ESANN'2009 proceedings, European Symposium on Artificial Neural Networks - Advances in Computational Intelligence and Learning. Bruges (Belgium), 22-24 April 2009, d-side publi., ISBN 2-930307-09-9.

Semi-supervised Bipartite Ranking with the Normalized Rayleigh Coefficient

Liva Ralaivola *

Laboratoire d'Informatique Fondamentale de Marseille
39, rue F. Joliot Curie, F-13013 Marseille - France

Abstract. We propose a new algorithm for semi-supervised learning in the bipartite ranking framework. It is based on the maximization of a so-called normalized Rayleigh coefficient, which differs from the usual Rayleigh coefficient of Fisher's linear discriminant in that the actual covariance matrices are used instead of the scatter matrices. We show that if the class conditional distributions are Gaussian, then the ranking function produced by our algorithm is the optimal linear ranking function. A kernelized version of the proposed algorithm and a semi-supervised formulation are provided. Preliminary numerical results are promising.

1 Introduction

We tackle the problem of learning a ranking function in the bipartite ranking framework. We, in particular, are interested in the problem of performing such learning in a semi-supervised setting, where both labelled data and unlabelled data are in the training set.

If a considerable amount of work has been devoted to semi-supervised classification – see the comprehensive references [1, 2, 3] for instance –, the literature semi-supervised ranking thereof, is much less abundant (see notable contributions [4, 5, 6]). Yet, being able to learn a ranking function from partially labelled data poses issues of the utmost interest from the theoretical, algorithmic and practical points of view. The present work essentially lies in the algorithmic side of the learning problem.

We propose a very straightforward learning algorithm for learning a ranking function, based on the *normalized Rayleigh coefficient*. This coefficient differs from the one that is usually used by Fisher's discriminant analysis in that it is not based on the scatter matrices of the data but on their covariances, i.e. the *normalized* scatter matrices. The learning algorithm we propose can take into account unlabelled data using the idea of manifold regularization [7]. This is an important feature that we take advantage of to provide an effective semi-supervised ranking procedure.

The paper is organized as follows. Section 2 introduces the notation and presents the problem we address. In Section 3, we describe our approach to learn a ranking function using the normalized Rayleigh coefficient in a semi-supervised setting. Section 4 reports preliminary simulation results on the problem of ranking handwritten digits.

2 Problem

From here on, the following notation is used. $\mathcal{X} \subset \mathbb{R}^d$ is the input space and $\mathcal{Y} = \{-1, +1\}$ is the target/output space. $S_l = \{(X_i, Y_i)\}_{i=1}^n$ is an identically and indepen-

*This work is partially supported by the IST Program of the EC, under the FP7 Pascal2 Network of Excellence, ICT-216886-NOE.

dently distributed (iid) sample of n *labelled* variables over $\mathcal{Z} := \mathcal{X} \times \mathcal{Y}$, with fixed and unknown distribution $D_{\mathcal{Z}}$ on $\mathcal{Z}$. $S_u = \{X_{n+i}\}_{i=1}^{m}$ is an iid sample of m *unlabelled* data distributed according to $D_{\mathcal{X}}$, the marginal probability of $D_{\mathcal{Z}}$ with respect to $\mathcal{X}$, i.e. $D_{\mathcal{X}}(\cdot) = \sum_{y \in \mathcal{Y}} D_{\mathcal{Z}}(\cdot, Y = y)$. S_X is the sample $S_X = \{X_i\}_{i=1}^{n+m}$. D_y, for $y \in \mathcal{Y}$, is the distribution over $\mathcal{X}$ defined as $D_y(\cdot) = D_{\mathcal{Z}}(\cdot | Y = y)$. I is the identity matrix.

The learning problem that we tackle is that of bipartite ranking where the aim is to learn a function $f \in \mathbb{R}^{\mathcal{X}}$ that minimizes the ranking risk R^{rank} defined as follows[1]:

$$R^{\text{rank}}(f) := \mathbb{P}_{\substack{X^+ \sim D_{+1} \\ X^- \sim D_{-1}}} \left(f(X^+) \leq f(X^-)\right) = \mathbb{E}_{\substack{(X,Y) \sim D \\ (X',Y') \sim D}} \left[\mathbb{I}_{f(X) \leq f(X')} | Y = +1, Y' = -1\right].$$

We are specifically interested in learning in a semi-supervised setting, where f is learned from a sample $S = S_l \cup S_u$ of labelled and unlabelled data. To this end, we propose a simple and effective learning strategy inspired by quadratic discriminant classifiers [8].

3 Normalized Rayleigh Coefficient and Manifold Regularization

3.1 Quadratic and Linear Discriminant

Quadratic discriminant analysis (QDA) is primarily a classification model [8]; it makes the assumption that each class is normally distributed, i.e., $D_y(\cdot) = \mathcal{N}(\cdot; \boldsymbol{\mu}_y, \Sigma_y)$, where $\boldsymbol{\mu}_y$ and Σ_y are the mean and covariance parameters of the distribution. For sake of completeness, we recall that

$$\mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Sigma) = \frac{1}{(2\pi)^{\frac{d}{2}} |\Sigma|^{\frac{1}{2}}} \exp\left(-\frac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \Sigma^{-1} (\mathbf{x} - \boldsymbol{\mu})\right),$$

where $^\top$ denotes the matrix/vector transpose. Learning a QDA consists in estimating $\boldsymbol{\mu}_y$ and Σ_y using their maximum likelihood (ML) estimates $\hat{\boldsymbol{\mu}}_y$ and $\hat{\Sigma}_y$. Given a realization of the labelled training sample $S_l = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n}$, these are defined by

$$\hat{\boldsymbol{\mu}}_y = \frac{1}{n^y} \sum_{i: y_i = y} \mathbf{x}_i, \text{ and } \hat{\Sigma}_y = \frac{1}{n^y} \sum_{i: y_i = y} (\mathbf{x}_i - \hat{\boldsymbol{\mu}}_y)(\mathbf{x}_i - \hat{\boldsymbol{\mu}}_y)^\top \tag{1}$$

where n^y is the number of instances of class y in S_l. Regularized versions of the matrices $\hat{\Sigma}_y$ may be preferred over the plain ML estimates: a stricly positive definite matrix, e.g. λI with $\lambda > 0$, may be added to the ML estimates [9].

Once these estimates are computed, together with the estimates $\hat{\pi}_y = n^y / n$ of the prior probabilities for each class, the decision of the quadratic classifier for an example $\mathbf{x}$ is made according to (recall that we consider the binary case)

$$\mathbb{P}(Y = 1 | X = \mathbf{x}) = \frac{\hat{\pi}_{+1} \mathcal{N}(\mathbf{x}; \hat{\mu}_{+1}, \hat{\Sigma}_{+1})}{\hat{\pi}_{+1} \mathcal{N}(\mathbf{x}; \hat{\mu}_{+1}, \hat{\Sigma}_{+1}) + \hat{\pi}_{-1} \mathcal{N}(\mathbf{x}; \hat{\mu}_{-1}, \hat{\Sigma}_{-1})} \tag{2}$$

such as the predicted class is $+1$ if $\mathbb{P}(Y = 1 | X = \mathbf{x}) \geq 1/2$ and -1 otherwise; this classifier is 'quadratic' because, after rearranging the terms of (2), the decision rule

[1] This is 1 minus the well known expected Area under the ROC Curve (AUC).

depends on the computation of a quantity like $\mathbf{x}^\top A\mathbf{x} + B\mathbf{x} + c$, for A, B matrices, A positive definite, and $c \in \mathbb{R}$. This classifier is obviously Bayes optimal if the classes are indeed Gaussian distributed. We note that QDA is a generalization of the Fisher discriminant analysis (FDA) [10] as it assumes class-dependent covariance matrices.

3.2 From QDA to Ranking via the Normalized Rayleigh Coefficient

Even if QDA is essentially a classification algorithm, its estimation process (1) may serve to learn a *linear* ranking function f of the following form: $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$:

Proposition 1. *If $(D_y)_{y\in\mathcal{Y}}$ are Gaussian with parameters $\boldsymbol{\mu}_y, \Sigma_y$, then*

$$\mathbf{w}^* = \operatorname*{argmin}_{\mathbf{w}} \rho(\mathbf{w}), \text{ with } \rho(\mathbf{w}) := \frac{(\mathbf{w}^\top \boldsymbol{\mu}_{+1} - \mathbf{w}^\top \boldsymbol{\mu}_{-1})^2}{\mathbf{w}^\top (\Sigma_{+1} + \Sigma_{-1})\, \mathbf{w}}$$

realizes the minimum ranking risk.

Proof. For a linear ranking function f with $f(\mathbf{x}) = \mathbf{w}^\top \mathbf{x}$, we have:

$$\begin{aligned} R^{\text{rank}}(f) &= \mathbb{P}_{X^+,X^-}(f(X^+) \le f(X^-)) = \mathbb{P}_{X^+,X^-}(w^\top(X^+ - X^-) \le 0) \\ &= \frac{1}{(2\pi)^{\frac{d}{2}}\sqrt{\sigma_+^2(\mathbf{w}) + \sigma_-^2(\mathbf{w})}} \int_{-\infty}^{0} \exp\left(-\frac{1}{2}\left[\frac{z - (\mu_{+1}(\mathbf{w}) - \mu_{-1}(\mathbf{w}))}{\sqrt{\sigma_{+1}^2(\mathbf{w}) + \sigma_{-1}^2(\mathbf{w})}}\right]^2\right) dz \\ &= \frac{1}{(2\pi)^{\frac{d}{2}}} \int_{-\infty}^{-\frac{\mu_{+1}(\mathbf{w}) - \mu_{-1}(\mathbf{w})}{\sqrt{\sigma_{+1}^2(\mathbf{w}) + \sigma_{-1}^2(\mathbf{w})}}} \exp\left(-\frac{u^2}{2}\right) du, \end{aligned}$$

where $\mu_y(\mathbf{w}) := \mathbf{w}^\top \boldsymbol{\mu}_y$ and $\sigma_y^2(\mathbf{w}) := \mathbf{w}^\top \Sigma_y \mathbf{w}$; we used that $X^+ - X^-$ is distributed as $\mathcal{N}(X^+ - X^-; \boldsymbol{\mu}_{+1} - \boldsymbol{\mu}_{-1}, \Sigma_{+1} + \Sigma_{-1})$ and, for Z distributed according to $\mathcal{N}(Z; \boldsymbol{\mu}_z, \Sigma_z)$, $\mathbf{w}^\top Z$ is distributed according to $\mathcal{N}(\mathbf{w}^\top Z; \mathbf{w}^\top \boldsymbol{\mu}_z, \mathbf{w}^\top \Sigma_z \mathbf{w})$. This directly gives that $R^{\text{rank}}(f)$ is minimized when $-\frac{\mu_{+1}(\mathbf{w}) - \mu_{-1}(\mathbf{w})}{\sqrt{\sigma_{+1}^2(\mathbf{w}) + \sigma_{-1}^2(\mathbf{w})}}$ $(= -\sqrt{\rho(\mathbf{w})})$ is minimal, which occurs for $\mathbf{w}^*$. □

A supervised learning procedure. This result calls for a specific algorithm given a labelled sample S_l. It simply consists in computing the empirical estimates of the mean and covariances as in (1), forming the empirical counterpart $\hat{\rho}(\mathbf{w})$ of $\rho(\mathbf{w})$ as

$$\hat{\rho}(\mathbf{w}) := \frac{(\mathbf{w}^\top \hat{\boldsymbol{\mu}}_{+1} - \mathbf{w}^\top \hat{\boldsymbol{\mu}}_{-1})^2}{\mathbf{w}^\top \left(\hat{\Sigma}_{+1} + \hat{\Sigma}_{-1}\right) \mathbf{w}} = \frac{(\hat{\mu}_{+1}(\mathbf{w}) - \hat{\mu}_{-1}(\mathbf{w}))^2}{\hat{\sigma}_{+1}^2(\mathbf{w}) + \hat{\sigma}_{-1}^2(\mathbf{w})}, \tag{3}$$

and finding $\hat{\mathbf{w}}$ such that $\hat{\mathbf{w}} = \operatorname{argmin}_{\mathbf{w}} \hat{\rho}(\mathbf{w})$.

The coefficient $\hat{\rho}$ is very similar to the Rayleigh coefficient considered in FDA. There is, however, a *slight but very important* difference between the usual Rayleigh coefficient and the proposed coefficient: here, the matrices that are used in the denominator of (3) are the actual covariance matrices whereas FDA uses scatter matrices,

i.e. the unnormalized covariances. This is the reason why we term $\hat{\rho}$ the *normalized Rayleigh coefficient*. For the same reasons as with FDA, a solution for $\hat{\mathbf{w}}$ is

$$\hat{\mathbf{w}} = (\hat{\Sigma}_{+1} + \hat{\Sigma}_{-1})^{-1}(\hat{\boldsymbol{\mu}}_{+1} - \hat{\boldsymbol{\mu}}_{-1}), \tag{4}$$

while any vector positively colinear to $\hat{\mathbf{w}}$ defines the same ranking.

Noticeable features. If the distributions of the classes are Gaussian then QDA is Bayes optimal for classification *and* directly provides a way to compute the optimal linear ranking function. Besides, as noted in [11], minimizing the ranking error and the misclassification error is not necessarily achieved by the same procedure (note the difference between the classification using (2) and the linear scoring entailed by (4)); however, we note that FDA and (4) provide the same vectors if the classes are balanced or if the covariances are equal. Finally, the learning procedure that we have shown here does not build instances made of pairs of positive/negative data to learn a ranking function as do many ranking algorithms (with the notable exception of [12]).

3.3 Kernels and Semi-Supervised Learning

We now consider the problem of semi-supervised ranking from a partially labelled sample $S = S_u \cup S_l$ and the introduction of kernels to learn nonlinear ranking functions.

Even if it is straightforward to reformulate (3) with kernel functions in a way similar to Mika et al. [10] for FDA, we do not undertake this strategy. We instead implement a kernel subspace projection strategy similar to that of the Kernel Projection Machines [13]. Namely, given a kernel k, and $\phi(\mathbf{x}) = k(\mathbf{x}, \cdot)$, a sample $S = S_l \cup S_u$ and a target dimension p, we project the data of S_X onto the subspace V_p of dimension p generated by the Incomplete Kernel Gram-Schmidt Orthogonalization (see [14] for details) of $\phi(S_X)$, the images $\phi(\mathbf{x})$ by ϕ of the $\mathbf{x}$'s from S. We thus end up with a semi-supervised learning problem where $\mathcal{X} \subset \mathbb{R}^p$. The learning procedure can be limited to the search for a linear ranking function on the transformed data. Note that [13] uses the space spanned by the p first eigenvectors of the covariance of the $\phi(\mathbf{x})$'s; for p fixed, our strategy is computationally less demanding while it still provides good results.

In addition, we constrain the smoothness of the ranking function with respect to the geodesics of $D_{\mathcal{X}}$ by using the normalized Laplacian regularizer proposed by [7]: given a number c, we first construct the symmetric graph of c nearest (wrt e.g., the euclidean norm) neighbors of S (we do this before the projection step), form its adjacency matrix W and compute the normalized Laplacian $n \times n$ matrix $L = I - D^{-1/2} W D^{-1/2}$, where D is the diagonal matrix with elements $D_{ii} = \sum_j W_{ij}$ and use L as a regularizer in (3).

Semi-supervised nonlinear learning algorithm. Given a partially labelled sample S, a kernel k, ϕ, such that $\phi(\mathbf{x}) = k(\mathbf{x}, \cdot)$, $c \in \mathbb{N}$, $p \in \mathbb{N}$, $\lambda \geq 0$ and $\gamma \geq 0$ the complete learning algorithm can be described as follows

1. Compute L using c as the nearest neighbor parameter.
2. Perform an Incomplete kernel Gram Schmidt Orthogonalization of $\phi(S_X)$ using k to get a subspace V_p of dimension p and an orthonormal basis $\{\mathbf{v}_1, \ldots, \mathbf{v}_p\}$.

3. Form the new partially labelled training set $S^p = S^p_l \cup S^p_u$ with $S^p_l = \{(X^p_i, Y_i)\}_{i=1}^n$, $S^p_u = \{X^p_{n+i}\}_{i=1}^m$ and $X^p_i = [\mathbf{v}_1^\top \phi(X_i) \cdots \mathbf{v}_p^\top \phi(X_i)]^\top, i = 1, \ldots, n+m$; note that the computations simply require kernel evaluations.

4. Compute the estimates $\hat{\boldsymbol{\mu}}^p_y$ and $\hat{\Sigma}^p_y$ with respect to S^p_l according to (1).

5. Letting $\mathbf{X}^\top = [X^p_1 \cdots X^p_{n+m}]$, compute a linear ranking/scoring function $\hat{\mathbf{w}}$ as[2]:

$$\hat{\mathbf{w}} = \left(\lambda I + \hat{\Sigma}^p_{+1} + \hat{\Sigma}^p_{-1} + \frac{\gamma}{(n+m)^2}\mathbf{X}^\top L \mathbf{X}\right)^{-1} (\hat{\boldsymbol{\mu}}^p_{+1} - \hat{\boldsymbol{\mu}}^p_{-1}). \quad (5)$$

To predict the score of X, it thus suffices to compute X^p and then $f(X) = \hat{\mathbf{w}}^\top X^p$.

The $\mathbf{X}^\top L \mathbf{X}$ term of (5) makes the predictions of $\hat{\mathbf{w}}$ be smooth with respect to the geodesics of the data, as in [7]: the scores computed by $\mathbf{w}$ are given by $\mathbf{X}\mathbf{w}$, which entails a $\mathbf{w}^\top \mathbf{X}^\top L \mathbf{X} \mathbf{w}$ term in the denominator of (3), which, in turn, leads to (5).

4 Numerical Illustration

We have carried out simulations on the OPTDIGITS problem from the UCI repository. We have considered the bipartite ranking situations where (a) digit '0' is of class +1 and all other digits are of class -1, and (b) digits '0' to '4' are of class +1 and all other digits are of class -1. The dataset set is made of ~ 3900 training patterns (~ 390 patterns per digit) and 1800 test patterns.

We have made arbitrary choices for some hyperparameters: p is set to $p = 10$ (which is very small), the nearest neighbor parameter c to $c = 2$, and $\lambda = 0.001$. The width of the Gaussian kernel we use and the value of γ are set according to the performance of the learning procedure on a validation set (a part of the training set that is not used for learning). In the following table we report the AUCs (=1-$R^{\text{rank}}(f)$) computed on the independent test patterns for various ratios of labelled data.

% Labelled	10	20	30	40	50	60	70	80	90	100
0 vs. all	99.5	99.5	99.5	99.5	99.5	99.5	99.5	99.5	99.5	99.5
0-4 vs 5-9	70.6	72.2	73.1	77.0	77.1	79.5	80.2	81.1	81.2	81.2

The results are good even in the situation where the ratio of labelled data is very low, particularly for the 0 vs all problem, which is an 'easy' problem (see [15]). As for the 0-4 vs 5-9 problem, the results are still very competitive with the situation where all the data are labelled (last column), even when very few data are labelled. If we compare our preliminary results to those given in [15], we observe that the results of our semi-supervised learning are among the top ranking methods for this particular problem.

5 Conclusion

We have proposed a ranking algorithm inspired by the idea of QDA. We have shown that if the class conditional distributions are Gaussian, then our algorithm provides the

[2] Notice that the matrix to be inverted is of order p.

best linear ranking function. The use of kernels makes our method a sensible strategy to provide nonlinear ranking functions and the problem of learning from partially labelled data is made possible by resorting to the well-known method of manifold regularization. Preliminary empirical resutls are very encouraging.

As a first extension to this work, we plan to carry out more intensive simulations on various datasets. Then, we would like to establish the statistical performances of the regularization through the subspace projection method that we have used. Finally, we will try to see how the theoretical results of [16] on generalization bounds on the AUC can be used to select the hyperparameters of the presented algorithm.

References

[1] O. Chapelle, B. Schölkopf, and A. Zien, editors. *Semi-Supervised Learning*. The MIT Press, 2006.

[2] X. Zhu. Semi-supervised learning literature survey. Technical Report 1530, Computer Sciences, University of Wisconsin-Madison, 2005.

[3] M. Seeger. Learning with labeled and unlabeled data. Technical report, Institute for ANC, Edinburgh, UK, 2000.

[4] M.-R. Amini, T.-V. Truong, and C. Goutte. A Boosting Algorithm for Learning Bipartite Ranking Functions with Partially Labeled Data. In *Proc. of SIGIR 2008*, pages 99–106, 2008.

[5] K. Duh and K. Kirchhoff. Learning to rank with partially-labeled data. In *Proc. of SIGIR 2008*, pages 251–258, 2008.

[6] S. Hoi and R. Jin. Semi-supervised ensemble ranking. In *Proc. of the 23rd AAAI Conf. on Artificial Intelligence*, 2008.

[7] M. Belkin, P. Niyogi, and V. Sindhwani. Manifold regularization: a geometric framework for learning from examples. *JMLR*, 7(Nov):2399–2434, 2006.

[8] K. Fukunaga. *Introduction to Statistical Pattern Recognition.* Academic Press, 1990.

[9] J. H. Friedman. Regularized discriminant analysis. *Journal of the American Statistical Association*, 84:165–175, 1989.

[10] 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.

[11] C. Cortes and M. Mohri. AUC optimization vs. error rate minimization. In *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2004.

[12] Y. Freund, R. Iyer, R.E. Schapire, and Y. Singer. An efficient boosting algorithm for combining preferences. *Journal of Machine Learning Research*, 4:933–969, 2003.

[13] L. Zwald, R. Vert, G. Blanchard, and P. Massart. Kernel projection machine: a new tool for pattern recognition. In *Adv. in Neural Information Processing Systems*, volume 17, 2004.

[14] N. Cristianini, J. Shawe-Taylor, and H. Lodhi. Latent Semantic Kernels. *Journal of Intelligent Information Systels*, 18(2–3):127–152, 2002.

[15] A. Rakotomamonjy. SVMs and area under roc curves. Technical report, PSI- INSA de Rouen, 2004.

[16] L. Ralaivola, M. Szafranski, and G. Stempfel. Chromatic PAC-Bayes Bounds for non-IID Data. In *Proc. of AISTATS 09*, 2009.