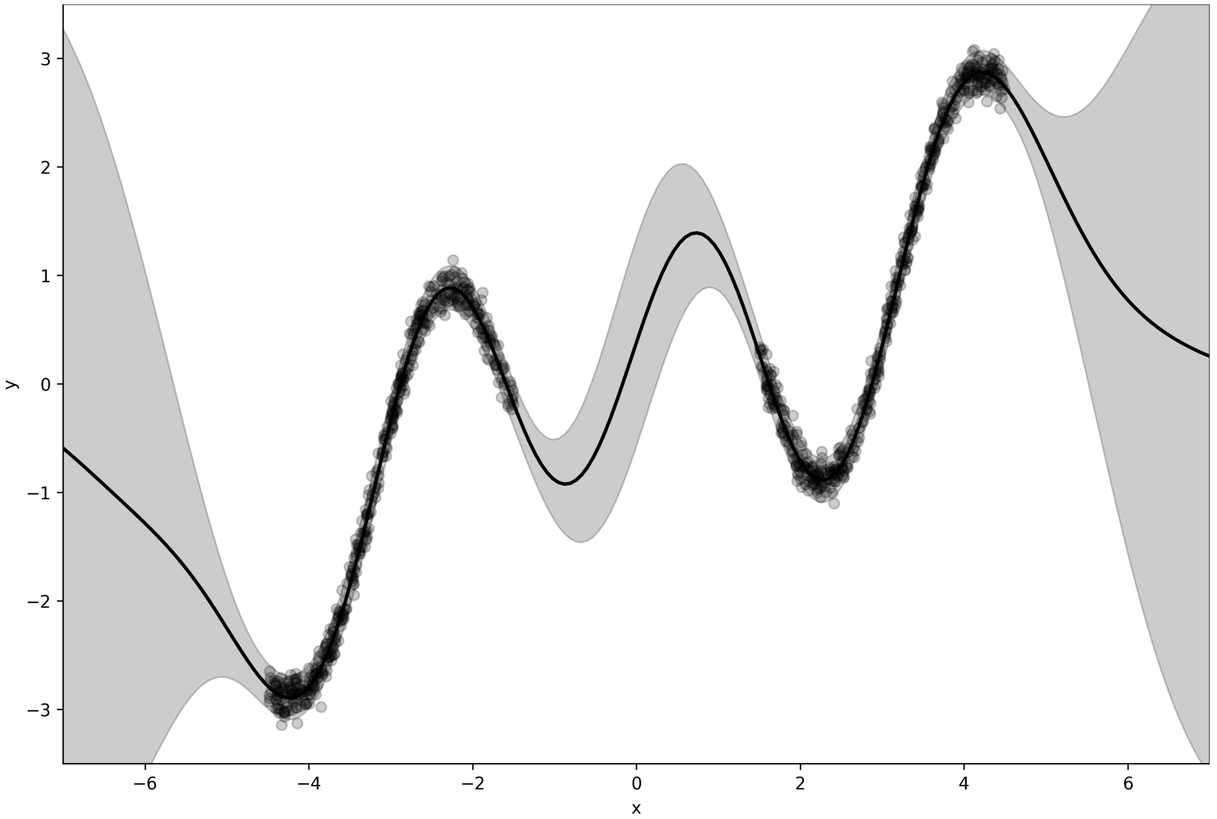


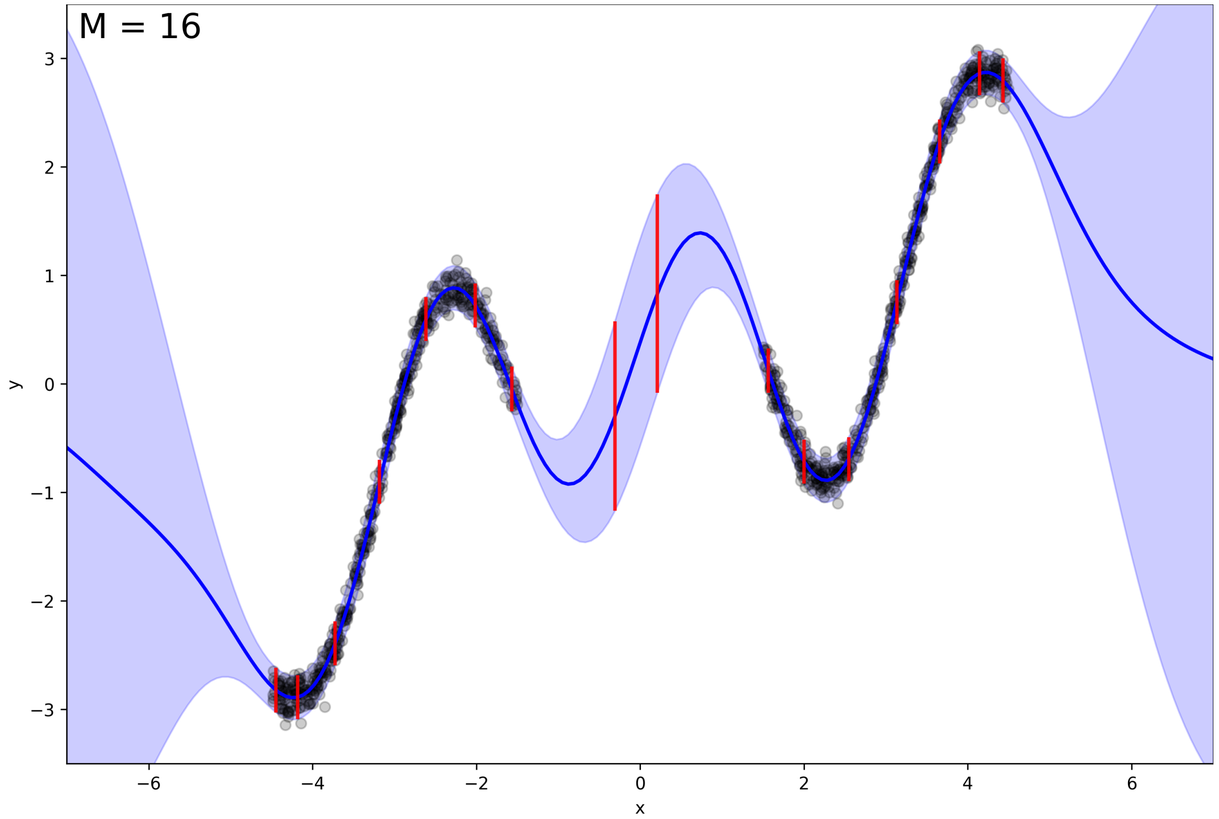
How many pseudo-data points is enough?



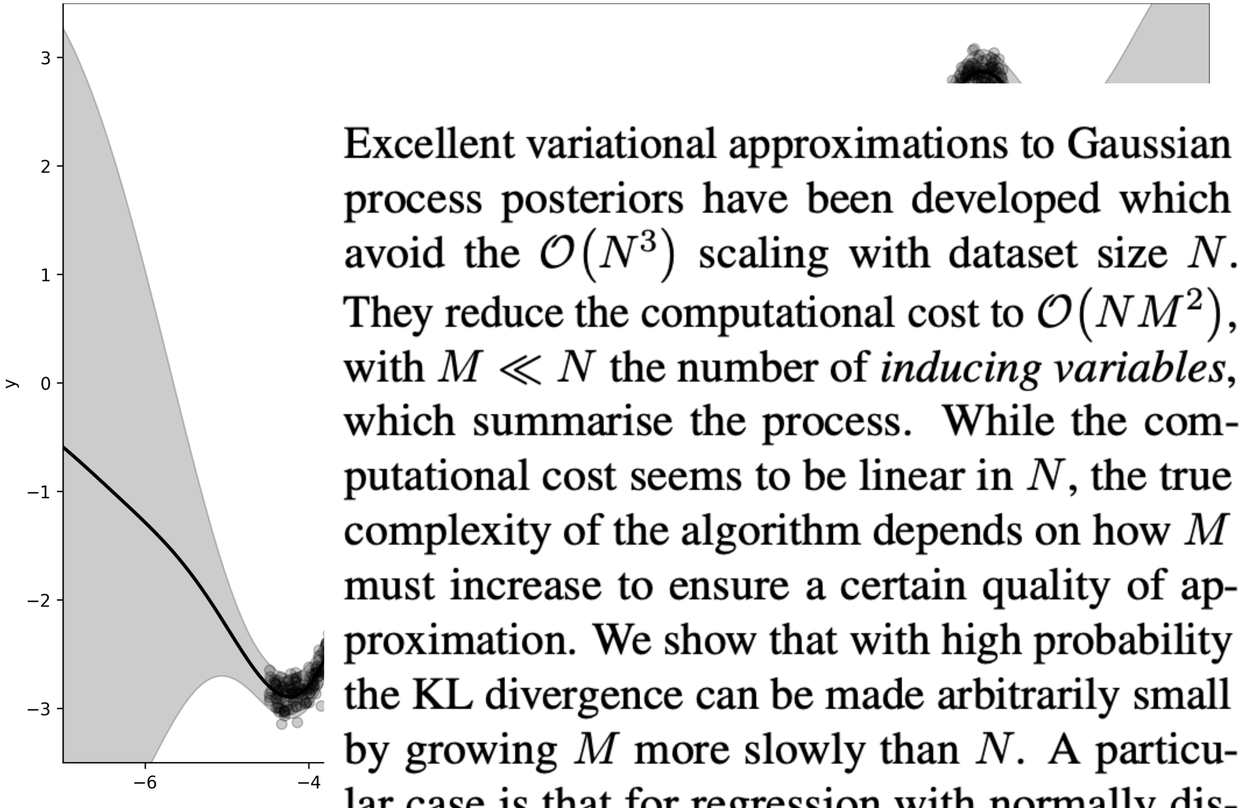


How many pseudo-data points is enough?

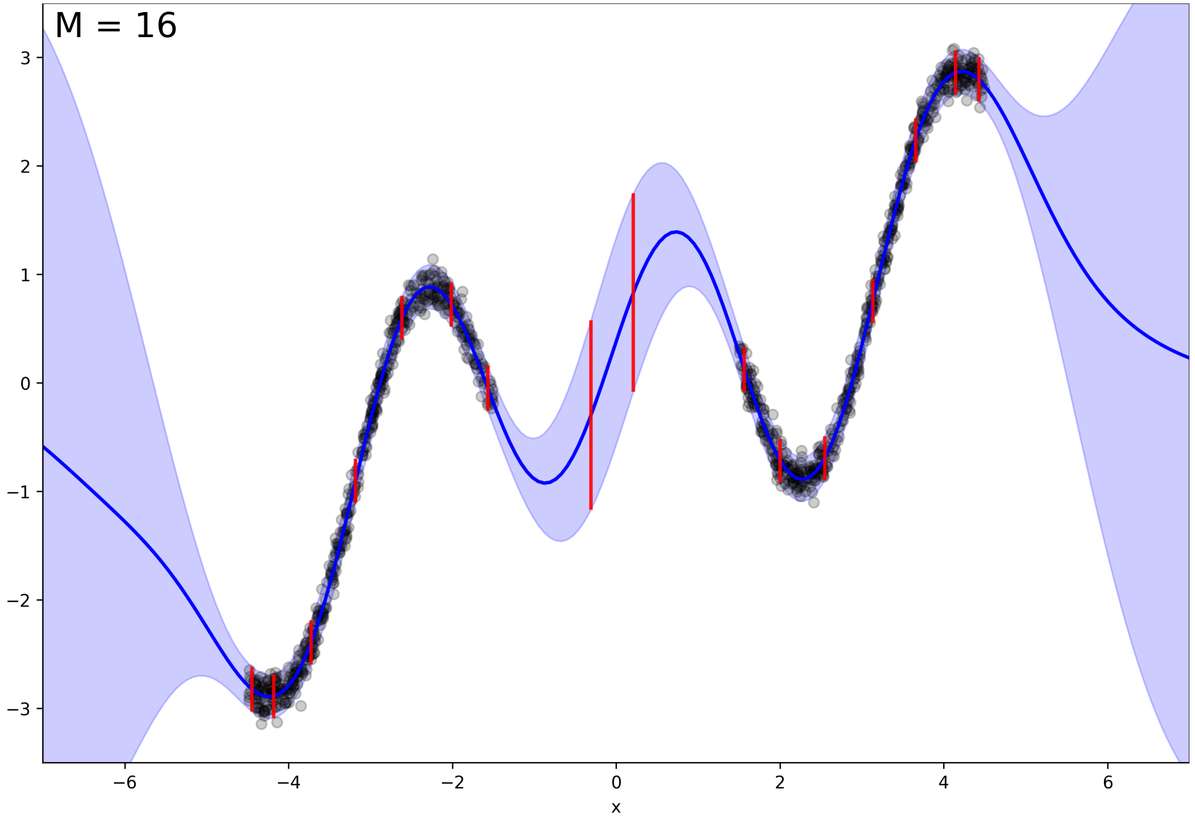




How many pseudo-data points is enough?



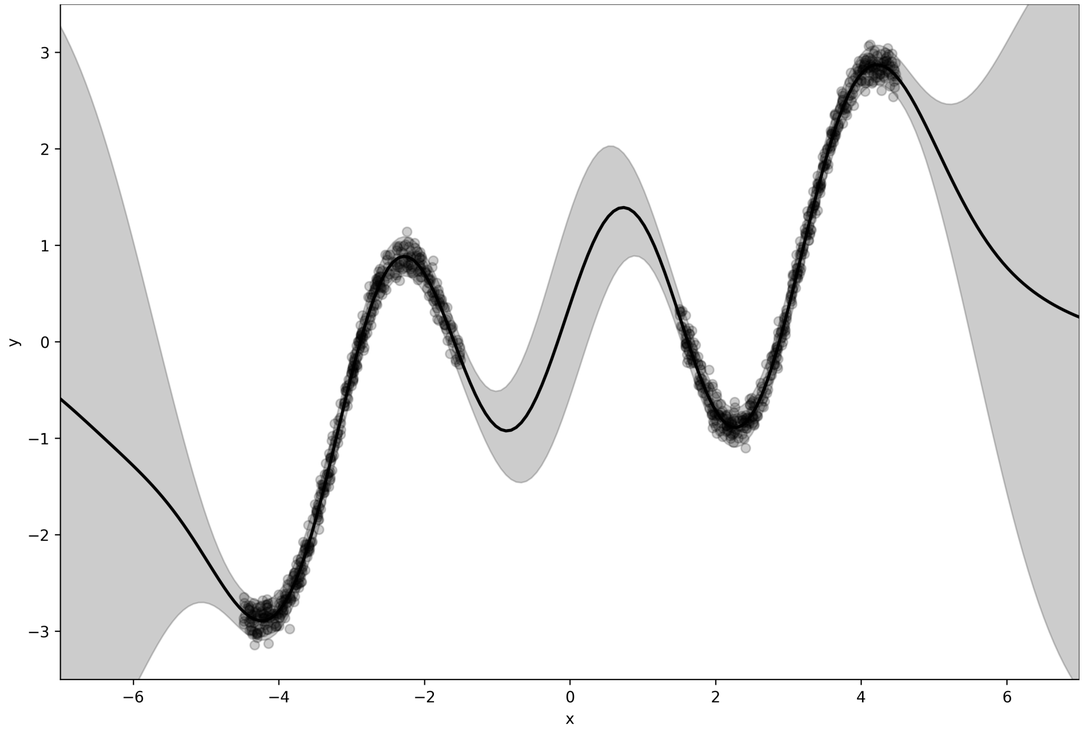
Excellent variational approximations to Gaussian process posteriors have been developed which avoid the $\mathcal{O}(N^3)$ scaling with dataset size N. They reduce the computational cost to $\mathcal{O}(NM^2)$, with $M \ll N$ the number of *inducing variables*, which summarise the process. While the computational cost seems to be linear in N, the true complexity of the algorithm depends on how M must increase to ensure a certain quality of approximation. We show that with high probability the KL divergence can be made arbitrarily small by growing M more slowly than N. A particular case is that for regression with normally distributed inputs in D-dimensions with the Squared Exponential kernel, $M = \mathcal{O}(\log^D N)$ suffices. Our results show that as datasets grow, Gaussian process posteriors can be approximated cheaply, and provide a concrete rule for how to increase M in continual learning scenarios.



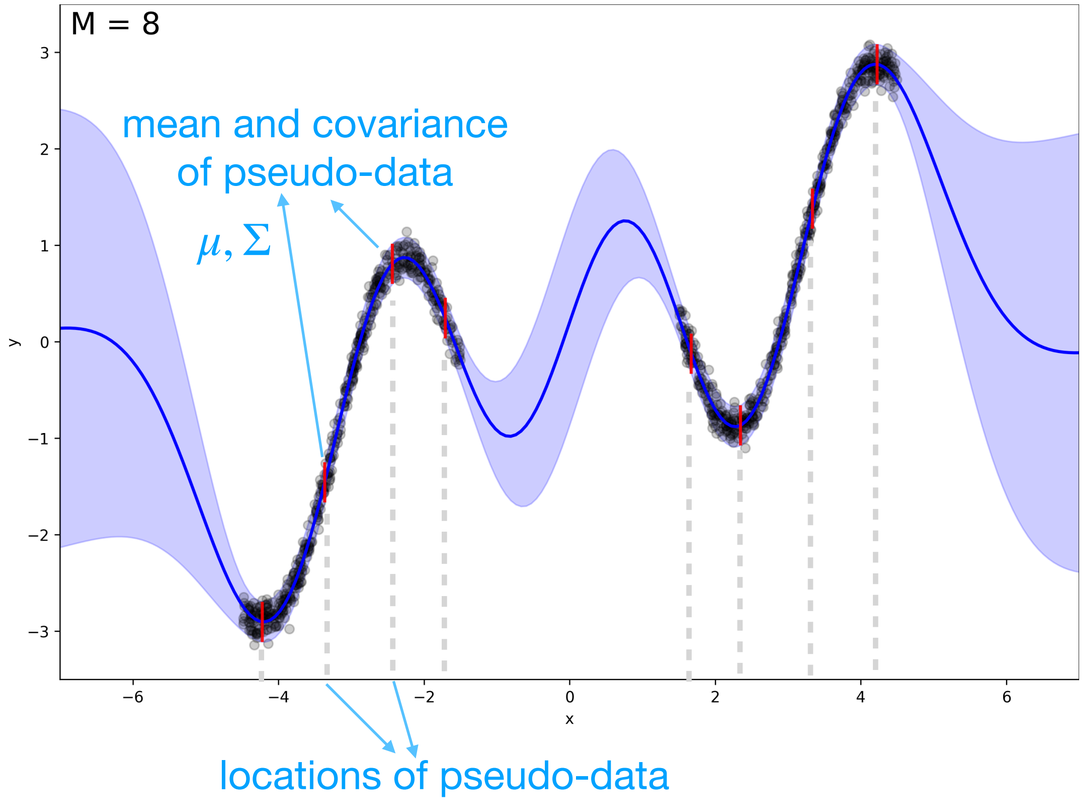
Rates of Convergence for Sparse Variational Gaussian Process Regression
Burt, Rasmussen and van der Wilk
ICML 2019

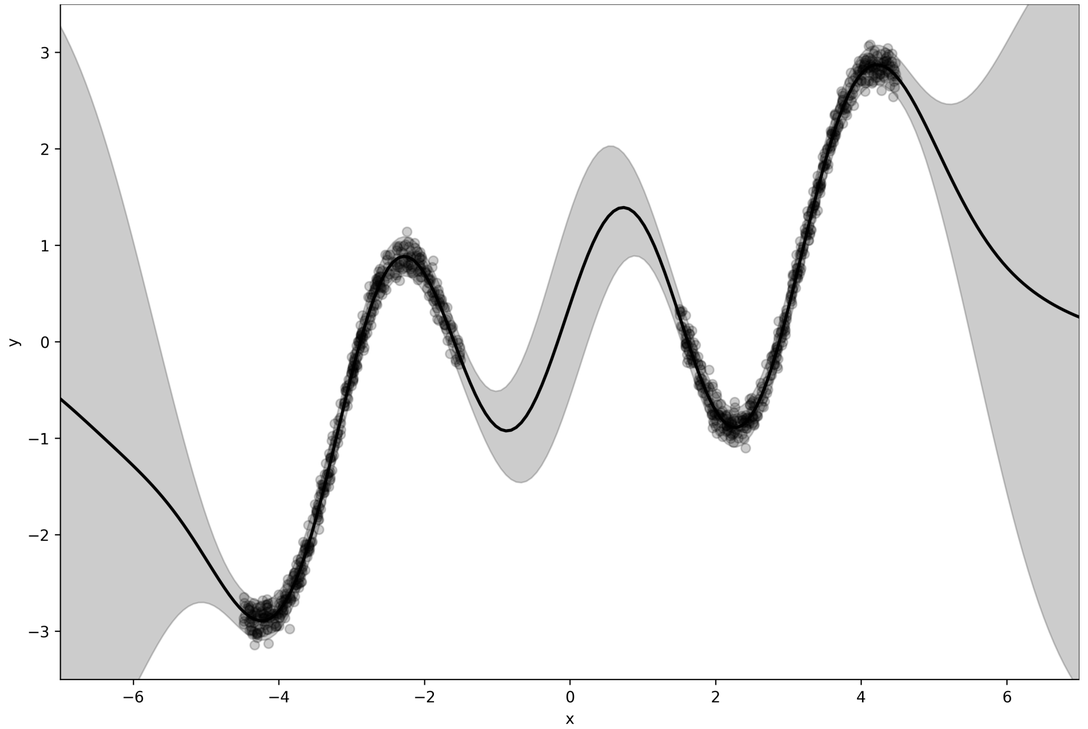
Research frontier 1: Power Expectation Propagation

Research frontier 1: Power Expectation Propagation



Research frontier 1: Power Expectation Propagation

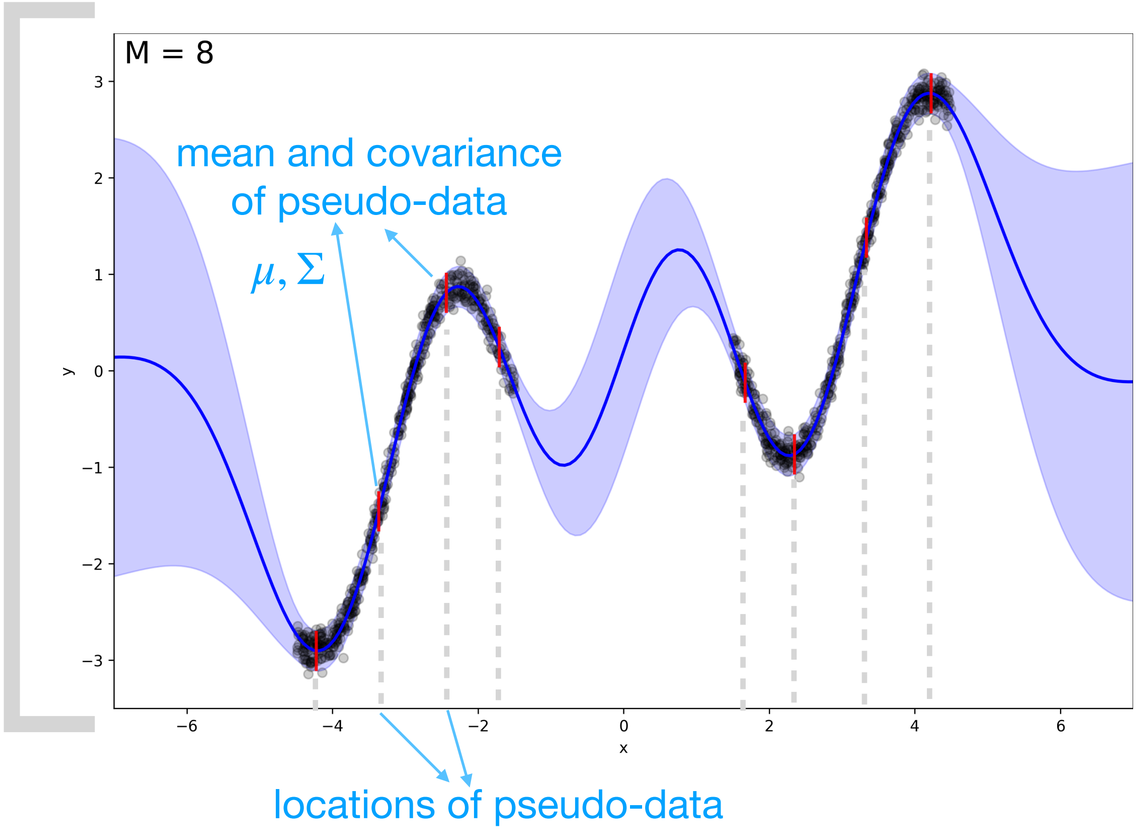


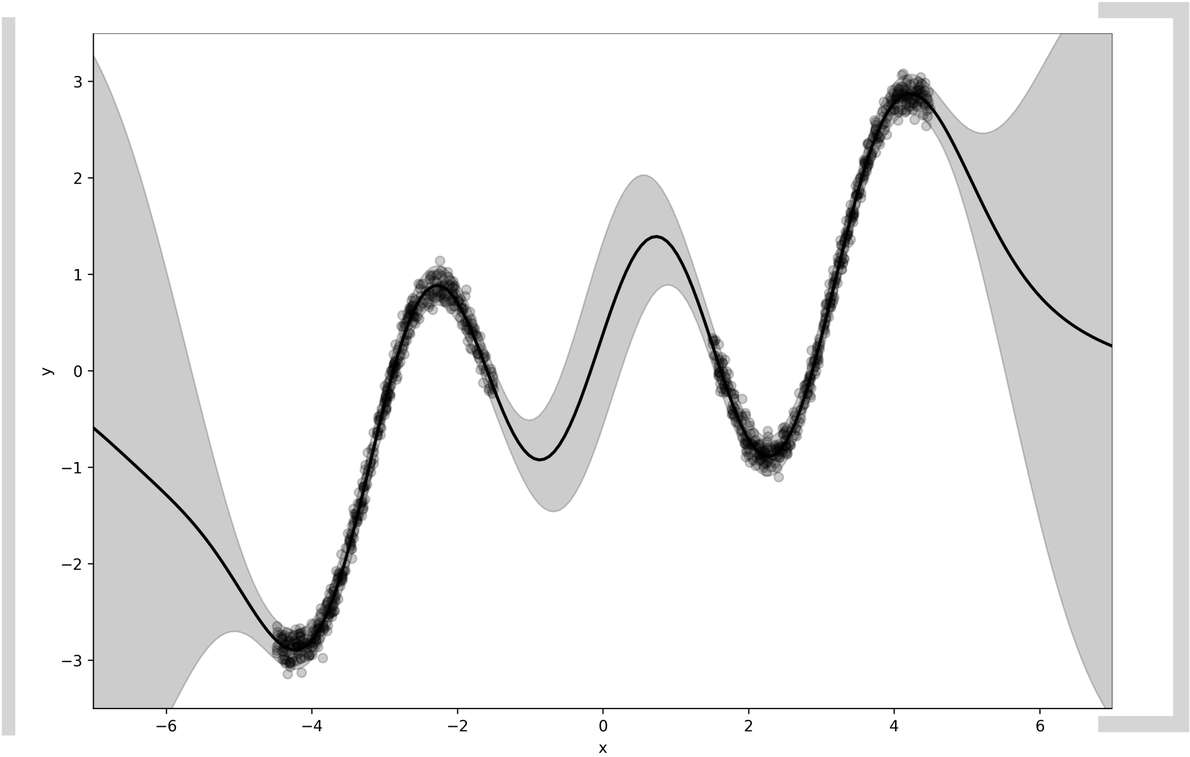


Research frontier 1: Power Expectation Propagation

KL

variational inference

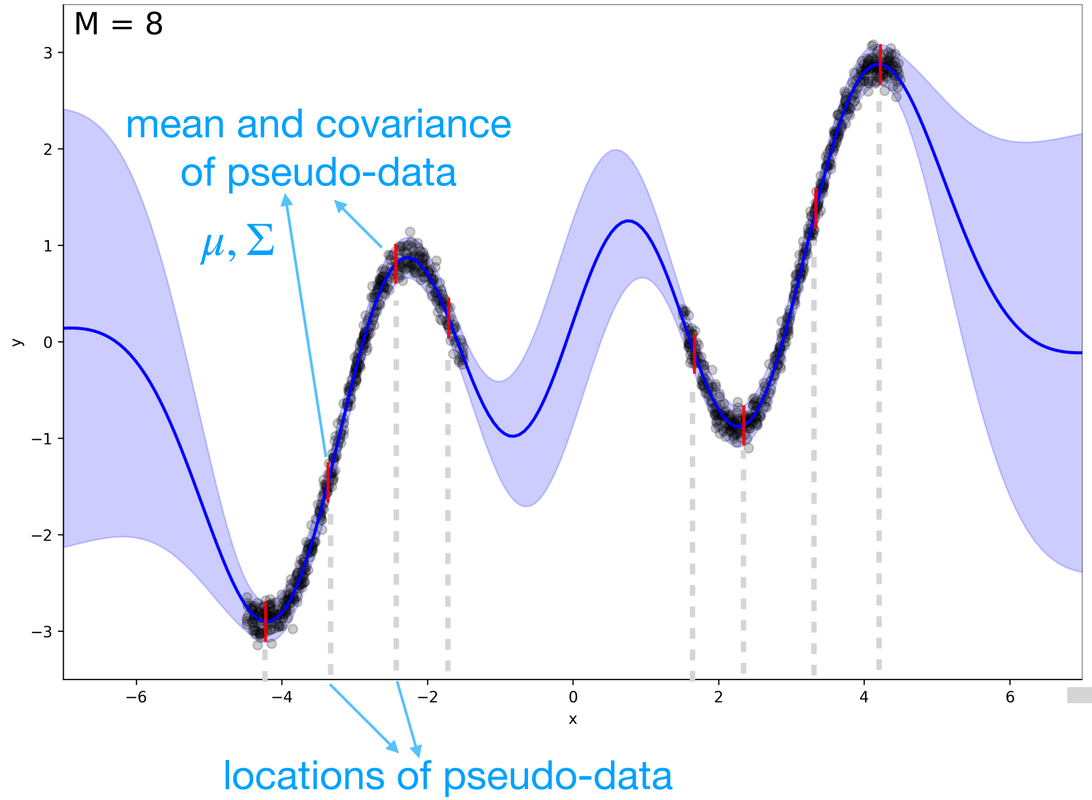




Research frontier 1: Power Expectation Propagation

KL

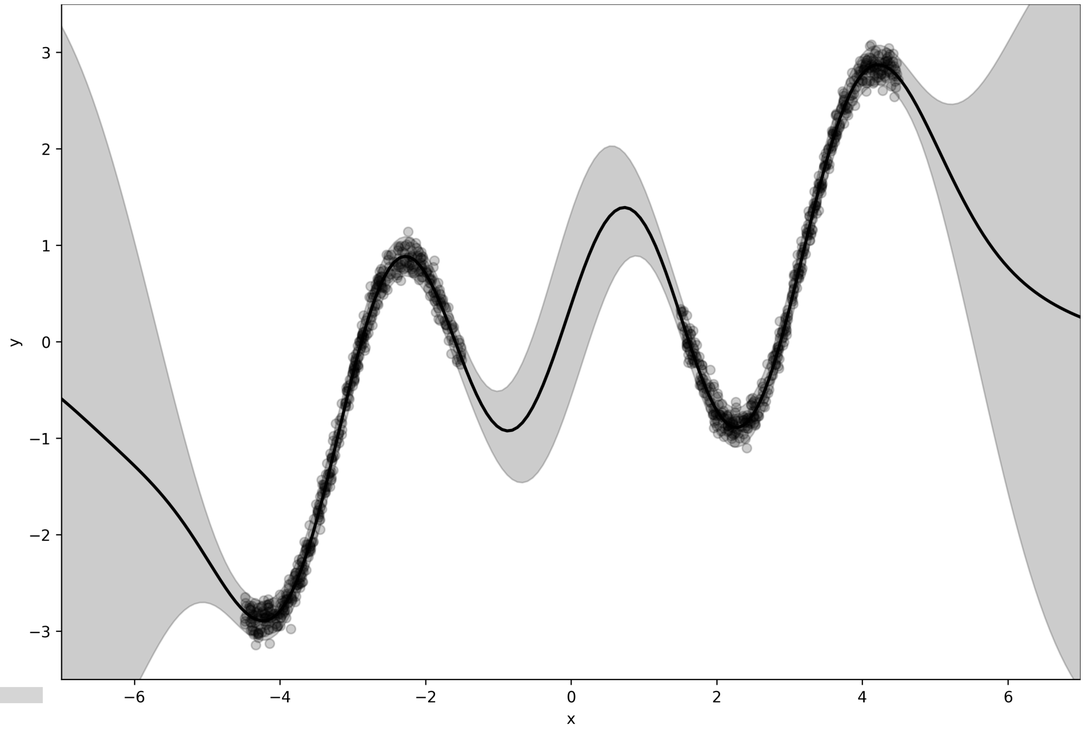
variational inference

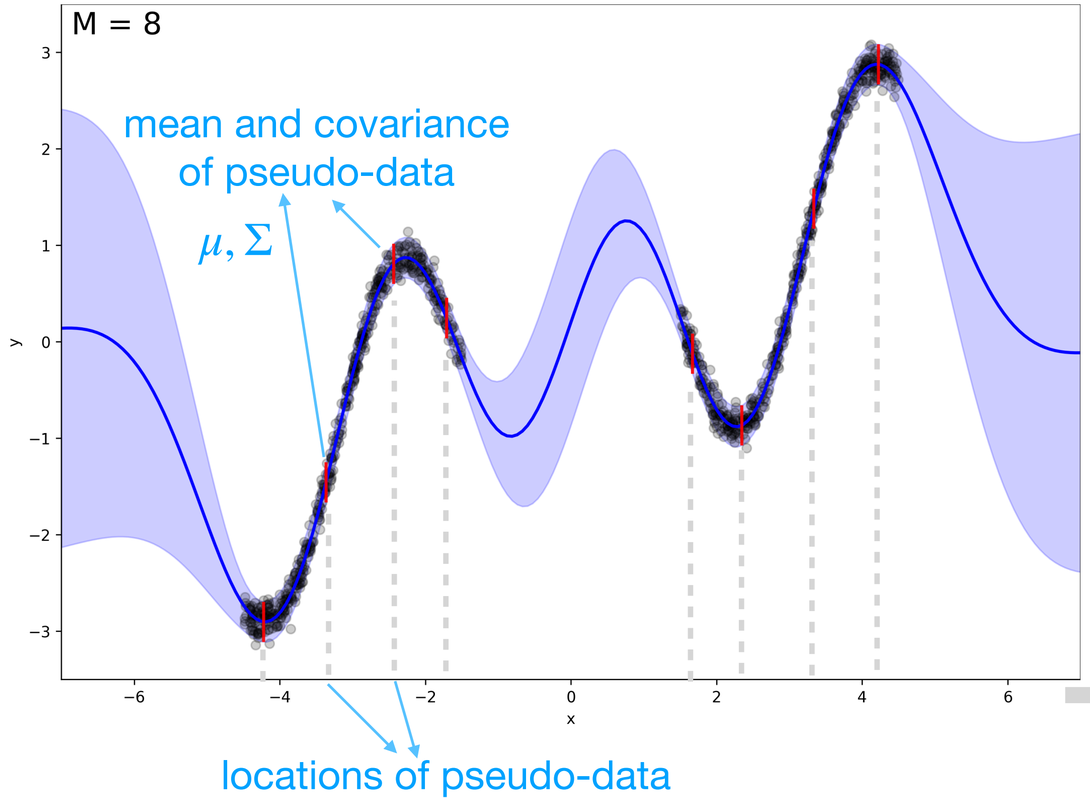


Research frontier 1: Power Expectation Propagation

KL

variational inference

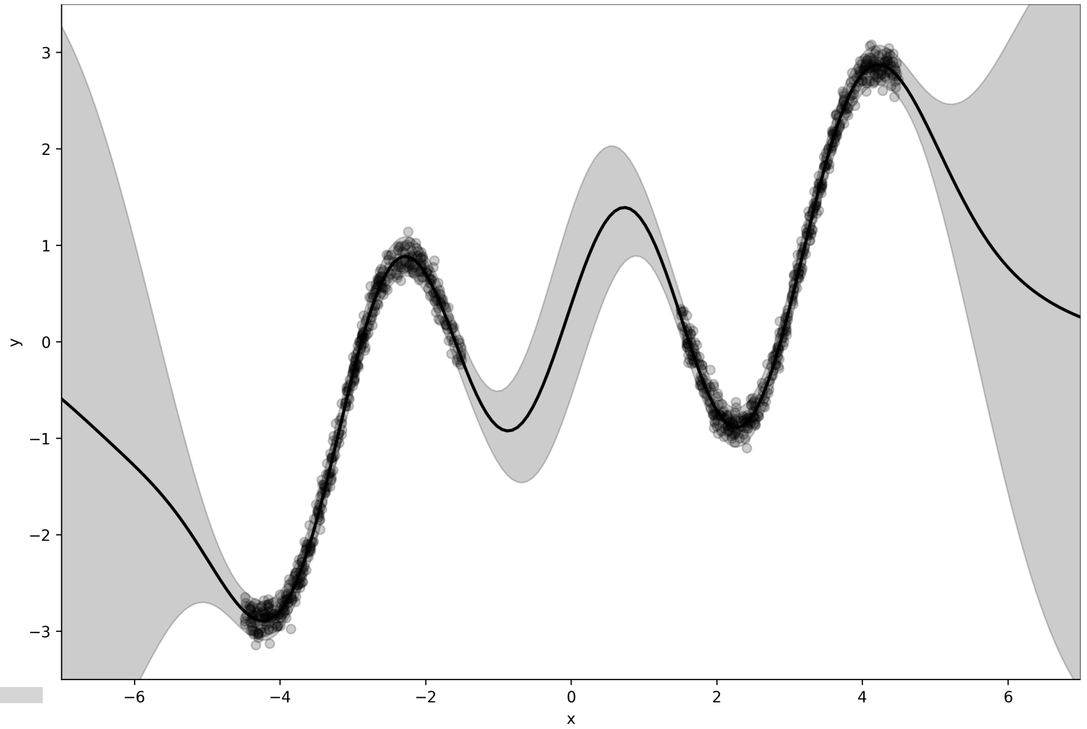


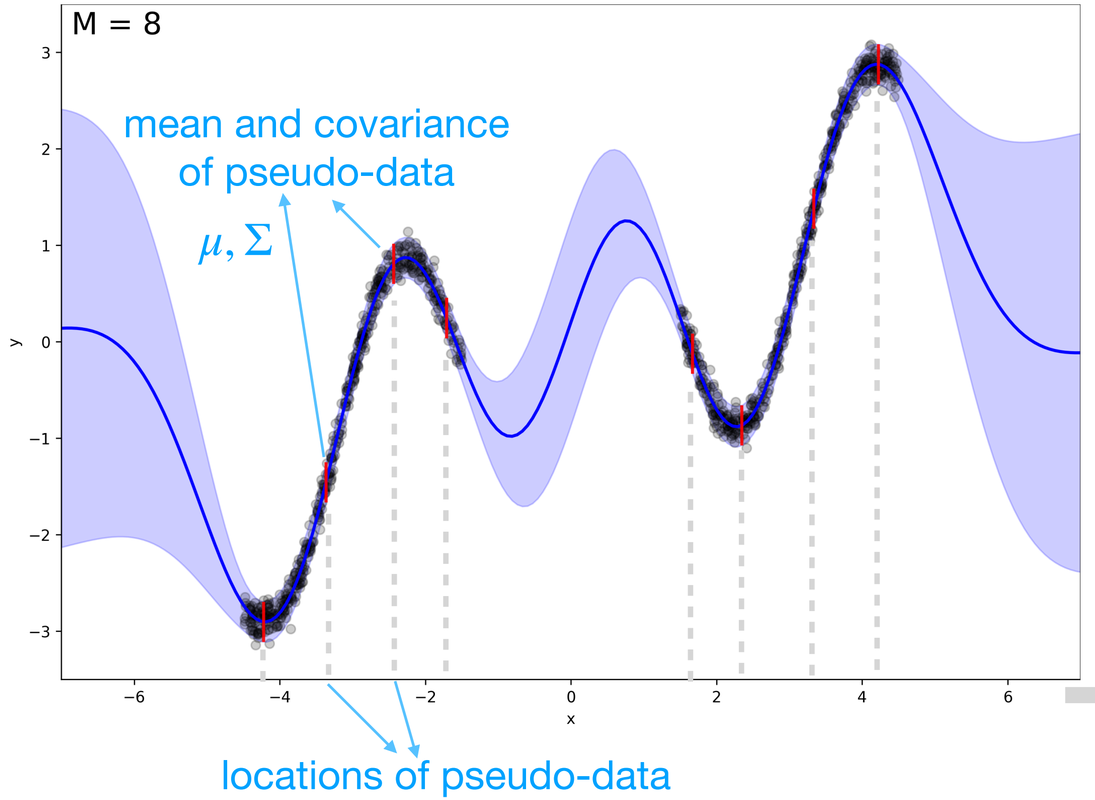


Research frontier 1: Power Expectation Propagation

KL

Expectation Propagation*



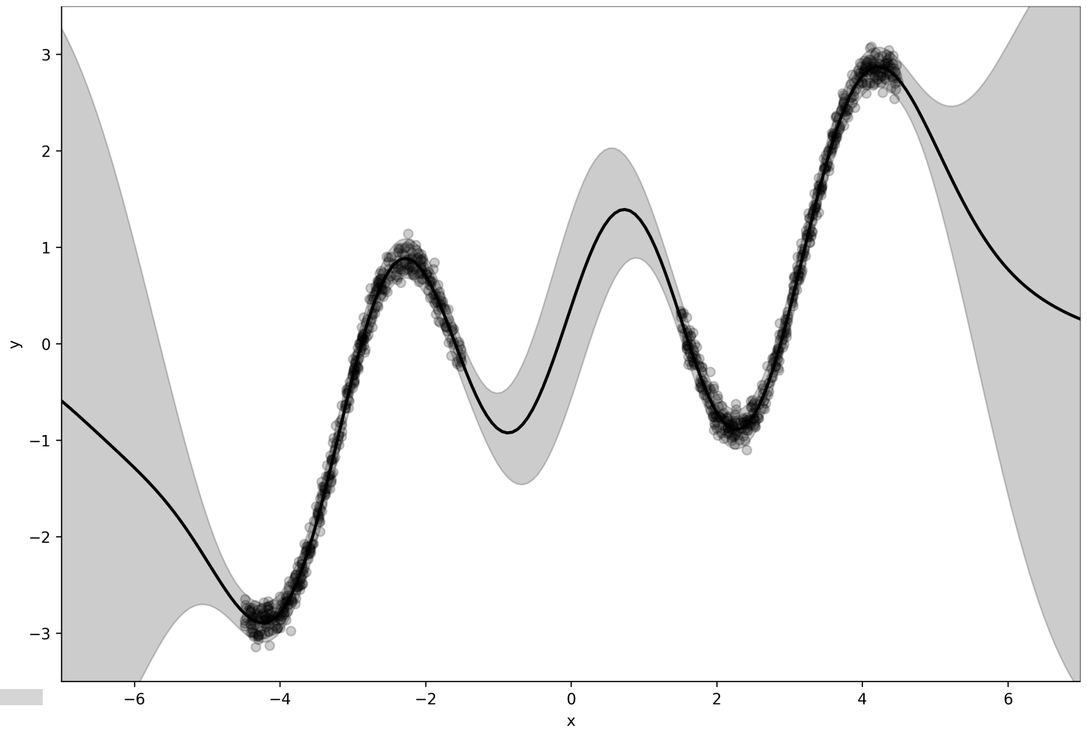


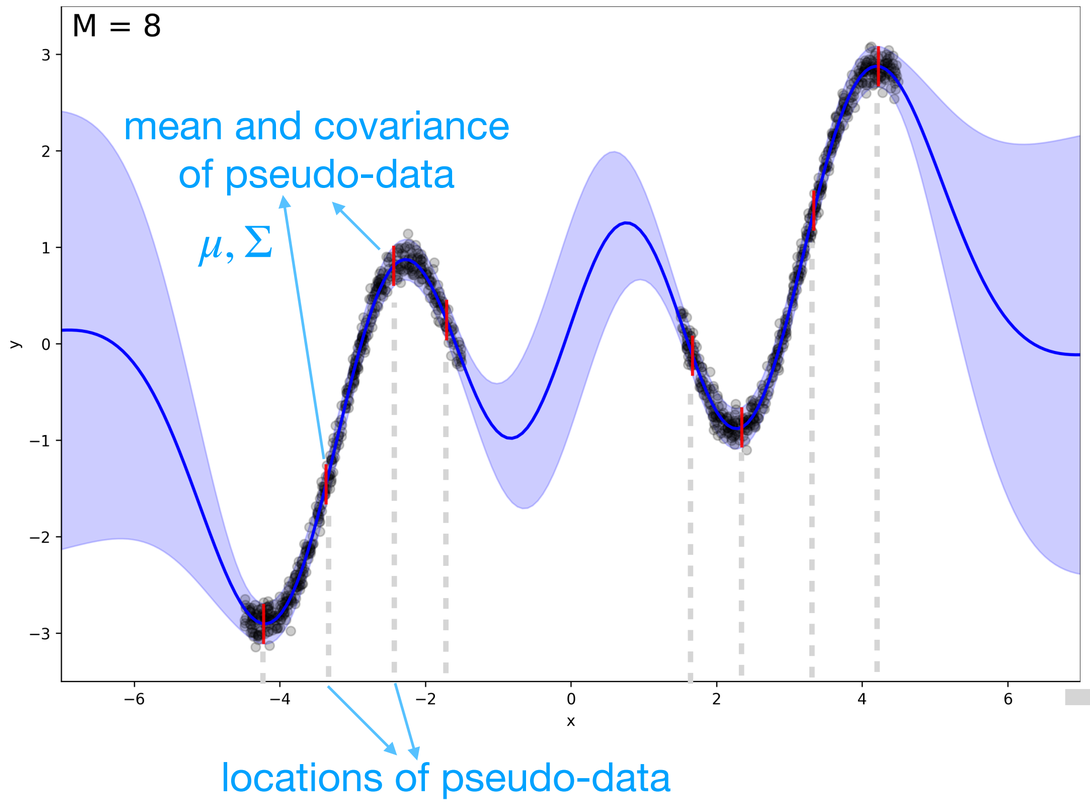
*Local KL or alpha-div, only deal with one likelihood factor at a time

Research frontier 1: Power Expectation Propagation

α-div

Power Expectation Propagation*





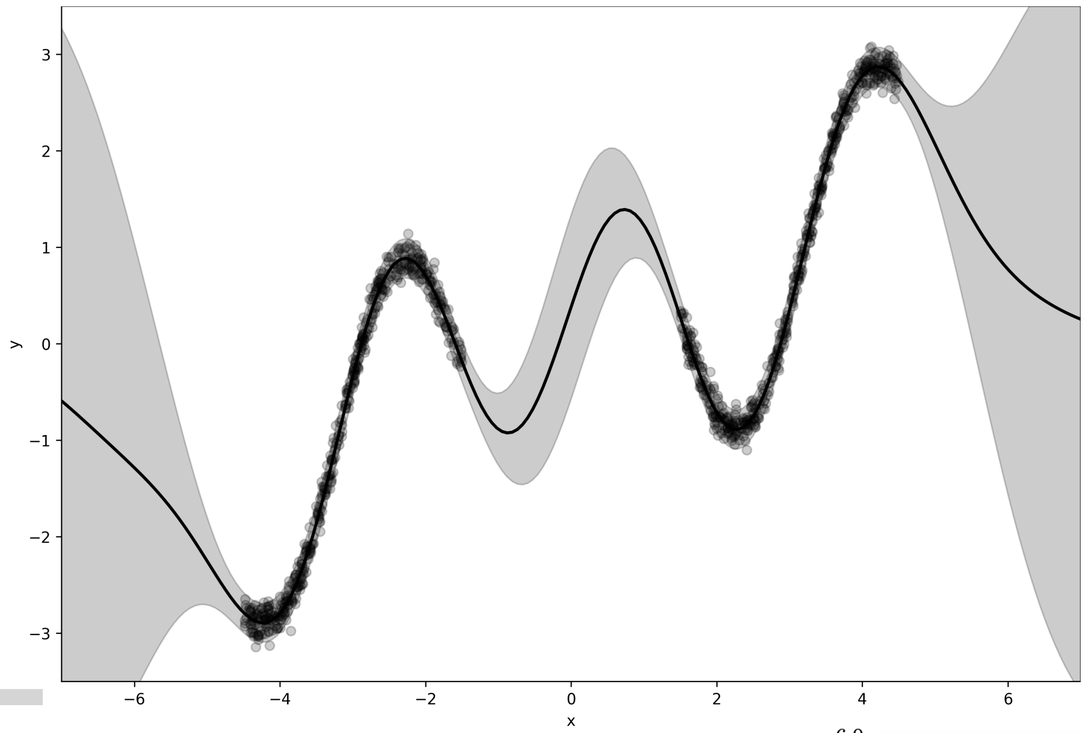
*Local KL or alpha-div, only deal with one likelihood factor at a time

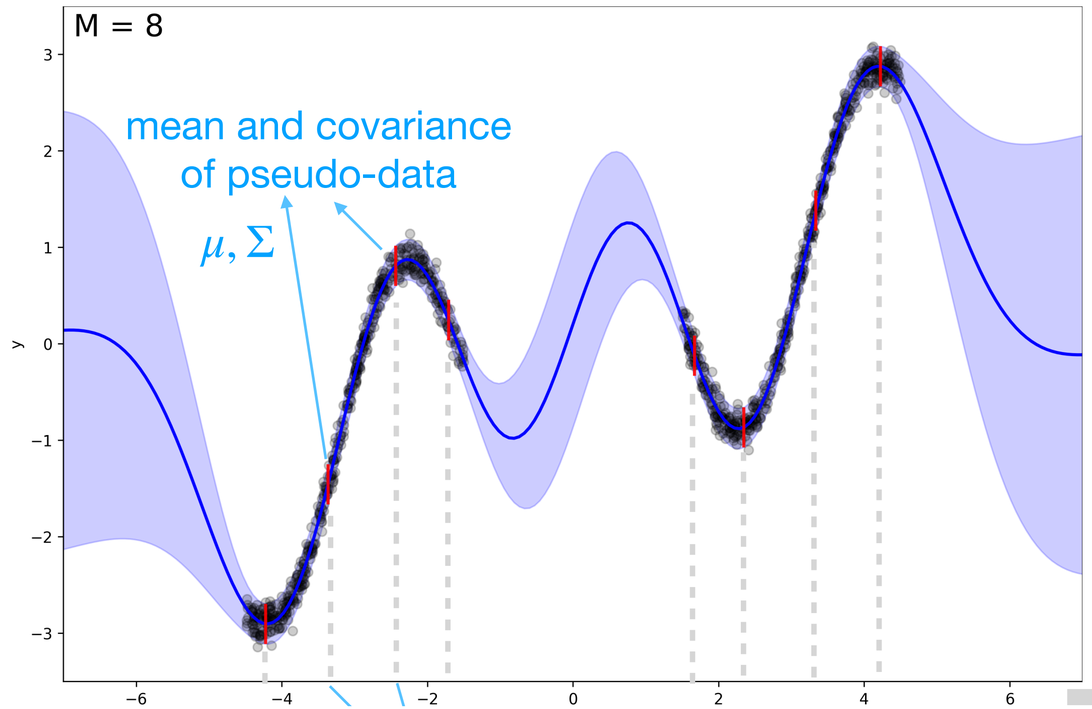
Bui, Yan and Turner, A Unifying Framework for Gaussian Process Pseudo-Point Approximations using Power Expectation Propagation, JMLR 2017

Research frontier 1: Power Expectation Propagation

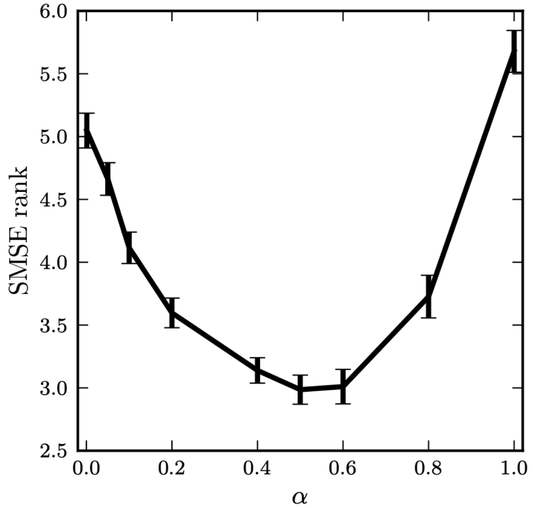
α-div

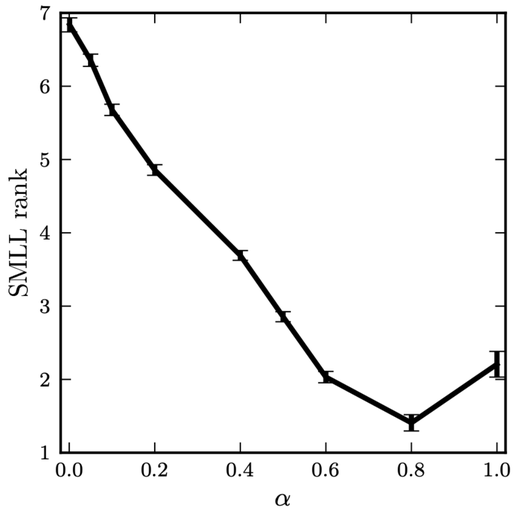
Power Expectation Propagation*





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





alpha = 0.5 performs well on average

*Local KL or alpha-div, only deal with one likelihood factor at a time

Bui, Yan and Turner, A Unifying Framework for Gaussian Process Pseudo-Point Approximations using Power Expectation Propagation, JMLR 2017

Research frontier 2: Orthogonal inducing points

	Sparse GP	Orthogonal Sparse GP
Inducing points	$(\mathbf{z}, \mathbf{u})$	$(\mathbf{z}_\mathbf{u}, \mathbf{u}), (\mathbf{z}_\mathbf{v}, \mathbf{v})$
Prior factorisation	$p(f) = p(f_{\neq \mathbf{u}}\|\mathbf{u})p(\mathbf{u})$ $p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{0}, \mathbf{K}_{\mathbf{uu}})$	$p(f) = p(f_{\neq \mathbf{u},\mathbf{v}}\|\mathbf{u}, \mathbf{v})p(\mathbf{u}, \mathbf{v})$ $p(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{0}, \mathbf{K}_{\mathbf{uu}})$ $p(\mathbf{v}\|\mathbf{u}) = \mathcal{N}(\mathbf{v}; \mathbf{K}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u}, \mathbf{K}_{\mathbf{vv}} - \mathbf{K}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{K}_{\mathbf{uv}})$
Approx. posterior	$q(f) = p(f_{\neq \mathbf{u}}\|\mathbf{u})q(\mathbf{u})$ $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}, \Sigma)$	$q(f) = p(f_{\neq \mathbf{u},\mathbf{v}}\|\mathbf{u}, \mathbf{v})q(\mathbf{u})q(\mathbf{v}\|\mathbf{u})$ $q(\mathbf{u}) = \mathcal{N}(\mathbf{u}; \mathbf{m}_\mathbf{u}, \Sigma_\mathbf{u})$ $q(\mathbf{v}\|\mathbf{u}) = \mathcal{N}(\mathbf{v}; \mathbf{K}_{\mathbf{vu}}\mathbf{K}_{\mathbf{uu}}^{-1}\mathbf{u} + \mathbf{m}_\mathbf{v}, \Sigma_\mathbf{v})$

This might seem poor compared to $q(\mathbf{u}, \mathbf{v}) = \mathcal{N}([\mathbf{u}, \mathbf{v}]^\intercal; \mathbf{m}_{\mathbf{uv}}, \Sigma_{\mathbf{uv}})$ but is more efficient

Shi, Titsias and Mnih, Sparse orthogonal variational inference for GPs, AISTATS 2020

Research frontier 3: $q(f_{\neq \mathbf{u}} \,|\, \mathbf{u}) \neq p(f_{\neq \mathbf{u}} \,|\, \mathbf{u})$

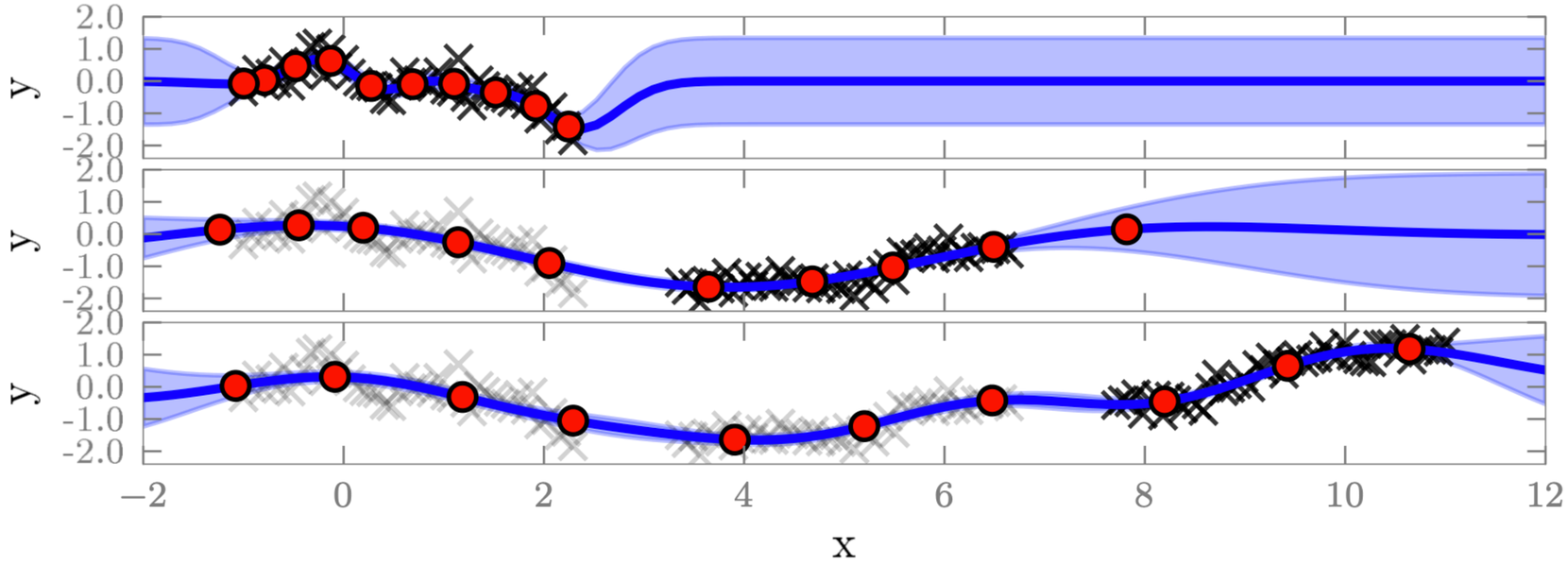
The variational approximation in eq. (4) is chosen such that the conditional $q(\boldsymbol{f}|\boldsymbol{u})$ identically matches the prior conditional $p(\boldsymbol{f}|\boldsymbol{u})$. Instead, we propose using the following variational posterior,

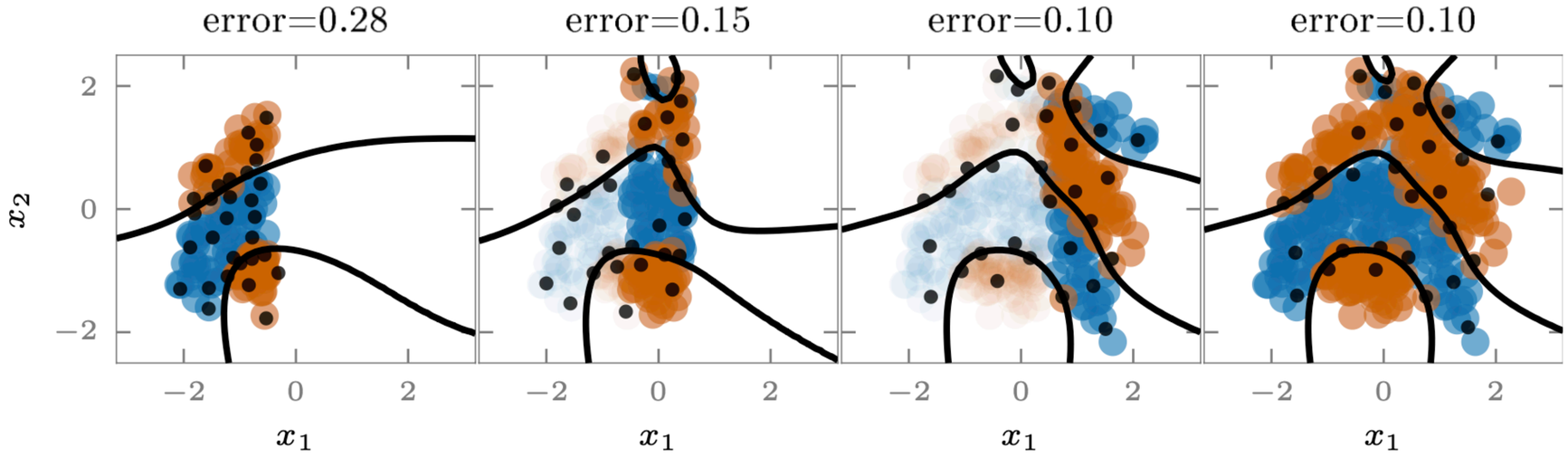
$$q(f) = p(f_{\neq \boldsymbol{f},\boldsymbol{u}}|\boldsymbol{f},\boldsymbol{u})q(\boldsymbol{f}|\boldsymbol{u})q(\boldsymbol{u}), \tag{7}$$

where $q(\boldsymbol{f}|\boldsymbol{u}) = \mathcal{N}(\boldsymbol{f}; \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\boldsymbol{u}; \mathbf{D_{ff}^{1/2}}\mathbf{M}\mathbf{D_{ff}^{\top/2}})$, $\mathbf{M}$ is a diagonal matrix, $\mathbf{M} = \text{diag}([m_1, m_2, \ldots, m_N])$ and $m_n > 0$. Note that the mean of the prior conditional $p(\boldsymbol{f}|\boldsymbol{u}) = \mathcal{N}(\boldsymbol{f}; \mathbf{K_{fu}}\mathbf{K_{uu}^{-1}}\boldsymbol{u}; \mathbf{D_{ff}})$ is retained in $q(\boldsymbol{f}|\boldsymbol{u})$.

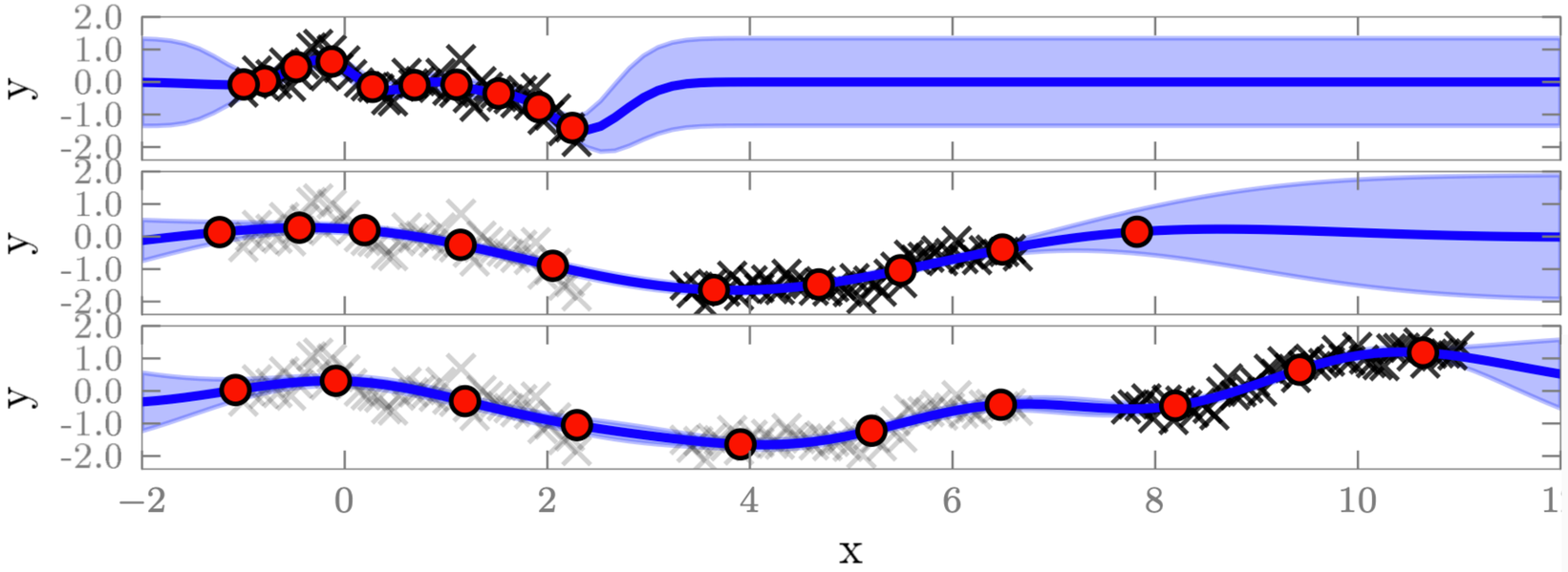
Bui, Ashman and Turner, Tighter sparse variational Gaussian processes, 2025
Titsias, New bounds for sparse variational Gaussian processes, 2025

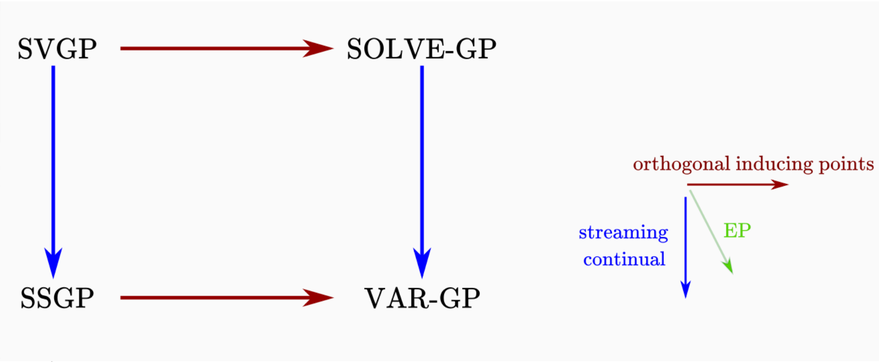
Research frontier 4: Continual learning

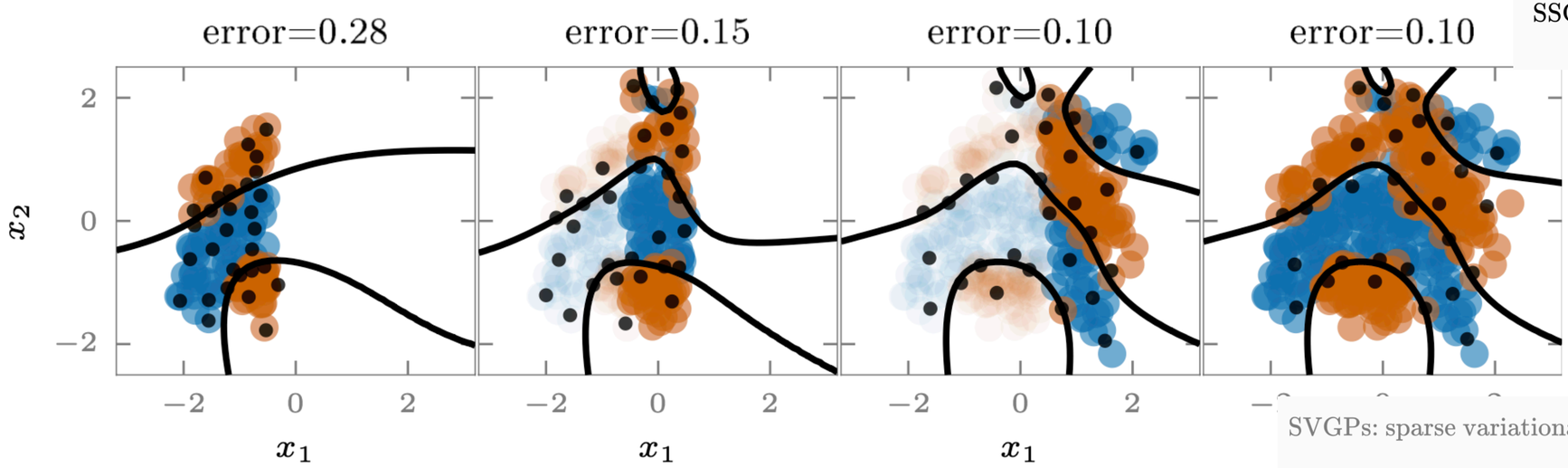




Research frontier 4: Continual learning







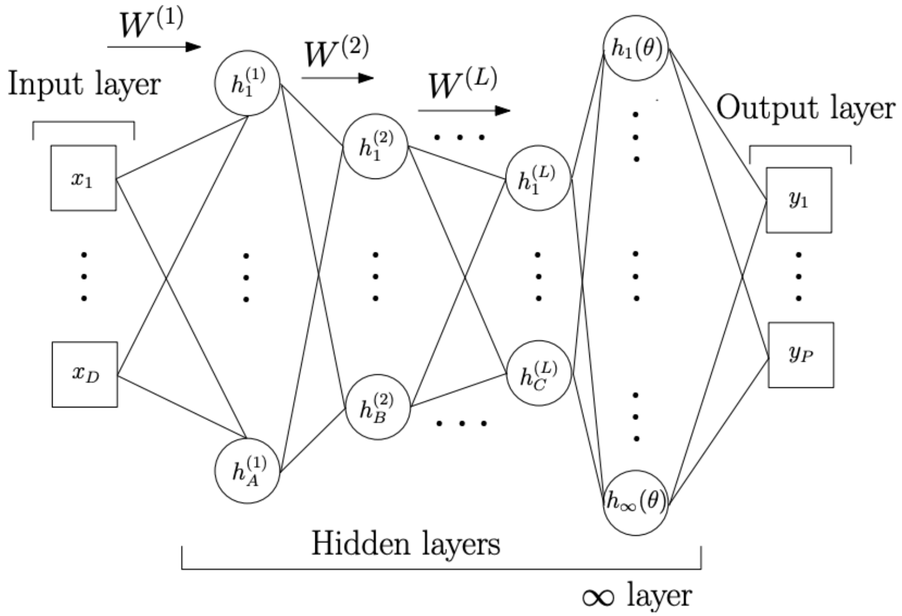
SVGPs: sparse variational inference for GPs (Titsias, 2009; Hensman et al, 2013)

SOLVE-GPs: sparse orthogonal variational inference for GPs (Shi et al, 2020)

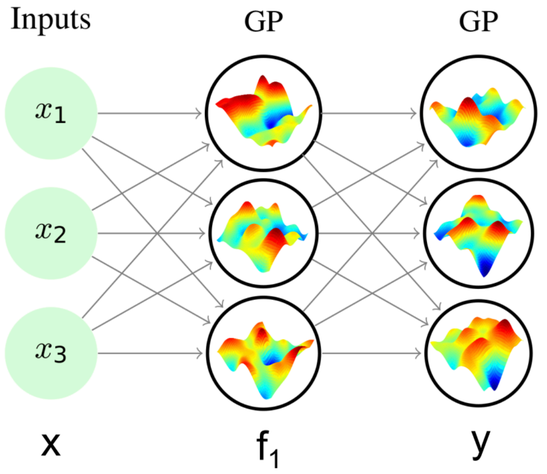
SSGP: streaming sparse variational inference for GPs (B.*, Nguyen*, and Turner, 2017)

VAR-GP: variational autoregressive GPs (Kapoor, Karaletsos and B., 2020)

Research frontier 5: Merging deep nets and GPs



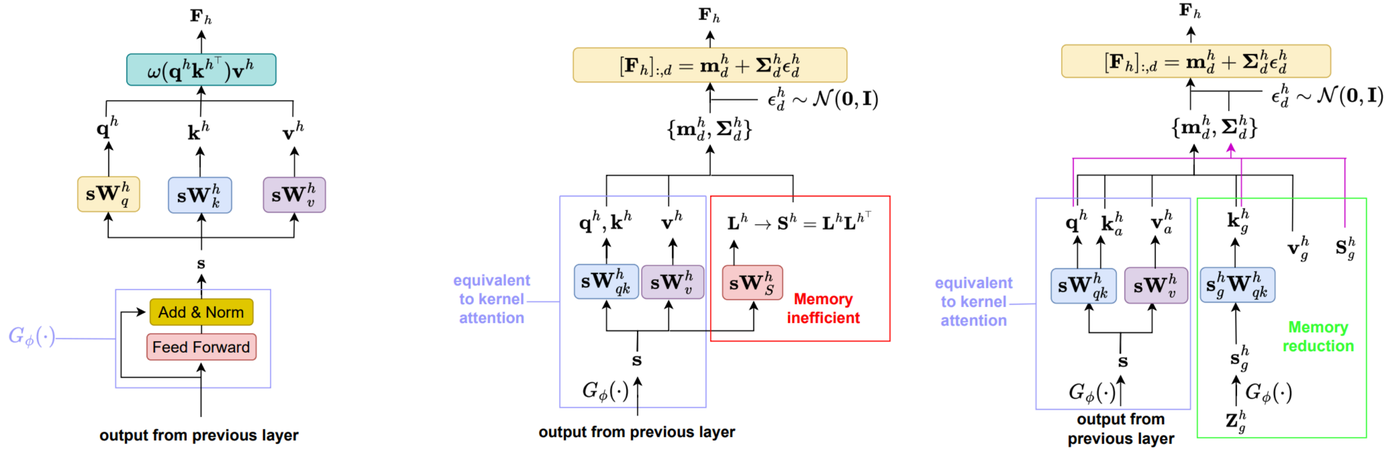
Deep Gaussian processes (Damianou and Lawrence, 2013)



awarded "Test of time" at AISTATS 2023

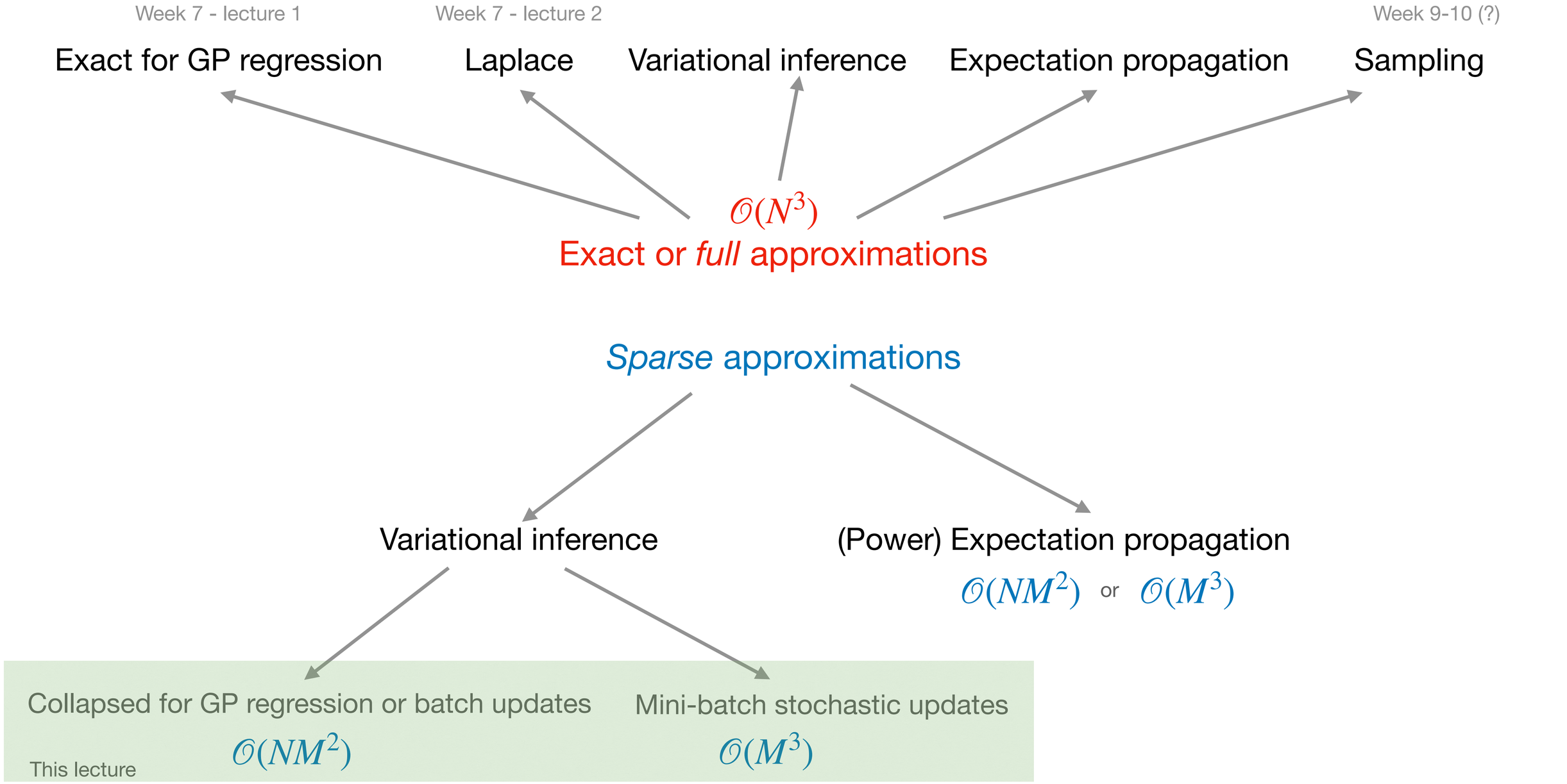
Deep kernel learning (Wilson et al, 2016)

Sparse GP attention (Chen and Li, 2023)

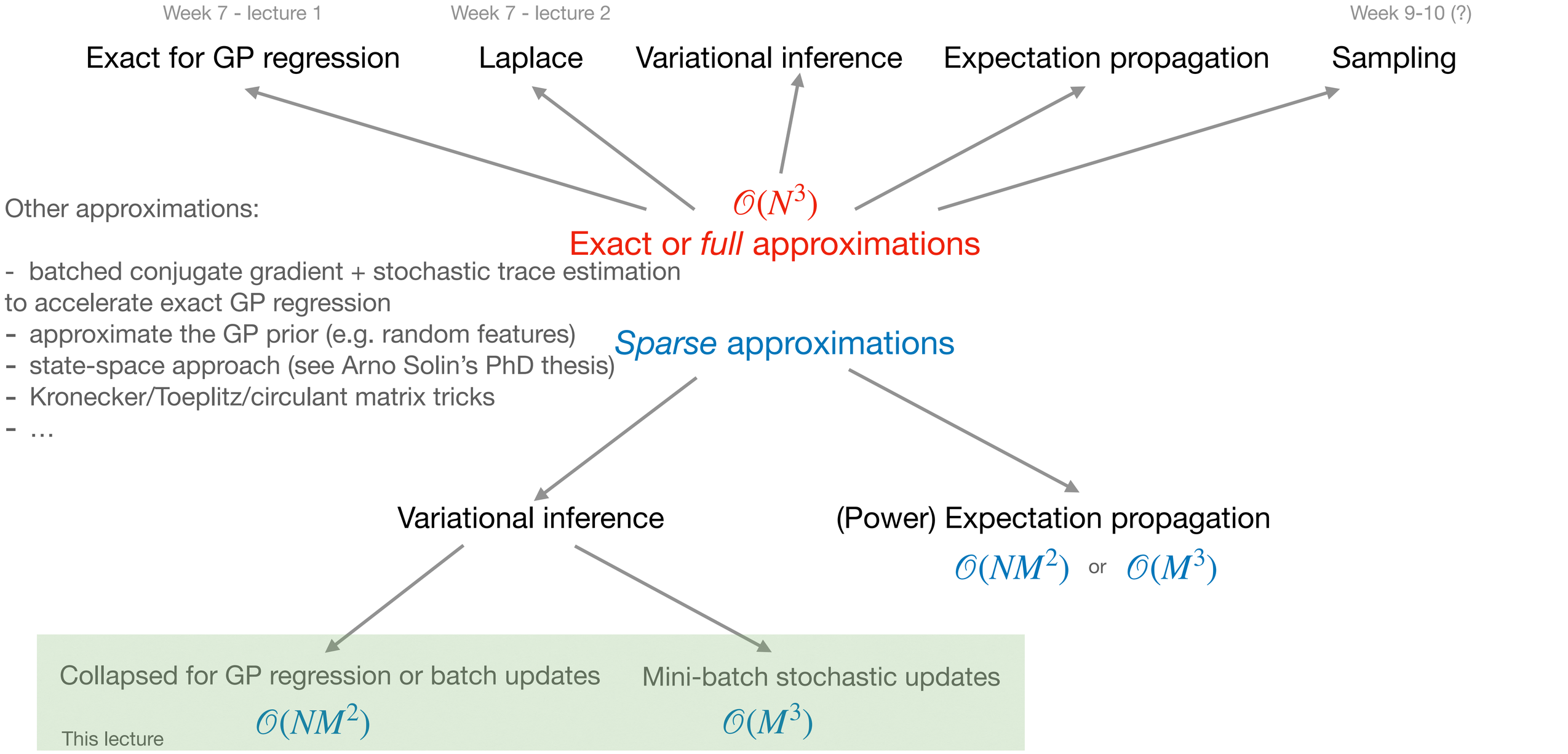


(a) Vanilla Transformer (b) Standard SGPA (ours) (c) Decoupled SGPA (ours)

Big picture


Week 7 - lecture 1
Week 7 - lecture 2
Week 9-10 (?)
Exact for GP regression
Laplace
Variational inference
Expectation propagation
Sampling
$\mathcal{O}(N^3)$
Exact or *full* approximations
Sparse approximations
Variational inference
(Power) Expectation propagation
$\mathcal{O}(NM^2)$ or $\mathcal{O}(M^3)$
Collapsed for GP regression or batch updates
$\mathcal{O}(NM^2)$
Mini-batch stochastic updates
$\mathcal{O}(M^3)$
This lecture

Big picture


Week 7 - lecture 1
Week 7 - lecture 2
Week 9-10 (?)
Exact for GP regression
Laplace
Variational inference
Expectation propagation
Sampling
$\mathcal{O}(N^3)$
Exact or *full* approximations
Other approximations:
- batched conjugate gradient + stochastic trace estimation
to accelerate exact GP regression
- approximate the GP prior (e.g. random features)
- state-space approach (see Arno Solin's PhD thesis)
- Kronecker/Toeplitz/circulant matrix tricks
- …
Sparse approximations
Variational inference
(Power) Expectation propagation
$\mathcal{O}(NM^2)$ or $\mathcal{O}(M^3)$
Collapsed for GP regression or batch updates
$\mathcal{O}(NM^2)$
Mini-batch stochastic updates
$\mathcal{O}(M^3)$
This lecture

Thanks for your attention!

Questions?

Feedback and additional questions, email thang.bui@anu.edu.au