Prediction at an Uncertain Input for Gaussian Processes and Relevance Vector Machines Application to Multiple-Step Ahead Time-Series Forecasting

Joaquin Quiñonero-Candela
Informatics and Mathematical Modelling
Technical University of Denmark
Richard Petersens Plads, Build. 321
2800 Kongens Lyngby, Denmark
jqc@imm.dtu.dk

Agathe Girard
Dept. of Computer Science
Glasgow University
17 Lilybank Gardens
Glasgow G12 8QQ, Scotland
agathe@dcs.gla.ac.uk

Carl Edward Rasmussen
Biological Cybernetics
Max Planck Institute
Spemann straße 38
72076 Tübingen, Germany
carl@tuebingen.mpg.de

October 5, 2003

1 Introduction

We consider in this report non-linear models that map an input D-dimensional column vector $\mathbf{x}$ into a single dimensional output $f(\mathbf{x})$. The non-linear mapping $f(\cdot)$ is implemented by means of a Gaussian process (GP) or a Relevance Vector Machine (RVM), see for example [Rasmussen, 1996] and [Tipping, 2001]. We are given a training data set $\mathcal{D} = \{\mathbf{x}_i, y_i\}_{i=1}^N$ where the target y_i relates to the input $\mathbf{x}_i$ through

$$y_i = f(\mathbf{x}_i) + \epsilon \tag{1}$$

where $\epsilon \sim \mathcal{N}(0, \sigma_\epsilon^2)$ is additive i.i.d. Gaussian noise of variance σ_ϵ^2.

We address in this technical report the issue of making predictions, that is of evaluating $f(\mathbf{x}^*)$ in the case where the input $\mathbf{x}^*$ is not deterministic, but rather a stochastic variable whose distribution we observe.

In Section 2 we give a brief description of Gaussian processes, and in Section 3 a brief description of the Relevance Vector Machine. In Section 4 we address the problem of predicting at an uncertain input, and derive exact expressions for the mean and the variance of the predictive distribution in the case of Gaussian kernels for GPs and Gaussian basis functions for RVMs. In Section 5 we apply our ability to predict on uncertain inputs to iterated time-series predictions, and describe how the uncertainty of the predictions can be propagated.

2 Gaussian Processes

The Gaussian Process (GP) modeling framework consists in placing a Gaussian prior over the function values. The joint distribution of the values of the function evaluated at a set of inputs is then a Gaussian of the form:

$$p(\mathbf{f}) \sim \mathcal{N}(0, \mathbf{\Sigma}) \tag{2}$$

where $\mathbf{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]^\top$, and we have set the mean to be zero. The covariance matrix $\mathbf{\Sigma}$ can be parameterized, and computed by means of a covariance function, such that

$$\mathrm{Cov}(f(\mathbf{x}_p), f(\mathbf{x}_q)) = \mathbf{\Sigma}_{pq} = C_{\mathrm{GP}}(\mathbf{x}_p, \mathbf{x}_q) \tag{3}$$

where $C_{\mathrm{GP}}(\cdot, \cdot)$ is the kernel, or covariance function. Notice that in this way the covariance between two model outputs is related to the distance between the two corresponding inputs under the kernel metric.

The prior on the function values, eq. (2), and the noise model, eq. (1) allow us to obtain the evidence,

$$p(\mathbf{y}|\theta) \sim \mathcal{N}(0, \mathbf{\Sigma} + \sigma_\epsilon^2\, \mathbf{I}), \tag{4}$$

where $\mathbf{y} = [y_1, \ldots, y_N]^\top$ and $\mathbf{I}$ is the identity matrix. θ are the parameters of the covariance function, or kernel, and of the noise. They are in fact the hyper-parameters of the model. The model is trained by estimating the value of θ that maximizes the evidence.

2.1 Gaussian Kernel

While there exists number of different ways of choosing kernels for Gaussian processes, see for example [Gibbs, 1997], we will in this report concentrate on a Gaussian type of kernel that is also a very common choice:

$$C_{\mathrm{GP}}(\mathbf{x}_p, \mathbf{x}_q) = \exp\left[-\frac{1}{2}(\mathbf{x}_p - \mathbf{x}_q)^\top \mathbf{\Lambda}^{-1}(\mathbf{x}_p - \mathbf{x}_q)\right] \ , \tag{5}$$

We consider $\mathbf{\Lambda} = \mathrm{diag}[\lambda_1^2, \ldots, \lambda_D^2]^\top$, allowing for different length scales in different input directions.

2.2 Prediction at $\mathbf{x}^*$

Given our set of training data $\mathcal{D}$, the predictive distribution of $f(\mathbf{x}^*)$ at a new input $\mathbf{x}^*$ is obtained by first building the joint probability distribution of $f(\mathbf{x}^*)$ and the training data $\mathbf{y}$. This distribution is obtained by augmenting the evidence, eq. (4) with $f(\mathbf{x}^*)$. The distribution $p(f(\mathbf{x}^*), \mathbf{y}|\mathbf{x}^*, \mathcal{D}, \theta)$ is Gaussian with zero-mean and covariance matrix $\tilde{\mathbf{K}}$ which we can write

$$\tilde{\mathbf{K}} = \begin{bmatrix} \mathbf{K} & \mathbf{k}(\mathbf{x}^*) \\ \mathbf{k}(\mathbf{x}^*)^\top & k \end{bmatrix} \tag{6}$$

where $\mathbf{K}$ is the covariance matrix of the evidence, $\mathbf{K} = \mathbf{\Sigma} + \sigma_\epsilon^2 \mathbf{I}$, also called 'data covariance matrix'. We also have that $k = C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}^*) = 1$ with our choice of kernel, and $\mathbf{k}$ is the vector of covariances between the new inputs and the training inputs, $\mathbf{k}(\mathbf{x}^*) = [C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_1) \ldots C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_n)]^\top$. By conditioning on the observed cases, we obtain the predictive distribution,

$$p(f(\mathbf{x}^*)|\mathbf{x}^*, \mathcal{D}) \sim \mathcal{N}(\mu(\mathbf{x}^*), \sigma^2(\mathbf{x}^*)), \tag{7}$$

where $\mu(\mathbf{x}^*)$ and $\sigma^2(x^*)$ are the mean and the variance of the Gaussian predictive distribution and are given by:

$$\mu(\mathbf{x}^*) = \mathbf{k}(\mathbf{x}^*)^\top \mathbf{K}^{-1} \mathbf{t} \tag{8}$$

$$\sigma^2(\mathbf{x}^*) = 1 - \mathbf{k}(\mathbf{x}^*)^\top \mathbf{K}^{-1} \mathbf{k}(\mathbf{x}^*). \tag{9}$$

Note that the input dependent variance of the estimate of the function value $\sigma^2(\mathbf{x}^*)$ should not be confused with the estimate of the output noise σ_ϵ^2, which is independent of the input $\mathbf{x}^*$.

3 The Relevance Vector Machine

The Relevance Vector Machine (RVM) is a probabilistic sparse kernel model, identical in functional form to the Support Vector Machine (SVM) model, which is

$$f(\mathbf{x}) = \sum_{j=1}^{M} \omega_j \phi_j(\mathbf{x}) + \omega_0 = \boldsymbol{\omega}^\top \boldsymbol{\phi}(\mathbf{x}) + \omega_0 \tag{10}$$

where $\{\omega_j\}$ are the model weights and $\phi_j(.)$ is an arbitrary basis function. We also write in vector form the weights vector $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_M]^\top$ and the responses of all basis functions $\boldsymbol{\phi}(\mathbf{x}) \equiv [\phi_1(\mathbf{x}), \ldots, \phi_M(\mathbf{x})]^\top$ to the input $\mathbf{x}$. In the RVM case, a prior is put over the weights, governed by a set of hyperparameters, one associated with each weight. For the specific choice of a factorized distribution with variance α_j^{-1}:

$$p(\omega_j|\alpha_j) = \sqrt{\frac{\alpha_j}{2\pi}} \exp\left(-\frac{1}{2}\alpha_j\, \omega_j^2\right) \tag{11}$$

the prior over functions $p(\mathbf{y}|\boldsymbol{\alpha})$ is $\mathcal{N}(0, \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^\top)$, i.e. a Gaussian process with covariance function given by

$$C_{\mathrm{RVM}}(\mathbf{x}_p, \mathbf{x}_q) = \sum_{j=1}^{M} \frac{1}{\alpha_j} \phi_j(x_p)\phi_j(x_q) \tag{12}$$

where $\boldsymbol{\alpha} = (\alpha_0, \dots, \alpha_N)^\top$ and $\mathbf{A} = \mathrm{diag}(\alpha_0, \dots, \alpha_N)$, and matrix $\boldsymbol{\Phi}$ is such that $\boldsymbol{\Phi}_{pq} = \phi_q(\mathbf{x}_p)$. Sparseness in terms of the basis vectors may arise if for some j $\alpha_j^{-1} = 0$. Then the jth basis function will not contribute to the model. Associating a basis function with each input point may thus lead to a model with a sparse representation in the inputs, i.e. the solution is only spanned by a subset of all input points. This is exactly the idea behind the relevance vector machine.

3.1 Gaussian basis functions

One way of associating a basis function with each training input point is to choose (non-normalized) Gaussian basis functions of the form:

$$\phi_j(\mathbf{x}) = \exp\left(-\frac{1}{2}(\mathbf{x} - \mathbf{x}_j)^\top \boldsymbol{\Lambda}^{-1}(\mathbf{x} - \mathbf{x}_j)\right) \tag{13}$$

where x_j are the training inputs, and the functions are isotropic with $\boldsymbol{\Lambda} = \lambda \mathbf{I}$.

The resulting covariance function is obtained by inserting expression (13) into equation (12), and is given by:

$$C_{\mathrm{RVM}}(\mathbf{x}_p, \mathbf{x}_q) = \sum_{j=1}^{M} \frac{1}{\alpha_j} \exp\left[-\frac{1}{2}\left((\mathbf{x}_p - \mathbf{x}_j)^\top \boldsymbol{\Lambda}^{-1}(\mathbf{x}_p - \mathbf{x}_j) + (\mathbf{x}_q - \mathbf{x}_j)^\top \boldsymbol{\Lambda}^{-1}(\mathbf{x}_q - \mathbf{x}_j)\right)\right] \tag{14}$$

One clear advantage of Gaussian basis functions is that they allow the exact analytical computation of the mean and variance of the predictive distribution for the case where the input is uncertain. These derivations are made in section 4.

Furthermore, it can be shown that for an infinite number of equally spaced Gaussian basis functions, equation (14) converges to the Gaussian covariance function of a GP, given by equation (5) [Mackay, 1997].

3.2 RVMs viewed as GPs

RVMs are Gaussian processes where the covariance between the training targets, based on equation (12), is given by the 'data covariance matrix' (see section 2.2) of the RVM:

$$\mathbf{K} = \sigma_\epsilon^2 \mathbf{I} + \boldsymbol{\Phi}\mathbf{A}^{-1}\boldsymbol{\Phi}^\top \quad \text{or} \quad \mathbf{K}_{pq} = \sigma_\epsilon^2 \delta_{pq} + \sum_{j=1}^{M} \frac{1}{\alpha_j} \phi_j(\mathbf{x}_p)\phi_j(\mathbf{x}_q) \tag{15}$$

The vector of covariances between the new prediction and the training targets is given by

$$\mathbf{k}(\mathbf{x}^*) = \mathbf{\Phi}\mathbf{A}^{-1}\boldsymbol{\phi}^* \quad \text{or} \quad [\mathbf{k}(\mathbf{x}^*)]_p = \sum_{j=1}^{M} \frac{1}{\alpha_j}\phi_j(\mathbf{x}_p)\phi_j(\mathbf{x}^*) \tag{16}$$

where we set $\boldsymbol{\phi}^* = \boldsymbol{\phi}(\mathbf{x}^*)$. Finally, the variance of the function value of the new prediction is given by $k = C_{\mathrm{RVM}}(\mathbf{x}^*, \mathbf{x}^*) = \boldsymbol{\phi}^{*\top}\mathbf{A}^{-1}\boldsymbol{\phi}^*$, for the RVM case this value is not necessarily 1.

Prediction of $f(\mathbf{x}^*)$ at a new input $\mathbf{x}^*$ can be computed using the same approach as for GPs (section 2.2), by computing the joint distribution of $f(\mathbf{x}^*)$ and the data first, and conditioning then on the data to obtain the predictive distribution $p(f(\mathbf{x}^*)|\mathbf{x}^*, \mathcal{D})$.

Plugging the expressions of $\mathbf{K}$, $\mathbf{k}(\mathbf{x}^*)$ and k for the RVM into equations (8) and (9) we obtain:

$$\mu(\mathbf{x}^*) = \boldsymbol{\phi}^{*\top}\boldsymbol{\omega}_{MP} \tag{17}$$

$$\sigma^2(\mathbf{x}^*) = \boldsymbol{\phi}^{*\top}\mathbf{\Sigma}^{-1}\boldsymbol{\phi}^* \tag{18}$$

where $\boldsymbol{\omega}_{MP}$ and $\mathbf{\Sigma}$ are the mean and the variance of the posterior distribution over the weights. They are given by:

$$\boldsymbol{\omega}_{MP} = \sigma_\epsilon^{-2}\mathbf{\Sigma}\mathbf{\Phi}^\top\mathbf{t} \tag{19}$$

$$\mathbf{\Sigma} = (\sigma_\epsilon^{-2}\mathbf{\Phi}^\top\mathbf{\Phi} + \mathbf{A})^{-1} \tag{20}$$

Equations (17)and (18) correspond to the classical expression of the mean and variance of the predictive distribution for the RVM [Tipping, 2001].

4 Prediction at $\mathbf{x}^* \sim \mathcal{N}(\mathbf{u}, \mathbf{S})$

The predictive distribution of the function value, $p(f^*)$ (we will for simplicity write from now on $f^* = f(\mathbf{x}^*)$), when the input is the random variable $\mathbf{x}^*$ with input distribution given by $p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) \sim \mathcal{N}(\mathbf{u}, \mathbf{S})$, is obtained by integrating over the input distribution:

$$p(f^*|\mathbf{u}, \mathbf{S}, \mathcal{D}) = \int p(f^*|\mathbf{x}^*, \mathcal{D})p(\mathbf{x}^*|\mathbf{u}, \mathbf{S})d\mathbf{x}^* \ , \tag{21}$$

where $p(f^*|\mathbf{x}^*) = \frac{1}{\sigma(\mathbf{x}^*)\sqrt{2\pi}} \exp\left[-\frac{1}{2}\frac{(f^*-\mu(\mathbf{x}^*))^2}{\sigma^2(\mathbf{x}^*)}\right]$ with mean and variance depending on the model. Equation (21) gives a distribution that we will call marginal predictive distribution, since the predictive distribution has been marginalized with respect to the input $\mathbf{x}^*$ and the conditioning is now on the parameters of the input distribution $\mathbf{u}$ and $\mathbf{S}$.

4.1 Numerical approximation

Given that the integral (21) is analytically intractable ($p(f^*|\mathbf{x}^*, \mathcal{D})$ is a complicated function of $\mathbf{x}^*$), one possibility is to perform a numerical approximation of the integral by a simple Monte-Carlo approach:

$$p(f^*|\mathbf{u}, \mathbf{S}) = \int p(f^*|\mathbf{x}^*, \mathcal{D}) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* \simeq \frac{1}{T} \sum_{t=1}^{T} p(f^*|\mathbf{x}^{*t}, \mathcal{D}) \ , \tag{22}$$

where $\mathbf{x}^{*t}$ are independent samples from $p(\mathbf{x}^*|\mathbf{u}, \mathbf{S})$.

4.2 Gaussian approximation

The analytical Gaussian approximation consists in only computing the mean and variance of $f^*|\mathbf{u}, \mathbf{S}, \mathcal{D}$. They are obtained using respectively the law of iterated expectations and law of conditional variances:

$$\begin{aligned} m(\mathbf{u}, \mathbf{S}) &= E_{\mathbf{x}^*}[E_{f^*}[f^*|\mathbf{x}^*]] = E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)] && (23) \\ v(\mathbf{u}, \mathbf{S}) &= E_{\mathbf{x}^*}[\mathrm{var}_{f^*}(f^*|\mathbf{x}^*)] + \mathrm{var}_{\mathbf{x}^*}(E_{f^*}[f^*|\mathbf{x}^*]) \\ &= E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + \mathrm{var}_{\mathbf{x}^*}(\mu(\mathbf{x}^*)) && (24) \end{aligned}$$

where $E_{\mathbf{x}^*}$ indicates the expectation under $\mathbf{x}^*$.

4.2.1 Approximate solution

The approximate solution consists in approximating the mean and the variance of the predictive distribution by their Taylor expansion, of order 1 and 2 respectively. The details can be found in [Girard et al., 2003] and more extended in [Girard et al., 2002].

4.2.2 Exact solution

What is exact in the exact solution is the estimate of the first and second order moments of the marginal predictive distribution [Quiñonero-Candela et al., 2003]. We are able to compute them exactly when the kernel we use for GPs and RVMs is of the Gaussian kind, as previously described. For deriving the following results, we use the fact that the mean and the variance of the predictive distribution for a deterministic input is given by very similar expressions both for GPs and RVMs. For the GP case we have:

$$\mu(\mathbf{x}^*) = \sum_i \beta_i C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i) \tag{25}$$

$$\sigma^2(\mathbf{x}^*) = 1 - \sum_i \sum_j {\mathbf{K}_{ij}}^{-1} C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i) C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_j) \tag{26}$$

where we define $\boldsymbol{\beta} = [\beta_1, \ldots, \beta_N]^\top = \mathbf{K}^{-1}\mathbf{t}$, and for the RVM case we have:

$$\mu(\mathbf{x}^*) = \sum_i \beta_i \phi_i(\mathbf{x}^*) \tag{27}$$

$$\sigma^2(\mathbf{x}^*) = \sum_i \sum_j \boldsymbol{\Sigma}_{ij}{}^{-1} \phi_i(\mathbf{x}^*)\phi_j(\mathbf{x}^*) \tag{28}$$

with $\boldsymbol{\beta} = \boldsymbol{\omega}_{MP}$, as given by (19), and $\boldsymbol{\Sigma}$ as given by (20). It is worth noticing that $C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i)$ and $\phi_i(\mathbf{x}^*)$ are given by the same expression:

$$C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i) = \phi_i(\mathbf{x}^*) = \exp\left(-\frac{1}{2}(\mathbf{x}^* - \mathbf{x}_j)^\top \boldsymbol{\Lambda}^{-1}(\mathbf{x}^* - \mathbf{x}_j)\right) \tag{29}$$

Computing the mean (for GPs and RVMs)

Using equation (25) and (27) we have:

$$\begin{aligned} m(\mathbf{u}, \mathbf{S}) = E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)] &= \int \mu(\mathbf{x}^*) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* = \sum_j \beta_j \int h(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* \\ &= \sum_j \beta_j l_j = \boldsymbol{\beta}^\top \boldsymbol{l} \end{aligned} \tag{30}$$

where $\boldsymbol{l}$ is a column vector with elements l_j, and $l_j = \int h(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^*$. We also have that $h(\mathbf{x}^*, \mathbf{x}_j) = C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_j)$ is as given by (5) for GPs and $h(\mathbf{x}^*, \mathbf{x}_j) = \phi_j(\mathbf{x}^*)$ is as given by (13) for RVMs.

Computing l_j is a simple task, since it is the integral in $\mathbf{x}^*$ of the product of two Gaussians in $\mathbf{x}^*$ except for a normalization constant. We can write

$$l_j = \frac{(2\pi)^{-D/2}|\boldsymbol{\Lambda}|^{-1/2}}{(2\pi)^{-D/2}|\boldsymbol{\Lambda}|^{-1/2}} \int h(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* \tag{31}$$

so that we have that $(2\pi)^{-D/2}|\boldsymbol{\Lambda}|^{-1/2} h(\mathbf{x}^*, \mathbf{x}_j)$ as a function of $\mathbf{x}^*$ is a normal distribution with mean $\mathbf{x}_j$ and covariance $\boldsymbol{\Lambda}$. Now, using the formula giving the multiplication of two Gaussian distributions[1] we have $l_j = (2\pi)^{D/2}|\boldsymbol{\Lambda}|^{1/2} z_c$ with

$$\begin{aligned} z_c &= (2\pi)^{-D/2}|C|^{1/2}|\boldsymbol{\Lambda}|^{-1/2}|\mathbf{S}|^{-1/2} \exp\left[-\frac{1}{2}(\mathbf{x}_j{}^\top \boldsymbol{\Lambda}^{-1}\mathbf{x}_j + \mathbf{u}^\top \mathbf{S}^{-1}\mathbf{u} - c_j^\top C^{-1} c_j)\right] \\ C &= (\boldsymbol{\Lambda}^{-1} + \mathbf{S}^{-1})^{-1} \\ c_j &= C(\boldsymbol{\Lambda}^{-1}\mathbf{x}_j + \mathbf{S}^{-1}\mathbf{u}) \end{aligned} \tag{32}$$

[1] $\mathcal{N}(a, A)\mathcal{N}(b, B) \propto \mathcal{N}(c, C)$ with $C = (A^{-1} + B^{-1})^{-1}$, $c = C(A^{-1}a + B^{-1}b)$ and normalizing constant $z_c = (2\pi)^{-D/2}|C|^{1/2}|A|^{-1/2}|B|^{-1/2} \exp\left[-\frac{1}{2}(a^\top A^{-1} a + b^\top B^{-1} b - c^\top C^{-1} c)\right]$.

With a little bit of algebra[2], l_j simplifies into

$$l_j = |\mathbf{\Lambda}^{-1}\mathbf{S} + I|^{-1/2} \exp\left(-\frac{1}{2}(\mathbf{u}-\mathbf{x}_j)^\top(\mathbf{S}+\mathbf{\Lambda})^{-1}(\mathbf{u}-\mathbf{x}_j)\right) \tag{33}$$

where I is the $D \times D$ identity matrix.

It is worth noticing that if the covariance matrix of the input distribution, $\mathbf{S}$, is the zero matrix, that is if the inputs are certain, then $l_j = h(\mathbf{u}, \mathbf{x}_j)$ and $m(\mathbf{u}, \mathbf{S}) = \mu(\mathbf{u})$ both for GPs and RVMs as would be expected.

Computing the variance for GPs

We have $v(\mathbf{u}, \mathbf{S}) = E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + \mathrm{var}_{\mathbf{x}^*}(\mu(\mathbf{x}^*)) = E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)^2] - E^2_{\mathbf{x}^*}[\mu(\mathbf{x}^*)]$, which translates into

$$\begin{aligned} v(\mathbf{u}, \mathbf{S}) = & \int \left(1 - \sum_i \sum_j C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i) {\mathbf{K}_{ij}}^{-1} C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_j)\right) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* \\ & + \sum_i \sum_j \beta_i \beta_j \int C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_i) C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^* \\ & - \left[\sum_j \beta_j \int C_{\mathrm{GP}}(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^*\right]^2, \end{aligned} \tag{34}$$

which simplifies into

$$\begin{aligned} v(\mathbf{u}, \mathbf{S}) &= 1 - \sum_i \sum_j ({\mathbf{K}_{ij}}^{-1} - \beta_i\beta_j) L_{ij} - [\sum_j \beta_j l_j]^2 \\ &= 1 - \mathrm{Tr}\left((\mathbf{K}^{-1} - \boldsymbol{\beta}\boldsymbol{\beta}^\top)\boldsymbol{L}\right) - \mathrm{Tr}\left(\boldsymbol{l}\boldsymbol{l}^\top\boldsymbol{\beta}\boldsymbol{\beta}^\top\right) \\ &= \sigma^2_{\mathrm{GP}}(\mathbf{u}) + \mathrm{Tr}\left(\mathbf{K}^{-1}(\mathbf{k}\mathbf{k}^\top - \boldsymbol{L})\right) + \mathrm{Tr}\left(\boldsymbol{\beta}\boldsymbol{\beta}^\top(\boldsymbol{L} - \boldsymbol{l}\boldsymbol{l}^\top)\right), \end{aligned} \tag{35}$$

where $\mathrm{Tr}(\cdot)$ is the trace operator, l_j is given by (33), $\mathbf{k}$ is a vector whose i-th element is $\mathbf{k}_i = C_{\mathrm{GP}}(\mathbf{u}, \mathbf{x}_i)$ and $\sigma^2_{\mathrm{GP}}(\mathbf{u})$ is the variance of the GP model output $f(\mathbf{u})$ evaluated at the deterministic input $\mathbf{u}$. The element L_{ij} of matrix $\boldsymbol{L}$ is the following integral:

$$L_{ij} = \int h(\mathbf{x}^*, \mathbf{x}_i) h(\mathbf{x}^*, \mathbf{x}_j) p(\mathbf{x}^*|\mathbf{u}, \mathbf{S}) d\mathbf{x}^*. \tag{36}$$

[2]We use here the following identities:

$$(\mathbf{S}+\mathbf{\Lambda})^{-1} = \mathbf{S}^{-1} - \mathbf{S}^{-1}(\mathbf{S}^{-1}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{S}^{-1} = \mathbf{\Lambda}^{-1} - \mathbf{\Lambda}^{-1}(\mathbf{S}^{-1}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{\Lambda}^{-1}$$
$$(\mathbf{S}+\mathbf{\Lambda})^{-1} = \mathbf{S}^{-1}(\mathbf{S}^{-1}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{\Lambda}^{-1} = \mathbf{\Lambda}^{-1}(\mathbf{S}^{-1}+\mathbf{\Lambda}^{-1})^{-1}\mathbf{S}^{-1}$$

where again $h(\mathbf{x}^*, \mathbf{x}_j) = C_{\text{GP}}(\mathbf{x}^*, \mathbf{x}_j)$ is as given by (5) for GPs and $h(\mathbf{x}^*, \mathbf{x}_j) = \phi_j(\mathbf{x}^*)$ is as given by (13) for RVMs. Using the expression for the product of two Gaussians twice this time, we obtain the following expression:

$$\begin{aligned} L_{ij} =& |2\boldsymbol{\Lambda}^{-1}\mathbf{S} + I|^{-1/2} \\ &\cdot \exp\left(-\frac{1}{2}\left[(\mathbf{u} - \mathbf{x}_d)^\top(\frac{\boldsymbol{\Lambda}}{2} + \mathbf{S})^{-1}(\mathbf{u} - \mathbf{x}_d) + (\mathbf{x}_i - \mathbf{x}_j)^\top(2\boldsymbol{\Lambda})^{-1}(\mathbf{x}_i - \mathbf{x}_j)\right]\right) \end{aligned} \tag{37}$$

where we define $\mathbf{x}_d = \frac{1}{2}(\mathbf{x}_i + \mathbf{x}_j)$. An expression of L_{ij} that gives more insight into the variance in equation (35) is

$$L_{ij} = \mathbf{k}_i\,\mathbf{k}_j\,|2\boldsymbol{\Lambda}^{-1}\mathbf{S} + I|^{-1/2}\exp\left(2(\mathbf{u} - \mathbf{x}_d)^\top\boldsymbol{\Lambda}^{-1}(2\boldsymbol{\Lambda}^{-1} + \mathbf{S}^{-1})^{-1}\boldsymbol{\Lambda}^{-1}(\mathbf{u} - \mathbf{x}_d)\right). \tag{38}$$

Notice that as $\mathbf{S}$ goes to zero, i.e. the input uncertainty collapses, $\boldsymbol{L}$ tends to $\mathbf{k}\,\mathbf{k}^\top$. Since we also know that when $\mathbf{S}$ goes to zero, $\boldsymbol{l}$ tends to $\mathbf{k}$, then it can easily be seen that $v(\mathbf{u}, \mathbf{S})$ tends to $\sigma^2_{\text{GP}}(\mathbf{u})$ as the input uncertainty disappears, as we would expect.

Computing the variance for RVMs

We have $v(\mathbf{u}, \mathbf{S}) = E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + \text{var}_{\mathbf{x}^*}(\mu(\mathbf{x}^*)) = E_{\mathbf{x}^*}[\sigma^2(\mathbf{x}^*)] + E_{\mathbf{x}^*}[\mu(\mathbf{x}^*)^2] - E^2_{\mathbf{x}^*}[\mu(\mathbf{x}^*)]$, which for RVMs, using (18) translates into

$$\begin{aligned} v(\mathbf{u}, \mathbf{S}) =& \int \sum_i \sum_j \boldsymbol{\Sigma}_{ij}{}^{-1}\phi_i(\mathbf{x}^*)\phi_j(\mathbf{x}^*)p(\mathbf{x}^*|\mathbf{u}, \mathbf{S})d\mathbf{x}^* \\ &+ \sum_i \sum_j \omega_i\omega_j \int \phi_i(\mathbf{x}^*)\phi_j(\mathbf{x}^*)p(\mathbf{x}^*|\mathbf{u}, \mathbf{S})d\mathbf{x}^* \\ &- \left[\sum_j \omega_j \int \phi_j(\mathbf{x}^*)p(\mathbf{x}^*|\mathbf{u}, \mathbf{S})d\mathbf{x}^*\right]^2 \end{aligned} \tag{39}$$

where ω_i is the i-th component of the maximum posterior estimate of the weights $\boldsymbol{\omega} = \boldsymbol{\omega}_{\text{MP}}$, given by (19), and $\boldsymbol{\Sigma}$ is the maximum posterior estimate of the covariance of the weights, eq. (20). The expression for the RVM variance can be simplified to

$$\begin{aligned} v(\mathbf{u}, \mathbf{S}) &= \sum_i \sum_j (\boldsymbol{\Sigma}_{ij}{}^{-1} + \omega_i\omega_j)L_{ij} - [\sum_j \omega_j l_j]^2 \\ &= \text{Tr}\left((\boldsymbol{\Sigma}^{-1} + \boldsymbol{\omega}\,\boldsymbol{\omega}^\top)\boldsymbol{L}\right) - \text{Tr}\left(\boldsymbol{\omega}\,\boldsymbol{\omega}^\top\boldsymbol{l}\boldsymbol{l}^\top\right) \\ &= \sigma^2_{\text{RVM}}(\mathbf{u}) + \text{Tr}\left(\boldsymbol{\Sigma}^{-1}(\boldsymbol{L} - \mathbf{k}\mathbf{k}^\top)\right) + \text{Tr}\left(\boldsymbol{\omega}\,\boldsymbol{\omega}^\top(\boldsymbol{L} - \boldsymbol{l}\boldsymbol{l}^\top)\right) \end{aligned} \tag{40}$$

again, l_j is as given by (33) and L_{ij} as given by (38) where in the RVM case $\mathbf{k}_i = \phi_i(\text{u})$. Notice here too that as $\mathbf{S}$ goes to zero, $\boldsymbol{L}$ tends to $\mathbf{k}\,\mathbf{k}^\top$, $\boldsymbol{l}$ tends to $\mathbf{k}$, and it can easily be seen that $v(\mathbf{u}, \mathbf{S})$ tends to $\sigma^2_{\text{RVM}}(\mathbf{u})$ as we would expect.

5 Time-Series Forecasting

Multiple step ahead time-series predictions can typically be performed under two approaches. The first approach consists in training the model to learn to predict on a fixed horizon of interest (direct method) and the second in training the model to learn to predict on a short horizon, and in reaching the horizon of interest by making repetitive one-step ahead predictions (iterative method). The direct method has the disadvantages that as the forecast horizon increases, the complexity of the non-linear mapping increases as well, and the number of available input-output training pairs decreases. For the iterative method, the complexity of the non-linear mapping is much lower, and the model only needs to be trained once no matter what the forecast horizon of interest is. The disadvantage of the iterative method is that as the forecast horizon increases, the performance is diminished by the accumulated uncertainty of the intermediate predictions.

Naïve iterative methods do not account for the accumulated uncertainty in the predictive distribution at a given horizon. We believe that the quality of the prediction of the iterative approach can be much improved if it is able to account for this uncertainty. We are concerned with the iterative approach and suggest to propagate the uncertainty as we predict ahead in time.

5.1 "Naïve" iterative k-step ahead prediction

Consider the discrete time series given by a set $\{y_t\}$ of samples ordered according to is an integer index t, and where the sampling period is constant. Consider as well the state-space model

$$\begin{cases} \mathbf{x}_t = [y_{t-1}, \ldots, y_{t-\mathcal{L}}]^\top \\ y_t = f(\mathbf{x}_t) + \epsilon \end{cases} \tag{41}$$

where the *state* $\mathbf{x}$ at time t is composed of previous outputs, up to a given lag[3] $\mathcal{L}$ and we have an additive (white) noise with variance σ_ϵ^2.

The naive iterative k-step ahead prediction method works as follows: it predicts only one time step ahead, using the estimate of the output of the current prediction, as well as previous outputs (up to the lag $\mathcal{L}$), as the input to the prediction of the next time step, until the prediction k steps ahead is made.

Using the model (41) and assuming the data is known up to time step T, the prediction

[3] We are not concerned with the identification of the lag and assume it has a known, fixed value.

of y_{T+k} is computed via

$$\begin{aligned}
\mathbf{x}_{T+1} = [y_T, y_{T-1}, \ldots, y_{T+1-\mathcal{L}}]^\top &\rightarrow f(\mathbf{x}_{T+1}) \sim \mathcal{N}(\mu(\mathbf{x}_{T+1}), \sigma^2(\mathbf{x}_{T+1})) \\
& \quad\ \hat{y}_{T+1} = \mu(\mathbf{x}_{T+1}) \\
\mathbf{x}_{T+2} = [\hat{y}_{T+1}, y_t, \ldots, y_{T+2-\mathcal{L}}]^\top &\rightarrow f(\mathbf{x}_{T+2}) \sim \mathcal{N}(\mu(\mathbf{x}_{T+2}), \sigma^2(\mathbf{x}_{T+2})) \\
& \quad\ \hat{y}_{T+2} = \mu(\mathbf{x}_{T+2}) \\
& \vdots \\
\mathbf{x}_{T+k} = [\hat{y}_{T+k-1}, \hat{y}_{T+k-2}, \ldots, \hat{y}_{T+k-\mathcal{L}}]^\top &\rightarrow f(\mathbf{x}_{T+k}) \sim \mathcal{N}(\mu(\mathbf{x}_{T+k}), \sigma^2(\mathbf{x}_{T+k})) \\
& \quad\ \hat{y}_{T+k} = \mu(\mathbf{x}_{T+k})
\end{aligned}$$

where the point estimates $\mu(x_{T+k-i})$ are computed using equation (8) for GPs, and equation (17) for RVMs. This setup does not account for the uncertainty induced by each successive prediction (variance $\sigma^2(x_{t+k-i}) + \sigma_\epsilon^2$ associated to each $\hat{y}$, given by (9) for GPs, by (18) for RVMs). For each recursion, the current state vector is considered deterministic, ignoring the the fact that the previous predictions that it contains as elements are in fact random variables distributed according to the predictive distribution given by the model.

5.2 Propagating the uncertainty

Using the results derived in the previous section, we propose to formally incorporate the uncertainty information about the future regressor. That is, as we predict ahead in time, we now view the lagged outputs as random variables. The input vectors, or states, will as well be random variables as they incorporate predictions recursively, $\mathbf{x}_t \sim \mathcal{N}(\mathbf{u}_t, \mathbf{S}_t)$.

Suppose as before, that data samples have been observed up to time T, we call Y_T the set observed samples up to that time, and we wish to predict k steps ahead. The predictive distribution we want to compute is

$$p(y_{T+k}|Y_T) = \int p(y_{T+k}|\mathbf{x}_{T+k})\, p(\mathbf{x}_{T+k}|Y_T), d\mathbf{x}_{T+k}. \tag{42}$$

The problem is that the distribution of the state at time $t = T+k$, $p(\mathbf{x}_{T+k}|Y_T)$, does depend on the distribution of the output at the previous time $t = T + K - 1$, $p(y_{T+k-1}|Y_T)$ since the random variable y_{T+k-1} is incorporated in the state $\mathbf{x}_{T+k}$. We thus need a recursive algorithm.

Let us now give a detailed description of the recursive estimation of the predictive distribution at time $t = T + k$. Bear in mind that the input distribution at step $t = T + n$ is given by $p(\mathbf{x}_{T+n}|Y_T) \sim \mathcal{N}(\mathbf{u}_{T+n}, \mathbf{S}_{T+n})$.

- at $t = T + 1$,

$$\mathbf{u}_{T+1} = \begin{bmatrix} y_T \\ \ldots \\ y_{T+1-\mathcal{L}} \end{bmatrix} \quad \text{and} \quad \mathbf{S}_{T+1} = \begin{bmatrix} 0 & \ldots & 0 \\ \ldots & \ldots & \ldots \\ 0 & \ldots & 0 \end{bmatrix} \tag{43}$$

and since $\mathbf{x}_{T+1}$ is not random,

$$p(y_{T+1}|Y_T) \sim \mathcal{N}(\mu(\mathbf{u}_{T+1}), \sigma^2(\mathbf{u}_{T+1}) + \sigma_\epsilon^2), \tag{44}$$

and we can use eqs. (8) and (9) for GPs, and (17) and (18) for RVMs.

- at $t = T + 2$,

$$\mathbf{u}_{T+2} = \begin{bmatrix} \mu(\mathbf{u}_{T+1}) \\ \dots \\ y_{T+2-\mathcal{L}} \end{bmatrix} \quad \text{and} \quad \mathbf{S}_{T+2} = \begin{bmatrix} \sigma^2(\mathbf{u}_{T+1}) + \sigma_\epsilon^2 & \dots & 0 \\ \dots & \dots & \dots \\ 0 & \dots & 0 \end{bmatrix} \tag{45}$$

and since $\mathbf{x}_{T+2}$ has now one random component,

$$p(y_{T+2}|Y_T) \sim \mathcal{N}(m(\mathbf{u}_{T+2}, \mathbf{S}_{T+2}), v(\mathbf{u}_{T+2}, \mathbf{S}_{T+2}) + \sigma_\epsilon^2)) \tag{46}$$

and we use eqs. (30) for the mean of both GPs and RVMs, and (35) for the variance for GPs, and (40) for the variance for RVMs.

...

- at $t = T + k$,

$$\mathbf{u}_{T+k} = \begin{bmatrix} m(\mathbf{u}_{T+k-1}, \mathbf{S}_{T+k-1}) \\ \dots \\ m(\mathbf{u}_{T+k-\mathcal{L}}, \mathbf{S}_{T+k-\mathcal{L}}) \end{bmatrix} \quad \text{and}$$

$$\mathbf{S}_{T+k} = \begin{bmatrix} v(\mathbf{u}_{T+k-1}, \mathbf{S}_{T+k-1}) + \sigma_\epsilon^2 & \mathrm{cov}(y_{T+k-1}, y_{T+k-2}) & \dots & \mathrm{cov}(y_{T+k-1}, y_{T+k-\mathcal{L}}) \\ \mathrm{cov}(y_{T+k-1}, y_{T+k-2}) & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots \\ \mathrm{cov}(y_{T+k-1}, y_{T+k-\mathcal{L}}) & \dots & \dots & v(\mathbf{u}_{T+k-\mathcal{L}}, \mathbf{S}_{T+k-\mathcal{L}} + \sigma_\epsilon^2) \end{bmatrix} \tag{47}$$

and since $\mathbf{x}_{T+k}$ has k random components if $k \leq \mathcal{L}$, or all random otherwise,

$$p(y_{T+k}|Y_T) \sim \mathcal{N}(m(\mathbf{u}_{t+k}, \mathbf{S}_{t+k}), v(\mathbf{u}_{t+k}, \mathbf{S}_{t+k}) + \sigma_\epsilon^2)). \tag{48}$$

5.2.1 Input distribution

We can easily find a general expression for the input distribution at any time $t = T + k$ by making a few observations. First, notice that both for RVMs and GPs we have that $m(\mathbf{u}, \mathbf{S}) = \mu(\mathbf{u})$ and $v(\mathbf{u}, \mathbf{S}) = \sigma^2(\mathbf{u})$ when $\mathbf{S}$ is the zero matrix. This allows us to use a single notation for the mean and the variance of the function value evaluated at a given input, be it random or not.

At time $t = T + k + 1$, the covariance matrix $\mathbf{S}_{T+k+1}$ of the state is computed by updating its first column (and row, since it is symmetric):

$$[\mathbf{S}_{T+k+1}]_{1:\mathcal{L},1} = \begin{bmatrix} v(\mathbf{u}_{T+k}, \mathbf{S}_{T+k}) + \sigma_\epsilon^2 \\ \mathrm{cov}(y_{T+k}, y_{T+k-1}) \\ \dots \\ \mathrm{cov}(y_{T+k}, y_{T+k-\mathcal{L}+1}) \end{bmatrix} = \begin{bmatrix} v(\mathbf{u}_{T+k}, \mathbf{S}_{T+k}) + \sigma_\epsilon^2 \\ \mathrm{cov}(y_{T+k}, \tilde{\mathbf{x}}_{T+k}) \end{bmatrix} \tag{49}$$

where $\tilde{\mathbf{x}}_{T+k}$ is a shorter version of the state vector $\mathbf{x}_{T+k}$ where the last element has been truncated: as we incorporate the new prediction in the state vector, we need to get rid of the oldest prediction. For simplicity in the notation, we will compute the covariance $\mathrm{cov}(y_{T+k}, \mathbf{x}_{T+k})$ and then throw away the last element of that vector to obtain $\mathrm{cov}(y_{T+k}, \tilde{\mathbf{x}}_{T+k})$. We have

$$\mathrm{cov}(y_{T+k}, \mathbf{x}_{T+k}) = E_{\mathbf{x}}[E_y[y_{T+k} \cdot \mathbf{x}_{t+K}]] - E[y_{T+k}]E[\mathbf{x}_{T+k}], \tag{50}$$

and we know that $E[y_{T+k}] = m(\mathbf{u}_{T+k}, \mathbf{S}_{T+k})$, eq. (30), and that $E[\mathbf{x}_{T+k}] = \mathbf{u}_{T+k}$. We also have that

$$E_{\mathbf{x}}[E_y[y_{T+k} \cdot \mathbf{x}_{t+K}]] = E_{\mathbf{x}}[\mu(\mathbf{x}_{T+k}) \cdot \mathbf{x}_{t+K}] = \int \mathbf{x}_{T+k}\, \mu(\mathbf{x}_{T+k})\, p(\mathbf{x}_{T+k})\, d\mathbf{x}_{T+k}, \tag{51}$$

which is a quite similar expression to equation (30), with $\mu(\mathbf{x}_{T+k})$ as given by (8) for GPs, and by (17) for RVMs.

Replacing and solving this integral in a similar way to what we did for the calculation of the mean, using again the properties of the product of two Gaussians, we get

$$E[y_{t_k} \mathbf{x}_{t_k}] = \sum_j \beta_j l_j c_j, \tag{52}$$

where l_j and c_j are given by (33) and (32) respectively. The covariance terms are then given by

$$\mathrm{cov}(y_{T+k}, \mathbf{x}_{T+k}) = \sum_j \beta_j l_j (c_j - \mathbf{u}_{T+k}). \tag{53}$$

6 Conclusion

We have considered regression using Gaussian Processes (GPs) and Relevance Vector Machines (RVMs) in the case where the test inputs are not deterministic, but rather random variables with known distribution. In particular, we have considered Gaussian distributed inputs. For such uncertain inputs, the predictive distribution of the regressors is not Gaussian anymore. For the widely used family of Gaussian covariancee functions, we approximate the predictive distribution of GPs and RVMs by a Gaussian distribution, by computing the exact value of the mean and the variance of the actual predictive distribution. As an application of our results, we consider iterated time-series prediction, where the predictions are fed back into the input vector to perform predictions further ahead in time. For GPs and RVMs the predictions are probabilistic, and there is need to propagate ahead the corresponding uncertainty as new predictions are made based on previous predictions. We derive analytic expressions for the covariance of the iterated predictions, and provide a framework for the obtention of realistic predictive distributions for iterated predictions. Experiments where we perform multiple-step ahead time-series forecasting are described in [Girard et al., 2003, Quiñonero-Candela et al., 2003], where we obtain much better predictive distributions than when using a naïve approach. Our expressions have also recently been applied to reinforcement learning [Rasmussen and Kuss, 2004].

References

[Gibbs, 1997] Gibbs, M. N. (1997). *Bayesian Gaussian Processes for Regression and Classification*. PhD thesis, Cambridge University.

[Girard et al., 2002] Girard, A., Rasmussen, C. E., and Murray-Smith, R. (2002). Gaussian process priors with uncertain inputs: Multiple-step ahead prediction. Technical report, Department of Computing Science, Glasgow University. http://www.dcs.gla.ac.uk/~agathe/reports.html.

[Girard et al., 2003] Girard, A., Rasmussen, C. E., Quiñonero-Candela, J., and Murray-Smith, R. (2003). Gaussian process with uncertain inputs - application to multiple-step ahead time-series forecasting. In *Advances in Neural Information Processing Systems 15*. MIT Press.

[Mackay, 1997] Mackay, D. J. C. (1997). Gaussian Processes: A replacement for supervised Neural Networks? Technical report, Cavendish Laboratory, Cambridge University. Lecture notes for a tutorial at NIPS 1997.

[Quiñonero-Candela et al., 2003] Quiñonero-Candela, J., Girard, A., Larsen, J., and Rasmussen, C. E. (2003). Propagation of uncertainty in bayesian kernels models - application to multiple-step ahead forecasting. In *International Conference on Acoustics, Speech and Signal Processing*.

[Rasmussen, 1996] Rasmussen, C. E. (1996). *Evaluation of Gaussian Processes and Other Methods for Non-linear Regression*. PhD thesis, Dept. of Computer Science, University of Toronto.

[Rasmussen and Kuss, 2004] Rasmussen, C. E. and Kuss, M. (2004). Gaussian processes in reinforcement learning. In *Advances in Neural Information Processing Systems 16*. MIT Press.

[Tipping, 2001] Tipping, M. E. (2001). Sparse bayesian learning and the relevance vector machine. *Journal of Machine Learning Research*, 1:211–244.