Available online at www.sciencedirect.com
SCIENCE DIRECT®
European Journal of Operational Research 172 (2006) 979–1003
EUROPEAN JOURNAL OF OPERATIONAL RESEARCH
www.elsevier.com/locate/ejor

Interfaces with Other Disciplines

Bayesian kernel based classification for financial distress detection

Tony Van Gestel [a,b], Bart Baesens [c,*], Johan A.K. Suykens [b], Dirk Van den Poel [d], Dirk-Emma Baestaens [e], Marleen Willekens [c]

[a] *DEXIA Group, Credit Risk Modelling, RMG, Square Meeus 1, Brussels B-1000, Belgium*
[b] *Katholieke Universiteit Leuven, Department of Electrical Engineering, ESAT, SCD-SISTA, Kasteelpark Arenberg 10, Leuven B-3001, Belgium*
[c] *Katholieke Universiteit Leuven, Department of Applied Economic Sciences, LIRIS, Naamsestraat 69, Leuven B-3000, Belgium*
[d] *Ghent University, Department of Marketing, Hoveniersberg 24, Gent 9000, Belgium*
[e] *Fortis Bank Brussels, Financial Markets Research, Warandeberg 3, Brussels B-1000, Belgium*

Received 7 August 2003; accepted 3 November 2004
Available online 18 January 2005

Abstract

Corporate credit granting is a key commercial activity of financial institutions nowadays. A critical first step in the credit granting process usually involves a careful financial analysis of the creditworthiness of the potential client. Wrong decisions result either in foregoing valuable clients or, more severely, in substantial capital losses if the client subsequently defaults. It is thus of crucial importance to develop models that estimate the probability of corporate bankruptcy with a high degree of accuracy. Many studies focused on the use of financial ratios in linear statistical models, such as linear discriminant analysis and logistic regression. However, the obtained error rates are often high. In this paper, Least Squares Support Vector Machine (LS-SVM) classifiers, also known as kernel Fisher discriminant analysis, are applied within the Bayesian evidence framework in order to automatically infer and analyze the creditworthiness of potential corporate clients. The inferred posterior class probabilities of bankruptcy are then used to analyze the sensitivity of the classifier output with respect to the given inputs and to assist in the credit assignment decision making process. The suggested nonlinear kernel based classifiers yield better performances than linear discriminant analysis and logistic regression when applied to a real-life data set concerning commercial credit granting to mid-cap Belgian and Dutch firms.
© 2004 Elsevier B.V. All rights reserved.

* Corresponding author.
E-mail addresses: tony.vangestel@dexia.com, tony.vangestel@esat.kuleuven.ac.be (T. Van Gestel), bart.baesens@econ.kuleuven.ac.be (B. Baesens), johan.suykens@esat.kuleuven.ac.be (J.A.K. Suykens), dirk.vandenpoel@ugent.be (D. Van den Poel), dirk.baestaens@fortisbank.com (D.-E. Baestaens), marleen.willekens@econ.kuleuven.ac.be (M. Willekens).

0377-2217/$ - see front matter © 2004 Elsevier B.V. All rights reserved.
doi:10.1016/j.ejor.2004.11.009

Keywords: Credit scoring; Kernel Fisher discriminant analysis; Least Squares Support Vector Machine classifiers; Bayesian inference

1. Introduction

Corporate bankruptcy does not only cause substantial losses to the business community, but also to society as a whole. Therefore, accurate bankruptcy prediction models are of critical importance to various stakeholders (i.e. management, investors, employees, shareholders and other interested parties) as it provides them with timely warnings. From a managerial perspective, financial failure forecasting tools allow to take timely strategic actions such that financial distress can be avoided. For other stakeholders, such as banks, efficient and automated credit rating tools allow to detect clients that are to default their obligations at an early stage. Hence, accurate bankruptcy prediction tools will enable them to increase the efficiency of one of their core activities, i.e. commercial credit assignment.

Financial failure occurs when the firm has chronic and serious losses and/or when the firm becomes insolvent with liabilities that are disproportionate to assets. Widely identified causes and symptoms of financial failure include poor management, autocratic leadership and difficulties in operating successfully in the market. The common assumption underlying bankruptcy prediction is that a firm's financial statements appropriately reflect all these characteristics. Several classification techniques have been suggested to predict financial distress using ratios and data originating from these statements. While early univariate approaches used ratio analysis, multivariate approaches combine multiple ratios and characteristics to predict potential financial distress [1–3]. Linear multiple discriminant approaches (LDA), like Altman's Z-Scores, attempt to identify the most efficient hyperplane to linearly separate between successful and non-successful firms. At the same time, the most significant combination of predictors is identified by using a stepwise selection procedure. However, these techniques typically rely on the linear separability assumption, as well as normality assumptions.

Motivated by their universal approximation property, multilayer perceptron (MLP) neural networks [4] have been applied to model nonlinear decision boundaries in bankruptcy prediction and credit assignment problems [5–11]. Although advanced learning methods like Bayesian inference [12,13] have been developed for MLPs, their practical design suffers from drawbacks like the non-convex optimization problem and the choice of the number of hidden units. In Support Vector Machines (SVMs), Least Squares SVMs (LS-SVMs) and related kernel based learning techniques [14–17], the inputs are first mapped into a high dimensional kernel induced feature space in which the regressor or classifier are constructed by minimizing an appropriate convex cost function. Applying Mercer's theorem, the solution is obtained in the dual space from a finite dimensional convex quadratic programming problem for SVMs or a linear Karush–Kuhn–Tucker system in the case of LS-SVMs, avoiding explicit knowledge of the high dimensional mapping and using only the related positive (semi) definite kernel function.

In this paper, we apply LS-SVM classifiers [16,18], also known as kernel Fisher Discriminant Analysis [19,20], within the Bayesian evidence framework [20,21] to predict financial distress of Belgian and Dutch firms with middle market capitalization. After having inferred the hyperparameters of the LS-SVM classifier on different levels of inference, we apply a backward input selection procedure by ranking the model evidence of the different input sets. Posterior class probabilities are obtained by marginalizing over the model parameters in order to infer the probability of making a correct decision and to detect difficult cases that should be referred to further investigation. The obtained results are compared with linear discriminant analysis and logistic regression using leave-one-out cross-validation [22].

This paper is organized as follows. The linear and nonlinear kernel based classification techniques are reviewed in Sections 2–4. Bayesian learning for LS-SVMs is outlined in Section 5. Empirical results on financial distress prediction are reported in Section 6.

2. Empirical linear discriminant analysis

Given a number n of explanatory variables or inputs $\boldsymbol{x} = [x_1; \ldots; x_n] \in \mathbb{R}^n$ of a firm, the problem we are concerned with is to predict whether this firm will default its obligations ($y = -1$) or not ($y = +1$). This problem corresponds to a binary classification problem with class $\mathscr{C}_-(y = -1)$ denoting the class of (future) bankrupt firms and class $\mathscr{C}_+(y = +1)$ the class of solvent firms. Let $p(\boldsymbol{x}|y)$ denote the class probability density of observing the inputs $\boldsymbol{x}$ given the class label y and let $\pi_+ = \mathrm{P}(y = +1)$, $\pi_- = \mathrm{P}(y = -1)$ denote the prior class probabilities, then the Bayesian decision rule to predict $\hat{y}$ is as follows:

$$\hat{y} = \operatorname{sign}[\mathrm{P}(y = +1|\boldsymbol{x}) - \mathrm{P}(y = -1|\boldsymbol{x})], \tag{1}$$

$$\hat{y} = \operatorname{sign}[\log(\mathrm{P}(y = +1|\boldsymbol{x})) - \log(\mathrm{P}(y = -1|\boldsymbol{x}))], \tag{2}$$

$$\hat{y} = \operatorname{sign}[\log(\mathrm{p}(\boldsymbol{x}|y = +1)) - \log(\mathrm{p}(\boldsymbol{x}|y = -1)) + \log(\pi_+/\pi_-)], \tag{3}$$

where the third expression is obtained by applying Bayes' formula

$$\mathrm{p}(y|\boldsymbol{x}) = \frac{\mathrm{p}(y)\mathrm{p}(\boldsymbol{x}|y)}{\mathrm{P}(y = +1)\mathrm{p}(\boldsymbol{x}|y = +1) + \mathrm{P}(y = -1)\mathrm{p}(\boldsymbol{x}|y = -1)}$$

and omitting the normalizing constant in the denominator. This Bayesian decision rule is known to yield optimal performance as it minimizes the risk of misclassification for each instance $\boldsymbol{x}$. In the case of Gaussian class densities with means $\boldsymbol{m}_-$, $\boldsymbol{m}_+$ and equal covariance matrix $\boldsymbol{\Sigma}_{\mathrm{x}}$, the Bayesian decision rule becomes [4,23,24]

$$\hat{y} = \operatorname{sign}[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b] = \operatorname{sign}[z] \tag{4}$$

with latent variable $z = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x} + b$ and where $\boldsymbol{w} = \boldsymbol{\Sigma}_{\mathrm{x}}^{-1}(\boldsymbol{m}_+ - \boldsymbol{m}_-)$ and $b = \boldsymbol{w}^{\mathrm{T}}(\boldsymbol{m}_+ + \boldsymbol{m}_-)/2 + \log(\pi_+/\pi_-)$. This is known as Linear Discriminant Analysis (LDA). In the case of unequal class covariance matrices, a quadratic discriminant is obtained [23].

As the class densities $\mathrm{p}(\boldsymbol{x}|y)$ are typically unknown in practice, one has to estimate the decision rule from given training data $\mathscr{D} = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$. A common way to estimate the linear discriminant (4) is by solving

$$(\hat{\boldsymbol{w}}, \hat{b}) = \arg\min_{\boldsymbol{w},b} \frac{1}{2}\sum_{i=1}^{N}(y_i - (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b))^2. \tag{5}$$

The solution $(\hat{\boldsymbol{w}}, \hat{b})$ follows from a linear set of equations of dimension $(n + 1) \times (n + 1)$ and corresponds [1] to the Fisher Discriminant solution [25], which has been used in the pioneering paper of Altman [1]. The least squares formulation with binary targets $(-1, +1)$ has the additional interpretation as an asymptotical optimal least squares approximation to the Bayesian discriminant function $\mathrm{P}(y = +1|\boldsymbol{x}) - \mathrm{P}(y = -1|\boldsymbol{x})$ [23]. This formulation is also often used for training neural network classifiers [4,16].

Instead of minimizing a least squares cost function or estimating the covariance matrices, one may also relate the probability $\mathrm{P}(y = +1)$ to the latent variable z via the logistic link function [26]. The probabilistic interpretation of the inverse link function $\mathrm{P}(y = +1) = 1/(1 + \exp(-z))$ allows to estimate $\hat{\boldsymbol{w}}$ and $\hat{b}$ from maximum likelihood [26]:

$$(\hat{\boldsymbol{w}}, \hat{b}) = \arg\min_{\boldsymbol{w},b} \sum_{i=1}^{N} \log\left(1 + \exp(-y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b))\right). \tag{6}$$

[1] More precisely, Fisher related the maximization of the Rayleigh quotient to a regression approach with targets $(-N/n_{\mathscr{D}}^-, N/n_{\mathscr{D}}^+)$, with $n_{\mathscr{D}}^+$ and $n_{\mathscr{D}}^-$ the number of positive and negative training instances. The solution only differs in the choice of the bias term b and a scaling of the coefficients $\boldsymbol{w}$.

No analytic solution exists, but the solution can be obtained by applying Newton's method corresponding to an iteratively reweighted least squares algorithm [24]. The first use of applying logistic regression for bankruptcy prediction has been reported in [27].

3. Support vector machines and kernel based learning

The Multilayer Perceptron (MLP) neural network is a popular neural network for both regression and classification and has often been used for bankruptcy prediction and credit scoring in general [6,28–30]. Although there exist good training algorithms (e.g. Bayesian inference) to design the MLP, there are still a number of drawbacks like the choice of the architecture of the MLP and the existence of multiple local minima, which implies that the estimated parameters may not be uniquely determined. Recently, a new learning technique emerged, called Support Vector Machines (SVMs) and related kernel based learning methods in general, in which the solution is unique and follows from a convex optimization problem [15,16,31,32]. The regression formulations are also related to kernel Fisher discriminant analysis [20], Gaussian processes and regularization networks [33], where the latter have been applied to modelling option prices [34].

Although the general nonlinear version of Support Vector Machines (SVM) is quite recent, the roots of the SVM approach for constructing an optimal separating hyperplane for pattern recognition date back to 1963 and 1964 [35,36].

3.1. Linear SVM classifier: Separable case

Consider a training set of N data points $\{(\mathbf{x}_i, y_i)\}_{i=1}^{N}$, with input data $\mathbf{x}_i \in \mathbb{R}^n$ and corresponding binary class labels $y_i \in \{-1, +1\}$. When the data of the two classes are separable (Fig. 1a), one can say that

$$\begin{cases} \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b \geqslant +1 & \text{if } y_i = +1, \\ \boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b \leqslant -1 & \text{if } y_i = -1. \end{cases}$$

This set of two inequalities can be combined into one single set as follows:

$$y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant +1, \quad i = 1, \ldots, N. \tag{7}$$

As can be seen from Fig. 1a, multiple solutions are possible. From a generalization perspective, it is best to choose the solution with largest margin $2/\|\boldsymbol{w}\|_2$.

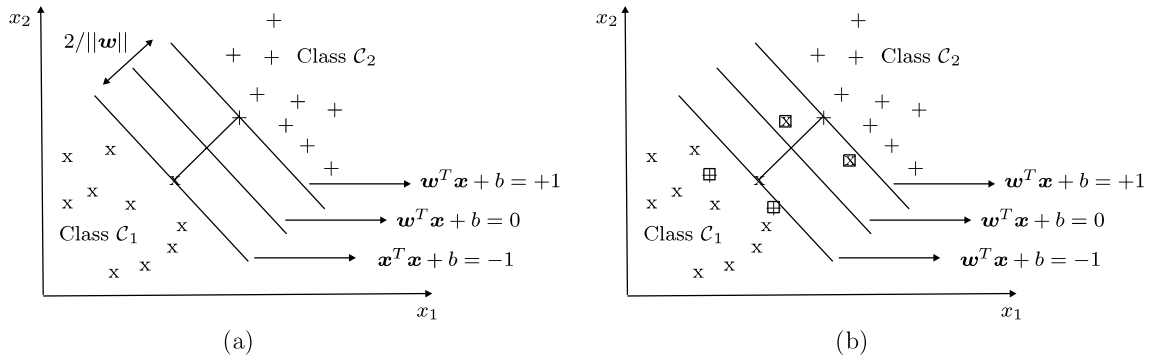


Fig. 1. Illustration of linear SVM classification in a two dimensional input space: (a) separable case; (b) non-separable case. The margin of the SVM classifier is equal to $2/\|\boldsymbol{w}\|_2$.

Support vector machines are modelled within a context of convex optimization theory [37]. The general methodology is to start formulating the problem in the primal weight space as a constrained optimization problem, next formulate the Lagrangian, take the conditions for optimality and finally solve the problem in the dual space of Lagrange multipliers, which are also called support values. The optimization problem for the separable case aims at maximizing the margin $2/\|\boldsymbol{w}\|_2$ subject to the constraint that all training data points need to be correctly classified. This gives the following primal (P) problem in $\boldsymbol{w}$:

$$\begin{aligned} \min_{\boldsymbol{w},b} \quad & \mathscr{J}_{\mathrm{P}}(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant 1, \quad i = 1, \ldots, N. \end{aligned} \tag{8}$$

The Lagrangian for this constraint optimization problem is $\mathscr{L}(\boldsymbol{w}, b; \boldsymbol{\alpha}) = 0.5\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} - \sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1)$, with Lagrange multipliers $\alpha_i \geqslant 0$ ($i = 1, \ldots, N$). The solution is the saddle point of the Lagrangian:

$$\max_{\boldsymbol{\alpha}} \min_{\boldsymbol{w},b} \mathscr{L}. \tag{9}$$

The conditions for optimality for $\boldsymbol{w}$ and b are

$$\begin{cases} \dfrac{\partial \mathscr{L}}{\partial \boldsymbol{w}} \to \boldsymbol{w} = \sum_{i=1}^{N}\alpha_i y_i \boldsymbol{x}_i, \\ \dfrac{\partial \mathscr{L}}{\partial b} \to \sum_{i=1}^{N}\alpha_i y_i = 0. \end{cases} \tag{10}$$

From the first condition in (10), the classifier (4) expressed in terms of the Lagrange multipliers (support values) becomes

$$y(\mathbf{x}) = \operatorname{sign}\left(\sum_{i=1}^{N}\alpha_i y_i \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x} + b\right). \tag{11}$$

Replacing (10) into (9), the dual (D) problem in the Lagrange multipliers $\boldsymbol{\alpha}$ is the following Quadratic Programming problem (QP):

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \mathscr{J}_{\mathrm{D}}(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{i,j=1}^{N} y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j \alpha_i \alpha_j + \sum_{i=1}^{N}\alpha_i = -\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\Omega}\boldsymbol{\alpha} + \mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha} \\ \text{s.t.} \quad & \sum_{i=1}^{N}\alpha_i y_i = 0, \\ & \alpha_i \geqslant 0, \quad i = 1, \ldots, N, \end{aligned} \tag{12}$$

with $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^{\mathrm{T}}$, $\mathbf{1} = [1, \ldots, 1]^{\mathrm{T}} \in \mathbb{R}^N$ and $\boldsymbol{\Omega} \in \mathbb{R}^{N\times N}$, where $\boldsymbol{\Omega}_{ij} = y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$ ($i, j = 1, \ldots, N$). The matrix $\boldsymbol{\Omega}$ is positive (semi-) definite by construction. In the case of a positive definite matrix, the solution to this QP problem is global and unique. In the case of a positive semi-definite matrix, the solution is global, but not necessarily unique in terms of the Lagrange multipliers α_i, while still a unique solution in terms of $\boldsymbol{w} = \sum_{i=1}^{N}\alpha_i y_i \boldsymbol{x}_i$ is obtained [37]. An interesting property, called the sparseness property, is that many of the resulting α_i values are equal to zero. The training data points $\boldsymbol{x}_i$ corresponding to non-zero α_i are called support vectors. These support vectors are located close to the decision boundary. From a non-zero support value $\alpha_i > 0$, b is obtained from $y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1 = 0$.

3.2. Linear SVM classifier: Non-separable case

In most practical, real-life classification problems, the data are non-separable in linear or nonlinear sense, due to the overlap between the two classes (see Fig. 1b). In such cases, one aims at finding a classifier

that separates the data as much as possible. The SVM classifier formulation (8) is extended to the non-separable case by introducing slack variables $\xi_i \geqslant 0$ in order to tolerate misclassifications [38]. The inequalities are changed into

$$y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant 1 - \xi_i, \quad i = 1, \ldots, N, \tag{13}$$

where the ith inequality is violated when $\xi_i > 1$.

In the primal weight space, the optimization problem becomes

$$\begin{aligned} \min_{\boldsymbol{w},b,\boldsymbol{\xi}} \quad & \mathscr{J}_{\mathrm{P}}(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + c\sum_{i=1}^{N}\xi_i \\ \text{s.t.} \quad & y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) \geqslant 1 - \xi_i, \quad i = 1, \ldots, N, \\ & \xi_i \geqslant 0, \quad i = 1, \ldots, N, \end{aligned} \tag{14}$$

where c is a positive real constant that determines the trade-off between the large margin term $0.5\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}$ and error term $\sum_{i=1}^{N}\xi_i$. The Lagrangian is equal to $\mathscr{L} = 0.5\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + c\sum_{i=1}^{N}\xi_i - \sum_{i=1}^{N}\alpha_i(y_i(\boldsymbol{w}^{\mathrm{T}}\boldsymbol{x}_i + b) - 1 + \xi_i) - \sum_{i=1}^{N}v_i\xi_i$, with Lagrange multipliers $\alpha_i \geqslant 0$, $v_i \geqslant 0$ ($i = 1, \ldots, N$). The solution is given by the saddle point of the Lagrangian $\max_{\boldsymbol{\alpha},\boldsymbol{v}}\min_{\boldsymbol{w},b,\boldsymbol{\xi}}\mathscr{L}(\boldsymbol{w}, b, \boldsymbol{\xi}; \boldsymbol{\alpha}, \boldsymbol{v})$, with conditions for optimality

$$\begin{cases} \dfrac{\partial\mathscr{L}}{\partial\boldsymbol{w}} \to \boldsymbol{w} = \sum_{i=1}^{N}\alpha_i y_i \boldsymbol{x}_i, \\ \dfrac{\partial\mathscr{L}}{\partial b} \to \sum_{i=1}^{N}\alpha_i y_i = 0, \\ \dfrac{\partial\mathscr{L}}{\partial\xi_i} \to 0 \leqslant \alpha_i \leqslant c, \quad i = 1, \ldots, N. \end{cases} \tag{15}$$

Replacing (15) in (14) yields the following dual QP-problem:

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \mathscr{J}_{\mathrm{D}}(\boldsymbol{\alpha}) = -\frac{1}{2}\sum_{i,j=1}^{N} y_i y_j \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j\alpha_i\alpha_j + \sum_{i=1}^{N}\alpha_i = -\frac{1}{2}\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\Omega}\boldsymbol{\alpha} + \mathbf{1}^{\mathrm{T}}\boldsymbol{\alpha} \\ \text{s.t.} \quad & \sum_{i=1}^{N}\alpha_i y_i = 0, \\ & 0 \leqslant \alpha_i \leqslant c, \quad i = 1, \ldots, N. \end{aligned} \tag{16}$$

The bias term b is obtained as a by-product of the QP-calculation or from a non-zero support value.

3.3. Kernel trick and Mercer condition

The linear SVM classifier is extended to a nonlinear SVM classifier by first mapping the inputs in a nonlinear way $\boldsymbol{x} \mapsto \boldsymbol{\varphi}(\boldsymbol{x})$ into a high dimensional space, called feature space in SVM terminology. In this high dimensional feature space, a linear separating hyperplane $\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}) + b = 0$ is constructed using (12), as is depicted in Fig. 2.

A key element of nonlinear SVMs is that the nonlinear mapping $\boldsymbol{\varphi}(\cdot) : \boldsymbol{x} \mapsto \boldsymbol{\varphi}(\boldsymbol{x})$ may not be explicitly known, but is defined implicitly in terms of the positive (semi-) definite kernel function satisfying the Mercer condition

$$K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{\varphi}(\boldsymbol{x}_1)^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_2). \tag{17}$$

Given the kernel function $K(\boldsymbol{x}_1, \boldsymbol{x}_2)$, the nonlinear classifier is obtained by solving the dual QP-problem, in which the product $\boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$ is replaced by $\boldsymbol{\varphi}(\boldsymbol{x}_i)^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$, e.g., $\boldsymbol{\Omega} = [y_i y_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_j)]$. The nonlinear SVM classifier is then obtained as

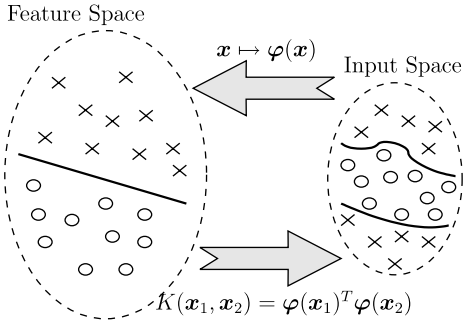


Fig. 2. Illustration of SVM based classification. The inputs are first mapped in a nonlinear way to a high-dimensional feature space ($x \mapsto \varphi(x)$), in which a linear separating hyperplane is constructed. Applying the Mercer condition ($K(x_i,x_j) = \varphi(x_1)^{\mathrm{T}}\varphi(x_2)$), a nonlinear classifier in the input space is obtained.

$$y(\mathbf{x}) = \operatorname{sign}[\mathbf{w}^{\mathrm{T}}\boldsymbol{\varphi}(\mathbf{x}) + b] = \operatorname{sign}\left[\sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b\right]. \tag{18}$$

In the dual space, the score $z = \sum_{i=1}^{N} \alpha_i y_i K(\boldsymbol{x}_i, \boldsymbol{x}) + b$ is obtained as a weighted sum of the kernel functions evaluated in the support vectors and the evaluated point $\boldsymbol{x}$, with weights $\alpha_i y_i$.

A popular choice for the kernel function is the radial basis function (RBF) kernel $K(\mathbf{x}_i, \mathbf{x}_j) = \exp\{-\|\mathbf{x}_i - \mathbf{x}_j\|_2^2/\sigma^2\}$, where σ is a tuning parameter. Other typical kernel functions are the linear kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j$; the polynomial kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\tau + \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j)^d$ with degree d and tuning parameter $\tau \geqslant 0$; and MLP kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \tanh(\kappa_1 \boldsymbol{x}_i^{\mathrm{T}}\boldsymbol{x}_j + \kappa_2)$. The latter is not positive semi-definite for all choices of the tuning parameters κ_1 and κ_2.

4. Least Squares Support Vector Machines

The LS-SVM classifier formulation can be obtained by modifying the SVM classifier formulation as follows:

$$\min_{\boldsymbol{w},b,\boldsymbol{e}} \quad \mathscr{J}_{\mathrm{P}}(\boldsymbol{w}) = \frac{1}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \frac{\gamma}{2}\sum_{i=1}^{N} e_{C,i}^2 \tag{19}$$

$$\text{s.t.} \quad y_i[\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b] = 1 - e_{C,i}, \quad i = 1, \ldots, N. \tag{20}$$

Besides the quadratic cost function, an important difference with standard SVMs is that the formulation consists now of equality instead of inequality constraints [16].

The LS-SVM classifier formulation (19), (20) implicitly corresponds to a regression interpretation (22), (23) with binary targets $y_i = \pm 1$. By multiplying the error $e_{C,i}$ with y_i and using $y_i^2 = 1$, the sum of squared error term $\sum_{i=1}^{N} e_{C,i}^2$ becomes

$$\sum_{i=1}^{N} e_{C,i}^2 = \sum_{i=1}^{N} (y_i e_{C,i})^2 = \sum_{i=1}^{N} e_i^2 = (y_i - (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b))^2 \tag{21}$$

with the regression error $e_i = y_i - (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}_i) + b) = y_i e_{C,i}$. The LS-SVM classifier is then constructed as follows:

$$\min_{w,b,e} \quad \mathcal{J}_P = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{i=1}^{N} e_i^2 \tag{22}$$

$$\text{s.t.} \quad e_i = y_i - (w^T \varphi(x_i) + b), \quad i = 1, \ldots, N. \tag{23}$$

Observe that the cost function is a weighted sum of a regularization term $\mathcal{J}_w = 0.5 w^T w$ and an error term $\mathcal{J}_e = 0.5 \sum_{i=1}^{N} e_i^2$.

One then solves the constrained optimization problem (22), (23) by constructing the Lagrangian $\mathcal{L}(w, b, e; \alpha) = w^T w + \gamma \frac{1}{2} \sum_{i=1}^{N} e_i^2 - \sum_{i=1}^{N} \alpha_i (w^T \varphi(x_i) + b + e_i - y_i)$, with Lagrange multipliers $\alpha_i \in \mathbb{R}$ ($i = 1, \ldots, N$). The conditions for optimality are given by

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial w} = 0 \to w = \sum_{i=1}^{N} \alpha_i x_i, \\ \dfrac{\partial \mathcal{L}}{\partial b} = 0 \to \sum_{i=1}^{N} \alpha_i y_i = 0, \\ \dfrac{\partial \mathcal{L}}{\partial e} = 0 \to \alpha = \gamma e, \\ \dfrac{\partial \mathcal{L}}{\partial \alpha_i} = 0 \to w^T \varphi(x_i) + b + e_i - y_i = 0, \quad i = 1, \ldots, N. \end{cases} \tag{24}$$

After elimination of the variables w and e, one gets the following linear Karush–Kuhn–Tucker (KKT) system of dimension $(N + 1) \times (N + 1)$ in the dual space [16,18,20]:

$$\left[\begin{array}{c|c} 0 & 1^T \\ \hline 1 & \Omega + \gamma^{-1} I_N \end{array}\right] \left[\begin{array}{c} b \\ \hline \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ \hline y \end{array}\right], \tag{25}$$

with $y = [y_1; \ldots; y_N]$, $1 = [1; \ldots; 1]$, and $\alpha = [\alpha_1; \ldots; \alpha_N] \in \mathbb{R}^N$ and where Mercer's theorem [14,15,17] is applied within the Ω matrix: $\Omega_{ij} = \varphi(x_i)^T \varphi(x_j) = K(x_i, x_j)$. The LS-SVM classifier is then obtained as follows:

$$\hat{y} = \operatorname{sign}[w^T \varphi(x) + b] = \operatorname{sign}\left[\sum_{i=1}^{N} \alpha_i K(x, x_i) + b\right] \tag{26}$$

with latent variable $z = \sum_{i=1}^{N} \alpha_i K(x, x_i) + b$. The support values α_i ($i = 1, \ldots, N$) in the dual classifier formulation determine the relative weight of each data point x_i in the classifier decision (26).

5. Bayesian interpretation and inference

The LS-SVM classifier formulation allows to estimate the classifier support values α and bias term b from the data $\mathcal{D}$, given the regularization parameter γ and the kernel function K, e.g., an RBF kernel with parameter σ. Together with the set of explanatory ratios/inputs $\mathcal{I} \subseteq \{1, \ldots, n\}$, the kernel function and its parameters define the model structure $\mathcal{M}$. These regularization and kernel parameters and input set need to be estimated from the data as well. This is achieved within the Bayesian evidence framework [12,13,20,21] that applies Bayes' formula on three levels of inference [20,21]:

$$\text{Posterior} = \frac{\text{Likelihood} \times \text{Prior}}{\text{Evidence}}. \tag{27}$$

(1) The primal and dual model parameters $\boldsymbol{w}$, b and α, b are inferred on the first level.
(2) The regularization parameter $\gamma = \zeta/\mu$ is inferred on the second level, where μ and ζ are additional parameters in the probabilistic inference.
(3) The parameter of the kernel function, e.g., σ, the (choice of) the kernel function K and the optimal input set are represented in the structural model description $\mathscr{M}$, which is inferred on level 3.

A schematic overview of the three levels of inference is depicted in Fig. 3, from which the hierarchical approach is observed in which the likelihood of level i is obtained from level $i-1$ ($i = 2, 3$). Given the least squares formulation, the model parameters are multivariate normal distributed allowing for analytic expressions [2] on all levels of inference. In each subsection, Bayes' formula is explained first, while practical expressions, computations and interpretations are given afterwards. All complex derivations are given in Appendix A.

5.1. Inference of model parameters (level 1)

5.1.1. Bayes' formula

Applying Bayes' formula on level 1, one obtains the posterior probability of the model parameters $\boldsymbol{w}$ and b:

$$\begin{aligned} \mathrm{p}(\boldsymbol{w}, b \mid \mathscr{D}, \log\mu, \log\zeta, \mathscr{M}) &= \frac{\mathrm{p}(\mathscr{D} \mid \boldsymbol{w}, b, \log\mu, \log\zeta, \mathscr{M})\mathrm{p}(\boldsymbol{w}, b \mid \log\mu, \log\zeta, \mathscr{M})}{\mathrm{p}(\mathscr{D} \mid \log\mu, \log\zeta, \mathscr{M})} \\ &\propto \mathrm{p}(\mathscr{D} \mid \boldsymbol{w}, b, \log\mu, \log\zeta, \mathscr{M})\mathrm{p}(\boldsymbol{w}, b \mid \log\mu, \log\zeta, \mathscr{M}), \end{aligned} \tag{28}$$

where the last step is obtained since the evidence $\mathrm{p}(\mathscr{D} \mid \log\mu, \log\zeta, \mathscr{M})$ is a normalizing constant that does not depend upon $\boldsymbol{w}$ and b.

For the prior, no correlation between $\boldsymbol{w}$ and b is assumed: $\mathrm{p}(\boldsymbol{w}, b \mid \log\mu, \mathscr{M}) = \mathrm{p}(\boldsymbol{w} \mid \log\mu, \mathscr{M})\mathrm{p}(b \mid \mathscr{M}) \propto \mathrm{p}(\boldsymbol{w} \mid \log\mu, \mathscr{M})$, with a multivariate Gaussian prior on $\boldsymbol{w}$ with zero mean and covariance matrix $\mu^{-1}\boldsymbol{I}_{n_\varphi}$ (n_φ being the dimension of the feature space) and an uninformative, flat prior on b:

$$\begin{aligned} &\mathrm{p}(\boldsymbol{w} \mid \log\mu, \mathscr{M}) = \left(\frac{\mu}{2\pi}\right)^{\frac{n_f}{2}} \exp\left(-\frac{\mu}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}\right), \\ &\mathrm{p}(b \mid \mathscr{M}) = \text{constant}. \end{aligned} \tag{29}$$

The uniform prior distribution on b can be approximated by a Gaussian distribution with standard deviation $\sigma_b \to \infty$. The prior states a belief that without any learning from data, the coefficients are zero with an uncertainty denoted by the variance $1/\mu$.

[2] Matlab implementations for the dual space expressions are available from http://www.esat.kuleuven.ac.be/sista/lssvmlab. Practical examples on classification with LS-SVMs are given in the demo `democlass.m`. For classification, the basic routines are `trainlssvm.m` for training by solving (25) and `simlssvm.m` for evaluating (26). For Bayesian learning the main routines are `bay_lssvm.m` for computation of the level 1, 2 and 3 cost functions (35), and (41), (51), respectively, `bay_optimize.m` for optimizing the hyperparameters with respect to the cost functions, `bay_lssvmARD.m` for input/ratio selection and `bay_modoutClass.m` for evaluation of the posterior class probabilities (58), (59). Initial estimates for the hyperparameters γ and σ^2 of, e.g., an LS-SVM with RBF-kernel, are obtained using `bay_initlssvm.m`. More details are found in the LS-SVMlab tutorial on the same website.

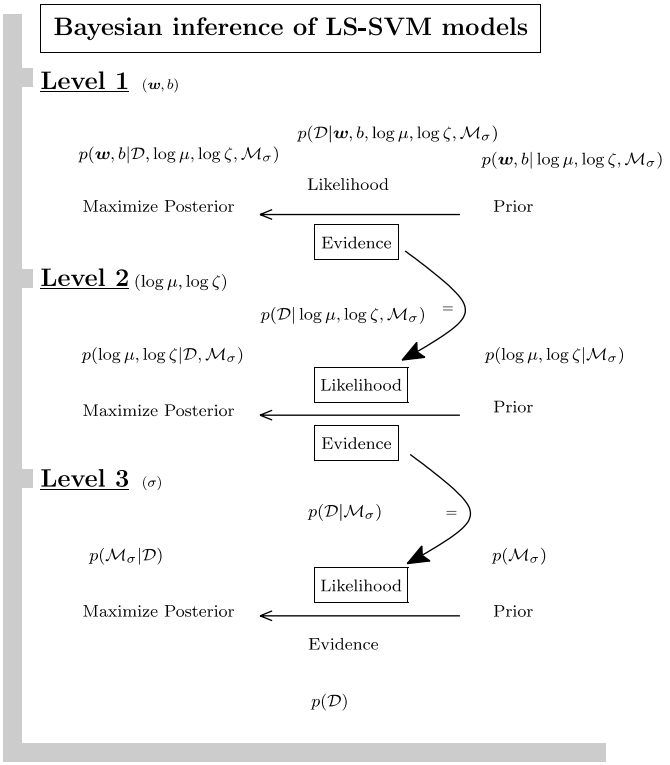


Fig. 3. Different levels of Bayesian inference. The posterior probability of the model parameters w and b is inferred from the data $\mathscr{D}$ by applying Bayes formula on the first level for given hyperparameters μ (prior) and ζ (likelihood) and the model structure $\mathscr{M}$. The model parameters are obtained by maximizing the posterior. The evidence on the first level becomes the likelihood on the second level when applying Bayes formula to infer μ and ζ (with $\gamma = \zeta/\mu$) from the given data $\mathscr{D}$. The optimal hyperparameters μ_{mp} and ζ_{mp} are obtained by maximizing the corresponding posterior on level 2. Model comparison is performed on the third level in order to compare different model structures, e.g., with different candidate input sets and/or different kernel parameters. The likelihood on the third level is equal to the evidence from level 2. Comparing different model structures $\mathscr{M}$, that model structure with the highest posterior probability is selected.

It is assumed that the data are independently identically distributed for expressing the *likelihood*

$$\mathrm{p}(\mathscr{D}|w, b, \log\zeta, \mathscr{M}) \propto \prod_{i=1}^{N} \mathrm{p}(y_i, x_i|w, b, \log\zeta, \mathscr{M}) \propto \prod_{i=1}^{N} \mathrm{p}(e_i|w, b, \log\zeta, \mathscr{M}) \propto \left(\frac{\zeta}{2\pi}\right)^{\frac{N}{2}} \exp\left(-\frac{\zeta}{2}\sum_{i=1}^{N} e_i^2\right), \tag{30}$$

where the last step is by assumption. This corresponds to the assumption that the z-score $w^{\mathrm{T}}\varphi(x) + b$ is Gaussian distributed around the targets $+1$ and -1.

Given that the prior (29) and likelihood (30) are multivariate normal distributions, the *posterior* (28) is a multivariate normal distribution [3] in $[w; b]$ with mean $[w_{\mathrm{mp}}; b_{\mathrm{mp}}] \in \mathbb{R}^{n_\varphi+1}$ and covariance matrix $Q \in \mathbb{R}^{(n_\varphi+1)\times(n_\varphi+1)}$. An alternative expression for the posterior is obtained by substituting (29) and (30) into (28). These approaches yield

[3] The notation $[x; y] = [x, y]^{\mathrm{T}}$ is used here.

$$\mathrm{p}(\boldsymbol{w}, b|\log\mu, \log\zeta, \mathscr{M}) = \sqrt{\frac{\det(\boldsymbol{Q}^{-1})}{(2\pi)^{n_\varphi+1}}}\exp\left(-\frac{1}{2}[\boldsymbol{w}-\boldsymbol{w}_{\mathrm{mp}}; b-b_{\mathrm{mp}}]\boldsymbol{Q}^{-1}[\boldsymbol{w}-\boldsymbol{w}_{\mathrm{mp}}; b-b_{\mathrm{mp}}]\right) \tag{31}$$

$$\propto \left(\frac{\mu}{2\pi}\right)^{\frac{n_f}{2}}\exp\left(-\frac{\mu}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}\right)\left(\frac{\zeta}{2\pi}\right)^{\frac{N}{2}}\exp\left(-\frac{\zeta}{2}\sum_{i=1}^{N}e_i^2\right), \tag{32}$$

respectively.

The *evidence* is a normalizing constant in (28) independent of $\boldsymbol{w}$ and b such that $\int\int\cdots\int \mathrm{p}(\boldsymbol{w}, b|\mathscr{D}, \log\mu, \log\zeta, \mathscr{M})\mathrm{d}w_1\cdots\mathrm{d}w_{n_\varphi}\mathrm{d}b = 1$. Substituting the expressions for the prior (29), likelihood (30) and posterior (32) into (28), one obtains

$$\mathrm{p}(\mathscr{D}|\log\mu, \log\zeta, \mathscr{M}) = \frac{\mathrm{p}(\boldsymbol{w}_{\mathrm{mp}}|\log\mu, \mathscr{M})\mathrm{p}(\mathscr{D}|\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}, \log\zeta, \mathscr{M})}{\mathrm{p}(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}|\mathscr{D}, \log\mu, \log\zeta, \mathscr{M})}. \tag{33}$$

5.1.2. Computation and interpretation

The model parameters with maximum posterior probability are obtained by minimizing the negative logarithm of (31) and (32):

$$(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}) = \arg\min_{\boldsymbol{w},b}\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}, b)$$

$$= \mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}) + \frac{1}{2}([\boldsymbol{w}-\boldsymbol{w}_{\mathrm{mp}}; b-b_{\mathrm{mp}}]\boldsymbol{Q}^{-1}[\boldsymbol{w}-\boldsymbol{w}_{\mathrm{mp}}; b-b_{\mathrm{mp}}]) \tag{34}$$

$$= \frac{\mu}{2}\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w} + \frac{\zeta}{2}\sum_{i=1}^{N}e_i^2, \tag{35}$$

where constants are neglected in the optimization problem. Both expressions yield the same optimization problem and the covariance matrix $\boldsymbol{Q}$ is equal to the inverse of the Hessian $\boldsymbol{H}$ of $\mathscr{J}_{\mathrm{P},1}$. The Hessian is expressed in terms of the matrix $\boldsymbol{\Phi} = [\boldsymbol{\varphi}(\boldsymbol{x}_1), \ldots \boldsymbol{\varphi}(\boldsymbol{x}_N)]^{\mathrm{T}}$ with regressors, as derived in the appendix.

Comparing (35) with (22), one obtains the same optimization problem for $\gamma = \zeta/\mu$ up to a constant scaling. The optimal $\boldsymbol{w}_{\mathrm{mp}}$ and b_{mp} are computed in the dual space from the linear KKT-system (25) with $\gamma = \zeta/\mu$ and the scoring function $z = \boldsymbol{w}_{\mathrm{mp}}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}) + b_{\mathrm{mp}}$ is expressed in terms of the dual parameters $\boldsymbol{\alpha}$ and bias term b_{mp} via (26).

Substituting (29), (30) and (32) into (33), one obtains

$$\mathrm{p}(\mathscr{D}|\log\mu, \log\zeta, \mathscr{M}) \propto \left(\frac{\mu^{n_\varphi}\zeta^N}{\det\boldsymbol{H}}\right)^{\frac{1}{2}}\exp(-\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}})). \tag{36}$$

As $\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}, b) = \mu\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}) + \zeta\mathscr{J}_{e}(\boldsymbol{w}, b)$, the evidence can be rewritten as

$$\underbrace{\mathrm{p}(\mathscr{D}|\log\mu, \log\zeta, \mathscr{M})}_{\text{evidence}} \propto \underbrace{\mathrm{p}(\mathscr{D}|\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}, \log\zeta, \mathscr{M})}_{\text{likelihood}|_{\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}}} \underbrace{\mathrm{p}(\boldsymbol{w}_{\mathrm{mp}}|\log\mu, \mathscr{M})(\det\boldsymbol{H})^{-1/2}}_{\text{Occam factor}}.$$

The model evidence consists of the likelihood of the data and an Occam factor that penalizes for too complex models. The Occam factor consists of the regularization term $0.5\boldsymbol{w}_{\mathrm{mp}}^{\mathrm{T}}\boldsymbol{w}_{\mathrm{mp}}$ and the ratio $(\mu^{n_\varphi}/\det\boldsymbol{H})^{1/2}$ which is a measure for the volume of the posterior probability divided by the volume of the prior probability. Strong contractions of the posterior versus prior space indicates too many free parameters and, hence, overfitting on the training data. The evidence will be maximized on level 2, where also dual space expressions are derived.

5.2. Inference of hyper-parameters (level 2)

5.2.1. Bayes' formula

The optimal regularization parameters μ and ζ are inferred from the given data $\mathscr{D}$ by applying Bayes' rule on the second level [20,21]:

$$\mathrm{p}(\log\mu,\log\zeta|\mathscr{D},\mathscr{M}) = \frac{\mathrm{p}(\mathscr{D}|\log\mu,\log\zeta,\mathscr{M})\mathrm{p}(\log\mu,\log\zeta)}{\mathrm{p}(\mathscr{D}|\mathscr{M})}. \tag{37}$$

The prior $\mathrm{p}(\log\mu,\log\zeta|\mathscr{M}) = \mathrm{p}(\log\mu|\mathscr{M})\mathrm{p}(\log\zeta|\mathscr{M}) = \text{constant}$ is taken to be a flat uninformative prior ($\sigma_{\log\mu}, \sigma_{\log\zeta} \to \infty$). The level 2 *likelihood* $\mathrm{p}(\mathscr{D}|\log\mu,\log\zeta,\mathscr{M})$ is equal to the level 1 evidence (36). In this way, Bayesian inference implicitly embodies Occam's razor: on level 2 the evidence of the level 1 is optimized so as to find a trade-off between the model fit and a complexity term to avoid overfitting [12,13]. The level 2 *evidence* is obtained in a similar way as on level 1 as the likelihood for the maximum a posteriori times the ratio of the volume of the posterior probability and the volume of the prior probability:

$$\mathrm{p}(\mathscr{D}|\mathscr{M}) \simeq \mathrm{p}(\mathscr{D}|\log\mu_{\mathrm{mp}},\log\zeta_{\mathrm{mp}},\mathscr{M})\frac{\sigma_{\log\mu|\mathscr{D}}\sigma_{\log\zeta|\mathscr{D}}}{\sigma_{\log\mu}\sigma_{\log\zeta}}, \tag{38}$$

where one typically approximates the posterior probability by a multivariate normal probability function with diagonal covariance matrix $\mathrm{diag}([\sigma^2_{\log\mu|\mathscr{D}},\sigma^2_{\log\mu|\mathscr{D}}]) \in \mathbb{R}^{2\times 2}$.

Neglecting all constants, Bayes' formula (37) becomes

$$\mathrm{p}(\log\mu,\log\zeta|\mathscr{D},\mathscr{M}) \propto \mathrm{p}(\mathscr{D}|\log\mu,\log\zeta,\mathscr{M}), \tag{39}$$

where the expressions for the level 1 evidence are given by (33) and (36).

5.2.2. Computation and interpretation

In the primal space, the hyperparameters are obtained by minimizing the negative logarithm of (36) and (39):

$$(\mu_{\mathrm{mp}},\zeta_{\mathrm{mp}}) = \arg\min_{\mu,\zeta} \mathscr{J}_{\mathrm{P},2}(\mu,\zeta) = \mu\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \zeta\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}) + \frac{1}{2}\log\det\boldsymbol{H} - \frac{n_\varphi}{2}\log\mu - \frac{N}{2}\log\zeta. \tag{40}$$

Observe that in order to evaluate (40) one needs also to calculate $\boldsymbol{w}_{\mathrm{mp}}$ and b_{mp} for the given μ and ζ and evaluate the level 1 cost function.

The determinant of $\boldsymbol{H}$ is equal to (see Appendix A for details)

$$\det(\boldsymbol{H}) = (\zeta N)\det(\mu\boldsymbol{I}_{n_\varphi} + \zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi}),$$

with the idempotent centering matrix $\boldsymbol{M}_c = \boldsymbol{I}_N - 1/N\boldsymbol{1}\boldsymbol{1}^{\mathrm{T}} = \boldsymbol{M}_c^2 \in \mathbb{R}^{N\times N}$. The determinant is also equal to the product of the eigenvalues. The n_e non-zero eigenvalues $\lambda_1,\ldots,\lambda_{n_e}$ of $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi}$ are equal to the n_e non-zero eigenvalues of $\boldsymbol{M}_c\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c = \boldsymbol{M}_c\boldsymbol{\Omega}\boldsymbol{M}_c \in \mathbb{R}^{N\times N}$, which can be calculated in the dual space. Substituting the determinant $\det(\boldsymbol{H}) = \zeta N\mu^{n_\varphi - n_e}\prod_{i=1}^{n_e}(\mu+\zeta\lambda_i)$ into (40), one obtains the optimization problem in the dual space

$$\mathscr{J}_{\mathrm{D},2}(\mu,\zeta) = \mu\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \zeta\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}) \frac{1}{2}\sum_{i=1}^{n_e}\log(\mu+\zeta\lambda_i) - \frac{n_e}{2}\log\mu - \frac{n_e-1}{2}\log\zeta, \tag{41}$$

where it can be shown by matrix algebra that $\mu\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \zeta\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}) = \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{M}_c\left(\frac{1}{\mu}\boldsymbol{M}_c\boldsymbol{\Omega}\boldsymbol{M}_c + \frac{1}{\zeta}\boldsymbol{I}_N\right)^{-1}\boldsymbol{M}_c\boldsymbol{y}$.

An important concept in neural networks and Bayesian learning in general is the *effective number of parameters*. Although there are $n_\varphi + 1$ free parameters $w_1,\ldots,w_{n_\varphi}, b$ in the primal space, the use of these

parameters (35) is restricted by the use of the regularization term $0.5\boldsymbol{w}^{\mathrm{T}}\boldsymbol{w}$. The effective number of parameters d_{eff} is equal to $d_{\mathrm{eff}} = \sum_i \lambda_{i,u}/\lambda_{i,r}$, where $\lambda_{i,u}$, $\lambda_{i,r}$ denote the eigenvalues of the Hessian of the unregularized cost function $\mathscr{J}_{1,u} = \zeta E_D$ and the regularized cost function $\mathscr{J}_{1,r} = \mu E_W + \zeta E_D$ [4,12]. For LS-SVMs, the effective number of parameters is equal to

$$d_{\mathrm{eff}} = 1 + \sum_{i=1}^{n_e} \frac{\zeta\lambda_i}{\mu + \zeta\lambda_i} = 1 + \sum_{i=1}^{n_e} \frac{\gamma\lambda_i}{1 + \gamma\lambda_i}, \tag{42}$$

with $\gamma = \zeta/\mu \in \mathbb{R}^+$. The term $+1$ appears because no regularization is applied on the bias term b. As shown in the appendix, one has that $n_e \leqslant N - 1$ and, hence, also that $d_{\mathrm{eff}} \leqslant N$, even in the case of high dimensional feature spaces.

The conditions for optimality for (41) are obtained by putting $\partial\mathscr{J}_2/\partial\mu = \partial\mathscr{J}_2/\partial\zeta = 0$. One obtains [4]

$$\partial\mathscr{J}_2/\partial\mu = 0 \rightarrow 2\mu_{\mathrm{mp}}\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}};\mu_{\mathrm{mp}},\zeta_{\mathrm{mp}}) = d_{\mathrm{eff}}(\mu_{\mathrm{mp}},\zeta_{\mathrm{mp}}) - 1, \tag{43}$$

$$\partial\mathscr{J}_2/\partial\zeta = 0 \rightarrow 2\zeta_{\mathrm{mp}}\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}};\mu_{\mathrm{mp}},\zeta_{\mathrm{mp}}) = N - d_{\mathrm{eff}}, \tag{44}$$

where the latter equation corresponds to the unbiased estimate of the noise variance $1/\zeta_{\mathrm{mp}} = \frac{1}{2}\sum_{i=1}^{N} e_i^2/(N - d_{\mathrm{eff}})$.

Instead of solving the optimization problem in μ and ζ, one may also reformulate (41) using (43), (44) in terms of $\gamma = \zeta/\mu$ and solve the following scalar optimization problem:

$$\min_{\gamma} \sum_{i=1}^{N-1} \log\left(\lambda_i + \frac{1}{\gamma}\right) + (N-1)\log(\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \gamma\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}})) \tag{45}$$

with

$$\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}) = \frac{1}{2\gamma^2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{V}(\boldsymbol{\Lambda} + \boldsymbol{I}_N/\gamma)^{-2}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{y}, \tag{46}$$

$$\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) = \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{V}\boldsymbol{\Lambda}(\boldsymbol{\Lambda} + \boldsymbol{I}/\gamma)^{-2}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{y}, \tag{47}$$

$$\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \gamma\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}) = \frac{1}{2}\boldsymbol{y}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{V}(\boldsymbol{\Lambda} + \boldsymbol{I}_N/\gamma)^{-1}\boldsymbol{V}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{y} \tag{48}$$

with the eigenvalue decomposition $\boldsymbol{M}_c\boldsymbol{\Omega}\boldsymbol{M}_c = \boldsymbol{V}^{\mathrm{T}}\boldsymbol{\Lambda}\boldsymbol{V}$. Given the optimal γ_{mp} from (45) one finds the effective number of parameters d_{eff} from $d_{\mathrm{eff}} = 1 + \sum_{i=1}^{n_e}\gamma\lambda_i/(1 + \gamma\lambda_i)$. The optimal μ_{mp} and ζ_{mp} are obtained from $\mu_{\mathrm{mp}} = (d_{\mathrm{eff}} - 1)/(2\mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}))$ and $\zeta_{\mathrm{mp}} = (N - d_{\mathrm{eff}})/(2\mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}))$.

5.3. Model comparison (level 3)

5.3.1. Bayes' formula

The model structure $\mathscr{M}$ of the model determines the remaining parameters of the kernel based model: the selected kernel function (linear, RBF, etc.), the kernel parameter (RBF kernel parameter σ) and selected explanatory inputs. The model structure is inferred on level 3.

Consider, e.g., the inference of the RBF-kernel parameter σ, where the model structure is denoted by $\mathscr{M}_\sigma$. Bayes' formula for the inference of $\mathscr{M}_\sigma$ is equal to

[4] In this derivation, one uses that

$$\partial(\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}))/\partial\mu = \delta(\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}))/\delta\mu + \delta(\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}))/\delta[\boldsymbol{w};b]|_{[\boldsymbol{w}_{\mathrm{mp}};b_{\mathrm{mp}}]} \times \delta([\boldsymbol{w}_{\mathrm{mp}};b_{\mathrm{mp}}])/\delta\mu = \mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}),$$

since $\delta(\mathscr{J}_{\mathrm{P},1}(\boldsymbol{w}_{\mathrm{mp}},b_{\mathrm{mp}}))/\delta[\boldsymbol{w};b]|_{[\boldsymbol{w}_{\mathrm{mp}};b_{\mathrm{mp}}]} = 0$ [13,16,31].

$$p(\mathscr{M}_\sigma|\mathscr{D}) \propto p(\mathscr{D}|\mathscr{M}_\sigma)p(\mathscr{M}_\sigma), \tag{49}$$

where no evidence $p(\mathscr{D})$ is used in the expression on level 3 as it is in practice impossible to integrate over all model structures. The prior probability $p(\mathscr{M}_\sigma)$ is assumed to be constant. The likelihood is equal to the level 2 evidence (38).

5.3.2. Computation and interpretation

Substituting the evidence (38) into (49) and taking into account the constant prior, the Bayes' rule (38) becomes

$$p(\mathscr{M}|\mathscr{D}) \simeq p(\mathscr{D}|\log\mu_{mp}, \log\zeta_{mp}, \mathscr{M}) \frac{\sigma_{\log\mu|\mathscr{D}}\sigma_{\log\zeta|\mathscr{D}}}{\sigma_{\log\mu}\sigma_{\log\zeta}}. \tag{50}$$

As uninformative priors are used on level 2, the standard deviations $\sigma_{\log\mu}$ and $\sigma_{\log\zeta}$ of the prior distribution both tend to infinity and are omitted in the comparisons of different models in (50). The posterior error bars can be approximated analytically as $\sigma^2_{\log\mu|\mathscr{D}} \simeq 2/(d_{\text{eff}} - 1)$ and $\sigma^2_{\log\zeta|\mathscr{D}} \simeq 2/(N - d_{\text{eff}})$, respectively [13]. The level 3 posterior becomes

$$p(\mathscr{M}_\sigma|\mathscr{D}) \simeq p(\mathscr{D}|\log\mu_{mp}, \log\zeta_{mp}, \mathscr{M}_\sigma) \frac{\sigma_{\log\mu|\mathscr{D}}\sigma_{\log\zeta|\mathscr{D}}}{\sigma_{\log\mu}\sigma_{\log\zeta}} \propto \sqrt{\frac{\mu_{mp}^{n_e}\zeta_{mp}^{N-1}}{(d_{\text{eff}} - 1)(N - d_{\text{eff}})\prod_{i=1}^{n_e}(\mu_{mp} + \zeta_{mp}\lambda_i)}}, \tag{51}$$

where all expressions can be calculated in the dual space. A practical way to infer the kernel parameter σ is to calculate (51) for a grid of possible kernel parameters $\sigma_1, \ldots, \sigma_m$ and to compare the corresponding posterior model parameters $p(\mathscr{M}_{\sigma_1}|\mathscr{D}), \ldots, p(\mathscr{M}_{\sigma_m}|\mathscr{D})$. An additional observation is that the RBF-LS-SVM classifier may not always yield a monotonic relation between the evolution of the ratio (e.g., solvency ratio) and the default risk. This is due to the nonlinearity of the classifier and/or multivariate correlations. In case monotonous relations are important, one may choose to use a combined kernel function $K(\boldsymbol{x}_1, \boldsymbol{x}_2) = \kappa K_{\text{lin}}(\boldsymbol{x}_1, \boldsymbol{x}_2) + (1 - \kappa)K_{\text{RBF}}(\boldsymbol{x}_1, \boldsymbol{x}_2)$, where the parameter $\kappa \in [0, 1]$ can be determined on level 3. In this paper, the use of an RBF-kernel is illustrated.

Model comparison is also used to infer the set of most relevant inputs [21] out of the given set of candidate explanatory variables by making pairwise comparisons of models with different input sets. In a backward input selection procedure, one starts from the full candidate input set and removes in each input pruning step that input that yields the best model improvement (or smallest decrease) in terms of the model probability (51). The procedure is stopped when no significant decrease of the model probability is observed. In the case of equal prior model probabilities $p(\mathscr{M}_i) = p(\mathscr{M}_j)$ $(\forall i, j)$ the models $\mathscr{M}_i$ and $\mathscr{M}_j$ are compared according to their Bayes factor

$$\mathscr{B}_{ij} = \frac{p(\mathscr{D}|\mathscr{M}_i)}{p(\mathscr{D}|\mathscr{M}_j)} = \frac{p(\mathscr{D}|\log\mu_i, \log\zeta_i, \mathscr{M}_i)}{p(\mathscr{D}|\log\mu_j, \log\zeta_j, \mathscr{M}_j)} \frac{\sigma_{\log\mu_i|\mathscr{D}}\sigma_{\log\zeta_i|\mathscr{D}}}{\sigma_{\log\mu_j|\mathscr{D}}\sigma_{\log\zeta_j|\mathscr{D}}}. \tag{52}$$

According to [39], one uses the values in Table 1 in order to report and interpret the significance of model $\mathscr{M}_i$ improving on model $\mathscr{M}_j$.

5.4. Moderated output of the classifier

5.4.1. Moderated output

Based on the Bayesian interpretation, an expression is derived for the likelihood $p(\boldsymbol{x}|y, \boldsymbol{w}, b, \zeta, \mathscr{M})$ of observing $\boldsymbol{x}$ given the class label y and the parameters $\boldsymbol{w}, b, \zeta, \mathscr{M}$. However, the parameters [5] $\boldsymbol{w}$ and b are multivariate normal distributed. Hence, the moderated likelihood is obtained as

[5] The uncertainty on ζ only has a minor influence in a limited number of directions [13] and is neglected.

Table 1
Evidence against H_0 (no improvement of $\mathscr{M}_i$ over $\mathscr{M}_j$) for different values of the Bayes factor $\mathscr{B}_{ij}$ [39]

$2\ln \mathscr{B}_{ij}$	$\mathscr{B}_{ij}$	Evidence against H_0
0–2	1–3	Not worth more than a bare mention
2–5	3–12	Positive
5–10	12–150	Strong
>10	>150	Decisive

$$\mathrm{p}(\boldsymbol{x}|y,\zeta,\mathscr{M}) = \int \mathrm{p}(\boldsymbol{x}|y,\boldsymbol{w},b,\zeta,\mathscr{M})\mathrm{p}(\boldsymbol{w},b|y,\mu,\zeta,\mathscr{M})\mathrm{d}w_1\cdots\mathrm{d}w_{n_\varphi}\mathrm{d}b. \tag{53}$$

This expression will then be used in Bayes' rule (3).

5.4.2. Computation and interpretation

In the level 1 formulation, it was assumed that the errors e are normally distributed around the targets ± 1 with variance ζ^{-1} , i.e.,

$$\mathrm{p}(\boldsymbol{x}|y=+1,\boldsymbol{w},b,\zeta,\mathscr{M}) = (2\pi/\zeta)^{-1/2}\exp(-1/2\zeta e_+^2), \tag{54}$$

$$\mathrm{p}(\boldsymbol{x}|y=-1,\boldsymbol{w},b,\zeta,\mathscr{M}) = (2\pi/\zeta)^{-1/2}\exp(-1/2\zeta e_-^2), \tag{55}$$

with $e_+ = +1 - (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}) + b)$ and $e_- = -1 - (\boldsymbol{w}^{\mathrm{T}}\boldsymbol{\varphi}(\boldsymbol{x}) + b)$, respectively. The assumption that the mean z-scores per class are equal to $+1$ and -1 will be relaxed and for the calculation of the moderated output, it is assumed that the scores z are normally distributed with centers t_+ (Class $+1$) and t_- (Class -1) [20]. Defining the Boolean vectors $\mathbf{1}_+ = [y_i = +1] \in \mathbb{R}^N$ and $\mathbf{1}_- = [y_i = -1] \in \mathbb{R}^N$, with elements 0 and 1 whether the observation i is an element of $\mathscr{C}_-$ and $\mathscr{C}_+$ for $\mathbf{1}_+$ and vice versa for $\mathbf{1}_-$. The centers are estimated as $t_+ = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_{\varphi_+} + b$ and $t_- = \boldsymbol{w}^{\mathrm{T}}\boldsymbol{m}_{\varphi_-} + b$ with the feature vector class means $\boldsymbol{m}_{\varphi,+} = 1/N_+\sum_{y_i=+1}\boldsymbol{\varphi}(\boldsymbol{x}_i) = 1/N_+\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}_+$ and $\boldsymbol{m}_{\varphi,-} = 1/N_-\sum_{y_i=-1}\boldsymbol{\varphi}(\boldsymbol{x}_i) = 1/N_-\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}_-$. The variances are denoted by $1/\zeta_+$ and $1/\zeta_-$, respectively, and represent the uncertainty around the projected class centers t_+ and t_-. It is typically assumed that $\zeta_+ = \zeta_- = \zeta_\pm$.

The parameters $\boldsymbol{w}$ and b are estimated from the data with resulting probability density function (31). Due to the uncertainty on $\boldsymbol{w}$ (and b), the errors e_+ and e_- have expected value [6]

$$\hat{e}_\bullet = \boldsymbol{w}_{\mathrm{mp}}^{\mathrm{T}}(\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{m}_{\varphi\bullet}) = \sum_{i=1}^{N} K(\boldsymbol{x},\boldsymbol{x}_i) - \hat{t}_\bullet,$$

where $\hat{t}_\bullet = \boldsymbol{w}_{\mathrm{mp}}^{\mathrm{T}}\boldsymbol{m}_{\varphi\bullet}$ is obtained in the dual space as $\hat{t}_\bullet = 1/N_\bullet\boldsymbol{\alpha}^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{1}_\bullet$. The expression for the variance is

$$\sigma_{e\bullet}^2 = [\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{m}_{\varphi\bullet}]^{\mathrm{T}}\boldsymbol{Q}_{11}[\boldsymbol{\varphi}(\boldsymbol{x}) - \boldsymbol{m}_{\varphi\bullet}]. \tag{56}$$

The dual formulations for the variance are derived in the appendix based on the singular value decomposition (A.7) of $\boldsymbol{Q}_{11}$ and is equal to

$$\begin{aligned}\sigma_{e\bullet}^2 = {} & \frac{1}{\mu}K(\boldsymbol{x},\boldsymbol{x}) - \frac{2}{\mu N_\bullet}\boldsymbol{\theta}(\boldsymbol{x})^{\mathrm{T}}\mathbf{1}_\bullet + \frac{1}{\mu N_\bullet^2}\mathbf{1}_\bullet^{\mathrm{T}}\boldsymbol{\Omega}\mathbf{1}_\bullet - \frac{\zeta}{\mu}\left(\boldsymbol{\theta}(\boldsymbol{x}) - \frac{1}{N_+}\mathbf{1}_\bullet^{\mathrm{T}}\right)^{\mathrm{T}} \\ & \times \boldsymbol{M}_c(\mu\boldsymbol{I}_N + \zeta\boldsymbol{M}_c\boldsymbol{\Omega}\boldsymbol{M}_c)^{-1}\boldsymbol{M}_c\left(\boldsymbol{\theta}(\boldsymbol{x}) - \frac{1}{N_\bullet}\mathbf{1}_\bullet^{\mathrm{T}}\boldsymbol{\Omega}\right),\end{aligned} \tag{57}$$

with $\bullet$ either $+$ or $-$. The vector $\boldsymbol{\theta}(\boldsymbol{x}) \in \mathbb{R}^N$ has elements $\theta_i(\boldsymbol{x}) = K(\boldsymbol{x},\boldsymbol{x}_i)$.

[6] The $\bullet$ notation is used to denote either $+$ or $-$, since analogous expressions are obtained for classes $\mathscr{C}_+$ and $\mathscr{C}_-$, respectively.

Applying Bayes' formula, the posterior class probability of the LS-SVM classifier is obtained

$$\mathrm{p}(y|\boldsymbol{x},\mathscr{D},\mathscr{M}) = \frac{\mathrm{p}(y)\mathrm{p}(\boldsymbol{x}|y,\mathscr{D},\mathscr{M})}{\mathrm{P}(y=+1)\mathrm{p}(\boldsymbol{x}|y=+1,\mathscr{D},\mathscr{M}) + \mathrm{P}(y=-1)\mathrm{p}(\boldsymbol{x}|y=-1,\mathscr{D},\mathscr{M})},$$

where we omitted the hyperparameters μ, ζ, $\zeta_\pm$ for notational convenience. Approximate analytic expressions exist for marginalizing over the hyperparameters, but can be neglected in practice as the additional variance is rather small [13].

The moderated likelihood (53) is then equal to

$$\mathrm{p}(\boldsymbol{x}|y=\bullet 1,\zeta,\mathscr{M}) = (2\pi/(\zeta_\pm + \sigma_{e_\bullet}^2))^{-1/2}\exp(-1/2\hat{e}_\bullet^2/(\zeta_\pm^{-1}+\sigma_{e_\bullet}^2)). \tag{58}$$

Substituting (58) into the Bayesian decision rule (3), one obtains a quadratic decision rule as the class variances $\zeta_\pm^{-1}+\sigma_{e-}^2$ and $\zeta_\pm^{-1}+\sigma_{e+}^2$ are not equal. Assuming that $\sigma_{e+}^2 \simeq \sigma_{e-}^2$ and defining $\sigma_e = \sqrt{\sigma_{e+}\sigma_{e-}}$, the Bayesian decision rule becomes

$$\hat{y} = \mathrm{sign}\left[\frac{1}{\mu}\sum_{i=1}^{N}\alpha_i K(\boldsymbol{x},\boldsymbol{x}_i) - \frac{m_{d+}+m_{d-}}{2} + \frac{\zeta_\pm^{-1}+\sigma_e^2(\boldsymbol{x})}{m_{d+}-m_{d-}}\log\frac{\mathrm{P}(y=+1)}{\mathrm{P}(y=-1)}\right]. \tag{59}$$

The variance $\zeta_\pm^{-1} = \sum_{i=1}^{N} e_{\pm,i}^2/(N-d_{\mathrm{eff}})$ is estimated in the same way as ζ_{mp} on level 2.

The prior probabilities $\mathrm{P}(y=+1)$ and $\mathrm{P}(y=-1)$ are typically estimated as $\hat{\pi}_+ = N_+/(N_+ + N_-)$ and $\hat{\pi}_- = N_-/(N_+ + N_-)$, but can also be adjusted to reject a given percentage of applicants or to optimize the total profit taking into account misclassification costs. As (59) depends explicitly on the prior probabilities, it also allows to make point-in-time credit decisions where the default probabilities and recovery rates depend upon the point in the business cycle. Difficult cases having almost equal posterior class probabilities $\mathrm{P}(y=+1|\boldsymbol{x},\mathscr{D},\mathscr{M}) \simeq \mathrm{P}(y=-1|\boldsymbol{x},\mathscr{D},\mathscr{M})$ can be decided to not being automatically processed and to being referred to a human expert for further investigation.

5.5. Bayesian classifier design

Based on the previous theory, the following practical design scheme to design the LS-SVM classifier in the Bayesian framework is suggested:

(1) Preprocess the data by completing missing values and handling outliers. Standardize the inputs to zero mean and unit variance.
(2) Define models $\mathscr{M}_i$ by choosing a candidate input set $\mathscr{I}_i$, a kernel function K_i and kernel parameter, e.g., σ_i in the RBF kernel case. For all models $\mathscr{M}_i$, with $i = 1,\ldots,n_{\mathscr{M}}$ (with $n_{\mathscr{M}}$ the number of models to be compared), compute the level 3 posterior:
   (a) Find the optimal hyperparameters μ_{mp} and ζ_{mp} by solving the scalar optimization problem (45) in $\gamma = \zeta/\mu$ related to maximizing the level 2 posterior. [7] With the resulting γ_{mp}, compute the effective number of parameters, the hyperparameters μ_{mp} and ζ_{mp}.
   (b) Evaluate the level 3 posterior (51) for model comparison.
(3) Select the model $\mathscr{M}_i$ with maximal evidence. If desired, refine the model tuning parameters $K_i, \sigma_i, \mathscr{I}_i$ to further optimize the classifier and go back to Step 2; else: go to step 4.
(4) Given the optimal $\mathscr{M}_i^\star$, calculate $\boldsymbol{\alpha}$ and b from (25), with kernel K_i, parameter σ_i and input set $\mathscr{I}_i$. Calculate $\zeta_\star$ and select $\hat{\pi}_+$ and $\hat{\pi}_-$ to evaluate (59).

For illustrative purposes, the design scheme is illustrated for a kernel function with one parameter σ like the RBF-kernel. The design scheme is easily extended to other kernel functions or combinations of kernel functions.

[7] Observe that this implies in each iteration step maximizing the level 1 posterior in $\boldsymbol{w}$ and b.

6. Financial distress prediction for mid-cap firms in the benelux

6.1. Data set description

The bankruptcy data, obtained from a major Benelux financial institution, were used to build an internal rating system [40] for firms with middle-market capitalization (mid-cap firms) in the Benelux countries (Belgium, The Netherlands, Luxembourg) using linear modelling techniques. Firms in the mid-cap segment are defined as follows: they are not stocklisted, the book value of their total assets exceeds 10 mln euro, and they generate a turnover that is smaller than 0.25 bln euro. Note that more advanced methods like option based valuation models are not applicable since these companies are not listed. Together with small and medium enterprises, mid-cap firms represent a large proportion of the economy in the Benelux. The mid-cap market segment is especially important as it reflects an important business orientation of the bank.

The data set consists of $N = 422$ observations, $n_{\mathscr{D}}^{-} = 74$ bankrupt and $n_{\mathscr{D}}^{+} = 348$ solvent companies. The data on the bankrupt firms were collected from 1991 to 1997, while the other data were extracted from the period 1997 only (for reasons of data retrieval difficulties). One out of five non-bankrupt observations of the 1997 database was used to train the model. Observe that a larger sample of solvent firms could have been selected, but involves training on an even more unbalanced [8] training set. A total number of 40 candidate input variables was selected from financial statement data, using standard liquidity, profitability and solvency measures. As can be seen from Table 2, both ratios as well as trends of ratios are considered.

The data were preprocessed as follows. Median imputation was applied to missing values. Outliers outside the interval $[\hat{m} - 2.5 \times s, \hat{m} + 2.5 \times s]$ were put equal to the upper limit and lower limit, respectively; where $\hat{m}$ is the sample mean and s the sample standard deviation. A similar procedure is, e.g., used in the calculation of the Winsorized mean [41]. The log transformation was applied to size variables.

6.2. Performance measures

The performance of all classifiers will be quantified using both the classification accuracy and the area under the receiver operating characteristic curve (AUROC). The classification accuracy simply measures the percentage of correctly classified (PCC) observations. Two closely related performance measures are the sensitivity which is the percentage of positive observations being classified as positive (PCC_p) and the specificity which is the percentage of negative observations being classified as negative (PCC_n). The receiver operating characteristic curve (ROC) is a two-dimensional graphical illustration of the sensitivity on the y-axis versus 1-specificity on the x-axis for various values of the classifier threshold [42]. It basically illustrates the behaviour of a classifier without regard to class distribution or misclassification cost. The AUROC then provides a simple figure-of-merit for the performance of the constructed classifier. We will use McNemar's test to compare the PCC, PCC_p and PCC_n of different classifiers [43] and the test of De Long et al. [44] to compare the AUROCs. The ROC curve is also closely related to the Cumulative Accuracy Profile which is in turn related to the power statistic and Gini-coefficient [45].

6.3. Models with full candidate input set

The Bayesian framework was applied to infer the hyper- and kernel parameters. The kernel parameter σ of the RBF kernel [9] was inferred on level 3 by selecting the parameter from the grid $\sqrt{n} \times [0.1, 0.5, 1, 1.2, 1.5, 2, 3, 4, 10]$. For each of these bandwidth parameters, the kernel matrix was constructed and its

[8] In practice, one typically observes that the percentage of defaults in training databases varies from 50% to about 70% or 80% [29].

[9] The use of an RBF-kernel is illustrated here because of its consistently good performance on 20 benchmark data sets [31]. The other kernel functions can be applied in a similar way.

Table 2
Benelux data set: description of the 40 candidate inputs

Input variable description	LDA	LOGIT	LS-SVM
L: Current ratio (R)	36	1	23
L: Current ratio (Tr)	34	27	28
L: Quick ratio (R)	22	26	24
L: Quick ratio (Tr)	35	30	29
L: Numbers of days to customer credit (R)	29	19	11
L: Numbers of days to customer credit (Tr)	6	14	19
L: Numbers of days of supplier credit (R)	21	21	27
L: Numbers of days of supplier credit (Tr)	25	33	21
S:Capital and reserves (% TA)	5	5	2
S: Capital and reserves (Tr)	20	18	35
S: Financial debt payable after one year (% TA)	37	37	31
S: Financial debt payable after one year (Tr)	40	39	8
S: Financial debt payable within one year (% TA)	38	38	18
S: Financial debt payable within one year (Tr)	39	40	17
S: Solvency Ratio (%)(R)	3	2	1
S: Solvency Ratio (%)(Tr)	14	16	10
P: Turnover (% TA)	2	4	5
P: Turnover (Trend)	19	12	32
P: Added value (% TA)	18	28	13
P: Added value (Tr)	24	36	40
V: Total assets (Log)	4	6	3
P: Total assets (Tr)	7	11	20
P: Current profit/current loss before taxes (R)	28	25	38
P: Current profit/current loss before taxes (Tr)	33	31	30
P: Gross operation margin (%)(R)	32	3	25
P: Gross operation margin (%)(Tr)	15	23	7
P: Current profit/current loss (R)	27	35	36
P: Current profit/current loss (Tr)	30	34	37
P: Net operation margin (%)(R)	31	20	26
P: Net operation margin (%)(Tr)	26	32	15
P: Added value/sales (%)(R)	13	17	6
P: Added value/sales (%)(Tr)	10	9	9
P: Added value/pers. employed (R)	23	29	39
P: Added value/pers. employed (Tr)	17	10	34
P: Cash-flow/equity (%)(R)	16	8	33
P: Cash-flow/equity (%)(Tr)	11	24	14
P: Return on equity (%)(R)	8	7	4
P: Return on equity (%)(Tr)	9	22	12
P: Net return on total assets before taxes and debt charges (%)(R)	1	13	16
P: Net return on total assets before taxes and debt charges (%)(Tr)	12	15	22

The inputs include various liquidity (L), solvency (S), profitability (P) and size (V) measures. Trends (Tr) are used to describe the evolution of the ratios (R). The results of backward input selection are presented by reporting the number of remaining inputs in the LDA, LOGIT and LS-SVM model when an input is removed. These ranking numbers are underlined when the corresponding input is used in the model having optimal leave-one-out cross-validation performance. Hence, inputs with low importance have a high number, while the most important input has rank 1.

eigenvalue decomposition computed. The optimal hyperparameter γ was determined from the scalar optimization problem (45) and then, μ, ζ, d_{eff} and the level 3 cost were calculated. As the number of default data is low, no separate test data set was used. The generalization performance is assessed by means of the leave-one-out cross-validation error, which is a common measure in the bankruptcy prediction literature [22]. In Table 3, we have contrasted the PCC, PCC_p, PCC_n and AUROC performance of the LS-SVM (26) and the

Table 3
Leave-one-out classification performances (percentages) for the LDA, Logit and LS-SVM model using the full candidate input set

	LDA	LOGIT	LS-SVM	LS-SVM$_{Bay}$
PCC	84.83% (0.13%)	85.78% (6.33%)	88.39% (100%)	88.39% (100%)
PCC$_p$	95.98% (0.77%)	93.97% (0.02%)	98.56% (100%)	98.56% (100%)
PCC$_n$	32.43% (0.01%)	47.30% (100%)	40.54% (26.7%)	40.54% (26.7%)
AUROC	79.51% (0.02%)	80.07% (0.36%)	86.58% (43.27%)	86.65% (100%)

The corresponding p-values (percentages) are denoted in parentheses.

Bayesian LS-SVM decision rule (59) classifier with the performance of the linear LDA and Logit classifiers. The numbers between brackets represent the p-values of the tests between each classifier and the classifier scoring best on the particular performance measure. It is easily observed that both the LS-SVM and LS-SVM$_{Bay}$ classifiers yield very good performances when compared to the LDA and Logit classifiers. The corresponding ROC curves are depicted in the left pane of Fig. 4.

6.4. Models with optimized input set

Given the models with full candidate input set, a backward input selection procedure is applied to infer the most relevant inputs from the data. For the LDA and Logit classifiers, each time the input i was removed for which the coefficient had the highest p-value to test whether the coefficient is significantly different from zero. The procedure was stopped when all coefficients were significantly different from zero at the 1% level. A backward input selection procedure was applied with the LS-SVM model, computing each time the model probability (on level 3) with one of the inputs removed. The input that yielded the best decrease (or smallest increase) in the level 3 cost function was then selected. The procedure was stopped just before the difference with the optimal model became decisive according to Table 1. In order to reduce the numbers of inputs as much as possible, but still retain a liquidity ratio in the model, 11 inputs are selected, which is one before the limit of becoming decisively different. The level 3 cost function and the corresponding leave-one-out PCC are depicted in Fig. 5 with respect to the number of removed inputs. Notice the similarities between both curves during the input removal process. Table 4 reports the performances of all classifiers using the optimally pruned set of inputs. Again it can be observed that the LS-SVM and LS-SVM$_{Bay}$

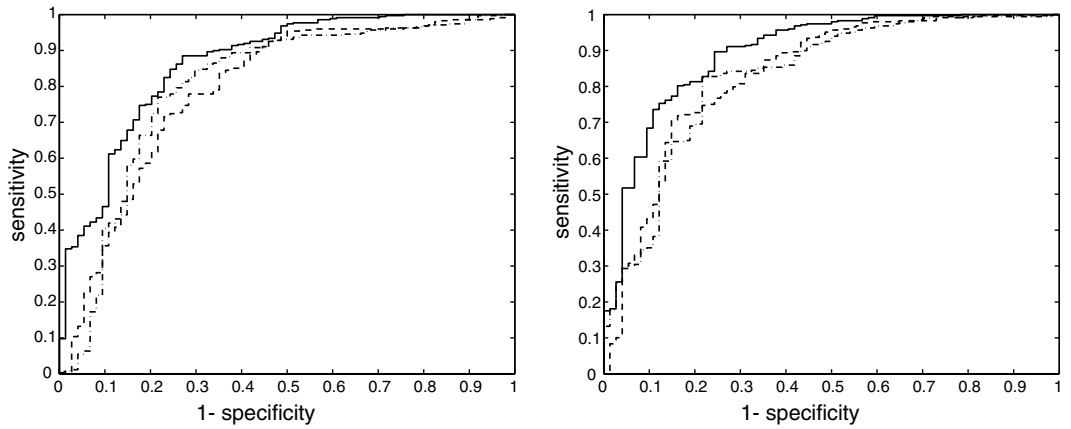


Fig. 4. Receiver operating characteristic curves for the full input set (left) and pruned input set (right): LS-SVM (solid line), Logit (dashed–dotted line) and LDA (dashed line).

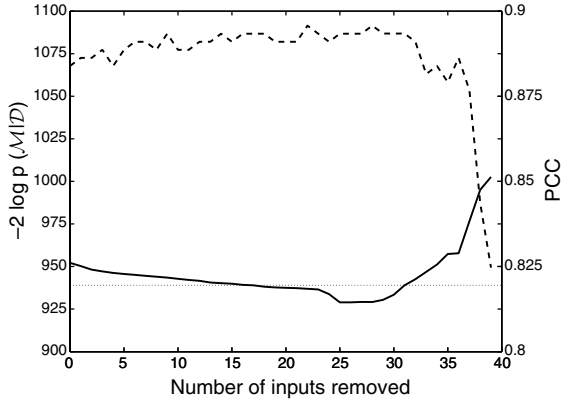


Fig. 5. Evolution of the level 3 cost function $-\log p(\mathscr{M}|\mathscr{D})$ and the leave-one-out cross-validation classification performance. The dashed line denotes where the model becomes different from the optimal model in a decisive way.

Table 4
Leave-one-out classification performances for the LDA, Logit and LS-SVM model using the optimized input sets

	LDA	LOGIT	LS-SVM	$\text{LS-SVM}_{\text{Bay}}$
PCC	86.49 (3.76)	86.49 (4.46)	89.34 (100)	89.34 (100)
PCC_p	98.28% (100%)	97.13% (34.28%)	98.28% (100%)	98.28% (100%)
PCC_n	31.08% (1.39%)	36.49% (9.90%)	47.30% (100%)	47.30% (100%)
AUROC	83.32% (0.81%)	83.13% (0.58%)	89.46% (100%)	89.35% (47.38%)

classifiers yield very good performances when compared to the LDA and Logit classifiers. The ROC curves on the optimized input sets are reported in the right pane of Fig. 4. The order of input removal is reported in Table 2. It can be seen that the pruned LS-SVM classifier has 11 inputs, the pruned LDA classifier 10 inputs and the pruned Logit classifier 6 inputs. Starting from a total set of 40 inputs, this clearly illustrates the efficiency of the suggested input selection procedure. All classifiers seem to agree on the importance of the turnover variable and the solvency variable. Consistent with prior studies [1,2], the inputs of the LS-SVM classifier consist of a mixture of profitability, solvency and liquidity ratios; but the exact ratios that are selected differ. Also, liquidity ratios seem to be less decisive as compared to prior bankruptcy studies. The number of days to customer credit is the only liquidity ratio that is withheld and only classifies as the 11th input; its trend is the second most important liquidity input in the backward input selection procedure. The three most important inputs for the LS-SVM classifier are the 2 solvency measures (solvency ratio, capital and reserves (percentage of total assets)), the size variable total assets and the profitability measures return on equity and turnover (percentage of total assets). Note that the five most important inputs for the LS-SVM classifier are also present in the optimally pruned LDA classifier.

The posterior class probabilities were computed for the evaluation of the decision rule (59) in a leave-one-out procedure, as mentioned above. These probabilities can also be used to identify the most difficult cases, which can be classified in an alternative way requiring e.g. human intervention. Referring the 10% most difficult cases to further analysis, the following classification performances were obtained on the remaining cases: PCC 93.12%, PCC_p 99.69%, PCC_n 52.83%. In the case of 25% removal, we obtained PCC 94.64%, PCC_p 99.65%, PCC_n 52.94%. These results clearly motivate the use of posterior class probabilities to allow the system to detect whether it should remark that its decision is too uncertain and needs further investigation.

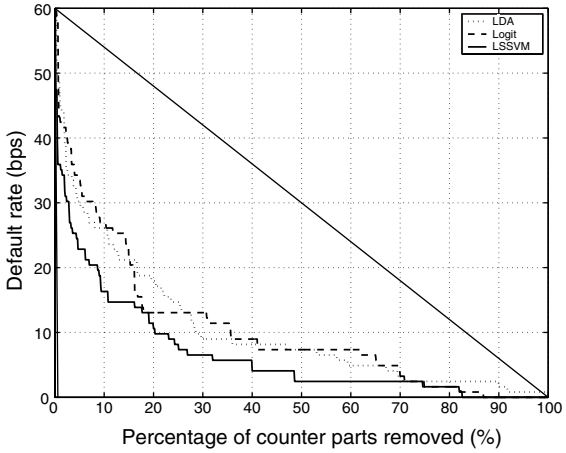


Fig. 6. Default rates (leave-one-out) on the full portfolio as a function of the percentage of refused counterparts for the LDA (dotted line), Logit (dashed line) and LS-SVM (solid line).

In order to gain insight in the performance improvements of the different models, the full data sample was used, oversampling the non-defaults 7 times so as to obtain a more realistic sample because 7 years of defaults were combined with 1 year of non-defaults. The corresponding average default/bankruptcy rate is equal to 0.60% or 60 bps (basis points). The graph depicted in Fig. 6 reports the remaining default rate on the full portfolio as a function of the percentage of the ordered portfolio. In the ideal case, the curve would be a straight line from (0%, 60 bps) to (0.6%, 0 bps); a random scoring function that does not succeed in discriminating between weak and strong firms results into a diagonal line. The slope of the curve is a measure for the default rate at that point. Consider, e.g., the case where one decides not to grant credits to the 10% counterparts with the worst scores. The default rates on the full 100% portfolio (with 10% liquidities) are 26 bps (LDA), 27 bps (Logit) and 16 bps (LS-SVM), respectively. Taking into account the fact that the number of counterparts is reduced from 100% to 90%, the default rates on the invested part of the portfolio are obtained by multiplication with 1/0.90 and are equal to 29 bps (LDA), 30 bps (Logit) and 18 bps (LS-SVM), respectively, corresponding to the slope between the points at 10% and 100% (x-axis). From this graph, the better performance of the LS-SVM classifier becomes obvious from a practical perspective.

7. Conclusions

Prediction of business failure is becoming more and more a key component of risk management for financial institutions nowadays. In this paper, we illustrated and evaluated the added value of Bayesian LS-SVM classifiers in this context. We conducted experiments using a bankruptcy data set on the Benelux mid-cap market. The suggested Bayesian nonlinear kernel based classifiers yield better performances than the more traditional methods, such as logistic regression and linear discriminant analysis, in terms of classification accuracy and area under the receiver operating characteristic curve. The set of relevant explanatory variables was inferred from the data by applying Bayesian model comparison in a backward input-selection procedure. By adopting the Bayesian way of reasoning, one easily obtains posterior class probabilities that can be of high importance to credit managers for analysing the sensitivities of the classifier decisions with respect to the given inputs.

Acknowledgments

This research was supported by Dexia, Fortis, the K.U. Leuven, the Belgian federal government (IUAP V, GOA-Mefisto 666) and the national science foundation (FWO) with project G.0407.02. This research was initiated when TVG was at the K.U. Leuven and continued at Dexia. TVG is a honorary postdoctoral researcher with the FWO-Flanders. The authors wish to thank Peter Van Dijcke, Joao Garcia, Luc Leonard, Eric Hermann, Marc Itterbeek, Daniel Saks, Daniel Feremans, Geert Kindt, Thomas Alderweireld, Carine Brasseur and Jos De Brabanter for helpful comments.

Appendix A. Primal–dual formulations for Bayesian inference

A.1. Expression for the Hessian and covariance matrix

The level 1 posterior probability $p([\boldsymbol{w};b]|\mathscr{D},\mu,\zeta,\mathscr{M})$ is a multivariate normal distribution in $\mathbb{R}^{n_\varphi}$ with mean $[\boldsymbol{w}_{\mathrm{mp}};b_{\mathrm{mp}}]$ and covariance matrix $\boldsymbol{Q}=\boldsymbol{H}^{-1}$, where $\boldsymbol{H}$ is the Hessian of the least squares cost function (19). Defining the matrix of regressors $\boldsymbol{\Phi}^{\mathrm{T}}=[\boldsymbol{\varphi}(\boldsymbol{x}_1),\ldots,\boldsymbol{\varphi}(\boldsymbol{x}_{n_\varphi})]$, the identity matrix $\boldsymbol{I}$ and the vector with all ones $\mathbf{1}$ of appropriate dimension; the Hessian is equal to

$$\boldsymbol{H}=\begin{bmatrix}\boldsymbol{H}_{11} & \boldsymbol{h}_{12}\\ \boldsymbol{h}_{21} & h_{22}\end{bmatrix}=\begin{bmatrix}\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi} & \zeta\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}\\ \zeta\mathbf{1}^{\mathrm{T}}\boldsymbol{\Phi} & \zeta N\end{bmatrix} \tag{A.1}$$

with corresponding block matrices $\boldsymbol{H}_{11}=\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}$, $\boldsymbol{h}_{12}=\boldsymbol{h}_{21}^{\mathrm{T}}=\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}$ and $h_{22}=N$. The inverse Hessian $\boldsymbol{H}^{-1}$ is then obtained via a Schur complement type argument:

$$\begin{aligned}\boldsymbol{H}^{-1}&=\left(\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & \boldsymbol{X}\\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & -\boldsymbol{X}\\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{H}_{11} & \boldsymbol{h}_{12}\\ \boldsymbol{h}_{12}^{\mathrm{T}} & h_{22}\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & \mathbf{0}\\ -\boldsymbol{X}^{\mathrm{T}} & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & \mathbf{0}\\ \boldsymbol{X}^{\mathrm{T}} & 1\end{bmatrix}\right)^{-1}\\ &=\left(\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & \boldsymbol{X}\\ \mathbf{0}^{\mathrm{T}} & 1\end{bmatrix}\begin{bmatrix}\boldsymbol{H}_{11}-\boldsymbol{h}_{12}h_{22}^{-1}\boldsymbol{h}_{12}^{\mathrm{T}} & \mathbf{0}\\ \mathbf{0}^{\mathrm{T}} & h_{22}\end{bmatrix}\begin{bmatrix}\boldsymbol{I}_{n_\varphi} & \mathbf{0}\\ \boldsymbol{X}^{\mathrm{T}} & 1\end{bmatrix}\right)^{-1} \qquad (A.2)\\ &=\begin{bmatrix}(\boldsymbol{H}_{11}-\boldsymbol{h}_{12}h_{22}^{-1}\boldsymbol{h}_{12}^{\mathrm{T}})^{-1} & -\boldsymbol{F}_{11}^{-1}\boldsymbol{h}_{12}h_{22}^{-1}\\ -h_{22}^{-1}\boldsymbol{h}_{12}^{\mathrm{T}}\boldsymbol{F}_{11}^{-1} & h_{22}^{-1}+h_{22}^{-1}\boldsymbol{h}_{12}^{\mathrm{T}}\boldsymbol{F}_{11}^{-1}\boldsymbol{h}_{12}h_{22}^{-1}\end{bmatrix} \qquad (A.3)\end{aligned}$$

with $\boldsymbol{X}=\boldsymbol{h}_{12}h_{22}^{-1}$ and $\boldsymbol{F}_{11}=\boldsymbol{H}_{11}-\boldsymbol{h}_{12}h_{22}^{-1}\boldsymbol{h}_{12}^{\mathrm{T}}$. In matrix expressions, it is useful to express $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{\Phi}-\frac{1}{N}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}\mathbf{1}^{\mathrm{T}}\boldsymbol{\Phi}$ as $\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi}$ with the idempotent centering matrix $\boldsymbol{M}_c=\boldsymbol{I}_N-\frac{1}{N}\mathbf{1}\mathbf{1}^{\mathrm{T}}\in\mathbb{R}^{N\times N}$ having $\boldsymbol{M}_c=\boldsymbol{M}_c^2$. Given that $\boldsymbol{F}_{11}^{-1}=(\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})^{-1}$, the inverse Hessian $\boldsymbol{H}^{-1}=\boldsymbol{Q}$ is equal to

$$\boldsymbol{Q}=\begin{bmatrix}(\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})^{-1} & -\frac{1}{N}(\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}\\ -\frac{1}{N}\mathbf{1}^{\mathrm{T}}\boldsymbol{\Phi}(\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})^{-1} & \frac{1}{\zeta N}+\frac{1}{N^2}\mathbf{1}^{\mathrm{T}}\boldsymbol{\Phi}(\mu\boldsymbol{I}_n+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})^{-1}\boldsymbol{\Phi}^{\mathrm{T}}\mathbf{1}\end{bmatrix}.$$

A.2. Expression for the determinant

The determinant of $\boldsymbol{H}$ is obtained from (A.2) using the fact that the determinant of a product is equal to the product of the determinants and is thus equal to

$$\det(\boldsymbol{H})=\det(\boldsymbol{H}_{11}-\boldsymbol{h}_{12}^{\mathrm{T}}h_{22}^{-1}\boldsymbol{h}_{12})\times\det(h_{22})=\det(\mu\boldsymbol{I}_{n_\varphi}+\zeta\boldsymbol{\Phi}^{\mathrm{T}}\boldsymbol{M}_c\boldsymbol{\Phi})\times(\zeta N), \tag{A.4}$$

which is obtained as the product of ζN and the eigenvalues λ_i ($i = 1, \ldots, n_\varphi$) of $\mu \boldsymbol{I}_{n_\varphi} + \zeta \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi}$, noted as $\lambda_i(\mu \boldsymbol{I}_{n_\varphi} + \zeta \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi})$. Because the matrix $\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi} \in \mathbb{R}^{n_\varphi \times n_\varphi}$ is rank deficient with rank $n_e \leqslant N - 1$, $n_\varphi - n_e$ eigenvalues are equal to μ.

The dual space expressions can be obtained in terms of the singular value decomposition

$$\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c = \boldsymbol{U} \boldsymbol{S} \boldsymbol{V}^{\mathrm{T}} = [\, \boldsymbol{U}_1 \quad \boldsymbol{U}_2 \,] \begin{bmatrix} \boldsymbol{S}_1 & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{0} \end{bmatrix} [\, \boldsymbol{V}_1 \quad \boldsymbol{V}_2 \,], \tag{A.5}$$

with $\boldsymbol{U} \in \mathbb{R}^{n_\varphi \times n_\varphi}$, $\boldsymbol{S} \in \mathbb{R}^{n_\varphi \times N}$, $\boldsymbol{V} \in \mathbb{R}^{N \times N}$ and with the block matrices $\boldsymbol{U}_1 \in \mathbb{R}^{n_\varphi \times n_e}$, $\boldsymbol{U}_2 \in \mathbb{R}^{n_\varphi \times (n_\varphi - n_e)}$, $\boldsymbol{S}_1 = \mathrm{diag}([s_1, s_2, \ldots, s_{n_e}]) \in \mathbb{R}^{n_e \times n_e}$, $\boldsymbol{V}_1 \in \mathbb{R}^{N \times n_e}$ and $\boldsymbol{V}_2 \in \mathbb{R}^{N \times (N - n_e)}$, with $0 \leqslant n_e \leqslant N - 1$. Due to the orthonormality property we have $\boldsymbol{U}\boldsymbol{U}^{\mathrm{T}} = \boldsymbol{U}_1 \boldsymbol{U}_1^{\mathrm{T}} + \boldsymbol{U}_2 \boldsymbol{U}_2^{\mathrm{T}} = \boldsymbol{I}_{n_\varphi}$ and $\boldsymbol{V}\boldsymbol{V}^{\mathrm{T}} = \boldsymbol{V}_1 \boldsymbol{V}_1^{\mathrm{T}} + \boldsymbol{V}_2 \boldsymbol{V}_2^{\mathrm{T}} = \boldsymbol{I}_N$. Hence, one obtains the primal and dual eigenvalue decompositions

$$\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi} = \boldsymbol{U}_1 \boldsymbol{S}_1^2 \boldsymbol{U}_1^{\mathrm{T}}, \tag{A.6}$$

$$\boldsymbol{M}_c \boldsymbol{\Phi} \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c = \boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c = \boldsymbol{V}_1 \boldsymbol{S}_1^2 \boldsymbol{V}_1^{\mathrm{T}}. \tag{A.7}$$

The n_φ eigenvalues of $\mu \boldsymbol{I}_{n_\varphi} + \zeta \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi}$ are equal to $\lambda_1 = \mu + \zeta s_1^2, \ldots, \lambda_{n_e} = \mu + \zeta s_{n_e}^2, \lambda_{n_e+1} = \mu, \ldots, \lambda_{n_\varphi} = \mu$, where the non-zero eigenvalues s_i^2 ($i = 1, \ldots, n_e$) are obtained from the eigenvalue decomposition of $\boldsymbol{M}_c \boldsymbol{\Phi} \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c$ from (A.7). The expression for the determinant is equal to $N \zeta \mu^{N - n_e} \prod_{i=1}^{n_e} (\mu + \zeta \lambda_i (\boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c)$, with $\boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c = \boldsymbol{V}_1 \mathrm{diag}([\lambda_1, \ldots, \lambda_{n_e}]) \boldsymbol{V}_1^{\mathrm{T}}$ and $\lambda_i = s_i^2$, $i = 1, \ldots, n_e$.

A.3. Expression for the level 1 cost function

The dual space expression for $\mathscr{J}_1(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}})$ is obtained by substituting $[\boldsymbol{w}_{\mathrm{mp}}; b_{\mathrm{mp}}] = \boldsymbol{H}^{-1}[\boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{y}; \boldsymbol{1}^{\mathrm{T}} \boldsymbol{y}]$ in (19). Applying a similar reasoning and algebra as for the calculation of the determinant, one obtains the dual space expression:

$$\mathscr{J}_1(\boldsymbol{w}, b) = \mu \mathscr{J}_{\boldsymbol{w}}(\boldsymbol{w}_{\mathrm{mp}}) + \zeta \mathscr{J}_{\boldsymbol{e}}(\boldsymbol{w}_{\mathrm{mp}}, b_{\mathrm{mp}}) = \frac{1}{2} \boldsymbol{y}^{\mathrm{T}} \boldsymbol{M}_c (\mu^{-1} \boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c + \zeta^{-1} \boldsymbol{I}_N)^{-1} \boldsymbol{M}_c \boldsymbol{y}. \tag{A.8}$$

Given that $\boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c = \boldsymbol{V} \boldsymbol{\Lambda} \boldsymbol{V}^{\mathrm{T}}$, with $\boldsymbol{\Lambda} = \mathrm{diag}([s_1^2, \ldots, s_{n_e}^2, 0, \ldots, 0])$, one obtains that (48). In a similar way, one obtains (46) and (47).

A.4. Expression for the moderated likelihood

The primal space expression for the variance in the moderated output is obtained from (56) and is equal to

$$\sigma_{e_\bullet}^2 = [\boldsymbol{\varphi}(\boldsymbol{x}) - 1/N_\bullet \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{1}_\bullet]^{\mathrm{T}} \boldsymbol{Q}_{11} [\boldsymbol{\varphi}(\boldsymbol{x}) - 1/N_\bullet \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{1}_\bullet]. \tag{A.9}$$

Substituting (A.5) into the expression for $\boldsymbol{Q}_{11}$ from (A.3), one can write $\boldsymbol{Q}_{11}$ as

$$\begin{aligned} \boldsymbol{Q}_{11} &= (\mu \boldsymbol{I}_{n_\varphi} + \zeta \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi})^{-1} = (\mu \boldsymbol{U}_2 \boldsymbol{U}_2^{\mathrm{T}} + \boldsymbol{U}_1 (\mu \boldsymbol{I}_{n_e} + \zeta \boldsymbol{S}_1^2) \boldsymbol{U}_1^{\mathrm{T}})^{-1} \\ &= \mu^{-1} \boldsymbol{U}_2 \boldsymbol{U}_2^{\mathrm{T}} + \boldsymbol{U}_1 (\mu \boldsymbol{I}_{n_e} + \zeta \boldsymbol{S}_1^2)^{-1} \boldsymbol{U}_1^{\mathrm{T}} = \mu^{-1} \boldsymbol{I}_{n_\varphi} + \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{V}_1 \boldsymbol{S}_1^{-1} \Big((\mu \boldsymbol{I}_{n_e} + \zeta \boldsymbol{S}_1^2)^{-1} - \mu^{-1} \Big) \boldsymbol{U}_1^{\mathrm{T}} \\ &= \mu^{-1} \boldsymbol{I}_{n_\varphi} + \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{V}_1 \boldsymbol{S}_1^{-1} ((\mu \boldsymbol{I}_{n_e} + \zeta \boldsymbol{S}_1^2)^{-1} - \mu^{-1}) \boldsymbol{S}_1^{-1} \boldsymbol{V}_1^{\mathrm{T}} \boldsymbol{M}_c \boldsymbol{\Phi} \\ &= 1/\mu \boldsymbol{I} n_\varphi - \zeta/\mu \boldsymbol{\Phi}^{\mathrm{T}} \boldsymbol{M}_c (\mu \boldsymbol{I}_N + \zeta \boldsymbol{M}_c \boldsymbol{\Omega} \boldsymbol{M}_c)^{-1} \boldsymbol{M}_c \boldsymbol{\Phi}. \end{aligned} \tag{A.10}$$

Substituting (A.9) into (A.10), one obtains (57) given that $\boldsymbol{\Phi}\boldsymbol{\Phi}^{\mathrm{T}} = \boldsymbol{\Omega}$, $\boldsymbol{\varphi}(\boldsymbol{x}_i)^{\mathrm{T}} \boldsymbol{\varphi}(\boldsymbol{x}_j) = K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $\boldsymbol{\Phi} \boldsymbol{\varphi}(\boldsymbol{x}) = \boldsymbol{\theta}(\boldsymbol{x})$.

References

[1] E. Altman, Financial ratios, discriminant analysis and the prediction of corporate bankruptcy, Journal of Finance 23 (1968) 589–609.
[2] E. Altman, Corporate Financial Distress and Bankruptcy: A Complete Guide to Predicting and Avoiding Distress and Profiting from Bankruptcy, Wiley Finance Edition, 1993.
[3] W. Beaver, Financial ratios as predictors of failure, empirical research in accounting selected studies, Journal of Accounting Research 5 (Suppl.) (1966) 71–111.
[4] C. Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.
[5] E. Altman, G. Marco, F. Varetto, Corporate distress diagnosis: Comparisons using linear discriminant analysis and neural networks (the Italian experience), Journal of Banking and Finance 18 (1994) 505–529.
[6] A. Atiya, Bankruptcy prediction for credit risk using neural networks: A survey and new results, IEEE Transactions on Neural Networks 12 (4) (2001) 929–935.
[7] D.-E. Baestaens, W.-M. van den Bergh, D. Wood, Neural Network Solutions for Trading in Financial Markets, Pitman, London, 1994.
[8] K. Lee, I. Han, Y. Kwon, Hybrid neural network models for bankruptcy predictions, Decision Support Systems 18 (1996) 63–72.
[9] S. Piramuthu, H. Ragavan, M. Shaw, Using feature construction to improve the performance of neural networks, Management Science 44 (3) (1998) 416–430.
[10] C. Serrano Cinca, Self organizing neural networks for financial diagnosis, Decision Support Systems 17 (1996) 227–238.
[11] B. Wong, T. Bodnovich, Y. Selvi, Neural network applications in business: A review and analysis of the literature (1988–1995), Decision Support Systems 19 (4) (1997) 301–320.
[12] D. MacKay, Bayesian interpolation, Neural Computation 4 (1992) 415–447.
[13] D. MacKay, Probable networks and plausible predictions—A review of practical Bayesian methods for supervised neural networks, Network: Computation in Neural Systems 6 (1995) 469–505.
[14] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines, Cambridge University Press, 2000.
[15] B. Schölkopf, A. Smola, Learning with Kernels, MIT Press, Cambridge, MA, 2002.
[16] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, J. Vandewalle, Least Squares Support Vector Machines, World Scientific, New Jersey, 2002.
[17] V. Vapnik, Statistical Learning Theory, Wiley, New York, 1998.
[18] J.A.K. Suykens, J. Vandewalle, Least squares support vector machine classifiers, Neural Processing Letters 9 (3) (1999) 293–300.
[19] G. Baudat, F. Anouar, Generalized discriminant analysis using a kernel approach, Neural Computation 12 (2000) 2385–2404.
[20] T. Van Gestel, J.A.K. Suykens, G. Lanckriet, A. Lambrechts, B. De Moor, J. Vandewalle, A Bayesian framework for least squares support vector machine classifiers, Gaussian processes and kernel Fisher discriminant analysis, Neural Computation 14 (2002) 1115–1147.
[21] T. Van Gestel, J.A.K. Suykens, D.-E. Baestaens, A. Lambrechts, G. Lanckriet, B. Vandaele, B. De Moor, J. Vandewalle, Predicting financial time series using least squares support vector machines within the evidence framework, IEEE Transactions on Neural Networks (Special Issue on Financial Engineering) 12 (2001) 809–821.
[22] R. Eisenbeis, Pitfalls in the application of discriminant analysis in business, The Journal of Finance 32 (3) (1977) 875–900.
[23] R. Duda, P. Hart, Pattern Classification and Scene Analysis, John Wiley, New York, 1973.
[24] B. Ripley, Pattern Classification and Neural Networks, Cambridge University Press, 1996.
[25] R. Fisher, The use of multiple measurements in taxonomic problems, Annals of Eugenics 7 (1936) 179–188.
[26] P. McCullagh, J. Nelder, Generalized Linear Models, Chapman & Hall, London, 1989.
[27] J. Ohlson, Financial ratios and the probabilistic prediction of bankruptcy, Journal of Accounting Research 18 (1980) 109–131.
[28] B. Baesens, R. Setiono, C. Mues, J. Vanthienen, Using neural network rule extraction and decision tables for credit-risk evaluation, Management Science 49 (3) (2003) 312–329.
[29] B. Baesens, T. Van Gestel, S. Viaene, M. Stepanova, J.A.K. Suykens, J. Vanthienen, Benchmarking state of the art classification algorithms for credit scoring, Journal of the Operational Research Society 54 (6) (2003) 627–635.
[30] B. Baesens, Developing intelligent systems for credit scoring using machine learning techniques, Ph.D. thesis, Department of Applied Economic Sciences, Katholieke Universiteit Leuven, 2003.
[31] T. Van Gestel, J.A.K. Suykens, B. Baesens, S. Viaene, J. Vanthienen, G. Dedene, B. De Moor, J. Vandewalle, Benchmarking least squares support vector machine classifiers, Machine Learning 54 (2004) 5–32.
[32] V. Vapnik, Statistical Learning Theory, John Wiley, New York, 1998.
[33] T. Evgeniou, M. Pontil, T. Poggio, Regularization networks and support vector machines, Advances in Computational Mathematics 13 (2001) 1–50.
[34] J. Hutchinson, A. Lo, T. Poggio, A nonparametric approach to pricing and hedging derivative securities via learning networks, Journal of Finance 49 (1994) 851–889.

[35] V. Vapnik, A. Lerner, Pattern recognition using generalized portrait method, Automation and Remote Control 24 (1963) 774–780.
[36] V. Vapnik, A.J. Chervonenkis, On the one class of the algorithms of pattern recognition, Automation and Remote Control 25 (6).
[37] R. Fletcher, Practical Methods of Optimization, John Wiley, Chichester and New York, 1987.
[38] C. Cortes, V. Vapnik, Support vector networks, Machine Learning 20 (1995) 273–297.
[39] H. Jeffreys, Theory of Probability, Oxford University Press, 1961.
[40] D.-E. Baestaens, Credit risk modelling strategies: The road to serfdom, International Journal of Intelligent Systems in Accounting, Finance & Management 8 (1999) 225–235.
[41] A. Van der Vaart, Asymptotic Statistics, Cambridge University Press, 1998.
[42] J. Egan, Signal Detection Theory and ROC analysis. Series in Cognition and Perception, Academic Press, New York, 1975.
[43] B. Everitt, The Analysis of Contingency Tables, Chapman & Hall, London, 1977.
[44] E. De Long, D. De Long, D. Clarke-Pearson, Comparing the areas under two or more correlated receiver operating characteristic curves: A nonparametric approach, Biometrics 44 (1988) 837–845.
[45] J. Soberhart, S. Keenan, R. Stein, Validation methodologies for default risk models, Credit Magazine 1 (4) (2000) 51–56.