PRML 6.4-6.4.5

MI Lab

Yuki Ichihara

June 26, 2023

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Gaussian Processes

In the Gaussian process framework, we bypass the need for explicit parameterization and directly define a prior distribution over functions. While this may initially appear challenging due to the infinite space of functions, we can effectively work in a finite space by considering only the function values at the input values of the training and test data points.

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Linear Combination

Let's reconsider linear regression using a fixed set of M basis functions, $\phi(x)$, to represent the model.

$$y(\boldsymbol{x}) = \boldsymbol{w}^{\mathrm{T}}\phi(\boldsymbol{x}) \tag{1}$$

w follows the next distribution.

$$p(\boldsymbol{w}) = \mathcal{N}\left(\boldsymbol{w} \mid 0, \alpha^{-1}\boldsymbol{I}\right) \tag{2}$$

where α is the hyperparameter representing the distribution's precision.

Linear Combination

From previous slides, w is produced by the probability distribution.
In practice, x might be training data points, which consist of specific values$(x_1, x_2, ..., x_n)$.
We reformulate (1) equation.

$$\boldsymbol{y} = \Phi \boldsymbol{w} \tag{3}$$

where Φ is the design matrix with elements $\Phi_{nk} = \phi_k(\boldsymbol{x}_n)$.

Linear Combination

We can find the probability distribution of $\boldsymbol{y}$ as follows in terms of the elements of $\boldsymbol{w}$.

$$\begin{aligned} \mathbb{E}[\boldsymbol{y}] &= \Phi\mathbb{E}[\boldsymbol{w}] = 0 \\ \mathrm{cov}[\boldsymbol{y}] &= \mathbb{E}\left[\boldsymbol{y}\boldsymbol{y}^{\mathrm{T}}\right] = \Phi\mathbb{E}\left[\boldsymbol{w}\boldsymbol{w}^{\mathrm{T}}\right]\Phi^{\mathrm{T}} = \frac{1}{\alpha}\Phi\Phi^{\mathrm{T}} = \boldsymbol{K} \end{aligned} \tag{4}$$

where $\boldsymbol{K}$ is the Gram matrix with elements

$$K_{nm} = k\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right) = \frac{1}{\alpha}\phi\left(\boldsymbol{x}_n\right)^{\mathrm{T}}\phi\left(\boldsymbol{x}_m\right) \tag{5}$$

Summary

- A key point about Gaussian stochastic processes is that the joint distribution over N variables $y_1, ..., y_n$ is specified completely by second-order statistics, namely the mean and the covariance.
- We usually choose the mean of the prior over weight values $p(\boldsymbol{w} \mid \alpha)$ to be zero in the basis function.

$$\mathbb{E}\left[y\left(\boldsymbol{x}_n\right) y\left(\boldsymbol{x}_m\right)\right] = k\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right) \tag{6}$$

Summary

We can also define the kernel function directly, rather than indirectly through a choice of basis function. The first of these is a 'Gaussian' kernel is given by

$$k\left(x, x'\right) = \exp\left(-\left\|x - x'\right\|^2 / 2\sigma^2\right) \tag{7}$$

And the second is the exponential kernel given by

$$k\left(x, x'\right) = \exp\left(-\theta\left|x - x'\right|\right) \tag{8}$$

Summary

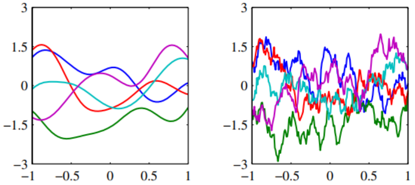

Figure: Samples from Gaussian processes for a ‘Gaussian’ kernel (left) and an exponential kernel (right).

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Target with noises

In order to apply Gaussian process models to the problem of regression, we need to take account of the noise on the observed target values, which are given by

$$t_n = y_n + \epsilon_n \tag{9}$$

where $y_n = y(\boldsymbol{x}_n)$, and ϵ_n is a random noise variable whose value is chosen independently for each observation n. So this distribution is represented by

$$p(t_n \mid y_n) = \mathcal{N}\left(t_n \mid y_n, \beta^{-1}\right) \tag{10}$$

where β is a hyperparameter of the noise.

Target with noises

The joint distribution of the target values $\boldsymbol{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$ conditioned on the values of $\boldsymbol{y} = (y_1, \ldots, y_N)^{\mathrm{T}}$ is given by an isotropic Gaussian of the form

$$p(\boldsymbol{t} \mid \boldsymbol{y}) = \mathcal{N}\left(\boldsymbol{t} \mid \boldsymbol{y}, \beta^{-1}\boldsymbol{I}_N\right) \tag{11}$$

In the previous chapter, $p(\boldsymbol{y})$ is given by

$$p(\boldsymbol{y}) = \mathcal{N}(\boldsymbol{y} \mid 0, \boldsymbol{K}) \tag{12}$$

Find the marginal distribution $p(\boldsymbol{t})$

We are able to derive the marginal distribution $p(\boldsymbol{t})$.

$$p(\boldsymbol{t}) = \int p(\boldsymbol{t} \mid \boldsymbol{y}) p(\boldsymbol{y}) \mathrm{d}\boldsymbol{y} = \mathcal{N}(\mathbf{t} \mid 0, \boldsymbol{C}) \tag{13}$$

where the covariance matrix $\boldsymbol{C}$ has elements

$$C\left(x_n, x_m\right) = k\left(x_n, x_m\right) + \beta^{-1} \delta_{nm} \tag{14}$$

Combination kernel

One widely used kernel function for Gaussian process regression is given by the exponential of a quadratic form, with the addition of constant and linear terms to give

$$k\left(x_n, x_m\right) = \theta_0 \exp\left\{-\frac{\theta_1}{2}\left\|x_n - x_m\right\|^2\right\} + \theta_2 + \theta_3 x_n^{\mathrm{T}} x_m \tag{15}$$

Note that the term involving θ_3 corresponds to a parametric model that is a linear function of the input variables.

Combination kernel

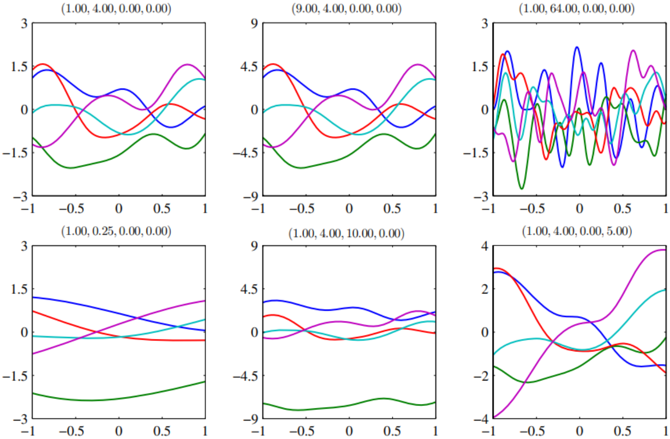

Figure: Samples from a Gaussian process prior defined by the covariance function

Our goal

Our goal is regression for new points, given a set of training data.

- suppose that $\boldsymbol{t}_N = (t_1, \ldots, t_N)^{\mathrm{T}}$, corresponding to input values $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$, comprise the observed training set
- our goal is to predict the target variable t_{N+1} for a new input vector $\boldsymbol{x}_{N+1}$. So we evaluate the predictive distribution $p\left(t_{N+1} \mid \boldsymbol{t}_N\right)$.

goal

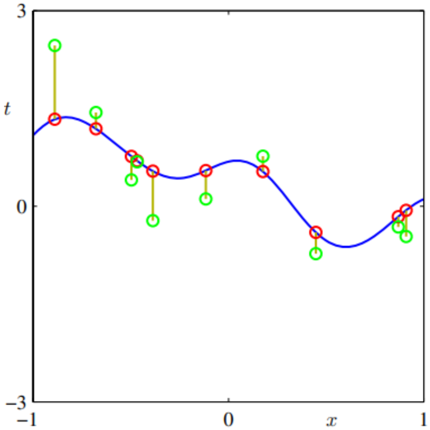

Figure: Illustration of the sampling of data points $\{t_n\}$ from a Gaussian process.

goal

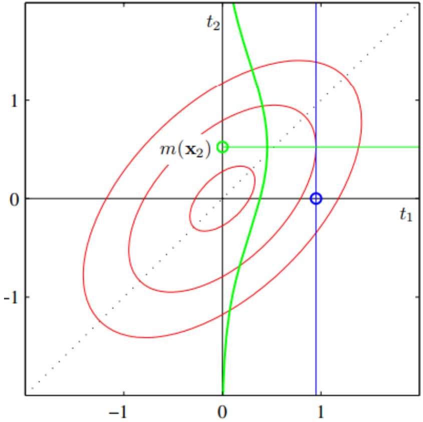

Figure: Illustration of the mechanism of Gaussian process regression for the case of one training point and one test point,

Our goal

From (13), the joint distribution over $t_1, .., t_n$ will be given by

$$p\left(\boldsymbol{t}_{N+1}\right) = \mathcal{N}\left(\boldsymbol{t}_{N+1} \mid 0, \boldsymbol{C}_{N+1}\right) \tag{16}$$

where $\boldsymbol{C}_{\boldsymbol{N}+1}$ is given by

$$\boldsymbol{C}_{N+1} = \begin{pmatrix} \boldsymbol{C}_N & \boldsymbol{k} \\ \boldsymbol{k}^{\mathrm{T}} & c \end{pmatrix} \tag{17}$$

where $\boldsymbol{C}_N$ is the $N \times N$ covariance matrix with elements given by (14) for $n, m = 1, \ldots, N$, the vector $\boldsymbol{k}$ has elements $k\left(\boldsymbol{x}_n, \boldsymbol{x}_{N+1}\right)$ for $n = 1, \ldots, N$, and the scalar $c = k\left(\boldsymbol{x}_{N+1}, \boldsymbol{x}_{N+1}\right) + \beta^{-1}$.

Our goal

We see that the conditional distribution $p\left(t_{N+1} \mid \boldsymbol{t}\right)$ is a Gaussian distribution with mean and covariance given by

$$\begin{aligned} m\left(\boldsymbol{x}_{N+1}\right) &= \boldsymbol{k}^{\mathrm{T}} \boldsymbol{C}_N^{-1} \boldsymbol{t} \\ \sigma^2\left(\boldsymbol{x}_{N+1}\right) &= c - \boldsymbol{k}^{\mathrm{T}} \boldsymbol{C}_N^{-1} \boldsymbol{k} \end{aligned} \tag{18}$$

These are the key results that define Gaussian process regression. Because the vector $\boldsymbol{k}$ is a function of the test point input value $\boldsymbol{x}_{N+1}$, we see that the predictive distribution is a Gaussian whose mean and variance both depend on $\boldsymbol{x}_{N+1}$.

Our goal

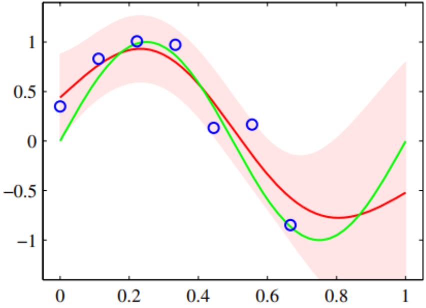

Figure: Illustration of Gaussian process regression applied to the sinusoidal data

Our goal

We redefine (18) with the techniques of constructing suitable kernels.

$$m\left(\boldsymbol{x}_{N+1}\right) = \sum_{n=1}^{N} a_n k\left(\boldsymbol{x}_n, \boldsymbol{x}_{N+1}\right) \tag{19}$$

where a_n is the n^{th} component of $\boldsymbol{C}_N^{-1}\boldsymbol{t}$. Thus, if the kernel function $k\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right)$ depends only on the distance $\|\boldsymbol{x}_n - \boldsymbol{x}_m\|$, then we obtain an expansion in radial basis functions.

Computation

- using Gaussian processes will involve the inversion of a matrix of size $N \times N$, for which standard methods require $O\left(N^3\right)$ computations
- in the basis function model, which has $O\left(M^3\right)$ computational complexity
- the number M of basis functions is smaller than the number N of data points, it will be computationally more efficient to work in the basis function

Applications

For large training data sets, however, the direct application of Gaussian process methods can become infeasible, and so a range of approximation schemes have been developed that have better scaling with training set size than the exact approach (Gibbs, 1997; Tresp, 2001; Smola and Bartlett, 2001; Williams and Seeger, 2001; Csató and Opper, 2002; Seeger et al., 2003). Practical issues in the application of Gaussian processes are discussed in Bishop and Nabney (2008).

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Covarince function

- In practice, rather than fixing the covariance function, we may prefer to use a parametric family of functions and then infer the parameter values from the data
- These parameters govern such things as the length scale of the correlations and the precision of the noise and correspond to the hyperparameters in a standard parametric model.

likelihood function

Techniques for learning the hyperparameters are based on the evaluation of the likelihood function $p(\boldsymbol{t} \mid \boldsymbol{\theta})$ where $\boldsymbol{\theta}$ denotes the hyperparameters of the Gaussian process model. cess model. The simplest approach is to make a point estimate of $\boldsymbol{\theta}$ by maximizing the log-likelihood function.

$$\ln p(\boldsymbol{t} \mid \boldsymbol{\theta}) = -\frac{1}{2} \ln |\boldsymbol{C}_N| - \frac{1}{2} \boldsymbol{t}^{\mathrm{T}} \boldsymbol{C}_N^{-1} \boldsymbol{t} - \frac{N}{2} \ln(2\pi) \tag{20}$$

log-likelihood

We have to calculate $\frac{\partial}{\partial\theta_i} \ln p(\boldsymbol{t} \mid \boldsymbol{\theta})$ with below techniques.

$$\frac{\partial}{\partial x}\left(\boldsymbol{C_N}^{-1}\right) = -\boldsymbol{C_N}^{-1}\frac{\partial \boldsymbol{C_N}}{\partial\theta_i}\boldsymbol{C_N}^{-1} \tag{21}$$

$$\frac{\partial}{\partial x}\ln|\boldsymbol{C_N}| = \operatorname{Tr}\left(\boldsymbol{C_N}^{-1}\frac{\partial \boldsymbol{C_N}}{\partial\theta_i}\right) \tag{22}$$

log-likelihood

$$\frac{\partial}{\partial \theta_i} \ln p(\boldsymbol{t} \mid \boldsymbol{\theta}) = -\frac{1}{2} \operatorname{Tr}\left(\boldsymbol{C}_N^{-1} \frac{\partial \boldsymbol{C}_N}{\partial \theta_i}\right) + \frac{1}{2} \boldsymbol{t}^{\mathrm{T}} \boldsymbol{C}_N^{-1} \frac{\partial \boldsymbol{C}_N}{\partial \theta_i} \boldsymbol{C}_N^{-1} \boldsymbol{t} \tag{23}$$

Because $\ln p(\boldsymbol{t} \mid \boldsymbol{\theta})$ will, in general, be a nonconvex function, it can have multiple maxima. The Gaussian process regression model gives a predictive distribution whose mean and variance are functions of the input vector $\boldsymbol{x}$. However, we have assumed that the contribution to the predictive variance arising from the additive noise, governed by the parameter β, is a constant. For some problems, known as heteroscedastic, the noise variance itself will also depend on x.

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

ARD

- the optimization of these parameters by maximum likelihood allows the relative importance of different inputs to be inferred from the data
- the Gaussian process context of automatic relevance determination, or ARD, which was originally formulated in the framework of neural networks (MacKay, 1994; Neal, 1996)

Consider a Gaussian process with a two-dimensional input space $\boldsymbol{x} = (x_1, x_2)$, having a kernel function of the form

$$k\left(\boldsymbol{x}, \boldsymbol{x}'\right) = \theta_0 \exp\left\{-\frac{1}{2}\sum_{i=1}^{2} \eta_i \left(x_i - x_i'\right)^2\right\} \tag{24}$$

ARD

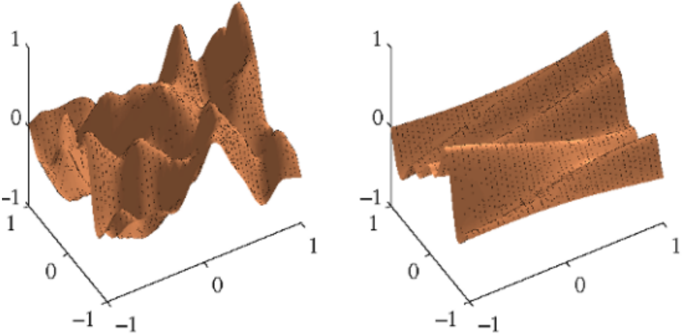

Figure: Samples from the ARD prior for Gaussian processes, in which the kernel function

ARD

- as a particular parameter η_i becomes small, the function becomes relatively insensitive to the corresponding input variable x_i.
- adapting these parameters to a data set using maximum likelihood.

ARD

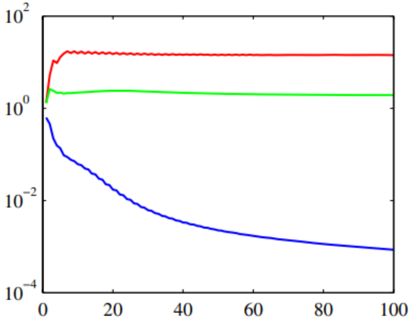

Figure: Illustration of automatic relevance determination in a Gaussian process for a synthetic problem having three inputs

ARD framework

The ARD framework is easily incorporated into the exponential-quadratic kernel (15) to give the following form of kernel function, which has been found useful for applications of Gaussian processes to a range of regression problems

$$k\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right) = \theta_0 \exp\left\{-\frac{1}{2}\sum_{i=1}^{D}\eta_i\left(x_{ni} - x_{mi}\right)^2\right\} + \theta_2 + \theta_3\sum_{i=1}^{D}x_{ni}x_{mi} \tag{25}$$

where D is the dimensionality of the input space.

Table of Contents

1. About Gaussian Processes
2. Linear regression revisited
3. Gaussian processes for regression
4. Learning the hyperparameters
5. Automatic relevance determination
6. Gaussian processes for classification

Introduction to classification

- Our goal is to model the posterior probabilities of the target variable for a new input vector, given a set of training data.]
- We can easily adapt Gaussian processes to classification problems by transforming the output of the Gaussian process using an appropriate nonlinear activation function.

Example

Consider first the two-class problem with a target variable $t \in \{0, 1\}$. If we define a Gaussian process over a function $a(\boldsymbol{x})$ and then transform the function using a logistic sigmoid $y = \sigma(a)$.

$$p(t \mid a) = \sigma(a)^t (1 - \sigma(a))^{1-t} \tag{26}$$

Example

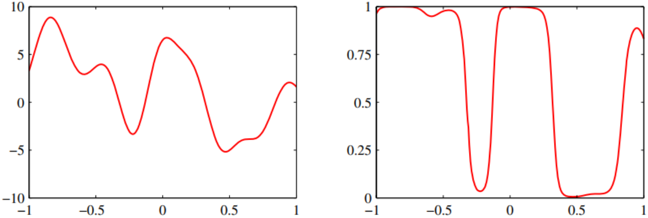

Figure: The left plot shows a sample from a Gaussian process prior over functions $a(\mathrm{x})$, and the right plot shows the result of transforming this sample using a logistic sigmoid function.

Example

We denote the training set inputs by $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ with corresponding observed target variables $\boldsymbol{t} = (t_1, \ldots, t_N)^{\mathrm{T}}$.

- Our goal is to determine the predictive distribution $p\left(t_{N+1} \mid \boldsymbol{t}\right)$, where we have left the conditioning on the input variables implicit.
- a Gaussian process prior over the vector $\boldsymbol{a}_{N+1}$, which has components $a\left(\boldsymbol{x}_1\right), \ldots, a\left(\boldsymbol{x}_{N+1}\right)$.

The Gaussian process prior for$\boldsymbol{a}_{N+1}$ takes the form

$$p\left(\boldsymbol{a}_{N+1}\right) = \mathcal{N}\left(\boldsymbol{a}_{N+1} \mid 0, \boldsymbol{C}_{N+1}\right). \tag{27}$$

Example

For numerical reasons it is convenient to introduce a noise-like term governed by a parameter ν that ensures that the covariance matrix is positive definite. Thus the covariance matrix $\boldsymbol{C}_{N+1}$ has elements given by

$$C\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right) = k\left(\boldsymbol{x}_n, \boldsymbol{x}_m\right) + \nu\delta_{nm} \tag{28}$$

2 Class Problem

For two-class problems, it is sufficient to predict $p\left(t_{N+1}=1 \mid \boldsymbol{t}_N\right)$ because the value of $p\left(t_{N+1}=0 \mid \boldsymbol{t}_N\right)$ is then given by $1-p\left(t_{N+1}=1 \mid \boldsymbol{t}_N\right)$. The required predictive distribution is given by

$$p\left(t_{N+1}=1 \mid \boldsymbol{t}_N\right)=\int p\left(t_{N+1}=1 \mid a_{N+1}\right) p\left(a_{N+1} \mid \boldsymbol{t}_N\right) \mathrm{d} a_{N+1} \qquad (29)$$

where $p\left(t_{N+1}=1 \mid a_{N+1}\right)=\sigma\left(a_{N+1}\right)$.