CS-E4075 Special course on Gaussian processes: Session #3

Markus Heinonen

Aalto University

markus.o.heinonen@aalto.fi

Monday 18.1.2021

Agenda for today

- **Quick summary of last session**
- **Covariance functions**
  - Definition and properties
  - Commonly used covariance functions
- **Model selection and evaluation**
  - Marginal likelihood
  - Mean log posterior predictive likelihood
- **Computational complexity of GPs**
  - Computational cost
  - Memory requirements

Last time (I)

- Weight view $p(\boldsymbol{w})$ vs. function view $p(\boldsymbol{f})$

$$p(\boldsymbol{y}, \boldsymbol{w}) = p(\boldsymbol{y}|\boldsymbol{w})p(\boldsymbol{w}) \qquad \text{vs.} \qquad p(\boldsymbol{y}, \boldsymbol{f}) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}) \tag{1}$$

- Gaussian process can be seen as prior distributions over functions

- GPs are characterized by a **mean function** $m(\boldsymbol{x})$ and **the covariance function** $k(\boldsymbol{x}, \boldsymbol{x}')$

$$f(\boldsymbol{x}) \sim \mathcal{GP}\left(m(\boldsymbol{x}), k(\boldsymbol{x}, \boldsymbol{x}')\right) \tag{2}$$

- The choice of covariance function determines the characteristics of the function f

$$\mathbb{E}\left[f(\boldsymbol{x})\right] = m\left(\boldsymbol{x}\right) \tag{3}$$

$$cov[f(\boldsymbol{x}), f(\boldsymbol{x'})] = k\left(\boldsymbol{x}, \boldsymbol{x}'\right) \tag{4}$$

Last time (II)

- Goal: Given the model $y_n = f(\mathbf{x}_n) + \epsilon_n$ and a training data set $\{\mathbf{x}_n, y_n\}_{n=1}^N$, predict the value of the function $f(\mathbf{x}_*)$ evaluated at the test point $\mathbf{x}_*$
- Joint model for training and test data

$$p(\mathbf{y}, \mathbf{f}, f_*) = p(\mathbf{y}|\mathbf{f})p(\mathbf{f}, f_*) = \mathcal{N}\left(\mathbf{y}|\mathbf{f}, \sigma_{obs}^2 \mathbf{I}\right) \mathcal{N}\left(\begin{bmatrix} \mathbf{f} \\ f_* \end{bmatrix} \middle| 0, \begin{bmatrix} \mathbf{K}_{ff} & \mathbf{k}_{f_* f} \\ \mathbf{k}_{f_* f} & K_{f_* f_*} \end{bmatrix}\right) \tag{5}$$

where

- $\mathbf{K}_{ff}$ is the covariance matrix for training inputs

$$(\mathbf{K}_{ff})_{ij} = \mathrm{cov}\left(f(\mathbf{x}_i), f(\mathbf{x}_j)\right) \tag{6}$$

- $\mathbf{K}_{f_* f}$ is the covariance vector for between test input and training inputs

$$(\mathbf{k}_{f_* f})_j = \mathrm{cov}\left(f(\mathbf{x}_*), f(\mathbf{x}_j)\right) \tag{7}$$

- $K_{f_* f_*}$ is the variance of the test input

$$K_{f_* f_*} = \mathrm{cov}\left(f(\mathbf{x}_*), f(\mathbf{x}_*)\right) \tag{8}$$

Last time (III)

- Step 1: Write the joint model

$$p(\boldsymbol{y}, \boldsymbol{f}, f_*) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}, f_*) = \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f}, \sigma_{obs}^2 \boldsymbol{I}\right) \mathcal{N}\left(\begin{bmatrix} \boldsymbol{f} \\ f_* \end{bmatrix} \Big| 0, \begin{bmatrix} \boldsymbol{K}_{ff} & \boldsymbol{k}_{f_* f} \\ \boldsymbol{k}_{f_* f} & K_{f_* f_*} \end{bmatrix}\right) \quad (9)$$

- Step 2: Marginalize over $\boldsymbol{f}$

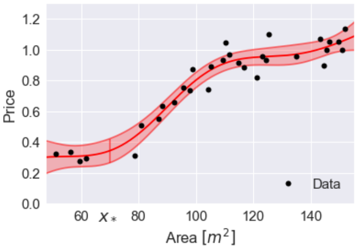
$$p(\boldsymbol{y}, f_*) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}, f_*)\mathrm{d}\boldsymbol{f} = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{y} \\ f_* \end{bmatrix} \boldsymbol{f} \Big| 0, \begin{bmatrix} \boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I} & \boldsymbol{K}_{f_* f} \\ \boldsymbol{K}_{f_* f} & K_{f_* f_*} \end{bmatrix}\right) \quad (10)$$

- Step 3: Compute conditional distribution $p(f_*|\boldsymbol{y})$

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_*|\mu_*, \sigma_*^2\right) \quad (11)$$

$$\mu_* = \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \quad (12)$$

$$\sigma_*^2 = K_{f_* f_*} - \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^T \quad (13)$$



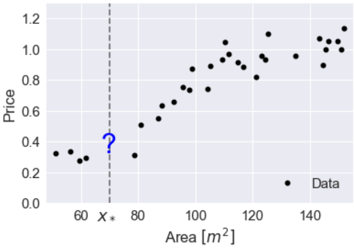
Example: The components of the posterior distribution I

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$

$N = 31$ data points

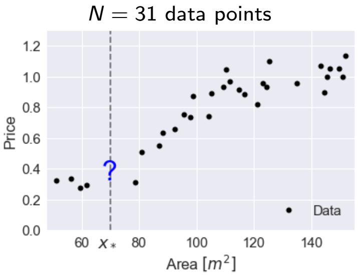


Example: The components of the posterior distribution I

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$



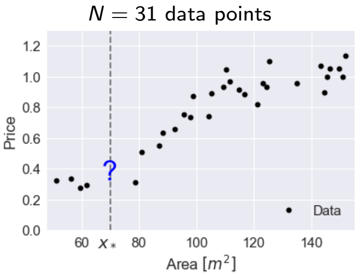
- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$

Example: The components of the posterior distribution I

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^T \tag{16}$$



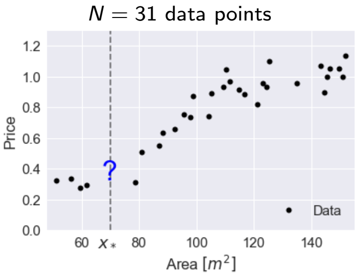
- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$

Example: The components of the posterior distribution I

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$



- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$

Example: The components of the posterior distribution I

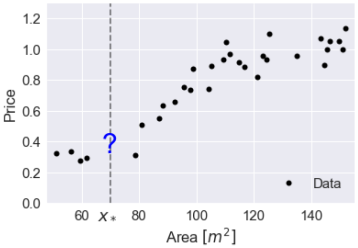
$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

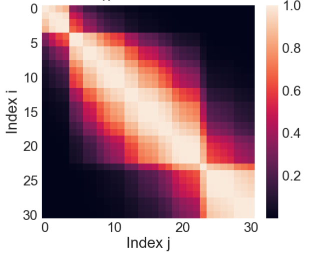
$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$

- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x, x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Cov. matrix of training: $[\mathbf{K}_{ff}]_{ij} = k(x_i, x_j)$

$N = 31$ data points



$\mathbf{K}_{ff} \in \mathbb{R}^{31\times 31}$

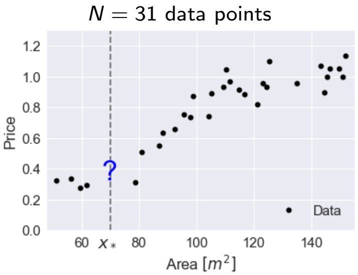


Example: The components of the posterior distribution I

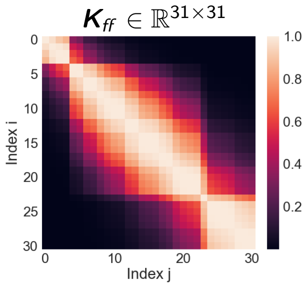
$$p(f_* | \boldsymbol{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^T \tag{16}$$



- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31 \times 1}$
- $k(x, x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2 \cdot 20^2}\right]$
- Cov. matrix of training: $[\boldsymbol{K}_{ff}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\boldsymbol{k}_{f_* f}]_j = k(x_*, x_j)$



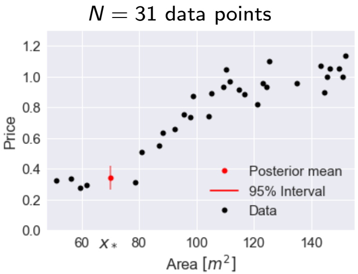
Example: The components of the posterior distribution I

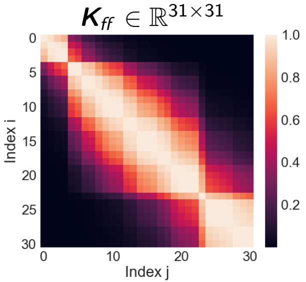
$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$

- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31 \times 1}$
- $k(x, x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2 \cdot 20^2}\right]$
- Cov. matrix of training: $[\mathbf{K}_{ff}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\mathbf{k}_{f_* f}]_j = k(x_*, x_j)$
- Covariance of $f(x_*)$: $K_{f_* f_*} = k(x_*, x_*)$





Example: The components of the posterior distribution I

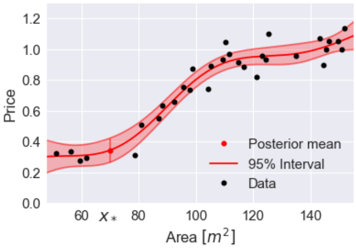
$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{14}$$

$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{15}$$

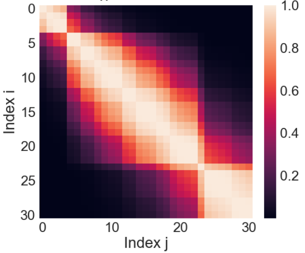
$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{16}$$

- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\mathbf{y} = [y_1, y_2, \ldots, y_{31}]^T \in \mathbb{R}^{31\times 1}$
- $k(x,x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Cov. matrix of training: $[\mathbf{K}_{ff}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\mathbf{k}_{f_* f}]_j = k(x_*, x_j)$
- Covariance of $f(x_*)$: $K_{f_* f_*} = k(x_*, x_*)$

$N = 31$ data points



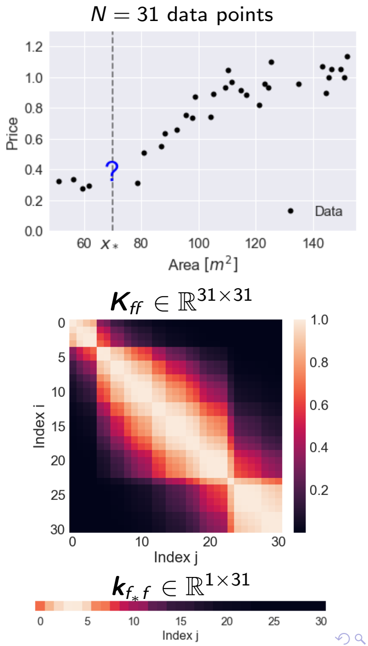
$\mathbf{K}_{ff} \in \mathbb{R}^{31\times 31}$



$\mathbf{k}_{f_* f} \in \mathbb{R}^{1\times 31}$

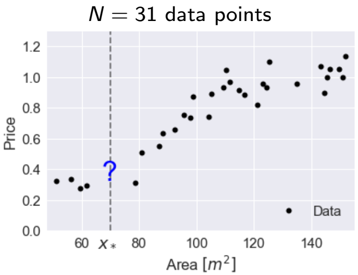
Example: The components of the posterior distribution II

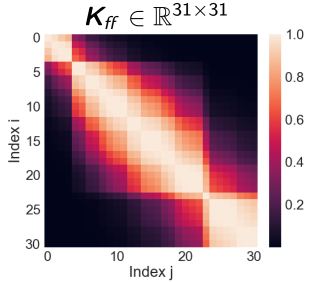
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$



Example: The components of the posterior distribution II

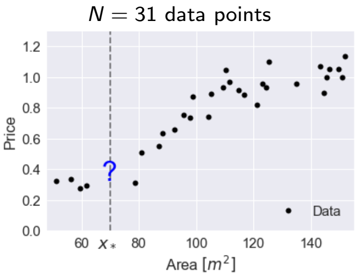
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \in \mathbb{R}^{1 \times 31}$

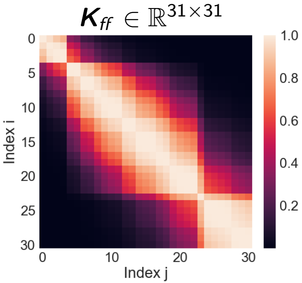




Example: The components of the posterior distribution II

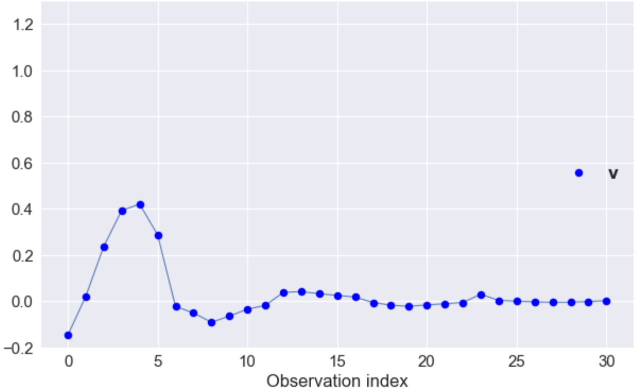
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs}\boldsymbol{I}\right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs}\boldsymbol{I}\right)^{-1} \in \mathbb{R}^{1\times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^T \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



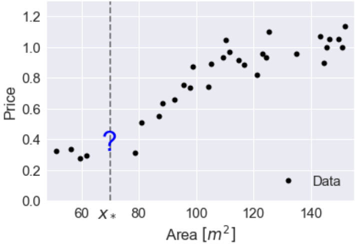


Example: The components of the posterior distribution II

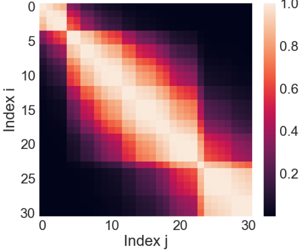
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs}\boldsymbol{I}\right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs}\boldsymbol{I}\right)^{-1} \in \mathbb{R}^{1\times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^T \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



$N = 31$ data points



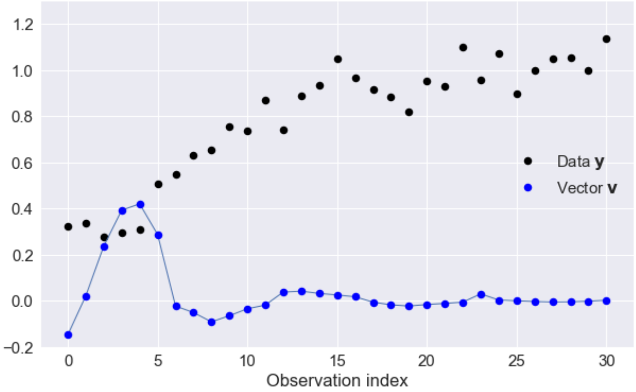
$\boldsymbol{K}_{ff} \in \mathbb{R}^{31\times 31}$



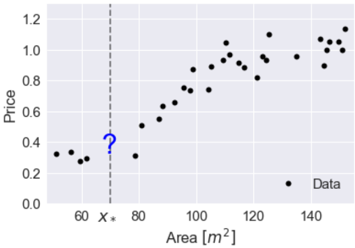
$\boldsymbol{k}_{f_* f} \in \mathbb{R}^{1\times 31}$

Example: The components of the posterior distribution II

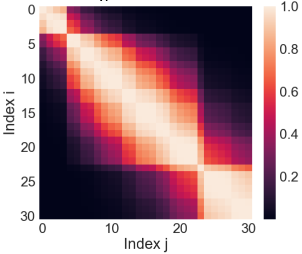
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \in \mathbb{R}^{1 \times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^T \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



$N = 31$ data points



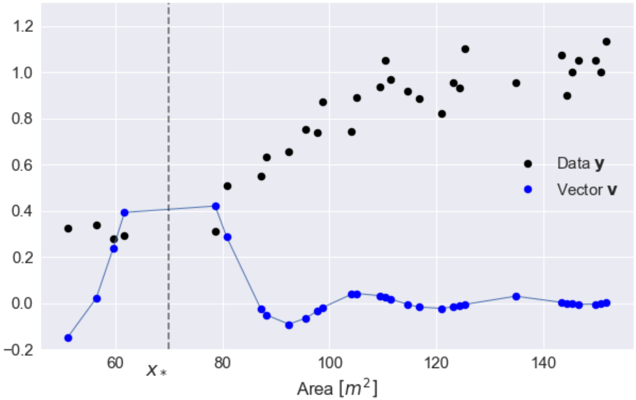
$\boldsymbol{K}_{ff} \in \mathbb{R}^{31 \times 31}$



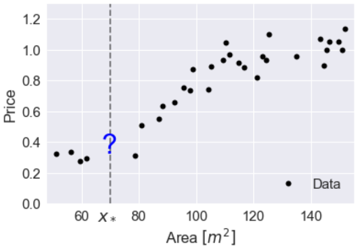
$\boldsymbol{k}_{f_* f} \in \mathbb{R}^{1 \times 31}$

Example: The components of the posterior distribution II

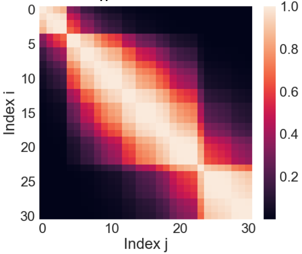
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \in \mathbb{R}^{1 \times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^T \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



$N = 31$ data points



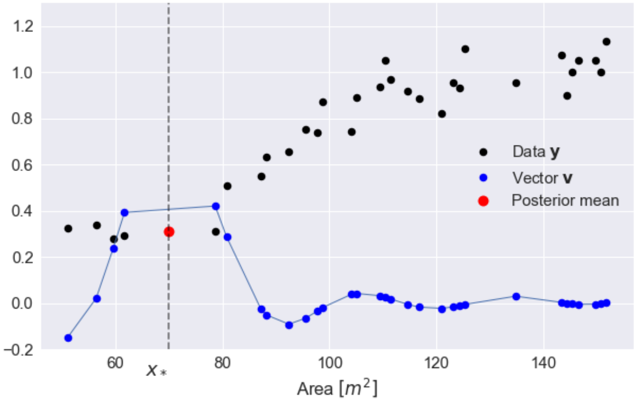
$\boldsymbol{K}_{ff} \in \mathbb{R}^{31 \times 31}$

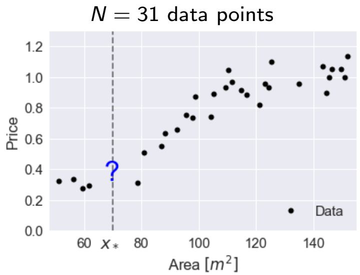


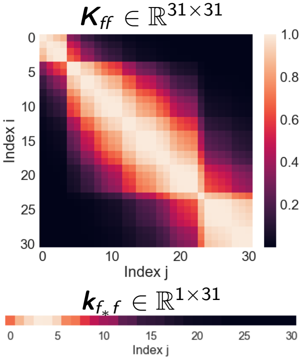
$\boldsymbol{k}_{f_* f} \in \mathbb{R}^{1 \times 31}$

Example: The components of the posterior distribution II

- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^T = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{obs} \boldsymbol{I} \right)^{-1} \in \mathbb{R}^{1\times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^T \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$







Question

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{17}$$

$$\mu_* = \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{18}$$

$$\sigma_*^2 = K_{f_* f_*} - \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{obs}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^T \tag{19}$$

1. What happens to the posterior distribution of f_* if $\boldsymbol{x}_*$ is so far away from the training data that the covariances between $\boldsymbol{x}_*$ and the training data $\{\boldsymbol{x}_n\}_{n=1}^N$ are effectively equal to zero?

2. How would you plot the of the vector $\boldsymbol{v}$ change (from the previous slide), if we changed the kernel function from k to k_2?

$$k(x, x') = \exp\left[-\frac{(x-x')^2}{2 \cdot 20^2}\right] \qquad k_2(x, x') = \exp\left[-\frac{(x-x')^2}{2 \cdot 40^2}\right] \tag{20}$$

3. What is the difference between σ_{obs}^2 and σ_*^2?

4. What is the difference between $p(f_*|\boldsymbol{y})$ and $p(y_*|\boldsymbol{y})$

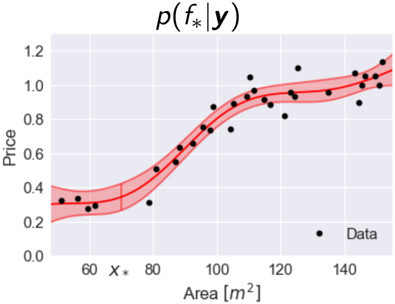
$p(f_*|\mathbf{y})$ vs $p(y_*|\mathbf{y})$

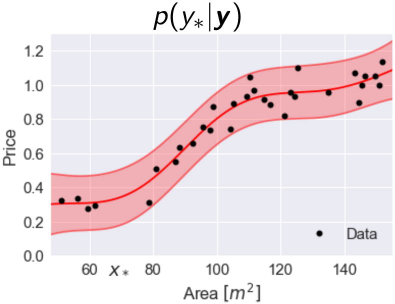
- The model is given by: $y_n = f(x_n) + \epsilon$
- The posterior of the function evaluated at x_*

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{21}$$

- The predictive distribution of y_*

$$p(y_*|\mathbf{y}) = \int p(y_*|f_*)p(f_*|\mathbf{y})\mathrm{d}f_* \tag{22}$$





Covariance functions

- A covariance function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ maps a pair of inputs $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$ from some input space $\mathcal{X}$ to the real line $\mathbb{R}$
- Not all functions of the form $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ are valid covariance functions
- Recall: the covariance / kernel matrix given by

$$\boldsymbol{K}_{ij} = \operatorname{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x})_j\right) = k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \tag{23}$$

- Covariance functions must be symmetric & Positive (Semi) Definite such that

$$\text{(Symmetric)} \quad \boldsymbol{K} = \boldsymbol{K}^T \tag{24}$$

$$\text{(PSD)} \quad \forall \boldsymbol{x} \neq 0 : \quad \boldsymbol{x}^T \boldsymbol{K} \boldsymbol{x} \geq 0 \tag{25}$$

  PSD matrices are (usually) invertible

- Must hold for all possible data sets $\{\boldsymbol{x}_n\}_{n=1}^N \subset \mathcal{X}$ in the input space $\mathcal{X}$

Stationary covariance function

- A covariance function k is said to be **stationary** if $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ only depends on the difference of the inputs

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\boldsymbol{x}_1 - \boldsymbol{x}_2), \qquad \textit{or} \qquad k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\boldsymbol{x}_1 + \boldsymbol{a}, \boldsymbol{x}_2 + \boldsymbol{a}) \tag{26}$$

- A covariance function is said to be **isotropic** (or rotation invariant) if $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ only depends on the norm of the difference of the inputs

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|) \tag{27}$$

Stationary covariance function

- A covariance function k is said to be **stationary** if $k(\mathbf{x}_1, \mathbf{x}_2)$ only depends on the difference of the inputs

$$k(\mathbf{x}_1, \mathbf{x}_2) = k(\mathbf{x}_1 - \mathbf{x}_2), \qquad \textit{or} \qquad k(\mathbf{x}_1, \mathbf{x}_2) = k(\mathbf{x}_1 + \mathbf{a}, \mathbf{x}_2 + \mathbf{a}) \tag{26}$$

- A covariance function is said to be **isotropic** (or rotation invariant) if $k(\mathbf{x}_1, \mathbf{x}_2)$ only depends on the norm of the difference of the inputs

$$k(\mathbf{x}_1, \mathbf{x}_2) = k(\|\mathbf{x}_1 - \mathbf{x}_2\|) \tag{27}$$

- Which of the following kernels are stationary? isotropic?

$$k(\mathbf{x}_1, \mathbf{x}_2) = \mathbf{x}_1^T \mathbf{x}_2 \tag{linear}$$

$$k(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2}\right) \tag{squared exponential1}$$

$$k(\mathbf{x}_1, \mathbf{x}_2) = \exp\left(-\frac{\sum_{d=1}^{D} \rho_d^{-1} |x_{1,d} - x_{2,d}|^2}{2}\right) \tag{squared exponential2}$$

Table of common covariance functions

From the book

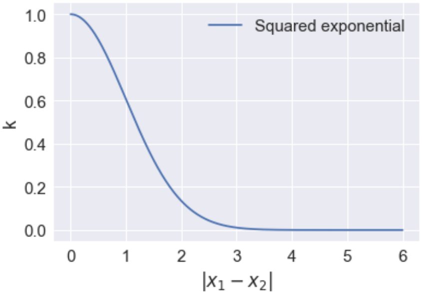
covariance function	expression	S	ND
constant	σ_0^2	$\surd$	
linear	$\sum_{d=1}^{D} \sigma_d^2 x_d x'_d$		
polynomial	$(\mathbf{x} \cdot \mathbf{x}' + \sigma_0^2)^p$		
squared exponential	$\exp(-\frac{r^2}{2\ell^2})$	$\surd$	$\surd$
Matérn	$\frac{1}{2^{\nu-1}\Gamma(\nu)}\left(\frac{\sqrt{2\nu}}{\ell}r\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}}{\ell}r\right)$	$\surd$	$\surd$
exponential	$\exp(-\frac{r}{\ell})$	$\surd$	$\surd$
γ-exponential	$\exp\left(-(\frac{r}{\ell})^{\gamma}\right)$	$\surd$	$\surd$
rational quadratic	$(1 + \frac{r^2}{2\alpha\ell^2})^{-\alpha}$	$\surd$	$\surd$
neural network	$\sin^{-1}\left(\frac{2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}}'}{\sqrt{(1+2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}})(1+2\tilde{\mathbf{x}}'^\top \Sigma \tilde{\mathbf{x}}')}}\right)$		$\surd$

Another great resource for covariance functions:
`www.cs.toronto.edu/~duvenaud/cookbook/`

The squared exponential covariance function (I)

- The squared exponential (also known as gaussian/exponentiad quadractic/radial basis) covariance function

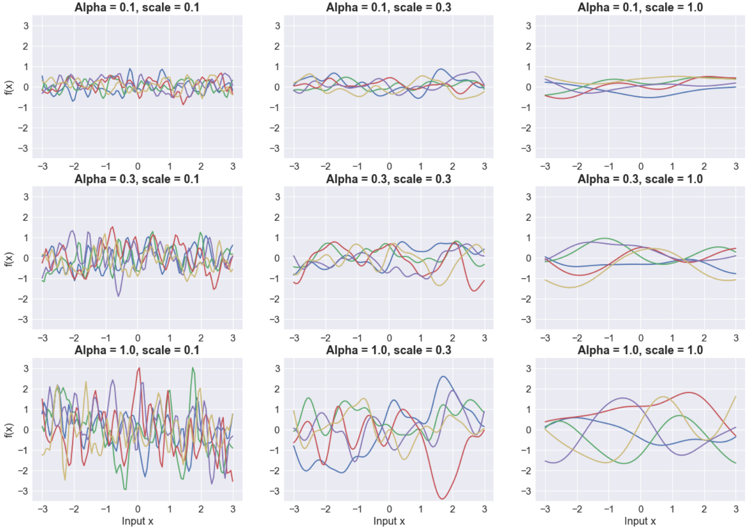
$$k(\mathbf{x}_1, \mathbf{x}_2) = k(\|\mathbf{x}_1 - \mathbf{x}_2\|) = \alpha \exp\left(-\frac{\|\mathbf{x}_1 - \mathbf{x}_2\|^2}{2\ell^2}\right) \tag{28}$$



- Parameters
  1. α: variance (magnitude / height)
  2. ℓ: lengthscale (smoothness)
- Stationary
- Produces very smooth functions (infitely differentiable)
- Some argue that such strong smoothness assumptions are unrealistic for many physical processes

The squared exponential covariance function (II)

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \alpha \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\ell^2}\right) \tag{29}$$

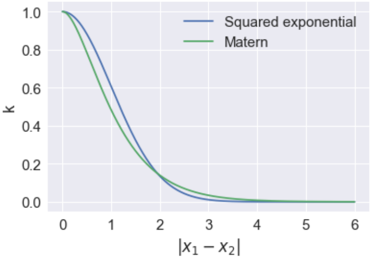


The Matern covariance function (I)

- Matern class covariance function

$$k(\mathbf{x}_1, \mathbf{x}_2) = \alpha \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\ell} \right)^{\nu} K_\nu \left(\sqrt{2\nu} \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\ell} \right) \quad (30)$$

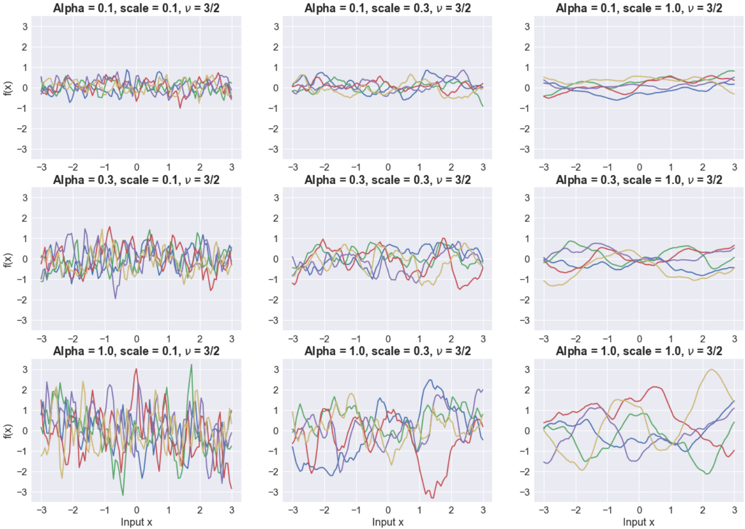
  where K_ν is a modified Bessel function.



- Parameters
  1. α: magnitude
  2. ℓ: lengthscale
  3. ν: Samples paths are $\lfloor \nu - 1 \rfloor$ times differentiable
- Stationary
- $\nu = \frac{3}{2}$ or $\nu = \frac{5}{2}$ are often used
- $\nu \to \infty$ gives SE kernel

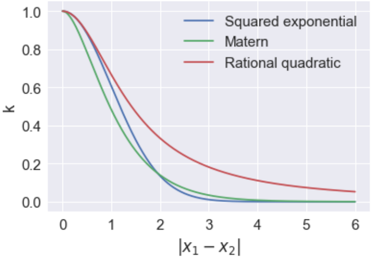
The Matern covariance function (II)

$$k(\mathbf{x}_1, \mathbf{x}_2) = \alpha \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\ell} \right)^{\nu} K_{\nu} \left(\sqrt{2\nu} \frac{\|\mathbf{x}_1 - \mathbf{x}_2\|}{\ell} \right) \quad (31)$$



Rational Quadratic (I)

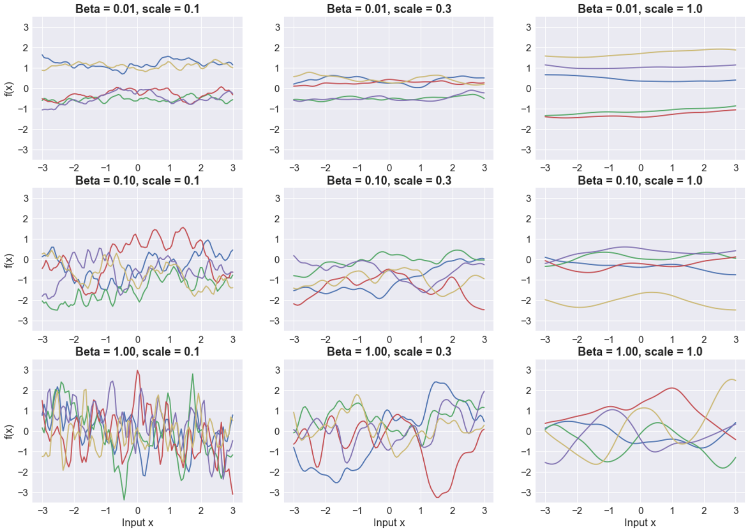
$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \alpha \left(1 + \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\beta\ell^2}\right)^{-\beta} \quad (32)$$



- Parameters
  1. α: magnitude
  2. β: power
  3. ℓ: lengthscale
- Becomes identical to the squared exponential as $\beta \to \infty$
- Interpretation as scale mixture of squared exponentials (adding many squared exponential kernels with different lengthscales)
- Can model functions that vary across several lengthscales
- Commonly used in spatial statistics (geostatistics, imageanalysis, etc..)

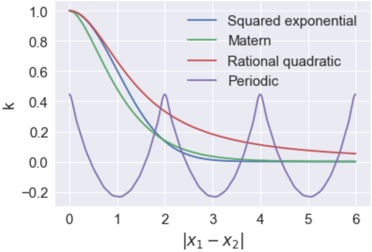
Rational Quadratic (II)

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \alpha\left(1 + \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\beta\ell^2}\right)^{-\beta} \tag{33}$$

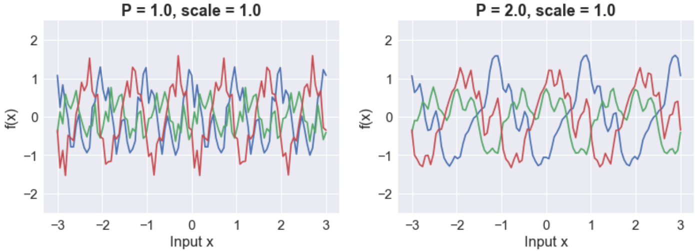


Covariance function for periodic functions

$$k(x_1, x_2) = \alpha \exp\left(-\frac{2}{\ell}\sin^2\left(\frac{\pi|x_1 - x_2|}{P}\right)\right) \tag{34}$$



- Parameters
  1. α: magnitude
  2. ℓ: lengthscale
  3. P: Period



Building new kernels from old ones (I)

Requirements for valid kernels:

$$\text{(Symmetric)} \quad \boldsymbol{K} = \boldsymbol{K}^T \tag{35}$$

$$\text{(PSD)} \quad \forall \boldsymbol{x} \neq 0 : \quad \boldsymbol{x}^T \boldsymbol{K} \boldsymbol{x} \geq 0 \tag{36}$$

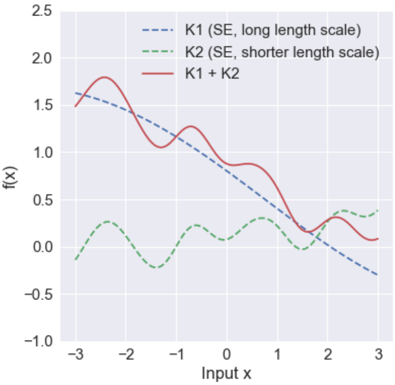
1. Sums of two kernels: $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k_1(\boldsymbol{x}_1, \boldsymbol{x}_2) + k_2(\boldsymbol{x}_1, \boldsymbol{x}_2)$

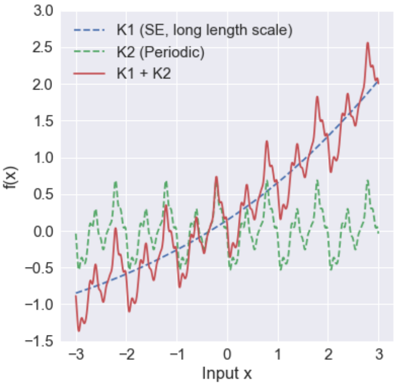
2. Products of two kernels: $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k_1(\boldsymbol{x}_1, \boldsymbol{x}_2)\, k_2(\boldsymbol{x}_1, \boldsymbol{x}_2)$

3. Scaling by $a(\boldsymbol{x})$: $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = a(\boldsymbol{x}_1) k_1(\boldsymbol{x}_1, \boldsymbol{x}_2)\, a(\boldsymbol{x}_2)$

Building new kernels from old ones (II)

- Adding two SEs kernels to model long term trends (long length scale) and short term fluctuations (short length scale)
- Adding SE and period kernels to model long term trends (long length scale) and periodic fluctuations





Building new kernels from old ones (III)

Techniques for Constructing New Kernels.

Given valid kernels $k_1(\mathbf{x}, \mathbf{x}')$ and $k_2(\mathbf{x}, \mathbf{x}')$, the following new kernels will also be valid:

$$
\begin{aligned}
k(\mathbf{x}, \mathbf{x}') &= ck_1(\mathbf{x}, \mathbf{x}') && (6.13)\\
k(\mathbf{x}, \mathbf{x}') &= f(\mathbf{x})k_1(\mathbf{x}, \mathbf{x}')f(\mathbf{x}') && (6.14)\\
k(\mathbf{x}, \mathbf{x}') &= q\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.15)\\
k(\mathbf{x}, \mathbf{x}') &= \exp\left(k_1(\mathbf{x}, \mathbf{x}')\right) && (6.16)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}') + k_2(\mathbf{x}, \mathbf{x}') && (6.17)\\
k(\mathbf{x}, \mathbf{x}') &= k_1(\mathbf{x}, \mathbf{x}')k_2(\mathbf{x}, \mathbf{x}') && (6.18)\\
k(\mathbf{x}, \mathbf{x}') &= k_3\left(\boldsymbol{\phi}(\mathbf{x}), \boldsymbol{\phi}(\mathbf{x}')\right) && (6.19)\\
k(\mathbf{x}, \mathbf{x}') &= \mathbf{x}^{\mathrm{T}}\mathbf{A}\mathbf{x}' && (6.20)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}_a') + k_b(\mathbf{x}_b, \mathbf{x}_b') && (6.21)\\
k(\mathbf{x}, \mathbf{x}') &= k_a(\mathbf{x}_a, \mathbf{x}_a')k_b(\mathbf{x}_b, \mathbf{x}_b') && (6.22)
\end{aligned}
$$

where $c > 0$ is a constant, $f(\cdot)$ is any function, $q(\cdot)$ is a polynomial with nonnegative coefficients, $\boldsymbol{\phi}(\mathbf{x})$ is a function from $\mathbf{x}$ to $\mathbb{R}^M$, $k_3(\cdot, \cdot)$ is a valid kernel in $\mathbb{R}^M$, $\mathbf{A}$ is a symmetric positive semidefinite matrix, $\mathbf{x}_a$ and $\mathbf{x}_b$ are variables (not necessarily disjoint) with $\mathbf{x} = (\mathbf{x}_a, \mathbf{x}_b)$, and k_a and k_b are valid kernel functions over their respective spaces.

Question: Can you prove that the squared exponential is a valid kernel?

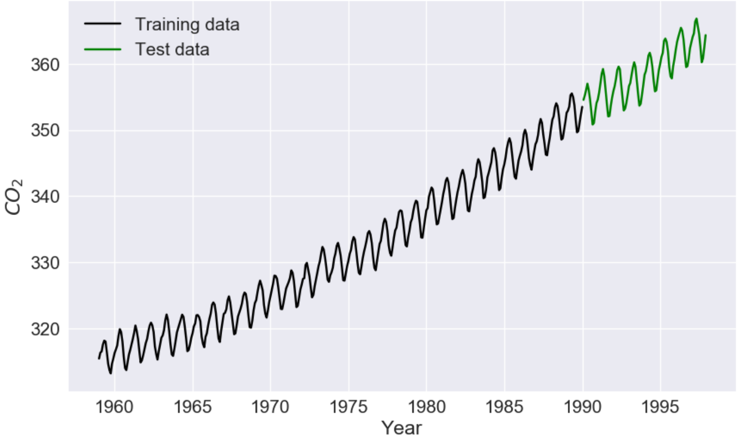
$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2}\right) \qquad (37)$$

Hint: $\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2 = (\boldsymbol{x}_1 - \boldsymbol{x}_2)^T(\boldsymbol{x}_1 - \boldsymbol{x}_2)$

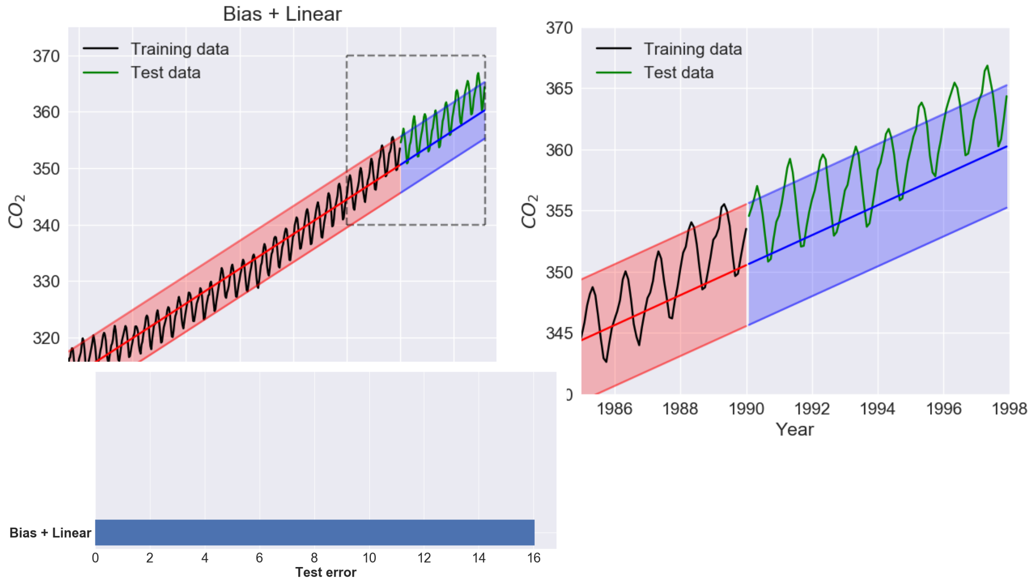
From Chris Bishops book: `https://www.microsoft.com/en-us/research/people/cmbishop`

Example: Mauna Loa data set

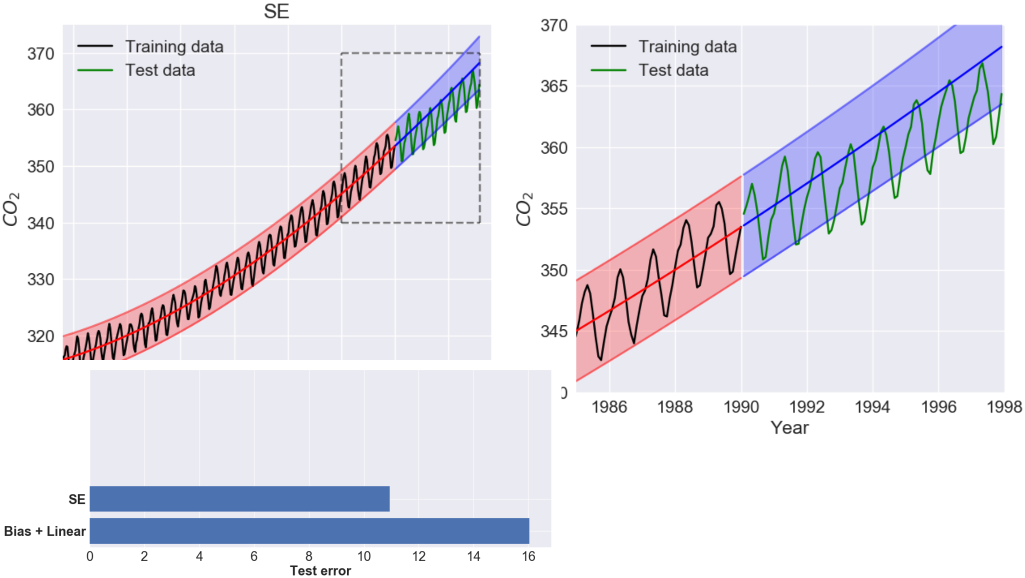
- Measurements of monthly average atmospheric CO_2 concentrations (in parts per million by volume (ppmv))
- Collected at Mauna Loa Observatory, Hawaii from 1958 to 1998



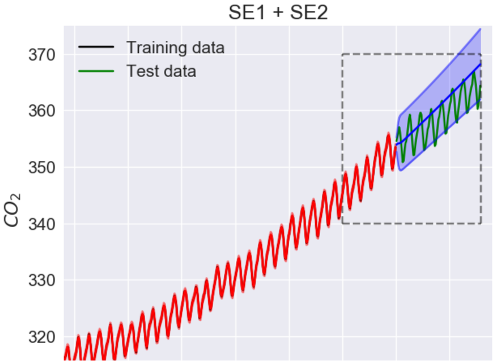
Example: Mauna Loa data set

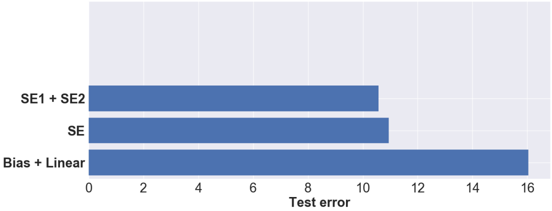


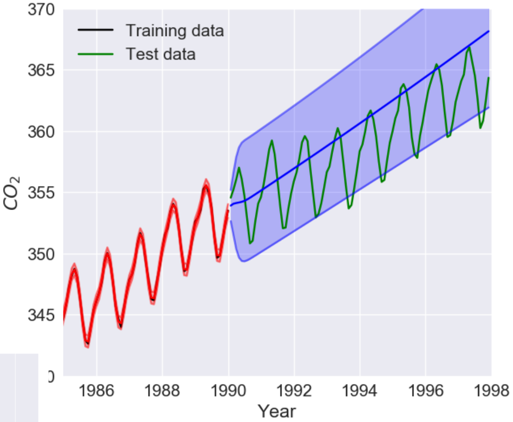
Example: Mauna Loa data set



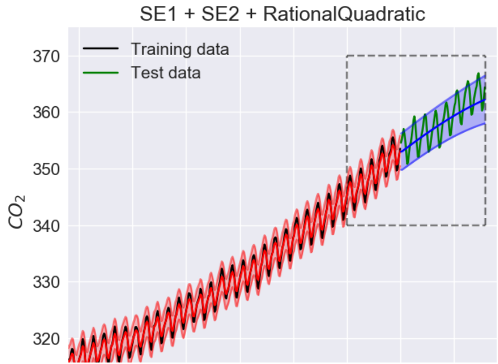
Example: Mauna Loa data set

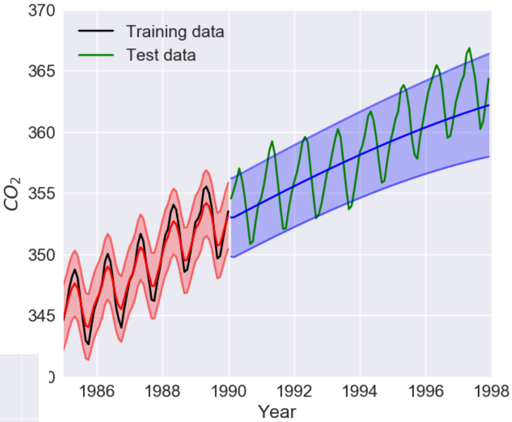


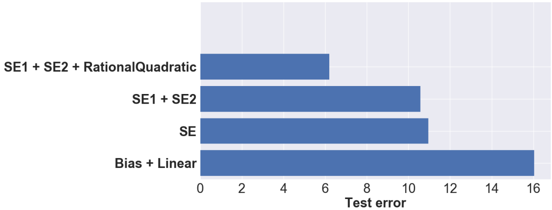




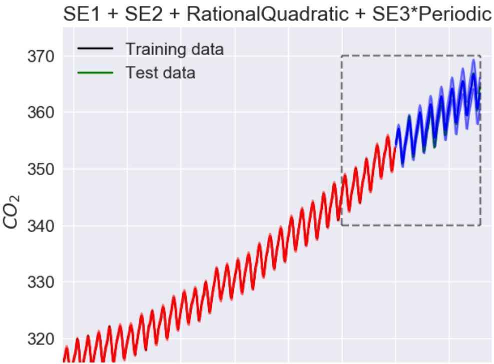
Example: Mauna Loa data set

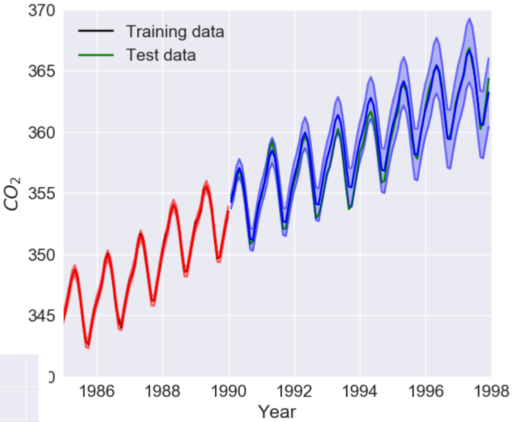


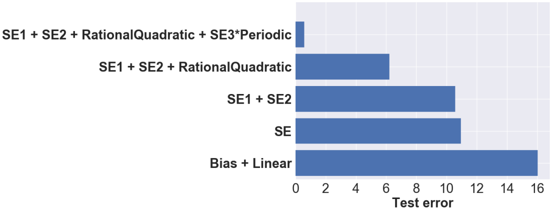




Example: Mauna Loa data set







Hyperparameters & model selection (I)

- Almost all covariance functions have hyperparameters
- How do we choose values for them?
- Ideally, we would like to put prior distributions on the hyperparameters and compute the posterior
- Let $\boldsymbol{\theta}$ be the hyperparameters of interest, then

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y})} \tag{38}$$

but in this case the marginal likelihood is almost always intractable

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \tag{39}$$

Hyperparameters & model selection (II)

- Approximation: We will use the MAP (Maximum a posterior estimate)
- $p(\boldsymbol{y})$ is constant wrt. $\boldsymbol{\theta}$

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y})} \propto p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{40}$$

- The MAP estimate is defined as

$$\hat{\theta}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{\theta}|\boldsymbol{y}) = \arg\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) \tag{41}$$

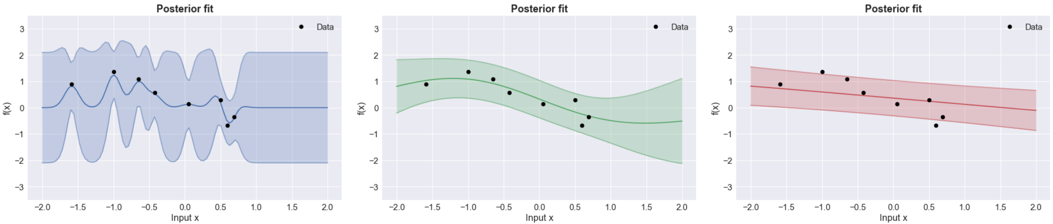
- If the prior $p(\boldsymbol{\theta}) \propto 1$ is uniform

$$\hat{\theta}_{\mathrm{MAP}} = \arg\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) + \ln k = \arg\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \hat{\theta}_{\mathrm{ML}} \tag{42}$$

- This is also sometimes called the maximum likelihood type II estimate

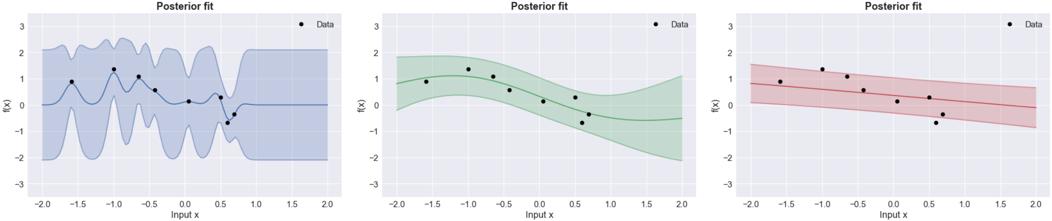
Model complexity for Gaussian processes

- Three GP fits with SE kernels with different lengthscales: 0.1, 1.3, 10
- Which figure correspond to which lengthscale?



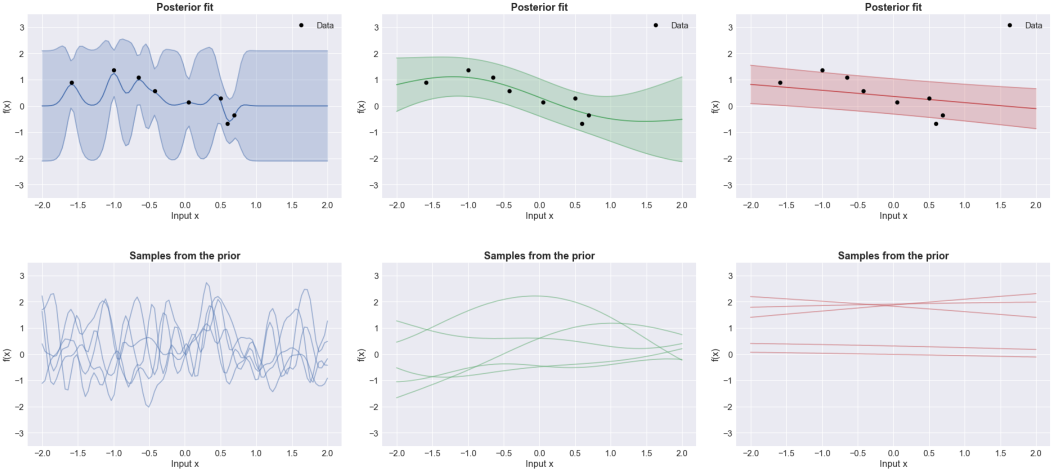
Model complexity for Gaussian processes

- Three GP fits with SE kernels with different lengthscales: 0.1, 1.3, 10
- Which figure correspond to which lengthscale?



Model complexity for Gaussian processes

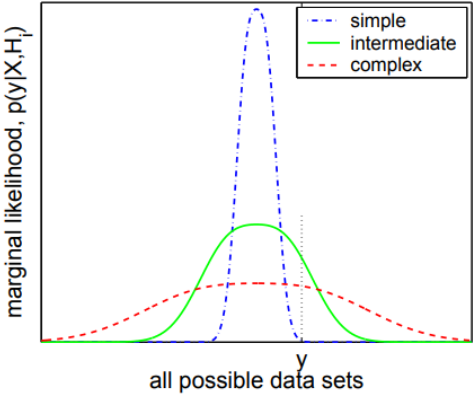
- Three GP fits with SE kernels with different lengthscales: 0.1, 1.3, 10
- Which figure correspond to which lengthscale?



- The lengthscale controls the "effective model complexity"

Marginal likelihood and Occam's razor

- Occam's razor: "When you have two competing models that produce similar predictions, the simpler one is the better"
- Example: If a simple linear model and a complex neural network produce equally good predictions, just we should choose the linear model
- Same concepts goes for Gaussian processes
- The marginal likelihood $p(\boldsymbol{y}|\boldsymbol{\theta})$ implements a version of Occam's razor



The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{f} \tag{43}$$

(45)

(48)

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$
\begin{aligned}
p(\boldsymbol{y} | \boldsymbol{\theta}) &= \int p(\boldsymbol{y} | \boldsymbol{f}) p(\boldsymbol{f} | \boldsymbol{\theta}) \mathrm{d}\boldsymbol{f} && (43) \\
&= \int \mathcal{N}\left(\boldsymbol{y} \middle| \boldsymbol{f}, \sigma_{obs}^2 \boldsymbol{I}\right) \mathcal{N}\left(\boldsymbol{f} \middle| 0, \boldsymbol{K}\right) \mathrm{d}\boldsymbol{f} && (44) \\
& && (45)
\end{aligned}
$$

$$ (48) $$

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{f} \tag{43}$$

$$= \int \mathcal{N}\left(\boldsymbol{y}\middle|\boldsymbol{f}, \sigma_{obs}^2 \boldsymbol{I}\right) \mathcal{N}\left(\boldsymbol{f}\middle|0, \boldsymbol{K}\right) \mathrm{d}\boldsymbol{f} \tag{44}$$

$$= \mathcal{N}\left(\boldsymbol{y}\middle|0, \sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right) \tag{45}$$

$$\tag{48}$$

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{f} \tag{43}$$

$$= \int \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f}, \sigma_{obs}^2 \boldsymbol{I}\right) \mathcal{N}\left(\boldsymbol{f}|0, \boldsymbol{K}\right) \mathrm{d}\boldsymbol{f} \tag{44}$$

$$= \mathcal{N}\left(\boldsymbol{y}|0, \sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right) \tag{45}$$

- Then

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \ln \mathcal{N}\left(\boldsymbol{y}|0, \sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right) \tag{46}$$

(48)

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$
\begin{aligned}
p(\mathbf{y}|\boldsymbol{\theta}) &= \int p(\mathbf{y}|\mathbf{f})p(\mathbf{f}|\boldsymbol{\theta})\mathrm{d}\mathbf{f} && (43)\\
&= \int \mathcal{N}\left(\mathbf{y}\middle|\mathbf{f}, \sigma_{obs}^2 \mathbf{I}\right) \mathcal{N}\left(\mathbf{f}\middle|0, \mathbf{K}\right) \mathrm{d}\mathbf{f} && (44)\\
&= \mathcal{N}\left(\mathbf{y}\middle|0, \sigma_{obs}^2 \mathbf{I} + \mathbf{K}\right) && (45)
\end{aligned}
$$

- Then

$$
\begin{aligned}
\ln p(\mathbf{y}|\boldsymbol{\theta}) &= \ln \mathcal{N}\left(\mathbf{y}\middle|0, \sigma_{obs}^2 \mathbf{I} + \mathbf{K}\right) && (46)\\
&= \ln \left[(2\pi)^{-\frac{N}{2}} \left|\sigma_{obs}^2 \mathbf{I} + \mathbf{K}\right|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}\mathbf{y}^T \left(\sigma_{obs}^2 \mathbf{I} + \mathbf{K}\right)^{-1} \mathbf{y}\right)\right] && (47)\\
& && (48)
\end{aligned}
$$

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$
\begin{aligned}
p(\boldsymbol{y}|\boldsymbol{\theta}) &= \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{f} && (43)\\
&= \int \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f},\sigma_{obs}^2\boldsymbol{I}\right)\mathcal{N}\left(\boldsymbol{f}|0,\boldsymbol{K}\right)\mathrm{d}\boldsymbol{f} && (44)\\
&= \mathcal{N}\left(\boldsymbol{y}|0,\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right) && (45)
\end{aligned}
$$

- Then

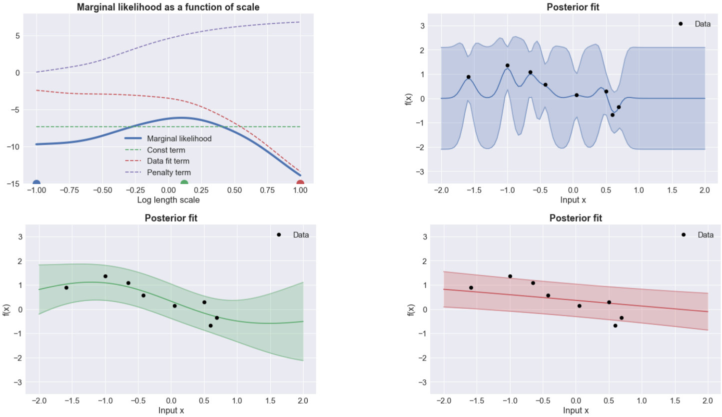
$$
\begin{aligned}
\ln p(\boldsymbol{y}|\boldsymbol{\theta}) &= \ln \mathcal{N}\left(\boldsymbol{y}|0,\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right) && (46)\\
&= \ln\left[(2\pi)^{-\frac{N}{2}}\left|\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right)^{-1}\boldsymbol{y}\right)\right] && (47)\\
&= -\frac{N}{2}\ln(2\pi)-\frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right|-\frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right)^{-1}\boldsymbol{y} && (48)
\end{aligned}
$$

The marginal likelihood computation (II)

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \underbrace{-\frac{N}{2}\ln(2\pi)}_{\text{Constant}} \underbrace{-\frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right|}_{\text{Complexity penalty}} \underbrace{-\frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}}_{\text{Data fit}} \quad (49)$$

The marginal likelihood computation (II)

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \underbrace{-\frac{N}{2}\ln(2\pi)}_{\text{Constant}} \underbrace{-\frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right|}_{\text{Complexity penalty}} \underbrace{-\frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I}+\boldsymbol{K}\right)^{-1}\boldsymbol{y}}_{\text{Data fit}} \quad (49)$$



MLL multimodality

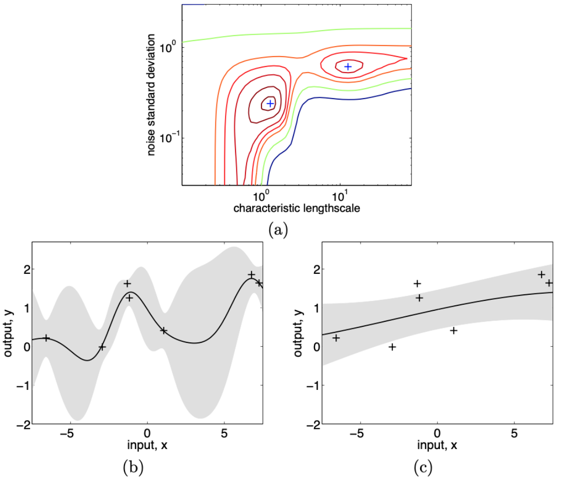


Figure 5.5: Panel (a) shows the marginal likelihood as a function of the hyperparameters ℓ (length-scale) and σ_n^2 (noise standard deviation), where $\sigma_f^2 = 1$ (signal standard deviation) for a data set of 7 observations (seen in panels (b) and (c)). There are two local optima, indicated with '+': the global optimum has low noise and a short length-scale; the local optimum has a high noise and a long length scale. In (b) and (c) the inferred underlying functions (and 95% confidence intervals) are shown for each of the two solutions. In fact, the data points were generated by a Gaussian process with $(\ell, \sigma_f^2, \sigma_n^2) = (1, 1, 0.1)$ in eq. (5.1).

The marginal likelihood computation (III)

- Log marginal likelihood for Gaussian likelihood

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{50}$$

- Optimize $p(\boldsymbol{y}|\boldsymbol{\theta})$ wrt. $\boldsymbol{\theta}$ using gradient based methods

$$\nabla_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) \tag{51}$$

- Modern ML libraries (Torch, TensorFlow) have autodiff. The gradient has to be derived for non-autodiff softare (numpy, Matlab)

- We can also use $p(\boldsymbol{y}|\boldsymbol{\theta})$ to compare the quality of the fit for two different kernels

- No need for cross-validation using this approach!

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\mathbf{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\mathbf{I} + \mathbf{K}\right| - \frac{1}{2}\mathbf{y}^T\left(\sigma_{obs}^2\mathbf{I} + \mathbf{K}\right)^{-1}\mathbf{y} \tag{52}$$

- In numpy: $\left|0.1\mathbf{I}_{400\times400}\right| = 0.0$, but $\ln\left|0.1\mathbf{I}_{400\times400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\mathbf{C} = \sigma_{obs}^2\mathbf{I} + \mathbf{K}$ such that $\mathbf{C} = \mathbf{L}\mathbf{L}^T$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $|0.1\boldsymbol{I}_{400\times 400}| = 0.0$, but $\ln|0.1\boldsymbol{I}_{400\times 400}| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln|\boldsymbol{C}| = \ln|\boldsymbol{L}\boldsymbol{L}^T| = \ln|\boldsymbol{L}|\cdot|\boldsymbol{L}^T| = \ln|\boldsymbol{L}|^2 = 2\ln|\boldsymbol{L}| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \tag{54}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} \tag{54}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y} \tag{54}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma^2_{obs}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y} = \left(\boldsymbol{L}^{-1}\boldsymbol{y}\right)^T\underbrace{\left(\boldsymbol{L}^{-1}\boldsymbol{y}\right)}_{=\boldsymbol{v}} = \boldsymbol{v}^T\boldsymbol{v} \tag{54}$$

The marginal likelihood computation (IV)

- In practice, we should avoiding computing determinants and inverses!

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma_{obs}^2 \boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^T\left(\sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{52}$$

- In numpy: $\left|0.1\boldsymbol{I}_{400\times 400}\right| = 0.0$, but $\ln\left|0.1\boldsymbol{I}_{400\times 400}\right| = -2302.58$ and $\exp(-2302.58) > 0$
- Step 1: Compute cholesky factorization of $\boldsymbol{C} = \sigma_{obs}^2\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^T$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^T\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{53}$$

- Step 3: Compute quadractic term as follows

$$\boldsymbol{y}^T\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^T\left(\boldsymbol{L}\boldsymbol{L}^T\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^T\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y} = (\boldsymbol{L}^{-1}\boldsymbol{y})^T\underbrace{(\boldsymbol{L}^{-1}\boldsymbol{y})}_{=\boldsymbol{v}} = \boldsymbol{v}^T\boldsymbol{v} \tag{54}$$

- Step 4: Sum components

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} - \frac{1}{2}\boldsymbol{v}^T\boldsymbol{v} \tag{55}$$

- Note that we never compute the determinant or the inverse of $\boldsymbol{C}$ directly!

Two metrics for model evaluation

- Assume we are given a training set $\{\boldsymbol{x}_n, y_n\}_{n=1}^{N}$ and now we want to evaluate our model using an independent test set $\left\{\boldsymbol{x}_p^*, \boldsymbol{y}_p^*\right\}_{p=1}^{P}$
- Let μ_{p*}, σ_{p*}^2 be the predictive mean and variance, respectively, of the test point $\left(\boldsymbol{x}_p^*, y_p^*\right)$
- The mean square error metric (does not take uncertainty into account)

$$\mathrm{MSE} = \frac{1}{P}\sum_{p=1}^{P}\left(\mu_{p*} - y_p^*\right)^2 \tag{56}$$

- The (pointwise) mean log posterior predictive density (MLPPD) is given by

$$\mathrm{MLPPD} = \frac{1}{P}\sum_{i=1}^{P}\ln\mathcal{N}\left(y_p^*\middle|\mu_{p*}, \sigma_{p*}^2\right) \tag{57}$$

Computational complexity of Gaussian Processes

- The key equations for predictions

$$p(f_*|\mathbf{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{58}$$
$$\mu_* = \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{y} \tag{59}$$
$$\sigma_*^2 = K_{f_* f_*} - \mathbf{k}_{f_* f}\left(\mathbf{K}_{ff} + \sigma_{obs}^2 \mathbf{I}\right)^{-1} \mathbf{k}_{f_* f}^T \tag{60}$$

- Recall: If $\mathbf{A} \in \mathbb{R}^{N \times M}$ and $\mathbf{b} \in \mathbb{R}^M$, then the cost of computing $\mathbf{Ab}$ is $\mathcal{O}(NM)$
- Recall: If $\mathbf{C} \in \mathbb{R}^{N \times N}$, then the cost of computing $\mathbf{C}^{-1}$ is $\mathcal{O}\left(N^3\right)$
- What is computational complexity for computing the posterior distribution for 1 test point based on a data set with N observations? What is the dominating operation?
- What about the memory footprint?

Next time

Thursday, we'll talk about

- How to speed up GP inference
- Inducing points and sparse Gaussian process
- Non-Gaussian likelihoods (classification)

Read:

- "Scalable Variational Gaussian Process Classification" by Hensman et al `https://arxiv.org/abs/1411.2005`

Assignments

- Assignment # 1 deadline wednesday midday
- Assignment session # 1 at wednesday and friday 12:15
- Assignment # 2 is online today
- After the assignment # 2, you should be able to
  1. Implement the squared exponential kernel and explain the interpretation of each parameter
  2. Generate samples from a Gaussian process prior
  3. Compute the posterior & predictive distributions for a Gaussian process model with Gaussian likelihood
  4. Compute the marginal likelihood and use it for model selection
- Deadline: wednesday the 27th of January