Bayesian Support Vector Regression

Martin H. Law (martin@cs.ust.hk) **James T. Kwok** (jamesk@cs.ust.hk)
Department of Computer Science
Hong Kong University of Science and Technology
Clear Water Bay
Hong Kong

Abstract

We show that the Bayesian evidence framework can be applied to both ϵ-support vector regression (ϵ-SVR) and ν-support vector regression (ν-SVR) algorithms. Standard SVR training can be regarded as performing level one inference of the evidence framework, while levels two and three allow automatic adjustments of the regularization and kernel parameters respectively, without the need of a validation set.

1 Introduction

In recent years, there has been a lot of interest in studying the use of support vector machines (SVMs) on various classification and regression problems. SVMs are motivated by results from the statistical learning theory [14] and, unlike other machine learning methods, its performance does not deteriorate even in problems with high input dimensionalities. In this paper, we consider two popular techniques for applying SVMs to the regression problems, namely the ϵ-support vector regression (ϵ-SVR) [1] and the ν-support vector regression (ν-SVR) algorithms [9].

To obtain a high level of performance, some parameters in both SVR algorithms have to be tuned. These include a kernel parameter that helps to define the feature space and a regularization parameter[1] that determines the tradeoff between training accuracy and model complexity. Data-resampling techniques such as cross-validation can be used, but they are usually very expensive in terms of computation and/or data.

In this paper, we address this issue by adopting the Bayesian approach. In general, the Bayesian approach is attractive in being logically consistent, simple and flexible. Recently, various Bayesian techniques have been applied to support vector classification (SVC) [2, 4, 7, 10, 13]. Here, we follow [2, 4] in applying the evidence framework [5] to SVR. The evidence framework is divided into three levels of inference, and is computationally equivalent to the type II maximum likelihood method in Bayesian statistics. Its use in feedforward neural networks [6] has allowed the automatic selection of the regularization parameters and network architectures, without the need of a validation set.

The rest of this paper is organized as follows. A brief overview of ϵ-SVR and ν-SVR will be given in Section 2. The connections between these two SVR algorithms and the evidence framework will be described in Sections 3 and 4 respectively. Simulation results are presented in Section 5, and the last section gives some concluding remarks.

2 ϵ-SVR and ν-SVR

In this section, we briefly review ϵ-SVR and ν-SVR. Interested readers may consult [11, 14] for more details and extensions.

Let the training set D be $\{(\mathbf{x}_i, y_i)\}_{i=1}^N$, with input $\mathbf{x}_i$ and output $y_i \in \Re$. In ϵ-SVR, $\mathbf{x}$ is first mapped to $\mathbf{z} = \phi(\mathbf{x})$ in feature space $\mathcal{F}$, then a linear function $f(\mathbf{x}, \mathbf{w}) = \mathbf{w}^T\mathbf{z} + b$ is constructed in $\mathcal{F}$ such that it deviates least from the training data according to the ϵ-insensitive loss function

$$|y - f(\mathbf{x})|_\epsilon = \begin{cases} 0 & \text{if } |y - f(\mathbf{x})| \le \epsilon, \\ |y - f(\mathbf{x})| - \epsilon & \text{otherwise,} \end{cases}$$

while at the same time is as "flat" as possible (i.e., $\|\mathbf{w}\|$ is as small as possible). Formally, this means

$$\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^N (\xi_i + \xi_i^*), \qquad (1)$$

[1] As will be mentioned in Section 2, ν-SVR requires one more regularization parameter to trade off ϵ against model complexity and training accuracy.

subject to

$$\begin{cases} y_i - f_i \le \epsilon + \xi_i^*, \\ f_i - y_i \le \epsilon + \xi_i, \quad 1 \le i \le N, \\ \xi_i, \xi_i^* \ge 0, \end{cases} \tag{2}$$

where C is a user-defined constant. It is now well-known that (1) can be transformed to a quadratic programming problem.

While the value of ϵ has to be set beforehand in ϵ-SVR, ν-SVR allows the automatic determination of ϵ by using an additional constant $\nu \ge 0$ to trade off ϵ against model complexity and training accuracy. Mathematically, this means

$$\text{minimize } \frac{1}{2}\|\mathbf{w}\|^2 + C\left(\nu\epsilon + \frac{1}{N}\sum_{i=1}^{N}(\xi_i + \xi_i^*)\right), \tag{3}$$

subject to (2) and $\epsilon \ge 0$. Again, (3) can be transformed to a quadratic programming problem.

3 ϵ-SVR and the Evidence Framework

Under the evidence framework, a model $\mathcal{H}$, with a k-dimensional parameter vector $\mathbf{w}$, consists of its functional form f, the distribution $p(D|\mathbf{w}, \beta, \mathcal{H})$ that the model makes about the data D, and a prior parameter distribution $p(\mathbf{w}|\alpha, \mathcal{H})$. Here, α and β are the hyperparameters associated with the two distributions.

3.1 ϵ-SVR and Level 1 Inference

For given values of α and β, the first level of inference infers the posterior distribution of $\mathbf{w}$ by the Bayes rule. Assuming that the patterns are i.i.d., the first level of inference infers the posterior distribution of $\mathbf{w}$ for given values of α, β using

$$p(\mathbf{w}|D, \alpha, \beta, \mathcal{H}) \propto p(\mathbf{w}|\alpha, \mathcal{H}) \prod_i p(y_i|\mathbf{x}_i, \mathbf{w}, \beta, \mathcal{H}) p(\mathbf{x}_i).$$

As discussed in [11, 12], the ϵ-insensitive cost function corresponds to the following probability density function[2]

$$p(y_i|\mathbf{x}_i, \mathbf{w}, \beta, \mathcal{H}) = c\exp(-\beta|y_i - f_i|_\epsilon), \tag{4}$$

where $c = \beta/(2(1+\epsilon\beta))$ is the normalizing factor. Hence, by using the Gaussian prior

$$p(\mathbf{w}|\alpha, \mathcal{H}) \propto \exp(-\frac{\alpha}{2}\|\mathbf{w}\|^2), \tag{5}$$

with $\alpha = \beta/C$, optimizing (1) can be regarded as finding the *maximum a posteriori* (MAP) estimate $\mathbf{w}_{MP}$ of $\mathbf{w}$ and thus performing the level one inference.

[2]Note that we have added a β into (4) to control the noise variance.

Moreover, note that while SVC can only be regarded as *approximately* performing the level one inference [4], here for ϵ-SVR we have exact correspondence as the density function in (4) is normalized. Thus, concerns about the problem of having an un-normalized probability model as in the classification case [13] do not apply here.

3.2 ϵ-SVR and Level 2 Inference

The second level of inference determines α and β by maximizing

$$p(\alpha, \beta|D, \mathcal{H}) \propto p(D|\alpha, \beta, \mathcal{H}) p(\alpha, \beta|\mathcal{H}).$$

When $p(\alpha, \beta|\mathcal{H})$ is a flat prior, the *evidence* for α and β, $p(D|\alpha, \beta, \mathcal{H})$, can be used to assign a preference to alternative values of α and β. In the following, define $E_W = \frac{1}{2}\|\mathbf{w}\|^2$ and $E_D = \sum_{i=1}^{N}(\xi_i + \xi_i^*)$. Similar to [5], we approximate the posterior weight distribution by a single Gaussian at $\mathbf{w}_{MP}$, and the evidence for α and β can then be obtained by integrating out $\mathbf{w}$ as:

$$\begin{aligned} &\log p(D|\alpha, \beta, \mathcal{H}) = \\ &\quad -\alpha E_W^{MP} - \beta E_D^{MP} - \frac{1}{2}\log\det\mathbf{A} + \frac{k}{2}\log\alpha \\ &\quad + N(\log\beta - \log 2 - \log(1+\epsilon\beta)), \end{aligned} \tag{6}$$

where

$$\mathbf{A} = \frac{\partial^2(\alpha E_W + \beta\sum_{i=1}^{N}(\xi_i + \xi_i^*))}{\partial \mathbf{w}^2},$$

E_W^{MP} and E_D^{MP} are the values of E_W and E_D evaluated at $\mathbf{w}_{MP}$.

Instead of directly maximizing $\log p(D|\alpha, \beta, \mathcal{H})$ to obtain the most probable values α_{MP} and β_{MP}, one usually proceeds in an iterative manner [5]: By setting the derivative of $\log p(D|\alpha, \beta, \mathcal{H})$ in (6) w.r.t. α to zero, we obtain the following re-estimation formula:

$$2\alpha E_W^{MP} = \gamma, \tag{7}$$

where $\gamma = k - \alpha\,\text{trace}\mathbf{A}^{-1}$ is often called the *effective number of parameters* [5]. Similarly, for β, we obtain

$$2\epsilon E_D^{MP}\beta^2 + (2E_D^{MP} + \gamma\epsilon)\beta - (2N - \gamma) = 0.$$

Based on the new iterates of α and β, $\mathbf{w}_{MP}$ can be re-estimated using level one inference and then the process re-iterated.

3.2.1 Computing the Hessian for ϵ-SVR

To determine the hessian $\mathbf{A}$, we write $\xi_i = [f_i - y_i - \epsilon]_+$ and $\xi_i^* = [y_i - f_i - \epsilon]_+$ respectively, where $[u]_+ = uI_{\{u>0\}}$. As $I_{\{u>0\}}$ is not smooth and does not have a second derivative, we replace it by the sigmoid function $\varsigma(u) = 1/(1+e^{-\eta u})$. Differentiating w.r.t. $\mathbf{w}$, we get

$$\begin{aligned}\frac{\partial^2 \xi_i}{\partial \mathbf{w}^2} &= r(f_i - y_i - \epsilon)\mathbf{z}_i\mathbf{z}_i^T, \\ \frac{\partial^2 \xi_i^*}{\partial \mathbf{w}^2} &= r(y_i - f_i - \epsilon)\mathbf{z}_i\mathbf{z}_i^T,\end{aligned}$$

where $r(u) = u\varsigma''(u) + 2\varsigma'(u)$ and the prime denotes the derivative w.r.t. the argument of $\varsigma(\cdot)$. Thus, $\mathbf{A} = \alpha\mathbf{I}_k + \beta\mathbf{B}$ where $\mathbf{I}_k$ is the k-dimensional identity matrix, $\mathbf{B} = \sum_{i=1}^N r_i\mathbf{z}_i\mathbf{z}_i^T$ and $r_i = r(f_i - y_i - \epsilon) + r(y_i - f_i - \epsilon)$. As for SVC [2, 4], eigenvalues ρ of $\mathbf{B}$ (and hence also of $\mathbf{A}$) can be obtained from $\rho\mathbf{u} = \mathbf{H}\mathbf{u}$, where $\mathbf{H}$ is a $N \times N$ matrix with entries $r_iK(\mathbf{x}_i, \mathbf{x}_j)$. Using this eigen decomposition, we obtain

$$\log\det\mathbf{A} = \sum_{i=1}^n \log(\alpha + \beta\rho_i) + (k-n)\log\alpha,$$

and

$$\gamma = \sum_{i=1}^n \frac{\beta\rho_i}{\alpha + \beta\rho_i}, \tag{8}$$

where $n \leq N$ is the number of nonzero eigenvalues of $\mathbf{H}$. These can then be used to compute $\log p(D|\alpha, \beta, \mathcal{H})$ and to iterate for α, β as mentioned in Section 3.2[3].

To reduce the $O(N^3)$ time complexity in sthe above eigen system, we notice that r_i becomes very small when $|y_i - f_i| - \epsilon$ is large. Hence, $\mathbf{B}$ is dominated by patterns lying close to the edges of the ϵ-tube, and we can thus significantly reduce the complexity by including only these patterns in $\mathbf{B}$.

3.3 ϵ-SVR and Level 3 Inference

The third level of inference ranks different models by examining their posterior probabilities

$$p(\mathcal{H}|D) \propto p(D|\mathcal{H})p(\mathcal{H}).$$

Assuming a flat prior $p(\mathcal{H})$ for all models, different models can then be rated by the *evidence* $p(D|\mathcal{H})$. Again, this is obtained by integrating[4] out α and β, as

$$p(D|\mathcal{H}) = \int p(D|\alpha, \beta, \mathcal{H})p(\alpha|\mathcal{H})p(\beta|\mathcal{H})d(\log\alpha)d(\log\beta).$$

Using a Gaussian approximation for $p(D|\alpha, \beta, \mathcal{H})$, we have

$$p(D|\mathcal{H}) \propto p(D|\alpha_{MP}, \beta_{MP}, \mathcal{H})\Delta\log\alpha\Delta\log\beta,$$

where

$$(\Delta\log\alpha)^2 = \frac{2}{\gamma}, \tag{9}$$

and

$$(\Delta\log\beta)^2 = \frac{(1+\epsilon\beta_{MP})^2}{\beta_{MP}(E_D^{MP}(1+\epsilon\beta_{MP})^2 + N\epsilon)}.$$

4 ν-SVR and the Evidence Framework

4.1 ν-SVR and Level 1 Inference

Following the probability model (4) and (5) in Section 3.1, we further on adopt the following prior on ϵ (Figure 1):

$$p(\epsilon|\beta, \nu, \mathcal{H}) \propto (1+\epsilon\beta)^N \exp(-N\nu\beta\epsilon). \tag{10}$$

It can then be shown that finding the MAP estimates $\mathbf{w}_{MP}$ and ϵ_{MP} amounts to minimizing

$$\frac{1}{2}\|\mathbf{w}\|^2 + \frac{\beta N}{\alpha}\left(\nu\epsilon + \frac{1}{N}\sum_{i=1}^N(\xi_i + \xi_i^*)\right),$$

which is the same as (3) on setting $C = \beta N/\alpha$.

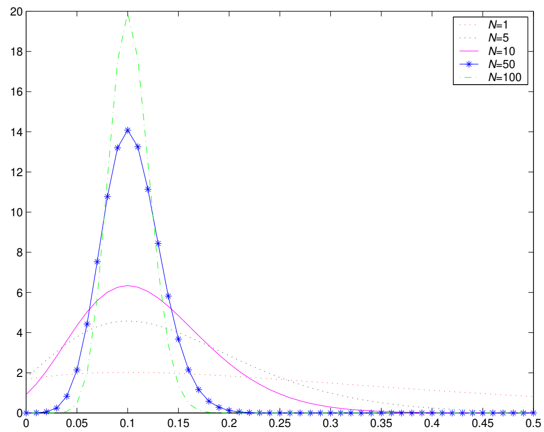


Figure 1: A plot of $p(\epsilon|\beta, \nu, \mathcal{H})$ at different values of N ($\beta = 10, \nu = 0.5$).

As can be seen from Figure 1, $p(\epsilon|\beta, \nu, \mathcal{H})$ has a single peak. This can be explained by first noting that at the peak, $\partial\log p(\epsilon|\beta, \nu, \mathcal{H})/\partial\epsilon = 0$, or

$$\nu = \frac{1}{1+\epsilon\beta}. \tag{11}$$

Now, under our probabilistic model (4), the probability that a particular $\mathbf{x}$ has its corresponding y lying outside the ϵ-tube (and is thus an error) is:

$$1 - \int_{-\epsilon}^{\epsilon} \frac{\beta}{2(1+\epsilon\beta)}\exp(-\beta\cdot 0)d\delta = \frac{1}{1+\epsilon\beta}. \tag{12}$$

[3] Note that this still holds when k is infinite (such as when the Gaussian kernel is used).

[4] As α and β are scale parameters, we perform the integration w.r.t. $\log\alpha$ and $\log\beta$.

Thus, by comparing this with (11), we see that at the peak of $p(\epsilon|\beta,\nu,\mathcal{H})$, the value of ν is in line with the observation that ν equals the fraction of errors (and also the fraction of support vectors) asymptotically with probability one [9]. To the left of this peak, ϵ is so small that the probability in (12) will be greater than ν; whereas to the right of the peak, ϵ is so large that the probability in (12) will be smaller than ν .

Moreover, notice that unlike traditional Bayesian inference, here we have a data dependent prior $p(\epsilon|\beta,\nu,\mathcal{H})$ which depends on N, the size of the data. In fact, as can be seen from Figure 1, the prior becomes more concentrated around the peak as N increases. This difference is due to the fact that in traditional Bayesian inference, with the arrival of more and more data, the effect of the prior diminishes and the posterior becomes more and more dominated by the likelihood term. However, as have been shown earlier, we want to use ν to control the fraction of data points lying outside the ϵ-tube in ν-SVR. This is impossible with a conventional prior. But with the prior in (10), its logarithm grows linearly with N. As the log likelihood also grows linearly with N, the net effect is that our prior belief (as expressed by the prior) will not be diminished with the arrival of more data. In other words, our prior belief that ν determines the fraction of errors has the same strength in comparison to the likelihood, no matter the size of the data set observed.

4.2 ν-SVR and Level 2 Inference

The second level of inference determines α,β and ν in ν-SVR by maximizing $p(\alpha,\beta,\nu|D,\mathcal{H}) \propto p(D|\alpha,\beta,\nu,\mathcal{H})p(\alpha,\beta,\nu|\mathcal{H})$. Again, we approximate the joint posterior distribution of $\mathbf{w}$ and $\log\epsilon$ by a single Gaussian at their MAP values. Thus

$$\begin{aligned}\log p(D|\alpha,\beta,\nu,\mathcal{H}) = \\ &-\alpha E_W^{MP} - \beta E_D^{MP} - \beta N\nu\epsilon_{MP} - \frac{1}{2}\log\det\mathbf{A} \\ &+\frac{k}{2}\log\alpha + N\log\frac{\beta}{2} + \frac{1}{2}\log 2\pi - N\nu \\ &+(N+1)\log(N\nu) + \log\beta - \log\Gamma(N+1,N\nu),\end{aligned}$$

where $\Gamma(N+1,N\nu) = \int_{N\nu}^{\infty} e^{-t}t^N dt$ is related to the incomplete gamma function and can be readily computed [8].

It can be shown that the re-estimation formula for α_{MP} is still (7), while that for β_{MP} is changed to

$$\beta = \frac{2N-\gamma+1}{2(E_D^{MP} + N\nu\epsilon_{MP})}.$$

For ν, the derivative of $\log p(D|\alpha,\beta,\nu,\mathcal{H})$ w.r.t. ν is:

$$\begin{aligned}&-N(1+\beta\epsilon_{MP}) + \frac{N+1}{\nu} \\ &\quad +N\exp(-N\nu + N\log(N\nu) - \log\Gamma(N+1,N\nu)),\end{aligned}$$

and it is not easy to obtain a re-estimation formula for ν by simply setting this derivative to zero. So, instead, we use the Newton's method [8] to find the root of the derivative. At each iteration, a new estimate for ν can be obtained from the old estimate ν^{old} as:

$$\nu^{\text{old}} - \frac{\partial\log p(D|\alpha,\beta,\nu,\mathcal{H})}{\partial\nu} \Big/ \left.\frac{\partial^2\log p(D|\alpha,\beta,\nu,\mathcal{H})}{\partial\nu^2}\right|_{\nu^{\text{old}}},$$

where

$$\begin{aligned}\frac{\partial^2\log p(D|\alpha,\beta,\nu,\mathcal{H})}{\partial\nu^2} = \\ &-\frac{N+1}{\nu^2} - \frac{\partial\log\Gamma(N+1,N\nu)}{\partial\nu}\Big(-N+\frac{N}{\nu} \\ &-\frac{\partial\log\Gamma(N+1,N\nu)}{\partial\nu}\Big),\end{aligned}$$

and

$$\begin{aligned}\frac{\partial\log\Gamma(N+1,N\nu)}{\partial\nu} = \\ &-N\exp(-N\nu + N\log(N\nu) - \log\Gamma(N+1,N\nu)).\end{aligned}$$

4.2.1 Computing the Hessian for ν-SVR

For ν-SVR, the hessian $\mathbf{A}$ now becomes

$$\frac{\partial^2(\alpha E_W + \beta E_D + N\nu\beta\epsilon)}{\partial([\mathbf{w}^T : \log\epsilon]^T)^2}.$$

As ϵ is non-negative, we take the Gaussian approximation w.r.t. to $\mathbf{w}$ and $\log\epsilon$. Similar to Section 3.2.1, we obtain

$$\mathbf{A} = \begin{pmatrix} \alpha\mathbf{I}_k + \beta\mathbf{B} & \beta\epsilon\sum_i s_i\mathbf{z}_i \\ \beta\epsilon\sum_i s_i\mathbf{z}_i^T & \beta\epsilon^2\sum_i r_i \end{pmatrix}, \qquad (13)$$

where $s_i = r(y_i - f_i - \epsilon) - r(f_i - y_i - \epsilon)$. To compute the determinant of $\mathbf{A}$, we make use of the following identity on partitioned matrices:

$$\det\begin{pmatrix} \mathbf{A}_{11} & \mathbf{A}_{12} \\ \mathbf{A}_{21} & \mathbf{A}_{22} \end{pmatrix} = \det(\mathbf{A}_{22})\det(\mathbf{A}_{11} - \mathbf{A}_{12}\mathbf{A}_{22}^{-1}\mathbf{A}_{21}).$$

Note that the sub-matrix $\beta\epsilon^2\sum_i r_i$ in (13) is just a number, and assuming that it is not equal to zero (which always holds in practice), then its inverse always exist. This leads to

$$\det\mathbf{A} = (\beta\epsilon^2\sum_i r_i)\det(\beta\tilde{\mathbf{B}} + \alpha\mathbf{I}_k),$$

where $\tilde{\mathbf{B}} = \mathbf{B} - \sum_{ij} s_i s_j \mathbf{z}_i \mathbf{z}_j^T / \sum_i r_i$. Denote the eigenvalues of $\tilde{\mathbf{B}}$ by $\tilde{\rho}$. It can be shown that these can be obtained by solving the eigen system $\tilde{\rho}\tilde{\mathbf{u}} = \tilde{\mathbf{H}}\tilde{\mathbf{u}}$, where $\tilde{\mathbf{H}} = (\mathrm{diag}(r_i) - \mathbf{s}\mathbf{s}^T / \sum_i r_i)\mathbf{K}$, with $\mathbf{s} = (s_1, \ldots, s_N)$ and $\mathbf{K} = \{K(\mathbf{x}_i, \mathbf{x}_j)\}_{ij}$. We then have

$$\log\det\mathbf{A} = \sum_i \log(\alpha + \beta\tilde{\rho}_i) + (k-n)\log\alpha + \log(\beta\epsilon^2 \sum_i r_i).$$

Moreover, γ is still given by (8) with $\tilde{\rho}_i$ replacing ρ_i.

4.3 ν-SVR and Level 3 Inference

Integrating out α, β and ν from $p(D|\alpha,\beta,\nu,\mathcal{H})$ and using a Gaussian approximation as in Section 3.3, it can be shown that

$$\begin{aligned}\log p(D|\mathcal{H}) = &\log p(D|\alpha_{MP}, \beta_{MP}, \nu_{MP}, \mathcal{H}) + \frac{3}{2}\log(2\pi) \\ &-\frac{1}{2}\log\gamma + \frac{1}{2}\log 2 - \frac{1}{2}\log\Big(\frac{2N+1-\gamma}{2}\big(N+1 \\ &-\nu_{MP}(N\nu_{MP}\beta_{MP}\epsilon_{MP} - 1)\frac{\partial \log\Gamma(N+1, N\nu)}{\partial\nu}\big) \\ &-(N\nu_{MP}\beta_{MP}\epsilon_{MP})^2\Big).\end{aligned}$$

5 Simulation

In this section, we illustrate the results on a toy problem. The target function is $\mathrm{sinc}(x) = \sin(x)/x$, with x uniformly distributed over $[-12, 12]$ and Gaussian noise $\mathrm{N}(0, 0.05^2)$ added. The test set has 10,000 patterns. We repeat the experiments with a total of 10 independent training sets, each of size 80. The Gaussian kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\omega\|\mathbf{x}_i - \mathbf{x}_j\|^2)$ is used.

Figure 2 illustrates the choice of the regularization parameter by level 2 inference. $p(D|\alpha,\beta,\mathcal{H})$ (for ϵ-SVR) and $p(D|\alpha,\beta,\nu,\mathcal{H})$ (for ν-SVR) are plotted against various testing errors at different values of C. Figure 3 illustrates the choice of the kernel width by level 3 inference. $p(D|\mathcal{H})$ is plotted against the testing errors at different values of ω. For a fixed ω, the hyperparameters α, β are obtained from the re-estimation formulas in Section 3.2 (for ϵ-SVR) and Section 4.2 (for ν-SVR). As can be seen from both figures, the evidence follows the testing errors closely.

6 Conclusion

In this paper, we extend the application of the evidence framework to both ϵ-SVR and ν-SVR. As in previous applications on neural networks [5] and SVC [2, 4], this allows automatic adjustment of the regularization and kernel parameters to their near-optimal values, without the need to set data aside in a validation set.

Building on this Bayesian connection, the posterior predictive distribution and error bars for ϵ-SVR and ν-SVR can also be computed (similar to the computation of the moderated outputs for SVC [3]). Thus, a measure of uncertainty can be associated with its output predictions. This can be a major advantage, especially in safety-critical applications. Details of this extension will be reported elsewhere.

References

[1] H.D. Drucker, C.J.C. Burges, L. Kaufman, A. Smola, and V. Vapnik. Support vector regression machines. In M.C. Mozer, M.I. Jordan, and T. Petsche, editors, *Advances in Neural Information Processing Systems 9*, pages 155–161, San Mateo, CA, 1997. Morgan Kaufmann.

[2] J.T. Kwok. Integrating the evidence framework and the support vector machine. In *Proceedings of the European Symposium on Artificial Neural Networks*, pages 177–182, Bruges, Belgium, April 1999.

[3] J.T. Kwok. Moderating the outputs of support vector machine classifiers. *IEEE Transactions on Neural Networks*, 10:1018–1031, 1999.

[4] J.T. Kwok. The evidence framework applied to support vector machines, 2000. To appear in the *IEEE Transactions on Neural Networks.*

[5] D.J.C. MacKay. Bayesian interpolation. *Neural Computation*, 4(3):415–447, May 1992.

[6] D.J.C. MacKay. A practical Bayesian framework for backpropagation networks. *Neural Computation*, 4(3):448–472, May 1992.

[7] M. Opper and O. Winther. GP classification and SVM: Mean field results and leave-one-out estimator. In A.J. Smola, editor, *Advances in Large Margin Classifiers.* MIT, 1999.

[8] W.H. Press, S.A. Teukolsky, W.T. Vetterling, and B.P. Flannery. *Numerical Recipes in C.* Cambridge University Press, New York, 2nd edition, 1992.

[9] B. Schölkopf and A.J. Smola. New support vector algorithms. NeuroCOLT2 Technical Report NC2-TR-1998-031, GMD FIRST, 1998.

[10] M. Seeger. Bayesian model selection for support vector machines, Gaussian processes and other kernel classifiers. In S.A. Solla, T.K. Leen, and

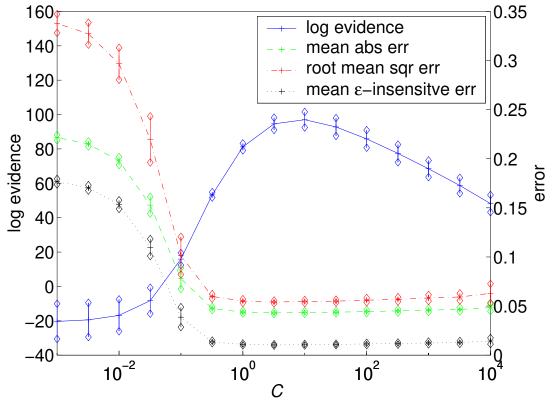
K.-R. Müller, editors, *Advances in Neural Information Processing Systems 12*, San Mateo, CA, 1999. Morgan Kaufmann.

[11] A.J. Smola and B. Schölkopf. A tutorial on support vector regression. NeuroCOLT2 Technical Report NC2-TR-1998-030, Royal Holloway College, 1998.

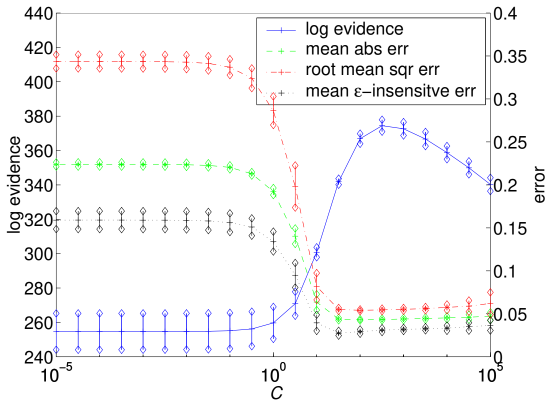
[12] A.J. Smola, B. Schölkopf, and K.-R. Müller. General cost functions for support vector regression. In *Australian Congress on Neural Networks*, 1998.

[13] P. Sollich. Bayesian methods for support vector machines: Evidence and error bars, 2000. Submitted to the Machine Learning Journal.

[14] V. Vapnik. *Statistical Learning Theory*. Wiley, 1998.

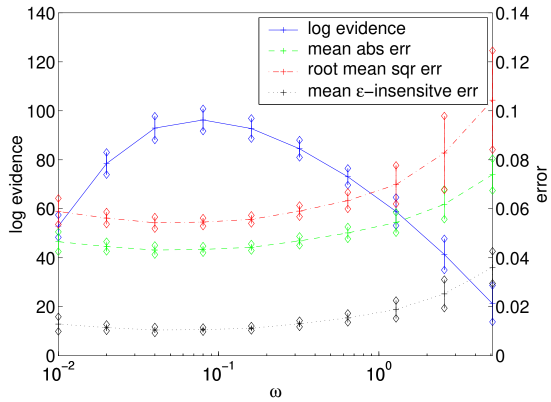


(a) ϵ-SVR ($\epsilon = 0.05$)

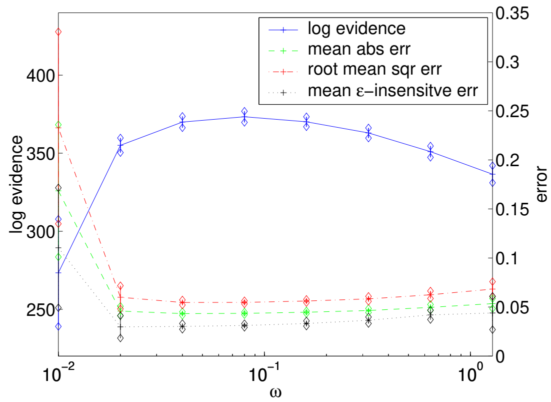


(b) ν-SVR ($\nu = 0.7$)

Figure 2: Level 2 inference results. The error bars correspond to the ± 1 standard deviations based on the converged results among the 10 repetitions.



(a) ϵ-SVR



(b) ν-SVR

Figure 3: Level 3 inference results. Again the error bars correspond to the ± 1 standard deviations based on the converged results among the 10 repetitions.