Machine Learning Models and Methods for Econometricians
Gaussian process Regression

Mattias Villani

Division of Statistics and Machine Learning
Department of Computer and Information Science
Linköping University

Topic overview

- Recall: **The multivariate normal distribution**
- Recall: Bayesian inference for **Gaussian linear/nonlinear regression**
- Introduction to **Gaussian Process Regression**
- **Kernel functions**
- Estimating the **GP hyperparameters**

The multivariate normal distribution

- The **density function** of a p-variate normal vector $\mathbf{x} \sim N(\mu, \Sigma)$ is

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

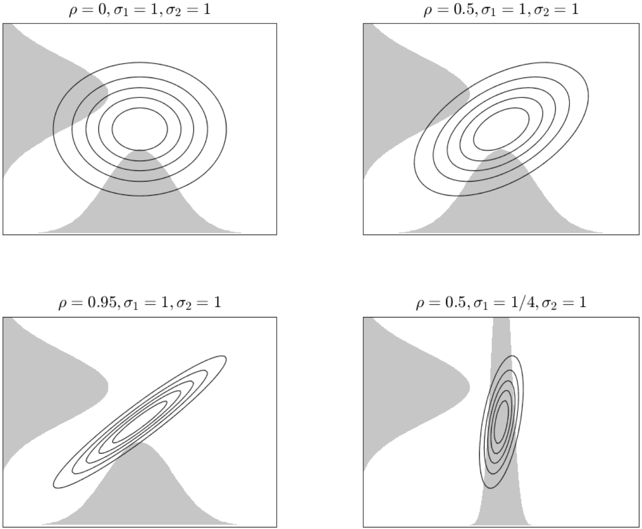
- Example: **Bivariate normal** $(p = 2)$

$$\Sigma = \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}$$

- Mean and variance

$$E(\mathbf{x}) = \mu \quad Var(\mathbf{x}) = \Sigma$$

Multivariate normal



Nonlinear regression

- **Linear regression**

$$y = f(\mathbf{x}) + \epsilon$$
$$f(\mathbf{x}) = \mathbf{w}^T \cdot \mathbf{x}$$

  and $\epsilon \sim N(0, \sigma_n^2)$ and iid over observations.
- The weights $\mathbf{w}$ are called regression coefficients (β) in statistics.
- **Polynomial regression**: $\phi(\mathbf{x}) = (1, x, x^2, x^3, ..., x^k)$:

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

- More generally: **splines** with **basis functions**.
- Polynomial and spline models are linear in $\mathbf{w}$. Least squares!

Bayesian linear regression - inference

- Linear regression for all n observations

$$\underset{n\times 1}{\mathbf{y}} = \underset{n\times p}{\mathbf{X}}\underset{p\times 1}{\mathbf{w}} + \underset{n\times 1}{\varepsilon}$$

- $\mathbf{w}$ is unknown. σ_n is assumed known.
- **Prior**

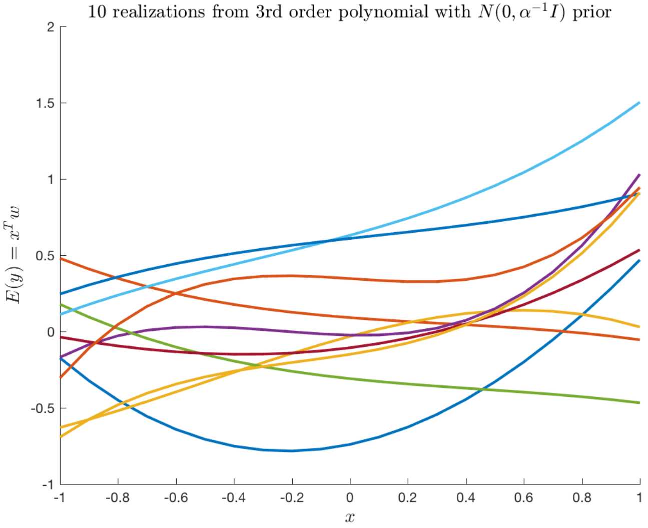
$$\mathbf{w} \sim N\left(0, \Sigma_p\right)$$

- Common choice (Ridge regression): $\Sigma_p = \alpha^{-1}\mathbf{I}$.
- **Posterior**

$$\begin{aligned} \mathbf{w}|\mathbf{X},\mathbf{y} &\sim N\left(\bar{\mathbf{w}}, \mathbf{A}^{-1}\right) \\ \mathbf{A} &= \sigma_n^{-2}\mathbf{X}^T\mathbf{X} + \Sigma_p^{-1} \\ \bar{\mathbf{w}} &= \left(\mathbf{X}^T\mathbf{X} + \sigma_n^2\Sigma_p^{-1}\right)^{-1}\mathbf{X}^T\mathbf{y} \end{aligned}$$

- **Posterior precision = Data Precision + Prior Precision.**

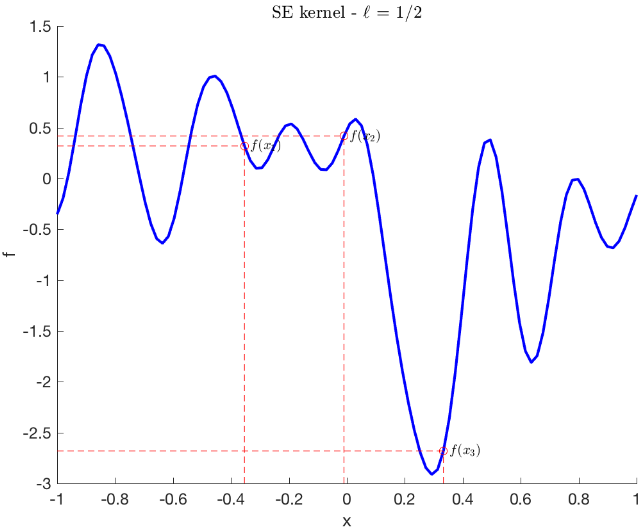
A PRIOR ON **w** IS A PRIOR ON FUNCTIONS



Non-parametric regression

- **Non-parametric regression**: avoiding a parametric form for $f(\cdot)$. Treat $f(\mathbf{x})$ as an unknown parameter for every $\mathbf{x}$.
- **Weight space view**
  - Restrict attention to a grid of (ordered) x-values: $x_1, x_2, .., x_k$.
  - Put a joint prior on the k function values: $f(x_1), f(x_2), ..., f(x_k)$.
- **Function space view**
  - Treat f as an **unknown function**.
  - Put a **prior over a set of functions**.

Nonparametric = one parameter for every x!



GAUSSIAN PROCESS REGRESSION

- Weight-space view. GP assumes

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

- But how do we specify the $k \times k$ **covariance matrix K**?

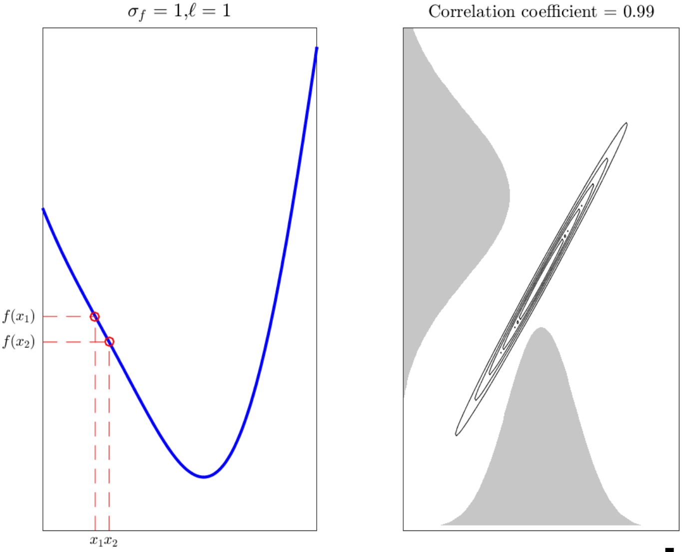
$$Cov(f(x_p), f(x_q))$$

- **Squared exponential covariance function**

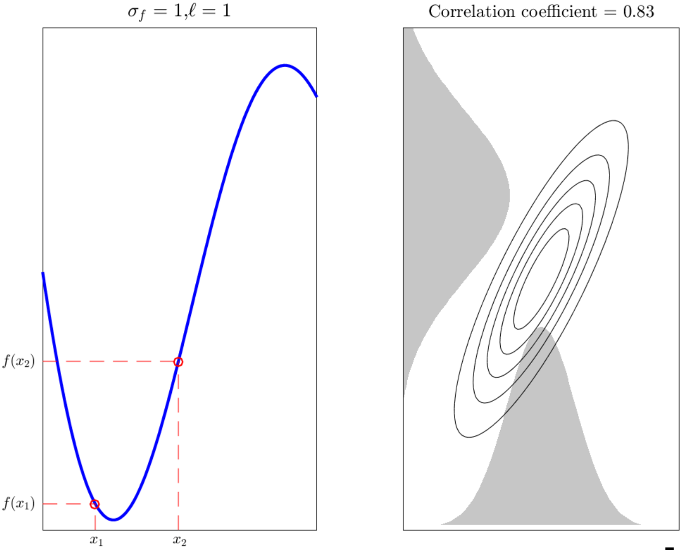
$$Cov(f(x_p), f(x_q)) = k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{x_p - x_q}{\ell}\right)^2\right)$$

- Nearby x's have highly correlated function ordinates $f(x)$.
- We can compute $Cov(f(x_p), f(x_q))$ for *any* x_p and x_q.
- Extension to multiple covariates: $(x_p - x_q)^2$ replaced by $(\mathbf{x}_p - \mathbf{x}_q)^T(\mathbf{x}_p - \mathbf{x}_q)$.

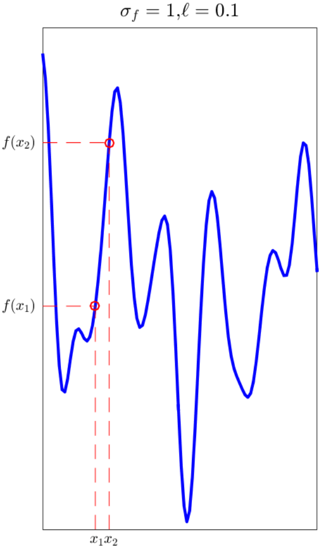
Smooth function - points nearby

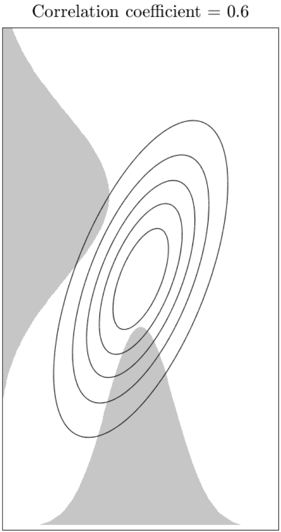


SMOOTH FUNCTION - POINTS FAR APART

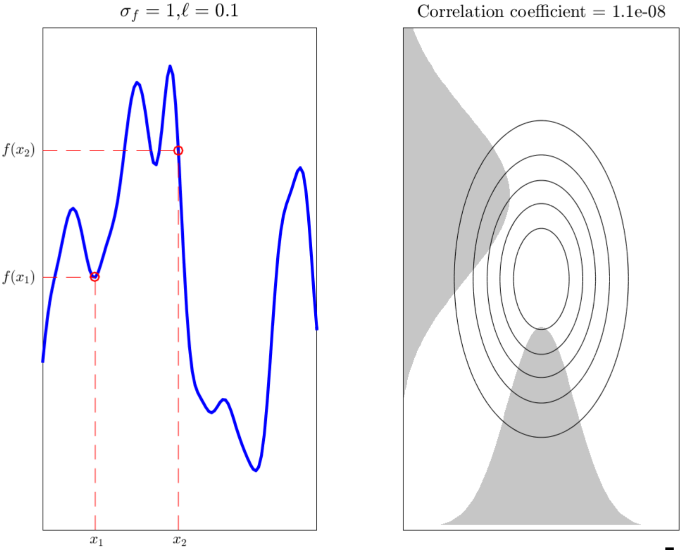


Jagged function - points nearby





Jagged function - points far apart



Gaussian process regression, cont.

Definition

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

- A Gaussian process is really a **probability distribution over functions** (curves).
- A GP is completely 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' (note: this is *not* the transpose here).
- A **Gaussian process** is denoted by

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

- **Bayesian**: $f(x) \sim GP$ encodes **prior beliefs** about the unknown $f(\cdot)$.

SIMULATING A GP

- Example:

$$m(x) = \sin(10x)$$
$$k(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{x - x'}{\ell}\right)^2\right)$$

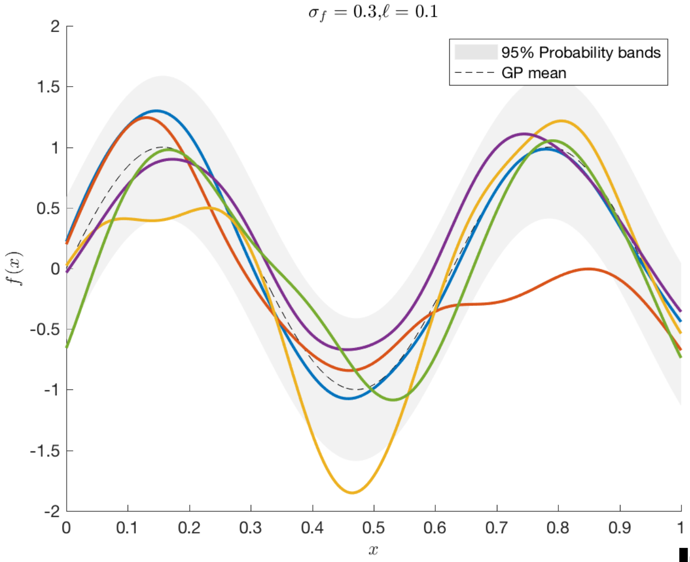
  where $\ell > 0$ is the length scale.

- Larger ℓ gives more smoothness in $f(x)$.
- Simulate draw from $f(x) \sim GP\left(m(x), k(x, x')\right)$ over a grid $\mathbf{x}_* = (x_1, ..., x_n)$ by using that

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

- Note that the **kernel** $k(x, x')$ produces a **covariance matrix** $K(\mathbf{x}_*, \mathbf{x}_*)$ when evaluated at the vector $\mathbf{x}_*$.

Simulating a GP



Three commonly used covariance kernels

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

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

  - Infinitely mean square differentiable. Very smooth.
- **Rational Quadratic (RQ)** ($\ell > 0$, $\sigma_f > 0$,$\alpha > 0$)

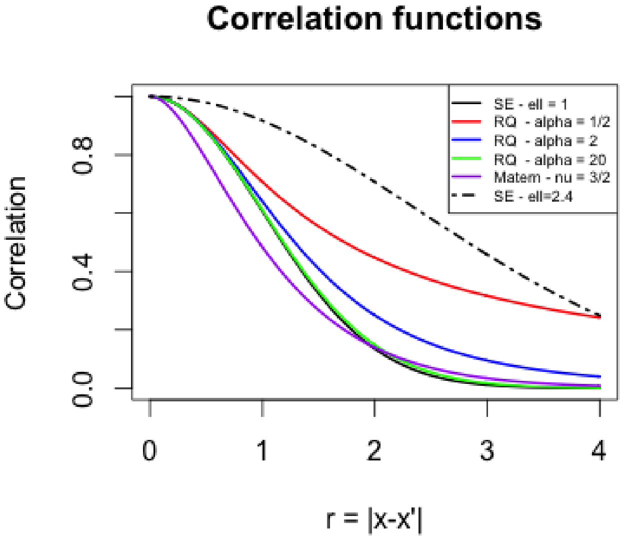
$$K_{RQ}(r) = \sigma_f^2 \left(1 + \frac{r^2}{2\alpha\ell^2}\right)^{-\alpha}$$

  - RQ is sum of SE with different ℓ. When $\alpha \to \infty$, $K_{RQ}(r) \to K_{SE}(r)$.
- **Matérn** ($\ell > 0$, $\sigma_f > 0$, $\nu > 0$)

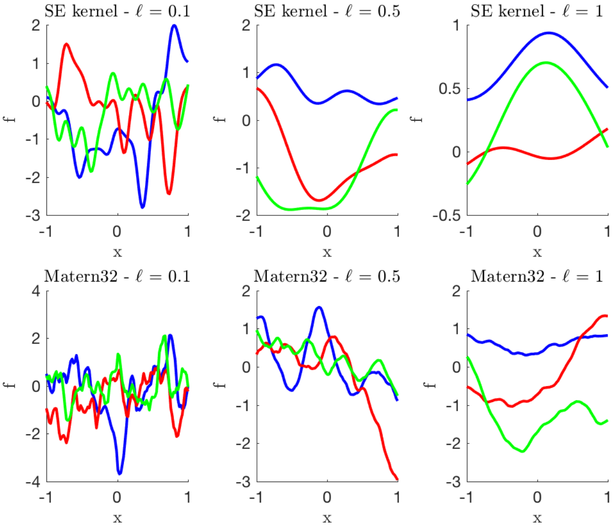
$$K_{Matern}(r) = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}r}{\ell}\right)^{\nu} K_{\nu}\left(\frac{\sqrt{2\nu}r}{\ell}\right)$$

  - $\nu = 3/2$ and $\nu = 5/2$ common. As $\nu \to \infty$, $K_{Matern}(r) \to K_{SE}(r)$.

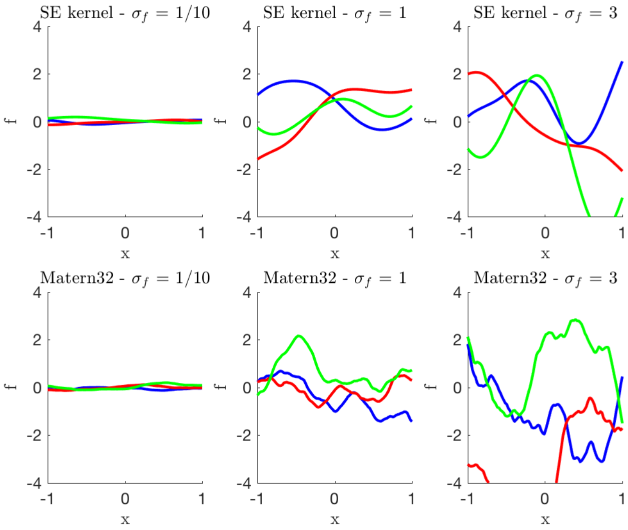
CORRELATION AS A FUNCTION OF DISTANCE



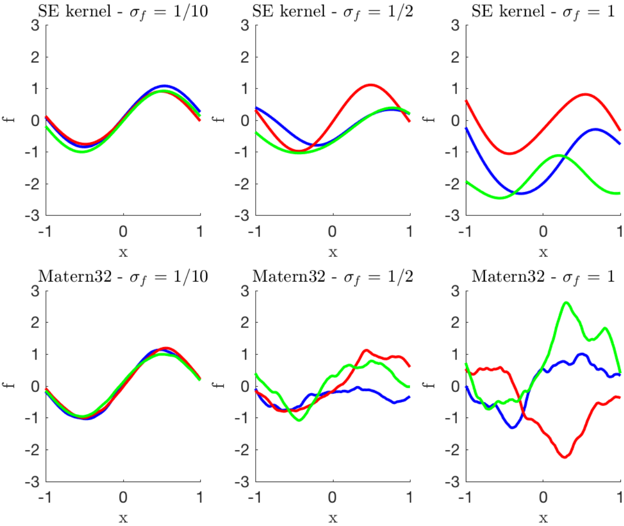
The length scale ℓ determines the smoothness



THE SCALE FACTOR σ_f DETERMINES THE VARIANCE



THE MEAN CAN BE $sin(3x)$. OR WHATEVER.



Simulating a GP

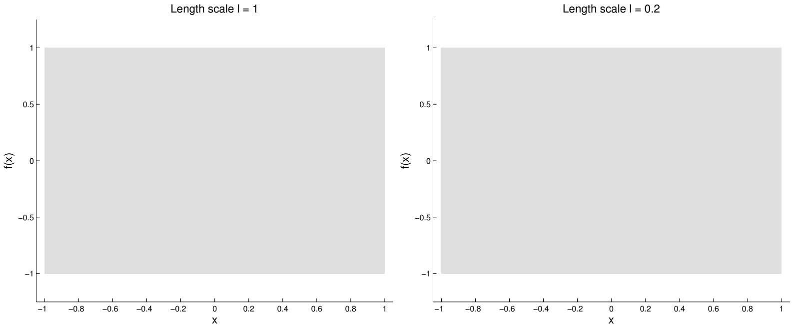
- The joint way: Choose a grid $x_1, ..., x_k$. Simulate the k-vector

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

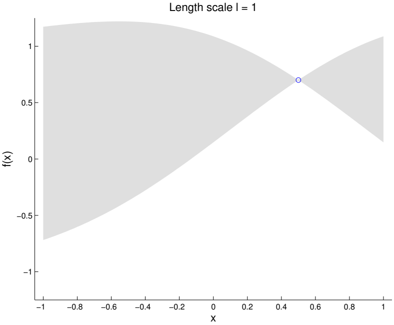
- More intuition from the conditional decomposition

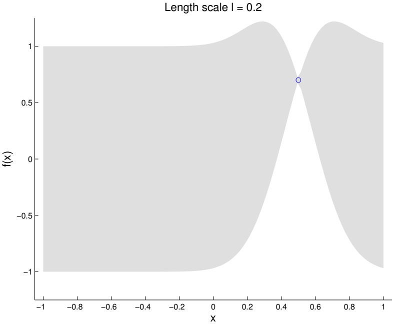
$$\begin{aligned} p(f(x_1), f(x_2), ..., f(x_k)) = & \, p(f(x_1))\, p(f(x_2)|f(x_1)) \cdots \\ & \times p(f(x_k)|f(x_1), ..., f(x_{k-1})) \end{aligned}$$

Simulation from ℓ=1 vs ℓ=0.2. Before first draw.

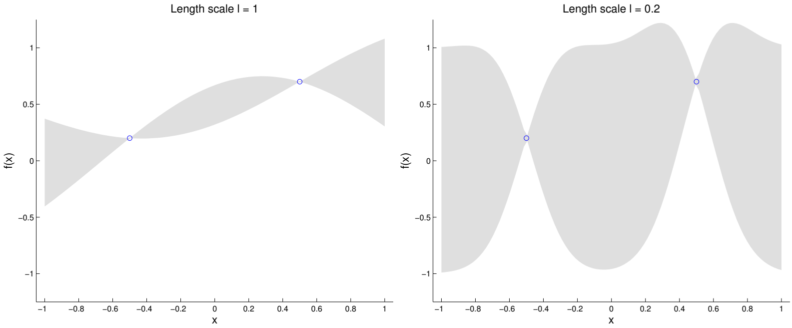


Simulation from ℓ=1 vs ℓ=0.2. Before second draw.

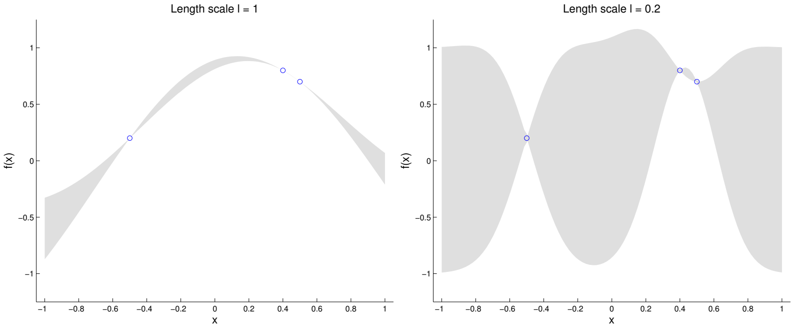




Simulation from ℓ=1 vs ℓ=0.2. Before third draw.



Simulation from ℓ=1 vs ℓ=0.2. Before fourth draw.



The posterior for a Gaussian Process Regression

- **Model**

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

- **Prior**

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

- You have observed the data: $\mathbf{x} = (x_1, ..., x_n)^T$ and $\mathbf{y} = (y_1, ..., y_n)^T$.
- Goal: the posterior of $f(\cdot)$ over a set of x-values: $\mathbf{f}_\star = \mathbf{f}(\mathbf{x}_\star)$.
- The **posterior** (use formula for conditional Gaussian above)

$$\mathbf{f}_\star | \mathbf{x}_\star, \mathbf{x}, \mathbf{y} \sim N\left(\bar{\mathbf{f}}_\star, \Omega\right)$$

$$\bar{\mathbf{f}}_\star = K(\mathbf{x}_\star, \mathbf{x})\left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1}\mathbf{y}$$

$$\Omega = K(\mathbf{x}_\star, \mathbf{x}_\star) - K(\mathbf{x}_\star, \mathbf{x})\left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} K(\mathbf{x}, \mathbf{x}_\star)$$

Proof sketch

- Aim: the conditional distribution $\mathbf{f}_\star|\mathbf{y}$ (x's are non-random)
- Remember:

$$y = f(x) + \varepsilon, \quad \varepsilon \sim N(0, \sigma^2), \quad f \sim GP(0, k(\mathbf{x}, \mathbf{x}'))$$

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

$$\begin{pmatrix} \mathbf{f}_\star \\ \mathbf{y} \end{pmatrix} \sim N\left[\begin{pmatrix} \mathbf{0} \\ \mathbf{0} \end{pmatrix}, \begin{pmatrix} K(\mathbf{x}_\star, \mathbf{x}_\star) & K(\mathbf{x}_\star, \mathbf{x}) \\ K(\mathbf{x}, \mathbf{x}_\star) & K(\mathbf{x}, \mathbf{x}) + \sigma^2 I_n \end{pmatrix}\right]$$

- Now just apply the resultat for conditionals of multivariate normal.

Conditional distribution from multivariate normal

- 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 $p_2 \times 1$ $(p_1 + p_2 = p)$.

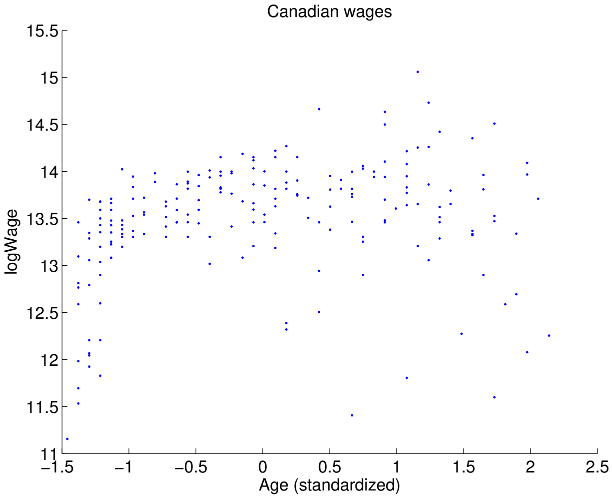
- Partition μ and Σ accordingly as

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

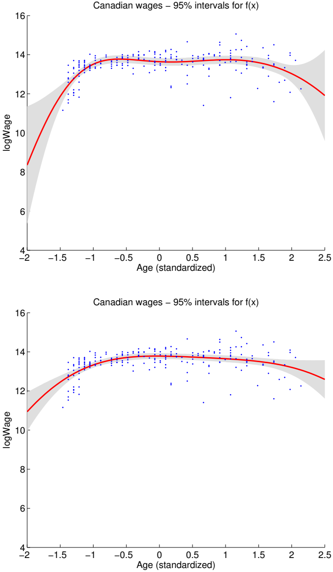
- **Conditionals are normal**. Let $\mathbf{x} \sim N(\mu, \Sigma)$, then

$$\mathbf{x}_1 | \mathbf{x}_2 = \mathbf{x}_2^* \sim N\left[\mu_1 + \Sigma_{12}\Sigma_{22}^{-1}(x_2^* - \mu_2),\ \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right]$$

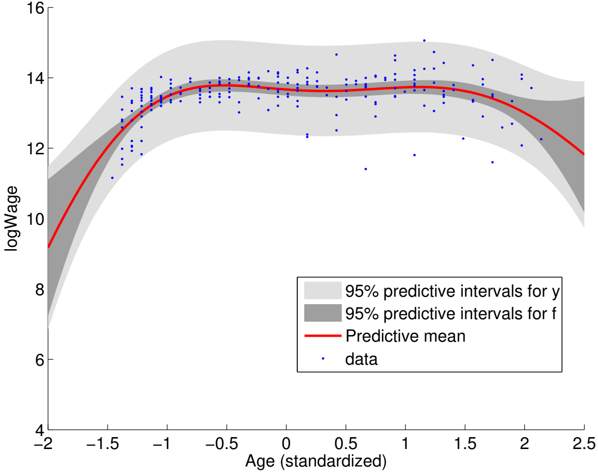
Example - Canadian wages



POSTERIOR OF F - $\ell = 0.2, 0.5, 1, 2$



Canadian wages - prediction with $\ell = 0.5$



ESTIMATING THE HYPERPARAMETERS

- Kernel depends on **hyperparameters** θ. Example SE kernel $[\theta = (\sigma_f, \ell)^T]$

$$k(\mathbf{x}, \mathbf{x}') = \sigma_f^2 \exp\left(-\frac{1}{2}\frac{\|\mathbf{x}-\mathbf{x}'\|^2}{\ell^2}\right)$$

- Common approach: choose the hyperparameters that maximizes the **marginal likelihood** (**evidence**):

$$p(\mathbf{y}|\mathbf{X}, \theta) = \int p(\mathbf{y}|\mathbf{X}, \mathbf{f}, \theta) p(\mathbf{f}|\mathbf{X}, \theta) d\mathbf{f}$$

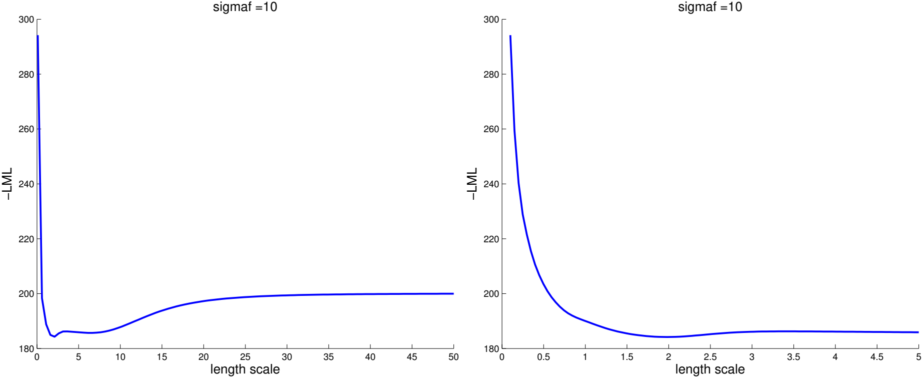
  where $\mathbf{f} = f(\mathbf{X})$ is a vector with function values in the training data.
- For Gaussian process regression:

$$\log p(\mathbf{y}|\mathbf{X}, \theta) = -\frac{1}{2}\mathbf{y}^T\left(K + \sigma^2 I\right)^{-1}\mathbf{y} - \frac{1}{2}\log\left|K + \sigma^2 I\right| - \frac{n}{2}\log(2\pi)$$

- Proper **Bayesian inference** **for hyperparameters**

$$p(\theta|\mathbf{y}, \mathbf{X}) \propto p(\mathbf{y}|\mathbf{X}, \theta)p(\theta).$$

Canadian wages - LML determination of ℓ



More than one input - ARD

- Anisotropic version of isotropic kernels by setting $r^2(\mathbf{x}, \mathbf{x}') = (\mathbf{x} - \mathbf{x}')^T \mathbf{M}(\mathbf{x} - \mathbf{x}')$ where $\mathbf{M}$ is positive definite.
- **Automatic Relevance Determination** (**ARD**): $\mathbf{M} = Diag(\ell_1^{-2}, ..., \ell_D^{-2})$ is diagonal with different length scales.
- ARD does 'variable selection' since large ℓ_j means that the jth input essentially drops out of $f(\mathbf{x})$.

More on kernels

- **Periodic kernels**. When $f(x)$ is believed to be periodic with period d. Example:

$$k(x,x') = \sigma_f^2 \exp\left(-\frac{2\sin^2\left(\pi\left|x-x'\right|/d\right)}{\ell^2}\right).$$

- Example periodic daily data: Mondays are correlated with Mondays.
- **Factor kernels**: $M = \Lambda\Lambda^T + \Psi$, where Λ is $D \times k$ for low rank k.
- **Adaptive smoothnes kernels**. Length-scales $\ell(\mathbf{x})$ that vary with $\mathbf{x}$. **Gibbs kernel** in RW Eq. 4.32.

Product of kernels

- Kernels are often combined into **composite kernels**.
- **Product** of kernels is a kernel.
- Example: Product of periodic and square exponential kernels. Locally periodic. Two nearby peaks are more dependent than two distant peaks.

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{2\sin^2\left(\pi\left|x - x'\right|^2/d\right)}{\ell^2}\right) \times \exp\left(-\frac{1}{2}\frac{\left|x - x'\right|^2}{\ell^2}\right)$$

- Example: ARD is a product of D one-dimensional kernels, one for each input variable

$$k_{ARD}(\mathbf{x}, \mathbf{x}') = \prod_{d=1}^{D} k_{SE,\ell_d}(x_d, x_d')$$

SUM OF KERNELS

- **Sum** of kernels is a kernel.
- Let $f_a \sim GP\left[m_a(\mathbf{x}), k_a(\mathbf{x}, \mathbf{x}')\right]$ independently of $f_b \sim GP\left[m_b(\mathbf{x}), k_b(\mathbf{x}, \mathbf{x}')\right]$ then

$$f_a + f_b \sim GP\left[m_a(\mathbf{x}) + m_b(\mathbf{x}), k_a(\mathbf{x}, \mathbf{x}') + k_b(\mathbf{x}, \mathbf{x}')\right]$$

- Adding up kernels is the same as adding up functions.

Discrete covariates

- Suppose: x_1 is continuous (mg/week) and x_2 is binary (sex).
- Linear regression: just use x_2 coded as $x_2 = 0$ if male, $x_2 = 1$ if female.
- Implicit model:

$$y = \begin{cases} \beta_0 + \beta_1 x_1 & \text{if } x_2 = 0 \\ \beta_0 + \tilde{\beta}_0 + (\beta_1 + \tilde{\beta}_1) x_1 & \text{if } x_2 = 1 \end{cases}$$

- GP: add the 0-1 coded covariate and use ARD kernel:

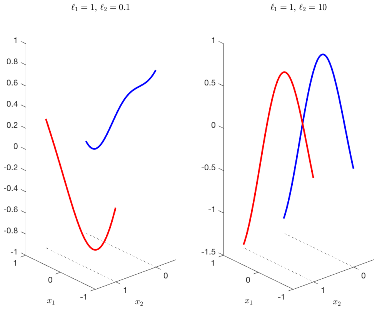
$$\exp\left(-\frac{1}{2}\left(\frac{x_1 - x_1'}{\ell_1}\right)^2\right) \exp\left(-\frac{1}{2}\left(\frac{x_2 - x_2'}{\ell_2}\right)^2\right)$$

So the covariance between $f(x_1, 0)$ and $f(x_1, 1)$ is

$$\exp\left(-\frac{1}{2}\left(\frac{1}{\ell_2}\right)^2\right)$$

Discrete covariates

- Large ℓ_2: men and female are believed to have similar profiles with respect to x_1.
- Small ℓ_2: men and female are believed to have potentially very different profiles with respect to x_1.



- Categorical covariates with K levels: create K *one-hot* variables.

SOFTWARE

- Python: GPy
- Matlab: Statistics and Machine Learning Toolbox, GPML, GPstuff.
- R: Kernlab,

Example Matlab's own toolbox

- Statistics and Machine Learning Toolbox.
- Many kernels, fitting methods etc.
- Limited to **regression** (continuous response).
- Can include explicit basis functions.

-
```
gprMdl = fitrgp(Xtrain,ytrain,'FitMethod','fic',
'KernelFunction','ardsquaredexponential',
'KernelParameters',[sigmaM0;sigmaF0],
'Sigma',sigma0);
```

- See `MatlabGPexample.m`

Example R - kernlab

- The kernlab package includes many Kernel methods (e.g. SVM), including also GPs.
- Non-traditional parametrization of kernel functions.
- Can do both **regression** (continuous response) or **classification** (categorical response).

- `GPfit <- gausspr(logWage ~ age, kernel = 'rbfdot', par = list(sigma = 1))`

- See `KernLabDemo.R`