Comparing Methods for Multi-class Probabilities in Medical Decision Making Using LS-SVMs and Kernel Logistic Regression

Ben Van Calster[1], Jan Luts[1], Johan A.K. Suykens[1], George Condous[2], Tom Bourne[3], Dirk Timmerman[4], and Sabine Van Huffel[1]

[1] Department of Electrical Engineering (ESAT-SISTA), Katholieke Universiteit Leuven, Kasteelpark Arenberg 10, B-3001, Leuven, Belgium
[2] Nepean Hospital, University of Sydney, Sydney, Australia
[3] St Georges Hospital Medical School, London, UK
[4] University Hospitals K.U.Leuven, Leuven, Belgium

Abstract. In this paper we compare thirteen different methods to obtain multi-class probability estimates in view of two medical case studies. The basic classification method used to implement all methods are least squares support vector machine (LS-SVM) classifiers. Results indicate that multi-class kernel logistic regression performs very well, together with a method based on ensembles of nested dichotomies. Also, a Bayesian LS-SVM method imposing sparseness performed very well for methods that combine binary probabilities into multi-class probabilities.

1 Introduction

This paper focuses on two issues: multi-class classification and probabilistic outputs. Multi-class classification is less straightforward than binary classification, in particular for margin-based classifiers such as support vector machines (SVMs). Methods for multi-class tasks are based on the combination of binary classifiers or on 'all-at-once' classification. Binary classifiers are typically constructed by contrasting all pairs of classes (1-vs-1) or by contrasting each class with all other classes combined (1-vs-All), or by other techniques such as error-correcting output coding [1].

When using binary classifiers, results are frequently combined using a voting strategy. Often, however, one is interested in class probabilities. These are important because they give information about the uncertainty of class membership as opposed to black-and-white class predictions. In medical decision making, uncertainty information can influence the optimal treatment of patients.

In this paper, we compare many methods to obtain multi-class probability estimates using two medium sized medical data sets dealing with pregnancies of unknown location and ovarian tumors. For both conditions, early probabilistic predictions are needed for optimizing patient care and its financial implications. All-at-once, 1-vs-1, and 1-vs-All methods were used. Section 2 describes these methods. Section 3 describes the data and the analysis methodology. The results

J. Marques de Sá et al. (Eds.): ICANN, Part II, LNCS 4669, pp. 139–148, 2007.

are reported and discussed in Section 4. We assume data sets with data $(\boldsymbol{x}_n, t_n)$, $n = 1, \ldots, N$, with $\boldsymbol{x}_n \in \mathbb{R}^q$ and t_n is one of k target classes ($k > 2$).

2 Obtaining Multi-class Probability Estimates

2.1 Least Squares Support Vector Machine Classifiers

The basic classification method used in this paper is the least squares support vector machine (LS-SVM) classifier [2],[3]. This variant of standard SVMs can be obtained by solving a linear system rather than a quadratic programming problem. This adjustment results, at least for standard LS-SVMs, in a lack of sparseness. However, methods to impose sparseness exist. The LS-SVM model formulation in the primal space is

$$\min_{\boldsymbol{w},b,\boldsymbol{e}} \left(\frac{1}{2}\boldsymbol{w}^T\boldsymbol{w} + \frac{1}{2}\gamma \sum_{n=1}^{N} e_n^2 \right), \text{ such that } y_n\Big[\boldsymbol{w}^T\varphi(\boldsymbol{x}_n) + b\Big] = 1 - e_n\,, \quad (1)$$

for the classifier $y(x) = \text{sign}[\boldsymbol{w}^T\varphi(\boldsymbol{x}) + b]$, where y_n is a binary class indicator (encoded as -1 vs $+1$), $\boldsymbol{w}$ is a parameter vector, b is a bias term, e_n is the error variable, and γ is a hyperparameter to control the amount of regularization. The mapping $\varphi : \mathbb{R}^q \rightarrow \mathbb{R}^r$ maps the input space into a high-dimensional feature space. By taking the Lagrangian for this problem, the classifier can be reformulated in the dual space as $y(x) = \text{sign}\big[\sum_{n=1}^{N} \alpha_n y_n K(\boldsymbol{x}, \boldsymbol{x}_n) + b\big]$, where $\alpha_1, \ldots, \alpha_N$ are the support values. In this way we implicitly work in the feature space by applying a positive definite kernel $K(\boldsymbol{x}, \boldsymbol{z}) = \varphi(\boldsymbol{x})^T\varphi(\boldsymbol{z})$. We used the radial basis function (RBF) kernel function throughout this paper. This kernel has parameter σ denoting the kernel width. Due to the 2-norm in the cost function, typically sparseness is lost.

2.2 Creating Probabilistic Output from Binary (LS-)SVMs

We implemented four techniques to get probabilistic outputs from the binary LS-SVM classifiers: a Bayesian LS-SVM [4] with and without sparseness induction, and a transformation of the LS-SVM output using either an improvement on Platt's method [5],[6] or isotonic regression [7]. For non-Bayesian LS-SVMs, the hyperparameters γ and σ have to be tuned using the training data. To do this, we applied the leave-one-out method suggested in [8] with the predicted residual sum-of-squares (PRESS) statistic.

Bayesian LS-SVM (`bay`; `bays`). MacKay's evidence procedure was used for the Bayesian LS-SVM [4],[9]. Here, two hyperparameters μ and ζ are used for the slightly modified cost function $0.5\mu\boldsymbol{w}^T\boldsymbol{w} + 0.5\zeta\sum_{n=1}^{N} e_n^2$ such that $\gamma = \zeta/\mu$. At the first level of the Bayesian procedure, the prior distribution for $\boldsymbol{w}$ and b is set to be multivariate normal. The prior is multiplied by the likelihood function to obtain the posterior. Maximization of the posterior (which is approximated by a normal distribution) yields the 'most probable' values for $\boldsymbol{w}$ and b, $\boldsymbol{w}_{\text{MP}}$ and

b_{MP}. Since the prior distribution corresponds to the regularization term $\boldsymbol{w}^T\boldsymbol{w}$ and the likelihood to the sum of the squared errors, such maximization is similar to solving an LS-SVM. On the second level, μ_{MP} and ζ_{MP} are obtained by using a uniform prior on $\log(\mu)$ and $\log(\zeta)$. At the third level, σ is updated. We denote this algorithm by `bay`. We also applied it with a sparseness procedure (`bays`). Since for LS-SVMs $\alpha_n = \gamma e_n$, support values can be negative for easy cases. Sparseness can be imposed by repeatedly pruning training data with negative support values until none are left [10].

LS-SVM + Platt (`plt`). Based on (LS-)SVM output, Platt [5] estimates the class probability $P(y=1|\boldsymbol{x}) = 1/\big(1+\exp(Af+B)\big)$, where the LS-SVM latent variable $f = \sum_{n=1}^{N} \alpha_n y_n K(\boldsymbol{x},\boldsymbol{x}_n)+b$. Training data f values were used to find A and B using maximum likelihood estimation. These f values were derived using 5-fold cross-validation (5CV) on the training data in order to obtain less biased values. We used an improved implementation of Platt's method [6].

LS-SVM + Isotonic Regression (`iso`). Zadrozny and Elkan [7] propose the use of isotonic regression to obtain probabilities based on (LS-)SVM output (f). This nonparametric regression technique maps f to probabilities using a monotonically increasing function. The isotonic regression method used is that of pooled adjacent violators [11]. The training cases are ranked with respect to f. The corresponding class membership (coded as 0 versus 1 instead of -1 versus $+1$) is taken as the initial estimated probability of belonging to class $y = +1$. If these initial probabilities are entirely separated in the ranking (i.e., they are isotonic), they are taken as the final estimates. If the initial estimates are not isotonic for two cases, both probabilities are averaged to serve as the new estimates. This is repeated until the estimated probabilities are isotonic. Training data f values used for the isotonic regression were again based on 5CV. When a new f value fell between two training data f values with different estimated probabilities, we estimated the probability for the new output value by means of simple linear interpolation.

2.3 Methods to Combine Binary Probabilities

Let us denote the probability of a case to belong to the i^{th} of k classes as $p_i = P(t=i|\boldsymbol{x})$ (estimated by $\hat{p}_i$), and the probability of a case to belong to the i^{th} class conditional on membership of class i or j as $p_{ij} = P(t=i|t\in\{i,j\},\boldsymbol{x})$, (estimated by $\hat{p}_{ij}$). To combine binary probabilities into multi-class probabilities we used ten methods that we will shortly describe.

Refregier and Vallet (`rvall`; `rvone`) [12]. Taking two classes i and j, one starts from the relation $\hat{p}_{ij}/\hat{p}_{ji} \approx p_i/p_j$ to estimate the multi-class probabilities using any $k-1$ binary classifiers. Because $\sum_{i=1}^{k} p_i = 1$ the multi-class probabilities are estimated by solving a linear system. We implemented this method once by averaging the multi-class probabilities for all subsets of $k-1$ binary classifiers (`rvall`) and once by randomly selecting one such subset (`rvall`).

Ensembles of Nested Dichotomies (`endall`; `endone`) [13]. This method obtains multi-class probabilities by advancing through a tree of binary classifiers. At the top level, all classes are divided into two subgroups for which a binary classifier is trained. Subgroups that consist of more than one class are again divided into two groups until all classes form a subgroup of their own. The tree then consists of k leaves. The multi-class probability p_i is obtained by multiplying all binary probabilities involving p_i. We implemented this method once by averaging the multi-class probabilities of all trees (`endall`) and once by randomly selecting one tree (`endone`). Of course, the number of different trees for k classes $T(k)$ grows exponentially with k such that `endall` is impractical for large classes: $T(4) = 15$ but $T(5) = 105$.

Price *et al.* (`pkpd`) [14]. This method uses the probability of a union of events, and by stating that $\hat{p}_{ij} \approx p_{ij} = p_i/(p_i + p_j)$, one ends up with $p_i \approx 1/\big[\sum_{j=1,i\neq j}^{k}(1/\hat{p}_{ij}) - (k-2)\big]$.

1-versus-All Normalization (`1van`). This method simply normalizes all obtained 1-versus-All probabilities such that they sum to 1.

Pairwise Coupling (`pc-ht`; `pc-wu1`; `pc-wu2`). The pairwise coupling method was introduced by Hastie and Tibshirani [15]. They estimate p_i by maximizing the negative weighted Kullback-Leibler distance ℓ between p_{ij} and $\hat{p}_{ij}$. After setting the derivative of ℓ with respect to p_i to zero, they aim to estimate the multi-class probabilities such that

$$\sum_{j=1,i\neq j}^{k} n_{ij}p_{ij} = \sum_{j=1,i\neq j}^{k} n_{ij}\hat{p}_{ij} \text{ with } \sum_{i=1}^{k} p_i = 1,\ p_i > 0\,. \tag{2}$$

An iterative procedure is used for this. This method is called `pc-ht`.

Wu *et al.* [16] derived two other pairwise coupling methods, based on the assumption that the multi-class data are balanced such that weighting is not necessary. They suggest to solve

$$p_i \sum_{j=1,i\neq j}^{k} \frac{p_i + p_j}{k-1}\hat{p}_{ij},\ \forall i,\ \text{with } \sum_{i=1}^{k} p_i = 1,\ p_i \geq 0\,, \tag{3}$$

which can be done by a linear system. We call this method `pc-wu1`. These authors also suggest a second approach

$$\min_{p_1,\ldots,p_k} \sum_{i=1}^{k}\sum_{j=1,i\neq j}^{k} \big(\hat{p}_{ji}p_i - \hat{p}_{ij}p_j\big)^2 \text{ with } \sum_{i=1}^{k} p_i = 1,\ p_i \geq 0\,, \tag{4}$$

which is again solved by a linear system. Note that the expression in (4) is similar to the basis of Refregier and Vallet's method [12]. We call this method `pc-wu2`.

Huang *et al.* (`gbt1va`) [17]. These authors proposed methods based on the generalized Bradley-Terry model [17]. The Bradley-Terry model estimates the probability that one object is preferred over another (in sports this can be the probability that one player beats another). In multi-class classification based on binary classifiers, $\hat{p}_{ij}$ can be seen as the probability that class i beats class j. Hastie and Tibshirani's pairwise coupling method [15] is formally similar to a Bradley-Terry model, but note that the different $\hat{p}_{ij}$ values are not independent. In the generalized Bradley-Terry model the players (classes) are repeatedly divided into two teams playing a series of games. The model is used to estimate the players' individual skill. We implemented the method where teams are created using the 1-versus-All strategy.

2.4 Duan *et al.*'s Softmax Methods (`sm1va`; `sm1v1`)

Duan *et al.* [18] propose to apply softmax functions to the outcomes of 1-versus-All or 1-versus-1 (LS-)SVMs to directly estimate multi-class probabilities. For the 1-versus-All strategy, the multi-class probabilities are computed as

$$\hat{p}_i = \exp(a_i f_i + b_i) \Bigg/ \left(\sum_{j=1}^{k} \exp(a_j f_j + b_j) \right), \tag{5}$$

where f_i is the LS-SVM output for the 1-versus-All classifier for class i, and a_i ($i = 1, \ldots, k$) and b_i ($i = 1, \ldots, k$) are parameters to be tuned by minimizing a negative log-likelihood function on the training data. For the 1-versus-1 strategy, the probabilities are computed analogously using the LS-SVM outputs for all 1-versus-1 classifiers [18]. Regularized versions were also suggested in [18] but it did not perform better in their own evaluation. The training data's 1-versus-All or 1-versus-1 LS-SVM outputs were obtained by 5CV on the training data.

2.5 Multi-class Kernel Logistic Regression (`mklr`)

Finally, an all-at-once method was used, being a weighted LS-SVM based version of multi-class kernel logistic regression [19],[20]. In standard multi-class logistic regression, the k^{th} of k classes is taken as the reference class and $k-1$ binary models are simultaneously fitted in which each class is contrasted with the reference. This leads to the multi-class probabilities

$$p_i = \exp\left(\boldsymbol{\beta}_i^T \boldsymbol{x}\right) \Bigg/ \left(1 + \sum_{j=1}^{k-1} \exp\left(\boldsymbol{\beta}_j^T \boldsymbol{x}\right) \right). \tag{6}$$

The regularized model parameters are found by optimizing the penalized log-likelihood function using iteratively regularized re-weighted least squares. This can be changed into an iteratively re-weighted LS-SVM algorithm where probabilities are computed as in (6) using $\varphi(\boldsymbol{x})$ instead of $\boldsymbol{x}$.The hyperparameters γ and σ were tuned using 10CV.

3 Experimental Setup

3.1 Data

Two real world medical datasets were used in this paper. The first data set contains 508 women with a pregnancy of unknown location (PUL), collected at St Georges Hospital in London (UK). PUL is said to occur if a woman has a positive pregnancy test but no physical signs of a pregnancy are found either intra- or extra-uterine. A PUL can turn out to be any of three outcomes ($k = 3$): a failing PUL (283 cases, 56%), a normal intra-uterine pregnancy (IUP) (183, 36%), or an ectopic pregnancy (42, 8%). The last class the most difficult one, because it is small does not seem to be well separated from both other groups. The inputs used to predict the outcome ($q = 5$) are the ratio of the hCG levels at presentation and 48 hours later, the logarithm of the average hCG level, the logarithm of the average progesterone level, vaginal bleeding, and the mother's age.

The second data set contains 754 patients with at least one ovarian tumor (in case of bilateral tumors only the worst tumor is included), collected by the International Ovarian Tumor Analysis (IOTA) study group in nine centers across Europe [21]. The type of ovarian tumor is one of the following four ($k = 4$): benign (563 cases, 75%), primary invasive (121, 16%), borderline (40, 5%), and metastatic (30, 4%). Again, the classes are clearly unbalanced. This time, the borderline class will probably be the most difficult one since it is small and contains 'borderline malignant' tumors. The inputs used to predict the outcome ($q = 9$) are the woman's age, the maximal diameter of the lesion and of the largest solid component, and the presence of i) a personal history of ovarian cancer, ii) blood flow within the papillary projections, iii) irregular internal cyst walls, iv) ascites, v) an entirely solid tumor, vi) bilateral tumors.

3.2 Performance Evaluation

The performance of all methods was evaluated using stratified 5CV [22]. The data were split up in 5 folds of equal size with equal target class distributions. Each fold served once as test set to test the methods after they were applied to the other four folds that served as the training set. This resulted in five test set results for each method. This 5CV procedure was repeated 20 times [22] such that we obtained 100 test set results. All inputs were standardized to zero mean and unit variance separately in each of the 100 training sets. The inputs in the accompanying test sets were transformed with the same formulas used for their training set colleagues.

The test set results were summarized by the area under the receiver operating characteristic (ROC) curve (AUC) [23] and by the average entropy error (log loss). The AUC indicates how well a classifier has separated two classes. Perfect classification yields an AUC of 1 while an AUC of 0.5 reflects random classification. Even though multi-class AUC measures exist, we preferred to consider a separate AUC for each class. The average entropy is computed as the average of each case's -log($\hat{p}_i$) for its true class. The higher the estimated class probability of each case's true class, the lower the average entropy. The AUC and entropy

results of the 100 test set evaluations were summarized by their median and interquartile range (IQR).

Due to the complex experimental setup, we adopted a heuristic strategy to compare methods. Based on [24], we used average ranks (AR). When comparing the ten methods to combine binary class probabilities, their performance was ranked for each of the four classification algorithms and these ranks were averaged. The opposite was done when comparing classification algorithms. This method summarized all obtained results well.

4 Results and Discussion

Because the first dataset has three classes and the second has four, we have AUC information on seven classes. To reduce the output, we averaged the k AUCs for each dataset. Lack of space precludes showing the full AUC results. The results for the PUL and IOTA data are shown in Table 1. Even though differences between methods were often not huge, the best probability combination methods were `endall` and `pc-wu1`. `pc-wu2` and `gbt1va` also performed very well. `gbt1va`'s AR of 5.0 on the IOTA dataset is caused by degraded performance for the two small classes when applying `gbt1va` to `bay`. Otherwise, this method performed excellent. The methods `pkpd`, `endone`, and `rvone` were outperformed.

Table 1. Average AUC for the different methods shown as the median over all 100 test set results after 20 stratified 5-fold CV runs (IQR between brackets)

PUL	bays	bay	plt	iso	AR	sm1va	sm1v1	mklr
endall	.966 (.015)	.959 (.024)	.961 (.020)	.961 (.024)	1.5	.963 (.024)	.960 (.025)	.964 (.018)
gbt1va	.966 (.016)	.960 (.025)	.960 (.023)	.960 (.025)	2.1			
pc-wu1	.964 (.015)	.946 (.038)	.960 (.023)	.959 (.027)	4.6			
pc-wu2	.964 (.015)	.946 (.040)	.960 (.024)	.958 (.026)	5.1			
pc-ht	.964 (.014)	.945 (.037)	.959 (.024)	.959 (.025)	5.9			
1van	.963 (.020)	.952 (.032)	.958 (.026)	.958 (.030)	6.0			
endone	.962 (.019)	.948 (.035)	.960 (.027)	.953 (.031)	6.3			
rvall	.966 (.013)	.946 (.037)	.954 (.031)	.951 (.030)	6.5			
pkpd	.964 (.016)	.945 (.053)	.958 (.025)	.956 (.028)	7.1			
rvone	.962 (.016)	.929 (.048)	.948 (.031)	.942 (.037)	9.9			
AR	1.0	3.9	2.3	2.9				
IOTA	bays	bay	plt	iso	AR	sm1va	sm1v1	mklr
endall	.881 (.029)	.859 (.037)	.869 (.027)	.868 (.029)	1.0	.862 (.032)	.861 (.034)	.883 (.027)
pc-wu1	.878 (.029)	.856 (.024)	.862 (.031)	.858 (.031)	2.5			
pc-wu2	.878 (.027)	.853 (.026)	.857 (.031)	.854 (.032)	4.1			
rvall	.877 (.029)	.857 (.026)	.857 (.034)	.851 (.032)	4.6			
gbt1va	.878 (.028)	.761 (.067)	.858 (.035)	.855 (.032)	5.0			
1van	.875 (.029)	.840 (.035)	.859 (.039)	.847 (.034)	5.8			
pc-ht	.865 (.039)	.852 (.037)	.850 (.048)	.853 (.026)	6.5			
pkpd	.874 (.029)	.847 (.030)	.849 (.036)	.845 (.033)	7.3			
endone	.873 (.032)	.835 (.038)	.837 (.055)	.831 (.031)	8.8			
rvone	.858 (.046)	.836 (.045)	.798 (.107)	.790 (.072)	9.5			
AR	1.0	3.4	2.4	3.3				

Differences in average AUC were reduced by the 'easier' classes. Bigger differences arose for the difficult classes ectopic pregnancy (PUL), borderline (IOTA), and metastatic (IOTA). Similar conclusions could be drawn when looking at

these classes only: `endall` performed best, followed by `pc-wu1`, `pc-wu2`, and `gbt1va`.

The entropy results are shown in Table 2. This leads to a similar observation: `endall`, `gbt1va`, and `pc-wu1` performed best among the probability combination methods. `pkpd`, `endone`, `rvone`, and `1van` performed worst.

With respect to Duan *et al.*'s softmax methods [18], we can see that their performance did not differ substantially, yet `sm1va` always had the advantage over `sm1v1`. Also, their performance was similar to that of the best probability combination methods. When looking at AUC performance, however, these methods seemed to perform less than the probability combination methods applied to `bays`.

Table 2. Entropy for the different methods shown as the median over all 100 test set results after 20 stratified 5-fold CV runs (IQR between brackets)

PUL	bays	bay	plt	iso	AR	sm1va	sm1v1	mklr
endall	.298 (.104)	.378 (.169)	.286 (.056)	.288 (.095)	2.0	.273 (.070)	.288 (.071)	.276 (.063)
gbt1va	.307 (.115)	.402 (.170)	.287 (.062)	.310 (.154)	3.5			
pc-wu1	.289 (.098)	.421 (.229)	.293 (.056)	.302 (.107)	3.9			
pc-ht	.291 (.092)	.430 (.225)	.292 (.058)	.299 (.085)	3.9			
pc-wu2	.293 (.100)	.427 (.260)	.292 (.054)	.312 (.115)	4.6			
rvall	.288 (.097)	.404 (.204)	.302 (.065)	.318 (.122)	4.8			
endone	.320 (.107)	.469 (.295)	.291 (.065)	.356 (.187)	7.0			
1van	.310 (.118)	.441 (.216)	.298 (.062)	.323 (.156)	7.3			
pkpd	.324 (.123)	.508 (.337)	.293 (.060)	.377 (.206)	8.6			
rvone	.313 (.129)	.561 (.419)	.308 (.067)	.412 (.224)	9.5			
AR	1.8	4.0	1.3	2.9				
IOTA	bays	bay	plt	iso	AR	sm1va	sm1v1	mklr
endall	.495 (.063)	.653 (.099)	.512 (.039)	.497 (.048)	1.0	.504 (.052)	.510 (.054)	.492 (.055)
pc-wu2	.503 (.069)	.847 (.159)	.527 (.039)	.510 (.053)	2.8			
pc-wu1	.502 (.065)	.834 (.149)	.536 (.035)	.512 (.047)	3.3			
gbt1va	.546 (.117)	.877 (.239)	.517 (.046)	.542 (.110)	5.5			
pc-ht	.530 (.074)	.873 (.134)	.545 (.045)	.518 (.042)	5.8			
rvall	.506 (.066)	.874 (.144)	.559 (.038)	.529 (.054)	5.8			
endone	.526 (.101)	1.18 (.530)	.543 (.061)	.576 (.115)	6.8			
pkpd	.526 (.082)	1.20 (.217)	.534 (.059)	.603 (.176)	7.0			
1van	.563 (.146)	1.26 (.239)	.531 (.044)	.553 (.116)	7.3			
rvone	.568 (.127)	1.41 (.664)	.612 (.182)	.764 (.331)	10.0			
AR	1.5	4.0	2.3	2.2				

Interestingly, `mklr` had excellent performance throughout. It was similar to `endall` applied to `bays`, the best approach involving a probability combination method.

The AR results for the four algorithms `bays`, `bay`, `plt`, and `iso` reveal that a Bayesian LS-SVM with sparseness was better than one without, that `bays` was better than `plt` and `iso`, and that `bay` is worse than `plt` and `iso`. Differences between `plt` and `iso` were typically in favor of `plt`. Differences in performance between probability combination methods were often smaller when `bays` or `plt` were used.

Overall, on the present data sets the best methods are `endall` and `mklr`. While the latter is an all-at-once method, the former is an ensemble method for which many binary models have to be fit (25 when $k = 4$). This makes `mklr` more

attractive. When using 1-versus-1 classifiers as the basis for obtaining multi-class probabilities, `pc-wu1` (preferably using `bays`) or `sm1v1` are good methods. When 1-versus-All classifiers are used, `sm1va` or `gbt1va` (preferably using `bays`) can be used. Out of the four implemented binary classification algorithms, `bays` seems a good choice to obtain probabilistic outputs. Using isotonic regression instead of Platt's method to obtain probabilities did not result in a clear benefit.

Acknowledgments. We thank Vanya Van Belle, Diana Sima, and Peter Karsmakers for constructive discussions. Research supported by Research Council KUL: GOA-AMBioRICS, CoE EF/05/006 Optimization in Engineering; Belgian Federal Science Policy Office IUAP P6/04 ('Dynamical systems, control and optimization', 2007-2011); EU: BIOPATTERN (FP6-2002-IST 508803), ETUMOUR (FP6-2002- LIFESCIHEALTH 503094).

References

1. Allwein, E.L., Schapire, R.E., Singer, Y.: Reducing multiclass to binary: a unifying approach for margin classifiers. J. Mach. Learn. Res. 1, 113–141 (2000)
2. Suykens, J.A.K., Vandewalle, J.: Least squares support vector machine classifiers. Neural Process. Lett. 9, 293–300 (1999)
3. Suykens, J.A.K., Van Gestel, T., De Brabanter, J., De Moor, B., Vandewalle, J.: Least squares support vector machines. World Scientific, Singapore (2002)
4. Van Gestel, T., Suykens, J.A.K., Lanckriet, G., Lambrechts, A., De Moor, B., Vandewalle, J.: Bayesian framework for least-squares support vector machine classifiers, Gaussian processes, and kernel Fisher discriminant analysis. Neural Comput. 14, 1115–1147 (2002)
5. Platt, J.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: Smola, A.J., Bartlett, P.L., Scholkopf, B., Schuurmans, D. (eds.) Advances in Large Margin Classifiers, pp. 61–74. MIT Press, Cambridge (2000)
6. Lin, H.-T., Lin, C.-J., Weng, R.C.: A note on Platt's probabilistic outputs for support vector machines. Technical Report, Department of Computer Science, National Taiwan University (2003)
7. Zadrozny, B., Elkan, C.: Transforming classifier scores into accurate multiclass probability estimates. In: Proc. 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 694–699 (2002)
8. Cawley, G.C.: Leave-one-out cross-validation based model selection criteria for weighted LS-SVMs. In: Proc. 19th International Joint Conference on Neural Networks, pp. 2970–2977 (2006)
9. MacKay, D.J.C.: Probable networks and plausible predictions - a review of practical Bayesian methods for supervised neural networks. Netw.-Comput. Neural. Syst. 6, 469–505 (1995)
10. Lu, C., Van Gestel, T., Suykens, J.A.K., Van Huffel, S., Vergote, I., Timmerman, D.: Preoperative prediction of malignancy of ovarian tumors using least squares support vector machines. Artif. Intell. Med. 28, 281–306 (2003)
11. Ayer, M., Brunk, H., Ewing, G., Reid, W., Silverman, E.: An empirical distribution function for sampling with incomplete information. Ann. Math. Stat. 26, 641–647 (1955)

12. Refregier, P., Vallet, F.: Probabilistic approach for multiclass classification with neural networks. In: Proc. International Conference on Artificial Networks, pp. 1003–1007 (1991)
13. Frank, E., Kramer, S.: Ensembles of nested dichotomies for multi-class problems. In: Proc. 21st International Conference on Machine Learning, 39 (2004)
14. Price, D., Knerr, S., Personnaz, L., Dreyfus, G.: Pairwise neural network classifiers with probabilistic outputs. In: Tesauro, G., Touretzky, D.S., Leen, T.K. (eds.) Neural Information Processing Systems, vol. 7, pp. 1109–1116. MIT Press, Cambridge (1995)
15. Hastie, T., Tibshirani, R.: Classification by pairwise coupling. Ann. Stat. 26, 451–471 (1998)
16. Wu, T.-F., Lin, C.-J., Weng, R.C.: Probability estimates for multi-class classification by pairwise coupling. J. Mach. Learn. Res. 5, 975–1005 (2004)
17. Huang, T.-K., Weng, R.C., Lin, C.-J.: Generalized Bradley-Terry models and multi-class probability estimates. J. Mach. Learn. Res. 7, 85–115 (2006)
18. Duan, K., Keerthi, S.S., Chu, W., Shevade, S.K., Poo, A.N.: Multi-category classification by soft-max combination of binary classifiers. In: Windeatt, T., Roli, F. (eds.) MCS 2003. LNCS, vol. 2709, pp. 125–134. Springer, Heidelberg (2003)
19. Karsmakers, P., Pelckmans, K., Suykens, J.A.K.: Multi-class kernel logistic regression: a fixed-size implementation. Accepted for presentation at the 20th International Joint Conference on Neural Networks (2007)
20. Zhu, J., Hastie, T.: Kernel logistic regression and the import vector machine. J. Comput. Graph. Stat., 185–205 (2005)
21. Timmerman, D., Valentin, L., Bourne, T.H., Collins, W.P., Verrelst, H., Vergote, I.: Terms, definitions and measurements to describe the sonographic features of adnexal tumors: a consensus statement from the International Ovarian Tumor Analysis (IOTA) group. Ultrasound. Obstet. Gynecol. 16, 500–505 (2000)
22. Kohavi, R.: A study of cross-validation and bootstrap for accuracy estimation and model selection. In: Proc. 14th International Joint Conference on Artificial Intelligence, pp. 1137–1143 (1995)
23. Hanley, J.A., McNeil, B.J.: The meaning and use of the area under a receiver operating characteristic (ROC) curve. Radiology 143, 29–36 (1982)
24. Brazdil, P.B., Soares, C.: A comparison of ranking methods for classification algorithm selection. In: López de Mántaras, R., Plaza, E. (eds.) ECML 2000. LNCS (LNAI), vol. 1810, pp. 63–74. Springer, Heidelberg (2000)