CS-E4895 Gaussian Processes
Lecture 3: Gaussian process regression

Ti John

Aalto University

Monday 6.3.2023

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

Bonus task: find the mistakes

Last session

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 processes 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 at any point $\mathbf{x} \in \mathcal{X}$

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

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

Last time (II)

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

$$p(\boldsymbol{y}, \boldsymbol{f}, f_*) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}, f_*) = \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f}, \sigma^2_{\text{obs}}\boldsymbol{I}\right)\mathcal{N}\left(\begin{bmatrix}\boldsymbol{f}\\ f_*\end{bmatrix}\Big|\boldsymbol{0}, \begin{bmatrix}\boldsymbol{K}_{ff} & \boldsymbol{k}_{ff_*}\\ \boldsymbol{k}_{f_*f} & k_{f_*f_*}\end{bmatrix}\right) \tag{5}$$

where

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

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

- $\boldsymbol{k}_{f_*f}$ is the covariance vector between test input and training inputs

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

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

$$k_{f_*f_*} = \operatorname{cov}\left(f(\boldsymbol{x}_*), f(\boldsymbol{x}_*)\right) \tag{8}$$

Last time (III)

- Step 1: Write the joint model

$$p(\boldsymbol{y}, \boldsymbol{f}, f_*) = p(\boldsymbol{y} | \boldsymbol{f}) p(\boldsymbol{f}, f_*) = \mathcal{N}\left(\boldsymbol{y} | \boldsymbol{f}, \sigma^2_{\text{obs}} \boldsymbol{I}\right) \mathcal{N}\left(\begin{bmatrix} \boldsymbol{f} \\ f_* \end{bmatrix} \middle| \boldsymbol{0}, \begin{bmatrix} \boldsymbol{K}_{ff} & \boldsymbol{k}_{ff_*} \\ \boldsymbol{k}_{f_*f} & k_{f_*f_*} \end{bmatrix}\right) \tag{9}$$

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

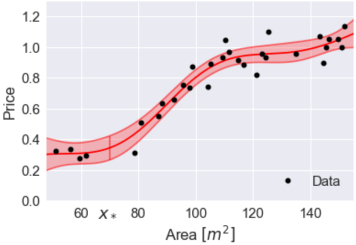
$$p(\boldsymbol{y}, f_*) = \int p(\boldsymbol{y} | \boldsymbol{f}) p(\boldsymbol{f}, f_*) \mathrm{d}\boldsymbol{f} = \mathcal{N}\left(\begin{bmatrix} \boldsymbol{y} \\ f_* \end{bmatrix} \middle| \boldsymbol{0}, \begin{bmatrix} \boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I} & \boldsymbol{k}_{ff_*} \\ \boldsymbol{k}_{f_*f} & k_{f_*f_*} \end{bmatrix}\right) \tag{10}$$

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

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

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

$$\sigma^2_* = k_{f_*f_*} - \boldsymbol{k}_{f_*f} \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I}\right)^{-1} \boldsymbol{k}^\top_{f_*f} \tag{13}$$



Non-zero prior mean function

- Step 1: Write the joint model

$$p(\boldsymbol{y}, \boldsymbol{f}, f_*) = p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}, f_*) = \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f}, \sigma^2_{\text{obs}}\boldsymbol{I}\right)\mathcal{N}\left(\begin{bmatrix}\boldsymbol{f}\\ f_*\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{m}\\ m_*\end{bmatrix}, \begin{bmatrix}\boldsymbol{K}_{ff} & \boldsymbol{k}_{ff_*}\\ \boldsymbol{k}_{f_*f} & k_{f_*f_*}\end{bmatrix}\right) \tag{14}$$

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

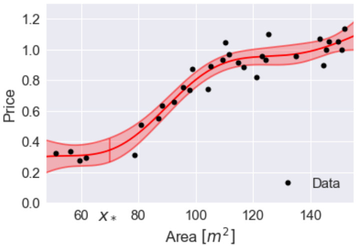
$$p(\boldsymbol{y}, f_*) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}, f_*)\mathrm{d}\boldsymbol{f} = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{y}\\ f_*\end{bmatrix} \middle| \begin{bmatrix}\boldsymbol{m}\\ m_*\end{bmatrix}, \begin{bmatrix}\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}}\boldsymbol{I} & \boldsymbol{k}_{ff_*}\\ \boldsymbol{k}_{f_*f} & k_{f_*f_*}\end{bmatrix}\right) \tag{15}$$

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

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_*|\mu_*, \sigma^2_*\right) \tag{16}$$

$$\mu_* = \boldsymbol{k}_{f_*f}\left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}}\boldsymbol{I}\right)^{-1}(\boldsymbol{y} - \boldsymbol{m}) + m_* \tag{17}$$

$$\sigma^2_* = k_{f_*f_*} - \boldsymbol{k}_{f_*f}\left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}}\boldsymbol{I}\right)^{-1}\boldsymbol{k}^\top_{f_*f} \tag{18}$$



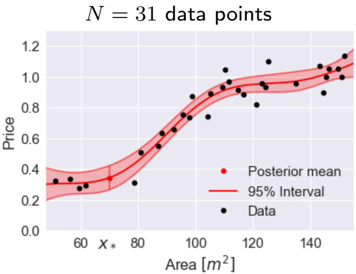
Example: The components of the posterior distribution I

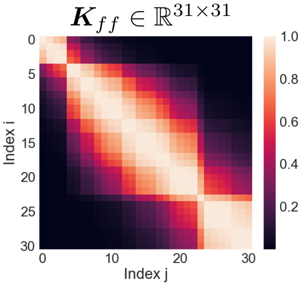
$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{19}$$

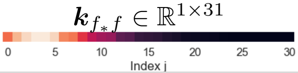
$$\mu_* = \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{\text{obs}}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{20}$$

$$\sigma_*^2 = k_{f_* f_*} - \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{\text{obs}}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^\top \tag{21}$$

- Predict $f_* \equiv f(x_*)$ for test input $x_* = 70$
- Observation vector $\boldsymbol{y} = [y_1, y_2, \ldots, y_{31}]^\top \in \mathbb{R}^{31\times 1}$
- Gaussian kernel $k(x, x') = k(f(x), f(x')) = \exp\left[-\frac{(x-x')^2}{2\cdot 20^2}\right]$
- Cov. matrix of training: $[\boldsymbol{K}_{ff}]_{ij} = k(x_i, x_j)$
- Cov. between test and training $[\boldsymbol{k}_{f_* f}]_j = k(x_*, x_j)$
- Covariance (here: variance) of $f(x_*)$: $k_{f_* f_*} = k(x_*, x_*)$

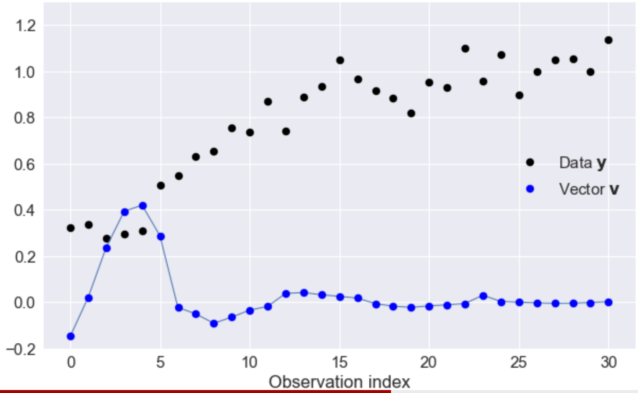




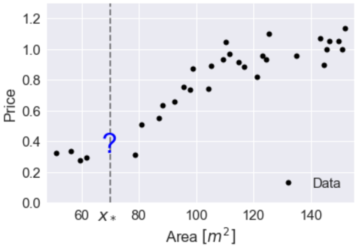


Example: The components of the posterior distribution II

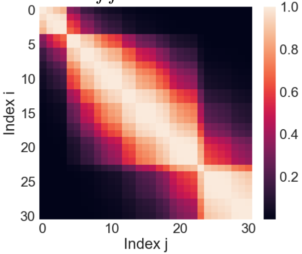
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I}\right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^\top = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I}\right)^{-1} \in \mathbb{R}^{1\times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^\top \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



$N = 31$ data points



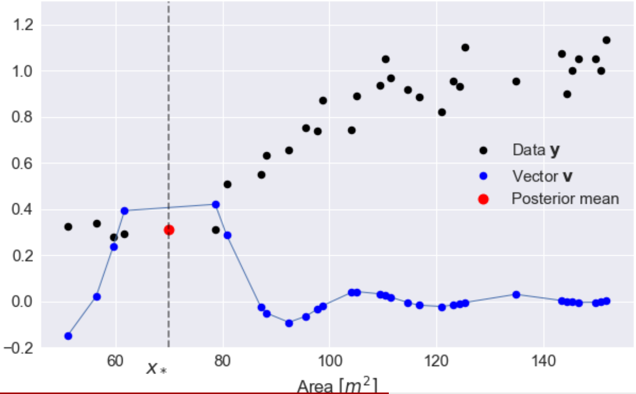
$\boldsymbol{K}_{ff} \in \mathbb{R}^{31\times 31}$



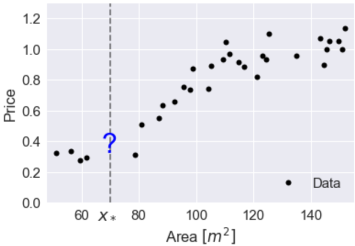
$\boldsymbol{k}_{f_* f} \in \mathbb{R}^{1\times 31}$

Example: The components of the posterior distribution II

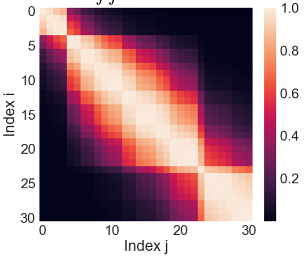
- $\mu_* = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I} \right)^{-1} \boldsymbol{y}$
- Let's define $\boldsymbol{v}^\top = \boldsymbol{k}_{f_* f} \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I} \right)^{-1} \in \mathbb{R}^{1 \times 31}$
- The posterior mean is a linear combination of the observations $\mu_* = \boldsymbol{v}^\top \boldsymbol{y} = \sum_{i=1}^{31} v_i y_i$



$N = 31$ data points



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



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

Quiz

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) \tag{22}$$

$$\mu_* = \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{\text{obs}}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{y} \tag{23}$$

$$\sigma_*^2 = k_{f_* f_*} - \boldsymbol{k}_{f_* f}\left(\boldsymbol{K}_{ff} + \sigma_{\text{obs}}^2 \boldsymbol{I}\right)^{-1} \boldsymbol{k}_{f_* f}^\top \tag{24}$$

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

$$k(x, x') = \exp\left[-\frac{(x - x')^2}{2 \cdot 20^2}\right] \qquad k_2(x, x') = \exp\left[-\frac{(x - x')^2}{2 \cdot 40^2}\right] \tag{25}$$

3. What is the difference between σ_{obs}^2 and σ_*^2?
4. What is the difference between $p(f_*|\boldsymbol{y})$ and $p(y_*|\boldsymbol{y})$?

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

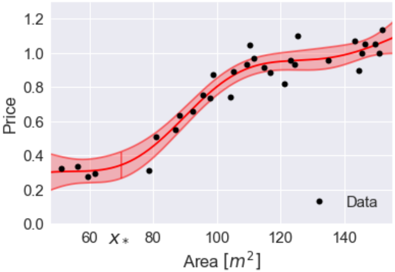
- The model is given by: $y_n = f(x_n) + \epsilon_n$
- The posterior of the function evaluated at x_*:

$$p(f_*|\boldsymbol{y}) = \mathcal{N}\left(f_*\middle|\mu_*, \sigma_*^2\right) \tag{26}$$

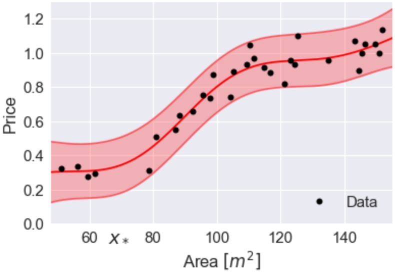
- The predictive distribution of y_*:

$$p(y_*|\boldsymbol{y}) = \int p(y_*|f_*)p(f_*|\boldsymbol{y})\mathrm{d}f_* \tag{27}$$

$p(f_*|\boldsymbol{y})$



$p(y_*|\boldsymbol{y})$



Covariance functions

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

$$\boldsymbol{K}_{ij} = \operatorname{cov}\left(f(\boldsymbol{x}_i), f(\boldsymbol{x}_j)\right) = k\left(\boldsymbol{x}_i, \boldsymbol{x}_j\right) \tag{28}$$

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

$$\text{(Symmetric)} \quad \boldsymbol{K} = \boldsymbol{K}^\top \tag{29}$$

$$\text{(PSD)} \quad \forall \boldsymbol{x} \neq 0 : \quad \boldsymbol{x}^\top \boldsymbol{K} \boldsymbol{x} \geq 0 \tag{30}$$

  PD matrices are 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 $k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right)$ only depends on the difference of the inputs

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k\left(\boldsymbol{x}_1 - \boldsymbol{x}_2\right), \qquad \text{or} \qquad k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\boldsymbol{x}_1 + \boldsymbol{a}, \boldsymbol{x}_2 + \boldsymbol{a}) \tag{31}$$

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

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k\left(\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|\right) \tag{32}$$

- Quiz: Which of the following kernels are stationary? isotropic?

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \boldsymbol{x}_1^\top \boldsymbol{x}_2 \tag{linear}$$

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2}\right) \tag{squared exponential 1}$$

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

Addendum: Properties of prior vs. posterior

- The posterior of a Gaussian process regression is just another Gaussian process, with mean function $\mu_*(x)$ and covariance function $k_*(x, x')$

$$
\begin{aligned}
\mu_*(x) &= \boldsymbol{k}_{f_* f}(x) \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I} \right)^{-1} \boldsymbol{y} \\
k_*(x, x') &= k(x, x') - \boldsymbol{k}_{f_* f}(x) \left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}} \boldsymbol{I} \right)^{-1} \boldsymbol{k}_{f_* f}(x')^\top \\
[\boldsymbol{k}_{f_* f}(x)]_j &= k(x, x_j)
\end{aligned}
$$

- Note: a stationary **prior** does not imply that the **posterior** is stationary!
- Just like the posterior mean can be non-zero even with a zero-mean prior

- Interactive GP visualization: `http://www.infinitecuriosity.org/vizgp/`
  Play around with different kernels, kernel combinations, hyperparameters...

Table of common covariance functions

From the book (ch. 4.2.3)

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	$\frac{1}{2^{\nu-1}\Gamma(\nu)} \Big(\frac{\sqrt{2\nu}}{\ell} r\Big)^{\nu} K_\nu \Big(\frac{\sqrt{2\nu}}{\ell} r\Big)$	$\surd$	$\surd$
exponential	$\exp(-\frac{r}{\ell})$	$\surd$	$\surd$
γ-exponential	$\exp\big(-(\frac{r}{\ell})^\gamma\big)$	$\surd$	$\surd$
rational quadratic	$(1 + \frac{r^2}{2\alpha\ell^2})^{-\alpha}$	$\surd$	$\surd$
neural network	$\sin^{-1}\Big(\frac{2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}}'}{\sqrt{(1+2\tilde{\mathbf{x}}^\top \Sigma \tilde{\mathbf{x}})(1+2\tilde{\mathbf{x}}'^\top \Sigma \tilde{\mathbf{x}}')}}\Big)$		$\surd$

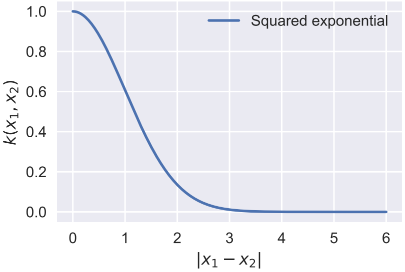
(S = stationary, ND = non-degenerate)

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/exponentiated quadratic/radial basis function/RBF) covariance function

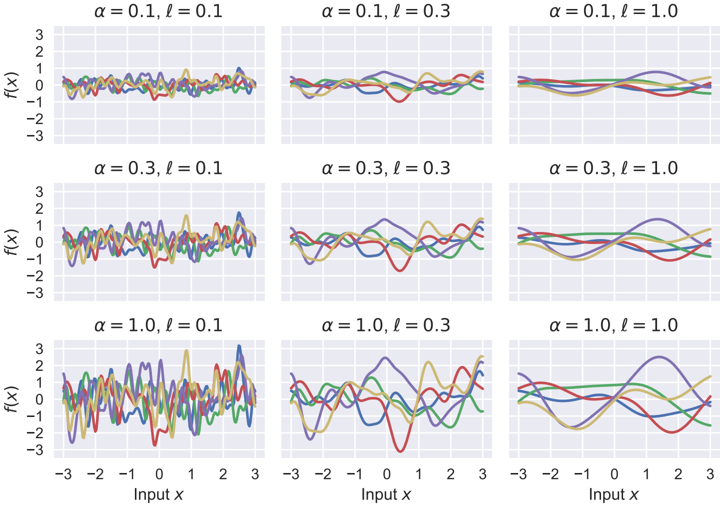
$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = k(\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|) = \alpha \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\ell^2}\right) \quad (33)$$



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

The squared exponential covariance function (II)

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \alpha \exp\left(-\frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\ell^2}\right) \tag{34}$$

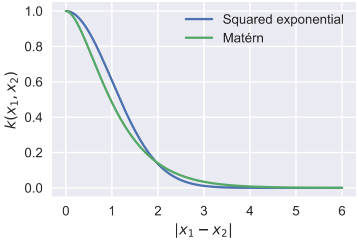


The Matérn covariance function (I)

- Matérn class covariance function

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \alpha \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|}{\ell}\right)^{\nu} K_\nu \left(\sqrt{2\nu} \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|}{\ell}\right) \tag{35}$$

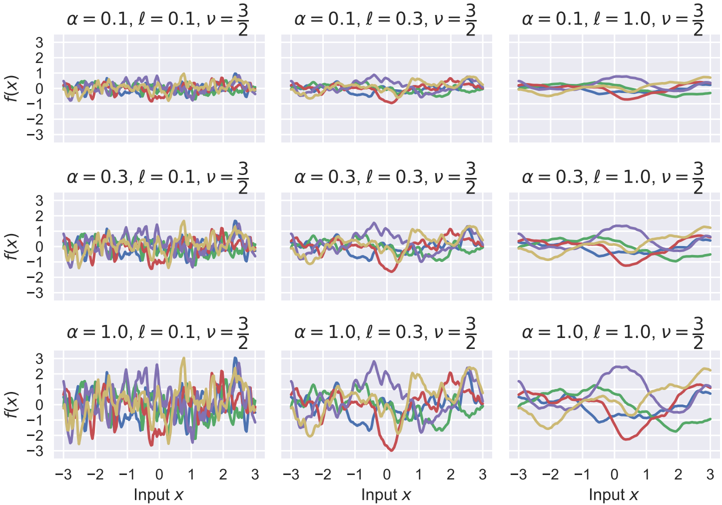
where K_ν is a modified Bessel function.



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

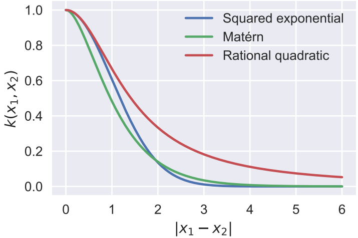
The Matérn covariance function (II)

$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \alpha \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu} \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|}{\ell} \right)^{\nu} K_{\nu} \left(\sqrt{2\nu} \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|}{\ell} \right) \tag{36}$$



Rational Quadratic (I)

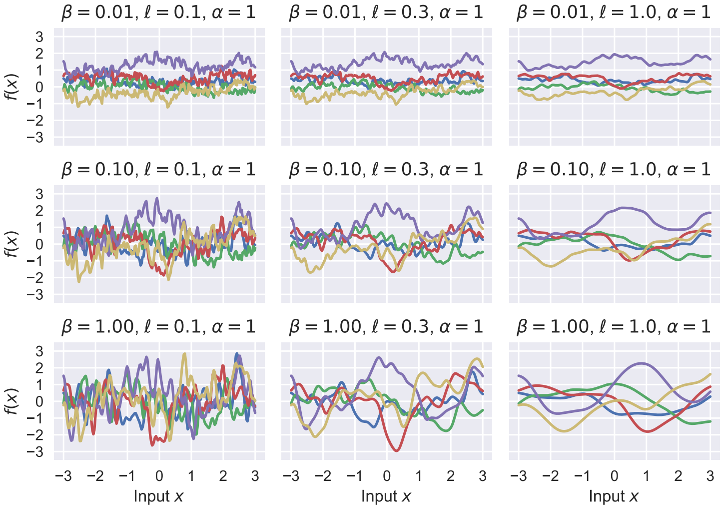
$$k(\boldsymbol{x}_1, \boldsymbol{x}_2) = \alpha \left(1 + \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\beta\ell^2}\right)^{-\beta} \tag{37}$$



- 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, image analysis, etc.)

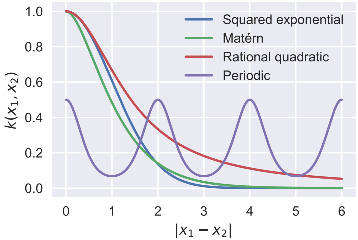
Rational Quadratic (II)

$$k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \alpha\left(1 + \frac{\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\beta\ell^2}\right)^{-\beta} \tag{38}$$

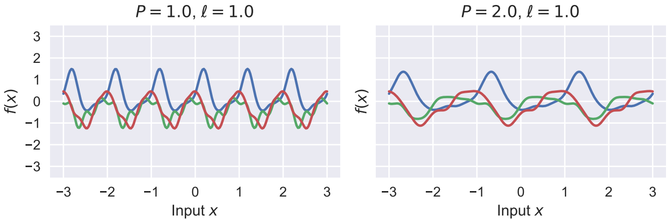


Covariance function for periodic functions

$$k(x_1, x_2) = \alpha \exp\left(-\frac{2}{\ell^2}\sin^2\left(\frac{\pi|x_1 - x_2|}{P}\right)\right) \quad (39)$$



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



Building new kernels from old ones (I)

Requirements for valid kernels:

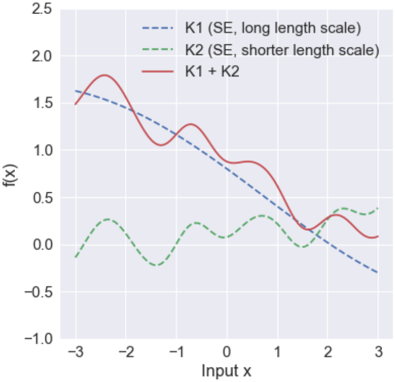
$$\text{(Symmetric)} \quad \boldsymbol{K} = \boldsymbol{K}^\top \tag{40}$$

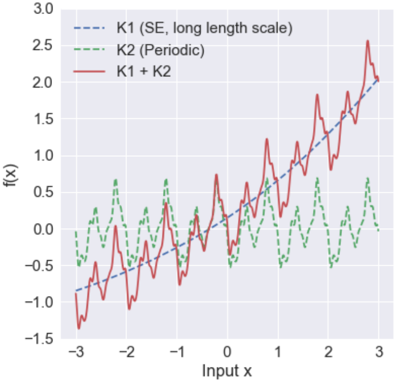
$$\text{(PSD)} \quad \forall \boldsymbol{x} \neq 0 : \quad \boldsymbol{x}^\top \boldsymbol{K} \boldsymbol{x} \geq 0 \tag{41}$$

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 $a(\boldsymbol{x})$: $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = a(\boldsymbol{x}_1) k_1(\boldsymbol{x}_1, \boldsymbol{x}_2)\, a(\boldsymbol{x}_2)$
(for arbitrary $a(\boldsymbol{x})$)

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 periodic 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:

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

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.

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

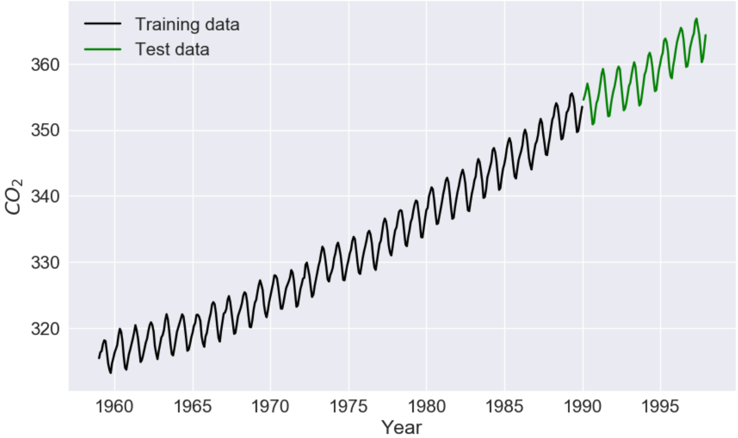
$$
k\left(\boldsymbol{x}_1, \boldsymbol{x}_2\right) = \exp\left(-\frac{\left\|\boldsymbol{x}_1 - \boldsymbol{x}_2\right\|^2}{2}\right) \qquad (42)
$$

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

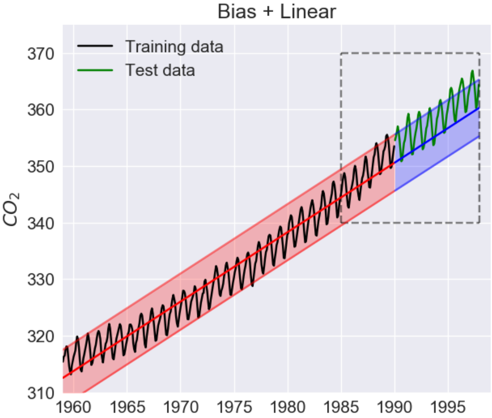
From Chris Bishop's book: `https://www.microsoft.com/en-us/research/people/cmbishop`

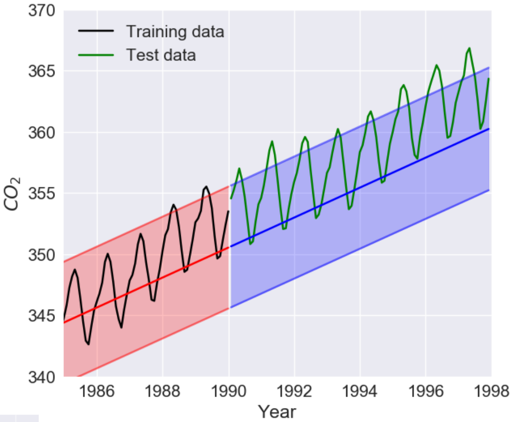
Example: Mauna Loa data set

- 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

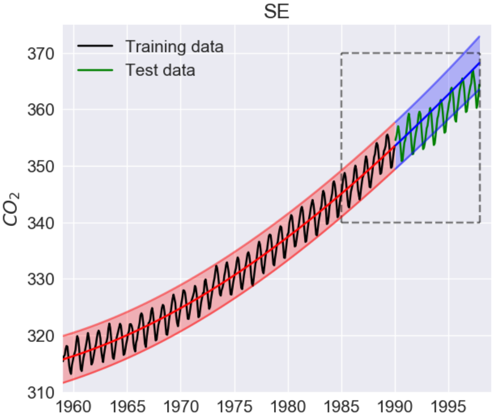


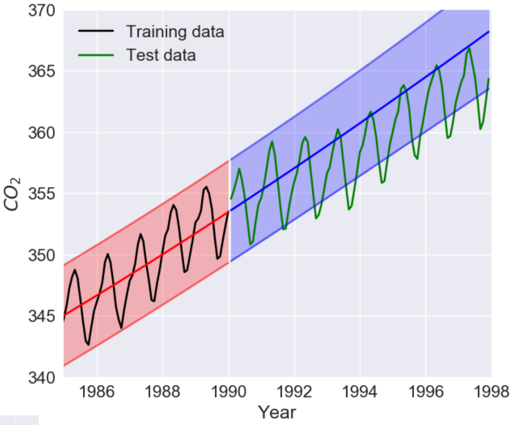
Example: Mauna Loa data set



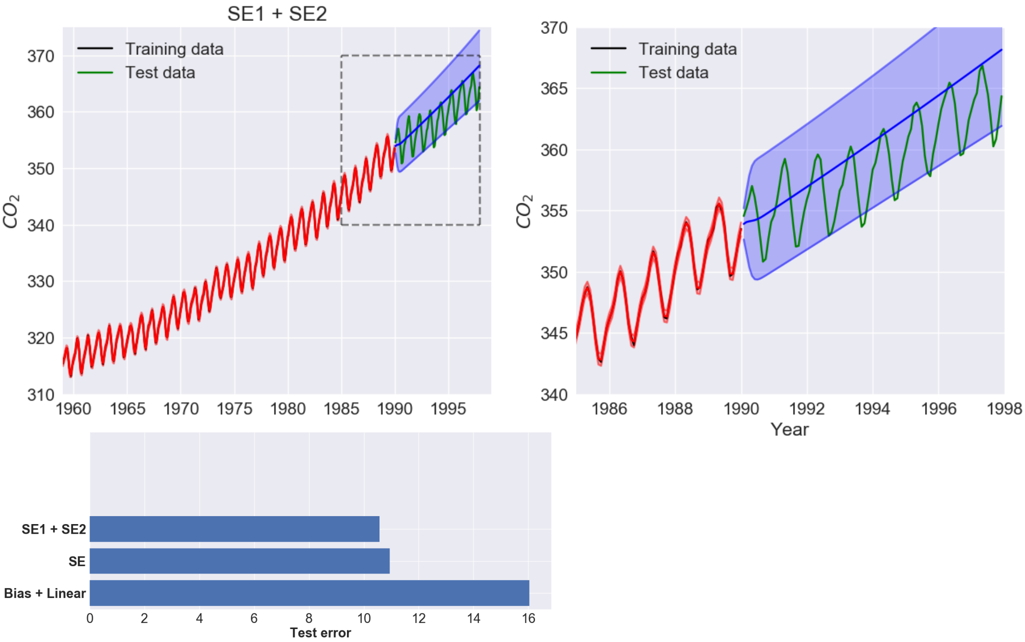


Example: Mauna Loa data set

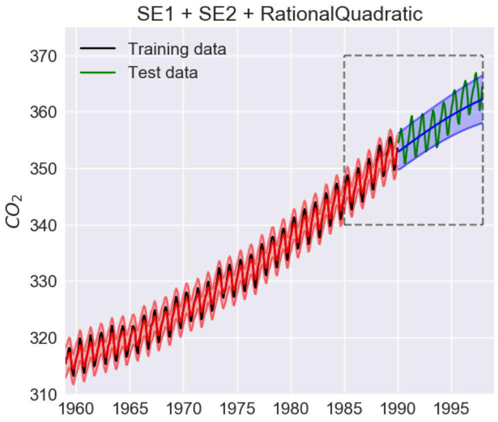


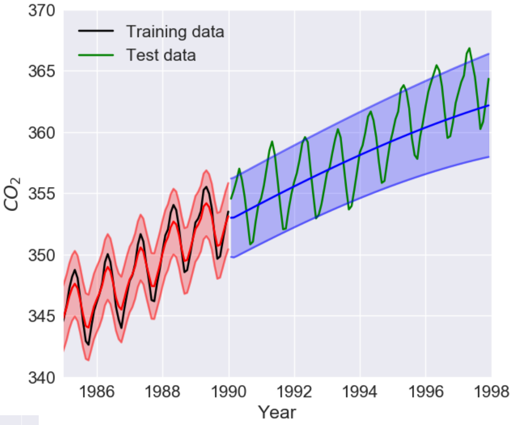


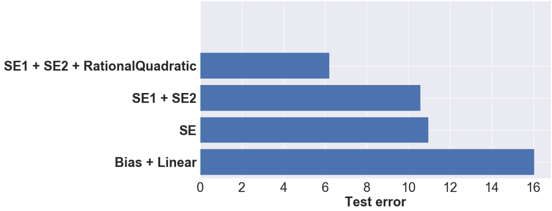
Example: Mauna Loa data set



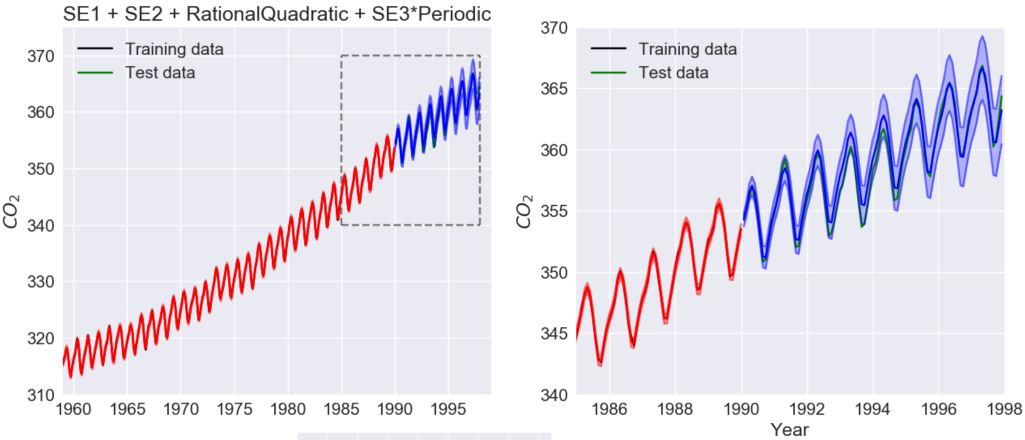
Example: Mauna Loa data set

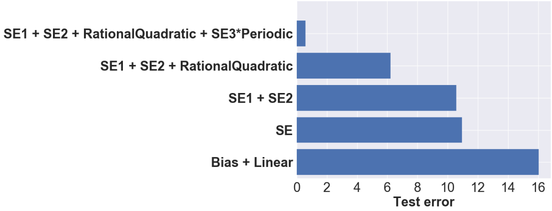






Example: Mauna Loa data set





Section 3

Model selection

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

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y})} \tag{43}$$

but in this case the marginal likelihood is almost always intractable

$$p(\boldsymbol{y}) = \int p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})\mathrm{d}\boldsymbol{\theta} \tag{44}$$

Hyperparameters & model selection (II)

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

$$p(\boldsymbol{\theta}|\boldsymbol{y}) = \frac{p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta})}{p(\boldsymbol{y})} \propto p(\boldsymbol{y}|\boldsymbol{\theta})p(\boldsymbol{\theta}) \tag{45}$$

- The MAP estimate is defined as

$$\hat{\theta}_{\mathsf{MAP}} = \mathrm{arg}\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{\theta}|\boldsymbol{y}) = \mathrm{arg}\max_{\boldsymbol{\theta}} \big(\ln p(\boldsymbol{y}|\boldsymbol{\theta}) + \ln p(\boldsymbol{\theta}) \big) \tag{46}$$

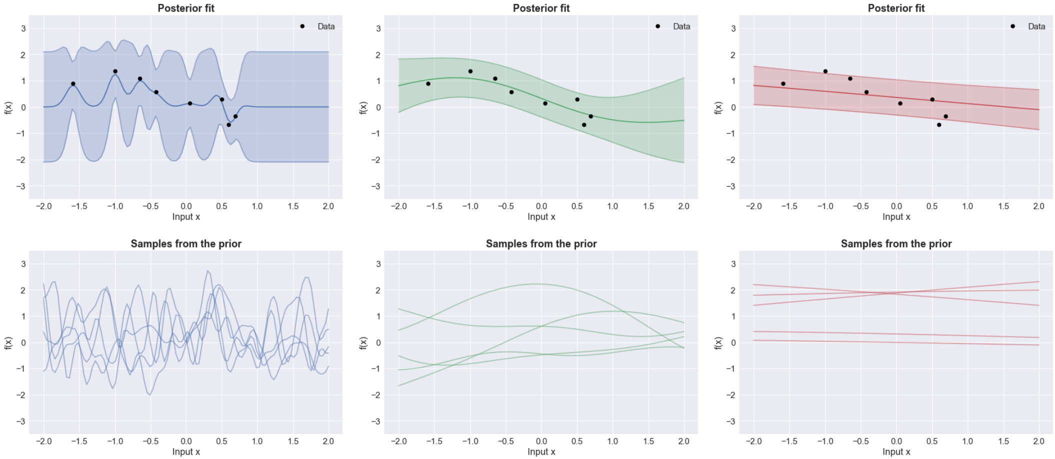
- If the prior $p(\boldsymbol{\theta}) \propto 1$ is uniform

$$\hat{\theta}_{\mathsf{MAP}} = \mathrm{arg}\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) + \ln k = \mathrm{arg}\max_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \hat{\theta}_{\mathsf{ML}} \tag{47}$$

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

Model complexity for Gaussian processes

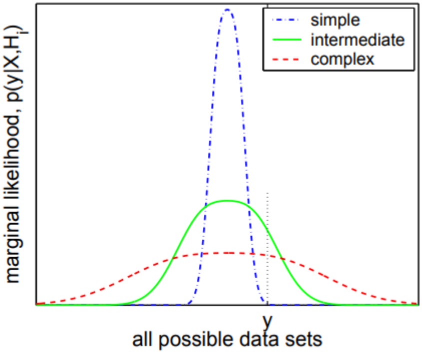
- Three GP fits with SE kernels with different lengthscales: 0.1, 1.3, 10
- Which figure corresponds to which lengthscale?



- 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, we should just choose the linear model
- Same concept goes for Gaussian processes
- The marginal likelihood $p(\boldsymbol{y}|\boldsymbol{\theta})$ implements a version of Occam's razor



(figure from the book)

The marginal likelihood computation (I)

- Marginal likelihood for Gaussian likelihood

$$p(\boldsymbol{y}|\boldsymbol{\theta}) = \int p(\boldsymbol{y}|\boldsymbol{f})p(\boldsymbol{f}|\boldsymbol{\theta})\mathrm{d}\boldsymbol{f} \tag{48}$$

$$= \int \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{f}, \sigma^2_{\mathrm{obs}}\boldsymbol{I}\right)\mathcal{N}\left(\boldsymbol{f}|\boldsymbol{0}, \boldsymbol{K}\right)\mathrm{d}\boldsymbol{f} \tag{49}$$

$$= \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{0}, \sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right) \tag{50}$$

- Then

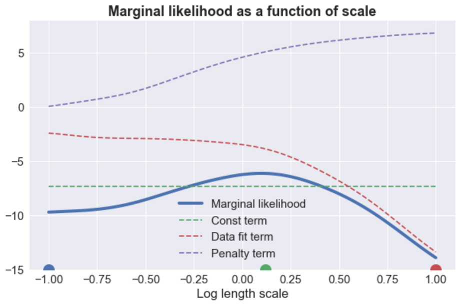
$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \ln \mathcal{N}\left(\boldsymbol{y}|\boldsymbol{0}, \sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right) \tag{51}$$

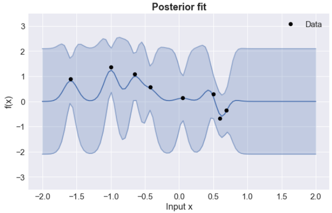
$$= \ln\left[(2\pi)^{-\frac{N}{2}}\left|\sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right|^{-\frac{1}{2}}\exp\left(-\frac{1}{2}\boldsymbol{y}^\top\left(\sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y}\right)\right] \tag{52}$$

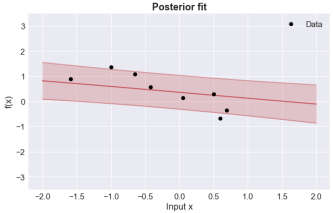
$$= -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^\top\left(\sigma^2_{\mathrm{obs}}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{53}$$

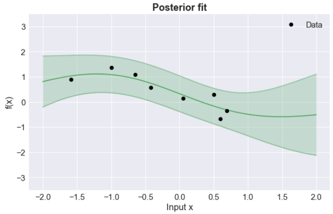
The marginal likelihood computation (II)

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = \underbrace{-\frac{N}{2}\ln(2\pi)}_{\text{Constant}} \underbrace{-\frac{1}{2}\ln\left|\sigma^2_{\text{obs}}\boldsymbol{I}+\boldsymbol{K}\right|}_{\text{Complexity penalty}} \underbrace{-\frac{1}{2}\boldsymbol{y}^\top\left(\sigma^2_{\text{obs}}\boldsymbol{I}+\boldsymbol{K}\right)^{-1}\boldsymbol{y}}_{\text{Data fit}} \tag{54}$$









Multimodality of the marginal likelihood

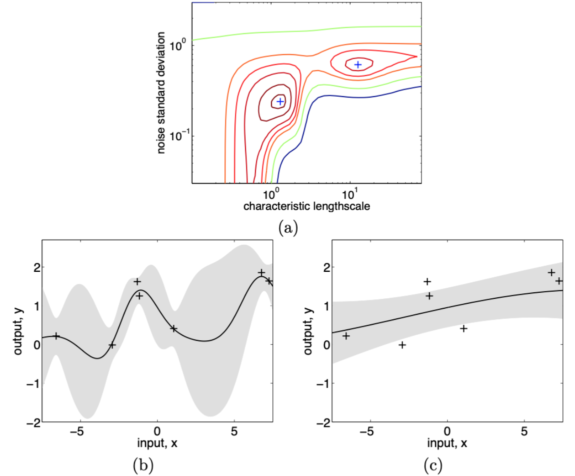


Figure 5.5: Panel (a) shows the marginal likelihood as a function of the hyperparameters ℓ (length-scale) and σ_n^2 (noise standard deviation), where $\sigma_f^2 = 1$ (signal standard deviation) for a data set of 7 observations (seen in panels (b) and (c)). There are two local optima, indicated with '+': 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

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{\text{obs}}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^\top\left(\sigma^2_{\text{obs}}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{55}$$

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

$$\nabla_{\boldsymbol{\theta}} \ln p(\boldsymbol{y}|\boldsymbol{\theta}) \tag{56}$$

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

- We can also use $p(\boldsymbol{y}|\boldsymbol{\theta})$ to compare the quality of the fit for two different kernels
  (caveat: different numbers of hyperparameters ⇒ BIC, AIC, . . .)

- No need for cross-validation using this approach!

$$p(\boldsymbol{y}) = p(y_1)p(y_2|y_1)p(y_3|y_1,y_2)\cdots p(y_N|y_1,\ldots,y_{N-1}) \tag{57}$$

The marginal likelihood computation (IV)

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

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}\ln\left|\sigma^2_{\text{obs}}\boldsymbol{I} + \boldsymbol{K}\right| - \frac{1}{2}\boldsymbol{y}^\top\left(\sigma^2_{\text{obs}}\boldsymbol{I} + \boldsymbol{K}\right)^{-1}\boldsymbol{y} \tag{58}$$

- In numpy: $\det(0.1\boldsymbol{I}_{400\times 400}) = 0.0$, but $\log\det(0.1\boldsymbol{I}_{400\times 400}) \approx -921.0$
- Step 1: Compute Cholesky factorization of $\boldsymbol{C} = \sigma^2_{\text{obs}}\boldsymbol{I} + \boldsymbol{K}$ such that $\boldsymbol{C} = \boldsymbol{L}\boldsymbol{L}^\top$
- Step 2: Compute the log determinant term as follows

$$\ln\left|\boldsymbol{C}\right| = \ln\left|\boldsymbol{L}\boldsymbol{L}^\top\right| = \ln\left|\boldsymbol{L}\right|\cdot\left|\boldsymbol{L}^\top\right| = \ln\left|\boldsymbol{L}\right|^2 = 2\ln\left|\boldsymbol{L}\right| = 2\ln\prod_{n=1}^{N}\boldsymbol{L}_{nn} = 2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} \tag{59}$$

- Step 3: Compute quadratic term as follows

$$\boldsymbol{y}^\top\boldsymbol{C}^{-1}\boldsymbol{y} = \boldsymbol{y}^\top\left(\boldsymbol{L}\boldsymbol{L}^\top\right)^{-1}\boldsymbol{y} = \boldsymbol{y}^\top\boldsymbol{L}^{-T}\boldsymbol{L}^{-1}\boldsymbol{y} = (\boldsymbol{L}^{-1}\boldsymbol{y})^\top\underbrace{(\boldsymbol{L}^{-1}\boldsymbol{y})}_{=\boldsymbol{v}} = \boldsymbol{v}^\top\boldsymbol{v} \tag{60}$$

- Step 4: Sum components

$$\ln p(\boldsymbol{y}|\boldsymbol{\theta}) = -\frac{N}{2}\ln(2\pi) - \frac{1}{2}2\sum_{n=1}^{N}\ln\boldsymbol{L}_{nn} - \frac{1}{2}\boldsymbol{v}^\top\boldsymbol{v} \tag{61}$$

- 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 $\{\boldsymbol{x}_p^*, y_p^*\}_{p=1}^P$
- 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)

$$\text{MSE} = \frac{1}{P} \sum_{p=1}^{P} \left(\mu_{p*} - y_p^*\right)^2 \tag{62}$$

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

$$\text{MLPPD} = \frac{1}{P} \sum_{p=1}^{P} \ln \mathcal{N}\left(y_p^* \middle| \mu_{p*}, \sigma_{p*}^2\right) \tag{63}$$

  - Sometimes called simply negative log likelihood (NLL)
  - Sometimes called negative log predictive density (NLPD)

Section 4

Computational complexity

Computational complexity of Gaussian Processes

- The key equations for predictions

$$
\begin{aligned}
p(f_*|\boldsymbol{y}) &= \mathcal{N}\left(f_* \middle| \mu_*, \sigma_*^2\right) && (64)\\
\mu_* &= \boldsymbol{k}_{f_*f}\left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}}\boldsymbol{I}\right)^{-1}\boldsymbol{y} && (65)\\
\sigma_*^2 &= K_{f_*f_*} - \boldsymbol{k}_{f_*f}\left(\boldsymbol{K}_{ff} + \sigma^2_{\text{obs}}\boldsymbol{I}\right)^{-1}\boldsymbol{k}_{f_*f}^\top && (66)
\end{aligned}
$$

- Recall: If $\boldsymbol{A} \in \mathbb{R}^{N\times M}$ and $\boldsymbol{b} \in \mathbb{R}^{M}$, then the cost of computing $\boldsymbol{Ab}$ is $\mathcal{O}\left(NM\right)$

- Recall: If $\boldsymbol{C} \in \mathbb{R}^{N\times N}$, then the cost of computing $\boldsymbol{C}^{-1}$ 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?

Key takeaways

- Gaussian process regression
- Covariance functions
  - properties: must be symmetric and PSD
  - what is stationary/isotropic
  - common kernels, their properties & parameters
  - kernel combinations
- Model selection
  - marginal likelihood
  - MAP/ML-II for hyperparameter point estimates
  - "model complexity" vs. data fit
  - multi-modality of marginal likelihood surface
  - how to evaluate numerically stably
- Computational complexity
  - time: $\mathcal{O}(N^3)$, memory: $\mathcal{O}(N^2)$

Next time

Tomorrow, we'll talk about

- Integration and model selection
- Practical examples

Assignments

Note: lecture slide had some mistakes, please see below for up-to-date information

- Assignment #1: deadline end of Wednesday 8th March
  - Complete and return via JupyterHub. Instructions available on MyCourses.
- Assignment Q&A sessions on Thursday 10:15
  - Participating will grant points towards final grade (2 points).
- Assignment #2 is online on Wednesday 8th March.
- After the assignment #2, you should be able to
  1. Implement the squared exponential kernel and explain the interpretation of each parameter.
  2. Compute the marginal likelihood and use it for model selection.
- Assignment #2: deadline end of Wednesday 15th of March.