Gaussian processes with monotonicity information

Jaakko Riihimäki
Dept. of Biomedical Engineering
and Computational Science
Aalto University
jaakko.riihimaki@tkk.fi

Aki Vehtari
Dept. of Biomedical Engineering
and Computational Science
Aalto University
aki.vehtari@tkk.fi

Abstract

A method for using monotonicity information in multivariate Gaussian process regression and classification is proposed. Monotonicity information is introduced with virtual derivative observations, and the resulting posterior is approximated with expectation propagation. Behaviour of the method is illustrated with artificial regression examples, and the method is used in a real world health care classification problem to include monotonicity information with respect to one of the covariates.

1 INTRODUCTION

In modelling problems there is sometimes a priori knowledge available, concerning the function to be learned, which can be used to improve the performance of the model. Such information may be inaccurate, and be related to the behaviour of the output variable as a function of the input variables. For instance, instead of having measurements on derivatives, the output function can be known to be monotonic with respect to an input variable.

For univariate and multivariate additive functions, the monotonicity can be forced by construction, see e.g. (Shively et al., 2009). A generic approach for multivariate models was proposed by (Sill and Abu-Mostafa, 1997), who introduced monotonicity information to multilayer perceptron (MLP) neural networks using *hints* that are virtual observations placed appropriately in the input space. See also (Lampinen and Selonen, 1997) for more explicit formulation. However, use of hints can be problematic with MLP due to nonstationarity of the smoothness properties, and difficulties in the integration over the posterior distribution.

In this paper, we propose a method similar to hint approach for including mononicity information into a Gaussian process (GP) model using virtual derivative observations with a Gaussian distribution. In Gaussian processes smoothness can be controlled in a more systematic way than in MLP by the selection of a covariance function. In this work, integrals are approximated using the fast expectation propagation (EP) algorithm.

We first illustrate the behaviour and examine the performance of the approach with artificial univariate regression data sets. We then illustrate the benefits of monotonicity information in a real world multivariate classification problem with monotonicity for one of the covariates.

Section 2 presents briefly the Gaussian process with derivative observations, and Section 3 describes the proposed method. In Section 4 experiments are shown, and conclusions are drawn in Section 5.

2 GAUSSIAN PROCESSES AND DERIVATIVE OBSERVATIONS

Gaussian process (GP) is a flexible nonparametric model in which the prior is set directly over functions of one or more input variables, see e.g. (O'Hagan, 1978; MacKay, 1998; Neal, 1999; Rasmussen and Williams, 2006). Gaussian process models are attractive in modelling complex phenomena since they allow possible nonlinear effects, and if there are dependencies between covariates, GP can handle these interactions implicitly.

Let $\mathbf{x}$ denote a D-dimensional covariate vector, and the matrix X, of size $N \times D$, all N training input vectors.

Appearing in Proceedings of the 13^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2010, Chia Laguna Resort, Sardinia, Italy. Volume 9 of JMLR: W&CP 9.

We assume a zero mean Gaussian process prior

$$p(\mathbf{f}|X) = \mathcal{N}(\mathbf{f}|\mathbf{0}, K(X,X)), \quad (1)$$

where $\mathbf{f}$ is a vector of N latent values. The covariance matrix $K(X,X)$ between the latent values depends on the covariates, and is determined by the covariance function. Throughout this work, we use the stationary squared exponential covariance function, which produces smooth functions, given by

$$\begin{aligned}\mathrm{Cov}\left[f^{(i)}, f^{(j)}\right] &= K(\mathbf{x}^{(i)}, \mathbf{x}^{(j)}) \\ &= \eta^2 \exp\left(-\tfrac{1}{2}\textstyle\sum_{d=1}^{D} \rho_d^{-2}(x_d^{(i)} - x_d^{(j)})^2\right), \quad (2)\end{aligned}$$

where η and $\boldsymbol{\rho} = \{\rho_1, \ldots, \rho_D\}$ are the hyperparameters of the GP model.

In the regression case, having the vector $\mathbf{y}$ of N noisy outputs, we assume the Gaussian relationship between the latent function values and the noisy observations

$$p(\mathbf{y}|\mathbf{f}) = \mathcal{N}(\mathbf{y}|\mathbf{f}, \sigma^2 I),$$

where σ^2 is the noise variance and I is the identity matrix. Given the training data X and $\mathbf{y}$, the conditional predictive distribution for a new covariate vector $\mathbf{x}^*$ is Gaussian with mean and variance

$$E[f^*|\mathbf{x}^*, \mathbf{y}, X, \boldsymbol{\theta}] = K(\mathbf{x}^*, X)(K(X,X) + \sigma^2 I)^{-1}\mathbf{y} \quad (3)$$

$$\begin{aligned}\mathrm{Var}[f^*|\mathbf{x}^*, \mathbf{y}, X, \boldsymbol{\theta}] &= K(\mathbf{x}^*, \mathbf{x}^*) - K(\mathbf{x}^*, X) \\ &\times (K(X,X) + \sigma^2 I)^{-1} K(X, \mathbf{x}^*), \quad (4)\end{aligned}$$

where $\boldsymbol{\theta} = \{\eta, \boldsymbol{\rho}, \sigma\}$.

Instead of integrating out the hyperparameters, for simplicity we find a point estimate for the values of the hyperparameters $\boldsymbol{\theta}$, by optimising the marginal likelihood

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

and in the computations we use the logarithm of the marginal likelihood

$$\begin{aligned}\log p(\mathbf{y}|X, \boldsymbol{\theta}) &= -\frac{1}{2}\mathbf{y}^T (K(X,X) + \sigma^2 I)^{-1}\mathbf{y} \\ &-\frac{1}{2}\log|K(X,X) + \sigma^2 I| - \frac{N}{2}\log(2\pi).\end{aligned}$$

The derivative of a Gaussian process remains a Gaussian process because differentiation is a linear operator, e.g., (Rasmussen, 2003; Solak et al., 2003). This makes it possible to include derivative observations in the GP model, or to compute predictions about derivatives. The mean of the derivative is equal to the derivative of the latent mean

$$E\left[\frac{\partial f^{(i)}}{\partial x_d^{(i)}}\right] = \frac{\partial E\left[f^{(i)}\right]}{\partial x_d^{(i)}}.$$

Likewise, the covariance between a partial derivative and a function value satisfies

$$\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_d^{(i)}}, f^{(j)}\right] = \frac{\partial}{\partial x_d^{(i)}}\mathrm{Cov}\left[f^{(i)}, f^{(j)}\right],$$

and the covariance between partial derivatives

$$\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_d^{(i)}}, \frac{\partial f^{(j)}}{\partial x_g^{(j)}}\right] = \frac{\partial^2}{\partial x_d^{(i)} \partial x_g^{(j)}}\mathrm{Cov}\left[f^{(i)}, f^{(j)}\right].$$

For the squared exponential covariance function (2), the covariances between function values and partial derivatives are given by

$$\begin{aligned}\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_g^{(i)}}, f^{(j)}\right] &= \eta^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\rho_d^{-2}(x_d^{(i)} - x_d^{(j)})^2\right) \\ &\times(-\rho_g^{-2}(x_g^{(i)} - x_g^{(j)})),\end{aligned}$$

and between partial derivatives by

$$\begin{aligned}\mathrm{Cov}\left[\frac{\partial f^{(i)}}{\partial x_g^{(i)}}, \frac{\partial f^{(j)}}{\partial x_h^{(j)}}\right] &= \eta^2 \exp\left(-\frac{1}{2}\sum_{d=1}^{D}\rho_d^{-2}(x_d^{(i)} - x_d^{(j)})^2\right) \\ &\times\rho_g^{-2}\left(\delta_{gh} - \rho_h^{-2}(x_h^{(i)} - x_h^{(j)})(x_g^{(i)} - x_g^{(j)})\right),\end{aligned}$$

where $\delta_{gh} = 1$ if $g = h$, and 0 otherwise. For instance, having observed the values of $\mathbf{y}$, mean of the derivative of the latent function f with respect to the dimension d, is

$$E\left[\frac{\partial f^*}{\partial x_d^*}\right] = \frac{\partial K(\mathbf{x}^*, X)}{\partial x_d^*}(K(X,X) + \sigma^2 I)^{-1}\mathbf{y},$$

and the variance

$$\begin{aligned}\mathrm{Var}\left[\frac{\partial f^*}{\partial x_d^*}\right] &= \frac{\partial^2 K(\mathbf{x}^*, \mathbf{x}^*)}{\partial x_d^* \partial x_d^*} - \frac{\partial K(\mathbf{x}^*, X)}{\partial x_d^*} \\ &\times(K(X,X) + \sigma^2 I)^{-1}\frac{\partial K(X, \mathbf{x}^*)}{\partial x_d^*},\end{aligned}$$

similar to the equations (3) and (4). To use the derivative observations in the Gaussian process, the observation vector $\mathbf{y}$ can be extended to include also the derivative observations, and the covariance matrix between the observations can be extended to include the covariances between the observations and partial derivatives, and the covariances between the partial derivatives.

3 EXPRESSING MONOTONICITY INFORMATION

In this section we present the method for introducing monotonicity information to a Gaussian process

model. Instead of evaluating the derivative everywhere, it is possible to choose a finite number of locations where the derivative is evaluated when the function is smooth.

Monotonicity conditions are the following: at the operating point $\mathbf{x}^{(i)}$, the derivative of the target function is non-negative with respect to the input dimension d_i. We use the notation $m_{d_i}^{(i)}$ for the derivative information where monotonicity is with respect to the dimension d_i at the location $\mathbf{x}^{(i)}$. We denote with $\mathbf{m}$ a set of M derivative points inducing the monotonicity at the operating points X_m (the matrix of size $M \times D$).

To express this monotonicity, the following probit likelihood

$$\begin{aligned} p\left(m_{d_i}^{(i)} \middle| \frac{\partial f^{(i)}}{\partial x_{d_i}^{(i)}}\right) &= \Phi\left(\frac{\partial f^{(i)}}{\partial x_{d_i}^{(i)}} \frac{1}{\nu}\right) \\ \Phi(z) &= \int_{-\infty}^{z} \mathcal{N}(t|0,1)dt, \end{aligned} \tag{5}$$

is assumed for the derivative observation. By using the probit function instead of step function the likelihood tolerates small errors. The probit function in (5) approaches the step function when $\nu \to 0$, and in all experiments in this work we fixed $\nu = 10^{-6}$.However, it is possible to adjust the steepness of the step, and thereby control the strictness of monotonicity information with the parameter ν in the likelihood.

To include the information from this likelihood into the GP model, the expectation propagation algorithm (Minka, 2001) is used to form virtual derivative observations.

For now we assume we have a set of locations X_m where the function is known to be monotonic. By assuming a zero mean Gaussian process prior (1) for latent function values, the joint prior for latent values and derivatives is given by

$$p(\mathbf{f}, \mathbf{f}'|X, X_m) = \mathcal{N}(\mathbf{f}_{\text{joint}}|\mathbf{0}, K_{\text{joint}}),$$

where

$$\mathbf{f}_{\text{joint}} = \begin{bmatrix} \mathbf{f} \\ \mathbf{f}' \end{bmatrix}, \text{and} \quad K_{\text{joint}} = \begin{bmatrix} K_{\mathbf{f},\mathbf{f}} & K_{\mathbf{f},\mathbf{f}'} \\ K_{\mathbf{f}',\mathbf{f}} & K_{\mathbf{f}',\mathbf{f}'} \end{bmatrix}. \tag{6}$$

In (6) $\mathbf{f}'$ is used as a shorthand notation for the derivative of latent function $\mathbf{f}$ with respect to some of the input dimensions, and the subscripts of K denote the variables between which the covariance is computed.

Using the Bayes rule, the joint posterior is obtained by

$$p(\mathbf{f}, \mathbf{f}'|\mathbf{y}, \mathbf{m}) = \frac{1}{Z} p(\mathbf{f}, \mathbf{f}'|X, X_m) p(\mathbf{y}|\mathbf{f}) p(\mathbf{m}|\mathbf{f}') \tag{7}$$

where

$$p(\mathbf{m}|\mathbf{f}') = \prod_{i=1}^{M} \Phi\left(\frac{\partial f^{(i)}}{\partial x_{d_i}^{(i)}} \frac{1}{\nu}\right) \tag{8}$$

and the normalisation term is

$$Z = \int p(\mathbf{f}, \mathbf{f}'|X, X_m) p(\mathbf{y}|\mathbf{f}) p(\mathbf{m}|\mathbf{f}') d\mathbf{f} d\mathbf{f}'.$$

Since the likelihood for the derivative observations in (8) is not Gaussian, the posterior is analytically intractable. We apply the EP algorithm, and compute the Gaussian approximation for the posterior distribution. The local likelihood approximations given by EP are then used in the model as virtual derivative observations, in addition to the observations $\mathbf{y}$.

The EP algorithm approximates the posterior distribution in (7) with

$$\begin{aligned} q(\mathbf{f}, \mathbf{f}'|\mathbf{y}, \mathbf{m}) = \frac{1}{Z_{\text{EP}}} p(\mathbf{f}, \mathbf{f}'|X, X_m) p(\mathbf{y}|\mathbf{f}) \\ \times \prod_{i=1}^{M} t_i(\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2), \end{aligned}$$

where $t_i(\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) = \tilde{Z}_i \mathcal{N}(f_i'|\tilde{\mu}_i, \tilde{\sigma}_i^2)$ are local likelihood approximations with site parameters $\tilde{Z}_i$, $\tilde{\mu}_i$ and $\tilde{\sigma}_i^2$. The posterior is a product of Gaussian distributions, and can be simplified to

$$q(\mathbf{f}, \mathbf{f}'|\mathbf{y}, \mathbf{m}) = \mathcal{N}(\mathbf{f}_{\text{joint}}|\boldsymbol{\mu}, \Sigma). \tag{9}$$

The posterior mean is $\boldsymbol{\mu} = \Sigma \tilde{\Sigma}_{\text{joint}}^{-1} \tilde{\boldsymbol{\mu}}_{\text{joint}}$ and the covariance $\Sigma = (K_{\text{joint}}^{-1} + \tilde{\Sigma}_{\text{joint}}^{-1})^{-1}$, where

$$\tilde{\boldsymbol{\mu}}_{\text{joint}} = \begin{bmatrix} \mathbf{y} \\ \tilde{\boldsymbol{\mu}} \end{bmatrix}, \quad \text{and} \quad \tilde{\Sigma}_{\text{joint}} = \begin{bmatrix} \sigma^2 I & 0 \\ 0 & \tilde{\Sigma} \end{bmatrix}. \tag{10}$$

In (10) $\tilde{\boldsymbol{\mu}}$ is the vector of site means $\tilde{\mu}_i$, and $\tilde{\Sigma}$ is a diagonal matrix with site variances $\tilde{\sigma}_i^2$ on the diagonal.

The desired posterior marginal moments with the likelihood (5) are updated as

$$\begin{aligned} \hat{Z}_i &= \Phi(z_i) \\ \hat{\mu}_i &= \mu_{-i} + \frac{\sigma_{-i}^2 \mathcal{N}(z_i|0,1)}{\Phi(z_i)\nu\sqrt{1 + \sigma_{-i}^2/\nu^2}} \\ \hat{\sigma}_i^2 &= \sigma_{-i}^2 - \frac{\sigma_{-i}^4 \mathcal{N}(z_i|0,1)}{\Phi(z_i)(\nu^2 + \sigma_{-i}^2)} \left(z_i + \frac{\mathcal{N}(z_i|0,1)}{\Phi(z_i)}\right), \end{aligned}$$

where

$$z_i = \frac{\mu_{-i}}{\nu\sqrt{1 + \sigma_{-i}^2/\nu^2}},$$

and μ_{-i} and σ_{-i}^2 are the parameters of the cavity distribution in EP. These equations are similar to those

of binary classification with the probit likelihood, and the EP algorithm is otherwise similar as presented, for example, in chapter 3 of (Rasmussen and Williams, 2006).

The normalisation term is approximated with EP as

$$\begin{aligned} Z_{\text{EP}} &= q(\mathbf{y}, \mathbf{m}|X, X_m, \boldsymbol{\theta}) \\ &= \int p(\mathbf{f}, \mathbf{f}'|X, X_m) p(\mathbf{y}|\mathbf{f}) \prod_{i=1}^{M} t_i(\tilde{Z}_i, \tilde{\mu}_i, \tilde{\sigma}_i^2) d\mathbf{f} d\mathbf{f}' \\ &= \int \mathcal{N}(\mathbf{f}_{\text{joint}}|\boldsymbol{\mu}, \Sigma) d\mathbf{f} d\mathbf{f}' Z_{\text{joint}}^{-1} \prod_{i=1}^{M} \tilde{Z}_i \\ &= Z_{\text{joint}}^{-1} \prod_{i=1}^{M} \tilde{Z}_i, \end{aligned}$$

where the normalisation term of the product of Gaussians is

$$\begin{aligned} Z_{\text{joint}}^{-1} &= (2\pi)^{-(N+M)/2} |K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}}|^{-1/2} \\ &\times \exp\left(-\frac{1}{2} \tilde{\boldsymbol{\mu}}_{\text{joint}}^T (K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}})^{-1} \tilde{\boldsymbol{\mu}}_{\text{joint}}\right), \end{aligned}$$

and the remaining terms $\tilde{Z}_i$ are the normalisation constants from EP.

In the computations we use the logarithm of the normalisation term, and after the convergence of EP, the approximation for the logarithm of marginal likelihood is computed as

$$\begin{aligned} \log Z_{\text{EP}} &= -\frac{1}{2} \log |K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}}| \\ &- \frac{1}{2} \tilde{\boldsymbol{\mu}}_{\text{joint}}^T (K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}})^{-1} \tilde{\boldsymbol{\mu}}_{\text{joint}} + \sum_{i=1}^{M} \frac{(\mu_{-i} - \tilde{\mu}_i)^2}{2(\sigma_{-i}^2 + \tilde{\sigma}_i^2)} \\ &+ \sum_{i=1}^{M} \log \Phi\left(\frac{\mu_{-i}}{\nu \sqrt{1 + \sigma_{-i}^2/\nu^2}}\right) + \frac{1}{2} \sum_{i=1}^{M} \log(\sigma_{-i}^2 + \tilde{\sigma}_i^2). \end{aligned}$$

The values for the hyperparameters are found by optimising the logarithm of the joint marginal likelihood approximation for the observations and derivative information. To use the virtual derivative samples in the GP model predictions, the approximative predictive mean and variance for the latent variable can be computed with

$$\begin{aligned} &E[f^*|\mathbf{x}^*, \mathbf{y}, X, \mathbf{m}, X_m] = K_{*,\mathbf{f}_{\text{joint}}} (K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}})^{-1} \tilde{\boldsymbol{\mu}}_{\text{joint}} \\ &\text{Var}[f^*|\mathbf{x}^*, \mathbf{y}, X, \mathbf{m}, X_m] = K_{*,*} - K_{*,\mathbf{f}_{\text{joint}}} \\ &\times (K_{\text{joint}} + \tilde{\Sigma}_{\text{joint}})^{-1} K_{\mathbf{f}_{\text{joint}},*} \end{aligned}$$

analogously to the standard GP prediction equations (3) and (4).

In classification examples, we assume the probit likelihood for class observations

$$p(\mathbf{c}|\mathbf{f}) = \prod_{i=1}^{N} \Phi(f^{(i)} c^{(i)}),$$

where now $c^{(i)} = \{-1, 1\}$ describes the two output classes. We apply the expectation propagation algorithm for both the class observations and virtual derivative observations. EP approximates the joint posterior of $\mathbf{f}$ and $\mathbf{f}'$ similarly to the regression case in (9), except that the vector of observations $\mathbf{y}$, and noise $\sigma^2 I$ in (10) are now replaced with site approximations $\tilde{\boldsymbol{\mu}}_{\text{class}}$ and $\tilde{\Sigma}_{\text{class}}$, denoting the mean and variance site terms given by EP, and associated with class observations.

The parameter ν in likelihood for virtual derivative observation causes the desired posterior marginal moments to be computed slightly differently, depending on whether the moments are computed for class observations or derivative observations. For class observations, the moments are given, for example, in chapter 3 of (Rasmussen and Williams, 2006), and for virtual derivative observations moments are computed as in the regression case.

The values for the hyperparameters are found by optimising the joint marginal likelihood approximation of class observations and virtual derivative observations. The normalisation term is computed as in regression, except that again $\mathbf{y}$ and noise $\sigma^2 I$ in (10) are replaced with site terms $\tilde{\boldsymbol{\mu}}_{\text{class}}$ and $\tilde{\Sigma}_{\text{class}}$. Furthermore, in the computation of the normalisation of joint posterior, the normalisation site terms $\tilde{Z}_{\text{class}}$ of class observations are also taken into account.

In classification, the predictions for the latent values using the class observations and virtual derivative observations are made by using the extended vector of site means and extended covariance matrix having site variances on the diagonal.

3.1 PLACING THE VIRTUAL DERIVATIVE POINTS

In low dimensional problems the derivative points can be placed on a grid to approximate monotonicity. A drawback is that the number of grid points increases exponentially with regard to the number of input dimensions. In higher dimensional cases the distribution for X can be assumed to be the empirical distribution of observations X, and the virtual points can be chosen to be at the unique locations of the observed input data points. Alternatively, a random subset of points from the empirical distribution can be chosen.

If the distance between derivative points is short

enough compared to the lengthscale, then monotonicity information affects also between the virtual points according to the correlation structure. Due to the computational scaling $\mathcal{O}((N+M)^3)$, it may be necessary to use a smaller number of derivative points. In such a case, a general solution is to use the GP predictions about the values of derivatives at the observed unique data points. The probability of the derivative being negative is computed, and at the locations where this probability is high, virtual derivative points are placed. After conditioning on the virtual data points, the new predictions for the derivative values at the remaining unique observed points can be computed, and virtual derivative points can be added, moved or removed if needed. This iteration can be continued to assure monotonicity at the interesting regions.

To place the virtual derivative points between the observed data points, or outside the convex hull of the observed X, a more elaborate distribution model for X is needed. Again, the probability of derivative being negative can easily be computed, and more virtual derivative points can be placed on locations where this probability is high.

4 EXPERIMENTAL RESULTS

4.1 DEMONSTRATION

An example of Gaussian process regression with monotonicity information is shown in Figure 1. Subfigure (a) illustrates the GP prediction (mean + 95% interval) without monotonicity information, with hyperparameter values found by optimising the marginal likelihood. Subfigures (b) and (c) show the predictions with monotonicity information, with hyperparameter values that maximise the approximation of the joint marginal likelihood. Short vertical lines in (b) and (c) are the locations of virtual derivative points. In Subfigure (b), the locations of virtual points are found by choosing a subset amongst the observed data points, on the locations where the probability of derivative being negative is large before conditioning to any monotonicity information (the derivative seen in Subfigure (d)). In Subfigure (c) the virtual points are placed on a grid. The predictions in (b) and (c) are similar, and (e) and (f) illustrate the corresponding derivatives of the latent functions. Since the probability of derivative being negative in (e) and (f) at the observed data range is very low, adding more virtual derivative points is unnecessary.

The effect of the monotonicity information is illustrated also in Figure 2. Subfigures (a)-(c) show the case without monotonicity information: (a) shows the marginal likelihood as a function of lengthscale and noise variance parameters (signal magnitude is fixed to be one), and (b) and (c) show two different solutions (mean + 95% interval) at two different modes shown in (a). The mode with the shorter lengthscale and smaller noise variance (function estimate in (b)) has higher density. Subfigures (d)-(f) show the case with monotonicity information. Subfigure (d) shows the approximated marginal likelihood for the observations and virtual derivative observations. Now the mode corresponding to the longer lengthscale and the monotone function shown in (f) has much higher density. Since virtual observations are not placed densely, there is still another mode at shorter lengthscale (function estimate in (e)) although with much lower density. This shows the importance of having enough virtual observations, and this second mode would eventually vanish if the number of virtual observations would be increased.

4.2 ARTIFICIAL EXAMPLES

We test the Gaussian process model with monotonicity information by performing simulation experiments on four artificial data sets. We consider the following functions:

(a) $f(x) = 0$ if $x < 0.5$, $f(x) = 2$ if $x \geq 0.5$ (step);

(b) $f(x) = 2x$ (linear);

(c) $f(x) = \exp(1.5x)$ (exponential);

(d) $f(x) = 2/\{1 + \exp(-8x + 4)\}$ (logistic),

and draw observations from the model $y_i = f(x_i) + \epsilon_i$, where x_i and ϵ_i are i.i.d. samples from the uniform distribution $U(x_i|0,1)$, and from the Gaussian $\mathcal{N}(\epsilon_i|0,1)$. We normalise x and y to have mean zero and standard deviation 0.5.

For the Gaussian process with monotonicity information, we introduce 10 virtual observations spaced equally between the observed minimum and maximum values of x variable. We compare the results of the model to a Gaussian process with no monotonicity information. The performances of the models are evaluated using a root-mean-square error (RMSE). The estimates for RMSE are evaluated against true function values on 500 equally spaced x-values.

Table 1 summarises the simulation results. The results are based on simulations repeated 50 times. Two sample sizes, $N = 100$ and $N = 200$, were used in the simulations. For the step function, the GP model with monotonicity information performs worse than GP without monotonicity assumption because the proposed method has tendency to favour smooth increasing functions. In a case of heavy truncation by the

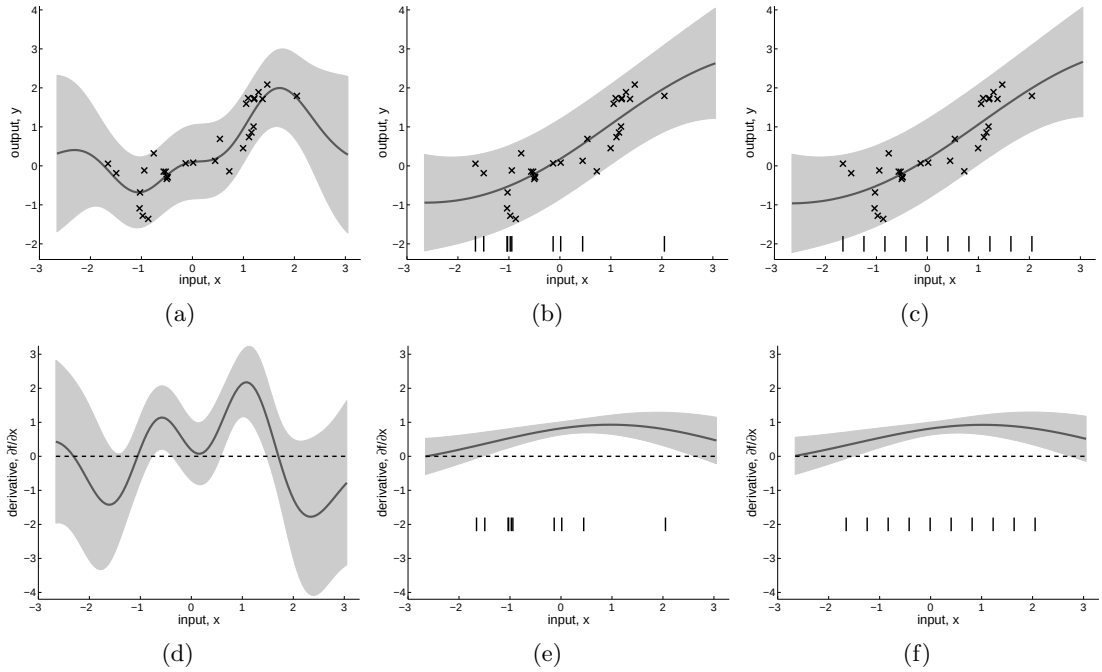

Figure 1: Example of Gaussian process solution (mean + 95% interval) without monotonicity information (a), and the corresponding derivative of the latent function (d). Subfigures (b) and (c) illustrate the solutions with monotonicity information, and the corresponding derivatives are shown in (e) and (f). The virtual derivative observations (shown with short vertical lines) in (b) are placed on locations where the probability of derivative being negative is large (seen in Subfigure (d)). In Subfigure (c) the derivative points are placed on a grid.

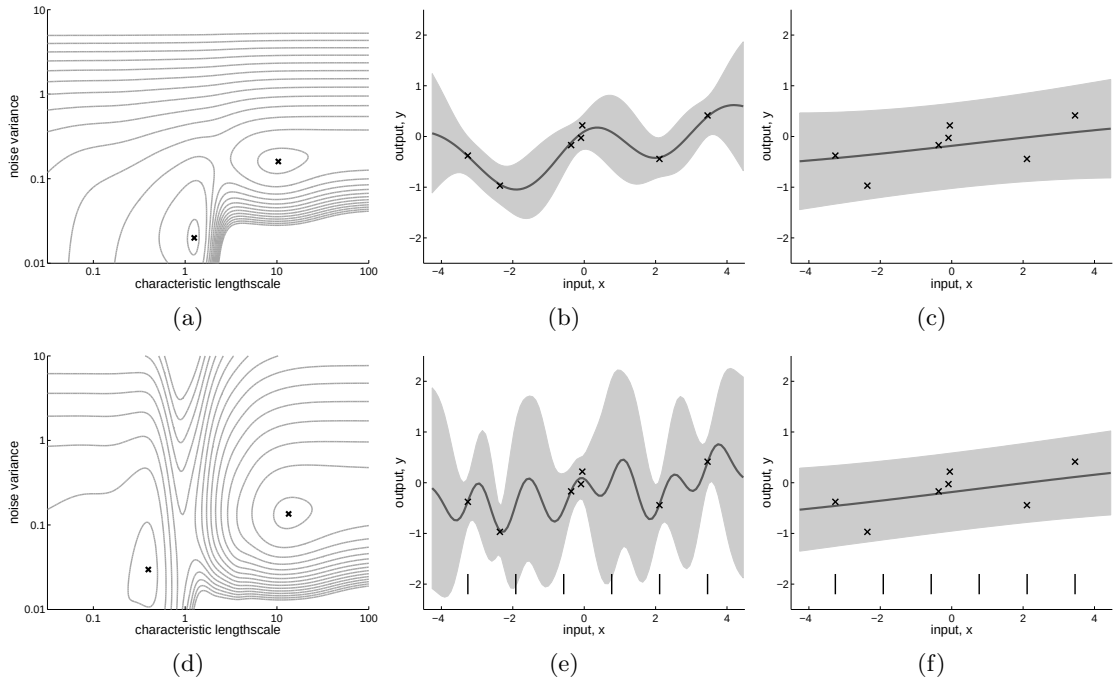

Figure 2: Contour plot of the log marginal likelihood without monotonicity information (a), and the corresponding solutions (b) and (c) at the modes. Subfigure (d) shows contour plot of the marginal likelihood with monotonicity information, and Subfigures (e) and (f) illustrate the corresponding solutions at the modes. The locations of virtual observations are shown with short vertical lines in Subfigures (e) and (f).

Table 1: Root-mean-square errors for the artificial examples.

function	root-mean-square error			
	$N = 100$		$N = 200$	
	GP	GP (monot.)	GP	GP (monot.)
step	0.135	0.176	0.109	0.167
linear	0.091	0.068	0.053	0.041
exponential	0.074	0.068	0.054	0.050
logistic	0.078	0.077	0.060	0.062

step likelihood, the result may not be well approximated with a Gaussian distribution, and thus derivative information presented by virtual observations can be slightly biased away from zero. On the other hand, the GP without monotonicity assumption estimates the step function with a shorter lengthscale producing a better fit but with wiggling behaviour. For linear and exponential functions the GP with monotonicity assumption gives better estimates, as monotonicity information favours smoother solutions and prevents the estimated functions from wiggling. In the case of 200 observations, the differences between the estimates for the two models were smaller with linear and exponential functions, as the possibility of overfit decreases. For logistic function both models gave similar results.

4.3 MODELLING RISK OF INSTITUTIONALISATION

In this section we report the results of assessing the institutionalisation risk of users of communal elderly care services. The risk of institutionalisation was modelled using data produced from health care registers, and the aim was to study whether a patient becomes institutionalised or not during the next three months. The actual study population consisted of patients over 65 years in the city of Vantaa during 2001–2004. In this study the following seven variables were used as predictors: age, whether the patient had nursing home periods, whether the patient had activities of daily living (ADL) evaluation, maximum memory problem score, maximum behavioral symptoms score, maximum number of daily home care visits, and number of days in hospital. Since only a small number of institutionalisation events were available in the whole data set, the training data was balanced such that approximately half of the patients institutionalised. The training data set consisted of 1222 observations.

Classification was done using a Gaussian process binary classification model with the probit likelihood function, and the squared exponential covariance function (with an individual lengthscale parameter for each input variable). We modelled the risk of institutionalisation with a GP where no information about monotonicity with respect to any of the covariates was assumed. This model was compared to a GP model where monotonicity information was added such that the institutionalisation risk was assumed to increase as a function of age. Virtual observations were placed at the unique locations of the input training data points.

To test the predictive abilities of these two GP models, receiver operating characteristic (ROC) curves were computed for younger and older (the oldest third) age groups using an independent test data of 20000 observation periods. The predictive performances of the models were similar for the younger age group but the GP model with monotonicity information gave better predictions for the older age group (Figure 3). As age increases, the data becomes more scarce and monotonicity assumption more useful.

We also studied the effect of monotonicity information in the model by comparing the predicted risks of institutionalisation as a function of age and different daily home care levels. The predictions for a low-risk subgroup are shown in Figure 4. The GP model without monotonicity information gives a slight decrease for the institutionalisation risk for patients over 80 (Subfigure (a)), whereas the GP model with monotonicity information gives smoother results (Subfigure (b)), suggesting more realistic estimates.

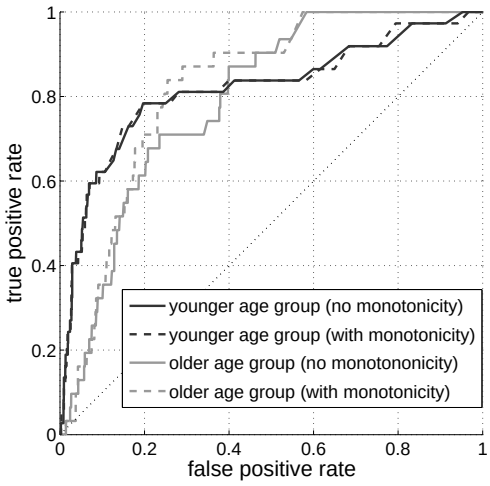

Figure 3: ROC curves for the probability of institutionalisation of elderly.

5 CONCLUSION

We have proposed a method for introducing monotonicity information to a nonparametric Gaussian process model. The monotonicity information is set us-

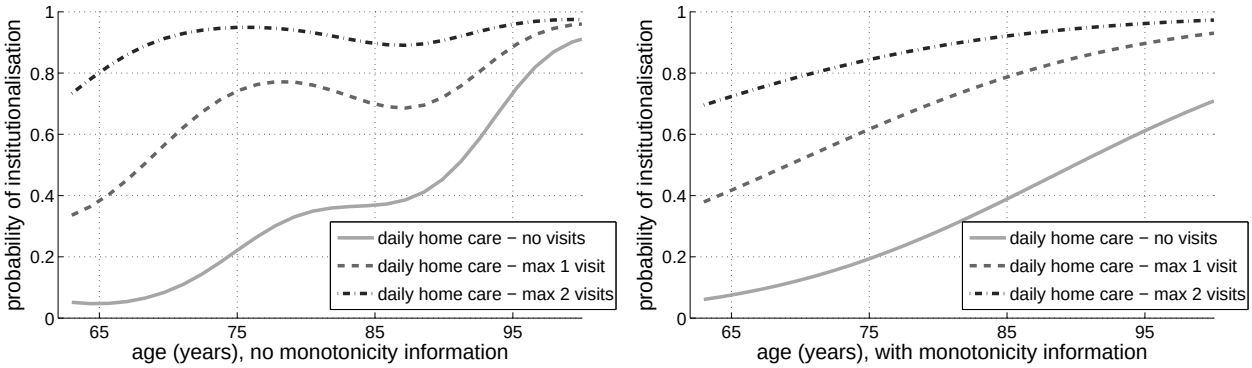

Figure 4: Simulated estimates for the probabilities of institutionalisation of elderly as a function of age and daily home care levels. The estimates using a Gaussian process model are shown in Subfigure (a), and the estimates using a Gaussian process with monotonicity information in Subfigure (b).

ing virtual derivative observations concerning the behaviour of the target function in the desired locations of input space. In the method a Gaussian approximation is found for the virtual derivative observations using the EP algorithm, and the virtual observations are used in the GP model in addition to the real observations.

In the cases where the target function is monotonic, a solution that is less prone to overfitting, and therefore better, can be achieved using monotonicity information. This is emphasized in the cases where there is only a small number of observations available. When the target function has flat areas with sharp steps, the virtual derivative observations can lead to a worse performance caused by a bias away from zero due to Gaussian approximation of the truncated derivative distribution. Therefore virtual derivative observations implying monotonicity are more useful in the cases when the target function is smooth. Further, if the distance between the virtual derivative observations is too large with respect to the estimated characteristic lengthscale, the solution can become non-monotonic. However, by placing and adding the virtual points iteratively, a monotonic solution can be made more likely.

Acknowledgements

The authors thank Matti Mäkelä and Elina Parviainen for providing the institutionalisation data, and Academy of Finland for funding.

References

Lampinen, J. and Selonen, A. (1997). Using background knowledge in multilayer perceptron learning. In *Proc. of The 10th Scandinavian Conference on Image Analysis, volume 2*, pages 545–549.

MacKay, D. J. C. (1998). Introduction to Gaussian processes. In Bishop, C. M., editor, *Neural Networks and Machine Learning*, pages 133–166. Springer-Verlag.

Minka, T. (2001). Expectation Propagation for approximative Bayesian inference. In *Proceedings of the 17th Annual Conference on Uncertainty in Artificial Intelligence (UAI-01)*, pages 362–369.

Neal, R. M. (1999). Regression and classification using Gaussian process priors (with discussion). In Bernardo, J. M., Berger, J. O., Dawid, A. P., and Smith, A. F. M., editors, *Bayesian Statistics 6*, pages 475–501. Oxford University Press.

O'Hagan, A. (1978). Curve fitting and optimal design for prediction. *Journal of the Royal Statistical Society (Series B)*, 40(1):1–42.

Rasmussen, C. E. (2003). Gaussian processes to speed up Hybrid Monte Carlo for expensive Bayesian integrals. In *Bayesian Statistics*, volume 7, pages 651–659. Oxford University Press.

Rasmussen, C. E. and Williams, C. K. I. (2006). *Gaussian Processes for Machine Learning*. The MIT Press.

Shively, T. S., Sager, T. W., and Walker, S. G. (2009). A Bayesian approach to non-parametric monotone function estimation. *Journal of the Royal Statistical Society (Series B)*, 71(1):159–175.

Sill, J. and Abu-Mostafa, Y. (1997). Monotonicity hints. In *Advances in Neural Information Processing Systems 9*, pages 634–640. MIT Press.

Solak, E., Murray-Smith, R., Leithead, W. E., Leith, D. J., and Rasmussen, C. E. (2003). Derivative observations in Gaussian process models of dynamic systems. In *Advances in Neural Information Processing Systems 15*, pages 1033–1040. MIT Press.