Fast Kernel Classifier Construction Using Orthogonal Forward Selection to Minimise Leave-One-Out Misclassification Rate

X. Hong[1], S. Chen[2], and C.J. Harris[2]

[1] Department of Cybernetics
University of Reading, Reading, RG6 6AY, U.K.
x.hong@reading.ac.uk
[2] School of Electronics and Computer Science
University of Southampton
Southampton SO17 1BJ, U.K.
{sqc, cjh}@ecs.soton.ac.uk

Abstract. We propose a simple yet computationally efficient construction algorithm for two-class kernel classifiers. In order to optimise classifier's generalisation capability, an orthogonal forward selection procedure is used to select kernels one by one by minimising the leave-one-out (LOO) misclassification rate directly. It is shown that the computation of the LOO misclassification rate is very efficient owing to orthogonalisation. Examples are used to demonstrate that the proposed algorithm is a viable alternative to construct sparse two-class kernel classifiers in terms of performance and computational efficiency.

1 Introduction

The two-class classification problems can be configured into a regression framework that solves a separating hyperplane for two classes, with the known class labels used as the desired output examples for model training in supervised learning. Models are usually identified according to some objective criteria. Information based criteria, such as the AIC [1], often include a penalty term to avoid an oversized model which may tend to overfit to the training data set. Parsimonious models are also preferable in engineering applications since a model's computational complexity scales with its model complexity. Moreover a parsimonious model is easier to interpret from the viewpoint of knowledge extraction. Consequently a practical nonlinear modelling principle is to find the smallest model that generalises well. Model construction techniques that have been widely studied include the support vector machine (SVM), relevance vector machine (RVM), and orthogonal forward regression [2,3,4,5]. The orthogonal least square algorithm [6] was developed as a practical linear-in-the-parameters models construction algorithm. A large class of nonlinear representations, e.g. the radial basis function (RBF) network and SVM, can be classified as the linear-in-the-parameters models. An orthogonal forward selection (OFS) procedure can be applied to construct parsimonious two-class classifiers by incrementally maximising the Fisher ratio of class separability measure [7,8]. Alternatively the SVM

D.-S. Huang, K. Li, and G.W. Irwin (Eds.): ICIC 2006, LNCS 4113, pp. 106–114, 2006.

is based on the structural risk minimisation (SRM) principle and approximately minimises an upper bound on the generalisation error [2] via minimising of the norm of weights in the feature space [9]. The SVM is characterised by a kernel function, lending its solution as that of the convex quadratic programming, such that the resultant model is sparse with a subset of the training data set used as support vectors.

In regression, a fundamental concept in the evaluation of model generalisation capability is that of cross validation [10]. The leave-one-out (LOO) cross validation is often used to estimate generalisation error for choosing among different model architectures [10]. LOO errors can be derived using algebraic operation rather than actually splitting the training data set for linear-in-the-parameters models. The calculation of LOO errors however is computational expensive. The generalised cross validation [11] has been introduced as a variant of LOO cross validation to improve computational efficiency. Regressors can incrementally be appended in an efficient OFS procedure while minimising the LOO errors [12,13] to yield a sparse regression model that generalises well.

This paper considers the construction of parsimonious two-class linear-in-the-parameters kernel classifiers using LOO cross validation. The proposed method extends the OFS procedure for regression in [12,13] to the classification problem by using the LOO misclassification rate, the true generalisation capability of a classifier, for model selection. Note that in classification the modelling objective is to minimise the number of misclassified samples rather than the mean square LOO error. An analytic formula for LOO misclassification rate is derived based on the regularised orthogonal least squares (ROLS) parameter estimates [13]. Furthermore it is shown that the orthogonalisation procedure brings the advantage of calculating the LOO misclassification rate via a set of forward recursive updating formula at minimal computational expense. A fast two-class kernel classifier construction algorithm is presented using this OFS procedure by directly minimising the LOO misclassification rate to optimise the model generalisation. Numerical examples are used to demonstrate the effectiveness of the proposed approach, in comparison with other current kernel based classifiers.

2 Two-Class Kernel Classifier

Consider the problem of training a two-class classifier $f(\mathbf{x}) : \Re^n \rightarrow \{1, -1\}$ based on a training data set $D_N = \{\mathbf{x}(i), y(i)\}_{i=1}^N$, where $y(i) \in \{1, -1\}$ denotes the class type for each data sample $\mathbf{x}(i) \in \Re^n$. We adopt the linear-in-the-parameters classifier given by

$$\hat{y}(i) = \text{sgn}(f(i)) \quad \text{with} \quad f(i) = \sum_{j=1}^{L} \theta_j p_j(\mathbf{x}(i)) \tag{1}$$

where $\hat{y}(i)$ is the estimated class label for $\mathbf{x}(i)$, $p_j(\bullet)$ denotes the classifier kernels with a known nonlinear basis function, θ_j are the classifier's coefficients and L is the number of kernels. The Gaussian kernel function

$$p_j(\mathbf{x}) = e^{-\frac{\|\mathbf{x}-\mathbf{c}_j\|^2}{2\sigma^2}} \tag{2}$$

is employed in this study, where $\mathbf{c}_j \in \Re^n$ is the j^{th} kernel centre and σ^2 the kernel variance. Other kernel functions can obviously be used here.

Define $\xi(i) = y(i) - f(i)$ as the modelling residual sequence. Then the model (1) over the training data set D_N can be written in vector form as

$$\mathbf{y} = \mathbf{P}\boldsymbol{\theta} + \boldsymbol{\Xi} \tag{3}$$

where $\boldsymbol{\Xi} = [\xi(1) \cdots \xi(N)]^T$, $\boldsymbol{\theta} = [\theta_1 \cdots \theta_L]^T$, and $\mathbf{P} = [\mathbf{p}_1 \cdots \mathbf{p}_L] \in \Re^{N \times L}$ is the regression matrix with column vectors $\mathbf{p}_j = [p_j(\mathbf{x}(1)) \cdots p_j(\mathbf{x}(N))]^T$. Denote the row vectors in $\mathbf{P}$ as $\mathbf{p}^T(i) = [p_1(i) \cdots p_L(i)]$, $1 \le i \le N$. Geometrically a parameter vector $\boldsymbol{\theta}$ defines a hyperplane by

$$\sum_{j=1}^{L} \theta_j p_j(\mathbf{x}) = 0 \tag{4}$$

which divides the data into two classes.

Let an orthogonal decomposition of $\mathbf{P}$ be $\mathbf{P} = \mathbf{WA}$, where $\mathbf{A} = \{a_{ij}\}$ is an $L \times L$ unit upper triangular matrix and $\mathbf{W}$ is an $N \times L$ matrix with orthogonal columns that satisfy

$$\mathbf{W}^T\mathbf{W} = \text{diag}\{\kappa_1, \cdots, \kappa_L\} \tag{5}$$

with

$$\kappa_j = \mathbf{w}_j^T\mathbf{w}_j, \ 1 \le j \le L \tag{6}$$

where $\mathbf{w}_j$ is the jth column vector of $\mathbf{W}$. The row vectors of $\mathbf{W}$ are denoted as $\mathbf{w}^T(i) = [w_1(i) \cdots w_L(i)]$, $1 \le i \le N$. The model (3) can alternatively be expressed as

$$\mathbf{y} = (\mathbf{PA}^{-1})(\mathbf{A}\boldsymbol{\theta}) + \boldsymbol{\Xi} = \mathbf{W}\boldsymbol{\gamma} + \boldsymbol{\Xi} \tag{7}$$

in which $\boldsymbol{\gamma} = [\gamma_1 \cdots \gamma_L]^T$ is an auxiliary weight vector, for which the ROLS parameter estimates are [13]

$$\gamma_j = \frac{\mathbf{w}_j^T\mathbf{y}}{\kappa_j + \lambda_j}, \quad 1 \le j \le L \tag{8}$$

where λ_j are small positive regularisation parameters. If all λ_j are set to zero, the parameter estimator is the usual least squares estimator. The original model coefficient vector $\boldsymbol{\theta}$ can be calculated from $\mathbf{A}\boldsymbol{\theta} = \boldsymbol{\gamma}$ through back-substitution.

The regularisation parameters λ_j can be optimised iteratively using an evidence procedure [14,3,13]. The following updating formula quoted from [13] are used to determine the regularisation parameters.

$$\lambda_j^{new} = \frac{\rho_j^{new}}{N - \rho^{old}} \frac{\boldsymbol{\Xi}^T\boldsymbol{\Xi}}{N - \gamma_j^2}, \quad 1 \le j \le L \tag{9}$$

where

$$\rho_j = \frac{\mathbf{w}_j^T\mathbf{w}_j}{\lambda_j + \mathbf{w}_j^T\mathbf{w}_j} \quad \text{and} \quad \rho = \sum_{j=1}^{L} \rho_j. \tag{10}$$

3 Leave-One-Out Misclassification Rate

The misclassification rate for a given two-class classifier based on (1) can be evaluated based on the misclassified data examples as

$$J = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Id}[y(i)f(i)] \tag{11}$$

where Id[•] denotes the misclassification indication function and is defined as

$$\mathrm{Id}[v] = \begin{cases} 1 \text{ if } v < 0, \\ 0 \text{ if } \ v \geq 0. \end{cases} \tag{12}$$

Cross validation criteria are metrics that measures a model's generalisation capability [10]. One commonly used version of cross-validation is the so-called LOO cross-validation. The idea is as follows. For any predictor, each data point in the estimation data set D_N is sequentially set aside in turn, a model is estimated using the remaining $(N-1)$ data, and the prediction error is derived for the data point that was removed from training. By excluding the i^{th} data example in estimation data set, the output of the model for the i^{th} data example using a model estimated by using remaining $(N-1)$ data examples is denoted as $f^{(-i)}(i)$. The associated predicted class label is calculated by

$$\hat{y}^{(-i)}(i) = \mathrm{sgn}(f^{(-i)}(i)). \tag{13}$$

It is highly desirable to derive a classifier with good generalisation capability, i.e. to derive a classifier with a minimal misclassification error rate over a new data set that is not used in model estimation. The LOO cross validation is often used to estimate generalisation error for choosing among different models [10]. The LOO misclassification rate is computed by

$$J^{(-)} = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Id}[y(i)f^{(-i)}(i)] = \frac{1}{N}\sum_{i=1}^{N} \mathrm{Id}[g(i)] \tag{14}$$

where $g(i) = y(i)f^{(-i)}(i)$. If $g(i) < 0$, this means that the i^{th} data sample is misclassified, as the class label produced by the model $f^{(-i)}$ is different from the actual class label $y(i)$.

Instead of directly calculating (13) for the predicted class labels, which requires extensive computational effort, it is shown in the following that the LOO misclassification rate can be evaluated without actually sequentially splitting the estimation data set.

4 The Proposed Fast Classifier Construction Algorithm

We propose a fast OFS kernel classifier construction algorithm that directly minimises the LOO misclassification rate (F-OFS-LOO). The LOO modelling residual is given by

$$\xi^{(-i)}(i) = y(i) - f^{(-i)}(i). \tag{15}$$

It has been shown that the LOO model residuals can be derived using an algebraic operation rather than actually splitting the training data set based on the Sherman-Morrison-Woodbury theorem [15]. For models evaluated using the ROLS parameter estimates, it can be shown that the LOO model residuals are given by [13]

$$\begin{aligned}\xi^{(-i)}(i) &= \frac{\xi(i)}{1-\mathbf{w}(i)^T[\mathbf{W}^T\mathbf{W}+\boldsymbol{\Lambda}]^{-1}\mathbf{w}(i)} \\ &= \frac{y(i)-f(i)}{1-\sum_{j=1}^{L}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}\end{aligned} \tag{16}$$

where $\boldsymbol{\Lambda}=\text{diag}\{\lambda_1,\cdots,\lambda_L\}$. Hence

$$y(i)-f^{(-i)}(i)=\frac{y(i)-f(i)}{1-\sum_{j=1}^{L}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}. \tag{17}$$

Multiplying the both sides of (17) with $y(i)$ and applying $y^2(i)=1,\forall i$, yields

$$1-y(i)f^{(-i)}(i)=\frac{1-f(i)y(i)}{1-\sum_{j=1}^{L}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}. \tag{18}$$

Thus

$$y(i)f^{(-i)}(i)=\frac{\sum_{j=1}^{L}\gamma_j w_j(i)y(i)-\sum_{j=1}^{L}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}{1-\sum_{j=1}^{L}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}. \tag{19}$$

In the following it is shown that computational expense associated with classifier regressors determination can be significantly reduced by utilising the OFS process via a recursive formula. In the OFS process, the model size is configured as a growing variable k. Consider the model construction by using a subset of k regressors ($k \ll L$), that is, a subset selected from the full model set consisting of the L initial regressors (given by (3)) to approximate the system. By replacing L with a variable model size k, and $y(i)f^{(-i)}(i)$ with $g_k(i)$, (19) is represented by

$$\begin{aligned}g_k(i) &= \frac{\sum_{j=1}^{k}\gamma_j w_j(i)y(i)-\sum_{j=1}^{k}\frac{w_j^2(i)}{\kappa_j+\lambda_j}}{1-\sum_{j=1}^{k}\frac{w_j^2(i)}{\kappa_j+\lambda_j}} \\ &= \frac{\alpha_k(i)}{\beta_k(i)}\end{aligned} \tag{20}$$

where $\alpha_k(i)$ and $\beta_k(i)$ can be represented using the following recursive formula, respectively

$$\begin{aligned}\alpha_k(i) &= \sum_{j=1}^{k}\gamma_j w_j(i)y(i)-\sum_{j=1}^{k}\frac{w_j^2(i)}{\kappa_j+\lambda_j} \\ &= \alpha_{k-1}(i)+\gamma_k w_k(i)y(i)-\frac{w_k^2(i)}{\kappa_k+\lambda_j},\end{aligned} \tag{21}$$

$$\beta_k(i) = 1 - \sum_{j=1}^{k} \frac{w_j^2(i)}{\kappa_j + \lambda_j} = \beta_{k-1}(i) - \frac{w_k^2(i)}{\kappa_k + \lambda_j}. \tag{22}$$

Thus the LOO misclassification rate for a new model with the size increased from $(k-1)$ to k is calculated by

$$J_k^{(-)} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Id}[g_k(i)]. \tag{23}$$

This is advantageous in that, for a new model whose size is increased from $(k-1)$ to k, we only need to adjust both the numerator $\alpha_k(i)$ and the denominator $\beta_k(i)$ based on those of the model size $(k-1)$, with a minimal computational effort. The Gram-Schmidt procedure is used to construct the orthogonal basis $\mathbf{w}_k$ in an OFS manner [12,13]. At the kth regression step the regressor that results in the minimum LOO misclassification rate $J_k^{(-)}$ is selected. The detailed algorithm is summarised as follows.

F-OFS-LOO based on Gram-Schmidt orthogonalisation:

1. Initialise $\alpha_0(i) = 0$ and $\beta_0(i) = 1$ for $1 \le i \le N$. Set regularisation parameters λ_j to a very small positive value λ.
2. At the kth step where $k \ge 1$, for $1 \le l \le L$, $l \ne l_1, \cdots, l \ne l_{k-1}$, compute

$$a_{jk}^{(l)} = \begin{cases} \frac{\mathbf{w}_j^T \mathbf{p}_l}{\mathbf{w}_j^T \mathbf{w}_j}, & 1 \le j < k, \\ 1, & j = k, \end{cases}$$

$$\mathbf{w}_k^{(l)} = \begin{cases} \mathbf{p}_l, & k = 1, \\ \mathbf{p}_l - \sum_{j=1}^{k-1} a_{jk}^{(l)} \mathbf{w}_j, & k \ge 2, \end{cases}$$

$$\kappa_k^{(l)} = (\mathbf{w}_k^{(l)})^T \mathbf{w}_k^{(l)},$$
$$\gamma_k^{(l)} = \frac{(\mathbf{w}_k^{(l)})^T \mathbf{y}}{\kappa_k^{(l)} + \lambda},$$

$$\alpha_k^{(l)}(i) = \alpha_{k-1}(i) + \gamma_k^{(l)} w_k^{(l)}(i) y(i) - \frac{[w_k^{(l)}(i)]^2}{\kappa_k^{(l)} + \lambda},$$
$$\beta_k^{(l)}(i) = \beta_{k-1}(i) - \frac{[w_k^{(l)}(i)]^2}{\kappa_k^{(l)} + \lambda},$$
$$g_k^{(l)}(i) = \frac{\alpha_k^{(l)}(i)}{\beta_k^{(l)}(i)},$$

for $1 \le i \le N$, and

$$J_k^{(-,\, l)} = \frac{1}{N} \sum_{i=1}^{N} \mathrm{Id}[g_k^{(l)}(i)].$$

Find

$$l_k = \arg[\min\{J_k^{(-,\,l)}, 1 \le l \le L, l \ne l_1, \cdots, l \ne l_{k-1}\}]$$

and select $J_k^{(-)} = J_k^{(-,l_k)}$, $a_{jk} = a_{jk}^{(l_k)}$ for $1 \le j \le k$, $\alpha_k(i) = \alpha_k^{(l_k)}(i)$ and $\beta_k(i) = \beta_k^{(l_k)}(i)$ for $1 \le i \le N$, and

$$\mathbf{w}_k = \mathbf{w}_k^{(l_k)} = \begin{cases} \mathbf{p}_{l_k}, & k = 1, \\ \mathbf{p}_{l_k} - \sum_{j=1}^{k-1} a_{jk}\mathbf{w}_j, & k \ge 2. \end{cases}$$

3. The procedure is monitored and terminated at the $k = n_\theta$ step, when $J_k^{(-)} \ge J_{k-1}^{(-)}$. Otherwise, set $k = k + 1$, and go to step 2.

The above procedure derives a model with $n_\theta \ll L$ regressors. Finally with a predetermined number of iterations, the procedure as given in (9) and (10) (with L replaced by n_θ) is applied to optimise the regularisation parameters that are used in the final model.

The computational complexity in the above F-OFS-LOO algorithm is in the order of $O(NL)$. The actual computation cost varies with the final model size, and the smaller the derived model size n_θ, the smaller the computation expense. When N is very large, e.g. over several thousands, a reduced subset of data points can be used as the kernel centres so that $L \ll N$ to control the computational complexity. Note that the proposed procedure for regularisation parameter optimisation is operated based on $n_\theta \ll L$ selected regressors, hence the additional computation cost involved in regularisation parameter optimisation is very small at the level $O(Nn_\theta)$.

5 Illustrative Examples

Experiments were performed to compare the proposed F-OFS-LOO algorithm with several existing classification algorithms as published in [16]. Three data sets investigated were Breast Cancer, Diabetes and Heart, which are available from [17]. Each data set contains 100 realizations of training and test data sets, respectively. Kernel classifiers were constructed over 100 training data sets and generalisation performance was evaluated using the average misclassification rate of the corresponding classifiers over the 100 test data sets. The Gaussian kernel function was employed in the experiments. Values of σ^2 were predetermined to derive individual models for each realization. For each realization of all three data sets, the full training data sets were used as the RBF centres to form the candidate regressor sets. The performance are summarised in Tables 1 to 3, respectively. The results have clearly shown that the proposed approach can construct parsimonious classifiers with competitive classification accuracy for these data sets.

6 Conclusions

Based upon the idea of using the orthogonal forward selection procedure to optimise model generalisation, a computationally efficient algorithm has been

Table 1. Average misclassification rate in % over 100 realizations of Breast Cancer test data set and classifier size. The results of first 6 methods are quoted from [16,17].

	Misclassification rate	Model Size
RBF	27.6 ± 4.7	5
Adaboost with RBF	30.4 ± 4.7	5
AdaBoost$_{Reg}$	26.5 ± 4.5	5
LP$_{Reg}$-AdaBoost	26.8 ± 6.1	5
QP$_{Reg}$-AdaBoost	25.9 ± 4.6	5
SVM with RBF kernel	26.0 ± 4.7	not available
Proposed F-OFS-LOO	25.74 ± 5	6 ± 2

Table 2. Average misclassification rate in % over 100 realizations of Diabetes test data set and classifier size. The results of first 6 methods are quoted from [16,17].

	Misclassification rate	Model Size
RBF	24.3 ± 1.9	15
Adaboost with RBF	26.5 ± 2.3	15
AdaBoost$_{Reg}$	23.8 ± 1.8	15
LP$_{Reg}$-AdaBoost	24.1 ± 1.9	15
QP$_{Reg}$-AdaBoost	25.4 ± 2.2	15
SVM with RBF kernel	23.5 ± 1.7	not available
Proposed F-OFS-LOO	23.0 ± 1.7	6 ± 1

Table 3. Average misclassification rate in % over 100 realizations of Heart test data set and classifier size. The results of first 6 methods are quoted from [16,17].

	Misclassification rate	Model Size
RBF	17.6 ± 3.3	4
Adaboost with RBF	20.3 ± 3.4	4
AdaBoost$_{Reg}$	16.5 ± 3.5	4
LP$_{Reg}$-AdaBoost	17.5 ± 3.5	4
QP$_{Reg}$-AdaBoost	17.2 ± 3.4	4
SVM with RBF kernel	16.0 ± 3.3	not available
Proposed F-OFS-LOO	15.8 ± 3.7	10 ± 3

introduced to construct sparse two-class kernel classifiers by directly minimising the leave-one-out misclassification rate. The contribution includes developing a set of forward recursive updating formula of the LOO misclassification rate in the proposed algorithm. Experimental results on three benchmark examples are used to demonstrate the effectiveness of the proposed approach.

Acknowledgement

S. Chen wishes to thank the support of the United Kingdom Royal Acedemy of Engineering.

References

1. Akaike,H.: A New Look at the Statistical Model Identification, *IEEE Trans. Automatic Control*, vol.AC-19,(1974)716–723
2. Vapnik,V.: *The Nature of Statictical Learning Theory*, New York: Springer-Verlag, (1995)
3. Tipping,M.E.:Sparse Bayesian Learning and the Relevance Vector Machine, *J. Machine Learning Research*, vol.1,(2001)211–244
4. Scholkopf,B., Smola,A.J.:*Learning with Kernels: Support Vector Machine, Regularization, Optimization and Beyond*, Cambridge, MA: MIT Press, (2002)
5. Hong,X., Harris,C.J .: Nonlinear Model Structure Design and Construction using Orthogonal Least Squares and D-optimality Design, *IEEE Trans. Neural Networks*, vol.13, no.5,(2001)1245–1250
6. Chen,S., Billings, S.A.,W. Luo.: Orthogonal Least Squares Methods and Their Applications to Non-linear System Identification,*Int. J. Control*, vol.50, (1989)1873–1896
7. Mao,K.Z.: RBF Neural Network Center Selection Based on Fisher Ratio Class Separability Measure, *IEEE Trans. Neural Networks*, vol.13, no.5, (2002)1211–1217
8. Chen,S., Wang,X.X., Hong,X., Harris,C.J.: Kernel classifier construction using orthogonal forward selection and boosting with fisher ratio class separability, *IEEE Trans. Neural Networks*, accepted for publication, 2006
9. Vapnik,V.: *Statistical Learning Theory: Adaptive & Learning Systems for Signal Processing, Communication & Control*, J. Wiley, 1998
10. Stone,M.: Cross Validatory Choice and Assessment of Statistical Predictions, *Applied Statistics*, vol.36, pp.117–147, 1974.
11. Golub,G.H., Heath,M., Wahba,G.: Generalized cross-validation as a method for choosing good ridge parameter, *Technometrics*, vol.21, no.2, 1979(215–223)
12. X. Hong, P.M. Sharkey, K. Warwick.:Automatic Nonlinear Predictive Model Construction using Forward Regression and the PRESS Statistic, *IEE Proc. - Control Theory and Applications*, vol.150, no.3,(2003)245–254
13. Chen,S., Hong,X., Harris, C.J., P.M. Sharkey.: Sparse modelling using orthogonal Forward Regression with PRESS Statistic and Regularization, *IEEE Trans. Systems, Man and Cybernetics, Part B: Cybernetics*, vol.34, no.2,(2004)898–911
14. Mackay,D.J.:Bayesian Interpolation, *Neural Computation*, vol.4, no.3, (1992)415–447
15. Myers,R.H.: *Classical and Modern Regression with Applications*, PWS-KENT, Boston, 2nd edition, (1990)
16. G. Rätsch, T. Onoda and K.R. Müller, Soft Margins for AdaBoost, *Machine Learning*, vol.42, no.3, (2001)287–320
17. `http://ida.first.fraunhofer.de/projects/bench/benchmarks.htm`