Continuous-Space Gaussian Process Regression and Generalized Wiener Filtering with Application to Learning Curves

Simo Särkkä and Arno Solin

Department of Biomedical Engineering and Computational Science,
Aalto University, Finland
{simo.sarkka,arno.solin}@aalto.fi

Abstract. Gaussian process regression is a machine learning paradigm, where the regressor functions are modeled as realizations from an a priori Gaussian process model. We study abstract continuous-space Gaussian regression problems where the training set covers the whole input space instead of consisting of a finite number of distinct points. The model can be used for analyzing theoretical properties of Gaussian process regressors. In this paper, we present the general continuous-space Gaussian process regression equations and discuss their close connection with Wiener filtering. We apply the results to estimation of learning curves as functions of training set size and input dimensionality.

Keywords: Gaussian process regression, continuous-space measurement, Wiener filter, learning curve

1 Introduction

Gaussian process (GP) regression [1] is a non-parametric Bayesian machine learning paradigm, where instead of estimating parameters of fixed-form functions $\mathbf{x} \mapsto \mathbf{f}(\mathbf{x}; \boldsymbol{\theta})$, we model the whole regressor functions $f(\mathbf{x})$ as Gaussian processes. That is, instead of postulating a prior for the function parameters $\boldsymbol{\theta}$, we postulate a prior for the functions $\mathbf{f}$. Learning in GPs amounts to conditioning the random function $\mathbf{f}$ on the measurement data. The predictive distributions of the conditioned process at test points then serve as the predictions of the model.

In this paper, we study an abstract continuous-space GP regression problem, where we measure the process continuously in the whole measurement space, not only in a finite of number training points. The model is useful in studying theoretical properties of Gaussian process regressors, and it is closely related to so-called equivalent kernels [2] which are tools for analyzing the theoretical properties of Gaussian process regressors, such as learning curves (see, e.g., [3–5]). We will show how learning curves can be analyzed in the presented framework as well. The model is also closely related to Wiener filtering (see, e.g., [6–8]). As we demonstrate here, the continuous-space GP regression is actually equivalent to Wiener filtering provided we generalize the Wiener filter to non-stationary processes.

2 Problem Formulation

In this paper, we consider continuous-measurement Gaussian process (GP) regression problems of the form:

$$\begin{aligned} f(\mathbf{x}) &\sim \mathcal{GP}(0, k_{\text{ff}}(\mathbf{x}, \mathbf{x}')) \\ y(\mathbf{x}) &= \mathcal{H}_{\mathbf{x}} f(\mathbf{x}) + e(\mathbf{x}), \end{aligned} \tag{1}$$

where the input is $\mathbf{x} \in \mathcal{X} \subseteq \mathbb{R}^d$ and the measurements cover the space $\mathcal{X} \subseteq \mathbb{R}^d$. The function $f(\mathbf{x})$ is a zero mean Gaussian process with a given covariance function $k_{\text{ff}}(\mathbf{x}, \mathbf{x}')$. The function is not observed directly, but instead, we measure a linear transformation of the signal, defined via the linear operator $\mathcal{H}_{\mathbf{x}}$ (cf. [9]), and the measurements $y(\mathbf{x})$ are also corrupted by measurement noise $e(\mathbf{x})$. Selection $\mathcal{H}_{\mathbf{x}} = 1$ leads to the ordinary Gaussian process regression model

$$y(\mathbf{x}) = f(\mathbf{x}) + e(\mathbf{x}). \tag{2}$$

For simplicity, we assume that the measurement functions y belong to the same space as the Gaussian processes f and thus both of them can be written as functions of $\mathbf{x} \in \mathcal{X}$. The measurement noise e is assumed to be a spatial Gaussian process with covariance function $k_{\text{ee}}(\mathbf{x}, \mathbf{x}')$. For notational convenience, we assume that both the function $f(\mathbf{x})$ and the measured signal $y(\mathbf{x})$ are scalar valued and zero mean.

The full solution to the finite-measurement version of the generalized Gaussian process regression problem in Equation (1) was presented in article [9] and the solution corresponding to the case $\mathcal{H}_{\mathbf{x}} = 1$ can be found in [1]. In image processing applications the including the operator $\mathcal{H}_{\mathbf{x}}$ into the model is very useful, because it can be used, for example, for modeling motion blurs or other linear degradations of images (cf. [10]).

We assume that the density of measurements in the input space is modeled by a density $w(\mathbf{x})$ such that the number of measurements dn in a small set of input space $\mathrm{d}\mathbf{x}$ is

$$dn = w(\mathbf{x})\, \mathrm{d}\mathbf{x}. \tag{3}$$

We could also easily replace the density with a more general measure.

3 Wiener Filtering

In this section we derive the solution to the continuous GP regression problem (1) by extending the methodology presented in [7] to multiple input dimensions. Classical continuous Wiener filtering (see, e.g., [6–8]) is considered with essentially the same problem that is specified in Equation (1). In the language of Wiener filtering, the model states that $f(\mathbf{x})$ is a zero-mean Gaussian process with covariance function

$$\mathrm{E}[f(\mathbf{x})\, f(\mathbf{x}')] = k_{\text{ff}}(\mathbf{x}, \mathbf{x}'). \tag{4}$$

It then follows from the model formulation that the process $y(\mathbf{x})$ is also a zero-mean Gaussian process and the covariance function of $y(\mathbf{x})$ as well as the cross-covariance of $f(\mathbf{x})$ and $y(\mathbf{x})$ are given as

$$\begin{aligned} k_{\mathrm{yy}}(\mathbf{x},\mathbf{x}') &= \mathcal{H}_{\mathbf{x}}\,\mathcal{H}_{\mathbf{x}'}\,k_{\mathrm{ff}}(\mathbf{x},\mathbf{x}') + k_{\mathrm{ee}}(\mathbf{x},\mathbf{x}') \\ k_{\mathrm{fy}}(\mathbf{x},\mathbf{x}') &= \mathcal{H}_{\mathbf{x}'}\,k_{\mathrm{ff}}(\mathbf{x},\mathbf{x}'). \end{aligned} \tag{5}$$

The derivation of the Wiener filter is based on variational minimization of the mean squared error functional

$$\mathcal{J}[m] = \mathrm{E}\left[(m(\mathbf{x}) - f(\mathbf{x}))^2 \mid \{y(\mathbf{x}) \;:\; \mathbf{x} \in \mathcal{X}\}\right], \tag{6}$$

where $\mathbf{x} \mapsto m(\mathbf{x})$ is the filtering result, the minimum mean squared error (MMSE) estimate of the signal. Because the functional is quadratic and all the processes are Gaussian, the solution is known to be a linear functional of the measurement signal. That is, there exist a kernel $g(\mathbf{x},\mathbf{x}')$ such that

$$m(\mathbf{x}) = \int_{\mathcal{X}} g(\mathbf{x},\mathbf{x}')\,y(\mathbf{x}')\,w(\mathbf{x})\,\mathrm{d}\mathbf{x}'. \tag{7}$$

The MSE functional can be now expanded and written in terms of covariance functions as follows:

$$\begin{aligned} \mathcal{J}[h] &= \mathrm{E}\left[(m(\mathbf{x}) - f(\mathbf{x}))^2 \mid \{y(\mathbf{x}) \;:\; \mathbf{x} \in \mathcal{X}\}\right] \\ &= \mathrm{E}\left[\left(\int_{\mathcal{X}} g(\mathbf{x},\mathbf{x}')\,y(\mathbf{x}')\,w(\mathbf{x})\,\mathrm{d}\mathbf{x}' - f(\mathbf{x})\right)^2 \mid \{y(\mathbf{x}) \;:\; \mathbf{x} \in \mathcal{X}\}\right] \\ &= \iint_{\mathcal{X}} g(\mathbf{x},\mathbf{x}')\,k_{\mathrm{yy}}(\mathbf{x}',\mathbf{x}'')\,g(\mathbf{x},\mathbf{x}'')\,w(\mathbf{x}')\,w(\mathbf{x}'')\,\mathrm{d}\mathbf{x}'\,\mathrm{d}\mathbf{x}'' \\ &\quad - 2\int_{\mathcal{X}} k_{\mathrm{fy}}(\mathbf{x},\mathbf{x}')\,g(\mathbf{x},\mathbf{x}')\,w(\mathbf{x}')\,\mathrm{d}\mathbf{x}' + k_{\mathrm{ff}}(\mathbf{x},\mathbf{x}), \end{aligned} \tag{8}$$

where k_{yy} and k_{fy} are given in Equation (5).

The minimizing kernel $g(\mathbf{x},\mathbf{x}')$ can be now solved using the standard ϵ-method from calculus of variations. That is, we replace g by $g + \epsilon\,\psi$, where $\psi(\mathbf{x},\mathbf{x}')$ is an arbitrary test function. Solving for $\partial\mathcal{J}/\partial\epsilon = 0$ and setting $\epsilon = 0$ then results in the equation

$$\begin{aligned} &\iint_{\mathcal{X}} g(\mathbf{x},\mathbf{x}'')\,k_{\mathrm{yy}}(\mathbf{x}'',\mathbf{x}')\,\psi(\mathbf{x},\mathbf{x}')\,w(\mathbf{x}')\,w(\mathbf{x}'')\,\mathrm{d}\mathbf{x}'\,\mathrm{d}\mathbf{x}'' \\ &- \int_{\mathcal{X}} k_{\mathrm{fy}}(\mathbf{x},\mathbf{x}')\,\psi(\mathbf{x},\mathbf{x}')\,w(\mathbf{x}')\,\mathrm{d}\mathbf{x}' = 0. \end{aligned} \tag{9}$$

Because this has to be true for arbitrary ψ, we must have

$$k_{\mathrm{fy}}(\mathbf{x},\mathbf{x}') = \int_{\mathcal{X}} g(\mathbf{x},\mathbf{x}'')\,k_{\mathrm{yy}}(\mathbf{x}'',\mathbf{x}')\,w(\mathbf{x}'')\,\mathrm{d}\mathbf{x}'' \tag{10}$$

which is the (generalized) *Wiener–Hopf integral equation* for the function g.

Thus the solution to the Wiener filtering problem can be obtained by solving the function g from the Wiener–Hopf equation above and then computing the estimate $m(\mathbf{x})$ via Equation (7). The classical ways to solve the Wiener–Hopf equation are by using basis function expansions (namely Karhunen–Loeve series) or via the Fourier transform. The latter solution method leads to the classical Wiener filter in the form that it is usually found in image processing literature (e.g. [10]). We will return to this solution method later in this paper, but let's first discuss the connection with Gaussian process regression.

The covariance function of estimation error can then be computed as to be

$$\begin{aligned} V(\mathbf{x},\mathbf{x}') &= \mathrm{E}\Bigg[\left(\int_{\mathcal{X}} g(\mathbf{x},\mathbf{x}'')\, y(\mathbf{x}'')\, w(\mathbf{x}'')\, \mathrm{d}\mathbf{x}'' - f(\mathbf{x}) \right) \\ &\qquad \times \left(\int_{\mathcal{X}} g(\mathbf{x}',\mathbf{x}''')\, y(\mathbf{x}''')\, w(\mathbf{x}'')\, \mathrm{d}\mathbf{x}''' - f(\mathbf{x}') \right) \mid \{y(\mathbf{x}) \ : \ \mathbf{x} \in \mathcal{X}\} \Bigg] \\ &= k_{\mathrm{ff}}(\mathbf{x},\mathbf{x}') - \int_{\mathcal{X}} g(\mathbf{x},\mathbf{x}'') k_{\mathrm{fy}}(\mathbf{x}'',\mathbf{x}')\, w(\mathbf{x}'')\, \mathrm{d}\mathbf{x}''. \end{aligned} \tag{11}$$

4 Continuous-Measurement Gaussian Process Regression

In this section we derive the continuous-space Gaussian process regression equations as continuous limits of the ordinary Gaussian process regression equations. The derivation is informal and merely demonstrates where the results come from, but the same idea would indeed work in a more rigorous derivation.

Consider the following discrete approximation to the Gaussian process regression problem in Equation (1):

$$\begin{aligned} \mathbf{f} &\sim \mathrm{N}(\mathbf{0}, \mathbf{K}_{\mathrm{ff}}) \\ \mathbf{y} &= \mathbf{H}\,\mathbf{f} + \mathbf{e}, \end{aligned} \tag{12}$$

where $\mathbf{f} = (f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N))$, $\mathbf{y} = (y(\mathbf{x}_1), \ldots, y(\mathbf{x}_N))$, $\mathbf{e} = (e(\mathbf{x}_1), \ldots, e(\mathbf{x}_N))$, and $\mathbf{H}$ is a suitable discrete approximation to the operator $\mathcal{H}_{\mathbf{x}}$. The posterior for the mean and covariance of $\mathbf{f}$ are now given as

$$\begin{aligned} \mathbf{m} &= \mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} \left[\mathbf{H}\,\mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} + \mathbf{K}_{\mathrm{ee}}\right]^{-1} \mathbf{y} \\ \mathbf{V} &= \mathbf{K}_{\mathrm{ff}} - \mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} \left[\mathbf{H}\,\mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} + \mathbf{K}_{\mathrm{ee}}\right]^{-1} \mathbf{H}\,\mathbf{K}_{\mathrm{ff}}, \end{aligned} \tag{13}$$

where the matrices $\mathbf{K}_{\mathrm{ff}}$ and $\mathbf{K}_{\mathrm{ee}}$ have been formed by evaluating $k_{\mathrm{ff}}(\mathbf{x},\mathbf{x}')$ and $k_{\mathrm{ee}}(\mathbf{x},\mathbf{x}')$ at each pair $\{(\mathbf{x},\mathbf{x}') \ : \ \mathbf{x},\mathbf{x}' \in \{\mathbf{x}_1,\ldots,\mathbf{x}_N\}\}$. We now define matrix $\mathbf{G}$ as follows:

$$\mathbf{G} = \mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} \left[\mathbf{H}\,\mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} + \mathbf{K}_{\mathrm{ee}}\right]^{-1}, \tag{14}$$

which implies that it is the solution to the equation

$$\mathbf{G}\,\mathbf{H}\,\mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}} + \mathbf{G}\,\mathbf{K}_{\mathrm{ee}} = \mathbf{K}_{\mathrm{ff}}\,\mathbf{H}^{\mathsf{T}}. \tag{15}$$

It is now easy to see that when $N \to \infty$ such that the set of points $\{\mathbf{x}_i : i = 1, \ldots, N\}$ dense, this converges to

$$\mathcal{G}_{\mathbf{x}} \mathcal{H}_{\mathbf{x}} \mathcal{H}_{\mathbf{x}'} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') + \mathcal{G}_{\mathbf{x}} k_{\mathrm{ee}}(\mathbf{x}, \mathbf{x}') = \mathcal{H}_{\mathbf{x}'} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}'), \tag{16}$$

where $\mathcal{G}_{\mathbf{x}}$ is a linear operator. The mean equation thus becomes

$$m(\mathbf{x}) = \mathcal{G}_{\mathbf{x}} y(\mathbf{x}). \tag{17}$$

We can now rewrite the covariance as $\mathbf{V} = \mathbf{K}_{\mathrm{ff}} - \mathbf{G}\,\mathbf{H}\,\mathbf{K}_{\mathrm{ff}}$, which thus gives the following expression for covariance function in the limit:

$$V(\mathbf{x}, \mathbf{x}') = k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') - \mathcal{G}_{\mathbf{x}} \mathcal{H}_{\mathbf{x}} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}'). \tag{18}$$

There now exists a kernel $g(\mathbf{x}, \mathbf{x}')$ such that

$$\mathcal{G}_{\mathbf{x}} f(\mathbf{x}) = \int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}')\, f(\mathbf{x}')\, w(\mathbf{x}')\, \mathrm{d}\mathbf{x}', \tag{19}$$

for all functions f and thus if we define

$$\begin{aligned} k_{\mathrm{yf}}(\mathbf{x}, \mathbf{x}') &= \mathcal{H}_{\mathbf{x}} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') \\ k_{\mathrm{fy}}(\mathbf{x}, \mathbf{x}') &= \mathcal{H}_{\mathbf{x}'} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') \\ k_{\mathrm{yy}}(\mathbf{x}, \mathbf{x}') &= \mathcal{H}_{\mathbf{x}} \mathcal{H}_{\mathbf{x}'} k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') + k_{\mathrm{ee}}(\mathbf{x}, \mathbf{x}'), \end{aligned} \tag{20}$$

then the mean and covariance equations can be expressed as

$$\begin{aligned} m(\mathbf{x}) &= \int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}')\, y(\mathbf{x}')\, w(\mathbf{x}')\, \mathrm{d}\mathbf{x}' \\ V(\mathbf{x}, \mathbf{x}') &= k_{\mathrm{ff}}(\mathbf{x}, \mathbf{x}') - \int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}'')\, k_{\mathrm{yf}}(\mathbf{x}'', \mathbf{x}')\, w(\mathbf{x}'')\, \mathrm{d}\mathbf{x}'' \end{aligned} \tag{21}$$

and Equation (16) reduces to the equation

$$\int_{\mathcal{X}} g(\mathbf{x}, \mathbf{x}'')\, k_{\mathrm{yy}}(\mathbf{x}'', \mathbf{x}')\, w(\mathbf{x}'')\, \mathrm{d}\mathbf{x}'' = k_{\mathrm{fy}}(\mathbf{x}, \mathbf{x}'). \tag{22}$$

Comparing to the equations in previous section, we can see that the continuous-space limit of GP regression equations is exactly the Wiener filter. The minimum mean squared estimate is now the posterior mean, and the error covariance function is the posterior covariance function.

5 Fourier Transform Solution

The classical method of solving the Wiener filtering problem is by using the Fourier transform. It can be used if we assume that the Gaussian processes $f(\mathbf{x})$ and $e(\mathbf{x})$ are stationary, the domain is the whole space $\mathcal{X} = \mathbb{R}^d$, and $w(\mathbf{x}) = 1$.

In this case the covariance function becomes a function of a single difference variable $\mathbf{x} - \mathbf{x}'$ and thus the model becomes

$$\begin{aligned} f(\mathbf{x}) &\sim \mathcal{GP}(0, k_{\mathrm{ff}}(\mathbf{x} - \mathbf{x}')) \\ y(\mathbf{x}) &= f(\mathbf{x}) + e(\mathbf{x}). \end{aligned} \tag{23}$$

Due to the stationarity the kernel g also becomes stationary and with suitable change of integration variables Equations (21) can be expressed as convolutions:

$$\begin{aligned} m(\mathbf{x}) &= \int g(\mathbf{x} - \mathbf{x}')\, y(\mathbf{x}')\, \mathrm{d}\mathbf{x}' \\ V(\mathbf{x}) &= k_{\mathrm{ff}}(\mathbf{x}) - \int g(\mathbf{x} - \mathbf{x}')\, k_{\mathrm{yf}}(\mathbf{x}')\, \mathrm{d}\mathbf{x}'. \end{aligned} \tag{24}$$

Similarly the Wiener–Hopf Equation (22) reduces to

$$k_{\mathrm{fy}}(\mathbf{x}) = \int g(\mathbf{x} - \mathbf{x}')\, k_{\mathrm{yy}}(\mathbf{x}')\, \mathrm{d}\mathbf{x}'. \tag{25}$$

Taking Fourier transforms[1] of the Equations (24) and (25) results in

$$\begin{aligned} M(\boldsymbol{\omega}) &= G(\boldsymbol{\omega})\, Y(\boldsymbol{\omega}) \\ V(\boldsymbol{\omega}) &= S_{\mathrm{ff}}(\boldsymbol{\omega}) - G(\boldsymbol{\omega})\, S_{\mathrm{yf}}(\boldsymbol{\omega}) \\ S_{\mathrm{fy}}(\boldsymbol{\omega}) &= G(\boldsymbol{\omega})\, S_{\mathrm{yy}}(\boldsymbol{\omega}), \end{aligned} \tag{26}$$

where M, G and Y are the Fourier transforms of the functions m, g and y, respectively, H is the Fourier transform of operator $\mathcal{H}$, and the spectral densities S_{yy}, S_{fy} and S_{yf} are given as:

$$\begin{aligned} S_{\mathrm{yy}}(\boldsymbol{\omega}) &= H(\boldsymbol{\omega})\, H^*(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega}) + S_{\mathrm{ee}}(\boldsymbol{\omega}) \\ S_{\mathrm{fy}}(\boldsymbol{\omega}) &= H^*(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega}) \\ S_{\mathrm{yf}}(\boldsymbol{\omega}) &= H(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega}), \end{aligned} \tag{27}$$

and by Wiener–Khinchin theorem, the spectral density $S_{\mathrm{ff}}(\boldsymbol{\omega})$ is simply the Fourier transform of the covariance function $k_{\mathrm{ff}}(\mathbf{x})$. In the above equations $(\cdot)^*$ denotes the complex conjugate. The expressions for the Fourier transforms of mean and covariance can be then written as

$$\begin{aligned} M(\boldsymbol{\omega}) &= \left[\frac{H^*(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega})}{|H(\boldsymbol{\omega})|^2\, S_{\mathrm{ff}}(\boldsymbol{\omega}) + S_{\mathrm{ee}}(\boldsymbol{\omega})} \right] Y(\boldsymbol{\omega}) \\ V(\boldsymbol{\omega}) &= \frac{S_{\mathrm{ee}}(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega})}{|H(\boldsymbol{\omega})|^2\, S_{\mathrm{ff}}(\boldsymbol{\omega}) + S_{\mathrm{ee}}(\boldsymbol{\omega})}. \end{aligned} \tag{28}$$

The former of these equation is the classical Fourier domain Wiener filter in the form that it is usually found in image processing literature (see, e.g., [10]). But it is also the Fourier transform of the mean of the Gaussian process regression solution. The latter equation is the spectral density of the posterior error which naturally arises as the spectral density of the posterior covariance in the Gaussian process regression interpretation.

[1] We define Fourier transform and its inverse via $F(\boldsymbol{\omega}) = \int_{\mathbb{R}^d} f(\mathbf{x})\, \exp(-i\, \boldsymbol{\omega}^{\mathsf{T}}\, \mathbf{x})\, \mathrm{d}\mathbf{x}$ and $f(\mathbf{x}) = \frac{1}{(2\pi)^d} \int_{\mathbb{R}^d} F(\boldsymbol{\omega})\, \exp(i\, \boldsymbol{\omega}^{\mathsf{T}}\, \mathbf{x})\, \mathrm{d}\boldsymbol{\omega}$, respectively.

6 Application to Estimation of Average Learning Curves

A simple approach to estimation of average learning curves (see, e.g., [3–5]) is to estimate the learning curve as

$$\epsilon(n) \simeq \int_{\mathcal{X}} V(\mathbf{x}, \mathbf{x}; n)\, p(\mathbf{x})\, \mathrm{d}\mathbf{x}, \tag{29}$$

where $V(\mathbf{x}, \mathbf{x}'; n)$ is the posterior error covariance function with n measurements. If we assume that all the processes are stationary and $w(\mathbf{x}) = 1$, then we can use the Fourier transform based solution presented in Section 5. The corresponding spectral density recursion for the spectral domain solution is

$$V(\boldsymbol{\omega}; n+1) = \frac{S_{\mathrm{ee}}(\boldsymbol{\omega})\, V(\boldsymbol{\omega}; n)}{|H(\boldsymbol{\omega})|^2\, V(\boldsymbol{\omega}; n) + S_{\mathrm{ee}}(\boldsymbol{\omega})}. \tag{30}$$

with $V(\boldsymbol{\omega}; 0) = S_{\mathrm{ff}}(\boldsymbol{\omega})$, which has the solution

$$V(\boldsymbol{\omega}; n) = \frac{S_{\mathrm{ee}}(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega})}{n\, |H(\boldsymbol{\omega})|^2\, S_{\mathrm{ff}}(\boldsymbol{\omega}) + S_{\mathrm{ee}}(\boldsymbol{\omega})} \tag{31}$$

Recalling that $V(\mathbf{0}; n) = (2\pi)^{-d} \int V(\boldsymbol{\omega}; n)\, \mathrm{d}\boldsymbol{\omega}$, the approximation to the learning curve now reduces to

$$\epsilon(n) \simeq (2\pi)^{-d} \int \frac{S_{\mathrm{ee}}(\boldsymbol{\omega})\, S_{\mathrm{ff}}(\boldsymbol{\omega})}{n\, |H(\boldsymbol{\omega})|^2\, S_{\mathrm{ff}}(\boldsymbol{\omega}) + S_{\mathrm{ee}}(\boldsymbol{\omega})}\, \mathrm{d}\boldsymbol{\omega}, \tag{32}$$

which can be seen to be a generalization to the equivalent kernel based learning curve of Sollich and Williams [2].

The above approximation is particularly useful when the covariance functions and thus spectral densities are isotropic, that is, they have the form $S_{\mathrm{ff}}(\boldsymbol{\omega}) = S_{\mathrm{ff}}(\|\boldsymbol{\omega}\|)$, $S_{\mathrm{ee}}(\boldsymbol{\omega}) = S_{\mathrm{ee}}(\|\boldsymbol{\omega}\|)$, and if the operator is isotropic as well $H(\boldsymbol{\omega}) = H(\|\boldsymbol{\omega}\|)$. If we denote $r = \|\boldsymbol{\omega}\|$, then by converting the above integral into spherical coordinates it can be expressed as

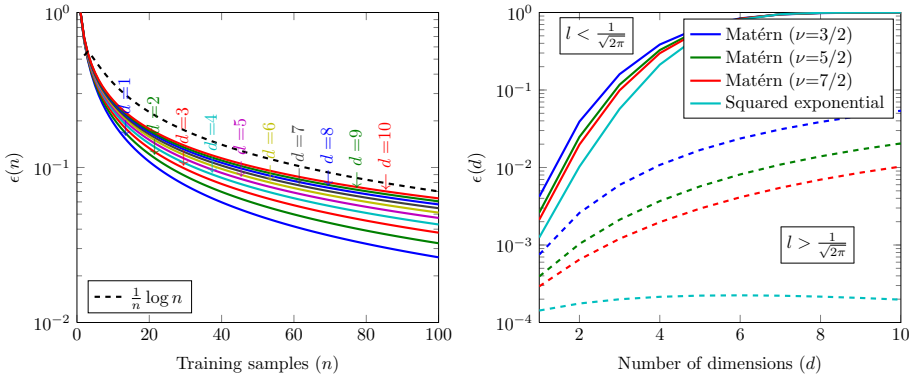
$$\epsilon(n) \simeq (2\pi)^{-d} A_d \int_0^{\infty} \frac{S_{\mathrm{ee}}(\boldsymbol{\omega})\, S_{\mathrm{ff}}(r)}{n\, |H(r)|^2\, S_{\mathrm{ff}}(r) + S_{\mathrm{ee}}(\boldsymbol{\omega})}\, r^{d-1}\, \mathrm{d}r, \tag{33}$$

where A_d is the surface area of the unit hypersphere of dimension d.

In the following examples we study this approximation. We consider the measurement model to be the identity operator, and the observations to be corrupted by white Gaussian noise with spectral density σ^2 (i.e., $S_{\mathrm{ee}}(\boldsymbol{\omega}) = \sigma^2$).

Example 1 (Learning curves for squared exponential covariance functions). The squared exponential covariance function has the form $\exp(-\alpha r^2)$, where $\alpha = 1/(2l^2)$. The corresponding spectral density is

$$S_{\mathrm{ff}}(\boldsymbol{\omega}) = \left(\frac{\pi}{\alpha}\right)^{d/2} \exp\left(-\frac{\|\boldsymbol{\omega}\|^2}{4\alpha}\right). \tag{34}$$



(a) Squared exponential (b) Matérn family

Fig. 1. (a) Learning curves for squared exponential covariance functions for different number of input dimensions. (b) Average errors for the Matérn family of covariance functions as function of the input dimensionality (solid $l = 0.1$, dashed $l = 1$).

Because the spectral densities are only functions of the norm $r = \|\boldsymbol{\omega}\|$ we can use Equation (33), which gives

$$\epsilon(n) \simeq -\sigma^2 \frac{1}{n} \left(\frac{\alpha}{\pi}\right)^{d/2} \mathrm{Li}_{d/2}\left(-\left(\frac{\pi}{\alpha}\right)^{d/2} \frac{n}{\sigma^2}\right), \tag{35}$$

where d is the dimensionality of the inputs, n the size of the training set, and $\mathrm{Li}_s(\cdot)$ the polylogarithm function. Figure 1a shows the behavior of the function for training samples $n = 1, 2, \ldots, 100$ and $d = 1, 2, \ldots, 10$. The scale parameters were fixed at $\sigma^2, s^2, l = 1$, and all the trajectories normalized to one when $n = 1$. Figure 1a shows how the the error reduces as the number of training inputs grows, and how adding dimensions reduces the effect of data. The solution of $-\frac{1}{n}\mathrm{Li}_{d/2}(-n)$ for $d = 2$ is $\frac{1}{n}\log(n+1)$, and as $n \to \infty$ it coincides with $\frac{1}{n}\log(n)$, which was suggested in [11] for a Gaussian input density.

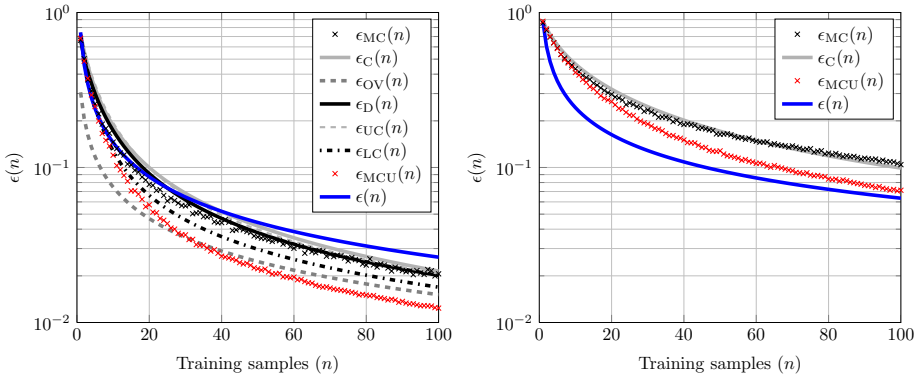
Example 2 (Learning curves for Matérn covariance functions). The Matérn covariance function is ($r = \|\mathbf{x} - \mathbf{x}'\|$)

$$k_{\mathrm{ff}}(r) = s^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\sqrt{2\nu}\,\frac{r}{l}\right)^{\nu} K_\nu \left(\sqrt{2\nu}\,\frac{r}{l}\right), \tag{36}$$

where $\nu, s, l > 0$ are the smoothness, magnitude and length scale parameters, and $K_\nu(\cdot)$ the modified Bessel function [1]. The spectral density is

$$S(\boldsymbol{\omega}) \propto \frac{1}{(\lambda^2 + \|\boldsymbol{\omega}_x\|^2)^{\nu+d/2}}, \tag{37}$$

where $\lambda = \sqrt{2\nu}/l$.



(a) 1d Matérn (b) 2d Matérn

Fig. 2. Learning curves for the Matérn covariance function in one and two dimensions.

Figure 1b shows the expected errors as function of the input dimensionality d evaluated numerically from (33) with parameters $s^2 = 1$, $\sigma^2 = 0.1$, fixed $n = 1000$, and $l = 0.1$ (solid) or $l = 1$ (dashed). The trajectories include the Matérn model for three different smoothness parameter values $\nu = 3/2, 5/2, 7/2$, and for comparison the squared exponential covariance function, for which $\nu \to \infty$. The smoothness assumption included in the model influences the learning curve estimates, and the smoother the model, the less the error. For $l < 1/\sqrt{2\pi}$ we get $\lim_{d\to\infty} \epsilon(d) = 1$, and for $l > 1/\sqrt{2\pi}$, we get $\lim_{d\to\infty} \epsilon(d) = 0$, when $\nu = \infty$.

We compare our learning curve approximation to several other estimates for $\nu = 3/2$. In Figure 2 $\epsilon_{\mathrm{MC}}(n)$ is the 'true' generalization error curve computed by 100 independent Monte Carlo samples for each n with unit Gaussian input density; $\epsilon_{\mathrm{C}}(n)$ is the Gauss–Hermite learning curve approximation due to [9] using the 60th and 20th order Gauss–Hermite rule in the 1d and 2d examples, respectively; $\epsilon_{\mathrm{OV}}(n)$ is the Opper–Vivarelli bound [3]; $\epsilon_{\mathrm{D}}(n)$, $\epsilon_{\mathrm{UC}}(n)$, $\epsilon_{\mathrm{LC}}(n)$ are the bounds considered by Sollich and Halees [4]; $\epsilon_{\mathrm{MCU}}(n)$ is the 'true' generalization error the unit variance uniform input density; and, finally, $\epsilon(n)$ is the proposed learning curve approximation normalized to the same scale with the rest of the curves. The figures show that the approximation $\epsilon(n)$ underestimates the error for small n and overestimates it for large n when compared to the 'true' values $\epsilon_{\mathrm{MCU}}(n)$. For higher d the intersection point will be reached at even larger n.

7 Conclusion and Discussion

In this paper, we have studied an abstract continuous-space Gaussian regression problem which is a useful theoretical tool for analyzing properties of Gaussian process regressors. We have also shown the connection of the formulation to generalized Wiener filtering and applied it to estimation of learning curves for Gaussian process regressors.

Even though we have here only considered scalar a priori Gaussian processes and measurements, the results could be easily extended to multiple dimensions (cf. [9]). We could also relax the assumption about the measurements belonging to the same function space as the a priori Gaussian process, which would allow analysis of more general inverse problems. However, the current formulation is sufficient for modeling, for example, image deformations, blurs, and other degradations, because in these models the operator maps images into images and thus the spaces are the same.

If we are interested in performing Gaussian process regression on a finite grid (such as in image processing), the Fourier domain solution allows for efficient computations via the use of the Fast Fourier Transform (FFT) algorithm. It turns out that an analogous Fourier domain solution is valid in the discrete case and thus we can use it to reduce the $O(N^3)$ complexity in the number of measurements N into $O(N \log N)$ complexity of the FFT based solution. This method is also commonly used in Wiener filters arising in image processing.

Another useful solution method is to use an eigenbasis of the prior covariance function. If we select a weight function such that $w(\mathbf{x}) = 1$ on a finite domain $\mathcal{X} \subset \mathbb{R}^d$ and zero elsewhere, this leads to a so-called Karhunen–Loeve expansion of $f(\mathbf{x})$ (see, e.g., [7, 8]). The posterior mean and covariance can then be expressed as a linear combination of the basis functions. However, it is also possible to form the expansion with respect to other weight functions $w(\mathbf{x})$ to obtain a similar basis function expansion (cf. [1]).

References

1. Rasmussen, C.E., Williams, C.K.I.: Gaussian Processes for Machine Learning. The MIT Press (2006)
2. Sollich, P., Williams, C.: Using the equivalent kernel to understand Gaussian process regression. In: NIPS 17. MIT Press (2005) 1313–1320
3. Opper, M., Vivarelli, F.: General bounds on Bayes errors for regression with gaussian processes. In: NIPS 11. The MIT Press (1999) 302–308
4. Sollich, P., Halees, A.: Learning curves for Gaussian process regression: Approximations and bounds. Neural Computation **14**(6) (2002) 1393–1428
5. Särkkä, S.: Learning curves for Gaussian processes via numerical cubature integration. In: Proceedings of ICANN. (2011)
6. Wiener, N.: Extrapolation, Interpolation and Smoothing of Stationary Time Series with Engineering Applications. John Wiley & Sons, Inc., New York (1950)
7. Van Trees, H.L.: Detection, Estimation, and Modulation Theory Part I. John Wiley & Sons, New York (1968)
8. Papoulis, A.: Probability, Random Variables, and Stochastic Processes. McGraw-Hill (1984)
9. Särkkä, S.: Linear operators and stochastic partial differential equations in Gaussian process regression. In: Proceedings of ICANN. (2011)
10. Gonzalez, R.C., Woods, R.E.: Digital image processing. Prentice Hall (2002)
11. Opper, M.: Regression with Gaussian processes: Average case performance. In: Theoretical Aspects of Neural Computation. Springer-Verlag (1997)