# Healing the Relevance Vector Machine through Augmentation

**Carl Edward Rasmussen** CARL@TUEBINGEN.MPG.DE
Max Planck Institute for Biological Cybernetics, 72076 Tübingen, Germany

**Joaquin Quiñonero-Candela** JQC@TUEBINGEN.MPG.DE
Friedrich Miescher Laboratory, Max Planck Society, 72076 Tübingen, Germany

## Abstract

The Relevance Vector Machine (RVM) is a sparse approximate Bayesian kernel method. It provides full predictive distributions for test cases. However, the predictive uncertainties have the unintuitive property, that *they get smaller the further you move away from the training cases*. We give a thorough analysis. Inspired by the analogy to non-degenerate Gaussian Processes, we suggest augmentation to solve the problem. The purpose of the resulting model, RVM*, is primarily to corroborate the theoretical and experimental analysis. Although RVM* could be used in practical applications, it is no longer a truly sparse model. Experiments show that sparsity comes at the expense of worse predictive distributions.

Bayesian inference based on Gaussian Processes (GPs) has become widespread in the machine learning community. However, their naïve applicability is marred by computational constraints. A number of recent publications have addressed this issue by means of sparse approximations, although ideologically sparseness is at variance with Bayesian principles[1]. In this paper we view sparsity purely as a way to achieve computational convenience and not as under other non-Bayesian paradigms where sparseness itself is seen as a means to ensure good generalization.

Sparsity is achieved by Csató (2002), Csató and Opper (2002), Seeger (2003), and Lawrence et al. (2003), by minimizing KL divergences between the approximated and true posterior, by Smola and Schölkopf (2000) and Smola and Bartlett (2001) by making low rank approximations to the posterior. In (Gibbs & MacKay, 1997) and in (Williams & Seeger, 2001) sparseness arises from matrix approximations, and in (Tresp, 2000) from neglecting correlations. The use of subsets of regressors has also been suggested by Wahba et al. (1999).

The Relevance Vector Machine (RVM) introduced by Tipping (2001) produces sparse solutions using an improper hierarchical prior and optimizing over hyperparameters. The RVM is exactly equivalent to a Gaussian Process, where the RVM hyperparameters are parameters of the GP covariance function (more on this in the discussion section). However, the covariance function of the RVM seen as a GP is *degenerate*: its rank is at most equal to the number of relevance vectors of the RVM. As a consequence, for localized basis functions, the RVM produces predictive distributions with properties opposite to what would be desirable. Indeed, the RVM is more certain about its predictions the further one moves away from the data it has been trained on. One would wish the opposite behaviour, as is the case with non-degenerate GPs, where the uncertainty of the predictions is minimal for test points in the regions of the input space where (training) data has been seen. For non-localized basis functions, the same undesired effect persists, although the intuition may be less clear, see the discussion.

In the next section we briefly review the RVM and explore the properties of the predictive distribution in some detail and through an illustrative example. Next, we propose a simple modification to the RVM to reverse the behaviour and remedy the problem. In section 3 we demonstrate the improvements on two problems, and compare to non-sparse GPs. A comparison to the many other sparse approximations is outside the scope of this paper, our focus is on enhancing the

[1] In the Bayesian paradigm one averages over all possible explanations of the data. Sparseness corresponds to making "hard" choices about these possible explanations.

Appearing in *Proceedings of the 22$^{nd}$ International Conference on Machine Learning*, Bonn, Germany, 2005. 

understanding of the properties of the RVM.

## 1. Classical Relevance Vector Machines

The RVM, introduced by Tipping (2001), is a sparse linear model. Given a set of training inputs $\{\boldsymbol{x}_i | i = 1, \ldots, N\} \subset \mathbb{R}^D$ organized as rows in matrix $X$, the model outputs are a linear combination of the responses of a set of basis functions $\{\phi_j(\boldsymbol{x}) | j = 1, \ldots, M\} \subset [\mathbb{R}^D \to \mathbb{R}]$:

$$f(\boldsymbol{x}_i) \;=\; \sum_{j=1}^{M} \phi_j(\boldsymbol{x}_i)\, w_j = \phi(\boldsymbol{x}_i)\, \mathbf{w}, \qquad \mathbf{f} \;=\; \phi\, \mathbf{w}, \quad (1)$$

where $\mathbf{f} = [f(\boldsymbol{x}_1), \ldots, f(\boldsymbol{x}_N)]^\top$ are the function outputs, $\mathbf{w} = [w_1, \ldots, w_M]^\top$ are the weights and $\phi_j(\boldsymbol{x}_i)$ is the response of the $j$-th basis function to input $x_i$. We adopt the following shorthand: $\phi(\boldsymbol{x}_i) = [\phi_1(\boldsymbol{x}_i), \ldots, \phi_M(\boldsymbol{x}_i)]$ is a row vector containing the response of all basis functions to input $\boldsymbol{x}_i$, $\phi_j = [\phi_j(\boldsymbol{x}_1), \ldots, \phi_j(\boldsymbol{x}_N)]^\top$ is a column vector containing the response of basis function $\phi_j(\boldsymbol{x})$ to all training inputs and $\phi$ is an $N \times M$ matrix whose $j$-th column is vector $\phi_j$ and whose $i$-th row is vector $\phi(\boldsymbol{x}_i)$.

The prior on the weights is independent Gaussian, $p(\mathbf{w}|A) \sim \mathcal{N}(0, A^{-1})$, with separate precision hyperparameters $A = \mathrm{diag}[\alpha_1, \ldots, \alpha_M]$. The output noise is assumed to be zero mean iid. Gaussian of variance $\sigma^2$, such that $p(\mathbf{y}|X, \mathbf{w}, \sigma^2) \sim \mathcal{N}(\mathbf{f}, \sigma^2 I)$, where $\mathbf{y} = [y_1, \ldots, y_N]^\top$ are the training targets.

Learning is achieved by maximizing[2] the marginal likelihood:

$$p(\mathbf{y}|X, A, \sigma^2) \;=\; \int p(\mathbf{y}|X, \mathbf{w}, \sigma^2) p(\mathbf{w}|A) \mathrm{d}\mathbf{w}, \quad (2)$$

(penalized by an inconsequential improper prior uniform in $\log(\alpha)$) wrt. the $\log(\alpha)$ parameters in $A$ and the noise variance $\sigma^2$. Sparsity results when a number of the $\alpha$'s go to infinity, thus effectively removing the corresponding basis functions; the surviving basis functions are called the *relevance vectors*. See (Tipping, 2001) for the details. It has been shown by Faul and Tipping (2002) that local maxima of the marginal likelihood occur when some of the $\alpha$ tend to infinity. Integrating over $\alpha$, the conditional Gaussian weight prior given $\alpha$ and the uniform top level prior in $\log(\alpha)$ imply an improper weight prior of the form $p(w_i) \propto 1/|w_i|$. Wipf et al. (2004) have shown that for any fixed $\boldsymbol{\alpha}$ the RVM Gaussian conditional priors on the weighs constitute a variational lower bound to the full-blown Bayesian treatment. The $\alpha$'s are then interpreted as variational parameters, and it can be shown that setting them to infinity maximizes the variational approximation to the marginal likelihood.

It is customary (but by no means necessary) to use localized squared exponential radial basis functions, centered on the training points, of the form $\phi_j(\boldsymbol{x}_i) = \exp(-\frac{1}{2}\sum_{d=1}^{D}(X_{id} - X_{jd})^2/\lambda_d^2)$. The metric parameters $\lambda_d$ can also be inferred (Tipping, 2001, appendix C). This procedure requires interleaved updates of the metric parameters, $\lambda_d$, the hyperparameters $\boldsymbol{\alpha}$ and the noise level $\sigma^2$. Note that unfortunately the final solution depends both on the initial parameter values and on the exact details of the interleaving of the different updates.

### A Peculiar Prior over Functions

The prior distribution over functions implied by the model is a weighted linear combination of basis functions, the weights distributed according to their prior. For localized basis functions, the prior thus excludes variation outside the range of the basis functions. When the basis functions are centered on the training data, this means that a priori no signal is expected far away from the training points, as was also pointed out by Tipping (2001). This seems like an unintuitive property, that the function is expected to be flat in regions where we happen not to see any data at training time.

The posterior distribution over the weights is proportional to the product of likelihood and prior. It is Gaussian, $p(\mathbf{w}|\mathbf{y}, X, A, \sigma^2) \sim \mathcal{N}(\boldsymbol{\mu}, \Sigma)$, with mean and covariance given by:

$$\begin{aligned} \boldsymbol{\mu} &= \sigma^{-2} \Sigma\, \phi^\top \mathbf{y}, \\ \Sigma &= [\sigma^{-2} \phi^\top \phi + A]^{-1}. \end{aligned} \quad (3)$$

The predictive distribution at a new test input $\boldsymbol{x}_*$ is obtained by integrating the weights from the model (1) over the posterior. The predictive distribution is also Gaussian:

$$\begin{aligned} &p(y_*|\boldsymbol{x}_*, X, \mathbf{y}, A, \sigma^2) \\ &\qquad = \int p(y_*|\boldsymbol{x}_*, X, \mathbf{w}, \sigma^2) p(\mathbf{w}|\mathbf{y}, X, A, \sigma^2) \mathrm{d}\mathbf{w} \\ &\qquad \sim \mathcal{N}(m(\boldsymbol{x}_*), v(\boldsymbol{x}_*)), \end{aligned}$$

with mean and variance given by:

$$\begin{aligned} m(\boldsymbol{x}_*) \;&= \phi(\boldsymbol{x}_*)\, \boldsymbol{\mu}, \\ v(\boldsymbol{x}_*) \;&=\; \sigma^2 + \phi(\boldsymbol{x}_*)\, \Sigma\, \phi(\boldsymbol{x}_*)^\top. \end{aligned} \quad (4)$$

For localized basis functions, if the input $\boldsymbol{x}_*$ lies far away from the centers of all relevance vectors, the response of the basis functions $\phi_j(\boldsymbol{x}_*)$ becomes small:

[2] This can be done either by direct optimization (eg. conjugate gradients) or perhaps faster by means of an EM-like algorithm (Tipping, 2001, appendix A.2).

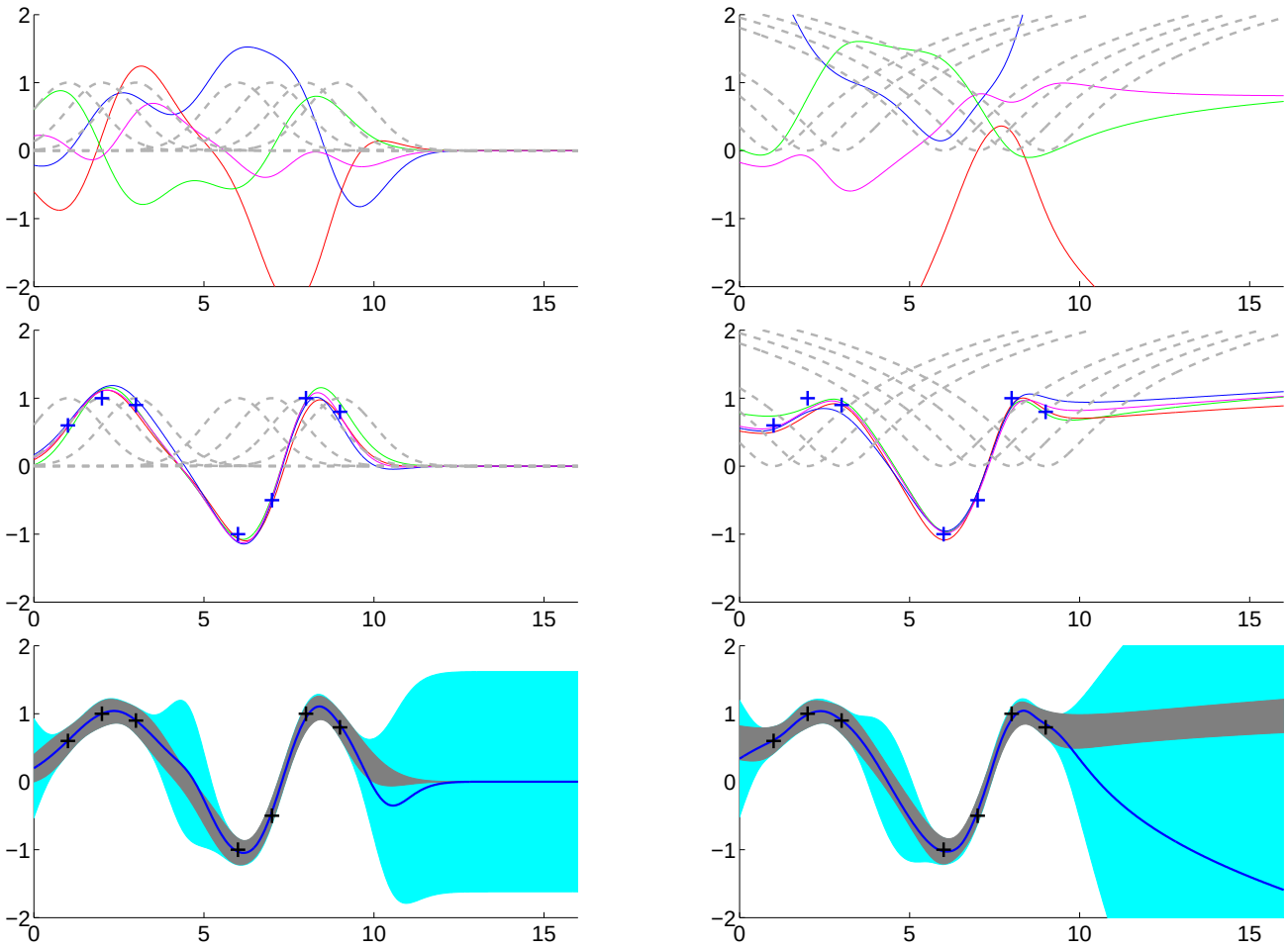

*Figure 1.* Properties of prior and posterior of RVMs with all $\alpha_i = 1$ and $\sigma^2 = 0.1$. In the *left* column the basis functions are localized squared exponentials (non-normalized Gaussians). *Top:* functions drawn from the RVM prior. The dashed lines represent the basis functions of the model. *Middle:* functions drawn from the RVM posterior after observing the training targets (crosses) *Bottom:* The dark gray stripes are the predictive distributions for the classical RVM: the mean, not represented, is in the middle of the stripe, and the width of the stripe is equal $\pm 2$ predictive standard deviations (95% confidence interval). The solid (blue) line is the mean of the predictive distribution of the RVM*, and the light (cyan) stripe has width $\pm 2$ RVM* predictive standard deviations. Only for RVM* does the predictive uncertainty grow when moving away from the training data. *Right* column: same as left with non-localized basis functions of the form $\phi_j(\boldsymbol{x}_i) = \log(1 + \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2)$. Notice that $\phi_i(\boldsymbol{x}_i) = 0$: augmentation cannot increase the variance by using the new basis function itself.

the predictive mean goes to zero and the predictive variance reduces to the noise level, eq. (4). Under the prior there is not much signal far away from the centers of the relevance vectors; this property persists in the posterior. In other words, the model uncertainty is maximal in the neighbourhood of the centers of the relevance vectors, and goes to zero as one moves far away from them, as illustrated in the left column of figure 1. The figure assumes all basis functions are relevance vectors. In the sparse case the RVM uncertainty is even smaller.

As a probabilistic model, the RVM with localized basis functions thus produces unreasonable predictive uncertainties, with a behaviour *opposite to what would seem desirable*.

## 2. Augmentation: RVM*

Consider having trained an RVM and facing the task of predicting at a new unseen test point. To solve the problem, that there might be no possibility of variation for outputs corresponding to inputs far from the centers of the relevance vectors, we propose to augment the model by an additional basis function centered at

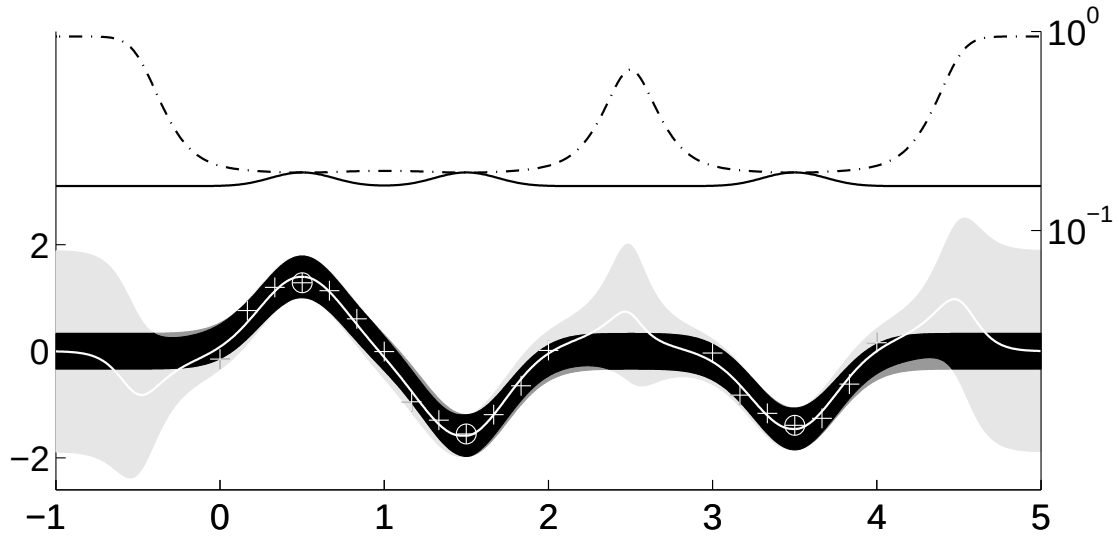


*Figure 2.* Behaviour of the predictive distributions. *Bottom* (with left y-axis): the white crosses represent the training data, the circled ones being the 3 cases whose input is the center of the relevance vectors obtained from training. The black region is the 95% confidence interval for the predictions of a standard RVM, and the gray region that for the RVM* (for the latter, the white line is the mean prediction). *Top* (with right y-axis): The solid line represents the predictive standard deviation of the RVM, and the dot-slashed one that of the RVM*. Note that the predictive variance *decreases* when moving away from the relevance vectors in the RVM, but *increases* for the RVM*.

the test input. The training stage, that is the setting of the $\alpha$'s, remains *unchanged*, the new basis function being introduced only at test time and for a specific test input. This is the idea behind the modification of the RVM that we propose: the RVM*.

For each test point $\boldsymbol{x}_*$, we modify the model obtained from training by introducing one new basis function centered on $\boldsymbol{x}_*$, and its associated weight with prior distribution $p(w^*) \sim \mathcal{N}(0, \alpha_*^{-1})$. The joint augmented Gaussian posterior distribution of the weights has now mean and covariance given by:

$$\begin{aligned} \boldsymbol{\mu}_* &= \sigma^{-2}\Sigma_* \begin{bmatrix} \phi^\top \\ \phi_*^\top \end{bmatrix} \mathbf{y}, \\ \Sigma_* &= \begin{bmatrix} \Sigma^{-1} & \sigma^{-2}\phi^\top \phi_* \\ \sigma^{-2}\phi_*^\top \phi & \alpha_* + \sigma^{-2}\phi_*^\top \phi_* \end{bmatrix}^{-1}, \end{aligned} \tag{5}$$

where $\boldsymbol{\mu}$ and $\Sigma$ are the posterior weight mean and covariance of the classical RVM, given in eq. (3), and $\phi_*$ is the newly introduced basis function evaluated at all training inputs.

The Gaussian predictive distribution of the augmented model at $\boldsymbol{x}_*$ has mean and variance which can be written as the terms from the classical RVM, eq. (4), plus correction terms:

$$\begin{aligned} m_*(\boldsymbol{x}_*) &= m(\boldsymbol{x}_*) + \frac{e_*\, q_*}{\alpha_* + s_*}, \\ v_*(\boldsymbol{x}_*) &= v(\boldsymbol{x}_*) + \frac{e_*^2}{\alpha_* + s_*}, \end{aligned} \tag{6}$$

as illustrated in figure 1 bottom left, where:

$$\begin{aligned} q_* &= \phi_*^\top(\mathbf{y} - \phi\boldsymbol{\mu})/\sigma^2, \\ e_* &= \phi_*(\boldsymbol{x}_*) - \sigma^{-2}\phi(\boldsymbol{x}_*)\Sigma\phi^\top\phi_*, \\ s_* &= \phi_*^\top(\sigma^2 I + \phi A^{-1}\phi^\top)^{-1}\phi_*. \end{aligned} \tag{7}$$

We have adopted the notation of Faul and Tipping (2002): $q_*$ is the projection of the vector of residuals onto the new basis function: it is therefore a 'quality' factor, that indicates how much the training error can be reduced by making use of the new basis function. $s_*$ is a 'sparsity' factor that indicates how much the new basis function is redundant for predicting the training data given the existing relevance vectors. $e_*$ is an 'error' term that is smaller the better the existing model can mimic the new basis function at $x_*$.

Note, that the predictive variance of RVM* in eq. (6) is guaranteed not to be smaller than for the RVM. Note also, that the predictive mean of the RVM* is modified as a result of the additional modelling flexibility, given by the new basis function. This new basis function is weighted according to how much it helps model the

part of the training data that was not well modelled by the classic RVM, whose sparseness may lead to underfitting, see the discussion section. Figure 2 illustrates this effect in the sparse RVM.

When introducing an additional basis function at test time, we also get an additional weight $w_*$ (which is integrated out when making predictions) and an extra prior precision parameter $\alpha_*$. How do we set $\alpha_*$? One naïve approach would be to take advantage of the work on incremental training done by Faul and Tipping (2002), where it is shown that the value of $\alpha_*$ that maximizes the marginal likelihood, given all the other $\alpha$'s (in our case obtained from training) is given by:

$$\alpha_* = \frac{s_*^2}{q_*^2 - s_*}, \quad \text{if } q_*^2 > s_*, \quad \alpha_* = \infty, \quad \text{otherwise.}$$

Unfortunately, this strategy poses the risk of deletion of the new basis function (when the new basis function doesn't help significantly with modelling the data, which is typically the case when when $\boldsymbol{x}_*$ lies far from all the training inputs). Thus the unjustifiably small error bars of RVM would persist.

In our setting learning $\alpha_*$ by maximizing the evidence makes little sense, since it contravenes the nature of our approach. We do want to impose an a priori assumption on the variation of the function. When far away from the relevance vectors, $\alpha_*^{-1}$ is the a priori variance of the function value. We find it natural to make $\alpha_*^{-1}$ equal to the empirical variance of the observed target values, corresponding to the prior assumption that the function may vary everywhere. One could conceive other reasonable ways of setting $\alpha_*$ as long as it remains finite.

Training is identical for the RVM and for the RVM*, so it has the same computational complexity for both. For predicting, the RVM needs only to retain $\boldsymbol{\mu}$ and $\Sigma$ from training, and the complexity is $\mathcal{O}(M)$ for computing the predictive mean and $\mathcal{O}(M^2)$ for the predictive variance. The RVM* needs to retain the whole training set in addition to $\boldsymbol{\mu}$ and $\Sigma$. The computational complexity is $\mathcal{O}(MN)$ both for computing the predictive mean and the predictive variance. The dependence on the full training set size $N$ is caused by the additional weight needing access to all targets[3] for updating the posterior. The RVM* is thus not really a sparse method anymore, and it is not necessarily an interesting algorithm in practice.

[3]One could get rid of the dependence on $N$ by re-fitting only using the targets associated with the relevance vectors; this leads to too large predictive variances, since the training set may have contained data close to the test input, which hadn't been designated as relevance vectors.

## 3. Experiments

We compare the classic RVM, the RVM* and a Gaussian process (GP) with squared exponential covariance function on two datasets: the Boston house-price dataset, (Harrison & Rubinfeld, 1978), with 13-dimensional inputs, and the KIN40K (robot arm) dataset[4], with 8-dimensional inputs. The KIN40K dataset represents the forward dynamics of an 8 link all-revolve robot arm.

We use a 10 fold cross-validation setup for testing on both datasets. For Boston house-price we use disjoint test sets of 50/51 cases, and training sets of the remaining 455/456 cases. For the robot arm we use disjoint test and training sets both of 2000 cases. For all models we learn individual length-scales for each input dimension, and optimize by maximizing the marginal likelihood, (Tipping, 2001, appendix C) and (Williams & Rasmussen, 1996). For each partition and model we compute the squared error loss, the absolute error loss and the negative log test density loss. In addition to the average values of the different losses, we compute the statistical significance of the difference in performance of each pair of models for each loss, and provide the p-value obtained from a (two-sided) paired t-test[5] on the test set averages.

The results for the Boston house-price example in table 1 show that the RVM* produces significantly better predictive distributions than the classic RVM. Whereas the losses which only depend on the predictive mean (squared and absolute) are not statistically significantly different between RVM an RVM*, the negative log test density loss is significantly smaller for RVM*, confirming that the predictive uncertainties are much better. The RVM models have a final average number of relevance vectors of $27 \pm 14$ (mean $\pm$ std. dev.) showing a high degree of sparsity[6] and quite some variability. The results for the KIN40K robot arm example in table 2 show a similar picture. For this (larger) data set, the difference between RVM and RVM* is statistically significant even for the losses only depending on the mean predictions. The final numbers of relevance vectors were $252 \pm 11$. We also compare to a non-degenerate (see section 4) Gaussian process. The GP has a significantly superior perfor-

[4]From the DELVE archive http://www.cs.toronto.edu/delve.

[5]For the Boston house-price dataset, due to dependencies (overlap) between the training sets, assumptions of independence needed for the t-test are compromised, but this is probably of minor effect.

[6]Note, that the degree of sparsity obtained depends on the (squared exponential) basis function widths; here the widths were optimized using the marginal likelihood.

*Table 1.* Results for the Boston house-price experiments for RVM, RVM* and GP. The upper sub-table indicates the average value of the losses for three loss functions. In the lower sub-table, the values in the cells are the p-values that indicate the significance with which the model in the corresponding column beats the model in the corresponding row.

| | Squared error loss | | | Absolute error loss | | | - log test density loss | | |
|---|---|---|---|---|---|---|---|---|---|
| | RVM | RVM* | GP | RVM | RVM* | GP | RVM | RVM* | GP |
| Loss: | 0.138 | 0.135 | 0.092 | 0.259 | 0.253 | 0.209 | 0.469 | 0.408 | 0.219 |
| RVM | · | not sig. | $< 0.01$ | · | 0.07 | $< 0.01$ | · | $< 0.01$ | $< 0.01$ |
| RVM* | | · | 0.02 | | · | $< 0.01$ | | · | $< 0.01$ |
| GP | | | · | | | · | | | · |

mance under all losses considered. Note also, that the difference between RVM and GPs is much larger than that between RVM and RVM*. This may indicate that sparsity in regression models may come at a significant cost in accuracy. To our knowledge, RVMs and GPs have not been compared previously experimentally in an extensive manner.

## 4. Discussion

The RVM is equivalent to a GP (Tipping, 2001, section 5.2) with covariance function given by:

$$k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sum_{k=1}^{M} \frac{1}{\alpha_k} \phi_k(\boldsymbol{x}_i)\, \phi_k(\boldsymbol{x}_j). \tag{8}$$

This covariance function is guaranteed to be *degenerate* since $M$ is finite, and even worse typically small for the RVM. The distribution over (noise free) functions is singular. This limits the range of functions that can be implemented by the model. The RVM* introduced in this paper is a GP with an augmented covariance function:

$$k_*(\boldsymbol{x}_i, \boldsymbol{x}_j) = k(\boldsymbol{x}_i, \boldsymbol{x}_j) + \frac{1}{\alpha_*} \phi_*(\boldsymbol{x}_i)\, \phi_*(\boldsymbol{x}_j), \tag{9}$$

which ensures prior variability at the test location (provided $\phi_*(\boldsymbol{x}_*) \neq 0$), that survives into the posterior if the data doesn't have a strong opinion in that region.

It is interesting to note that a GP with squared exponential covariance function coincides exactly with an RVM infinitely augmented, at all points in the input space. Following MacKay (1997), consider for simplicity a one-dimensional input space, with squared exponential basis functions $\phi_c(x_i) = \exp(-\frac{1}{2}(x_i - c)^2/\lambda^2)$, where $c$ is a given centre in input space and use the RVM weight prior in isotropic form $A = \alpha \mathbf{I}$. We want to make the number of basis functions $M$ go to infinity, and assume that the centres are uniformly spaced. To make sure that the integral converges, we set variance of the prior over the weights to $\alpha^{-1} = sM$, for some constant $s$. The covariance function becomes:

$$\begin{aligned} k(x_i, x_j) &= s \int_{c_{\min}}^{c_{\max}} \phi_c(x_i)\, \phi_c(x_j)\, \mathrm{d}\, c, \\ &= s \int_{c_{\min}}^{c_{\max}} \exp\left[-\frac{(x_i - c)^2}{2\lambda^2}\right] \exp\left[-\frac{(x_j - c)^2}{2\lambda^2}\right] \mathrm{d}\, c. \end{aligned}$$

Letting the limits of the integral go to plus and minus infinity, we obtain the integral of the product of two (non-normalized) Gaussians which evaluates to:

$$k(x_i, x_j) = s\sqrt{\pi\lambda^2} \exp\left[-\frac{(x_i - x_j)^2}{4\lambda^2}\right]. \tag{10}$$

Thus, we recover the squared exponential covariance GP as being equivalent to an infinite RVM. The infinite RVM becomes tractable when viewed as a GP, but of course it is not clear how to treat the infinitely many hyperparameters, or how to introduce sparsification from this standpoint.

It may be surprising that the experiments show that the performance using loss functions which depend only on the predictive means was improved for the RVM* (although sometimes the difference was not statistically significant). The reason for this is that the extra added basis function, which is fit to the training data, adds flexibility to the model. Since this extra flexibility turns out to improve performance, this shows that the classical RVM under-fits the data, ie. the models have become too sparse. Indeed the performance of the full non-degenerate GP is much better still.

### Non-localized Basis Functions

The exposition has so far concentrated on localized basis functions. Other basis functions could be (and have been) considered, although their use with multivariate inputs restrict them in practice mostly to radial form (so that the models remain linear in the parameters, which is essential for analytic treatment – this

*Table 2.* Results for the Robot Arm data; the table is read analogously to table 1.

| | Squared error loss | | | Absolute error loss | | | - log test density loss | | |
|---|---|---|---|---|---|---|---|---|---|
| | RVM | RVM* | GP | RVM | RVM* | GP | RVM | RVM* | GP |
| Loss: | 0.0043 | 0.0040 | 0.0024 | 0.0482 | 0.0467 | 0.0334 | -1.2162 | -1.3295 | -1.7446 |
| RVM | · | $< 0.01$ | $< 0.01$ | · | $< 0.01$ | $< 0.01$ | · | $< 0.01$ | $< 0.01$ |
| RVM* | | · | $< 0.01$ | | · | $< 0.01$ | | · | $< 0.01$ |
| GP | | | · | | | · | | | · |

precludes e.g. sigmoidal functions as used in neural networks).

One may be tempted to think that an RVM with non-localized basis functions would automatically guarantee larger predictive uncertainties as one moves away from the training data. As illustrated in figure 1 bottom right, this is not the case. In the figure we use basis functions of the form $\phi_j(\boldsymbol{x}_i) = \log(1 + \|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2)$ which grow unboundedly. Note that the value of the basis function at its center is zero, so one could have reservations about the effectiveness of augmentation with an additional basis function centered at the test input. However, as seen clearly in the bottom right of 1 the augmentation leads also in this case to the desired behaviour. In this case the additional uncertainty comes exclusively from the interaction between the new basis function with the existing ones (as the new basis function contributes zero at the test input).

No matter the kind of basis functions used, sparseness in the RVM leads to an over-constrained system of equations, corresponding to a tight posterior covariance. As a thought experiment, consider a GP with degenerate covariance function of rank $M$. Consider iteratively generating random function values from the process, while conditioning on the previously drawn samples. While the first $M$ draws have considerable flexibility, subsequent draws are essentially deterministic, since "freedom" of the process has been pinned down by the first $M$ samples. Differently put, one can only sample $M$ linearly independent functions from the prior. Prediction in a sparse RVM parallels exactly this thought experiment.

Although the present paper discusses only regression problems, it should be clear that analogous effects may play a rôle for classification, although in a more subtle way: underestimated uncertainties may lead to too confident predictive class probabilities, although since the mean and variance of the Bernoulli distribution are linked, moving away from the training cases may cause the probability to tend toward $\frac{1}{2}$, which automatically implies higher uncertainty.

## 5. Conclusions

The RVM has become a popular tool because it is a simple tractable probabilistic model. As we have shown, if one is interested in the predictive variance, the RVM should not be used. Even if one is interested only in predictive means, the sparsity mechanism of the RVM seems to come at the expense of accuracy. The proposed RVM* goes some way towards fixing this problem at an increased computational cost. We do not propose it as an alternative to the RVM in practice, but rather as a tool for our argumentation.

Although outside the scope of this paper, it is an important future task to experimentally compare the computation vs. accuracy tradeoff between different methods for sparsifying GPs. Some recent papers do attempt to assess these tradeoffs, however, regrettably, the performance measures often neglect the probabilistic nature of the predictions and focus exclusively on mean predictions.

A key distinction highlighted both theoretically and through experiments in this paper, is the difference between finite-dimensional models and proper process models – in GPs exemplified by degenerate and non-degenerate covariance functions. For probabilistic models, where faithful representation of uncertainties play a central rôle, we emphasize that non-degenerate models are probably best suited. It remains an important future goal to reconcile the sparsity requirement, needed for computational tractability, with proper non-degenerate process models. As we have shown, the RVM does not achieve this goal.

**Acknowledgements**: This work was supported by the EU Framework 6 PASCAL Network of Excellence, Contract Number 506778, and by the German Research Council (DFG) through grant RA 1030/1. JQC was partially supported by the Max Planck Institute for Biological Cybernetics.

## References


Csató, L. (2002). *Gaussian processes – iterative sparse approximation*. Doctoral dissertation, Aston University, Birmingham, United Kingdom.

Csató, L., & Opper, M. (2002). Sparse online gaussian processes. *Neural Computation*, *14*, 641–669.

Faul, A. C., & Tipping, M. E. (2002). Analysis of sparse Bayesian learning. *Advances in Neural Information Processing Systems 14* (pp. 383–389). Cambridge, Massachussetts: MIT Press.

Gibbs, M., & MacKay, D. J. C. (1997). *Efficient implementation of Gaussian processes* (Technical Report). Cavendish Laboratory, Cambridge University, Cambridge, United Kingdom.

Harrison, D., & Rubinfeld, D. (1978). Hedonic prices and the demand for clean air. *Journal of Environmental Economics & Management*, *5*, 81–102. Data available from http://lib.stat.cmu.edu/datasets/boston.

Lawrence, N., Seeger, M., & Herbrich, R. (2003). Fast sparse Gaussian process methods: The Informative Vector Machine. *Neural Information Processing Systems 15* (pp. 609–616). Cambridge, Massachussetts: MIT Press.

MacKay, D. J. C. (1997). *Gaussian Processes: A replacement for supervised Neural Networks?* (Technical Report). Cavendish Laboratory, Cambridge University, Cambridge, United Kingdom. Lecture notes for a tutorial at NIPS 1997.

Seeger, M. (2003). *Bayesian Gaussian process models: PAC-Bayesian generalisation error bounds and sparse approximations*. Doctoral dissertation, University of Edinburgh, Edinburgh, Scotland.

Smola, A. J., & Bartlett, P. L. (2001). Sparse greedy Gaussian process regression. *Advances in Neural Information Processing Systems 13* (pp. 619–625). Cambridge, Massachussetts: MIT Press.

Smola, A. J., & Schölkopf, B. (2000). Sparse greedy matrix approximation for machine learning. *International Conference on Machine Learning 17* (pp. 911–918). San Francisco, California: Morgan Kaufmann Publishers.

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

Tresp, V. (2000). A Bayesian committee machine. *Neural Computation*, *12*, 2719–2741.

Wahba, G., Lin, X., Gao, F., Xiang, D., Klein, R., & Klein, B. (1999). The bias-variance tradeoff and the randomized GACV. *Advances in Neural Information Processing Systems 11* (pp. 620–626). Cambridge, Massachussetts: MIT Press.

Williams, C., & Rasmussen, C. E. (1996). Gaussian proceses for regression. *Advances in Neural Information Processing Systems 8* (pp. 514–520). Cambridge, Massachussetts: MIT Press.

Williams, C., & Seeger, M. (2001). Using the Nyström method to speed up kernel machines. *Advances in Neural Information Processing Systems 13* (pp. 682–688). Cambridge, Massachussetts: MIT Press.

Wipf, D., Palmer, J., & Rao, B. (2004). Perspectives on sparse Bayesian learning. *Advances in Neural Information Processing Systems 16*. Cambridge, Massachussetts: MIT Press.