Automatic Relevance Determination for Least Squares Support Vector Machine Classifiers

T. Van Gestel, J.A.K. Suykens, B. De Moor & J. Vandewalle *

K.U.Leuven Dept. of Electrical Engineering, ESAT-SISTA,
Kasteelpark Arenberg 10, B-3001 Leuven, Belgium

Abstract. Automatic Revelance Determination (ARD) has been applied to multilayer perceptrons by inferring different regularization parameters for the input interconnection layer within the evidence framework. In this paper, this idea is extended towards Least Squares Support Vector Machines (LS-SVMs) for classification. Relating a probabilistic framework to the LS-SVM formulation on the first level of Bayesian inference, the hyperparameters are inferred on the second level. Model comparison is performed on the third level in order to select the parameters of the kernel function. ARD is performed by introducing a diagonal weighting matrix in the kernel function. These diagonal elements are obtained by evidence maximization on the third level of inference. Inputs with a low weight value are less relevant and can be removed.

1. Introduction

The Bayesian evidence framework has been succesfully applied to the design of multilayer perceptrons (MLPs) [1, 3]. The model parameters are inferred from the data by applying Bayes' rule on the first level of inference, with the prior and likelihood corresponding to the regularization and error term, respectively. The hyperparameters that control the trade-off between error minimization and regularization are inferred on the second level. Model comparison can be performed on the third level. Automatic Relevance Determination [3, 5] (ARD) involves the automatic determination of relevant inputs. Within the evidence framework, ARD is applied to MLPs by introducing additional regularization hyperparameters for the interconnections of each input. Evidence maximization is used to infer the regularization parameters and input selection can be performed by removing inputs with relatively large regularization constants.

*E-mail: {tony.vangestel, johan.suykens}@esat.kuleuven.ac.be. T. Van Gestel and J.A.K. Suykens are a Research Assistant and a Postdoctoral Researcher with the Fund for Scientific Research-Flanders (FWO-Vlaanderen), respectively. This work was partially supported by grants and projects from the Flemish Gov.: (Research council KULeuven: Grants, GOA-Mefisto 666; FWO-Vlaanderen: Grants, res. proj. G.0240.99, G.0256.97, and comm. (ICCoS and ANMMM); AWI: Bil. Int. Coll.; IWT: STWW Eureka SINOPSYS, IMPACT); from the Belgian Fed. Gov. (Interuniv. Attr. Poles: IUAP-IV/02, IV/24; Program Dur. Dev.); from the Eur. Comm.: (TMR Netw. (Alapedes, Niconet); Science: ERNSI).

In Support Vector Machines (SVMs) [2, 10] and Least Squares SVMs (LS-SVMs) [8], the inputs $x \in \mathbb{R}^n$ are preprocessed in a nonlinear way by the mapping $\varphi(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n_f}$ that maps the input $x \to \varphi(x)$ in a nonlinear way to a high n_f-dimensional feature space. A linear decision line is then constructed in the feature space space. The mapping $\varphi(x)$ is never explicitly calculated and Mercer's condition $\varphi(x_1)^T\varphi(x_2) = K(x_1, x_2)$ is applied instead. The weights and bias term of the SVM and LS-SVM can be obtained by applying Bayes' rule on the first level of inference [2, 9]. Hyperparameters are inferred on the second level, while the kernel parameters are obtained from model comparison on the third level of inference. In this paper, an ARD algorithm is proposed for LS-SVM classifiers [8] within the Bayesian evidence framework [9]. Since the mapping $\varphi(\cdot)$ is not explicitly known, also the weights of the input layer are unknown and in this sense ARD by optimal hyperparameter selection on level 2 cannot be applied. Instead, ARD for SVMs is obtained by assigning a weight [6] to each input of the kernel function K. These weights are inferred by applying model comparison on the third level of Bayesian inference.

This paper is organized as follows. The inference of the model- and hyperparameters on level 1 and 2 is reviewed in Sections 2 and 3, respectively. Automatic Relevance Determination by model comparison on level 3 is discussed in Section 4. An example is given in Section 5.

2. Inference of the Model Parameters (Level 1)

The LS-SVM classifier $y = \text{sign}[w^T\varphi(x) + b]$ is inferred from the data $D = \{(x_i, y_i)\}_{i=1}^N$ by minimizing the cost function [8]

$$\min_{w,b} \mathcal{J}_1(w, b) = \mu E_W + \zeta E_D = \tfrac{\mu}{2} w^T w + \tfrac{\zeta}{2} \textstyle\sum_{i=1}^N e_i^2 \tag{1}$$

subject to the constraints

$$e_i = 1 - y_i(w^T\varphi(x_i) + b), \qquad i = 1, \ldots, N. \tag{2}$$

The regularization and error term are defined as $E_W = \frac{1}{2}w^T w$ and $E_D = \frac{1}{2}\sum_{i=1}^N e_i^2$, respectively. The trade-off between regularization and training error is determined by the ratio $\gamma = \zeta/\mu$.

This cost function is obtained in [8] by modifying Vapnik's SVM formulation [10] so as to obtain a linear system in the dual space. Constructing the Lagrangian by introducing the Lagrange multipliers α_i for the equality constraints (2), a linear system is obtained in the dual space

$$\left[\begin{array}{c|c} 0 & Y^T \\ \hline Y & \Omega + \gamma^{-1} I_N \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline 1_v \end{array}\right] \tag{3}$$

with $Y = [y_1; \ldots; y_N]$, $1_v = [1; \ldots; 1]$, $e = [e_1; \ldots; e_N]$, $\alpha = [\alpha_1; \ldots; \alpha_N]$, and where Mercer's condition is applied within the Ω matrix $\Omega_{ij} = y_i y_j\, \varphi(x_i)^T\varphi(x_j)$ $= y_i y_j\, K(x_i, x_j)$. Possible kernel functions are, e.g., a linear kernel $K(x_1, x_2) =$

$x_1^T x_2$ and an RBF-kernel $K(x_1, x_2) = \exp(-||x_1 - x_2||_2^2/\sigma^2)$, where Mercer's condition holds for all possible choices of the kernel parameter $\sigma \in \mathbb{R}$. The LS-SVM classifier is then constructed as follows:

$$y(x) \quad = \quad \mathrm{sign}[\textstyle\sum_{i=1}^N \alpha_i y_i K(x, x_i) + b], \tag{4}$$

with latent variable $z = \sum_{i=1}^N \alpha_i y_i K(x, x_i) + b$, by definition.

A probabilistic interpretation for (1)-(2) is obtained by applying Bayes' rule

$$P(w, b|D, \log\mu, \log\zeta, \mathcal{H}) = \frac{P(D|w, b, \log\mu, \log\zeta, \mathcal{H})P(w, b|\log\mu, \log\zeta, \mathcal{H})}{P(D|\log\mu, \log\zeta, \mathcal{H})}, \tag{5}$$

where the model $\mathcal{H}$ corresponds to the kernel function K, possibly with kernel parameters. The evidence $P(D|\log\mu, \log\zeta, \mathcal{H})$ is a normalizing constant. The prior is assumed to be of the form $P(w, b|\log\mu, \log\zeta, \mathcal{H}) = P(w|\log\mu, \mathcal{H})P(b|\mathcal{H})$, with $P(b|\mathcal{H})$ a non-informative uniform distribution. A Gaussian prior $P(w|\log \mu, \mathcal{H}) = (\frac{\mu}{2\pi})^{n_f/2}\exp(-\frac{\mu}{2}w^T w)$ is assumed. The likelihood is equal to $P(D|w, b, \log\zeta, \mathcal{H}) = \prod_{i=1}^N P(x_i|y_i, w, b,\ \log\zeta, \mathcal{H})P(y_i|w, b, \log\zeta, \mathcal{H})$, with the constant prior probabilities $P(y_i|w, b, \log\zeta, \mathcal{H})$ and where the following conditional probability is assumed: $P(x_i|y_i, w, b, \log\zeta, \mathcal{H}) = (\frac{\zeta}{2\pi})^{1/2}\exp[-\frac{\zeta}{2}(1 - y_i(w^T\varphi(x_i) + b))^2]$. By applying Bayes' rule (5), we obtain the posterior probability $P(w, b|D, \log\mu, \log\zeta, \mathcal{H}) \propto \exp(-\frac{\mu}{2}w^T w)\ \exp(-\frac{\zeta}{2}\sum_{i=1}^N e_i^2)$. The maximum a posteriori estimates w_{MP} and b_{MP} are obtained by minimizing the corresponding negative logarithm (1). This is equivalent to solving the linear system (3) in the dual space.

3. Inference of Hyperparameters (Level 2)

Applying Bayes' rule on the second level of inference [2, 3, 9], we obtain:

$$P(\log\mu, \log\zeta|D, \mathcal{H}) \propto \frac{\sqrt{\mu^{n_f}\zeta^N}}{\sqrt{\det H}}\exp(-\mathcal{J}_1(w_{MP}, b_{MP})), \tag{6}$$

with the Hessian $H = \partial^2\mathcal{J}_1(w, b)/\partial[w; b]^2$. In the optimum, the following relations hold [2, 3, 9]: $2\mu_{MP}E_W(w_{MP}) = \gamma_{eff} - 1$ and $2\zeta_{MP}E_D(w_{MP}, b_{MP}) = N - \gamma_{eff}$, which is the Bayesian estimate estimate of the variance $\zeta^{-1} = \sum_{i=1}^N e_i^2/(N - \gamma_{eff})$ of the noise e_i. Combining both relations, we obtain a relation between μ_{MP} and the ratio $\gamma_{MP} = \zeta_{MP}/\mu_{MP}$: $2\mu_{MP}[E_W(w_{MP}) + \gamma_{MP}E_D(w_{MP}, b_{MP})] = N - 1$. For the LS-SVM, the effective number of parameters [1, 2, 3, 9] is equal to:

$$\gamma_{eff} = 1 + \textstyle\sum_{i=1}^{N_{eff}} \frac{\zeta_{MP}\lambda_{G,i}}{\mu_{MP} + \zeta_{MP}\lambda_{G,i}} = 1 + \sum_{i=1}^{N_{eff}} \frac{\gamma_{MP}\lambda_{G,i}}{1 + \gamma_{MP}\lambda_{G,i}}, \tag{7}$$

where the first term is obtained because no regularization on the bias term b is used. The N_{eff} non-zero eigenvalues $\lambda_{G,i}$ corresponds to the N_{eff} non-zero eigenvalues of the centered Gram matrix in the feature space and are the solutions to the eigenvalue problem [9]

$$(I_N - \tfrac{1}{N}YY^T)\Omega\nu_{G,i} \quad = \quad \lambda_{G,i}\nu_{G,i}, \quad i = 1, \ldots, N_{eff} \leq N - 1. \tag{8}$$

A practical way to find the maximum a posteriori estimates μ_{MP}, ζ_{MP} of (6) is to solve first the following scalar minimization problem in γ [9]:

$$\min_{\gamma} \mathcal{J}_2(\gamma) = \sum_{i=1}^{N-1} \log[\lambda_{G,i} + \frac{1}{\gamma}] + (N-1)\log[E_W(w_{MP}) + \gamma E_D(w_{MP}, b_{MP})], \quad (9)$$

with $\lambda_{G,i} = 0$ for $i > N_{eff}$. In this optimization problem, expressions for $E_{D,MP}$ and $E_{W,MP}$ are obtained from the conditions for optimality of the Lagrangian on level 1 [8, 9]: $E_{D,MP} = \frac{1}{2\gamma^2}\sum_{i=1}^{N} \alpha_i^2$ and $E_{W,MP} = \frac{\mu}{2}\alpha^T\Omega\alpha = \frac{1}{2}\sum_{i=1}^{N}\alpha_i(1 - \frac{\alpha_i}{\gamma} - y_i b_{MP})$. From the optimal γ_{MP}, one easily obtains μ_{MP} and ζ_{MP} using the relations in the optimum between μ, ζ, γ, $E_W(w_{MP})$ and $E_D(w_{MP}, b_{MP})$.

4. Automatic Relevance Determination by Inference of Kernel Parameters (Level 3)

By applying Bayes' rule on the third level, the posterior for the model $\mathcal{H}_j$ is obtained: $P(\mathcal{H}_j|D) \propto P(D|\mathcal{H}_j)P(\mathcal{H}_j)$. At this level, no evidence or normalizing constant is used since it is impossible to compare all possible models $\mathcal{H}_j$. The prior $P(\mathcal{H}_j)$ over all possible models is assumed to be uniform here. Hence, we obtain $P(\mathcal{H}_j|D) \propto P(D|\mathcal{H}_j)$. The likelihood $P(D|\mathcal{H}_j)$ corresponds to the evidence (6) of the previous level and can be approximated by [2, 3, 9]

$$P(D|\mathcal{H}_j) \quad \propto \quad P(D|\log\mu_{MP}, \log\zeta_{MP}, \mathcal{H}_j)\frac{\sigma_{\log\mu|D}\sigma_{\log\zeta|D}}{\sigma_{\log\mu}\sigma_{\log\zeta}}, \quad (10)$$

with $\sigma_{\log\mu}, \sigma_{\log\zeta}$ the standard deviations of the Gaussian priors (level 2) on $\log\mu$, $\log\zeta$, respectively.

The error bars $\sigma_{\log\mu|D}$ and $\sigma_{\log\zeta|D}$ can be approximated [3] as follows: $\sigma^2_{\log\mu|D} \simeq \frac{2}{\gamma_{eff}-1}$ and $\sigma^2_{\log\zeta|D} \simeq \frac{2}{N-\gamma_{eff}}$. The posterior (10) becomes [9]:

$$P(D|\mathcal{H}_j) \propto \sqrt{\frac{\mu_{MP}^{N_{eff}}\zeta_{MP}^{N-1}}{(\gamma_{eff}-1)(N-\gamma_{eff})\prod_{i=1}^{N_{eff}}(\mu_{MP}+\zeta_{MP}\lambda_{G,i})}}. \quad (11)$$

One selects the kernel parameters, e.g, σ_j for an RBF-kernel, with maximal posterior $P(D|\mathcal{H}_j)$.

For Automatic Relevance Determination, we now introduce a diagonal[1] weighting matrix [6] $U = \text{diag}([u(1);...;u(n)])$. Each $u(k) \in \mathbb{R}^+$ weights the corresponding input $x(k)$, $k = 1, ..., n$ in the kernel function K. For an RBF-kernel, the kernel function becomes

$$K(x_1, x_2) = \exp(-(x_1 - x_2)^T U(x_1 - x_2)/\sigma^2) = \exp(-(x_1 - x_2)^T \bar{U}(x_1 - x_2)),$$

where the positive scale parameter σ is taken into account by defining $\bar{U} = U/\sigma$ and $\bar{U} = \text{diag}(\bar{u}) = U/\sigma$. The weights $\bar{u}$ are inferred by maximizing the model

[1]Instead of using a diagonal weighting matrix, the approach may be generalized towards any positive definite weighting matrix $\bar{U} \in \mathbb{R}^{n\times n}$. However, a physical interpretation of the importance of the weights is less obvious when there are significantly non-zero off-diagonal elements.

evidence (11). The most relevant inputs will have larger weights, while the less important inputs will have relatively small weights.

In order to find a good starting value for optimizing $\bar{u}$ with respect to (11), we will first infer the optimal σ from (11) for $u = [1; ...; 1]$. This value then serves as the starting point $\bar{u} = [1; ...; 1]/\sigma$ for the more complex optimization of the weights $\bar{u}$. A practical algorithm consists of the following steps:

1. Normalize the inputs to zero mean and unit variance.
2. Optimize σ_j with respect to $P(D|\mathcal{H}_j)$ from (11). For each σ_j, the optimal μ_{MP}, ζ_{MP} and γ_{MP} are inferred on level 2 as follows:
   (a) Solve the eigenvalue problem (8).
   (b) Minimize $\mathcal{J}_2(\gamma)$ from (9), in each step one solves the linear system (3) on level 1 and calculates $E_{W|MP}$ and $E_{D|MP}$.
   (c) Given γ_{MP}, calculate μ_{MP}, ζ_{MP} and γ_{eff}.
   (d) Calculate $P(D|\mathcal{H}_j)$ from (11).
3. Select an initial choice for $\bar{u}$, e.g, $\bar{u} = [1; ...; 1]/\sigma$ (when Step 2 was the previous step) or $\bar{u} = \bar{u}_{prev}(...; l-1; l+1; ...)$ otherwise.
4. Optimize $\bar{u}_j$ with respect to $P(D|\mathcal{H}_j)$ from (11). See Step 2 for the different steps on level 1 (2b) and level 2 (2a-d).
5. Remove inputs l with low $\bar{u}(l)$ values, go back to Step 3.

The main differences of this approach with Gaussian Processes [5] is that GP typically infer the kernel parameters on level 2, together with the hyperparameters. The LS-SVM formulation also allows to derive analytical expressions [2, 9], while sampling techniques have been used to design and evaluate GP [5].

5. Example: ARD with an RBF-kernel

We illustrate the ARD algorithm for an RBF-kernel on the synthetic binary classification dataset from [7]. The data set consists of a training set and test set of $N = 250$ and $N_{test} = 1000$ data points, respectively. Both classes -1 and $+1$ have equal prior probabilities and each class is an equal mixture of two normal distributions. Due to the overlap of the distributions, the optimal theoretical performance that can be achieved is 92.0%. The original problem has two inputs ($n = 2$). The example created to illustrate ARD is inspired on [4]: a first additional input $x(3)$ is constructed from input $x(1)$ by adding Gaussian noise with variance 0.25. This input has some relevance. The second additional input $x(4)$ is zero mean, unit variance Gaussian noise. Then, all inputs $x(1:4)$ where normalized to zero mean and unit variance [1].

From Step 2, we obtained $\sigma = 2.54$, $\mu_{MP} = 1.52$ and $\zeta_{MP} = 2.67$ (with $u = [1; 1; 1; 1]$). The training set and test set performance are 89.6% and 88.5%, respectively. In Step 3, we optimized $\bar{u}$ with respect to (11) for all inputs $x(1:4)$. This yielded $\bar{u} = [0.2237, 0.1307, 0.1804, 0.0016]$, $\mu_{MP} = 1.56$, $\zeta_{MP} = 2.74$, with training and test set performances of 89.6% and 89.2%, respectively. The evolution of $\bar{u}$ during the optimization is depicted in Figure 1. Removing input $x(4)$ with very low relevance, we restarted the optimization with $\bar{u}_{old} = [0.2237, 0.1307, 0.1804]$ for inputs $x(1:3)$. We obtained $\bar{u} = [1.4276, 0.4996, 0.0869]$, $\mu_{MP} = 2.31$, $\zeta_{MP} = 3.02$, while the training and

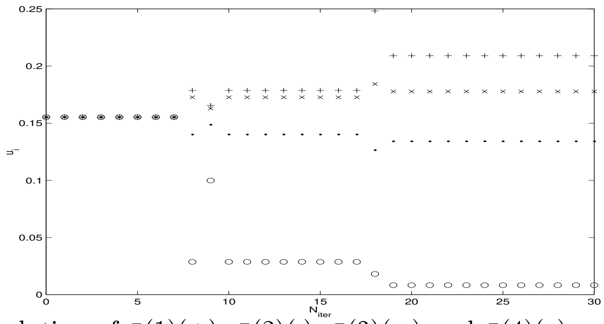

Figure 1: Evolution of $\bar{u}(1)(+)$, $\bar{u}(2)(\cdot)$, $\bar{u}(3)(\times)$ and $\bar{u}(4)(o)$ as a function of the number of iterations N_{iter} of the optimization algorithm.

test performances were 90.0% and 90.8%, respectively. Input $x(1)$ is now far more inportant than input $x(3)$. Removal of $x(3)$ and retraining with inputs $x(1:2)$ yields $\bar{u} = [1.9461, 0.1386]$, $\mu_{MP} = 1.56$ and $\zeta_{MP} = 2.87$. Training and test set performances are now 89.6% and 91.0%, respectively.

6. Conclusions

An Automatic Relevance Determination (ARD) algorithm is proposed for LS-SVM classifiers within the evidence framework. A diagonal weighting matrix is introduced for the inputs of the RBF-kernel. The weights are inferred on the third level of Bayesian inference. Inputs corresponding to small weights have low relevance in the kernel function and can be removed. Although the RBF kernel is known to be quite insensitive to irrevelant inputs, the generalization behavior in our experiment is improved by using a weighting matrix.

References

[1] Bishop, C.M. *Neural Networks for Pattern Recognition*, Oxford University Press, 1995.

[2] Kwok, J.T. Integrating the evidence framework and the Support Vector Machine. In *Proc. of the European Symposium on Artificial Neural Networks (ESANN 1999)*, 177-182, Bruges, Belgium, 1999.

[3] MacKay, D.J.C. Probable Networks and Plausible Predictions - A Review of Practical Bayesian Methods for Supervised Neural Networks. *Network: Computation in Neural Systems, 6*, 469-505, 1995.

[4] Nabney, I. *Netlab: Algorithms for Pattern Recognition*, 2001, to appear.

[5] Neal, R.M. *Bayesian Learning for Neural Networks.* Lecture Notes in Statistics 118, Springer, New York, 1996.

[6] Poggio, T. & Girosi, F. Networks for Approximation and Learning. *Proceedings of the IEEE*, 78(9), 1481-1497, 1990.

[7] Ripley, B.D. Neural Networks and Related Methods for Classification, *Journal Royal Statistical Society B, 56(3)*, 409-456, 1994.

[8] Suykens, J.A.K. & Vandewalle, J. Least squares support vector machine classifiers, *Neural Processing Letters, 9*, 293-300, 1999.

[9] Van Gestel, T., Suykens J.A.K., Lanckriet, G., Lambrechts, A., De Moor, B. & Vandewalle, J. A Bayesian Framework for Least Squares Support Vector Machine Classifiers. *Report TR00-65 ESAT-SISTA, K.U.Leuven, Belgium*, 2000. Submitted for publication.

[10] Vapnik, V. *Statistical learning theory,* John Wiley, New-York, 1998.