Asymmetric Least Squares Support Vector Machine Classifiers

Xiaolin Huang[a,*], Lei Shi[a,b], Johan A.K. Suykens[a]

[a]*Department of Electrical Engineering, ESAT-STADIUS, KU Leuven, B-3001 Leuven, Belgium*
[b]*School of Mathematical Sciences, Fudan University, 200433, Shanghai, P.R. China.*

Abstract

In the field of classification, the support vector machine (SVM) pursues a large margin between two classes. The margin is usually measured by the minimal distance between two sets, which is related to the hinge loss or the squared hinge loss. However, the minimal value is sensitive to noise and unstable to re-sampling. To overcome this weak point, the expectile value is considered to measure the margin between classes instead of the minimal value. Motivated by the relation between the expectile value and the asymmetric squared loss, asymmetric least squares SVM (aLS-SVM) is proposed. The proposed aLS-SVM also can be regarded as an extension to LS-SVM and L2-SVM. Theoretical analysis and numerical experiments on aLS-SVM illustrate its insensitivity to noise around the boundary and its stability to re-sampling.

Keywords: classification, support vector machine, least squares support vector machine, asymmetric least squares

1. Introduction

The task of binary classification is to classify the data into two classes. A large margin between the two classes plays an important role to obtain a good classifier. To maximize the margin, Vapnik (1995) proposed the support vector machine (SVM), which has been widely studied and applied. Traditionally, the SVM classifiers maximize the margin measured by the minimal distance between two classes. However, the minimal distance is sensitive to noise around the decision boundary and is not stable to re-sampling. To further improve the performance of SVMs, we will use the expectile value to measure the margin and propose the corresponding classifier to maximize the expectile distance.

Consider a data set $\mathbf{z} = \{(x_i, y_i)\}_{i=1}^m$, where $x_i \in \mathbb{R}^d$ and $y_i \in \{-1, 1\}$. Then $\mathbf{z}$ consists of two classes with the following sets of indices: $\mathbf{I} = \{i \mid y_i = 1\}$ and $\mathbf{II} = \{i \mid y_i = -1\}$. We are seeking a function $f(x)$ of which the sign $\mathrm{sgn}(f)$ is used for classification. To find a suitable function, we need a criterion to measure the quality of the classifier. For a given $f(x)$, the features are mapped into $\mathbb{R}$. A large margin between the two mapped sets is required for a good generalization capability. Traditionally, the margin is measured by the extreme value, i.e., $\min f(\mathbf{I}) + \min f(\mathbf{II})$, where $f(\mathbf{I}) = \{y_i f(x_i), i \in \mathbf{I}\}$ and $f(\mathbf{II}) = \{y_i f(x_i), i \in \mathbf{II}\}$. In this setting, a good classifier can be found by

$$\max_{\|f\|=1} \quad \min f(\mathbf{I}) + \min f(\mathbf{II}). \tag{1}$$

In the SVM classification framework, one achieves $\min f(\mathbf{I}) = \min f(\mathbf{II}) = 1$ by minimizing the hinge loss $\max\{0, 1 - y_i f(x_i)\}$ or the squared hinge loss $\max\{0, 1 - y_i f(x_i)\}^2$. When f is chosen from affine linear functions, i.e., $f(x) = w^T x + b$, we can equivalently formulate (1) as minimizing $w^T w$, since $2/\|w\|_2$ measures

*Corresponding author. ESAT-STADIUS, Kasteelpark Arenberg 10, bus 2446, 3001 Heverlee, Belgium; Tel: +32-16328653, Fax: +32-16321970.

Email addresses: huangxl06@mails.tsinghua.edu.cn (Xiaolin Huang), leishi@fudan.edu.cn (Lei Shi), johan.suykens@esat.kuleuven.be (Johan A.K. Suykens)

the distance between $f(x) = w^T x + b = \pm 1$. This geometric meaning of the SVM has been explained by Vapnik (1995). Accordingly, (1) is transformed into

$$\min_{w,b} \qquad \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} L\left(1 - y_i\left(w^T x_i + b\right)\right), \tag{2}$$

where the loss function can be the hinge loss or the squared hinge loss, resulting in L1-SVM and L2-SVM, respectively.

Measuring the margin by the extreme value is unstable to re-sampling, which is a common technique for large scale data sets. Suppose $\mathbf{I}'$ is a subset of $\mathbf{I}$. For different re-samplings from the same distribution, $\min f(\mathbf{I}')$ varies a lot and can be quite different from $\min f(\mathbf{I})$. Because of the same reason, we can also see that (1) is sensitive to noise on x_i around the decision boundary. Bi and Zhang (2005) called the noise on x_i *feature noise*, which can be caused by instrumental errors and sampling errors. Generally, L1-SVM or L2-SVM is sensitive to re-sampling and noise around the boundary, which has been observed by Guyon et al. (1996); Herbrich and Westion (1999); Song et al. (2002); Hu and Song (2004); Huang et al. (2013).

The sensitivity to noise around the decision boundary and the instability to re-sampling are related to the fact that the margin is measured by the extreme value. Hence, to improve the performance of the traditional SVMs, we can modify the measurement of margin by taking the quantile value. In the discrete form, the p (lower) quantile of a set of scalars $U = \{u_1, u_2, \ldots, u_m\}$ can be denoted by

$$\min{}^p\{U\} := \left\{t : t \in \mathbb{R}, t \text{ is larger than } p \text{ ratio of } u_i\right\}.$$

Then (1) is modified into

$$\max_{\|f\|=1} \quad \min{}^p f(\mathbf{I}) + \min{}^p f(\mathbf{II}). \tag{3}$$

Compared with the extreme value, the quantile value is more robust to re-sampling and noise. Hence the good performance of (3) can be expected. Similarly to L1-SVM or L2-SVM, (3) can be posed as minimizing $w^T w$ with the condition that $\min^p f(\mathbf{I}) = \min^p f(\mathbf{II}) = 1$. This idea has been implemented by Huang et al. (2013), where the pinball loss SVM (pin-SVM) classifier has been established and the related properties have been discussed.

Using the quantile distance instead of the minimal distance can improve the performance of L1-SVM classifier for re-sampling or noise around the decision boundary. To speed up the training process for pin-SVM, we use the expectile distance as a surrogate of the quantile distance and propose a new SVM classifier in this paper. This is motivated by the fact that the expectile value, which is related to minimizing the asymmetric squared loss, has similar statistical properties to the quantile value, which is related to minimizing the pinball loss. The expectile has been discussed insightfully by Newey (1987) and Efron (1991). Since computing the expectile is less time consuming than computing the quantile, the expectile value has been applied to approach the quantile value in many fields (Koenker et al. (1996); Taylor (2008); De Rossi and Harvey (2009); Sobotka and Thomas (2012)). Huang et al. (2013) have applied the pinball loss to find a large quantile distance and in this paper we focus on the expectile distance and propose asymmetric least squares SVM (aLS-SVM). The relationship between pin-SVM and aLS-SVM is similar to that between quantile regression and expectile regression, of which the latter one is an approximation of the first one and can be effectively solved. The proposed aLS-SVM also can be regarded as an extension to least squares support vector machine (LS-SVM, Suykens and Vandewalle (1999); Suykens et al. (2002b)). When no bias term is used, LS-SVM in the primal space corresponds to ridge regression, as discussed by Van Gestel et al. (2002). LS-SVM has been widely applied in many fields. Wei et al. (2011); Shao et al. (2012); Hamid et al. (2012); Luts et al. (2012) reported some recent progress on LS-SVM.

In the remainder of this paper, we first give aLS-SVM and its dual formulation in Section 2. Section 3 discusses the properties of aLS-SVM. In Section 4, the proposed method is evaluated by numerical experiments. Finally, Section 5 ends the paper with conclusions.

2. Asymmetric Least Squares SVM

Traditionally, classifier training focuses on maximizing the extreme distance. Minimizing the hinge loss or the squared hinge loss leads to $\min f(\mathbf{I}) = \min f(\mathbf{II}) = 1$. In linear classification, $w^T w$ measures the margin between the hyperplanes $w^T x + b = 1$ and $w^T x + b = -1$, which follows that (1) can be handled by L1-SVM or L2-SVM.

As discussed previously, to improve the performance of SVM for noise and re-sampling, we can maximize the quantile distance instead of (1). To handle the quantile distance maximization (3), we consider the following pinball loss,

$$L_p^{\text{pin}}(t) = \begin{cases} pt, & t \geq 0, \\ -(1-p)t, & t < 0, \end{cases}$$

which is related to the p (lower) quantile value and $0 \leq p \leq 1$. The pinball loss has been applied widely in quantile regression, see, e.g., Koenker (2005); Steinwart and Christmann (2008); Steinwart and Christmann (2011). Motivated by the approach of establishing L1-SVM, we can maximize the quantile distance by the following pinball loss SVM (pin-SVM) proposed by Huang et al. (2013),

$$\min_{w,b} \quad \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} L_p^{\text{pin}} \left(1 - y_i \left(w^T x_i + b\right)\right). \tag{4}$$

The pinball loss is non-smooth and its minimization needs more time than minimizing some smooth loss functions. Hence, to approximately calculate the quantile value in a short time, researchers proposed expectile regression, of which the statistical properties have been well discussed by Newey (1987); Efron (1991). Expectile regression minimizes the following squared pinball loss,

$$L_p^{\text{aLS}}(t) = \begin{cases} pt^2, & t \geq 0, \\ (1-p)t^2, & t < 0, \end{cases} \tag{5}$$

which is related to the p (lower) expectile value. The plots of $L_p^2(t)$ of several p values are shown in Fig.1. Because of its shape, we call (5) asymmetric squared loss. The expectile distance between two sets can be maximized by the following asymmetric least squares support vector machine (aLS-SVM),

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} L_p^{\text{aLS}} (e_i) \\ \text{s.t.} \quad & e_i = 1 - y_i \left(w^T x_i + b\right), \quad i = 1, 2, \ldots, m. \end{aligned} \tag{6}$$

From the definition of $L_p^{\text{aLS}}(t)$, one observes that when $p = 1$, the asymmetric squared loss becomes the squared hinge loss and aLS-SVM reduces to L2-SVM, which essentially focuses on the minimal distance. The relationship between pin-SVM (4) and aLS-SVM (6) is similar to that between quantile regression and expectile regression. Generally, aLS-SVM takes less computational time than pin-SVM and they have similar statistical properties.

Next, we study nonparametric aLS-SVM. Introducing a nonlinear feature mapping $\phi(x)$, we obtain the following nonlinear aLS-SVM,

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} L_p^{\text{aLS}} (e_i) \\ \text{s.t.} \quad & e_i = 1 - y_i \left(w^T \phi(x_i) + b\right), \quad i = 1, 2, \ldots, m, \end{aligned}$$

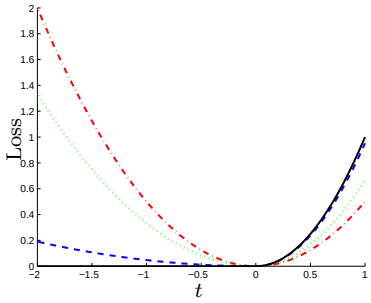


Figure 1: Plots of loss functions $L_p^{\mathrm{aLS}}(t)$ with $p = 0.5$ (red dash-dotted line), 0.667 (green dotted line), 0.957 (blue dashed line), and 1 (black solid line).

which then can be equivalently transformed into

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2}w^T w + \frac{C}{2}\sum_{i=1}^{m} e_i^2 \\ \text{s.t.} \quad & y_i\left(w^T\phi(x_i)+b\right) \geq 1 - \frac{1}{p}e_i, i = 1,2,\ldots,m, \\ & y_i\left(w^T\phi(x_i)+b\right) \leq 1 + \frac{1}{1-p}e_i, i = 1,2,\ldots,m. \end{aligned} \tag{7}$$

Since (7) is convex and there is no duality gap, we can solve (7) from the dual space. The Lagrangian with $\alpha_i \geq 0, \beta_i \geq 0$ is

$$\begin{aligned} \mathscr{L}(w,b,e;\alpha,\beta) &= \frac{1}{2}w^T w + \frac{C}{2}\sum_{i=1}^{m} e_i^2 - \sum_{i=1}^{m}\alpha_i\left(y_i\left(w^T\phi(x_i)+b\right) - 1 + \frac{1}{p}e_i\right) \\ & \quad - \sum_{i=1}^{m}\beta_i\left(-y_i\left(w^T\phi(x_i)+b\right) + 1 + \frac{1}{1-p}e_i\right). \end{aligned}$$

According to the following saddle point condition,

$$\begin{aligned} \frac{\partial\mathscr{L}}{\partial w} &= w - \sum_{i=1}^{m}(\alpha_i-\beta_i)y_i\phi(x_i) = 0, \\ \frac{\partial\mathscr{L}}{\partial b} &= -\sum_{i=1}^{m}(\alpha_i-\beta_i)y_i = 0, \\ \frac{\partial\mathscr{L}}{\partial e_i} &= Ce_i - \frac{1}{p}\alpha_i - \frac{1}{1-p}\beta_i = 0, \forall i = 1,2,\ldots,m, \end{aligned}$$

the dual problem of (7) is obtained as follows,

$$\begin{aligned} \max_{\alpha,\beta} \quad & -\frac{1}{2}\sum_{i=1}^{m}\sum_{j=1}^{m}(\alpha_i-\beta_i)y_i\phi(x_i)^T\phi(x_j)y_j(\alpha_j-\beta_j) - \frac{1}{2C}\sum_{i=1}^{m}\left(\frac{1}{p}\alpha_i + \frac{1}{1-p}\beta_i\right)^2 + \sum_{i=1}^{m}(\alpha_i-\beta_i) \\ \text{s.t.} \quad & \sum_{i=1}^{m}(\alpha_i-\beta_i)y_i = 0, \\ & \alpha_i \geq 0, \beta_i \geq 0, i = 1,2,\ldots,m. \end{aligned}$$

Now we let $\lambda_i = \alpha_i - \beta_i$ and introduce the positive definite kernel $\mathcal{K}(x_i, x_j) = \phi(x_i)^T \phi(x_j)$, which can be the radial basis function (RBF), polynomial and so on. Then, the nonparametric aLS-SVM is formulated as

$$\begin{aligned} \max_{\lambda,\beta} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i y_i \mathcal{K}(x_i, x_j) y_j \lambda_j - \frac{1}{2Cp} \sum_{i=1}^{m} \left(\lambda_i + \frac{1}{1-p} \beta_i \right)^2 \\ \text{s.t.} \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & \lambda_i + \beta_i \geq 0, \beta_i \geq 0, i = 1, 2, \ldots, m. \end{aligned} \tag{8}$$

At this stage, we again observe the relationship between aLS-SVM and L2-SVM by letting p tend to one. In that case, $\beta = 0$ will be optimal to (8), which then becomes the following dual formulation of L2-SVM,

$$\begin{aligned} \max_{\lambda} \quad & \sum_{i=1}^{m} \lambda_i - \frac{1}{2} \sum_{i=1}^{m} \sum_{j=1}^{m} \lambda_i y_i \mathcal{K}(x_i, x_j) y_j \lambda_j - \frac{1}{2C} \sum_{i=1}^{m} \lambda_i^2 \\ \text{s.t.} \quad & \sum_{i=1}^{m} \lambda_i y_i = 0, \\ & \lambda_i \geq 0, i = 1, 2, \ldots, m. \end{aligned} \tag{9}$$

Solving (8) leads to optimal λ, β value. After that, the aLS-SVM classifier is represented by dual variables as follows,

$$f(x) = w^T \phi(x) + b = \sum_{i=1}^{m} y_i \lambda_i \mathcal{K}(x, x_i) + b, \tag{10}$$

where the bias term b is computed according to

$$\begin{aligned} y_i \left(\sum_{j=1}^{m} y_j \lambda_j \mathcal{K}(x_i, x_j) + b \right) &= 1 - \frac{1}{p} e_i, \quad \forall i : \alpha_i > 0, \\ y_i \left(\sum_{j=1}^{m} y_j \lambda_j \mathcal{K}(x_i, x_j) + b \right) &= 1 + \frac{1}{1-p} e_i, \ \forall i : \beta_i > 0. \end{aligned}$$

The performance of nonparametric aLS-SVM with different p values is shown in Fig.2. Points in class **I** and **II** are shown by green stars and red crosses, respectively. Then we set $C = 1000$ and use RBF kernel $\mathcal{K}(x_i, x_j) = \exp(-\|x_i - x_j\|^2/\sigma^2)$ with $\sigma = 1.5$ to do classification by aLS-SVM with $p = 0.5, 0.667, 0.957$, and $p = 1$. The obtained surfaces $f(x) = \pm 1$ are shown in Fig.2. In aLS-SVM, $\{x : f(x) = \pm 1\}$ gives the expectile value and the expectile level is related to p. With an increasing value of p, $\{x : f(x) = \pm 1\}$ tends to the decision boundary.

3. Properties of aLS-SVM

3.1. Scatter minimization

The proposed aLS-SVM is trying to maximize the expectile distance between two sets. When $p = 1$, aLS-SVM reduces to the following L2-SVM,

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} \max\{0, e_i\}^2 \\ \text{s.t.} \quad & e_i = 1 - y_i \left(w^T \phi(x_i) + b \right), \quad i = 1, 2, \ldots, m, \end{aligned} \tag{11}$$

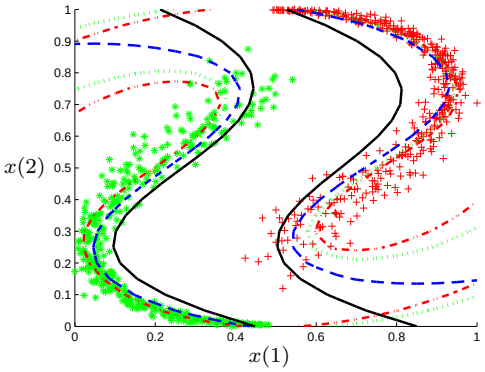


Figure 2: Sampling points and classification results of aLS-SVM. Points in class **I** and **II** are shown by green stars and red crosses, respectively. The surfaces $f(x) = \pm 1$ for $p = 0.5, 0.667, 0.957$, and $p = 1$ are illustrated by red dash-dotted, green dotted, blue dashed, and black solid lines, respectively.

which is to maximize the minimal distance between two sets. When $p = 0.5$, $L_p^{\mathrm{aLS}}(t)$ gives a symmetric penalty for negative and positive loss and then aLS-SVM becomes LS-SVM below,

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2} w^T w + \frac{C}{2} \sum_{i=1}^{m} e_i^2 \\ \text{s.t.} \quad & e_i = 1 - y_i \left(w^T \phi(x_i) + b \right), \quad i = 1, 2, \ldots, m. \end{aligned} \tag{12}$$

Thus, aLS-SVM (6) can be regarded as the trade-off between L2-SVM and LS-SVM:

$$\begin{aligned} \min_{w,b,e} \quad & \frac{1}{2} w^T w + \frac{C_1}{2} \sum_{i=1}^{m} \max\{0, e_i\}^2 + \frac{C_2}{2} \sum_{i=1}^{m} e_i^2 \\ \text{s.t.} \quad & e_i = 1 - y_i \left(w^T \phi(x_i) + b \right), \quad i = 1, 2, \ldots, m. \end{aligned}$$

For $C_1 = (2p-1)C$ and $C_2 = (1-p)C$, it is equivalent to (6).

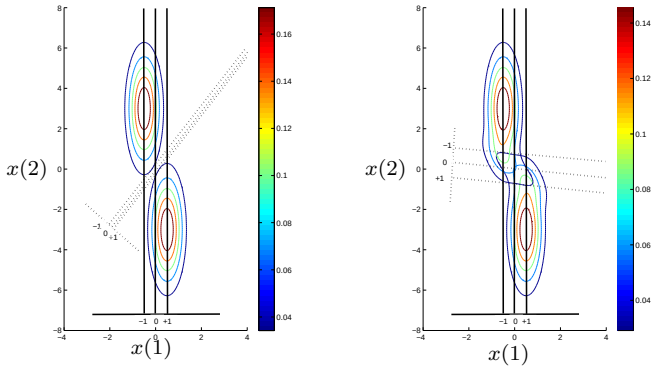
As mentioned previously, L2-SVM is considering two surfaces $w^T \phi(x_i) + b = \pm 1$, maximizing the distance between them and pushing the points to $y_i(w^T \phi(x_i) + b) \geq 1$. In LS-SVM, we are still searching two surfaces and maximizing the margin, but we are pushing the points to be located around the surface $y_i(w^T \phi(x_i) + b) = 1$, which is related to Fisher Discriminant Analysis (Suykens et al. (2002b); Van Gestel et al. (2002)). Briefly speaking, L2-SVM puts emphasis on the training misclassification error and LS-SVM tries to find small within-class scatter. In many applications, both small misclassification error and small within-class scatter lead to satisfactory results. Generally speaking, for noise-polluted data, LS-SVM is less sensitive. But in some cases, a small within-class scatter does not result in a good classifier, as illustrated by the following example.

In this example, points of two classes are drawn from two Gaussian distributions: $x_i, i \in \mathbf{I} \sim \mathcal{N}(\mu_\mathbf{1}, \mathbf{\Sigma_1})$ and $x_i, i \in \mathbf{II} \sim \mathcal{N}(\mu_\mathbf{2}, \mathbf{\Sigma_2})$, where $\mu_1 = [0.5, -3]^T$, $\mu_2 = [-0.5, 3]^T$, and

$$\Sigma_1 = \Sigma_2 = \begin{bmatrix} 0.2 & 0 \\ 0 & 3 \end{bmatrix}.$$

Suppose the training data $\{(x_i, y_i)\}_{i=1}^m$ are independently drawn from a probability measure ρ, which is given by $\mathrm{Prob}\{y_i = 1\}$, $\mathrm{Prob}\{y_i = -1\}$ and the conditional distribution of ρ at y, i.e., $\rho(x|y = -1)$ and $\rho(x|y = 1)$. In this example, $\mathrm{Prob}\{y_i = 1\} = \mathrm{Prob}\{y_i = -1\} = 0.5$, and the contour map of the probability density functions (p.d.f.) for $\rho(x|y = -1)$ and $\rho(x|y = 1)$ is illustrated in Fig.3(a).

LS-SVM (with C large enough) corresponds to a classifier with the smallest within-class scatter, shown by solid lines in Fig.3(a). From this example, we know that the smallest within-class scatter does not always



(a) (b)

Figure 3: Contour map of p.d.f. and the diagrammatic classification results. The hyperplanes $f(x) = -1, 0, 1$ obtained from LS-SVM and L2-SVM are illustrated by solid and dashed lines, respectively. (a) noise free case; (b) noise polluted case.

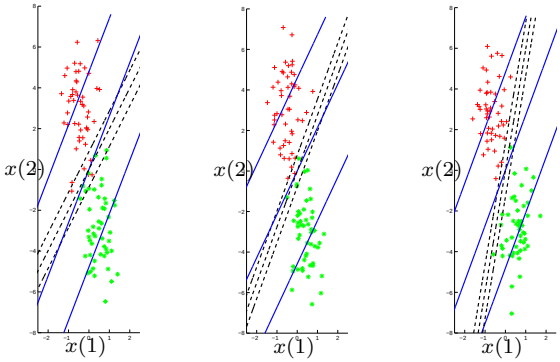
lead to a good classifier. L2-SVM (with C large enough) results in the classifier, which is illustrated by dashed lines and has a small misclassification error in this case. However, the result of L2-SVM is sensitive to noise. To show this point, we suppose that the sampling data contain the following noise. The labels of the noise points are selected from $\{1, -1\}$ with equal probability. The positions of these points follow Gaussian distribution $\mathcal{N}(\mu_n, \Sigma_n)$ with $\mu_n = [0, 0]^T$ and

$$\Sigma_n = \begin{bmatrix} 1 & -0.8 \\ -0.8 & 1 \end{bmatrix}.$$

Denoting the p.d.f. of the noise as $\rho_n(x)$, we have $\rho_n(x) = \rho_n(x|y = 1) = \rho_n(x|y = -1)$. The above noise equivalently means that the conditional distribution of ρ is polluted to be $(1-\zeta)\rho(x|y = -1)+\zeta\rho_n(x|y = -1)$ and $(1-\zeta)\rho(x|y = +1)+\zeta\rho_n(x|y = +1)$, where $\zeta \in [0, 1]$. We set $\zeta = 0.15$ and illustrate the disturbed p.d.f. by the contour map in Fig.3(b), where the corresponding classifiers obtained by LS-SVM and L2-SVM are given by solid and dashed lines, respectively. From the comparison with Fig.3(a), we can see that the result of L2-SVM is significantly affected by noise, since it focuses on the misclassification part, which is mainly caused by noise. In contrast, the within-class scatter is insensitive to noise. Generally, small within-class scatter and small training misclassification error are two desired targets for a good classifier. The proposed aLS-SVM considers both within-class scatter and misclassification error. It hence can provide a better classifier for data with noise around the decision boundary.

3.2. Stability to re-sampling

The insensitivity of aLS-SVM to noise comes from the statistical property of the expectile distance, which is also suitable for the re-sampling technique. To handle large scale problems, due to the limitation of computing time or storage space, we need to re-sample from the training set and use subsets to train a classifier. We can expect that the minimal value of $y_i f(x_i)$ is sensitive to re-sampling, which follows that the result of L2-SVM may differ a lot for different re-sampling sets. In contrast, the expectile value is more stable and so is the result of aLS-SVM. Consider three training sets drawn from the distribution in Fig.3(a). The samplings are displayed in Fig.4. Then linear L2-SVM with $C = 100$ is applied to the three data sets and the obtained classifiers are shown by black dashed lines. Though the training data come from the same distribution and there is no noise, the results of L2-SVM can be quite different. Next we use aLS-SVM with $p = 0.667$ to handle these training sets and the results are shown by blue solid lines. The comparison shows that aLS-SVM is more stable than L2-SVM to re-sampling, which coincides with the analysis for the minimal value and the expectile value.

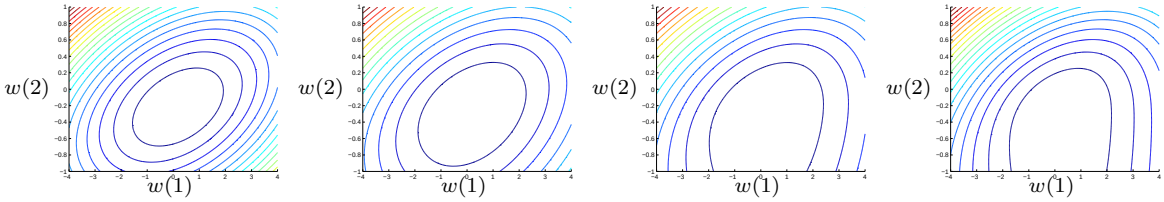


(a) (b) (c)

Figure 4: Sampling points and classification results. Points in class **I** and **II** are shown by green stars and red crosses. The data in (a), (b), and (c) are all sampled from the distribution shown in Fig.3(a). The decision boundary and the hyperplanes $w^T x + b = \pm 1$ obtained by L2-SVM are displayed by blue solid lines; while these of aLS-SVM with $p = 0.667$ are given by black dashed lines.

3.3. Computational aspects

Besides different statistical interpretations, L2-SVM and LS-SVM also have different computational burdens. L2-SVM (11) involves a constrained quadratic programming (QP), and LS-SVM (12) is related to a linear system which can be solved very efficiently. As discussed previously, aLS-SVM (6) is a trade-off between L2-SVM and LS-SVM. From this observation, we can expect that p controls the computational complexity of aLS-SVM. To give an intuitive interpretation in two dimensional figures, we omit the bias term and calculate the objective values for LS-SVM, aLS-SVM, and L2-SVM for different w values for the data displayed in Fig.4(a). The contour maps of the objective values are illustrated in Fig.5. For LS-SVM, the objective is a quadratic function and the solution can be directly found by the Newton method with a full stepsize. With an increasing value of p, the objective function becomes less similar to the quadratic function and more computation is needed.



(a) (b) (c) (d)

Figure 5: Contour map of the objective value for data in Fig.4(a). With an increasing value of p, the computational complexity increases: (a) LS-SVM (aLS-SVM with $p = 0.5$); (b) aLS-SVM with $p = 0.667$; (c) aLS-SVM with $p = 0.833$; (d) L2-SVM (aLS-SVM with $p = 1$).

For problems related to the asymmetric squared loss, one can consider the iteratively reweighted strategy. For linear expectile regression, an iteratively reweighted algorithm has been implemented by Efron (1991) and applied by Yee (2000); Kuan et al. (2009); Schnabel and Eliers (2009). Similarly, for the nonparametric aLS-SVM classifier, we establish the following iterative formulation,

$$\begin{bmatrix} b_{s+1} \\ \lambda_{s+1} \end{bmatrix} = \begin{bmatrix} 0 & Y^T \\ Y & \Omega + W^p(b_s, \lambda_s) \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ \mathbf{1} \end{bmatrix}, \tag{13}$$

where the subscript s denotes the iteration count, $\mathbf{1}$ is the vector with all components equal to one, $\Omega_{ij} = y_i y_j \mathcal{K}(x_i, x_j)$, $Y = [y_1, y_2, \ldots, y_m]^T$, and $W^p(b_s, \lambda_s)$ is the weight matrix. The weight matrix $W^p(b, \lambda)$ is diagonal and determined by the value of (10) with parameters b and λ:

$$W^p_{ii}(b, \lambda) = \begin{cases} \frac{1}{Cp}, & f(x_i) \geq 0, \\ \frac{1}{C(1-p)}, & f(x_i) < 0. \end{cases}$$

Essentially, (13) is the Newton-Raphson method for solving the optimality equations for aLS-SVM (6). The discontinuity of $W^p(b, \lambda)$ with respect to b and λ makes that the convergence of the iteratively reweighted algorithm (13) cannot be guaranteed. In practice, the convergence requires a good initial point. One can successively solve aLS-SVMs with an increasing values of p: i) apply (13) to get the solution of aLS-SVM with p_k; ii) consider a new aLS-SVM with $p_{k+1} > p_k$, which can be solved by (13) starting from the solution of aLS-SVM with p_k. We observe the convergence by setting $p_k = \frac{1}{1+\tau_k}$ with $\tau_0 = 0.5$ and $\tau_{k+1} = 0.8\tau_k$.

The properties of several SVM classifiers are summarized in Table 1, which includes sparseness, robustness to outliers, computational complexity, stability to re-sampling, and insensitivity to feature noise.

Table 1: Properties of several SVMs

	sparse	robust	complexity	stable	insensitive
L1-SVM	$\surd$	$\surd$	High	$\times$	$\times$
L2-SVM	$\surd$	$\times$	Medium	$\times$	$\times$
LS-SVM	$\times$	$\times$	Low	$\surd$	$\surd$
pin-SVM	$\times$	$\surd$	High	$\surd$	$\surd$
aLS-SVM	$\times$	$\times$	Medium	$\surd$	$\surd$

4. Numerical Examples

The purpose of aLS-SVM is to enable handling feature noise around the boundary and to pursue stability to re-sampling. In Section 3, we have illustrated its effectiveness by a linear classification problem. In the following, we consider nonparametric L2-SVM, aLS-SVM, and LS-SVM with the RBF kernel. Since LS-SVM can be solved very efficiently, we use 10 fold cross-validation based on LS-SVM (LS-SVMLab tool-box, De Brabanter et al. (2010)) to tune the parameters for RBF kernel and the parameter C. Then the obtained parameters are used in L2-SVM and aLS-SVM. We use the QP solver (interior-point algorithm) embedded in Matlab optimization tool-box to solve aLS-SVM (8) and L2-SVM (9). All the following experiments are done in Matlab R2011a in Core 2-2.83 GHz, 2.96G RAM.

First, synthetic data provided by the SVM-KM tool-box (Canu et al. (2005)) are used to evaluate the performance of aLS-SVM for re-sampling. We generate 5000 data for each data set. Then we randomly re-sample 500 data to train a classifier and use the obtained classifier to classify all the 5000 data. The re-sampling process is repeated 10 times. We illustrate the classification accuracy on the whole data by box plots in Fig.6. The mean and the standard deviation are reported in Table 2.

Table 2: Classification accuracy on the whole data set for re-sampling

Data Name	L2-SVM	aLS-SVM $p = 0.99$	aLS-SVM $p = 0.95$	aLS-SVM $p = 0.83$	LS-SVM
Clowns	85.65 ± 1.86	87.11 ± 1.04	87.13 ± 1.06	87.10 ± 1.05	86.94 ± 0.83
Checker	92.05 ± 1.29	93.47 ± 0.70	93.40 ± 0.54	93.34 ± 0.51	93.33 ± 0.57
Gaussian	91.21 ± 1.61	92.30 ± 0.40	92.30 ± 0.38	92.30 ± 0.38	92.21 ± 0.25
Cosexp	91.57 ± 2.69	94.20 ± 0.99	94.06 ± 0.87	93.96 ± 0.80	93.77 ± 0.67

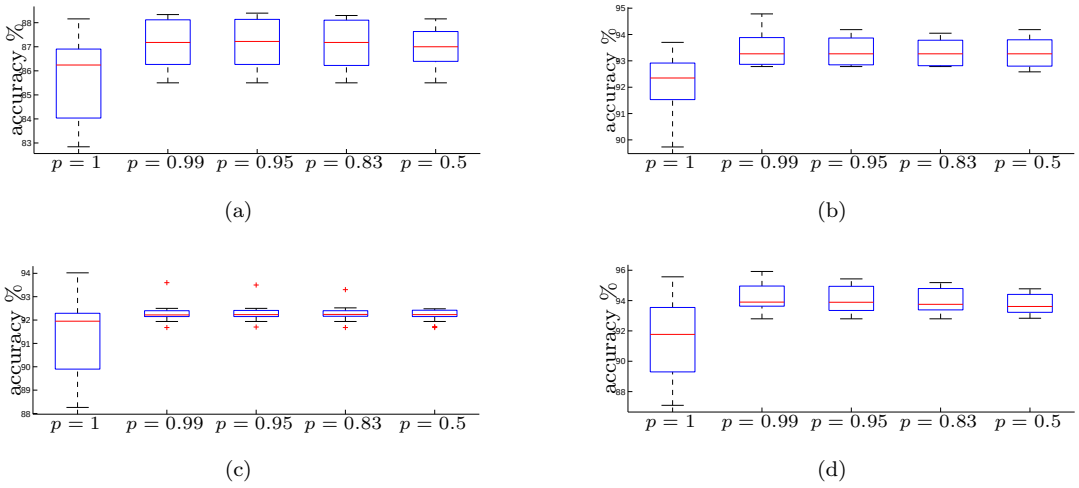


Figure 6: Box plots of the classification accuracy on the whole data set for re-sampling using L2-SVM (i.e., $p = 1$), aLS-SVM, and LS-SVM (i.e., $p = 0.5$). Each box-plot features the minimal, the lower quartile, the median, the upper quartile, and the maximal value: (a) Clowns; (b) Checker; (c) Gaussian; (d) Cosexp.

L2-SVM focuses on the minimal distance between two sets and it may lead to a good classifier for suitable re-sampling sets. For example, in our experiment on the data set "Gaussian", the highest accuracy is 94.02% and is achieved by L2-SVM for one re-sampling set. However, the performance of L2-SVM may differ a lot for different re-sampling cases, which can be observed from the standard deviation reported in Table 2. In contrast, the proposed aLS-SVM is more stable. When $p = 0.5$, i.e., LS-SVM is used, the results are the most stable. But it may be too conservative for some data sets and then introducing the flexibility of p can provide more accurate results.

Besides of re-sampling, we are also interested in the performance of aLS-SVM for feature noise. Here real-life data downloaded from the UCI Repository of Machine Learning Dataset (Frank and Asuncion (2010)) are considered. For data sets "Monk1", "Monk2", "Monk3" and "Spect", training and testing sets are provided and then we let the feature x corrupted by Gaussian noise, that means $x + \delta$ are used for training, where δ follows a normal distribution with zero mean. For each feature, we let the ratio of the variance of noise to that of the feature, denoted as r, equal to 0 (i.e., noise-free), 0.05, and 0.1. We apply L2-SVM, aLS-SVM, and LS-SVM to train the noise corrupted data and calculate the classification accuracy on testing data. Weighted least squares support vector machine (WLS-SVM, Suykens et al. (2002a)) is considered in this experiment as well. We repeat the above process 10 times and then report the mean accuracy and the standard deviation in Table 3. For other data sets, the process is the same, except that the data are randomly divided into training and testing set, both of which contain half of the data. Since the training data are randomly selected, the experiments for these data sets contain the re-sampling random factor as well. Based on the results reported in Table 3, we find that the result of aLS-SVM is not sensitive to p value. In practice, we suggest $p = 0.95$ for regular problems and a smaller value will be suitable when noise is heavy or the re-sampling size is small. WLS-SVM was proposed for sparseness and robustness. This experiment focuses on re-sampling and feature noise, for which WLS-SVM performs similarly to LS-SVM. If the data set contains outliers, one could consider a robust cross validation method given by De Brabanter et al. (2002) and explore the weighted technique of WLS-SVM to enhance the robustness of aLS-SVM.

5. Conclusion and Further Study

The basic idea of the support vector machine is to maximize the distance between two classes. The minimal distance is sensitive to noise around the decision boundary and re-sampling. In this paper, to

Table 3: Classification accuracy for noise corrupted real data set

Data Name	r	L2-SVM	aLS-SVM $p = 0.99$	aLS-SVM $p = 0.95$	aLS-SVM $p = 0.83$	LS-SVM	WLS-SVM
Monk1	0.00	80.30 ± 0.00	81.23 ± 0.00	81.06 ± 0.00	81.13 ± 0.00	81.06 ± 0.00	81.64 ± 0.00
	0.05	73.05 ± 7.01	80.64 ± 2.72	80.51 ± 2.23	80.02 ± 2.13	79.70 ± 2.09	79.03 ± 3.09
	0.10	72.71 ± 7.18	77.45 ± 2.66	77.18 ± 2.58	76.99 ± 2.79	76.92 ± 2.92	76.76 ± 2.85
Monk2	0.00	86.56 ± 0.00	87.41 ± 0.00	87.38 ± 0.00	87.43 ± 0.00	87.43 ± 0.00	84.60 ± 0.00
	0.05	81.13 ± 3.47	83.29 ± 1.49	83.29 ± 1.55	81.48 ± 1.53	81.48 ± 1.51	82.92 ± 1.72
	0.10	77.08 ± 8.81	79.72 ± 4.20	80.07 ± 4.26	79.70 ± 4.26	79.65 ± 4.37	77.80 ± 3.42
Monk3	0.00	91.91 ± 0.00	93.36 ± 0.00	92.96 ± 0.00	93.20 ± 0.00	92.01 ± 0.00	93.44 ± 0.00
	0.05	86.92 ± 11.1	91.16 ± 2.68	92.37 ± 2.66	91.30 ± 3.23	91.41 ± 3.05	91.62 ± 1.63
	0.10	80.64 ± 8.59	90.16 ± 3.08	90.32 ± 3.09	90.42 ± 3.12	90.65 ± 3.37	89.86 ± 1.05
Spect	0.00	85.03 ± 0.00	83.42 ± 0.00	84.49 ± 0.00	84.49 ± 0.00	82.96 ± 0.00	81.17 ± 0.00
	0.05	75.56 ± 5.99	81.60 ± 3.84	81.60 ± 3.84	81.60 ± 3.84	80.60 ± 2.84	80.00 ± 2.38
	0.10	71.82 ± 10.2	78.93 ± 5.47	77.81 ± 4.87	78.21 ± 5.27	77.91 ± 5.48	76.79 ± 3.67
Pima	0.00	73.80 ± 1.88	77.19 ± 1.06	77.19 ± 1.09	77.14 ± 1.11	76.12 ± 1.01	76.95 ± 1.69
	0.05	71.35 ± 4.10	77.40 ± 1.55	77.29 ± 1.48	77.34 ± 1.21	77.58 ± 1.59	77.65 ± 2.33
	0.10	71.33 ± 2.47	75.55 ± 2.56	75.52 ± 2.48	75.60 ± 2.42	74.65 ± 2.35	77.40 ± 3.68
Breast	0.00	94.85 ± 1.00	96.35 ± 0.81	96.34 ± 0.67	96.34 ± 0.67	95.31 ± 0.72	96.31 ± 1.27
	0.05	94.28 ± 1.05	96.69 ± 0.56	96.69 ± 0.59	96.69 ± 0.59	95.57 ± 0.69	93.20 ± 1.29
	0.10	91.54 ± 8.10	96.00 ± 0.63	95.80 ± 0.63	95.83 ± 0.94	95.89 ± 1.00	92.45 ± 2.91
Trans	0.00	73.70 ± 5.20	78.89 ± 1.50	81.75 ± 1.45	77.81 ± 1.47	76.07 ± 1.46	78.07 ± 1.93
	0.05	70.43 ± 5.99	77.83 ± 0.77	81.60 ± 1.87	77.78 ± 0.84	76.81 ± 0.89	77.27 ± 1.04
	0.10	69.92 ± 9.12	77.65 ± 1.44	77.65 ± 1.44	77.59 ± 1.42	76.62 ± 1.48	76.68 ± 1.80
Haber.	0.00	73.31 ± 3.04	72.29 ± 4.17	73.35 ± 4.25	72.49 ± 4.30	72.49 ± 4.33	73.27 ± 2.98
	0.05	69.16 ± 4.95	72.43 ± 2.51	72.37 ± 2.60	72.43 ± 2.69	72.31 ± 2.84	72.41 ± 2.38
	0.10	70.07 ± 3.94	72.97 ± 4.31	73.09 ± 3.14	72.79 ± 3.41	72.79 ± 3.41	72.71 ± 3.17
Iono.	0.00	90.94 ± 8.11	94.46 ± 2.03	94.51 ± 2.07	94.63 ± 2.09	94.35 ± 1.12	94.60 ± 1.14
	0.05	87.77 ± 4.24	93.20 ± 1.78	93.26 ± 1.78	93.31 ± 1.79	93.25 ± 1.78	93.08 ± 1.10
	0.10	83.91 ± 5.35	94.40 ± 1.47	94.40 ± 1.47	94.46 ± 1.45	94.46 ± 1.53	94.28 ± 1.14
Spam.	0.00	85.91 ± 3.35	89.22 ± 1.24	89.25 ± 1.26	89.13 ± 1.27	88.02 ± 1.24	87.96 ± 1.59
	0.05	83.92 ± 3.01	88.17 ± 1.18	88.19 ± 1.13	88.20 ± 1.17	88.11 ± 1.09	88.09 ± 1.82
	0.10	82.21 ± 4.52	88.53 ± 1.52	88.55 ± 1.40	88.53 ± 1.35	87.53 ± 1.28	83.88 ± 1.23
Stat.	0.00	82.59 ± 1.79	81.70 ± 3.99	81.78 ± 3.84	81.93 ± 3.62	81.62 ± 3.35	82.19 ± 3.08
	0.05	84.07 ± 2.45	84.07 ± 1.65	84.00 ± 1.53	83.85 ± 1.55	83.70 ± 1.75	83.59 ± 1.84
	0.10	83.19 ± 2.32	83.52 ± 3.68	83.52 ± 3.68	82.85 ± 2.12	82.51 ± 2.68	82.85 ± 2.23
Magic	0.00	80.24 ± 1.22	83.92 ± 1.20	83.87 ± 1.13	83.83 ± 1.08	83.02 ± 1.05	81.68 ± 0.88
	0.05	76.00 ± 1.37	83.26 ± 0.67	83.15 ± 0.71	83.08 ± 0.76	82.92 ± 0.79	77.78 ± 2.52
	0.10	72.02 ± 2.63	80.29 ± 5.04	80.28 ± 5.04	79.26 ± 5.03	79.22 ± 3.02	76.80 ± 2.95

further improve the performance of L2-SVM for noise and re-sampling, we use the expectile distance instead of the minimal distance and maximize the expectile distance between two classes to construct a classifier. The expectile value is related to the asymmetric squared loss and then asymmetric least squares support vector machine (aLS-SVM) is proposed. The dual formulation of aLS-SVM is given as well and positive definite kernels are applicable. aLS-SVM pursues a small within-class scatter and a small misclassification error, so it also can be regarded as an extension to L2-SVM or LS-SVM.

Since the expectile distance is less sensitive to noise than the minimal distance, aLS-SVM provides a more stable solution than L2-SVM. This expectation is supported by numerical experiments, where L2-SVM, LS-SVM, WLS-SVM, and aLS-SVM are compared on artificial and real-life data sets. One noticeable point is that aLS-SVM is neither sparse nor robust in view of the influence function. The lack of sparseness and robustness comes from the property of quadratic loss. Similarly, the original formulations of LS-SVM and L2-SVM are neither sparse nor robust (Steinwart (2003); Christmann and Steinwart (2004); Bartlett and Tewari (2004)). For LS-SVM, some techniques have been proposed to enhance sparseness and robustness by Suykens et al. (2002a); Valyon (2004); Abe (2007); Debruyne et al. (2010). From these studies, some experience can be learned to pursue sparseness and robustness for aLS-SVM.

Acknowledgements

The authors are grateful to anonymous reviewers for their helpful comments.

This work was supported in part by the scholarship of the Flemish Government; Research Council KUL: GOA/11/05 Ambiorics, GOA/10/09 MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC), IOF-SCORES4CHEM, several PhD/postdoc & fellow grants; Flemish Government:FWO: PhD/postdoc grants, projects: G0226.06 (cooperative systems and optimization), G.0302.07 (SVM/Kernel), G.0320.08 (convex MPC), G.0558.08 (Robust MHE), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine) research communities (WOG: ICCoS, ANMMM, MLDM); G.0377.09 (Mechatronics MPC), G.0377.12 (Structured models), IWT: PhD Grants, Eureka-Flite+, SBO LeCoPro, SBO Climaqs, SBO POM, O&O-Dsquare; Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011); IBBT; EU: ERNSI; ERC AdG A-DATADRIVE-B, FP7-HD-MPC (INFSO-ICT-223854), COST intelliCIS, FP7-EMBOCON (ICT-248940); Contract Research: AMINAL; Other: Helmholtz: viCERP, ACCM, Bauknecht, Hoerbiger. L. Shi is also supported by the National Natural Science Foundation of China (11201079). Johan Suykens is a professor at KU Leuven, Belgium.

References

Abe, S., 2007. Sparse least squares support vector training in the reduced empirical feature space. Pattern Analysis and Applications 10(3), 203–214.

Bartlett, P., Tewari, A., 2004. Sparseness versus estimating conditional probabilities: some asymptotic results. The Journal of Machine Learning Research 8, 775–790.

Bi, J., Zhang, T., 2005. Support vector classification with input data uncertainty. Advances in Neural Information Processing Systems 17, 161–168.

Canu, S., Grandvalet, Y., Guigue, V., Rakotomamonjy A., 2005. SVM and kernel methods matlab toolbox. Perception Systems et Information, INSA de Rouen, France.

Christmann, A., Steinwart, I., 2004. On robustness properties of convex risk minimization methods for pattern recognition. The Journal of Machine Learning Research 5, 1007–1034.

De Brabanter, K., Karsmakers, P., Ojeda, F., Alzate, C., De Brabanter, J., Pelckmans, K., De Moor, B., Vandewalle, J., Suykens, J.A.K., 2010. LS-SVMlab Toolbox User's Guide version 1.8, Internal Report 10-146, ESAT-SISTA, KU Leuven (Leuven, Belgium).

De Brabanter, J., Pelckmans, K., Suykens, J.A.K., Vandewalle, J., 2002. Robust cross-validation score function for non-linear function estimation. In: International Conference on Artificial Neural Networks, 713–719.

De Rossi, G., Harvey, A., 2009. Quantiles, expectiles and splines. Journal of Econometrics 152(2), 179–185.

Debruyne, M., Christmann, A., Hubert, M., Suykens, J.A.K., 2010. Robustness of reweighted least squares kernel based regression. Journal of Multivariate Analysis 101(2), 447–463.

Efron, B., 1991. Regression percentiles using asymmetric squared error loss. Statistica Sinica 1, 93–125.

Frank, A., Asuncion, A., 2010. UCI Machine Learning Repository, available from: `http://archive.ics.uci.edu/ml`.

Guyon, I., Matic, N., Vapnik, V., 1996. Discovering informative patterns and data cleaning. Advances in Knowledge Discovery and Data Mining, 181–203.

Hamid, J., Greenwood, C., Beyene, J., 2012. Weighted kernel Fisher discriminant analysis for integrating heterogeneous data. Computational Statistics and Data Analysis 56(6), 2031–2040.
Herbrich, R., Weston, J., 1999. Adaptive margin support vector machines for classification. In: International Conference on Artificial Neural Networks, 880–885.
Hu, W., Song, Q., 2004. An accelerated decomposition algorithm for robust support vector machines. IEEE Transactions on Circuits and Systems II, Express Briefs 51(5), 234–240.
Huang, X., Shi, L., Suykens, J.A.K., 2013. Support vector machine classifier with pinball loss. accepted by IEEE Transctions on Pattern Analysis and Machine Intelligence.
Koenker, R., 2005. Quantile Regression. Cambridge University Press.
Koenker, R., Zhao, Q., 1996. Conditional quantile estimation and inference for arch models. Econometric Theory 12, 793–813.
Kuan, C., Yeh, J., Hsu, Y., 2009. Assessing value at risk with care, the conditional autoregressive expectile models. Journal of Econometrics 150(2), 261–270.
Luts, J., Molenberghs, G., Verbeke, G., Van Huffel, S., Suykens, J.A.K., 2012. A mixed effects least squares support vector machine model for classification of longitudinal data. Computational Statistics and Data Analysis 56(3), 611–628.
Newey, W., Powell, J., 1987. Asymmetric least squares estimation and testing. Econometrica: Journal of the Econometric Society 55(4), 819–847.
Schnabel, S., Eilers, P., 2009. Optimal expectile smoothing. Computational Statistics and Data Analysis 53(12), 4168–4177.
Shao, Y., Deng, N., Yang, Z., 2012. Least squares recursive projection twin support vector machine for classification. Pattern Recognition 45(6), 2299–2307.
Sobotka, F., Thomas, K., 2012. Geoadditive expectile regression. Computational Statistics and Data Analysis 56(4), 755–767.
Song, Q., Hu, W., Xie, W., 2002. Robust support vector machine with bullet hole image classification. IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews 32(4), 440–448.
Steinwart, I., 2003. Sparseness of support vector machines. The Journal of Machine Learning Research 4, 1071–1105.
Steinwart, I., Christmann, A., 2008. How SVMs can estimate quantiles and the median. Advances in Neural Information Processing Systems 20 (2008) 305–312.
Steinwart, I., Christmann, A., 2011. Estimating conditional quantiles with the help of the pinball loss. Bernoulli 17(1), 211–225.
Suykens, J.A.K., De Brabanter, J., Lukas, L., Vandewalle, J., 2002. Weighted least squares support vector machines: robustness and sparse approximation. Neurocomputing 48(1-4), 85–105.
Suykens, J.A.K., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J., 2002. Least Squares Support Vector Machines. World Scientific, Singapore.
Suykens, J.A.K., Vandewalle, J., 1999. Least squares support vector machine classifiers. Neural Processing Letters 9(3), 293–300.
Taylor, J., 2008. Estimating value at risk and expected shortfall using expectiles. Journal of Financial Econometrics 6(2), 231–252.
Valyon J., Horváth, G., 2004. A sparse least squares support vector machine classifier. In: IEEE International Joint Conference on Neural Networks, 543–548.
Van Gestel, T., Suykens, J.A.K., Lanckriet, G., Lambrechts, A., De Moor, B., Vandewalle, J., 2002. Bayesian framework for least-squares support vector machine classifiers, gaussian processes, and kernel fisher discriminant analysis. Neural Computation 14(5), 1115–1147.
Vapnik, V., 1995. The Nature of Statistical Learning. Springer.
Wei, L., Chen, Z., Li, J., 2011. Evolution strategies based adaptive L_p LS-SVM. Information Sciences 181(14–15), 3000–3016.
Yee, T., 2000. Asymmetric Least Squares Quantile Regression, available from: `http://rss.acs.unt.edu/Rdoc/library/VGAM/html/alsqreg.html`.