00-03

Viaene S., Baesens B., Van Gestel T., Suykens J.A.K., Dedene D., De Moor B., Vanthienen J., ``Least squares support vector machine classifiers : An empirical evaluation", in *Proc. of the 12th Belgian-Dutch Artificial Intelligence Conference (BNAIC'00)*, Kaatsheuvel, The Netherlands, Nov. 2000, pp. 69-79., Lirias number: 178940.

An Initial Approach to Wrapped Input Selection using Least Squares Support Vector Machine Classifiers: Some Empirical Results

B. Baesens[1], S. Viaene[1], T. Van Gestel[2], J.A.K. Suykens[2]
G. Dedene[1], B. De Moor[2], J. Vanthienen[1]
[1]K.U.Leuven, Dept. of Applied Economic Sciences, LIRIS
Naamsestraat 69, B-3000 Leuven, Belgium
{Bart.Baesens; Stijn.Viaene; Guido.Dedene; Jan.Vanthienen}@econ.kuleuven.ac.be
[2]K.U.Leuven, Dept. of Electrical Engineering, ESAT-SISTA
Kardinaal Mercierlaan 94, B-3001 Leuven, Belgium
{Tony.Vangestel; Johan.Suykens; Bart.Demoor}@esat.kuleuven.ac.be

Abstract

In this paper, we evaluate least squares support vector machine (LS-SVM) classifiers with Radial Basis Function (RBF) kernels on five publicly available real-life benchmark UCI data sets. While standard SVM optimisation involves solving quadratic or linear programming problems, the least squares version corresponds to solving a set of linear equations, due to the use of equality constraints and the use of a 2-norm. Very promising results are reported indicating the good generalization behaviour of the estimated RBF LS-SVM classifiers. For many large scale real life applications least squares support vector machines in combination with the tuning technique presented in this paper may offer a fast and simple method for obtaining classifiers with good generalization performance. Furthermore, an initial approach to wrapped input selection using LS-SVM classifiers is presented which allows to reduce model complexity without decreasing the generalisation behaviour for the discussed data sets.

1 Introduction

Recently, support vector machines (SVMs) [5, 12, 13, 14] have been introduced for solving pattern recognition problems. In this method, one maps the data into a higher dimensional input space and constructs an optimal separating hyperplane in this space. This basically involves solving a quadratic programming problem, while gradient based training methods for neural network architectures on the other hand suffer from the existence of many local minima [2].

In this paper, we evaluate a least squares version of SVMs for binary classification as suggested in [10]. Due to the equality type constraints and the use

of a 2-norm in the formulation, the classifier follows from solving a set of linear equations, instead of quadratic programming for classical SVMs. The evaluation will be done on 5 publicly available data sets obtained from the University of California Irvine (UCI) repository. Finally, a wrapper input selection method will be presented as a means to reduce the number of inputs for the LS-SVM.

This paper is organised as follows. In Section 2 we elaborate on least squares support vector machine classifiers. The experimental set-up, consisting of both the calibration mechanism and the input selection method, together with the results are presented and discussed in Section 3.

2 LS-SVMs for binary classification

Given a training set $\{x_i, y_i\}_{i=1}^N$ with input data $x_i \in R^n$ and corresponding binary class labels $y_i \in \{-1, +1\}$, the SVM classifier, according to Vapnik's original formulation [5, 12, 13, 14], satisfies the following conditions:

$$\begin{cases} w^T\varphi(x_i) + b \geq +1, & \text{if } y_i = +1 \\ w^T\varphi(x_i) + b \leq -1, & \text{if } y_i = -1 \end{cases} \tag{1}$$

which is equivalent to

$$y_i[w^T\varphi(x_i) + b] \geq 1, \qquad i = 1, ..., N. \tag{2}$$

The nonlinear function $\varphi(\cdot)$: $R^n \rightarrow R^{n_h}$ maps the input space to a high dimensional (and possibly infinite dimensional) feature space. In the primal weight space the classifier then takes the form

$$y(x) = \text{sign}[w^T\varphi(x) + b], \tag{3}$$

but, on the other hand, is never evaluated in this form. One defines the optimisation problem:

$$\min_{w,b,\xi} \mathcal{J}(w, \xi) = \frac{1}{2} w^T w + c \sum_{i=1}^{N} \xi_i \tag{4}$$

subject to

$$\begin{cases} y_i[w^T\varphi(x_i) + b] \geq 1 - \xi_i, & i = 1, ..., N \\ \xi_i \geq 0, & i = 1, ..., N. \end{cases} \tag{5}$$

The variables ξ_i are slack variables which are needed in order to allow misclassifications in the set of inequalities (e.g. due to overlapping distributions). The positive real constant c should be considered as a tuning parameter in the algorithm. The solution of the classification problem is then characterised by a (convex) quadratic programming (QP) problem in the dual space [5, 12, 13, 14].

Vapnik's SVM classifier formulation was modified by Suykens [10] into the following LS-SVM formulation:

$$\min_{w,b,e} \mathcal{J}(w, e) = \frac{1}{2} w^T w + \gamma \frac{1}{2} \sum_{i=1}^{N} e_i^2 \tag{6}$$

subject to the equality constraints

$$y_i \left[w^T \varphi(x_i) + b \right] = 1 - e_i, \;\; i = 1, ..., N. \tag{7}$$

This formulation now consists of equality instead of inequality constraints and takes into account a squared error with regularization term similar to ridge regression. The solution is obtained after constructing the Lagrangian $\mathcal{L}(w, b, e; \alpha) =$

$$\mathcal{J}(w, e) - \sum_{i=1}^{N} \alpha_i \{ y_i [w^T \varphi(x_i) + b] - 1 + e_i \} \tag{8}$$

where α_i are the Lagrange multipliers. After taking the conditions for optimality, one obtains the following linear system [10]:

$$\left[\begin{array}{c|c} 0 & Y^T \\ \hline Y & \Omega + \gamma^{-1} I \end{array} \right] \left[\begin{array}{c} b \\ \hline \alpha \end{array} \right] = \left[\begin{array}{c} 0 \\ \hline \vec{1} \end{array} \right] \tag{9}$$

where $Z = [\varphi(x_1)^T y_1; ...; \varphi(x_N)^T y_N]$, $Y = [y_1; ...; y_N]$, $\vec{1} = [1; ...; 1]$, $\alpha = [\alpha_1; ...; \alpha_N]$, $\Omega = ZZ^T$ and Mercer's condition [10] is applied within the Ω matrix

$$\begin{array}{rcl} \Omega_{ij} & = & y_i y_j \, \varphi(x_i)^T \varphi(x_j) \\ & = & y_i y_j \, K(x_i, x_j). \end{array} \tag{10}$$

For the kernel function $K(\cdot, \cdot)$ one typically has the following choices: $K(x, x_i) = x_i^T x$ (linear kernel), $K(x, x_i) = (x_i^T x + 1)^d$ (polynomial kernel of degree d), $K(x, x_i) = \exp\{-\|x - x_i\|_2^2 / \sigma^2\}$ (RBF kernel), $K(x, x_i) = \tanh(\kappa \, x_i^T x + \theta)$ (MLP kernel), where d, σ, κ and θ are constants. Notice that the Mercer condition holds for all σ and d values in the RBF and the polynomial case, but not for all possible choices of κ and θ in the MLP case. The LS-SVM classifier is then constructed as follows:

$$y(x) = \text{sign} \left(\sum_{i=1}^{N} \alpha_i y_i K(x, x_i) + b \right). \tag{11}$$

Note that the matrix in (9) is of dimension $(N + 1) \times (N + 1)$. For large values of N, this matrix cannot easily be stored, such that an iterative solution method for solving it is needed. A Hestenes-Stiefel conjugate gradient algorithm is suggested in [8] to overcome this problem. Basically, the latter rests upon a transformation of the matrix in (9) to a positive definite form [8]. Iterative methods (e.g. interior point methods) for SVMs also exist but the computational efforts are still very high. Interior point methods use an iterative procedure for solving the SVM QP problem, where at each iteration step a linear system has to be solved which has the same complexity as of one single LS-SVM.

A straightforward extension of LS-SVMs to multiclass problems has been proposed in [11], where additional outputs are taken in order to encode multiple classes as is often done in classical neural networks methodology [2]. A drawback of LS-SVMs is that sparseness is lost due the choice of a 2-norm. However, this can be circumvented in a second stage by a pruning procedure which is based upon removing training points guided by the sorted support value spectrum [9].

3 Empirical Results

In this Section, we report on the application of LS-SVMs to five publicly available and often cited real-life benchmarking data sets, of which a brief description and references to past usage will also be included. The generalization ability of the obtained classifiers will be quantified by means of a calibration mechanism for tuning the LS-SVM hyperparameters. In addition, input selection is effectively introduced as a means to reduce model complexity [2, 3].

3.1 Calibration mechanism

The implementation of the LS-SVM classifier rests on the Hestenes-Stiefel conjugate gradient algorithm presented in [8]. We used RBF kernels for all data sets. The generalization ability of the LS-SVM classifier was estimated using the following procedure:

1. Set aside 2/3 of the data for the training/validation set and the remaining 1/3 for testing.
2. Perform 10-fold cross validation on the training/validation data for each (γ, σ) combination from the initial candidate tuning sets Γ and Σ typically chosen as follows :
   $\Gamma = \{0.01, 0.5, 1, 10, 50, 100, 500, 1000\}$
   $\Sigma = \{0.5, 5, 10, 15, 25, 50, 100, 250, 500\}$.
3. Choose optimal (γ, σ) from the initial candidate tuning sets Γ and Σ by looking at the best cross validation performance for each (γ, σ) combination.
4. Refine Γ and Σ iteratively by means of a grid search mechanism in order to further optimize the tuning parameters (γ, σ). In our experiments, we repeated this step three times.
5. Construct the LS-SVM classifier using the total training/validation set for the optimal choice of the tuned hyperparameters (γ, σ).
6. Assess the generalization ability by means of the independent test set.

3.2 Wrapped Input Selection

As an additional step, all inputs are pruned iteratively in order to reduce the model complexity [2, 3]. The pruning procedure is based upon a (greedy) best-first heuristic, guiding a backward search mechanism through the input space [3]. The mechanics of the heuristic are quite straightforward. We apply a strategy of constant substitution in which an input is perturbed to zero while all other inputs keep their values and compute the impact on the performance of the obtained LS-SVM classifier. For a classifier trained on k inputs, this gives rise to k reduced LS-SVM classifiers, each with $k - 1$ inputs, of which the best is then retained. The calibration procedure outlined above is thereupon resumed with a reduced input set corresponding to this latter model. Once all inputs are pruned, the

Data Set	*Model*	*Inputs*	*Train*	*Validation*	*Test*	γ	σ
Diabetes	*Full*	8	78.75	75.98	80.47	0.5	5
	Reduced	5	78.73	77.34	81.64	60	10
Liver	*Full*	6	82.90	71.74	71.30	2	3
	Reduced	5	85.89	73.48	72.17	60	3
Iono	*Full*	34	99.91	96.58	94.02	5	2
	Reduced	9	99.87	98.72	92.31	0.5	1
Heart	*Full*	13	85.56	84.44	90.00	1	10
	Reduced	7	86.60	87.22	84.44	1	20
Breast	*Full*	9	95.78	95.78	98.71	0.1	4.2
	Reduced	5	95.65	96.00	99.57	0.01	4

Table 1: Ten-fold cross validation percentages correctly classified (PCC) for Train and Validation set and independent Test set accuracy for five publicly available UCI data sets. Full and Reduced model performances are reported for each data set.

preferred reduced model is then determined by means of the highest validation set performance.

3.3 Results and Discussion

All data sets have been obtained from the publicly accessible University of California Irvine (UCI) repository at http:// kdd.ics.uci.edu/. These data sets have been referred to numerous times, which makes them excellent for benchmarking purposes. However, it falls beyond the scope of this paper to include an exhaustive overview of algoritmic performance for the diversity of techniques that are available in the literature. For a fairly thorough benchmarking study of the classification acccuracy of over thirty statistical and machine learning algorithms the reader is refered to [4]. For our purposes, every data set discussed here will be provided with some indicative published results. In this way, the reader will be able to position LS-SVMs among other well-known state of the art classification techniques.

Class labels are encoded to +1 and -1 for binary classification purposes. As a preprocessing step, all inputs, except for the Ionosphere data set (cfr. infra), have been statistically normalised by subtracting their mean and dividing by their standard deviation [2]. Furthermore, all records containing unknown values are removed from further consideration.

Table 1 summarises the results of the calibration mechanism and the input selection method outlined in the previous subsections, i.e. for the full and reduced models. Both training and validation set performance, as measured by the percentage correctly classified (PCC), have been averaged over the 10 folds of the cross-validation procedure for the optimal γ and σ values, which are also depicted in Table 1.

The **Pima Indians Diabetes** data set contains 768 records of female Pima Indians, of which 268 tested positive for diabetes. Each record is made up of 8 attributes partially describing a patient's medical history. It is considered quite a challenge on which even state of the art neural techniques still misclassify about one fourth of the population. The ADAP algorithm, a perceptron-like adaptive learning algorithm, trained on 576 examples achieved an accuracy rate of 76% on the remaining 192 elements [7]. From Table 1 it can be observed that the estimated LS-SVM classifier has an excellent generalization ability, i.c. 80.47% for the full model and 81.64% for the reduced model. Also, notice from our experiments how input selection can effectively be done without degrading test set performance.

The **BUPA Liver Disorders** data set consists of 345 instances of male patients with 6 inputs considered relevant for detecting liver disorders. Class 1 is represented by 145 instances and class 2 by 200 instances. In [1], Bennett et al. reported getting a maximal 5-fold cross validation accuracy rate of 68.7% on this data set using an SVM based decision tree method called Global Tree Optimisation. The other classification methods on which they report, among which C4.5 and standard SVM, obtain generalization within the range of $60 - 68\%$. For our experiments, the performance on the full and the reduced model respectively amounts to 71.30% and 72.17%, which is again a very good result.

The **Johns Hopkins University Ionosphere** data set represents radar returns from the ionosphere. The data set consists of 351 examples with 225 good returns and 126 bad returns. Each instance is described by 34 continuous attribute values. Sigillito et al. investigated the use of backpropagation and the perceptron training algorithm on this data set [6]. Using the first 200 instances for training, which were carefully split almost 50% positive and 50% negative, they found that a "linear" perceptron attained 90.7%. Using the same setup, a "non-linear" perceptron attained 92%, and backpropagation an average of over 96% accuracy on the remaining 150 test instances, consisting of 123 "good" and only 24 "bad" instances. Notice, however, that in our experimental setup training (including validation) and test instances were chosen randomly, giving rise to a full model test set performance of 94.02% and a reduced model test set performance of 92.31%. In this case, because all input values ranged between -1 and 1, no preliminary normalisation was performed.

The **Statlog Heart Disease** data set contains 270 instances each described by 13 inputs. The presence of heart disease was reported in 150 instances. Performance metrics on this data set for a variety of pattern recognition algorithms can be found within the Statlog repository which is accessible via the UCI world wide web pages. Compared to the results reported in the latter, the performance reported in Table 1 is again very good.

The **Wisconsin Breast Cancer** data set consists of 699 records with 9 inputs covering 458 benign examples and 241 malignant. The 16 records that contained missing values were removed from the data set. Wolberg & Mangasarian initially report on the usage of this data set for the evaluation of a linear programming based method called the Multisurface Method in [15]. Perfomance of other well-known statistical and machine learning techniques on this data set ranges between 94 – 98% [1].

For the above mentioned Γ and Σ sets, maximum generalization ability (on the independent test set) amounts to 95.61% for the full model and 95.18% for the reduced model. Further optimisation within the (γ,σ)-space by means of a refined grid search, however, reveals that for γ and σ respectively set to 0.1 and 4.2, a test set performance of 98.71% is achieved for the full model, as mentioned in Table 1. This illustrates the importance of tuning both parameters carefully so as to maximise the generalization behaviour of the constructed classifier.

In conclusion, with reference to the results reported in Table 1, it can be stated that the estimated RBF LS-SVM classifiers generalize very well on the discussed data sets, both with and without the suggested input selection method. Notice, however, that except for the last data set, parameters γ and σ have not further been optimized. Furthermore, all LS-SVM classifiers have been estimated with RBF kernels, leaving the option of the usage of other types of kernels (e.g. linear, polynomial, MLP) as an issue for future research.

4 Conclusion

We evaluated a least squares version of support vector machine classifiers. Due to the equality constraints and the use of a 2-norm in the LS-SVM formulation, the classifier reduces to solving a set of linear equations instead of a quadratic programming problem. Hence, the LS-SVM classifier can be found with low computational cost, being the solution to a convex optimisation problem, more specifically to a linear system of equations. Mercer's condition can still be applied as in other SVMs. Furthermore, efficient methods for solving large scale LS-SVMs are available. The classification accuracy of RBF LS-SVM classifiers was investigated on 5 publicly available data sets, making use of a calibration procedure with integrated cross-validation for tuning the hyperparameters of the RBF LS-SVM. Very promising results are reported indicating the excellent generalization behaviour of the estimated LS-SVM classifiers. For many real life applications least squares support vector machines may offer a fast method for obtaining classifiers with very good generalization performance. Furthermore, a method for wrapped input selection using LS-SVMs was presented and evaluated on the discussed data sets. Results indicated that for the majority of the discussed data sets, model complexity may be reduced.

Acknowledgements

This work was partly carried out at the Leuven Institute for Research in Information Systems (LIRIS) of the Dept. of Applied Economic Sciences of the K.U.Leuven in the framework of the KBC Insurance Research Chair. This work was partially carried out at the ESAT laboratory and the Interdisciplinary

Center of Neural Networks ICNN of the K.U.Leuven and supported by grants and projects from the Flemish Gov. (Res. Counc. K.U.Leuven: GOA-Mefisto; FWO-Flanders: res. projects and comm (ICCoS & ANMMM); IWT: STWW Eureka); the Belgium Gov. (IUAP-IV/02, IV-24); the Eur. Comm. (TMR, ERNSI). S. Viaene, holder of the KBC Research Chair, and B. Baesens are both Research Assistants of LIRIS T. Van Gestel is a research assistant, J. Suykens is a postdoctoral researcher, B. De Moor is a senior research associate with the Fund for Scientific Research Flanders (FWO-Flanders), resp. J. Vanthienen and G. Dedene are Senior Research Associates of LIRIS.

References

[1] K. P. Bennett and J. Blue. A support vector machine approach to decision trees. *R.P.I. Math Report, Rensselaer Polytechnic Insitute*, (97-100), 1997.

[2] C. M. Bishop. *Neural networks for pattern recognition.* Oxford University Press, 1995.

[3] G. John, R. Kohavi, and K. Pfleger. Irrelevant features and the subest selection problem. *Machine Learning: proceedings of the Eleventh International Conference*, pages 121–129, 1994.

[4] T. S. Lim and W. Y. Loh. A comparison of prediction accuracy, complexity and training time of thirty-three old and new classification algorithms. *Machine Learning*, 40:203–228, 2000.

[5] B Schölkopf, C. Burges, and A. Smola. *Advances in Kernel Methods - Support Vector Learning.* MIT Press, 1998.

[6] V. G. Sigillito, S. P. Wing, L. Hutton, and K. B. Baker. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest*, 10:262–266, 1989.

[7] J. W. Smith, J. E. Everhart, W. C. Dickson, W. C. Knowler, and R. S. Johannes. Using the adap learning algorithm to forecast the onset of diabetes mellitus. *Proceedings of the Symposium on Computer Applications an Medical Care*, pages 261–265, 1988.

[8] J. A. K. Suykens, L. Lukas, P. Van Dooren, B. De Moor, and J. Vandewalle. Least squares support vector machine classifiers: a large scale algorithm. *European Conf. on Circuits Theory and Design (ECCTD'99)*, pages 839–842, 1999.

[9] J. A. K. Suykens, L. Lukas, and J. Vandewalle. Sparse least squares support vector machine classifiers. *Proc.of the European Symposium on Artificial Neural Networks (ESANN 2000)*, 2000.

[10] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.

[11] J. A. K. Suykens and J. Vandewalle. Multiclass least squares support vector machines. *Intl. Joint Conf. on Neural Networks (IJCNN'99)*, 1999.

[12] V. Vapnik. *The nature of statistical learning theory.* Springer-Verlag, New-York, 1995.

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

[14] V. Vapnik. The support vector method of function estimation. *Nonlinear Modeling: advanced black-box techniques, Suykens J.A.K., Vandewalle J. (Eds.)*, pages 55–85, 1998.

[15] W. H. Wolberg and O. L. Mangasarian. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences*, 87:9193–9196, 1990.