Advanced Bayesian Learning
Gaussian Processes
Spring 2014

Mattias Villani

Division of Statistics
Department of Computer and Information Science
Linköping University

Topic overview

- Gaussian process regression
  - Recall: Bayesian inference for Gaussian linear/nonlinear regression.
  - Gaussian processes for nonparametric regression
  - Covariance kernels
  - Properties of GPs
  - Selecting the kernel and hyperparameters
- Gaussian process classification
  - Flexible classification using GPs
  - Laplace approximation of the posterior
- Main literature: Rasmussen and Williams (2006). *Gaussian Processes for Machine Learning*.

Flexible nonlinear regression

- **Linear regression**

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

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

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

  where $\phi(\mathbf{x}) = (\phi_1(\mathbf{x}), \phi_2(\mathbf{x}), ..., \phi_N(\mathbf{x}))^T$ for N basis functions
- Example: **thin plate splines** with N knots $\kappa_1, ..., \kappa_N$ in $\mathbf{x}$-space

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

Bayesian linear regression - inference

- $\mathbf{w}$ is unknown. σ_n is assumed known.
- **Prior** [note: RW do *not* use $\Sigma_p = \sigma_n^2 \Omega$]

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

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

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

- Recall: **Posterior precision = Data Precision + Prior Precision**.
- Marginal posterior of $\mathbf{w}$ is multivariate student-t when σ_n^2 is unknown with Inv-χ^2 prior (and $\Sigma_p = \sigma_n^2 \Omega$).

Bayesian linear regression - prediction

- **Predictive density for mean** $f(\mathbf{x}_*)$ at new location $\mathbf{x}_*$

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

- Proof: $f(\mathbf{x}_*) = \mathbf{x}_*^T \mathbf{w}$ and $\mathbf{w}$ has a normal posterior. Linear combs of normals is normal.

- **Predictive density for new response** y_*

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

- Replace $\mathbf{X}$ with $\Phi(\mathbf{X})$ in the above for the case with basis expansion (e.g. splines).

Non-parametric regression

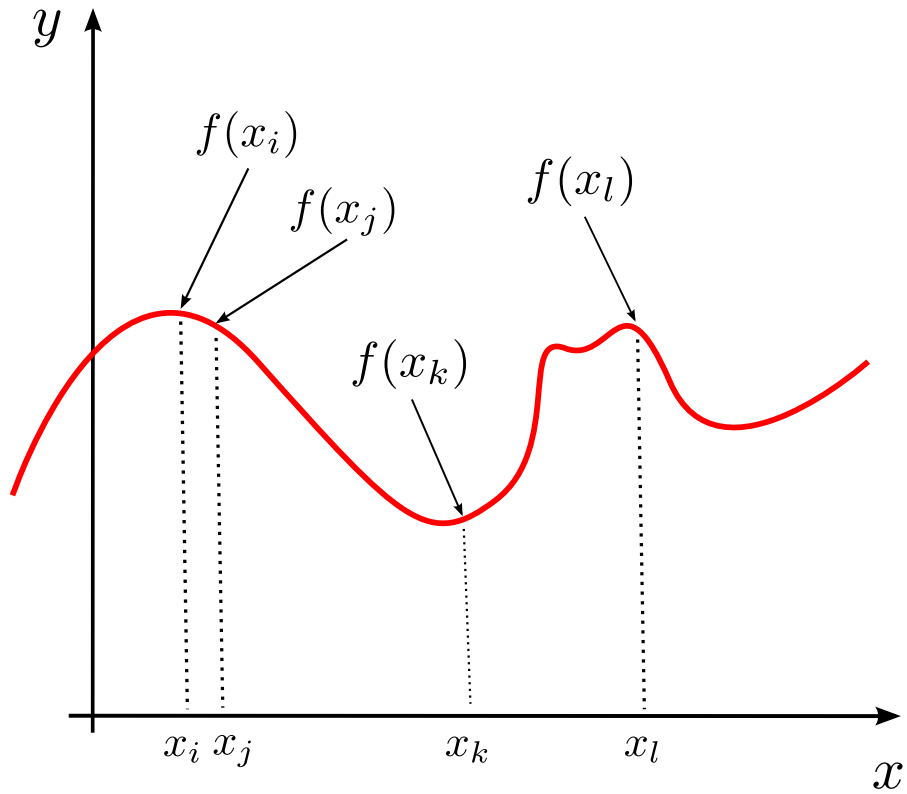
- **Non-parametric regression**: avoiding a parametric form for $f(\cdot)$. Treat $f(\mathbf{x})$ as an unknown parameter for every $\mathbf{x}$.

- **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**.

- Kolmogorov's existence theorem for stochastic processes equates the two views. Just make sure that the set of finite dimensional distributions are consistent: adding or deleting variables does not change the marginal of the original variable set.

NONPARAMETRIC = ONE PARAMETER FOR EVERY X!



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** $(p = 2)$

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

- **Linear combinations**. Let $\mathbf{y} = \mathbf{B}\mathbf{x} + \mathbf{b}$, where $\mathbf{x}$ is $p \times 1$ and $\mathbf{B}$ is a $m \times p$ constant matrix. Then

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

The multivariate normal distribution, cont.

- 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

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

- **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

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

- Life is beautiful ...

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** $\mathbf{K}$?

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

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**?

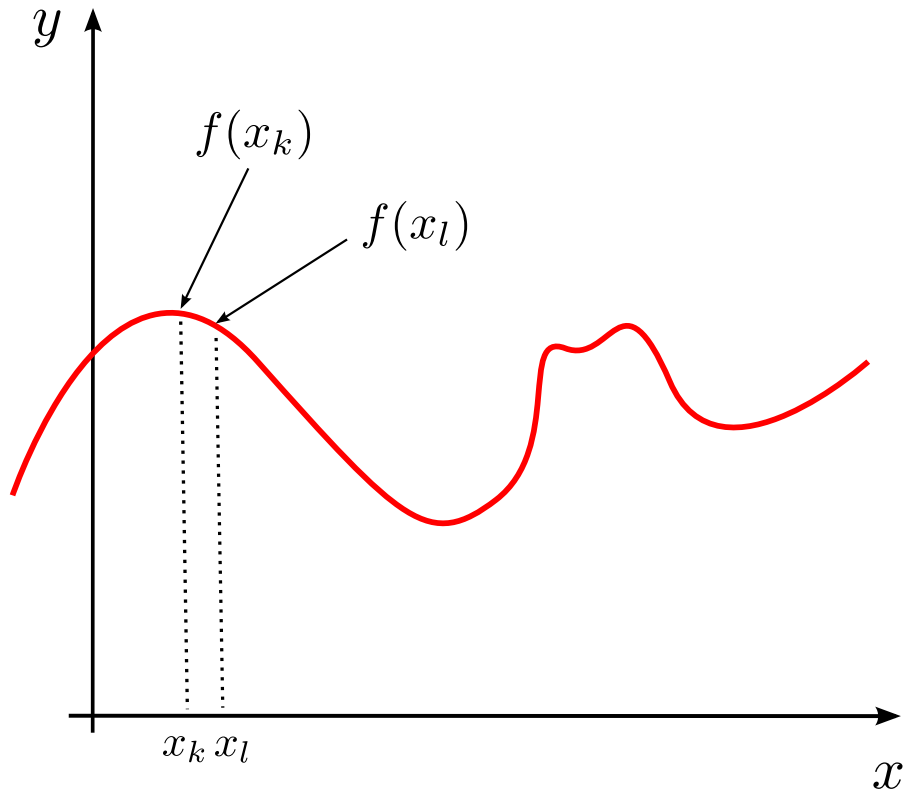
$$Cov\left(f(x_p), f(x_q)\right)$$

- **Squared exponential covariance function**

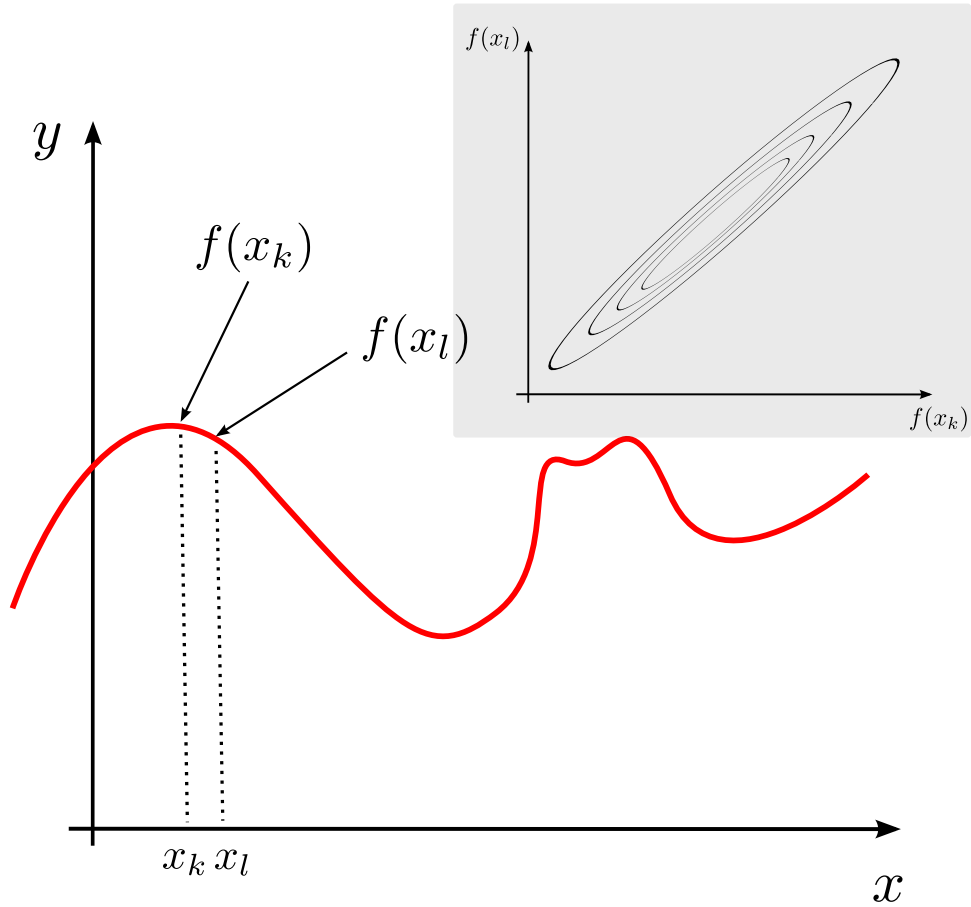
$$Cov\left(f(x_p), f(x_q)\right) = k(x_p, x_q) = \sigma_f^2 \exp\left(-\frac{1}{2}(x_p - x_q)^2\right)$$

- 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.
- Extension to multiple covariates: $(x_p - x_q)$ replaced by $\|\mathbf{x}_p - \mathbf{x}_q\|$.

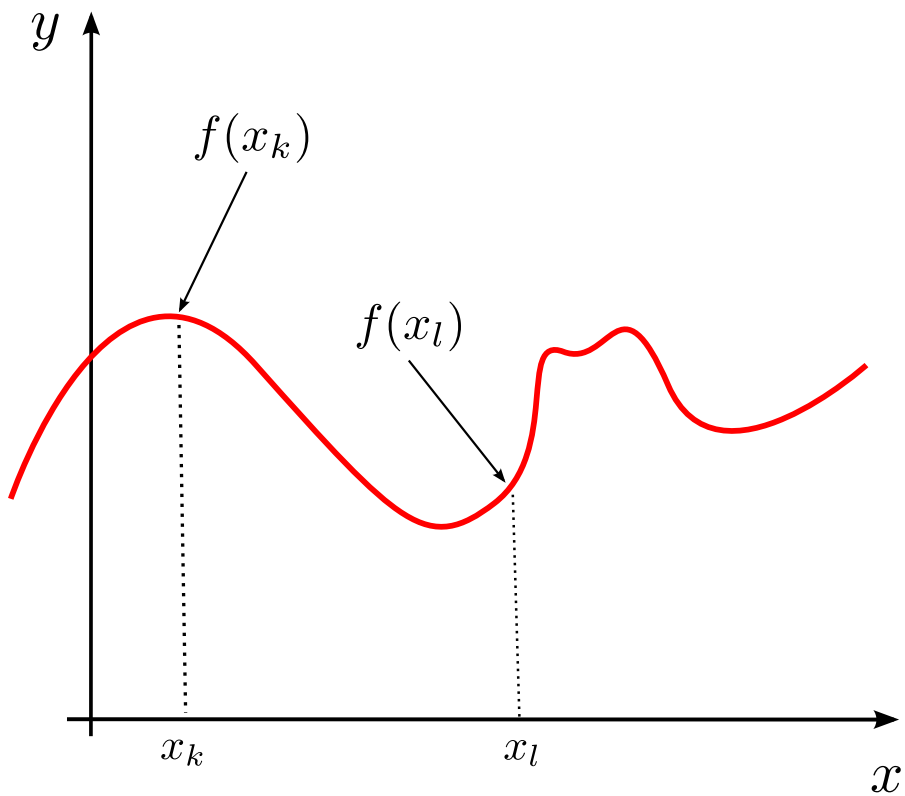
SMOOTH FUNCTION - POINTS NEARBY



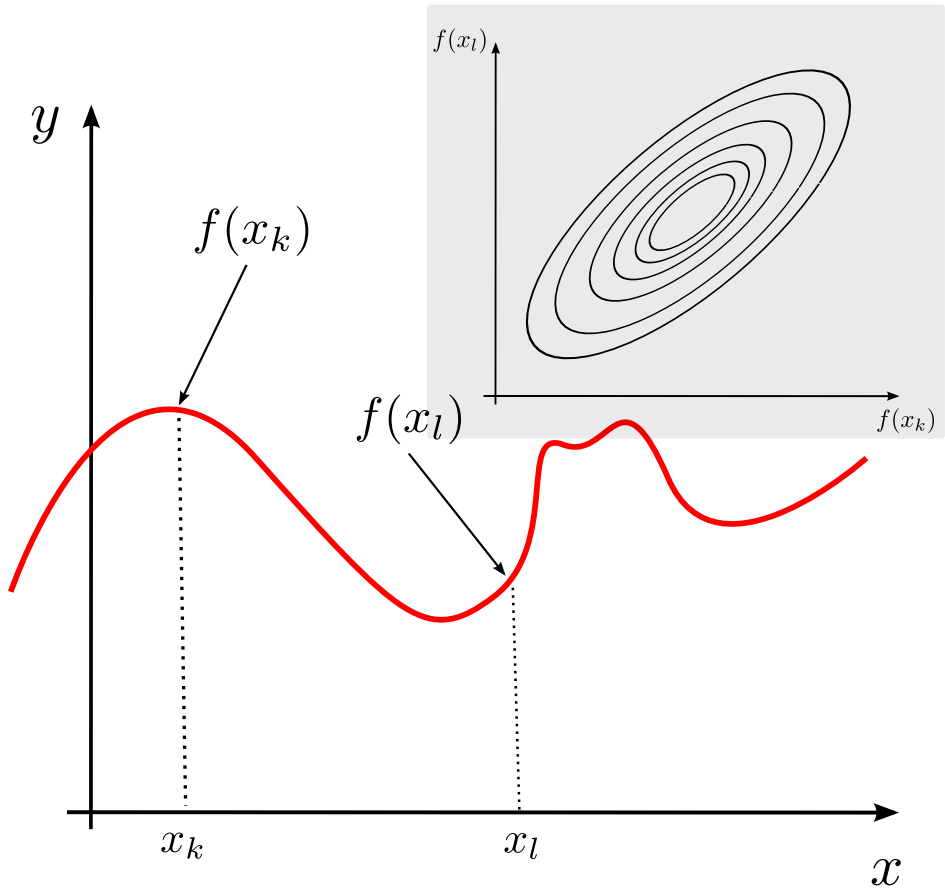
SMOOTH FUNCTION - POINTS NEARBY



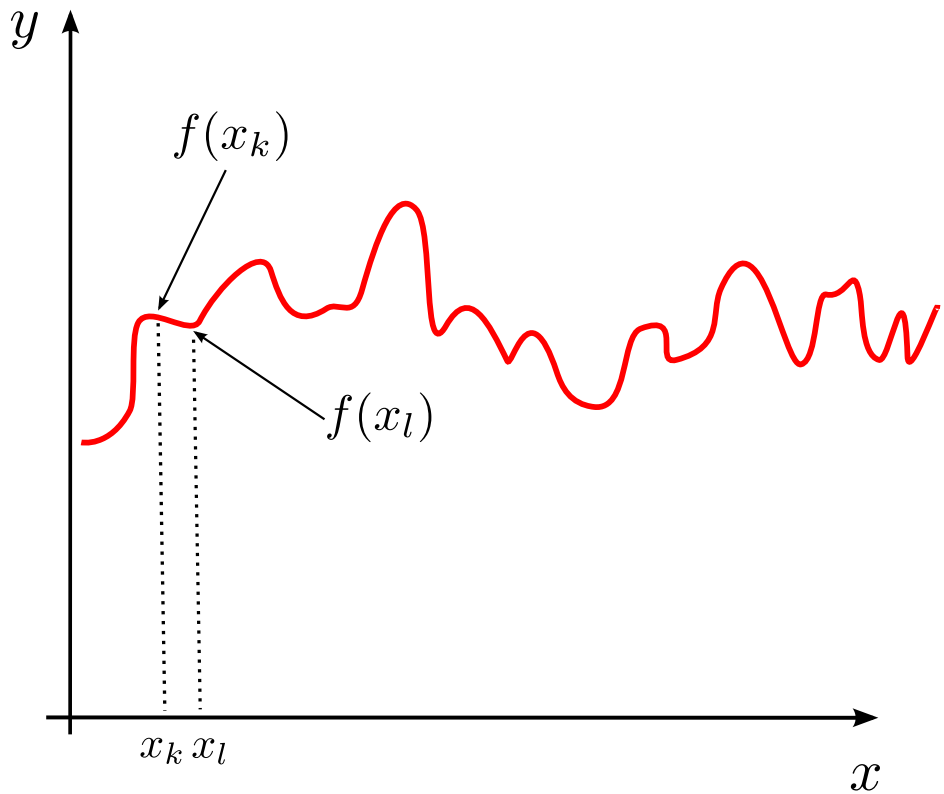
Smooth function - points far apart



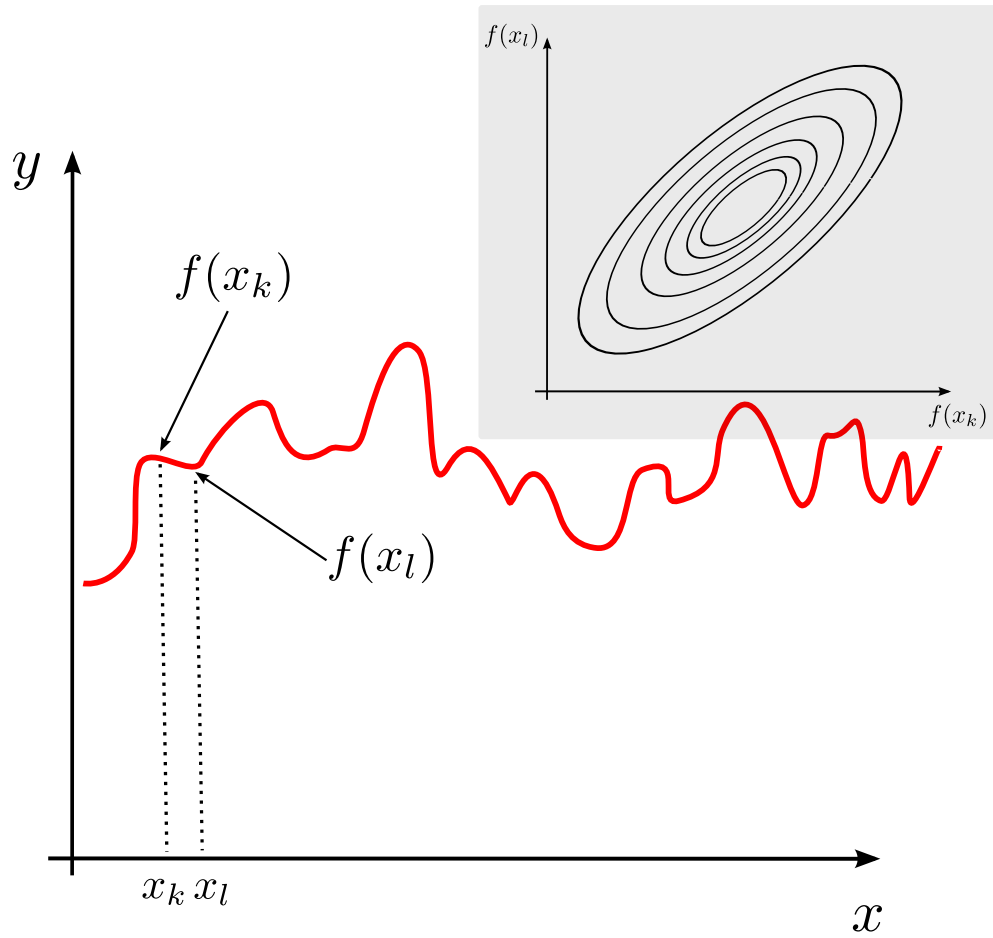
SMOOTH FUNCTION - POINTS FAR APART



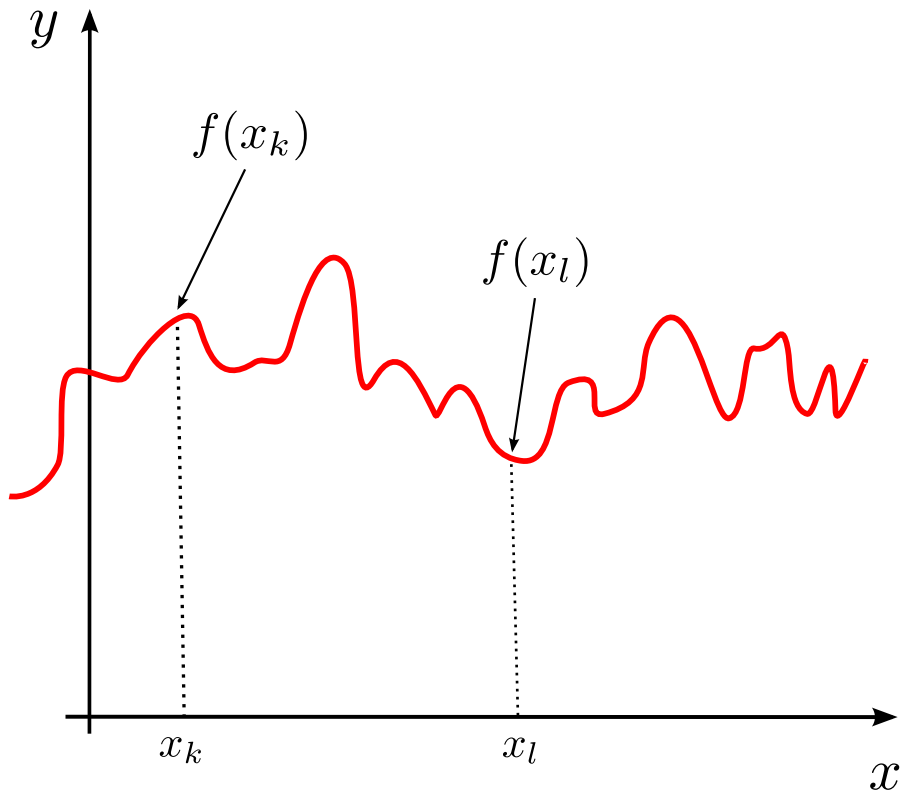
JAGGED FUNCTION - POINTS NEARBY



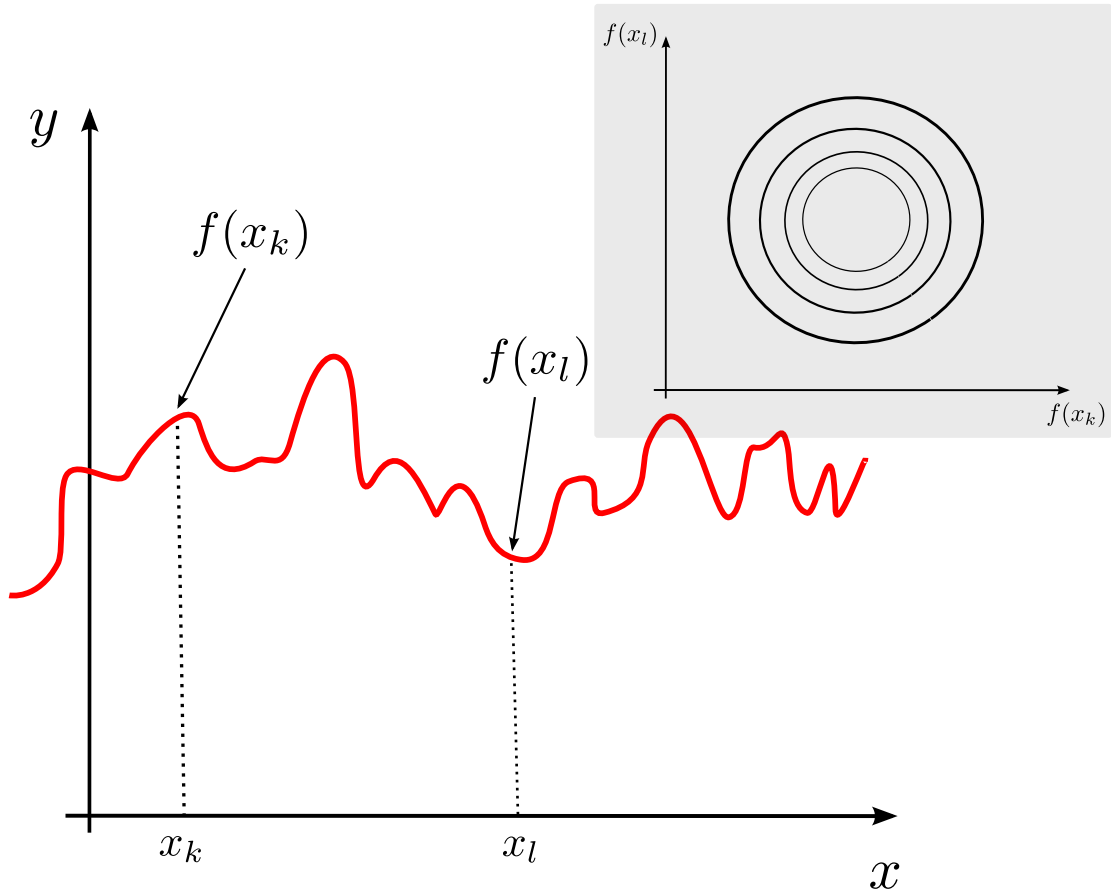
JAGGED FUNCTION - POINTS NEARBY



JAGGED FUNCTION - POINTS FAR APART



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). No need for a grid!
- 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 $f(\cdot)$.

A simple GP example

- Example:

$$m(x) = \sin(x)$$

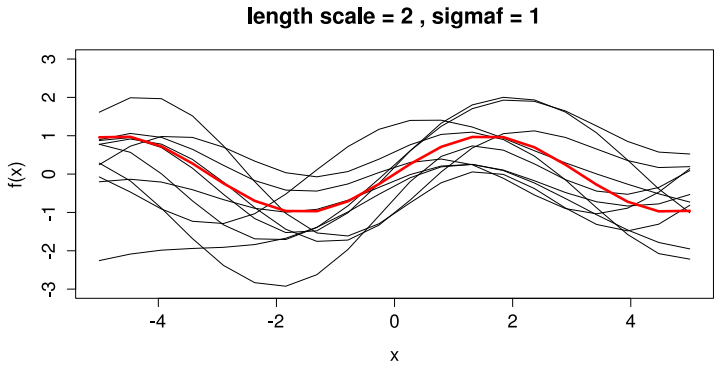
$$k(x, x') = \sigma_f^2 \exp\left(-\frac{1}{2}\left(\frac{x - x'}{\ell}\right)^2\right)$$

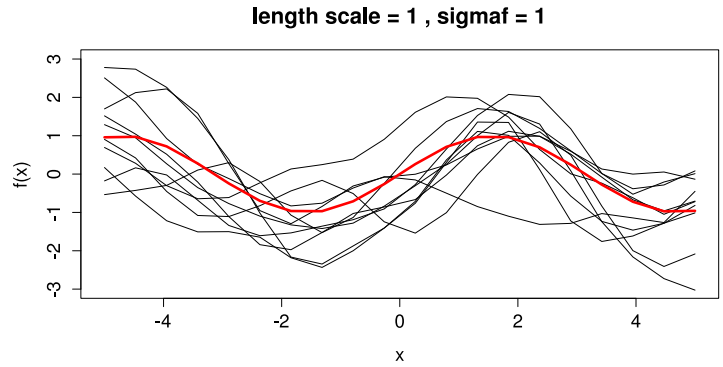
  where $\ell > 0$ is the length scale.
- Larger ℓ gives more smoothness in $f(x)$.
- Simulate draw from $f(x) \sim GP\left(m(x), k(x, x')\right)$ over a grid $\mathbf{x}_* = (x_1, ..., x_n)$ by using that

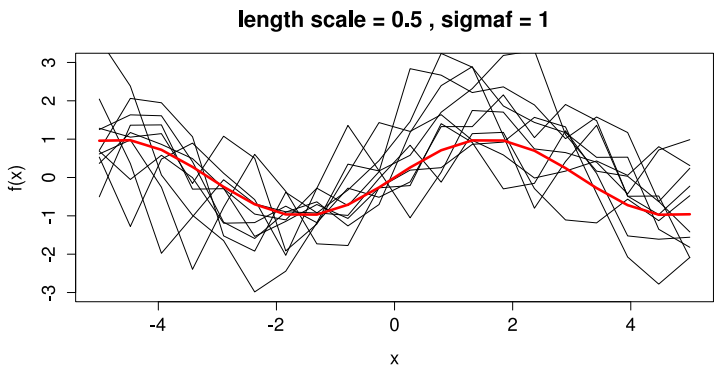
$$f(\mathbf{x}_*) \sim N\left(m(\mathbf{x}_*), K(\mathbf{x}_*, \mathbf{x}_*)\right)$$

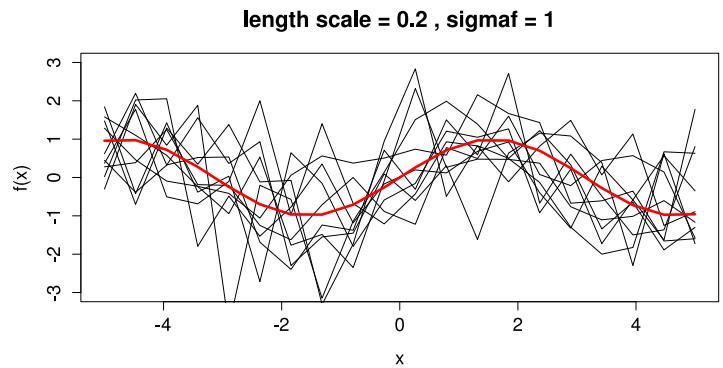
- Note that the **kernel** $k(x, x')$ produces a **covariance matrix** $K(\mathbf{x}_*, \mathbf{x}_*)$ when evaluated at the vector $\mathbf{x}_*$.

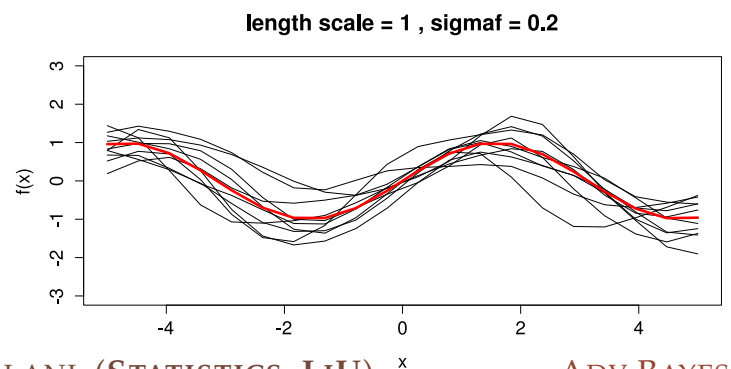
Simulating a GP - sine mean and SE kernel

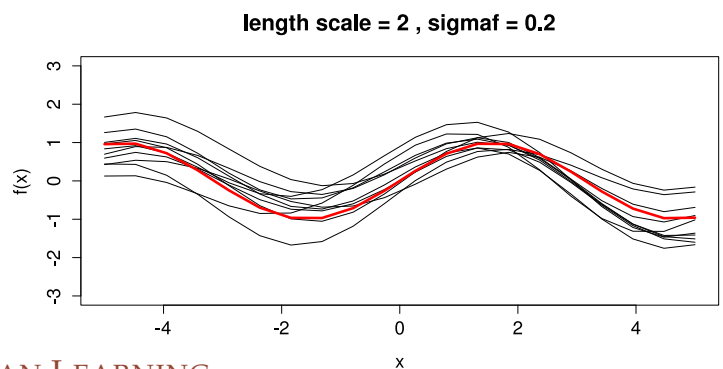












Simulating a GP

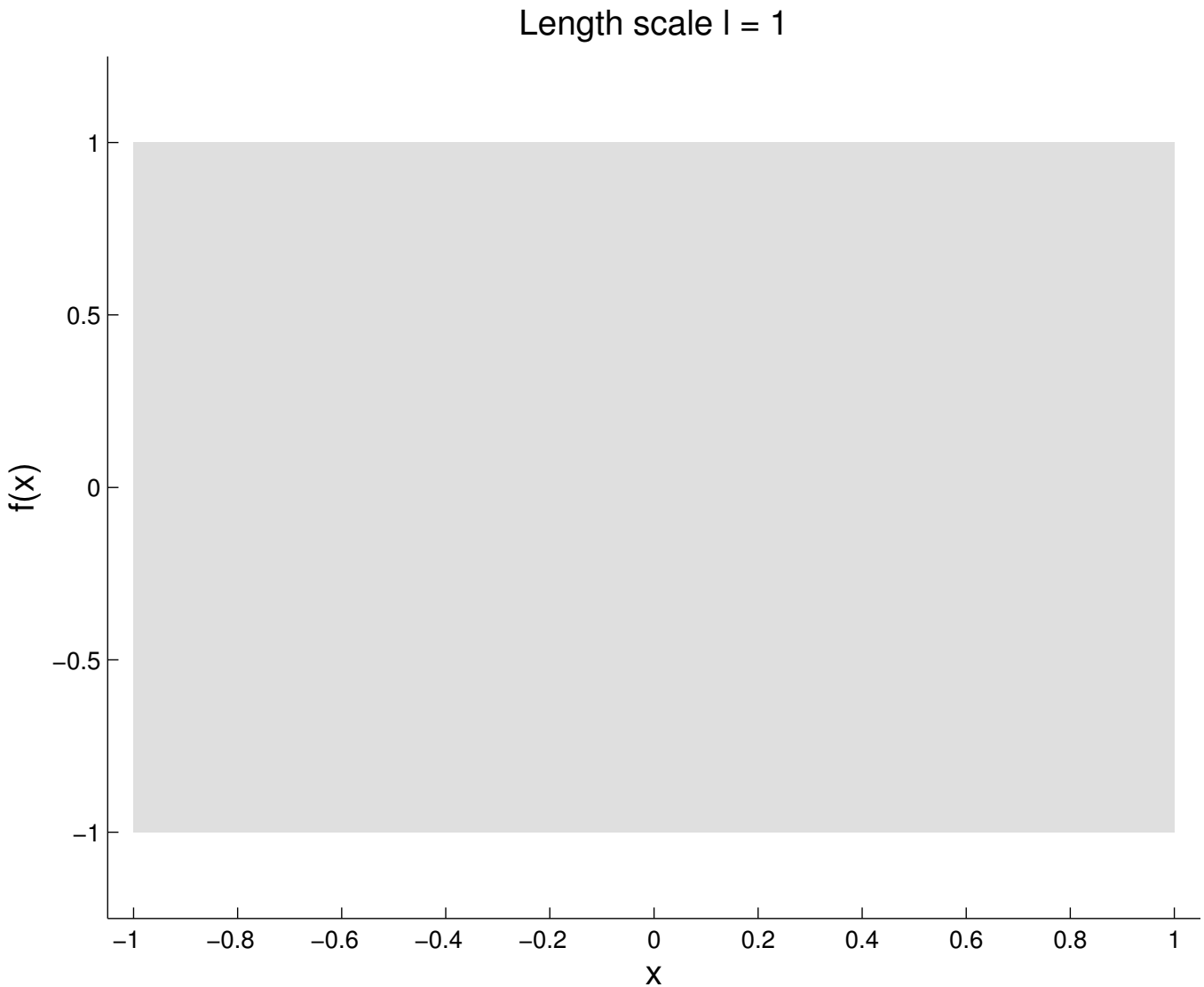
- The joint way: Choose a grid $x_1, ..., x_k$. Simulate the k-vector

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

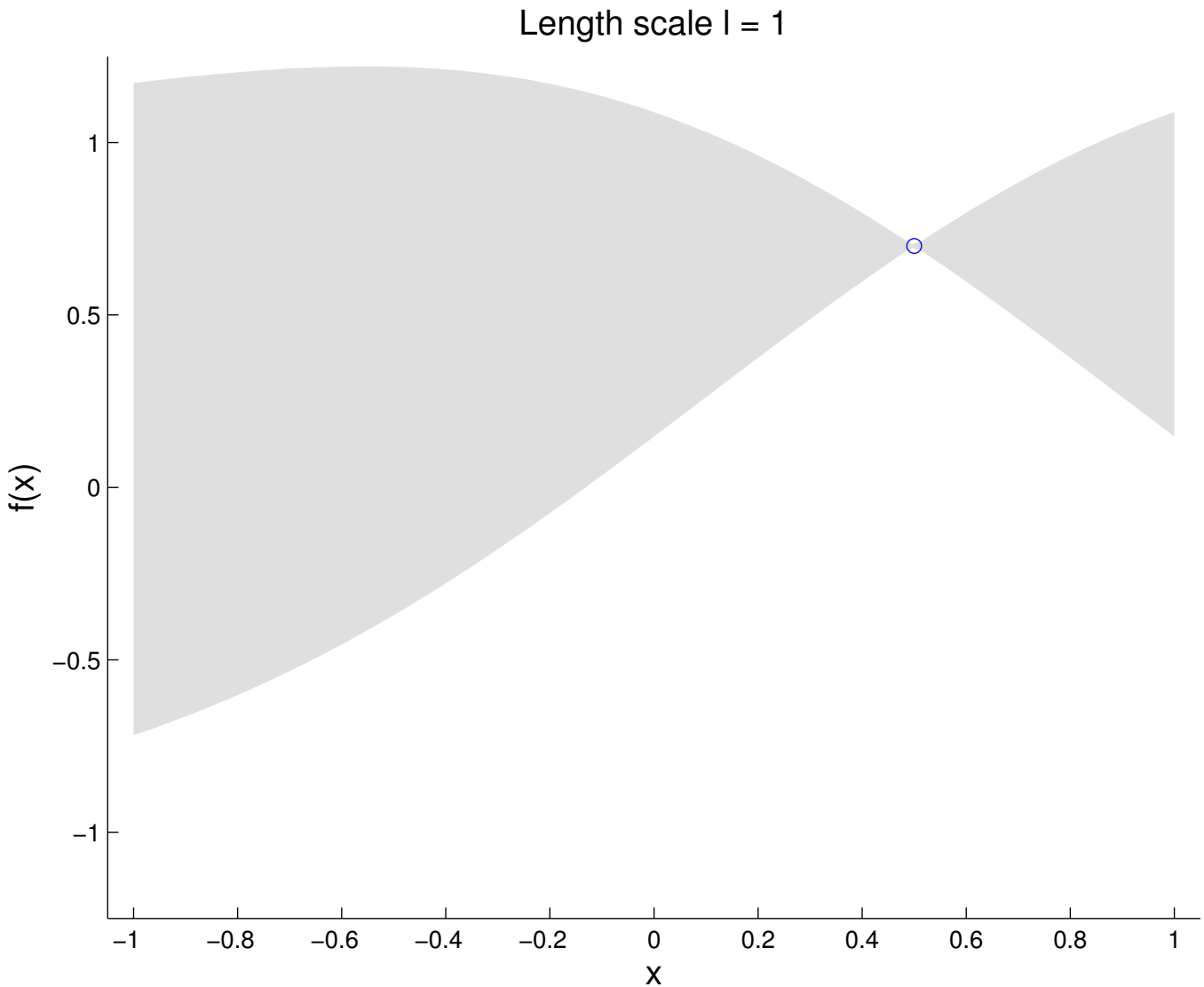
- More intuition from the conditional decomposition

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

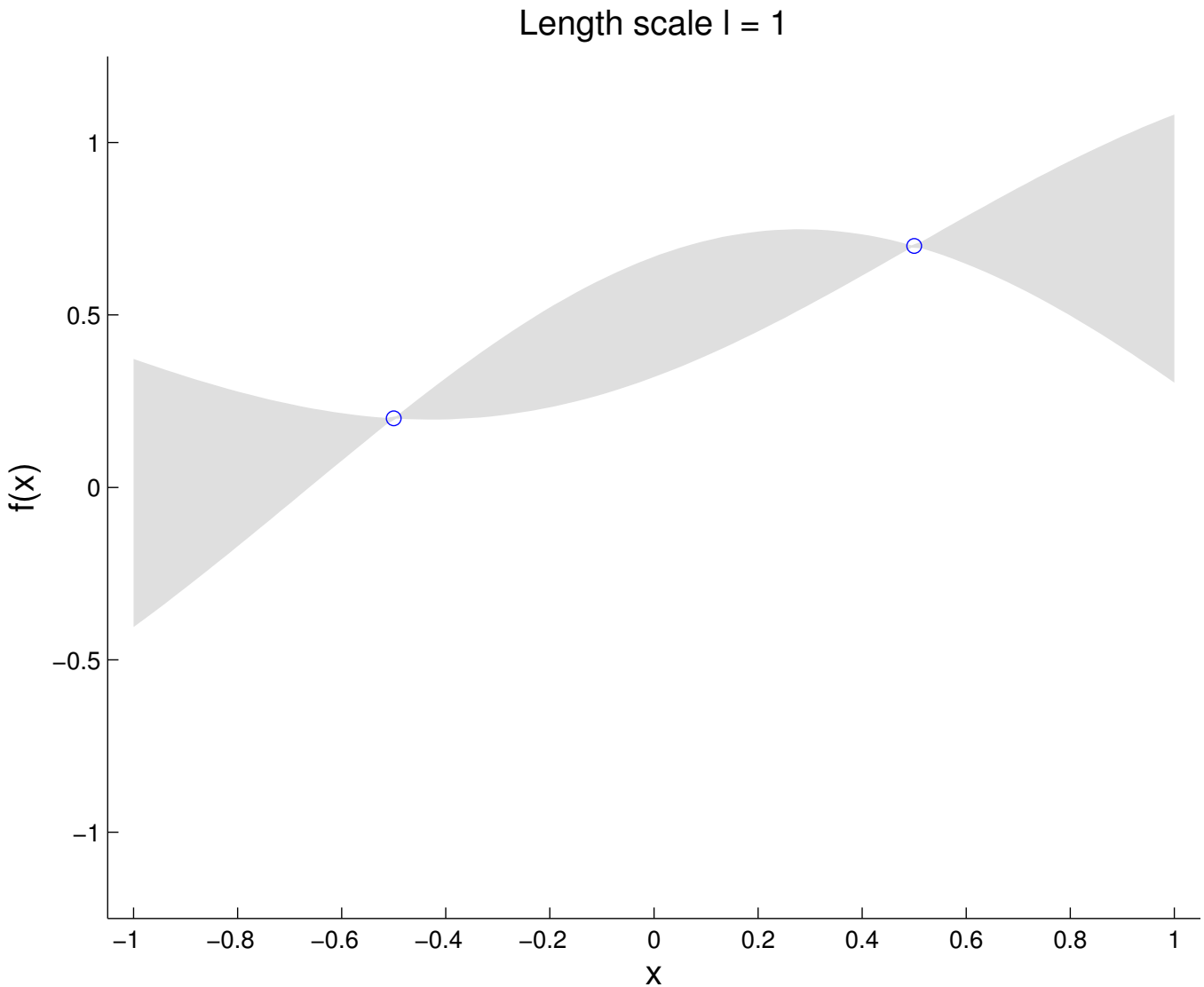
Density before first draw



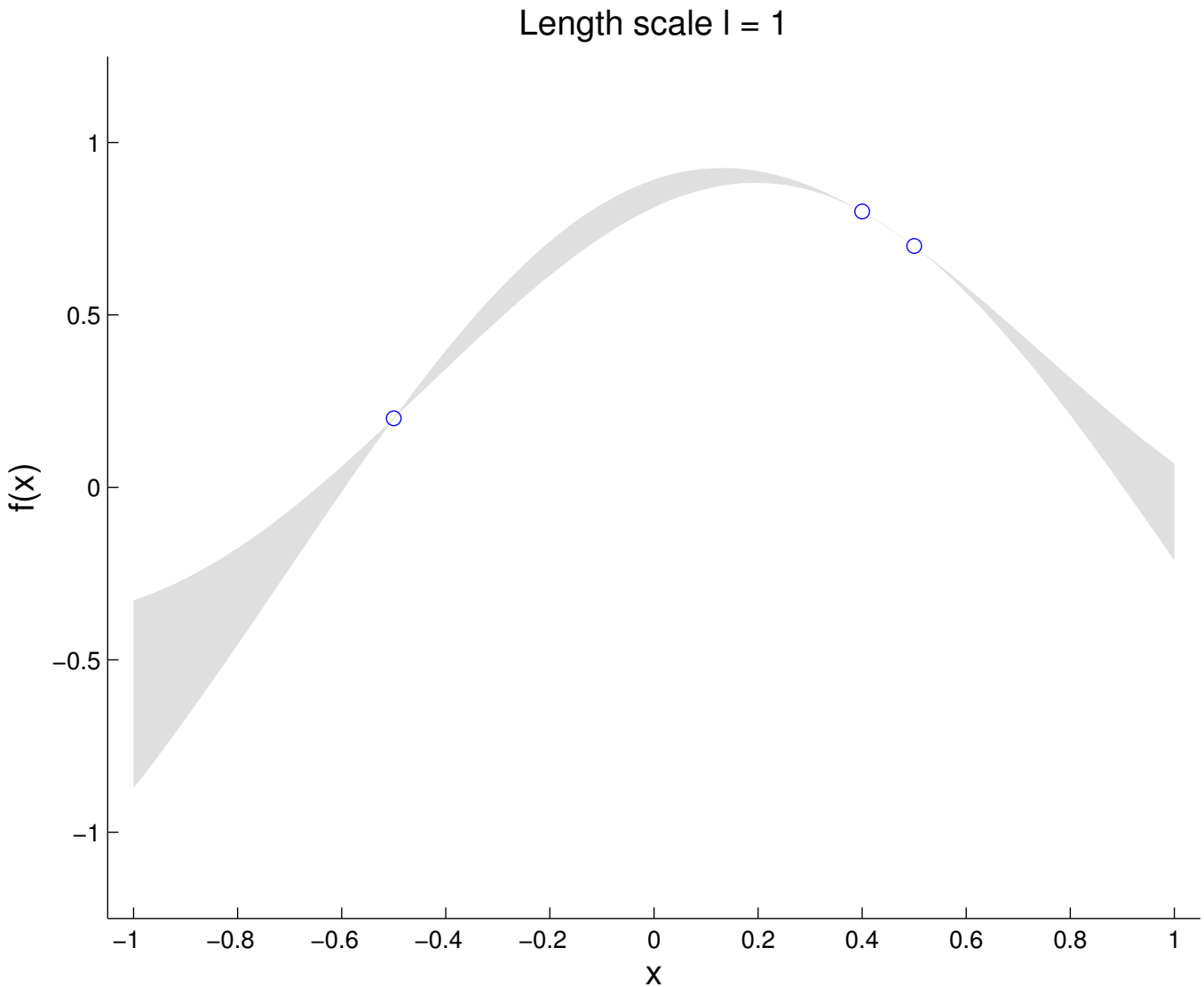
Density before second draw



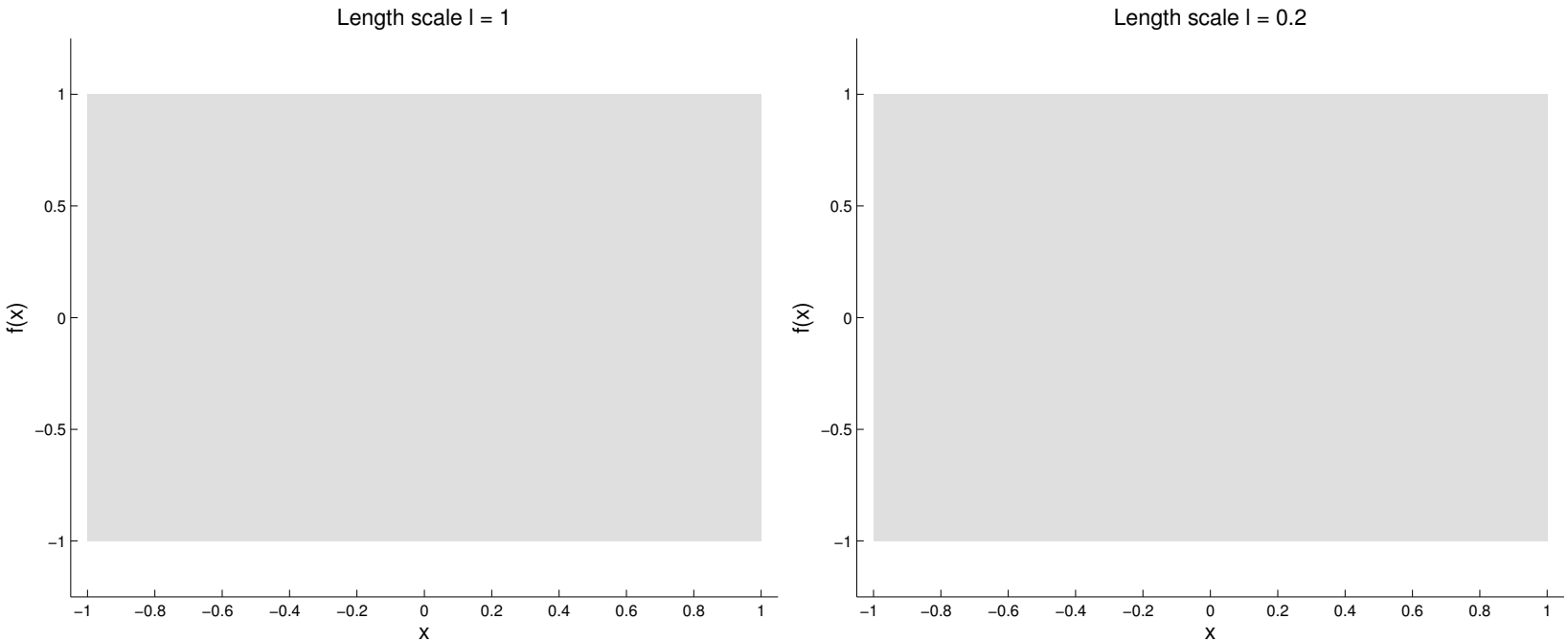
Density before third draw



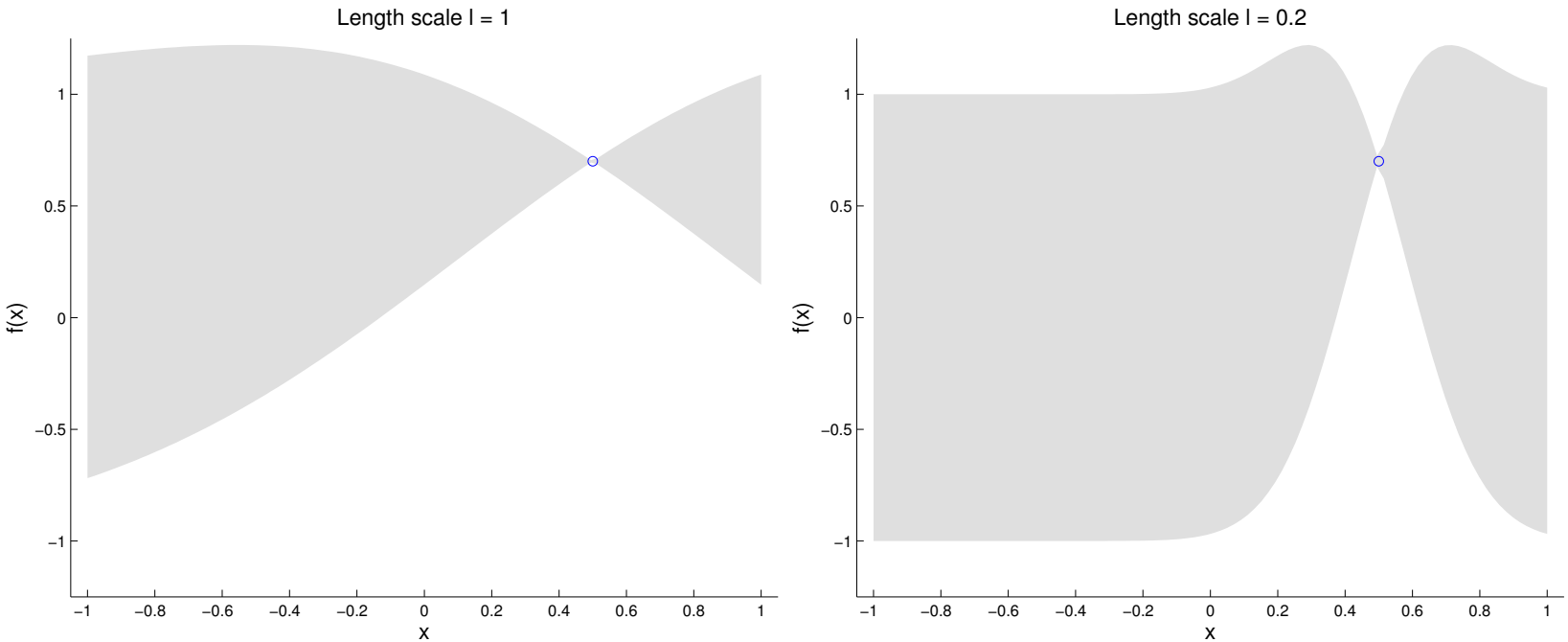
Density before fourth draw



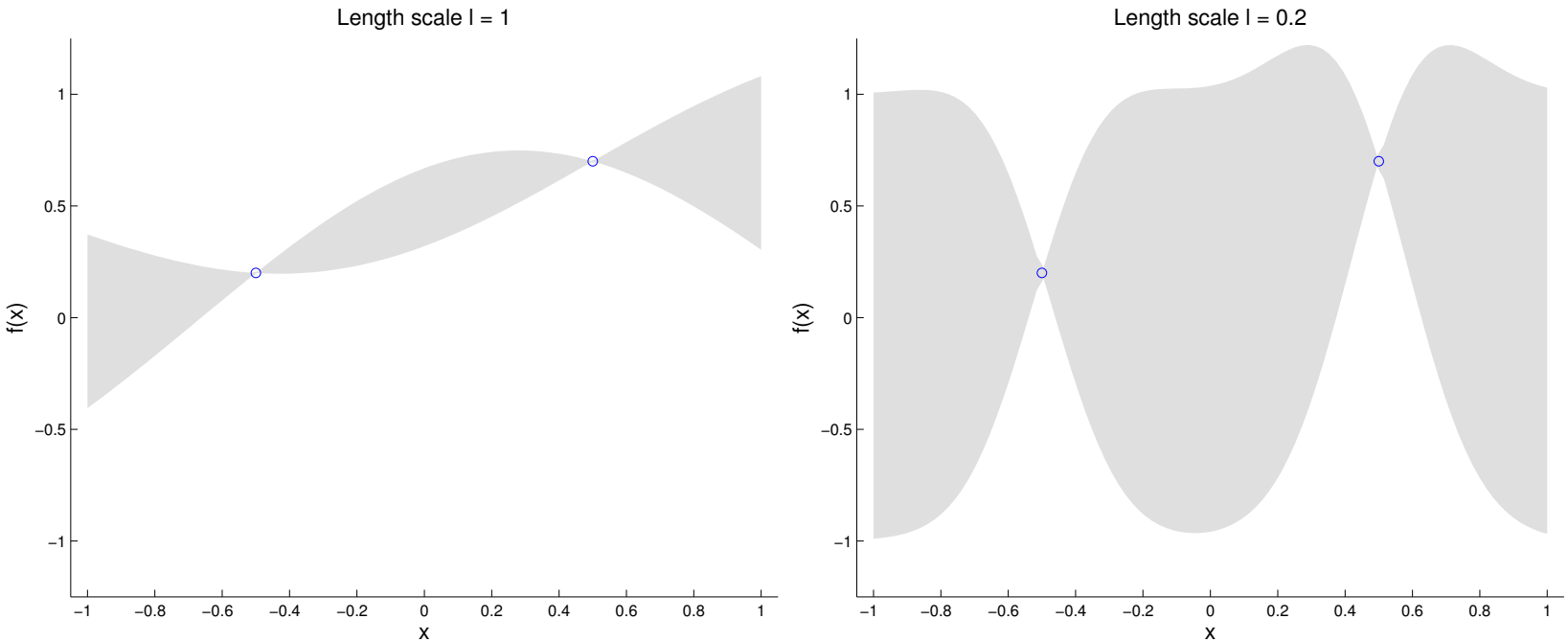
Simulation from l=1 vs l=0.2. Before first draw.



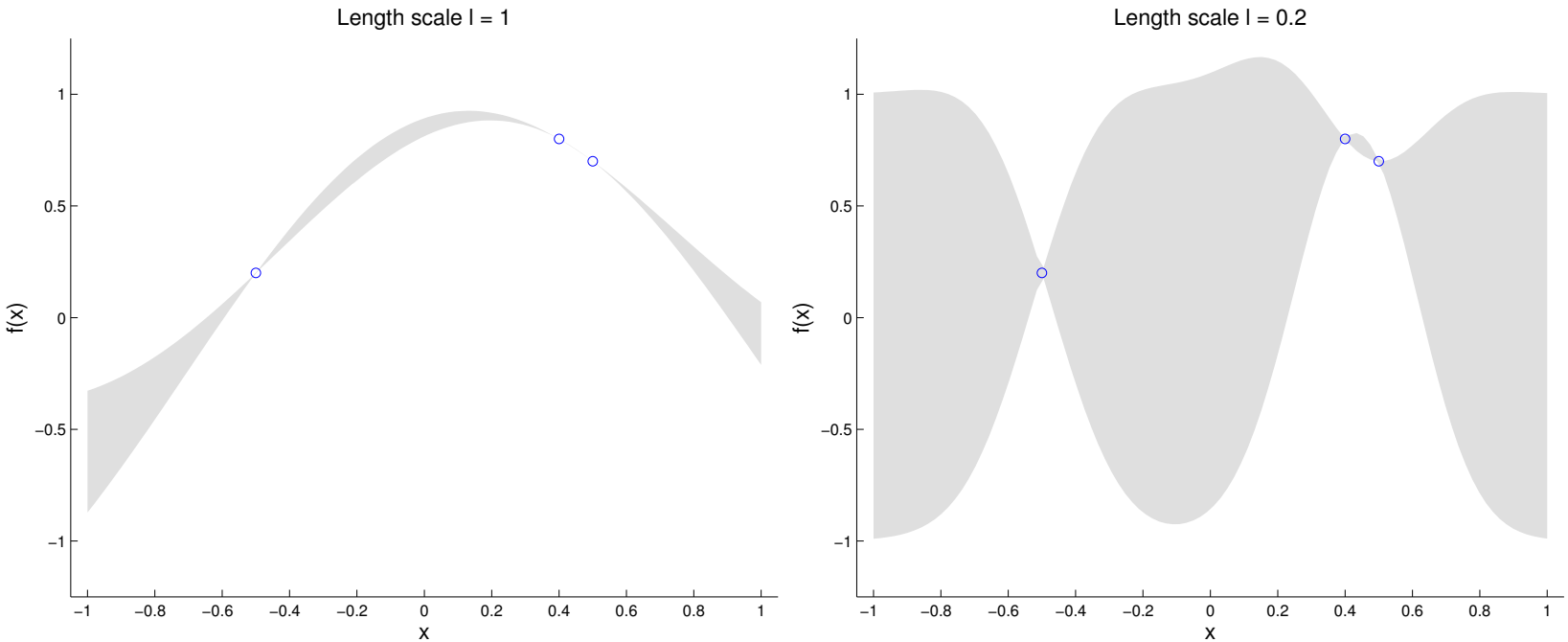
Simulation from l=1 vs l=0.2. Before second draw.



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 GPR

- **Model**

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

- **Prior**

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

- You have observed the data: $\mathbf{x} = (x_1, ..., x_n)'$ and $\mathbf{y} = (y_1, ..., y_n)'$.
- Goal: the posterior of $f(\cdot)$ over a grid of x-values: $\mathbf{f}_* = \mathbf{f}(\mathbf{x}_*)$.

The posterior for a GPR

- **Model**

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

- **Prior**

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

- You have observed the data: $\mathbf{x} = (x_1, ..., x_n)'$ and $\mathbf{y} = (y_1, ..., y_n)'$.
- Goal: the posterior of $f(\cdot)$ over a grid of x-values: $\mathbf{f}_* = \mathbf{f}(\mathbf{x}_*)$.
- Intermediate step: joint distribution of $\mathbf{y}$ and $\mathbf{f}_*$

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

The posterior for a GPR

- **Model**

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

- **Prior**

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

- You have observed the data: $\mathbf{x} = (x_1, ..., x_n)'$ and $\mathbf{y} = (y_1, ..., y_n)'$.
- Goal: the posterior of $f(\cdot)$ over a grid of x-values: $\mathbf{f}_* = \mathbf{f}(\mathbf{x}_*)$.
- Intermediate step: joint distribution of $\mathbf{y}$ and $\mathbf{f}_*$

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

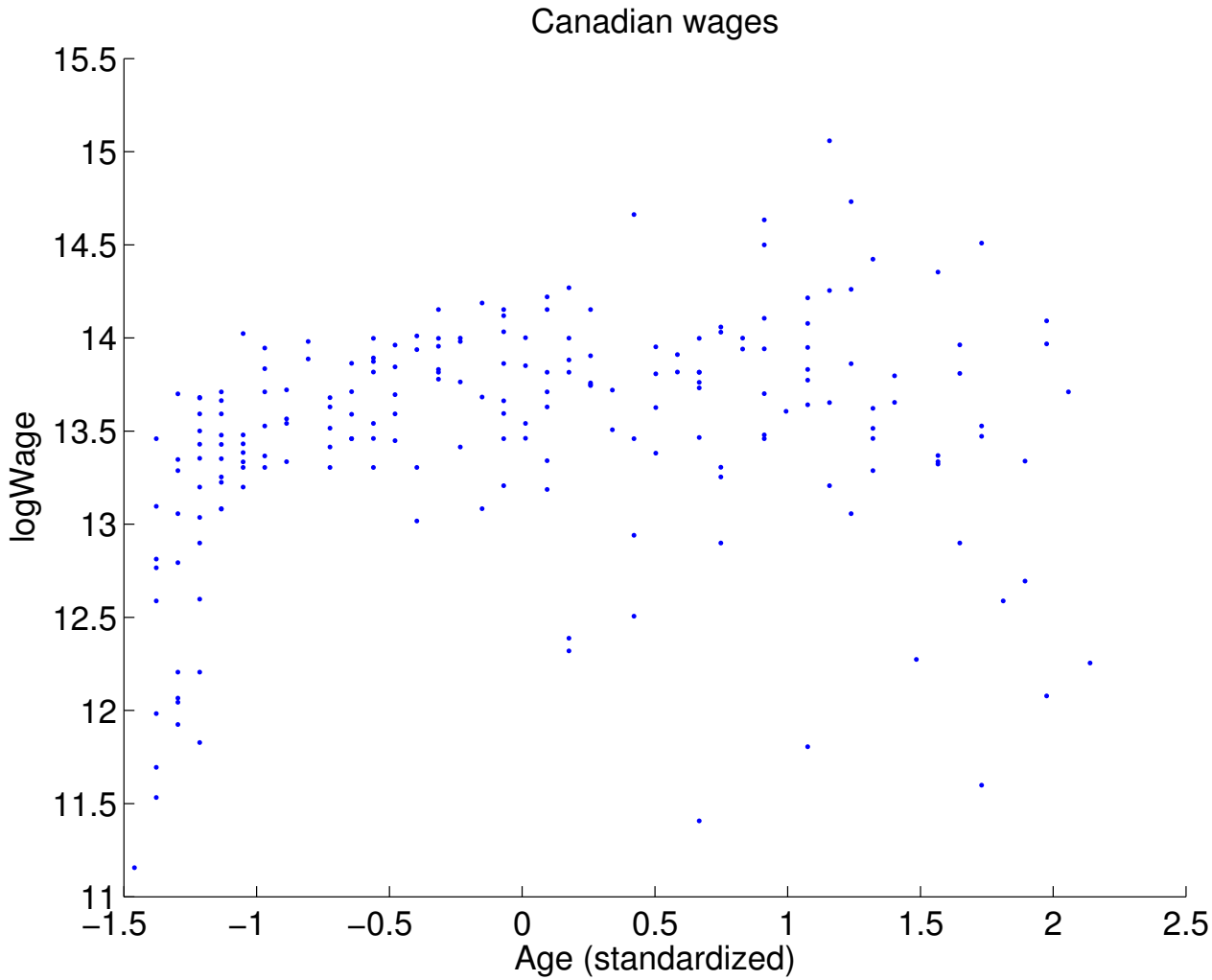
- The **posterior**

$$\mathbf{f}_* | \mathbf{x}, \mathbf{y}, \mathbf{x}_* \sim N\left(\bar{\mathbf{f}}_*, \text{cov}(\mathbf{f}_*)\right)$$

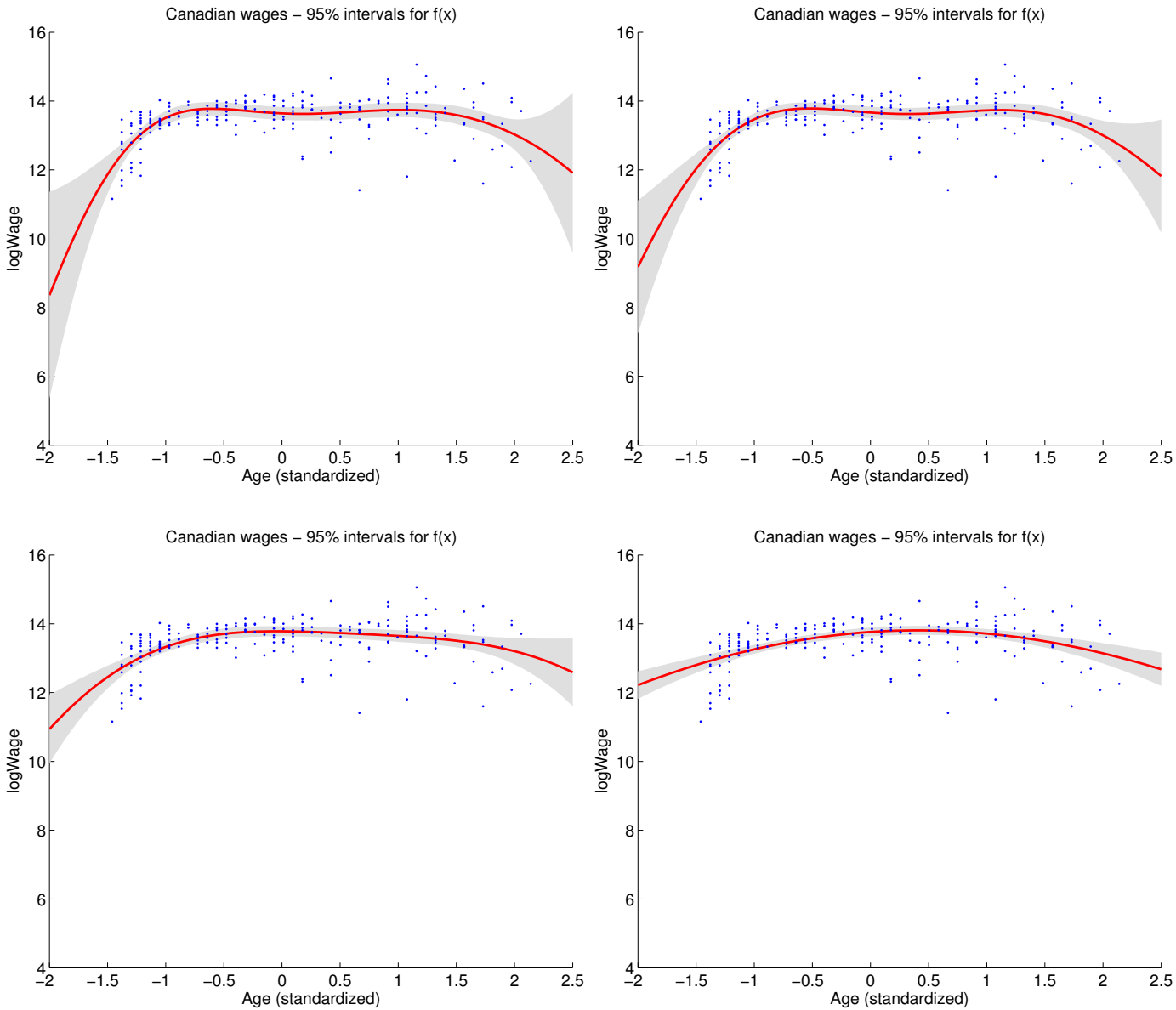
$$\bar{\mathbf{f}}_* = K(\mathbf{x}_*, \mathbf{x}) \left[K(\mathbf{x}, \mathbf{x}) + \sigma^2 I\right]^{-1} \mathbf{y}$$

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

Example - Canadian wages



Posterior of f - $\ell = 0.2, 0.5, 1, 2$



Prediction and Decision

- Predicting a new set of y-values $\mathbf{y}_* = f(\mathbf{x}_*) + \epsilon$ is easy

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

- Choosing a point prediction $\mathbf{y}_{guess}$ by maximizing expected utility

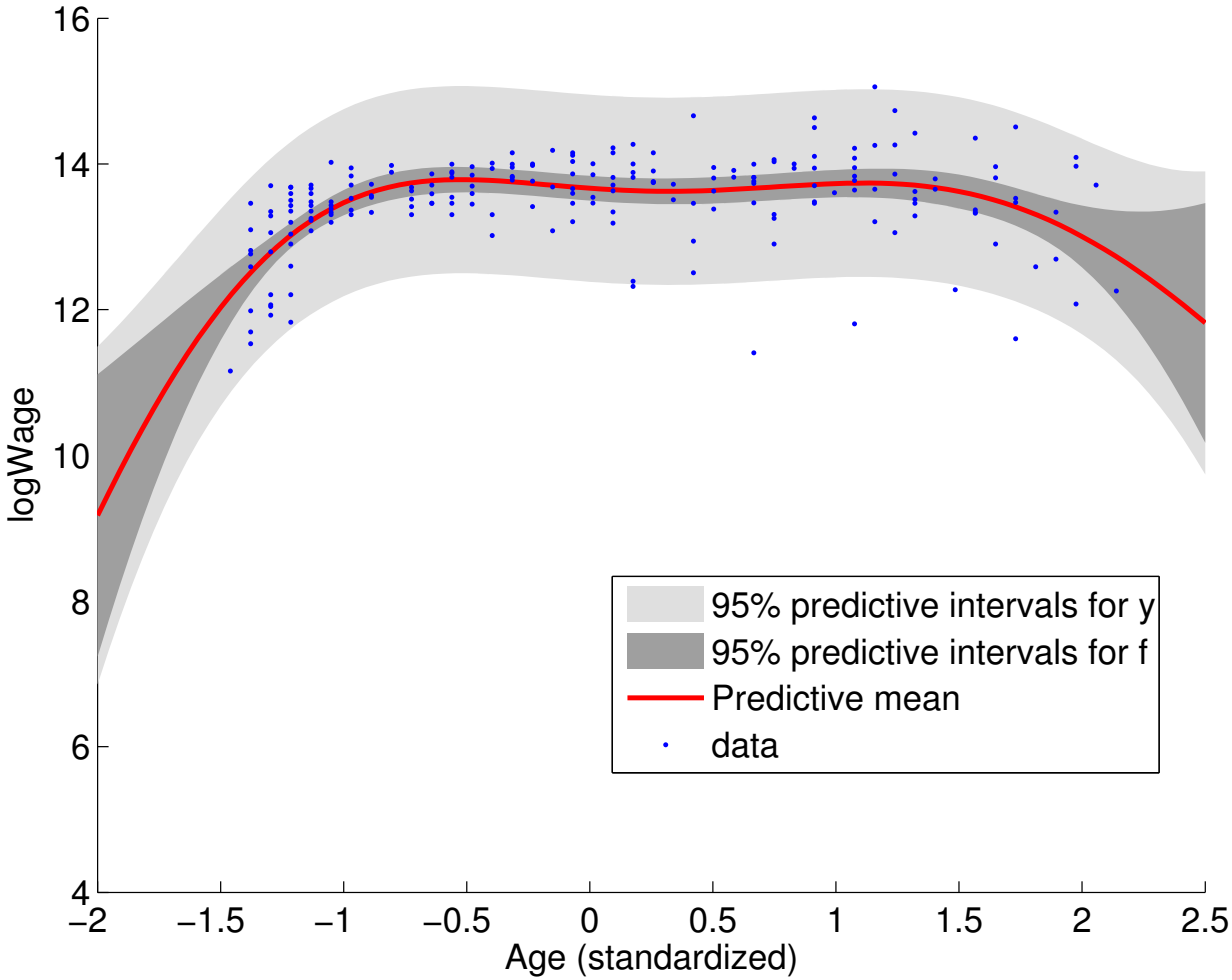
$$\bar{\mathcal{U}}(\mathbf{y}_{guess} | \mathbf{x}_*) = \int \mathcal{U}(\mathbf{y}_*, \mathbf{y}_{guess}) p(\mathbf{y}_* | \mathbf{x}_*, \mathbf{y}, \mathbf{x}) d\mathbf{y}_*$$

- Have to make a decision $a \in \mathcal{A}$ whose consequences (utility) depends on the uncertain $\mathbf{f}_*$ (or $\mathbf{y}_*$)? Just maximize expected utility

$$\bar{\mathcal{U}}(a) = \int \mathcal{U}(a, \mathbf{f}_*) p(\mathbf{f}_* | \mathbf{x}_*, \mathbf{y}, \mathbf{x}) d\mathbf{f}_*$$

where $\mathcal{U}(a, \mathbf{f}_*)$ is the utility of action $a \in \mathcal{A}$ if $\mathbf{f}_*$ turns out to be the "true state of the world".

CANADIAN WAGES - PREDICTION WITH $\ell = 0.5$



Stationary processes and smoothness

- A stochastic process (field) $\{f(\mathbf{x}), x \in \mathbb{R}^D\}$ is **weakly stationary** if $E(f(\mathbf{x})) = \mu$ and its covariance function $k(\mathbf{x}, \mathbf{x}')$ is a function of $\mathbf{t}=\mathbf{x}-\mathbf{x}'$

$$k(\mathbf{x}, \mathbf{x}') = Cov\left[f(\mathbf{x}), f(\mathbf{x}')\right] = k(\mathbf{t}).$$

- The covariance function is **isotropic** if it only depends on the distance $t = \|\mathbf{x}-\mathbf{x}\|$ (invariant to directions)

$$k(\mathbf{x}, \mathbf{x}') = Cov\left[f(\mathbf{x}), f(\mathbf{x}')\right] = k(t).$$

- The **smoothness** of a stationary process is determined by the smoothness of the covariance function.
- A stationary (isotropic) process is **continuous in quadratic mean**

$$E\left(|f(\mathbf{x}+t) - f(\mathbf{x})|^2\right) \to 0 \text{ as } t \to 0$$

iff $k(t)$ is continuous at $t = 0$.

- A little more is required to guarantee **continuous sample paths** (continuous $f(\mathbf{x}+t, \omega)$ for any $\mathbf{x} \in \mathbb{R}^D$ and $\omega \in \Omega$).

Kernels and spectral densities

- $k(\mathbf{x}, \mathbf{x}')$ is a **covariance function** (i.e. positive definite) $\rightarrow$ the $n \times n$ **Gram matrix** $\mathbf{K} = (k(\mathbf{x}_i, \mathbf{x}_j))_{i,j=1,\ldots,n}$ is a **covariance matrix**.
- **Bochner's theorem**: A complex valued function $k(\cdot)$ on $\mathbb{R}^D$ is the covariance function of a weakly stationary continuous complex-valued stochastic process on $\mathbb{R}^D$ iff

$$k(\mathbf{t}) = \int_{\mathbb{R}^D} e^{2\pi i \mathbf{s}\cdot\mathbf{t}} S(\mathbf{s}) d\mathbf{s}$$

- $S(\mathbf{s})$ is the **spectral density**. $S(\mathbf{s})$ is the energy allocated to the complex exponential basis function $e^{2\pi i \mathbf{s}\cdot\mathbf{t}}$ at frequency $\mathbf{s}$.
- For real-valued processes, think of $e^{2\pi i \mathbf{s}\cdot\mathbf{t}}$ as a multi-dimensional sine wave with frequency $\mathbf{s}$ and amplitude $S(\mathbf{s})$.
- Spectral density $\Longleftrightarrow$Covariance function of stationary process $\Longleftrightarrow$Smoothness properties of the process.

Commonly used covariance kernels

- Let $r = \|x - x'\|$. All kernels can be scaled by $\sigma_f > 0$.
- **Squared exponential (SE)** ($\ell > 0$)

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

  - Spectral density $S(s) = (2\pi\ell^2)^{D/2} \exp\left(-2\pi^2\ell^2 s^2\right)$. Higher frequencies tail of like a Gaussian (i.e. quickly).
  - Infinitely mean square differentiable. Very smooth.
- **Matérn** ($\ell > 0$, $\nu > 0$)

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

  - Spectral density behaves like a student-t density with 2ν degrees of freedom. For $\nu = 1/2$, $S(s)$ is Cauchy. Much weight on high frequency. Very rough.
  - $\nu = 3/2$ and $\nu = 5/2$ most useful for ML. As $\nu \to \infty$, Matérn's kernel approaches SE kernel.

Commonly used covariance kernels, cont.

- **γ-exponential** ($\ell > 0$, $0 < \gamma \leq 2$)

$$K_\gamma(r) = \exp\left[-\left(\frac{r}{\ell}\right)^\gamma\right]$$

  - Mean square differentiable only when $\gamma = 2$ (SE).

- **Rational quadratic** ($\ell > 0$, $\alpha > 0$)

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

  - Scale mixture of SE covariance functions with different length-scales.
  - $K_{RQ}(r)$ approaches the SE kernel as $\alpha \to \infty$.

- $k(r) = \int \exp\left(-r^2/2\ell^2\right) p(\ell) d\ell$ is the most general represenation of an isotropic kernel with a valid covariance function in all dimensions D.

More on kernels

- 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.
- **Factor kernels**: $M = \Lambda\Lambda^T + \Psi$, where Λ is $D \times k$ for low rank k.
- Length-scales $\ell(\mathbf{x})$ that vary with $\mathbf{x}$. Non-trivial to make positive definite, but see Gibbs kernel in Eq. 4.32.
- Kernels are often combined into **composite kernels**. Sum of kernels is a kernel. Product of kernels is a kernel.
- Kernels can be used for non-vectorial inputs by defining distance function between objects (e.g. words). String kernels for text analysis. Fisher kernels.

Bayesian inference for hyperparameters

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

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

- If the hyperparameters are unknown, just compute the posterior

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

- We need to compute

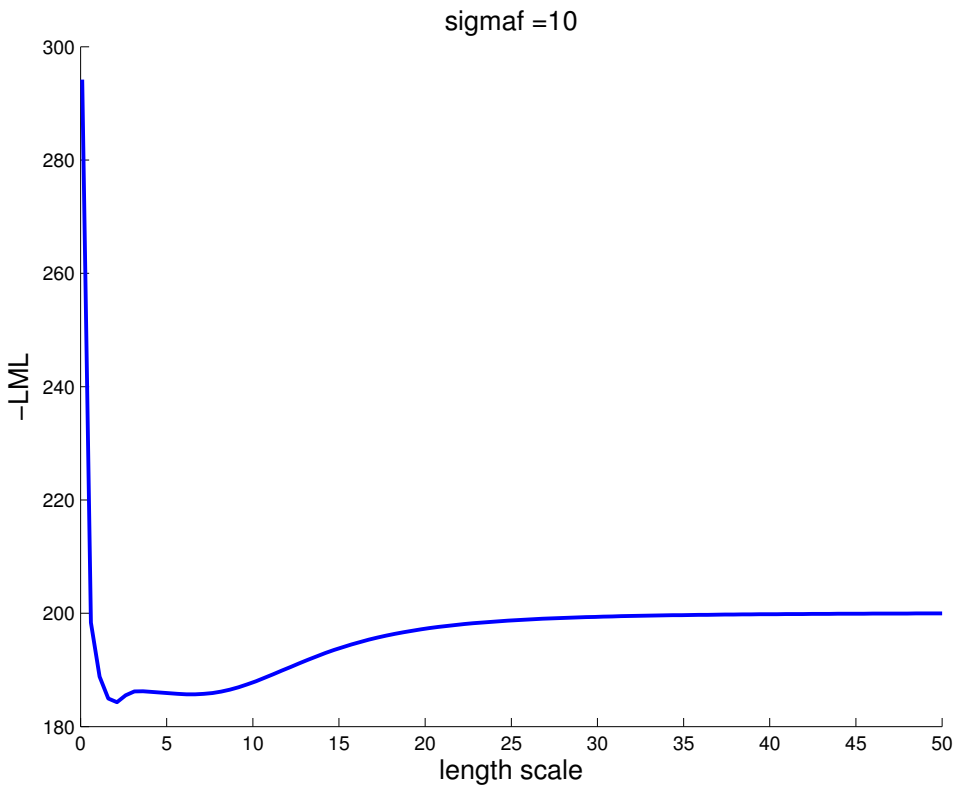
$$p(\mathbf{y}|\mathbf{X}, \theta) = \int p(\mathbf{y}|\mathbf{X}, \mathbf{f}, \theta)p(\mathbf{f}|\mathbf{X}, \theta)d\mathbf{f}$$

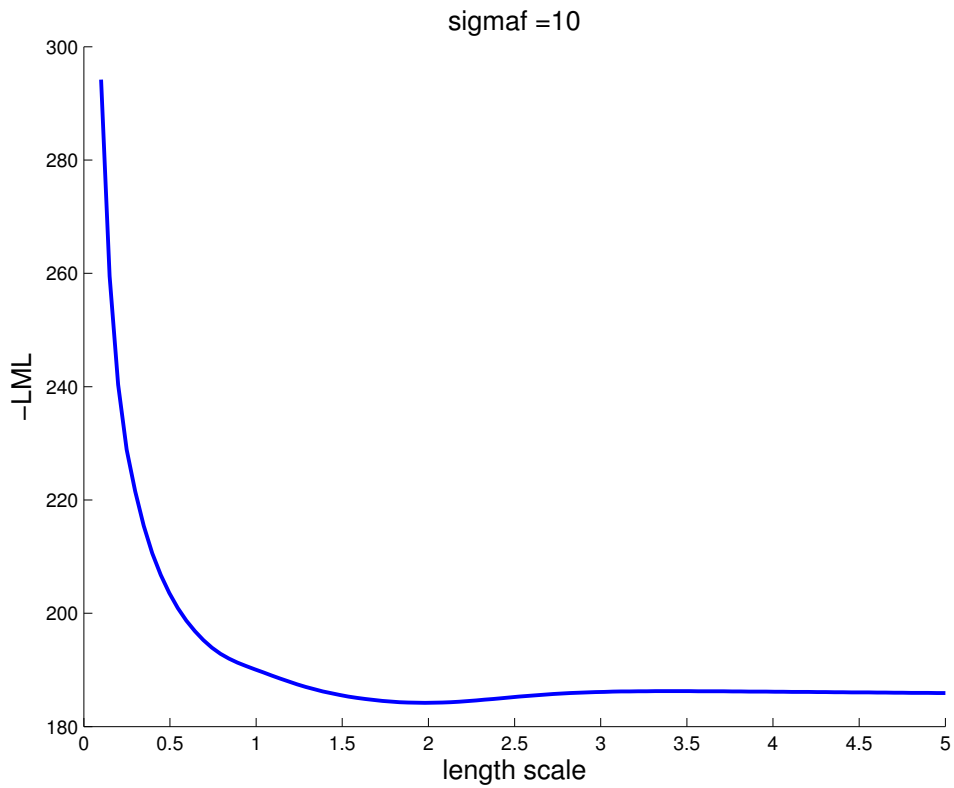
  where $\mathbf{f} = f(\mathbf{X})$ is a vector with function values in the training data.
- For Gaussian process regression we can actually do this analytically

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

- RW takes a short-cut and estimates θ by maximizing $\log p(\mathbf{y}|\mathbf{X}, \theta)$.

Canadian wages - LML determination of ℓ





Classification

- **Classification**: **binary response** $y \in \{-1, 1\}$ (or multi-class $y \in \{1, 2, ..., C\}$) explained/predicted by covariates/features $\mathbf{x}$.
- Aim: posterior distribution $p(y|\mathbf{x})$
- **Generative approach**

$$p(y|\mathbf{x}) \propto p(\mathbf{x}|y)p(y)$$

  Need to model the class-conditional distributions $p(\mathbf{x}|y)$ for each y.
- **Discriminative approach** models the posterior $p(y|\mathbf{x})$ directly.
  Example: linear logistic regression

$$Pr(y = 1|\mathbf{x}) = \lambda(\mathbf{x}^T\mathbf{w})$$

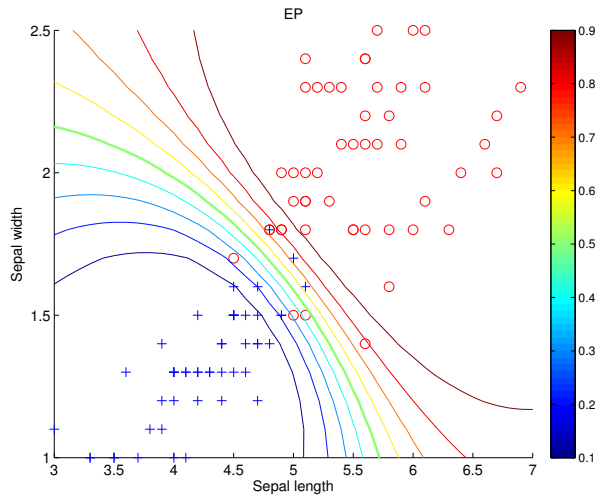
  where $\lambda(z)$ is the logistic **link function**

$$\lambda(z) = \frac{1}{1 + \exp(-z)}$$

- $\lambda(z)$ 'squashes' the linear prediction $\mathbf{x}^T\mathbf{w} \in \mathbb{R}$ into $\lambda(\mathbf{x}^T\mathbf{w}) \in [0, 1]$.
- Can also use the normal CDF $\Phi(z)$ for squashing. **Probit regression**.

Classification

- The posterior of the label $p(y|\mathbf{x})$ is clearly a probabilistic classifier.
- Visual representation: level contours of $p(y|\mathbf{x})$ over $\mathbf{x}$-space.



- Decision boundaries.
- **Zero-one loss** $\Rightarrow$ allocate to class c which maximizes $Pr(y = c|\mathbf{x})$.
- **Reject option**. Refuse to classify when $Pr(y = c|\mathbf{x})$ are small for all c.

GP CLASSIFICATION

- **Linear logistic regression**

$$Pr(y = 1|\mathbf{x}) = \lambda(\mathbf{x}^T\mathbf{w})$$

has linear decision boundaries (conditional on $\mathbf{w}$).

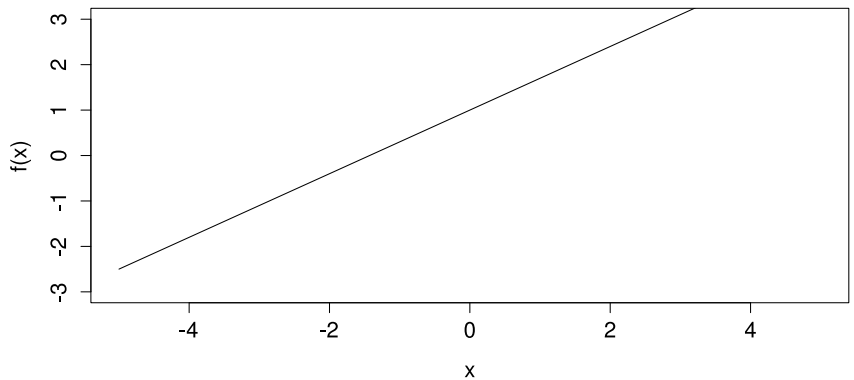
- Obvious **GP extension**: replace $\mathbf{x}^T\mathbf{w}$ by

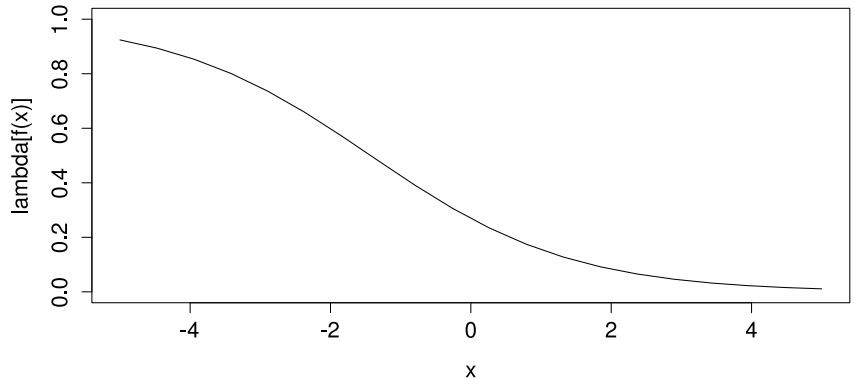
$$f(\mathbf{x}) \sim GP(0, k(\mathbf{x}, \mathbf{x}'))$$

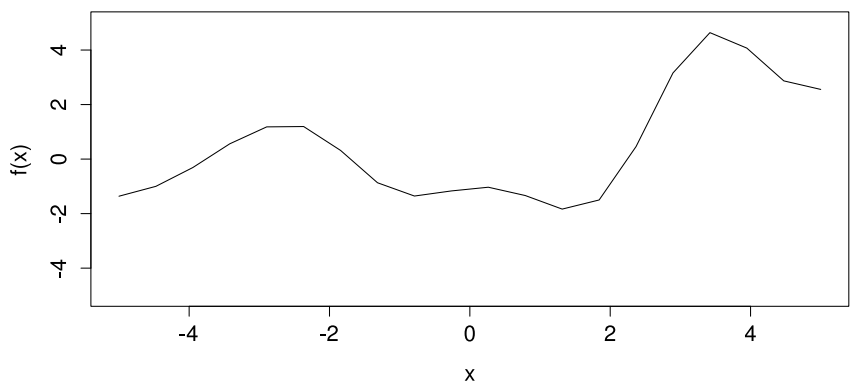
and squash f through logistic function (or normal CDF)

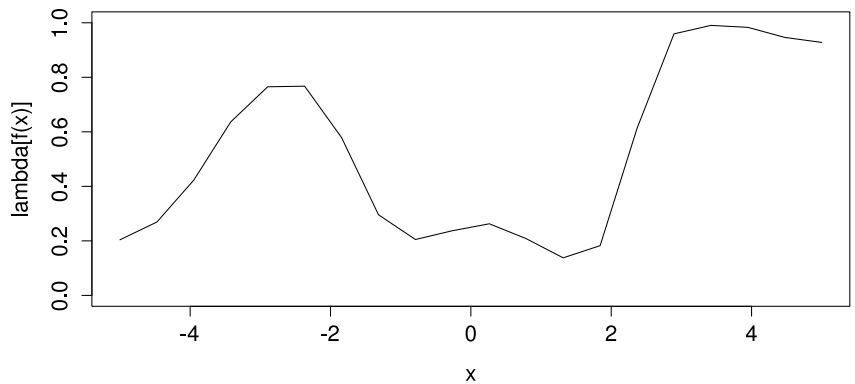
$$Pr(y = 1|\mathbf{x}) = \lambda(f(\mathbf{x}))$$

SQUASHING F


f(x)
-3
-2
-1
0
1
2
3
-4
-2
0
2
4
x


lambda[f(x)]
0.0
0.2
0.4
0.6
0.8
1.0
-4
-2
0
2
4
x


f(x)
-4
-2
0
2
4
-4
-2
0
2
4
x


lambda[f(x)]
0.0
0.2
0.4
0.6
0.8
1.0
-4
-2
0
2
4
x

GP CLASSIFICATION - INFERENCE

- Prediction for a test case $\mathbf{x}_*$:

$$Pr(y_* = +1|\mathbf{X}, \mathbf{y}, \mathbf{x}_*) = \int \sigma(f_*) p(f_*|\mathbf{X}, \mathbf{x}_*, \mathbf{f}) df_*$$

  where $\sigma(f_*)$ is some sigmoidal function (logistic, normal CDF...) and f_* is the latent f at the test input $\mathbf{x}_*$.

- The predictive distribution of f_* is

$$p(f_*|\mathbf{X}, \mathbf{x}_*, \mathbf{f}) = \int p(f_*|\mathbf{X}, \mathbf{x}_*, \mathbf{f}) p(\mathbf{f}|\mathbf{X}, \mathbf{y}) d\mathbf{f}$$

  where

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

  is the posterior of $\mathbf{f}$ from the training data.

- Note that $p(\mathbf{y}|\mathbf{f})$ is no longer Gaussian in classification problems. Posterior $p(\mathbf{f}|\mathbf{X}, \mathbf{y})$ is not analytically tractable.

The Laplace approximation

- Approximates $p(\mathbf{f}|\mathbf{X},\mathbf{y})$ with $N(\hat{\mathbf{f}},\mathbf{A}^{-1})$, where $\hat{\mathbf{f}}$ is the posterior mode and $\mathbf{A}$ is the negative Hessian of the log posterior at $\mathbf{f}=\hat{\mathbf{f}}$.
- The log posterior is (proportional to)

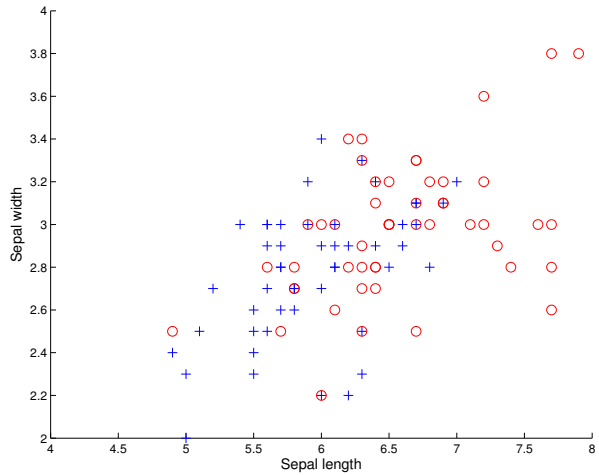
$$\begin{aligned}\Psi(\mathbf{f}) &= \log p(\mathbf{y}|\mathbf{f}) + \log p(\mathbf{f}|\mathbf{X}) \\ &= \log p(\mathbf{y}|\mathbf{f}) - \frac{1}{2}\mathbf{f}^T K^{-1}\mathbf{f} - \frac{1}{2}\log|K| - \frac{n}{2}\log 2\pi\end{aligned}$$

- Differentiating wrt $\mathbf{f}$

$$\begin{aligned}\nabla\Psi(\mathbf{f}) &= \nabla \log p(\mathbf{y}|\mathbf{f}) - K^{-1}\mathbf{f} \\ \nabla\nabla\Psi(\mathbf{f}) &= \nabla\nabla \log p(\mathbf{y}|\mathbf{f}) - K^{-1} = -W - K^{-1}\end{aligned}$$

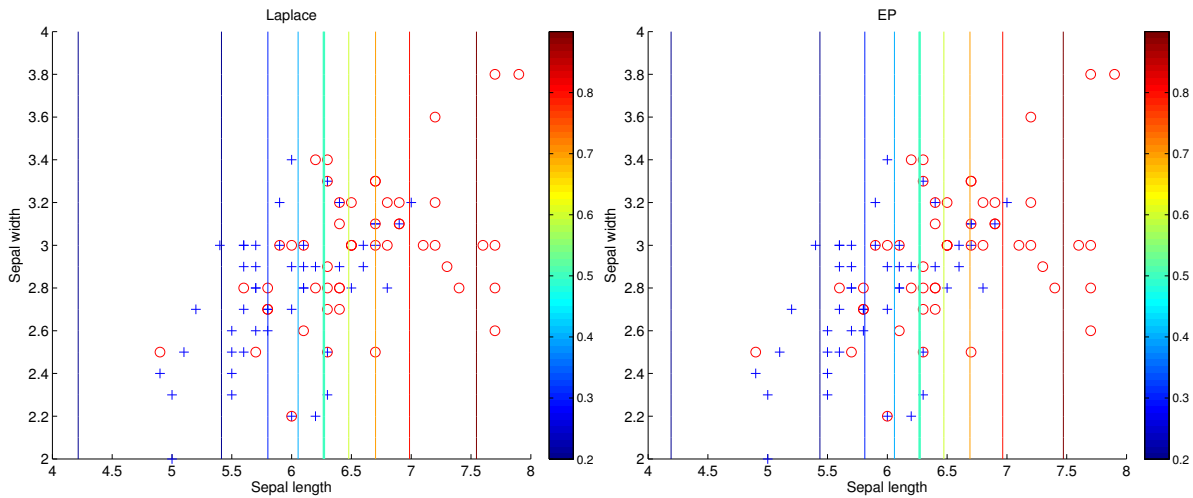
  where W is a diagonal matrix since each y_i only depends on its f_i.
- Use **Newton's method** to iterate to the mode.
- The elements $\partial/\partial f_i \log p(y_i|f_i)$ and $\partial^2/\partial f_i^2 \log p(y_i|f_i)$ are given in the table on page 43 in RW for the logistic and probit cases.
- **Approximate predictions** of $f_*$are possible. Predictions of y_* require one-dimensional numerical integration.

IRIS DATA - SEPAL - SE KERNEL WITH ARD

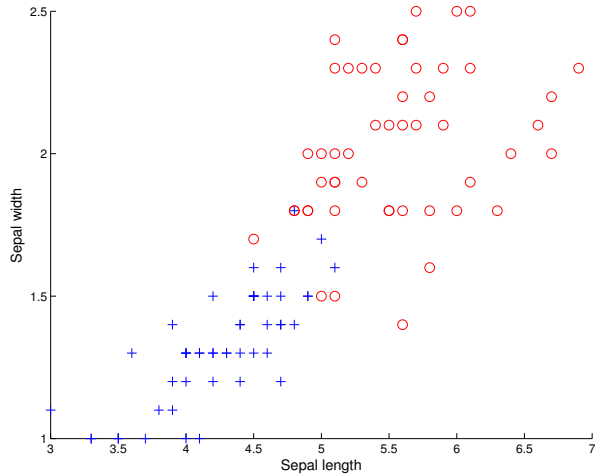


Laplace: $\hat{\ell}_1 = 1.7214, \hat{\ell}_2 = 185.5040, \sigma_f = 1.4361$

EP: $\hat{\ell}_1 = 1.7189, \hat{\ell}_2 = 55.5003, \sigma_f = 1.4343$

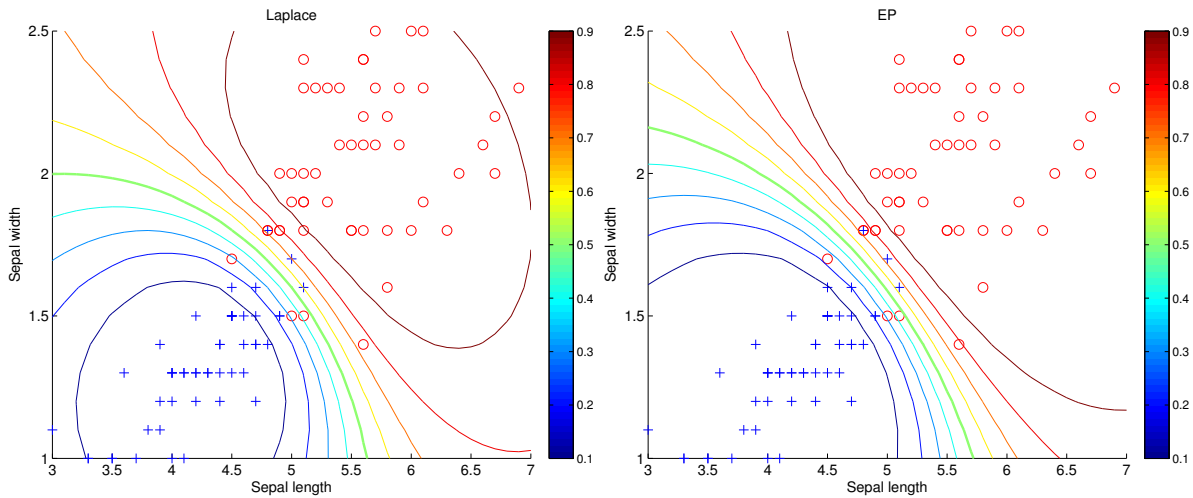


IRIS DATA - PETAL - SE KERNEL WITH ARD

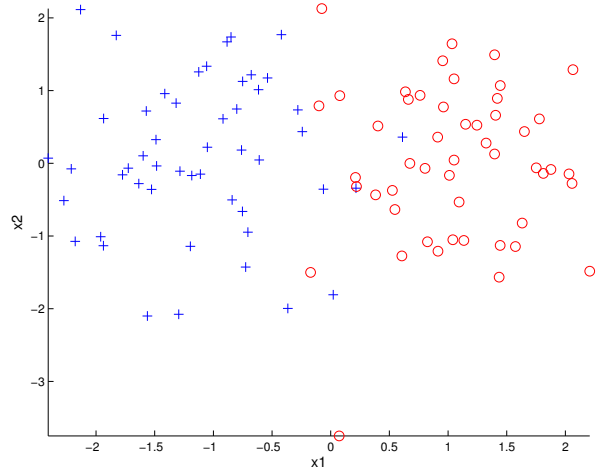


Laplace: $\hat{\ell}_1 = 1.7606, \hat{\ell}_2 = 0.8804, \sigma_f = 4.9129$

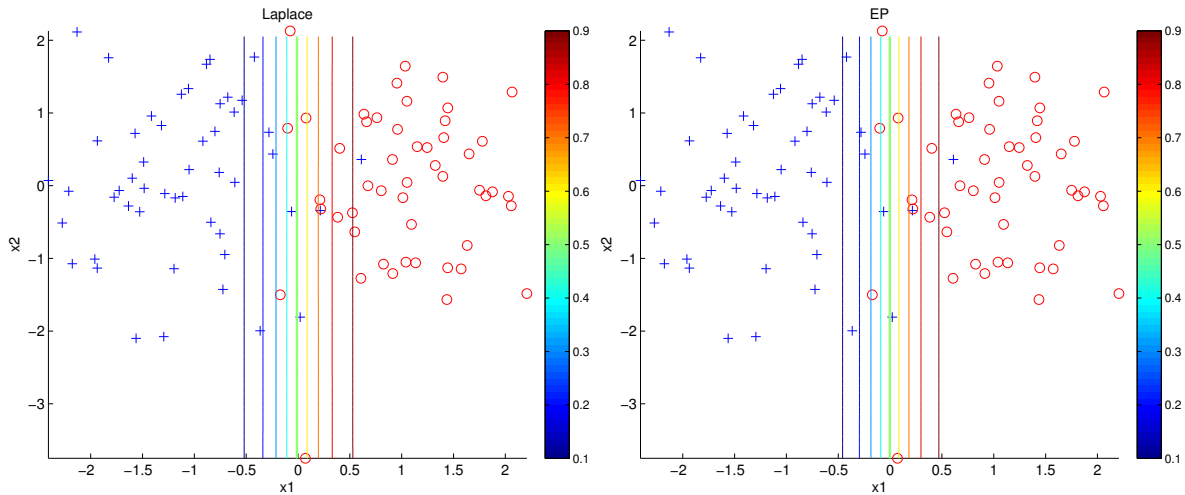
EP: $\hat{\ell}_1 = 2.1139, \hat{\ell}_2 = 1.0720, \sigma_f = 5.3369$



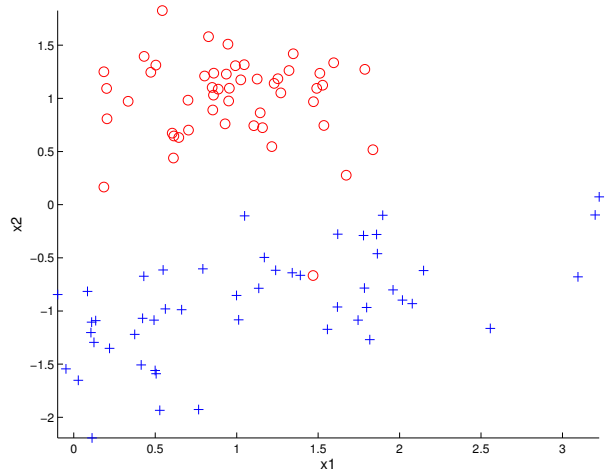
Toy data 1 - SE kernel with ARD



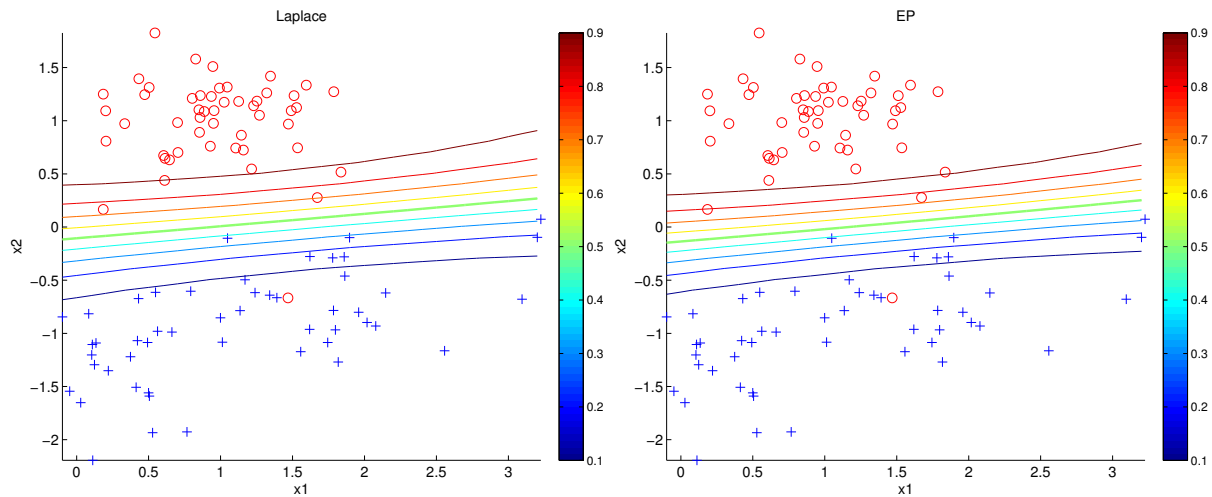
EP: $\hat{\ell}_1 = 2.4503, \hat{\ell}_2 = 721.7405, \sigma_f = 4.7540$



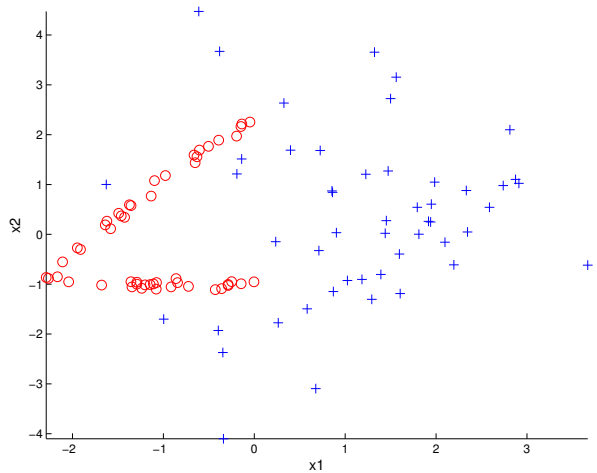
TOY DATA 2 - SE KERNEL WITH ARD



EP: $\hat{\ell}_1 = 8.3831, \hat{\ell}_2 = 1.9587, \sigma_f = 4.5483$



TOY DATA 3 - SE KERNEL WITH ARD



Laplace: $\hat{\ell}_1 = 0.7726, \hat{\ell}_2 = 0.6974, \sigma_f = 11.7854$

EP: $\hat{\ell}_1 = 1.2685, \hat{\ell}_2 = 1.0941, \sigma_f = 17.2774$