Advanced Bayesian Learning

Gaussian Process Regression and Classification - Lecture 1

Mattias Villani

Department of Statistics
Stockholm University

Department of Computer and Information Science
Linköping University

Course overview

Four topics

- **Gaussian Process Regression and Classification**
- **Bayesian Nonparametrics**
- **Variational Inference**
- **Bayesian Model Inference**

Examination

- Individual Lab/Exercise for each topic
- Deadline for submission: day before new topic starts.
- Extra deadline for all four topics: Sept 15, 2020.

Topic overview

- Gaussian Process Regression
- Gaussian Process Classification

Nonlinear regression

- **Linear regression**

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

$$f(\mathbf{x}) = \mathbf{x}^T \boldsymbol{w}$$

and $\epsilon \sim N(0, \sigma_n^2)$ and iid over observations.

- **Polynomial regression**: $\phi(\mathbf{x}) = (1, x, x^2, x^3, ..., x^k)$:

$$f(\mathbf{x}) = \phi(\mathbf{x})^T \boldsymbol{w}\cdot$$

- More generally: **splines** with **basis functions**.
- Example: **thin plate splines** with knots $\kappa_1, ..., \kappa_N$ in $\mathbf{x}$-space

$$\phi_k(\mathbf{x}) = \ln\left(\|\mathbf{x} - \kappa_k\|\right) \|\mathbf{x} - \kappa_k\|^2$$

Recap: Bayesian linear regression

- Prior

$$\mathbf{w} \sim N\left(0, \Sigma_p\right)$$

- Posterior [$\mathbf{X}$ is $D \times n$]

$$\mathbf{w}|\mathbf{X},\mathbf{y} \sim N\left(\bar{\mathbf{w}}, \mathbf{A}^{-1}\right)$$

$$\mathbf{A} = \sigma_n^{-2}\mathbf{X}\mathbf{X}^T + \Sigma_p^{-1}$$

$$\bar{\mathbf{w}} = \sigma_n^{-2}\mathbf{A}^{-1}\mathbf{X}\mathbf{y} = \left(\mathbf{X}\mathbf{X}^T + \sigma_n^2\Sigma_p^{-1}\right)^{-1}\mathbf{X}\mathbf{y}$$

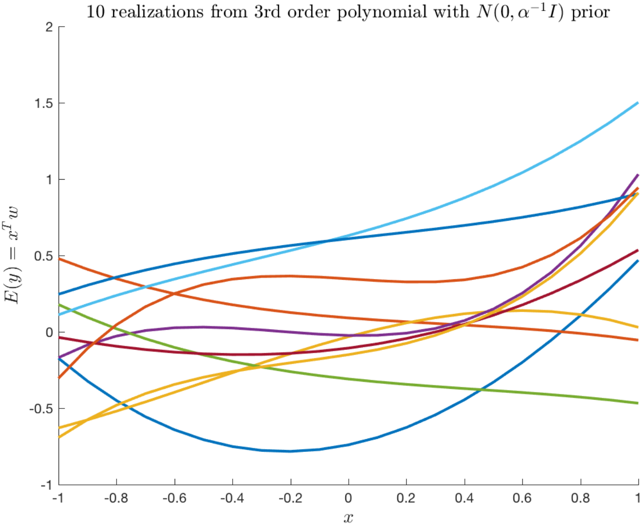
- Predictive density for mean $f(\mathbf{x}_*) = \mathbf{x}_*^T\mathbf{w}$ at new $\mathbf{x}_*$

$$f(\mathbf{x}_*)|\mathbf{x}_*, \mathbf{X}, \mathbf{y} \sim N\left(\mathbf{x}_*^T\bar{\mathbf{w}}, \mathbf{x}_*^T\mathbf{A}^{-1}\mathbf{x}_*\right)$$

- Predictive density for new response y_*

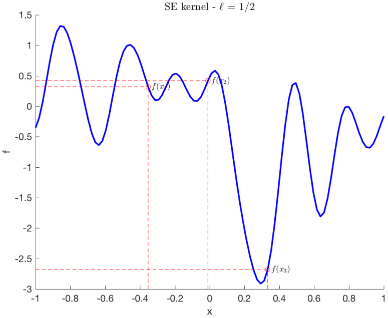
$$\mathbf{y}_*|\mathbf{x}_*, \mathbf{X}, \mathbf{y} \sim N\left(\mathbf{x}_*^T\bar{\mathbf{w}}, \mathbf{x}_*^T\mathbf{A}^{-1}\mathbf{x}_* + \sigma_n^2\right)$$

A prior on w is really a prior over functions



Non-parametric regression

- **Non-parametric**: avoid a parametric form for $f(\cdot)$.
- Treat $f(\mathbf{x})$ as **an unknown parameter for every x**.



- A *new* parameter for *every* $\mathbf{x}$!
- Instead of restricting to linear, impose **"prior smoothness"**.

Two views on GPs

- **Weight space view**
  - Restrict attention to a grid of x-values: $x_1, .., x_k$.
  - Put a joint prior on the **vector of k function values**

$$f(x_1), ..., f(x_k)$$

- **Function space view**
  - Treat **f as an unknown function**.
  - Put a prior over a set of functions (thank you, Kolmogorov!)

Gaussian process and its kernel

- A GP implies:

$$\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_k) \end{pmatrix} \sim N(\mathbf{m}, \mathbf{K})$$

- But how do we specify the $k \times k$ covariance matrix $\mathbf{K}$?

$$Cov\left(f(x_p), f(x_q)\right)$$

- Squared exponential covariance function

$$Cov\left(f(x_p), f(x_q)\right) = k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{x_p - x_q}{\ell}\right)^2\right)$$

- Nearby x's have highly correlated function ordinates $f(x)$.
- We can compute $Cov\left(f(x_p), f(x_q)\right)$ for *any* x_p and x_q.

Gaussian processes

Definition

A **Gaussian process** (**GP**) is a collection of random variables, any finite number of which have a multivariate Gaussian distribution.

- A GP is a **probability distribution over functions**.
- A GP is specified by a **mean** and a **covariance function**

$$m(x) = \mathrm{E}\left[f(x)\right]$$

$$k(x, x') = E\left[\left(f(x) - m(x)\right)\left(f(x') - m(x')\right)\right]$$

  for any two inputs x and x'.
- A **Gaussian process** is denoted by

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

- $f(x) \sim \mathrm{GP}$ encodes **prior beliefs** about the unknown $f(\cdot)$.

Gaussian processes

- Let $r = \|x - x'\|$.
- **Squared exponential (SE)** kernel ($\ell > 0$, $\sigma_f > 0$)

$$K_{SE}(r) = \sigma_f^2 \exp\left(-\frac{r^2}{2\ell^2}\right)$$

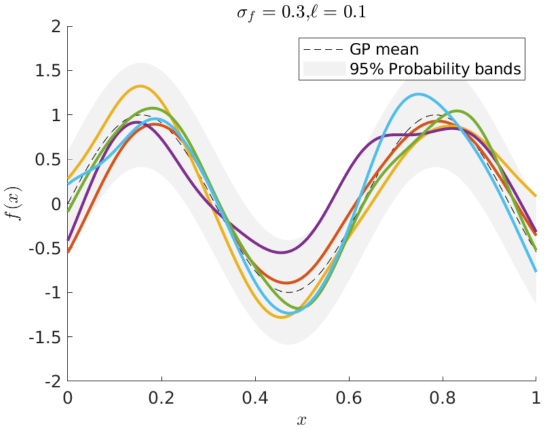
- **Matérn** kernel ($\ell > 0$, $\sigma_f > 0$, $\nu > 0$)

$$K_{Matern}(r) = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu} K_\nu\left(\frac{\sqrt{2\nu}r}{\ell}\right)$$

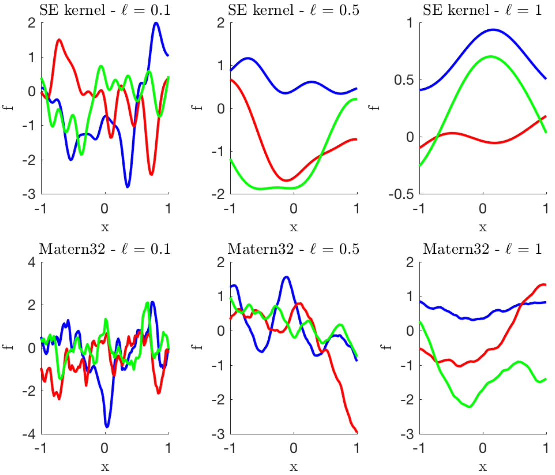
- **Simulate draw** from $f(x) \sim \mathrm{GP}(m(x), k(x, x'))$ by:
  - form a grid $\mathbf{x}_* = (x_1, ..., x_n)$
  - simulate function values from multivariate normal:

$$f(\mathbf{x}_*) \sim N(m(\mathbf{x}_*), K(\mathbf{x}_*, \mathbf{x}_*))$$

Simulating a GP

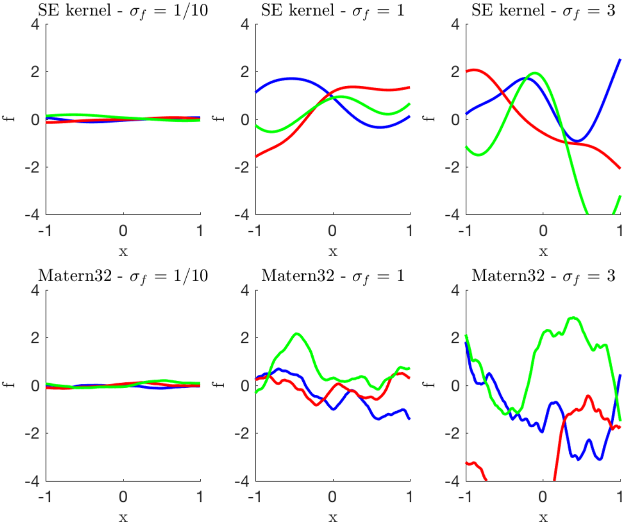


The length scale ℓ - the correlation distance

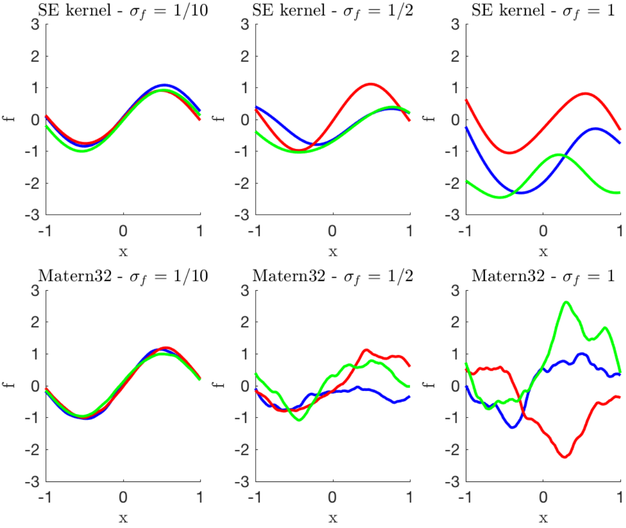


- SE: expected number of zero-crossings on $[0, 1]$: $(2\pi\ell)^{-1}$ (Eq. 4.3)

The scale factor σ_f determines the variance



The mean can be $\sin(3x)$. Or whatever.



Sequential simulation of GPs

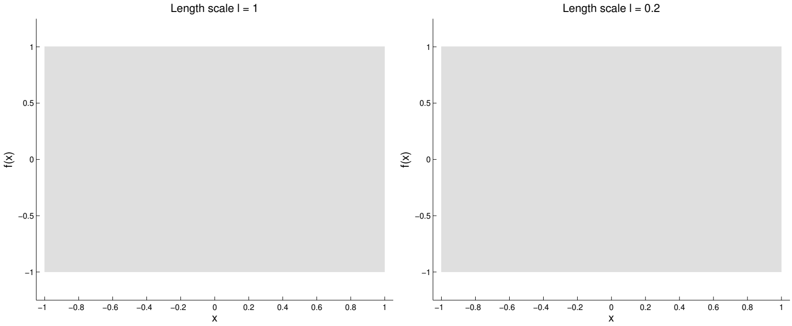
- The joint way: Choose a grid $x_1, ..., x_k$. Simulate the k-vector

$$\begin{pmatrix} f(x_1) \\ \vdots \\ f(x_k) \end{pmatrix} \sim N(\mathbf{m}, \mathbf{K})$$

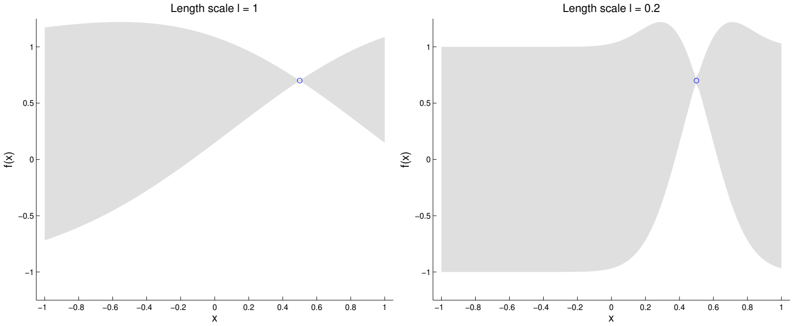
- More intuition from the conditional decomposition

$$\begin{aligned} p(f(x_1), f(x_2), ..., f(x_k)) &= p(f(x_1))\, p(f(x_2)|f(x_1)) \cdots \\ &\times p(f(x_k)|f(x_1), ..., f(x_{k-1})) \end{aligned}$$

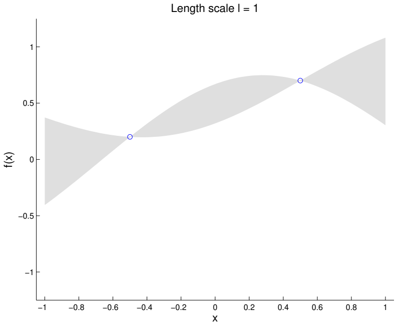
Simulating from $p(f(x_1))$



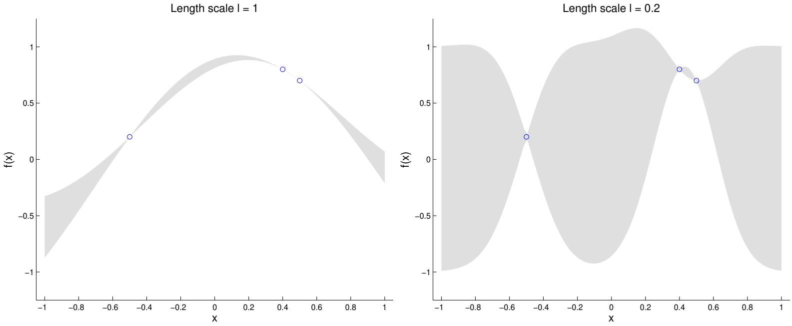
Simulating from $p\left(f\left(x_2\right) \middle| f\left(x_1\right)\right)$



Simulating from $p\left(f\left(x_3\right)\middle|f\left(x_1\right), f\left(x_2\right)\right)$



Simulating from $p\left(f\left(x_4\right) \middle| f\left(x_1\right), f\left(x_2\right), f\left(x_3\right)\right)$



Multivariate normal distribution

- The density of the p-variate $\mathbf{x} \sim N(\mu, \Sigma)$ is

$$f(\mathbf{x}) = \left(\frac{1}{2\pi}\right)^{p/2} \frac{1}{\sqrt{\det \Sigma}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mu)'\Sigma^{-1}(\mathbf{x}-\mu)\right\}$$

- Linear combinations. Let $\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{b}$, then

$$\mathbf{y} \sim N(\mathbf{B}\mu + \mathbf{b}, \mathbf{B}\Sigma\mathbf{B}')$$

- Let $\mathbf{x} = \begin{pmatrix} \mathbf{x}_1 \\ \mathbf{x}_2 \end{pmatrix}$ where $\mathbf{x}_1$ is $p_1 \times 1$ and $\mathbf{x}_2$ is $p_2 \times 1$ and

$$\mu = \begin{pmatrix} \mu_1 \\ \mu_2 \end{pmatrix} \text{ and } \Sigma = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}$$

- Marginals are normal. Let $\mathbf{x} \sim N(\mu, \Sigma)$, then

$$\mathbf{x}_1 \sim N(\mu_1, \Sigma_{11})$$

- Conditionals are normal. Let $\mathbf{x} \sim N(\mu, \Sigma)$, then

$$\mathbf{x}_2 | \mathbf{x}_1 = \mathbf{x}_1^* \sim N\left[\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_1^* - \mu_1),\ \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right]$$

The posterior for a Gaussian Process Regression

- **Model**

$$y_i = f(x_i) + \varepsilon_i, \quad \varepsilon \overset{iid}{\sim} N(0, \sigma_n^2)$$

- **Prior**

$$f(x) \sim GP\left(0, k(x, x')\right)$$

- **Observed**: $\mathbf{x} = (x_1, ..., x_n)^T$ and $\mathbf{y} = (y_1, ..., y_n)^T$.
- **Goal**: posterior of $f(\cdot)$ over test data: $\mathbf{f}_* = \mathbf{f}(\mathbf{x}_*)$.
- **Posterior**

$$\begin{aligned}
\mathbf{f}_* | \mathbf{x}, \mathbf{y}, \mathbf{x}_* &\sim N\left(\bar{\mathbf{f}}_*, \mathrm{cov}(\mathbf{f}_*)\right) \\
\bar{\mathbf{f}}_* &= K(\mathbf{x}_*, \mathbf{x}) \left[K(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I\right]^{-1} \mathbf{y} \\
\mathrm{cov}(\mathbf{f}_*) &= K(\mathbf{x}_*, \mathbf{x}_*) - K(\mathbf{x}_*, \mathbf{x}) \left[K(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I\right]^{-1} K(\mathbf{x}, \mathbf{x}_*)
\end{aligned}$$

- **Predictive distribution** for new test data

$$\mathbf{y}_* | \mathbf{x}, \mathbf{y}, \mathbf{x}_* \sim N\left(\bar{\mathbf{f}}_*, \mathrm{cov}(\mathbf{f}_*) + \sigma_n^2 I\right)$$

Sketch for proof of posterior

- Idea: obtain joint $p(\mathbf{y}, \mathbf{f}_*)$ and then $p(\mathbf{f}_*|\mathbf{y})$ by conditioning.
- **Model**

$$y_i = f(\mathrm{x}_i) + \varepsilon_i, \quad \varepsilon \overset{iid}{\sim} N(0, \sigma_n^2)$$

- **Prior**

$$f(x) \sim GP\left(0, k(x, x')\right)$$

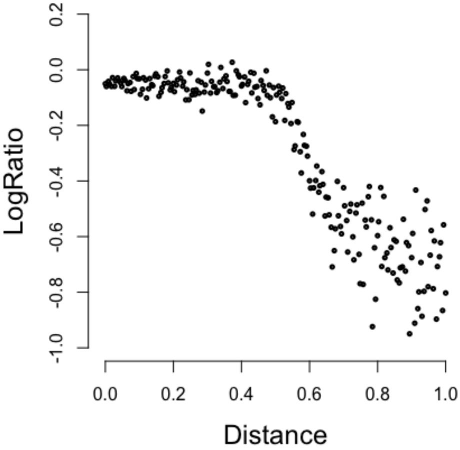
- Joint distribution of $(\mathbf{y}, \mathbf{f}_*)$

$$\begin{pmatrix} \mathbf{y} \\ \mathbf{f}_* \end{pmatrix} \sim \mathrm{N}\left[\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} K(\mathbf{x}, \mathbf{x}) + \sigma_n^2 I & K(\mathbf{x}, \mathbf{x}_*) \\ K(\mathbf{x}_*, \mathbf{x}) & K(\mathbf{x}_*, \mathbf{x}_*) \end{pmatrix}\right]$$

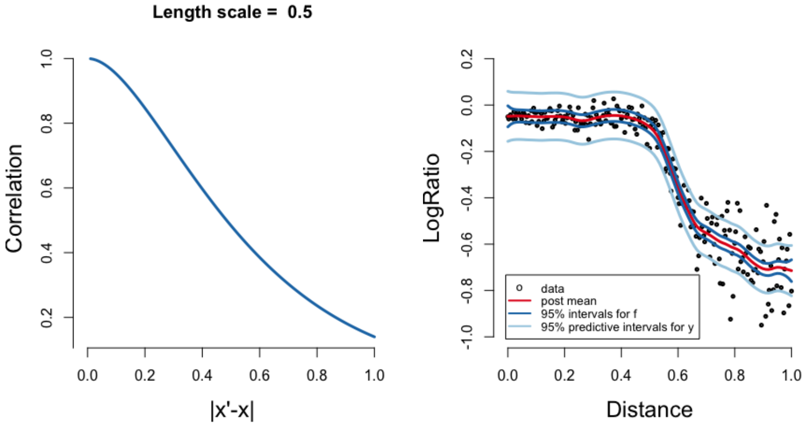
- Complete proof by result:

$$\mathrm{x}_2|\mathrm{x}_1 = \mathrm{x}_1^* \sim N\left[\mu_2 + \Sigma_{21}\Sigma_{11}^{-1}(x_1^* - \mu_1),\ \Sigma_{22} - \Sigma_{21}\Sigma_{11}^{-1}\Sigma_{12}\right]$$

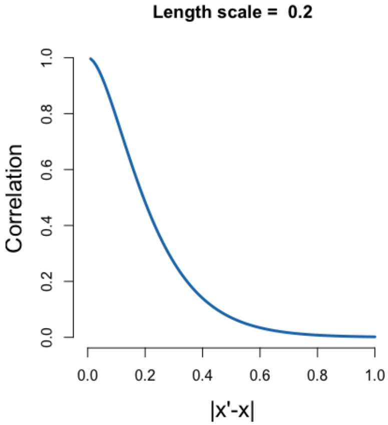
Example - LIDAR data

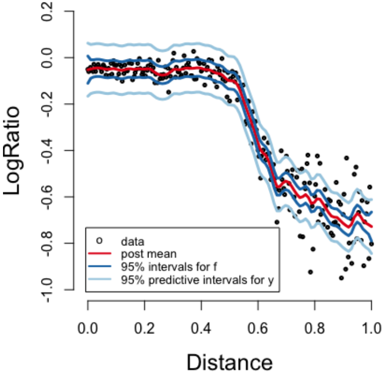


GP fit to LIDAR data $\ell = 0.5, \sigma_f = 0.5, \sigma_n = 0.05$

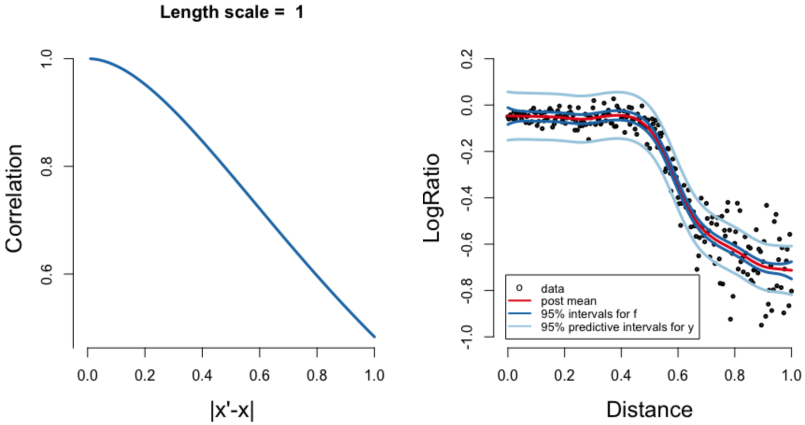


GP fit to LIDAR data $\ell = 0.2, \sigma_f = 0.5, \sigma_n = 0.05$

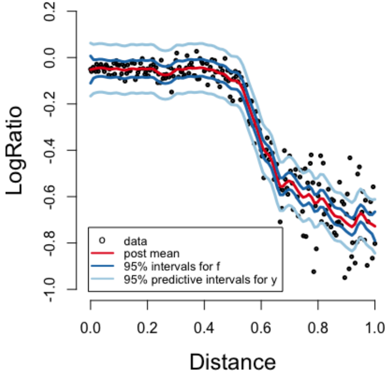


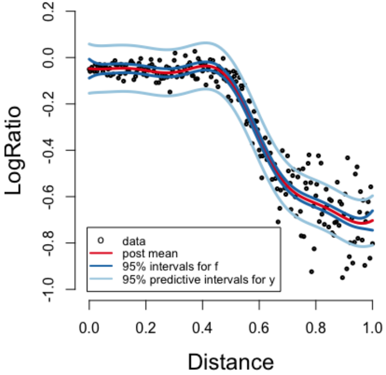


GP fit to LIDAR data $\ell = 1, \sigma_f = 0.5, \sigma_n = 0.05$



Matern32 vs SquaredExp for $\ell = 0.2$

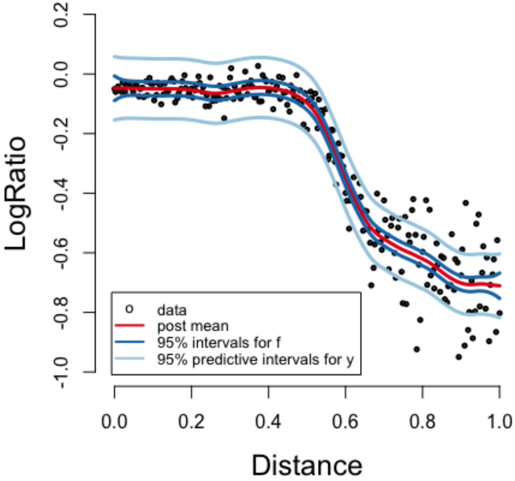




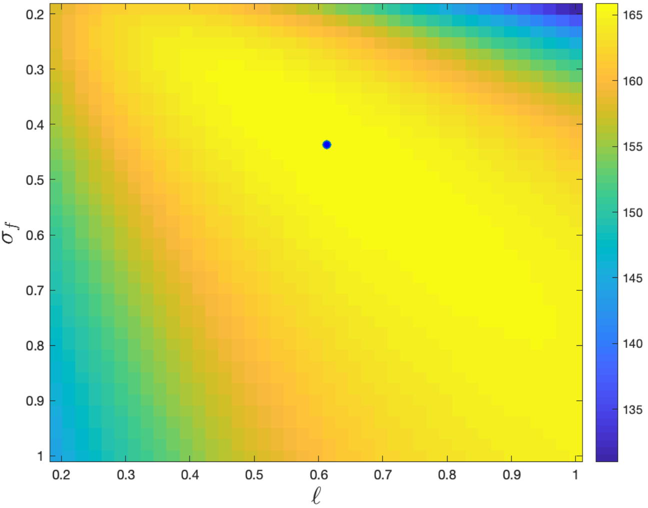
Inference for the hyperparameters

- Kernel depends on **hyperparameters** $\theta = (\sigma_f, \ell)^T$. Example
$$k(\mathbf{x},\mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{\ell^2}\right)$$
- Common: maximize the **marginal likelihood** wrt θ:
$$p(\mathbf{y}|\mathbf{X},\theta) = \int p(\mathbf{y}|\mathbf{X},\mathbf{f},\theta)p(\mathbf{f}|\mathbf{X},\theta)d\mathbf{f}$$
$\mathbf{f} = f(\mathbf{X})$ is a vector of function values in the training data.
- For **Gaussian process regression**: $\boldsymbol{y}|\mathbf{X},\theta \sim N(\mathbf{0}, K + \sigma_n^2 I)$ so
$$\log p(\mathbf{y}|\mathbf{X},\theta) = -\frac{1}{2}\mathbf{y}^T\left(K+\sigma_n^2 I\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|K+\sigma_n^2 I\right| - \frac{n}{2}\log(2\pi)$$
- Proper **Bayesian inference** **for hyperparameters** (HMC?)
$$p(\theta|\mathbf{y},\mathbf{X}) \propto p(\mathbf{y}|\mathbf{X},\theta)p(\theta).$$
- Choice of kernel family by Bayesian model inference. For kernel $K_i \in \mathcal{K}$: $p(K_i|\mathbf{y},\mathbf{X}) \propto p(\mathbf{y}|\mathbf{X},K_i)p(K_i)$

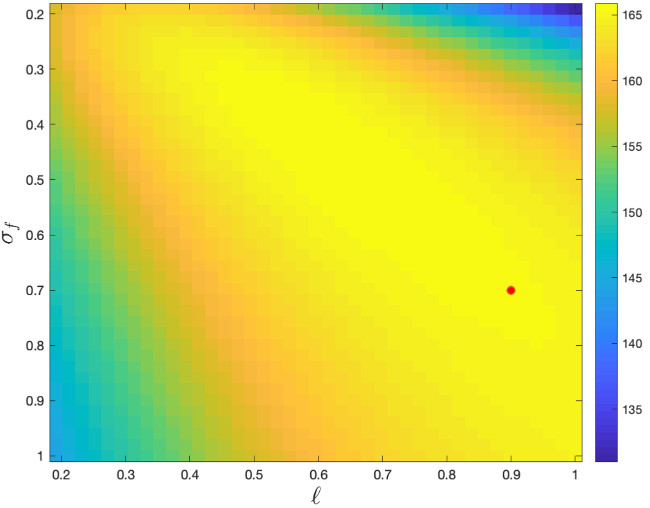
GP fit LIDAR $\ell_{opt} = 0.61, \sigma_{f,opt} = 0.44, \sigma_n = 0.05$



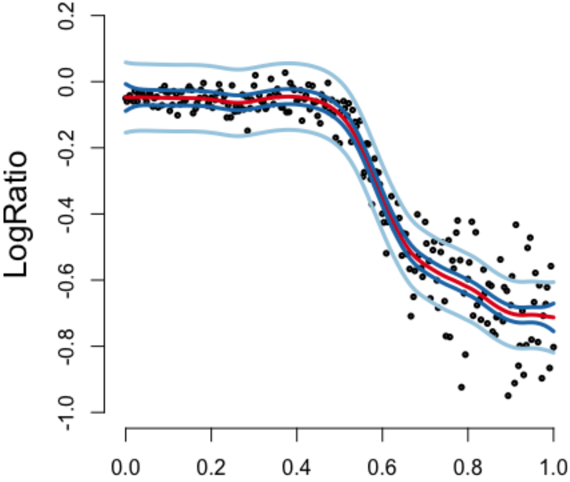
log marginal likelihood surface $\sigma_n = 0.05$



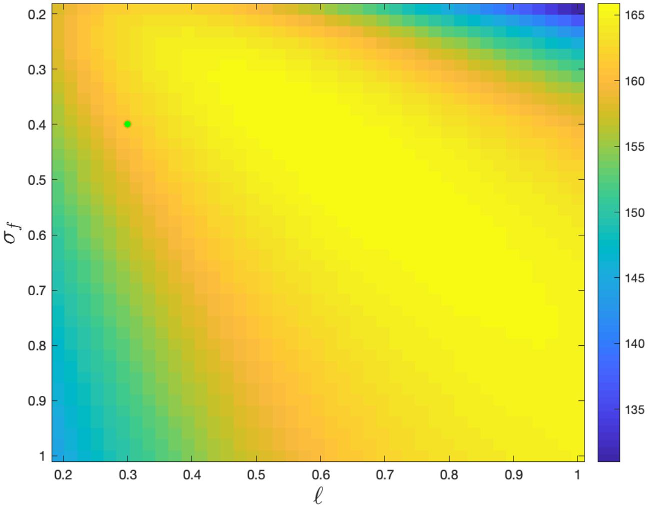
log marginal likelihood surface $\sigma_n = 0.05$



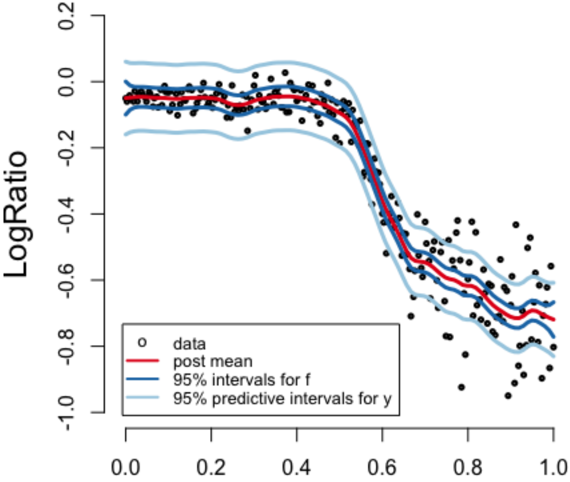
GP fit to LIDAR data $\ell = 0.9, \sigma_f = 0.70, \sigma_n = 0.05$



log marginal likelihood surface $\sigma_n = 0.05$



GP fit to LIDAR data $\ell = 0.3, \sigma_f = 0.4, \sigma_n = 0.05$



GP computations

- Covariance matrix K often numerically singular.
- Noise helps: $K + \sigma_n^2 I$.
- Artificial **jittering** $K + \epsilon I$ for small ϵ.
- Algorithm 2.1 and 3.1 in GPML for stable computations.
- We need to compute:
  - $\left(K + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$ (posterior)
  - $\boldsymbol{y}^T \left(K + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$ (log marginal likelihood)
  - $\left|\log K + \sigma_n^2 I\right|$ (log marginal likelihood)
- $\left(K + \sigma_n^2 I\right)^{-1} \boldsymbol{y}$ corresponds to solving $\left(K + \sigma_n^2 I\right) \boldsymbol{x} = \boldsymbol{y}$ wrt $\boldsymbol{x}$.

Cholesky + Forward and Backward substitution

- Solving $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}$ by $\boldsymbol{x} = \boldsymbol{A}^{-1}\boldsymbol{b}$ is numerically unstable.
- Cholesky factorization $\boldsymbol{A} = \boldsymbol{L}\boldsymbol{L}^T$ where $\boldsymbol{L}$ is lower triangular.
- Let $\boldsymbol{A}\boldsymbol{x} = \boldsymbol{L}\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{L}\boldsymbol{z} = \boldsymbol{b}$ if we define $\boldsymbol{z} = \boldsymbol{L}^T\boldsymbol{x}$.
- Solve $\boldsymbol{L}\boldsymbol{z} = \boldsymbol{b}$ wrt $\boldsymbol{z}$ by forward substitution: $\boldsymbol{z} = \boldsymbol{L} \backslash \boldsymbol{b}$
- Solve $\boldsymbol{L}^T\boldsymbol{x} = \boldsymbol{z}$ wrt $\boldsymbol{z}$ by backward substitution: $\boldsymbol{x} = \boldsymbol{L}^T \backslash \boldsymbol{z}$.
- So: $\boldsymbol{x} = \boldsymbol{L}^T \backslash (\boldsymbol{L} \backslash \boldsymbol{b})$.
- $|\boldsymbol{A}| = \left|\boldsymbol{L}\boldsymbol{L}^T\right| = |\boldsymbol{L}|^2 = \left(\prod_{i=1}^{p} L_{ii}\right)^2$.
- $\boldsymbol{y}^T\boldsymbol{A}^{-1}\boldsymbol{y} = \boldsymbol{y}^T(\boldsymbol{L}\boldsymbol{L}^T)^{-1}\boldsymbol{y} = (\boldsymbol{L} \backslash \boldsymbol{y})^T(\boldsymbol{L} \backslash \boldsymbol{y})$.
- Cholesky also preserves sparsity.[1]
- Pre-conditioned conjugate gradient (PCG). $\boldsymbol{A}\boldsymbol{x} \approx \boldsymbol{b}$. Fast.

[1] Rue and Held (2005). Gaussian Markov random fields. C&H.