Lecture 9: Gaussian Processes

Thibault Randrianarisoa

UTSC

March 26, 2026

Motivation

- In a parametric model, the model is represented using parameters
- a distribution over parameters can imply a distribution over functions (e.g. Bayesian linear regression)
- In Bayesian inference, we marginalize over parameters to make predictions
- Question: could we work directly in the space of functions?

Priors on parameters induce priors on functions

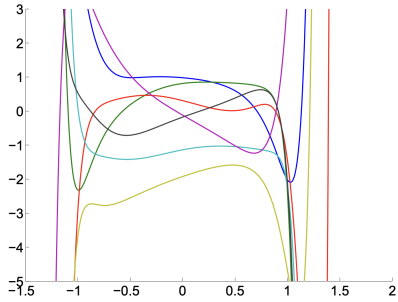
A model M is the choice of a structure and of parameter values.

$$f_{w|M}(x) = \sum_{j=1}^{J} w_j \phi_j(x)$$

A prior $p(w|M)$ determines what functions this model can generate.

Example:

- Imagine we choose $J = 17$, and independent $w_j \sim \mathcal{N}(0, \sigma_w^2)$.
- Use polynomial basis functions. $\phi_j(x) = x^j$.
- We have actually defined a prior distribution over functions $p(f|M)$.



Nuissance parameters and distributions over functions

We've seen that distributions over parameters can induce distributions over functions.

We've set up a schema where we

- first set up a model in terms a parameters
- then marginalize out the parameters

Typically, we're not really interested in parameters, we're interested in predictions.
Meaning, the parameters are a nuisance.

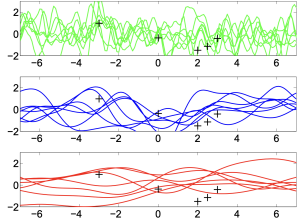
Could we possibly work directly in the space of functions?

- simpler inference (no need for marginalization over params)
- better understanding of the distributions over functions

A prior over functions view

We have learnt that linear-in-the-parameter models with priors on the weights indirectly specify priors over functions.

True. . . but those priors over functions might not be good.



. . . why not try to specify priors over functions directly?

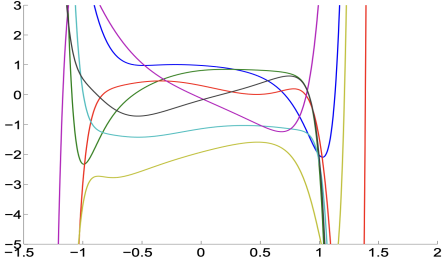
What does a probability density over functions even look like?

Posterior probability of a function

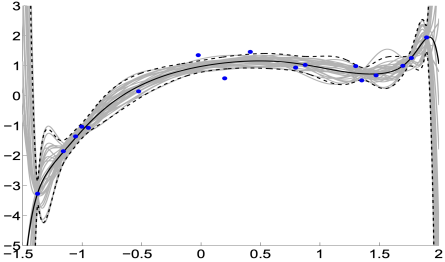
Given the prior functions $p(f)$ how can we make predictions and the function itself?

- Informally, of all functions generated from the prior, keep those that "nearly" fit the data.
- As before, the notion of closeness to the data is given by the likelihood $p(y|f)$.
- We are really interested in the posterior distribution over functions:

$$p(f|y) = \frac{p(y|f)p(f)}{p(y)} \quad \text{Bayes Rule}$$



Some samples from the prior



Samples from the posterior

Reminder: Conditionals and Marginals of a Gaussian

If $\mathbf{x}$ and $\mathbf{y}$ are jointly Gaussian

$$p(\mathbf{x}, \mathbf{y}) = p\left(\begin{bmatrix}\mathbf{x} \\ \mathbf{y}\end{bmatrix}\right) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{a} \\ \boldsymbol{b}\end{bmatrix}, \begin{bmatrix}A & B \\ B^\top & C\end{bmatrix}\right),$$

we get the marginal distribution of $\mathbf{x}$, $p(\mathbf{x})$ by

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{a} \\ \boldsymbol{b}\end{bmatrix}, \begin{bmatrix}A & B \\ B^\top & C\end{bmatrix}\right) \implies p(\mathbf{x}) = \mathcal{N}(\boldsymbol{a}, A),$$

and the conditional distribution of $\mathbf{x}$ given $\mathbf{y}$ by

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{a} \\ \boldsymbol{b}\end{bmatrix}, \begin{bmatrix}A & B \\ B^\top & C\end{bmatrix}\right) \implies p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\boldsymbol{a} + BC^{-1}(\mathbf{y} - \boldsymbol{b}), A - BC^{-1}B^\top),$$

where $\mathbf{x}$ and $\mathbf{y}$ can be scalars or vectors.

What is a Gaussian Process?

A *Gaussian process* is a generalization of a multivariate Gaussian distribution to infinitely many variables.
Informally: infinitely long vector $\simeq$ function

Definition

A **Gaussian process** is a collection of random variables, indexed by $\mathbf{x} \in \mathcal{X}$, any finite number of which have (consistent) Gaussian distributions.

- A Gaussian distribution is fully specified by a mean vector, $\boldsymbol{\mu}$, and covariance matrix Σ.
- A Gaussian process is fully specified by a mean function $m(\mathbf{x})$ and covariance function $k(\mathbf{x}, \mathbf{x}')$:

  $$f \sim \mathcal{N}(m, k)$$

  Here f and m are functions on $\mathcal{X}$, and k is a function on $\mathcal{X} \times \mathcal{X}$

The marginalization property

Thinking of a GP as a Gaussian distribution with an infinitely long mean vector and an infinite by infinite covariance matrix may seem impractical... Luckily we are saved by the marginalization property:

For Gaussians:

$$p(\mathbf{x}, \mathbf{y}) = \mathcal{N}\left(\begin{bmatrix}\boldsymbol{a}\\ \boldsymbol{b}\end{bmatrix}, \begin{bmatrix}A & B\\ B^\top & C\end{bmatrix}\right) \implies p(\mathbf{x}) = \mathcal{N}(\boldsymbol{a}, A),$$

which works irrespective of the size of $\mathbf{y}$.

For Gaussian processes:

$$f \sim \mathcal{N}(m, k) \implies \mathbf{f} = f(\mathbf{x}) \sim \mathcal{N}(\boldsymbol{\mu} = \mathbf{m} = m(\mathbf{x}), \Sigma = K(\mathbf{x}, \mathbf{x})).$$

Key: only ever ask finite dimensional questions about functions.

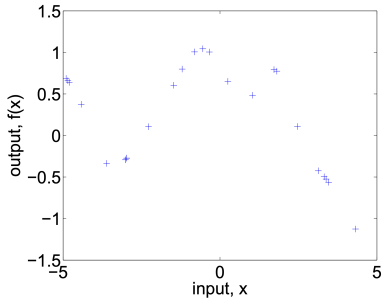
Random functions from a Gaussian Process

Example: $p(f) \sim \mathcal{N}(m, k)$, where $m(x) = 0$, and $k(x, x') = \exp(-\frac{1}{2}(x - x')^2)$.

To get an indication of what this distribution over functions looks like, focus on a finite subset of function values $\mathbf{f} = (f(x_1), f(x_2), \ldots, f(x_N))^\top$, for which

$$\mathbf{f} \sim \mathcal{N}(0, \Sigma), \text{ where } \Sigma_{ij} = k(x_i, x_j).$$

Then plot the coordinates of $\mathbf{f}$ as a function of the corresponding x values.



Joint Generation

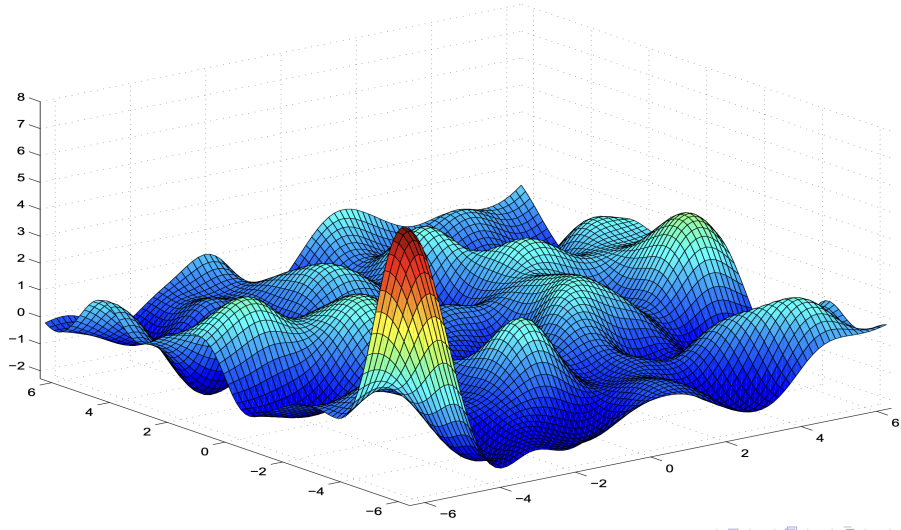
To generate a random sample from a D-dimensional joint Gaussian with covariance matrix K and mean vector $\mathbf{m}$:

1. Draw a vector of independent standard normal random variables, $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, I_D)$.
2. Compute the Cholesky factorization of the covariance matrix K such that $K = R^\top R$, where R is an upper triangular matrix.
3. Apply the affine transformation:
$$\mathbf{y} = R^\top \mathbf{z} + \mathbf{m}$$

Thus, the covariance of the resulting vector $\mathbf{y}$ is:

$$\mathbb{E}[(\mathbf{y} - \mathbf{m})(\mathbf{y} - \mathbf{m})^\top] = \mathbb{E}[R^\top \mathbf{z}\mathbf{z}^\top R] = R^\top \mathbb{E}[\mathbf{z}\mathbf{z}^\top] R = R^\top I_D R = K.$$

Function drawn at random from a Gaussian Process with Gaussian covariance



Recall Bayesian inference in a parametric model.

The posterior is proportional to the prior times the likelihood.

⚠**Dimensionality hurdle:** We cannot naively take the limit $d \to \infty$ of Bayes' rule. For a standard Gaussian $\mathcal{N}(\mathbf{0}, I_d)$, the density value at the origin is:

$$p(\mathbf{0}) = \frac{1}{(2\pi)^{d/2}}$$

As $d \to \infty$, $p(\mathbf{0}) \to 0$. In infinite-dimensional spaces, standard probability density functions (with respect to a Lebesgue measure) cease to exist, so we cannot simply evaluate the limit of a finite-dimensional density.

How does this work in a Gaussian Process model with infinite parameters then?

Notation: Covariance Matrices for Sets of Points

Before assembling the joint distribution, we need a concise notation for evaluating the covariance function $k(x, x')$ over *sets* of input points.

Let $X_1 = \{x_1, \ldots, x_N\}$ be a set of N points, and $X_2 = \{x'_1, \ldots, x'_M\}$ be a set of M points.

We define the cross-covariance matrix $K(X_1, X_2)$ as the $N \times M$ matrix where the (i, j)-th entry is the covariance evaluated between the i-th point of X_1 and the j-th point of X_2:

$$K(X_1, X_2) = \begin{bmatrix} k(x_1, x'_1) & k(x_1, x'_2) & \ldots & k(x_1, x'_M) \\ k(x_2, x'_1) & k(x_2, x'_2) & \ldots & k(x_2, x'_M) \\ \vdots & \vdots & \ddots & \vdots \\ k(x_N, x'_1) & k(x_N, x'_2) & \ldots & k(x_N, x'_M) \end{bmatrix}$$

Key properties:

- $K(X_1, X_1)$ (often just written as K) is an $N \times N$ symmetric, positive semi-definite matrix.
- For different sets, $K(X_1, X_2)$ is generally a non-square $(N \times M)$ matrix.
- By the symmetry of k, we always have $K(X_2, X_1) = K(X_1, X_2)^\top$.

Gaussian Process Regression: The Finite Setup

We consider a standard regression model with additive Gaussian noise:

$$y_i = f(x_i) + \varepsilon_i, \quad \varepsilon_i \sim \mathcal{N}(0, \sigma^2_{\text{noise}})$$

We place a zero-mean GP prior on the unknown function: $f \sim \mathcal{GP}(0, k)$.

The trick: To avoid the infinite-dimensional hurdle, we use the marginalization property to restrict ourselves to a finite set of locations. Let:

- $\mathbf{X} = (x_1, \ldots, x_N)$ be the training inputs yielding noisy observations $\mathbf{y}$.
- $\mathbf{X}_*$ be the test inputs yielding unknown function values $\mathbf{f}_* = f(\mathbf{X}_*)$.

By the definition of a GP, the joint distribution of the observations $\mathbf{y}$ and test function values $\mathbf{f}_*$ is simply a finite multivariate Gaussian:

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

Posterior via Gaussian Conditioning

We want to predict the function values $\mathbf{f}_*$ given our training data $\mathbf{y}$.

Since $\mathbf{y}$ and $\mathbf{f}_*$ are jointly Gaussian, we can directly apply the standard Gaussian conditioning formulas we derived earlier!

Recall: If $\begin{bmatrix} \mathbf{x} \\ \mathbf{y} \end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix} \boldsymbol{a} \\ \boldsymbol{b} \end{bmatrix}, \begin{bmatrix} A & B \\ B^\top & C \end{bmatrix} \right)$, then $p(\mathbf{x}|\mathbf{y}) = \mathcal{N}(\boldsymbol{a} + BC^{-1}(\mathbf{y} - \boldsymbol{b}), A - BC^{-1}B^\top)$.

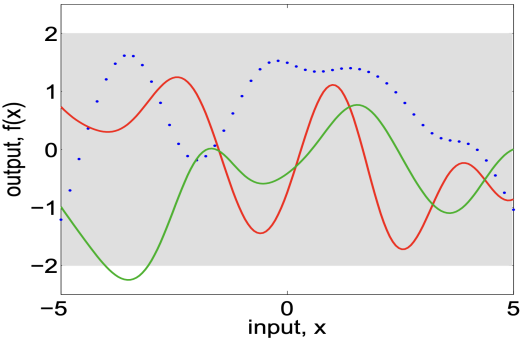
Applying this to our joint distribution gives the Gaussian process posterior (or predictive distribution):

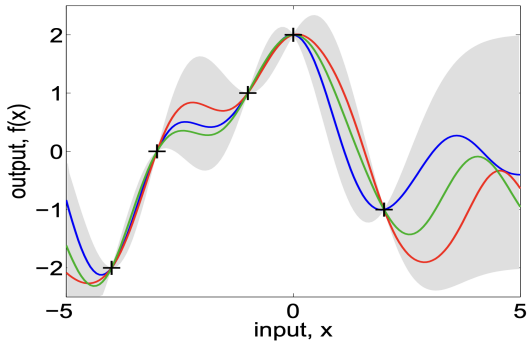
$$\mathbf{f}_*|\mathbf{X}, \mathbf{y}, \mathbf{X}_* \sim \mathcal{N}(\boldsymbol{\mu}_{*|\mathbf{y}}, \boldsymbol{\Sigma}_{*|\mathbf{y}})$$

where

$$\boldsymbol{\mu}_{*|\mathbf{y}} = \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2_{\text{noise}} I]^{-1}\mathbf{y}$$
$$\boldsymbol{\Sigma}_{*|\mathbf{y}} = \mathbf{K}(\mathbf{X}_*, \mathbf{X}_*) - \mathbf{K}(\mathbf{X}_*, \mathbf{X})[\mathbf{K}(\mathbf{X}, \mathbf{X}) + \sigma^2_{\text{noise}} I]^{-1}\mathbf{K}(\mathbf{X}, \mathbf{X}_*)$$

Prior and Posterior





Predictive distribution:

$$y_*|\mathbf{x}_*, \mathbf{x}, \mathbf{y} \sim \mathcal{N}(\mathbf{k}(\mathbf{x}_*, \mathbf{x})^\top [\mathbf{K} + \sigma^2_{\text{noise}} I]^{-1} \mathbf{y},$$
$$\mathbf{k}(\mathbf{x}_*, \mathbf{x}_*) + \sigma^2_{\text{noise}} - \mathbf{k}(\mathbf{x}_*, \mathbf{x})^\top [\mathbf{K} + \sigma^2_{\text{noise}} I]^{-1} \mathbf{k}(\mathbf{x}_*, \mathbf{x}))$$

The mean is linear in two ways:

$$\mu(\mathbf{x}_*) = \mathbf{K}(\mathbf{x}_*, \mathbf{x})[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2_{\text{noise}} I]^{-1}\mathbf{y} = \sum_{n=1}^{N} \beta_n y_n = \sum_{n=1}^{N} \alpha_n \mathbf{k}(\mathbf{x}_*, \mathbf{x}_n).$$

The last form is most commonly encountered in the kernel literature.

The variance is the difference between two terms:

$$V[\mathbf{x}_*] = \mathbf{K}(\mathbf{x}_*, \mathbf{x}_*) - \mathbf{K}(\mathbf{x}_*, \mathbf{x})[\mathbf{K}(\mathbf{x}, \mathbf{x}) + \sigma^2_{\text{noise}} I]^{-1}\mathbf{K}(\mathbf{x}, \mathbf{x}_*),$$

the first term is the prior variance, from which we subtract a (positive) term, telling how much the data $\mathbf{x}$ has explained.

Note, that the variance is independent of the observed outputs $\mathbf{y}$.

From random functions to covariance functions

Consider the class of linear functions:

$$f(x) = ax + b, \text{ where } a \sim \mathcal{N}(0, \alpha), \text{ and } b \sim \mathcal{N}(0, \beta).$$

We can compute the mean function:

$$\mu(x) = \mathbb{E}[f(x)] = \iint f(x)p(a)p(b)dadb = \int axp(a)da + \int bp(b)db = 0,$$

and covariance function:

$$\begin{aligned} k(x, x') &= \mathbb{E}[(f(x) - 0)(f(x') - 0)] = \iint (ax + b)(ax' + b')p(a)p(b)dadb \\ &= \int a^2 xx' p(a)da + \int b^2 p(b)db + (x + x') \int abp(a)p(b)dadb = \alpha xx' + \beta. \end{aligned}$$

Therefore: a linear model with Gaussian random parameters corresponds to a GP with covariance function $k(x, x') = \alpha xx' + \beta$.

From finite linear models to Gaussian processes (1)

Finite linear model with Gaussian priors on the weights:

$$f(x) = \sum_{j=1}^{J} w_j \phi_j(x) \qquad p(\mathbf{w}) = \mathcal{N}(\mathbf{w}; 0, A)$$

The *joint* distribution of any $\mathbf{f} = [f(x_1), \ldots, f(x_N)]^\top$ is a multivariate Gaussian – we have a Gaussian Process!

The prior $p(\mathbf{f})$ is fully characterized by the mean and covariance functions.

$$m(x) = \mathbb{E}_{\mathbf{w}}[f(x)] = \int \left(\sum_{j=1}^{J} w_j \phi_j(x) \right) p(\mathbf{w}) d\mathbf{w} = \sum_{j=1}^{J} \phi_j(x) \int w_j p(\mathbf{w}) d\mathbf{w}$$

$$= \sum_{j=1}^{J} \phi_j(x) \int w_j p(w_j) dw_j = 0$$

The mean function is zero.

From finite linear models to Gaussian processes (2)

Covariance function of a finite linear model

$$k(x_k, x_l) = \mathrm{Cov}_{\mathbf{w}}(f(x_k), f(x_l)) = \mathbb{E}_{\mathbf{w}}[f(x_k)f(x_l)] - \underbrace{\mathbb{E}_{\mathbf{w}}[f(x_k)]\mathbb{E}_{\mathbf{w}}[f(x_l)]}_{=0}$$

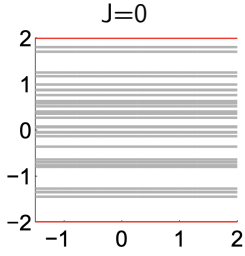
$$= \int \dots \int \left(\sum_{j=1}^{J} \sum_{i=1}^{J} w_i w_j \phi_k(x_k) \phi_l(x_l) \right) p(\mathbf{w}) d\mathbf{w}$$

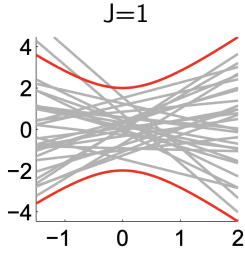
$$= \sum_{j=1}^{J} \sum_{i=1}^{J} \phi_i(x_k)\phi_j(x_l) \underbrace{\iint w_j w_i p(w_j, w_i) dw_j dw_i}_{A_{ji}} = \sum_{j=1}^{J} \sum_{i=1}^{J} A_{ji} \phi_j(x_k)\phi_i(x_l)$$

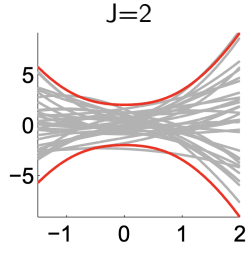
$$\boxed{k(x_k, x_l) = \boldsymbol{\phi}(x_k)^\top A \boldsymbol{\phi}(x_l)}$$

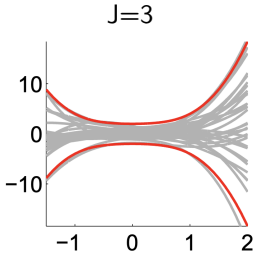
Note: If $A = \sigma_w^2 I$, then $k(x_k, x_l) = \sigma_w^2 \sum_{j=1}^{J} \phi_j(x_k)\phi_j(x_l) = \sigma_w^2 \boldsymbol{\phi}(x_k)^\top \boldsymbol{\phi}(x_l)$.

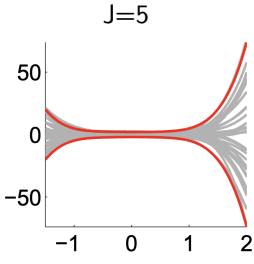
Are polynomials a good prior over functions?

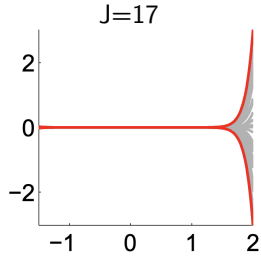












GPs and Linear in the parameters models are equivalent

We've seen that a "Linear in the parameters" model, with a Gaussian prior on the weights is also a GP.

Might it also be the case that every GP corresponds to a "Linear in the parameters" model?

The answer is yes, but not necessarily a finite one.

Mercer's Theorem and the Infinite Linear Model

Mercer's Theorem: Let $k(x, x')$ be a continuous, symmetric, positive semi-definite kernel function on a compact domain. Then k can be expressed as an absolutely and uniformly convergent series:

$$k(x, x') = \sum_{m=1}^{\infty} \lambda_m \phi_m(x) \phi_m(x')$$

where $\lambda_m \geqslant 0$ are the eigenvalues and $\phi_m(x)$ are orthonormal functions.

The equivalent GP: Using this eigen-decomposition, any zero-mean Gaussian Process $f \sim \mathcal{GP}(0, k)$ can be explicitly constructed as an infinite linear-in-the-parameters model:

$$f(x) = \sum_{j=1}^{\infty} w_j \sqrt{\lambda_j} \phi_j(x)$$

where the weights w_j are independent standard normal random variables, $w_j \sim \mathcal{N}(0, 1)$.

Conclusion: A GP is exactly equivalent to a Bayesian linear regression model utilizing an infinite number of basis functions $\psi_m(x) = \sqrt{\lambda_m} \phi_m(x)$ with a standard normal prior on the weights!

Example: squared-exponential kernel

Consider the class of functions (sums of squared exponentials):

$$f(x) = \lim_{N\to\infty} \frac{1}{N} \sum_{n=-N/2}^{N/2} \gamma_n \exp\left(-\left(x - \frac{n}{\sqrt{N}}\right)^2\right), \text{ where } \gamma_n \sim \mathcal{N}(0,1), \forall n$$

$$= \int_{-\infty}^{\infty} \gamma(u) \exp(-(x-u)^2) du, \text{ where } \gamma(u) \sim \mathcal{N}(0,1), \forall u.$$

The mean function is:

$$\mu(x) = \mathbb{E}[f(x)] = \int_{-\infty}^{\infty} \exp(-(x-u)^2) \int_{-\infty}^{\infty} \gamma(u) p(\gamma(u)) d\gamma(u)\ du = 0,$$

and the covariance function:

$$\mathbb{E}[f(x)f(x')] = \int \exp(-(x-u)^2 - (x'-u)^2) du$$

$$= \int \exp\left(-2\left(u - \frac{x+x'}{2}\right)^2 + \frac{(x+x')^2}{2} - x^2 - x'^2\right) du \propto \exp\left(-\frac{(x-x')^2}{2}\right).$$

Thus, the squared exponential covariance function is equivalent to regression using infinitely many Gaussian shaped basis functions placed everywhere, not just at your training points!

Hyperparameters: properties of covariance functions

The covariance function which we have seen before

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

encodes that $f(\mathbf{x})$ and $f(\mathbf{x}')$ have large covariance if $\mathbf{x}$ is close to $\mathbf{x}'$, but it doesn't really quantify what it means by close to?

We can *parameterize* the covariance function using hyperparameters such as l, in

$$k(\mathbf{x}, \mathbf{x}') = \exp\left(-\frac{(\mathbf{x} - \mathbf{x}')^2}{2l^2}\right).$$

Learning in Gaussian process models involves finding

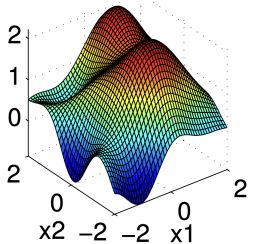
- the form of the covariance function, and
- any unknown (hyper-) parameters $\mathcal{H}$.

Model Selection, Hyperparameters, and ARD

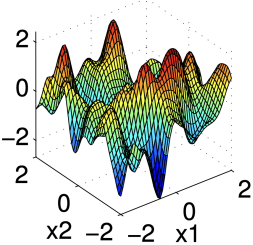
We need to determine both the *form* and *parameters* of the covariance function.
We typically use a hierarchical model, where the parameters of the covariance are called hyperparameters.
For instance, we can use the automatic relevance determination (ARD) covariance functions for feature/variable selection, e.g.:

$$k(\mathbf{x}, \mathbf{x}') = v_0^2 \exp\left(- \sum_{d=1}^{D} \frac{(x_d - x'_d)^2}{2 v_d^2} \right), \quad \text{hyperparameters } \theta = \{v_0, v_1, \ldots, v_D, \sigma_n^2\}.$$

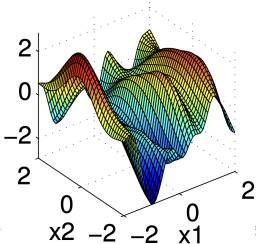
$v_1 = v_2 = 1$



$v_1 = v_2 = 0.32$



$v_1 = 0.32$ and $v_2 = 1$



Matérn covariance functions

Stationary covariance functions can also be based on the Matérn form:

$$k(\mathbf{x}, \mathbf{x}') = \frac{1}{\Gamma(\nu)2^{\nu-1}} \left[\frac{\sqrt{2\nu}}{\ell} \|\mathbf{x} - \mathbf{x}'\| \right]^{\nu} K_{\nu} \left(\frac{\sqrt{2\nu}}{\ell} \|\mathbf{x} - \mathbf{x}'\| \right),$$

where K_{ν} is the modified Bessel function of second kind of order ν, and ℓ is the characteristic length scale.

Sample functions from Matérn forms are $\lfloor \nu - 1/2 \rfloor$ times differentiable. Thus, the hyperparameter ν can control the degree of smoothness.

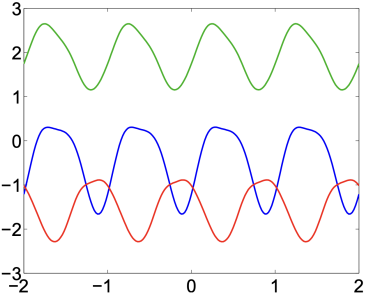
Special cases: Let $r = \|\mathbf{x} - \mathbf{x}'\|$

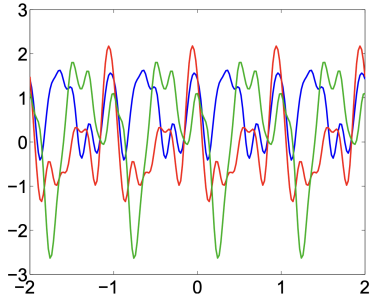
- $k_{\nu=1/2}(r) = \exp\left(-\frac{r}{\ell}\right)$: Laplacian covariance function, Ornstein-Uhlenbeck ($\sim$ Brownian motion)
- $k_{\nu=3/2}(r) = \left(1 + \frac{\sqrt{3}r}{\ell}\right) \exp\left(-\frac{\sqrt{3}r}{\ell}\right)$ (once differentiable)
- $k_{\nu=5/2}(r) = \left(1 + \frac{\sqrt{5}r}{\ell} + \frac{5r^2}{3\ell^2}\right) \exp\left(-\frac{\sqrt{5}r}{\ell}\right)$ (twice differentiable)
- $k_{\nu\to\infty} = \exp\left(-\frac{r^2}{2\ell^2}\right)$: smooth (infinitely differentiable)

Periodic, smooth functions

To create a distribution *over* periodic functions of x, we can first map the inputs to $\mathbf{u} = (\sin(x), \cos(x))^\top$, and then measure distances in the $\mathbf{u}$ space. Combined with the SE covariance function, which characteristic length scale ℓ, we get:

$$k_{\text{periodic}}(x, x') = \exp(-2\sin^2(\pi(x - x'))/\ell^2)$$





Three functions drawn at random; left $\ell > 1$, and right $\ell < 1$.

The Gaussian process marginal likelihood

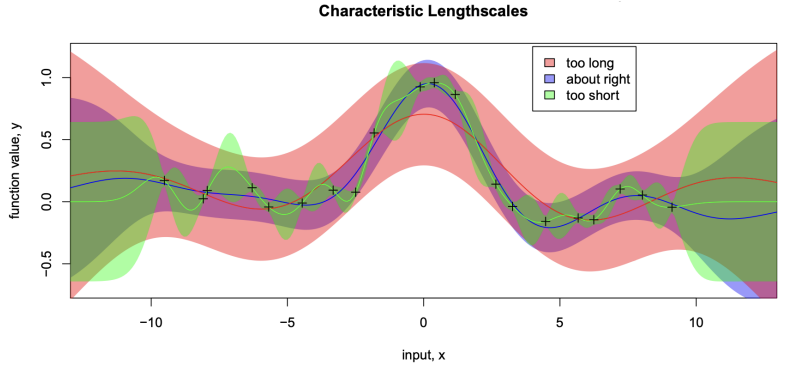
Writing $\mathcal{H}$ the set of hyperparameters, the log marginal likelihood has a closed form

$$\log p(\mathbf{y}|\mathbf{x}, \mathcal{H}) = -\frac{1}{2}\mathbf{y}^\top [\mathbf{K} + \sigma_n^2 \mathbf{I}]^{-1}\mathbf{y} - \frac{1}{2}\log|\mathbf{K} + \sigma_n^2 \mathbf{I}| - \frac{n}{2}\log(2\pi)$$

and is the combination of a data fit term and complexity penalty.

Example: Fitting the length scale parameter

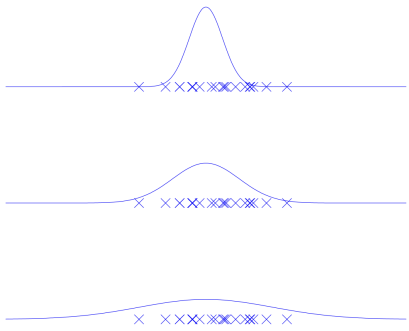
Parameterized covariance function: $k(x, x') = v^2 \exp\left(-\frac{(x-x')^2}{2l^2}\right)$.



The mean posterior predictive function is plotted for 3 different length scales (the blue curve corresponds to optimizing the marginal likelihood). Notice, that an almost exact fit to the data can be achieved by reducing the length scale – but the marginal likelihood does not favour this!

An illustrative analogous example

Imagine the simple task of fitting the variance, σ^2, of a zero-mean Gaussian to a set of n scalar observations.



The log likelihood is $\log p(\mathbf{y}|\mu, \sigma^2) = -\frac{1}{2}\mathbf{y}^\top\mathbf{y}/\sigma^2 - \frac{n}{2}\log(\sigma^2) - \frac{n}{2}\log(2\pi)$