©Springer-Verlag

Probabilistic Discriminative Kernel Classifiers for Multi-class Problems

Volker Roth

University of Bonn
Department of Computer Science III
Roemerstr. 164
D-53117 Bonn
Germany
roth@cs.uni-bonn.de

Abstract. Logistic regression is presumably the most popular representative of probabilistic discriminative classifiers. In this paper, a kernel variant of logistic regression is introduced as an iteratively re-weighted least-squares algorithm in kernel-induced feature spaces. This formulation allows us to apply highly efficient approximation methods that are capable of dealing with large-scale problems. For multi-class problems, a pairwise coupling procedure is proposed. Pairwise coupling for "kernelized" logistic regression effectively overcomes conceptual and numerical problems of standard multi-class kernel classifiers.

1 Introduction

Classifiers can be partitioned into two main groups, namely *informative* and *discriminative* ones. In the informative approach, the classes are described by modeling their structure, i.e. their generative statistical model. Starting from these class models, the posterior distribution of the labels is derived via the Bayes formula. The most popular method of informative kind is classical *Linear Discriminant Analysis* (LDA). However, the informative approach has a clear disadvantage: modeling the classes is usually a much harder problem than solving the classification problem directly.

In contrast to the informative approach, discriminative classifiers focus on modeling the decision boundaries or the class probabilities directly. No attempt is made to model the underlying class densities. In general, they are more robust as informative ones, since less assumptions about the classes are made. The most popular discriminative method is *logistic regression* (LOGREG), [1]. The aim of logistic regression is to produce an estimate of the posterior probability of membership in each of the c classes. Thus, besides predicting class labels, LOGREG additionally provides a probabilistic confidence measure about this labeling. This allows us to adapt to varying class priors.

A different approach to discriminative classification is given by the *Support Vector* (SV) method. Within a maximum entropy framework, it can be viewed as the discriminative model that makes the least assumptions about the estimated model parameters,

cf. [2]. Compared to LOGREG, however, the main drawback of the SVM is the absence of probabilistic outputs.[1]

In this paper, particular emphasis is put on a nonlinear "kernelized" variant of logistic regression. Compared to kernel variants of Discriminant Analysis (see [5, 6]) and to the SVM, the kernelized LOGREG model combines the conceptual advantages of both methods:

1. it is a discriminative method that overcomes the problem of estimating class conditional densities;
2. it has a clear probabilistic interpretation that allows us to quantify a confidence level for class assignments.

Concerning multi-class problems, the availability of probabilistic outputs allows us to overcome another shortcoming of the SVM: in the usual SVM framework, a multi-class problem with c classes is treated as a collection of c "one-against-all-others" subproblems, together with some principle of combining the c outputs. This treatment of multi-class problems, however, bears two disadvantages, both of a conceptual and a technical nature: (i) separating one class from all others may be an unnecessarily hard problem; (ii) all c subproblems are stated as quadratic optimization problems over the *full* learning set. For large-scale problems, this can impose unacceptably high computational costs.

If, on the other hand, we are given posterior probabilities of class membership, we can apply a different multi-class strategy: instead of solving c one-against-all problems, we might solve $c(c-1)/2$ pairwise classification problems, and try to couple the probabilities in a suitable way. Methods of this kind have been introduced in [7] and are referred to as *pairwise coupling*. Since kernelized LOGREG provides us with estimates of posterior probabilities, we can directly generalize it to multi-class problems by way of "plugging" the posterior estimates into the pairwise coupling procedure.

The main focus of this paper concerns pairwise coupling methods for *kernel classifiers*[2]. For kernel-based algorithms in general, the computational efficiency is mostly determined by the number of training samples. Thus, pairwise coupling schemes also overcome the numerical problems of the one-against-all strategy: it is much easier to solve $c(c-1)/2$ small problems that to solve c large problems. This leads to a reduction of computational costs that scales linear in the number of classes. We conclude this paper with performance studies for large-scale problems. These experiments effectively demonstrate that kernelized LOGREG attains a level of accuracy comparable to the SVM, while additionally providing the user with posterior estimates for class membership. Moreover, concerning the computational costs for solving multi-class problems, it outperforms one of the best SVM optimization packages available.

[1] Some strategies for approximating SVM posterior estimates in a post-processing step have been reported in the literature, see e.g. [3, 4]. In this paper, however, we restrict our attention to fully probabilistic models.

[2] A recent overview over kernel methods can be found in [8].

2 Kernelized Logistic Regression

The problem of classification formally consists of assigning observed vectors $\boldsymbol{x} \in \boldsymbol{R}^d$ into one of c classes. A *classifier* is a mapping that assigns labels to observations. In practice, a classifier is trained on a set of observed i.i.d. data-label pairs $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^N$, drawn from the unknown joint density $p(\boldsymbol{x}, y) = p(y|\boldsymbol{x})p(\boldsymbol{x})$. For convenience in what follows, we define a *discriminant function* for class k as

$$g_k(\boldsymbol{x}) = \log \frac{P(y = k|\boldsymbol{x})}{P(y = c|\boldsymbol{x})}, \quad k = 1, \ldots, c-1. \tag{1}$$

Assuming a *linear* discriminant function leads us to the logistic regression (LOGREG) model: $g_k(\boldsymbol{x}) = \boldsymbol{w}_k^T \boldsymbol{x}$.[3]

Considering a **two-class** problem with labels $\{0, 1\}$, it is sufficient to represent $P(1|\boldsymbol{x})$, since $P(0|\boldsymbol{x}) = 1 - P(1|\boldsymbol{x})$. Thus, we can write the "success probability" in the form

$$\pi_{\boldsymbol{w}}(\boldsymbol{x}) := P(1|\boldsymbol{x}) = \frac{1}{1 + \exp\{-(\boldsymbol{w}^T \boldsymbol{x})\}}. \tag{2}$$

For discriminative classifiers like LOGREG, the model parameters are chosen by maximizing the *conditional log-likelihood*:

$$l(\boldsymbol{w}) = \sum_{i=1}^{N} \left[y_i \log \pi_{\boldsymbol{w}}(\boldsymbol{x}_i) + (1 - y_i) \log(1 - \pi_{\boldsymbol{w}}(\boldsymbol{x}_i)) \right]. \tag{3}$$

In order to find the optimizing weight vector $\boldsymbol{w}$, we wish to solve the equation system $\nabla_{\boldsymbol{w}} l(\boldsymbol{w}) = \boldsymbol{0}$. Since the $\pi_i := \pi_{\boldsymbol{w}}(\boldsymbol{x}_i)$ depend nonlinearly on $\boldsymbol{w}$, however, this system cannot be solved analytically and iterative techniques must be applied. The *Fisher scoring* method updates the parameter estimates $\boldsymbol{w}$ at the r-th step by

$$\boldsymbol{w}_{r+1} = \boldsymbol{w}_r - \{E[H]\}^{-1} \nabla_{\boldsymbol{w}} l(\boldsymbol{w}), \tag{4}$$

with H being the Hessian of l.[4] The scoring equation (4) can be restated as an *Iterated Re-weighted Least Squares* (IRLS) problem, cf. [10]. Denoting with X the design matrix (the rows are the input vectors), the Hessian is equal to $(-X^T W X)$, where W is a diagonal matrix:

$$W = \text{diag}\ \{\pi_1(1 - \pi_1), \ldots, \pi_N(1 - \pi_N)\}.$$

The gradient of l (the scores) can be written as $\nabla_{\boldsymbol{w}} l(\boldsymbol{w}) = X^T W \boldsymbol{e}$, where $\boldsymbol{e}$ is a vector with entries $e_j = (y_j - \pi_j)/W_{jj}$. Forming a variable

$$\boldsymbol{q}_r = X \boldsymbol{w}_r + \boldsymbol{e},$$

[3] Throughout this paper we have dropped the constant b in the more general form $g(\boldsymbol{x}) = \boldsymbol{w}^T \boldsymbol{x} + b$. We assume that the data vectors are augmented by an additional entry of one.

[4] For LOGREG, the Hessian coincides with its expectation: $H = E[H]$. For further details on Fisher's method of scoring the reader is referred to [9].

the scoring updates read

$$(X^T W_r X)\boldsymbol{w}_{r+1} = X^T W_r \boldsymbol{q}_r. \tag{5}$$

These are the normal form equations of a least squares problem with input matrix $W_r^{1/2} X$ and dependent variables $W_r^{1/2}\boldsymbol{q}_r$. The values W_{ii} are functions of the actual $\boldsymbol{w}_r$, so that (5) must be iterated.

Direct optimization of the likelihood (3), however, often leads to severe overfitting problems, and a preference for smooth functions is usually encoded by introducing *priors* over the weights $\boldsymbol{w}$. In a regularization context, such prior information can be interpreted as adding some *bias* to maximum likelihood parameter estimates in order to reduce the estimator's variance. The common choice of a spherical Gaussian prior distribution with covariance $\Sigma_w \propto \lambda^{-1} I$ leads to a *ridge regression* model, [11]. The regularized update equations read

$$(X^T W_r X + \lambda I)\boldsymbol{w}_{r+1} = X^T W_r \boldsymbol{q}_r. \tag{6}$$

The above equation states LOGREG as a regularized IRLS problem. This allows us to extend the linear model to nonlinear kernel variants: each stage of iteration reduces to solving a system of linear equations, for which it is known that the optimizing weight vector can be expanded in terms of the input vectors, cf. [5]:

$$\boldsymbol{w} = \sum_{i=1}^{N} \boldsymbol{x}_i \alpha_i = X^T \boldsymbol{\alpha}. \tag{7}$$

Substituting this expansion of $\boldsymbol{w}$ into the update equation (6) and introducing the dot product matrix $(K)_{ij} = (\boldsymbol{x}_i \cdot \boldsymbol{x}_j)$, $K = XX^T$, we can write

$$(KW_r K + \lambda K)\boldsymbol{\alpha}_{r+1} = KW_r \boldsymbol{q'}_r, \tag{8}$$

with $\boldsymbol{q'}_r = K\boldsymbol{\alpha}_r + \boldsymbol{e}$. Equation (8) can be simplified to

$$(K + \lambda W_r^{-1})\boldsymbol{\alpha}_{r+1} = \boldsymbol{q'}_r. \tag{9}$$

With the usual kernel trick, the dot products can be substituted by kernel functions satisfying Mercer's condition. This leads us to a nonlinear generalization of LOGREG in kernel feature spaces which we call **kLOGREG**.

The matrix $(K+\lambda W_r^{-1})$ is symmetric, and the optimizing vector $\boldsymbol{\alpha}_{r+1}$ can be computed in a highly efficient way by applying approximative *conjugate gradient* inversion techniques, see cf. [12], p. 83. The availability of efficient approximation techniques from the well-studied field of numerical linear algebra constitutes the main advantage over a related approach to kLOGREG, presented in [13]. The latter algorithm computes the optimal coefficients α_i by a sequential approach. The problem with this on-line algorithm, however, is the following: for each new observation $\boldsymbol{x}_t,\ t = 1, 2, \ldots$ it imposes computational costs of the order $\mathcal{O}(t^2)$. Given a training set of N observations in total, this accumulates to an $\mathcal{O}(N^3)$ process, for which to our knowledge no efficient approximation methods are known.

3 Pairwise Coupling for Multi-class Problems

Typically two-class problems tend to be much easier to learn than multi-class problems. While for two-class problems only one decision boundary must be inferred, the general c-class setting requires us to apply a strategy for coupling decision rules. In the standard approach to this problem, c two-class classifiers are trained in order to separate each of the classes against all others. These decision rules are then coupled either in a probabilistic way (e.g. for LDA) or by some heuristic procedure (e.g. for the SVM).

A different approach to the multi-class problem was proposed in [7]. The central idea is to learn $c(c-1)/2$ pairwise decision rules and to couple the pairwise class probability estimates into a joint probability estimate for all c classes. It is obvious, that this strategy is only applicable for pairwise classifiers with probabilistic outputs.[5] From a theoretical viewpoint, pairwise coupling bears some advantages: (i) jointly optimizing over all c classes may impose unnecessary problems, pairwise separation might be much simpler; (ii) we can select a highly specific model for each of the pairwise subproblems.

Concerning *kernel classifiers* in particular, pairwise coupling is also attractive for practical reasons. For kernel methods, the computational cost are dominated by the size of the training set, N. For example, conjugate gradient approximations for kLOGREG scale as $\mathcal{O}(N^2 \cdot m)$, with m denoting the number of conjugate-gradient iterations. Keeping m fixed leads us to a $\mathcal{O}(N^2)$ dependency as a lower bound on the real costs.[6] Let us now consider c classes, each of which contains N_c training samples. For a one-against-all strategy, we have costs scaling as $\mathcal{O}(c(cN_c)^2) = \mathcal{O}(c^3(N_c)^2)$. For the pairwise approach, this reduces to $\mathcal{O}(1/2\, c(c-1)(2N_c)^2) = \mathcal{O}(2(c^2-c)(N_c)^2)$. Thus, we have a reduction of computational costs inverse proportional to the number of classes.

Pairwise coupling can be formalized as follows: considering a set of events $\{A_i\}_{i=1}^c$, suppose we are given probabilities $r_{ij} = \text{Prob}\,(A_i | A_i \text{ or } A_j)$. Our goal is to couple the r_{ij}'s into a set of probabilities $p_i = \text{Prob}\,(A_i)$. This problem has no general solution, but in [7] the following approximation is suggested: introducing a new set of auxiliary variables $\mu_{ij} = \frac{p_i}{p_i + p_j}$, we wish to find $\hat{p}_i$'s such that the corresponding $\hat{\mu}_{ij}$'s are in some sense "close" to the observed r_{ij}'s. A suitable closeness measure is the Kullback-Leibler divergence between r_{ij} and $\hat{\mu}_{ij}$

$$\mathcal{D}^{KL} = \sum_{i<j} r_{ij} \log \frac{r_{ij}}{\hat{\mu}_{ij}} + (1 - r_{ij}) \log \frac{1 - r_{ij}}{1 - \hat{\mu}_{ij}}. \tag{10}$$

The associated score equations read

$$\sum_{j \neq i} \hat{\mu}_{ij} = \sum_{j \neq i} r_{ij}\,, \quad i = 1, \ldots, c, \quad \text{subject to } \sum_i p_i = 1. \tag{11}$$

[5] In a former version of [7], available as Tech. Rep. at the University of Toronto, it has been suggested to apply "approximative" pairwise coupling to the SVM. However, we feel that this approach is not very promising since it lacks a clear probabilistic interpretation.

[6] For the SVM, the situation is more difficult and heavily depends on implementation details. As an example, the popular LOQO package [14], has even $\mathcal{O}(N^3)$ complexity, due to a Cholesky decomposition of a $N \times N$ matrix. Subset methods are usually much more efficient, but their performance is problem-dependent, and thus difficult to analyze.

Starting with an initial guess for the $\hat{p}_i$ and corresponding $\hat{\mu}_{ij}$, we can compute the $\hat{p}_i$'s that minimize (10) by iterating

- $\hat{p}_i \leftarrow \hat{p}_i \cdot \big(\sum_{j \neq i} r_{ij}\big) / \big(\sum_{j \neq i} \hat{\mu}_{ij}\big)$
- renormalize the $\hat{p}_i$'s and recompute the $\hat{\mu}_{ij}$.

Suppose, we have successfully trained all pairwise kLOGREG classifiers. Then, we can predict the class membership of a new observation $\boldsymbol{x}_*$ as follows:

1. Evaluate the $c(c-1)/2$ classification rules to obtain

$$r_{ij}(\boldsymbol{x}_*) = \text{Prob}\,(\boldsymbol{x}_* \in \text{class } i \,|\boldsymbol{x}_* \in \text{class } i \text{ or } \boldsymbol{x}_* \in \text{class } j),$$

   and initialize $\hat{\mu}_{ij} = r_{ij}$.
2. Starting with an initial guess for the $\hat{p}_i$'s, run the above iterations.
3. We finally obtain the posterior probabilities for class membership of pattern $\boldsymbol{x}_*$.

4 Experiments

Here we present results for the "MPI chairs", and "Isolet" datasets.[7] In both cases the number of classes is relatively high: the chair dataset consists of downscaled images from 25 different classes of chairs; the Isolet dataset contains spoken names of all 26 letters of the alphabet. We compared both the prediction accuracy and the computational costs of a "state-of-the-art" SVM package[8] and kLOGREG. The results are summarized in table 1. We conclude, that pairwise coupled kLOGREG attains a level of prediction accuracy comparable to the SVM, while imposing significantly lower computational costs. Concerning the training times, the reader should notice that we are comparing the highly tuned *SVMTorch* optimization package with our straight-forward kLOGREG implementation, which we consider to yet possess ample opportunities for further optimization.

Table 1. Test error rates (e) and computation times (t) on the **MPI chair** and the **Isolet** database. c = number of classes, N_c = number of samples per class, $Idim$ = input dimension, N_T = size of test set. The training times (t) are measured on a 500 MHz PC.

Dataset		c	N_c	$Idim$	N_T	**SVM**		**kLOGREG**	
Chairs	Images	25	89	256	2500	e = 1.48%	t = 152 s	e = 1.52%	t = 53 s
	Images + edges	25	89	1280	2500	e = 0.80%	t = 19 min	e = 0.84%	t = 3:05 min
Isolet		26	240	617	1560	e = 3.0%	t = 21 min	e = 3.0%	t = 18 min

[7] Available via `ftp://ftp.mpik-tueb.mpg.de/pub/chair_dataset` and `http://www.ics.uci.edu/~mlearn/MLRepository.html` respectively.

[8] We used the *SVMTorch II* V1.07 implementation, see [15].

5 Conclusion

In this paper we have presented a new approach to multi-class classification with kernel methods. In particular, we have focused on a kernelized variant of classical logistic regression, which we name *kLOGREG*. The kLOGREG classifier combines the advantages of related versions of kernel methods: it is a *discriminative* classifier that overcomes the problem of estimating class models, and it has a clear *probabilistic interpretation*. We have stated kLOGREG as an iteratively re-weighted least-squares problem in kernel feature spaces. The real payoff of this algorithmic formulation is the applicability of highly efficient approximation techniques from the well-studied field of numerical linear algebra.

Concerning multi-class problems, we can use the kLOGREG classifier as a building block in a *pairwise coupling* procedure. The main idea of pairwise coupling is to couple all pairwise decision rules into an estimate for the posterior probability of class membership. Besides of conceptual advantages over classical ways of handling multi-class problems, this technique additionally has a clear numerical advantage: for a fixed number of training patterns, the computational costs reduce linearly in the number of classes.

Experiments for large-scale problems with many classes have effectively demonstrated that kLOGREG attains a level of accuracy comparable to the SVM. Moreover, concerning the computational costs, our straight-forward implementation which basically uses routines from *Numerical Recipes*, [12], outperformed one of the best SVM packages available. We thus conclude that kLOGREG is a highly suited method for dealing with multi-class problems that require us to quantify the uncertainty about the predicted class labels.

Acknowledgments. The author wishes to thank J. Buhmann, M. Braun and L. Hermes for fruitful discussions. Thanks for financial support go to German Research Council (DFG).

References

1. D. R. Cox and E. J. Snell. *Analysis of Binary Data*. Chapman & Hall, London, 1989.
2. T. Jaakkola, M. Meila, and T. Jebara. Maximum entropy discrimination. In S.A. Solla, T.K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 470–476. MIT Press, 1999.
3. P. Sollich. Probabilistic methods for support vector machines. In S.A. Solla, T.K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 349–355. MIT Press, 1999.
4. L. Hermes, D. Frieauff, J. Puzicha, and J. Buhmann. Support vector machines for land usage classification in Landsat TM imagery. In *Proc. of the IEEE 1999 International Geoscience and Remote Sensing Symposium*, volume 1, pages 348–350, 1999.
5. V. Roth and V. Steinhage. Nonlinear discriminant analysis using kernel functions. In S.A. Solla, T.K. Leen, and K.-R. Müller, editors, *Advances in Neural Information Processing Systems*, volume 12, pages 568–574. MIT Press, 1999.

6. S. Mika, G. Rätsch, J. Weston, B. Schölkopf, and K.-R. Müller. Fisher discriminant analysis with kernels. In Y.-H. Hu, J. Larsen, E. Wilson, and S. Douglas, editors, *Neural Networks for Signal Processing IX*, pages 41–48. IEEE, 1999.
7. Trevor Hastie and Robert Tibshirani. Classification by pairwise coupling. In Michael I. Jordan, Michael J. Kearns, and Sara A. Solla, editors, *Advances in Neural Information Processing Systems*, volume 10. The MIT Press, 1998.
8. K.-R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf. An introduction to kernel-based learning algorithms. *IEEE Transactions on Neural Networks*, 12(2):181–201, March 2001.
9. M.R. Osborne. Fisher's method of scoring. *Internat. Statistical Review*, 60:99–117, 1992.
10. I. Nabney. Efficient training of RBF networks for classification. Technical Report NCRG/99/002, Aston University, Birmingham, UK., 1999.
11. A.E. Hoerl and R.W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.
12. W.H. Press, S.A. Teukolsky, W.T Vetterling, and B.P. Flannery. *Numerical Recipes in C*. Cambridge University Press, 1992.
13. T. Jaakkola and D. Haussler. Probabilistic kernel regression models. In David Heckerman and Joe Whittaker, editors, *Procs. 7th International Workshop on AI and Statistics*. Morgan Kaufmann, 1999.
14. R. J. Vanderbei. LOQO: An interior point code for quadratic programming. *Optimization Methods and Software*, 11:451–484, 1999.
15. Ronan Collobert and Samy Bengio. Support vector machines for large-scale regression problems. Technical Report IDIAP-RR-00-17, IDIAP, Martigny, Switzerland, 2000.