Max Margin Dimensionality Reduction

Gal Chechik, Google Research
1600 Amphitheater way, Mountain View, CA, 94303

February 8, 2008

A fundamental problem in machine learning is to extract compact but relevant representations of empirical data. Relevance can be measured by the ability to make good decisions based on the representations, for example in terms of classification accuracy. Compact representations can lead to more human-interpretable models, as well as improve scalability. Furthermore, in multi-class and multi-task problems, learning a unified input representation that is shared among classes or tasks can reduce the number of free parameters and sample-complexity. Sharing representation is also a powerful method for transfer learning [4, 11], where the representation learned from some tasks is used to facilitate learning other tasks.

The naive but common practice for finding a compact representation is an unsupervised pre-processing phase (like clustering or PCA) followed by classification in the reduced space. A preferable approach is to learn the features simultaneously with the classifier. Several methods approached this problem, including the well known *Fisher's Linear Discriminant Analysis* (LDA), and its variants [7], learning metrics for kNN [9, 8], and extracting features for multi-task classification [3, 2].

Here we focus on finding low-dimensional linear projections that are optimized for support vector machines, in a single- or multi- task setting. Formaly, we have $k = 1, \ldots, K$ binary classification tasks, each with labeled data $\mathbf{x}_{i_k}^k \in \mathbb{R}^D$, $y_{i_k}^k \in \{-1, +1\}$, $\forall i_k = 1, \ldots, n_k$, we look for a rank-$d$ linear projection $\mathbf{A}$, $d \leq D$, that is shared across all tasks, and K classifiers $\mathbf{w}^k$, one for each task. We optimize jointly over all the classifiers $\mathbf{w} = \left\{\mathbf{w}^k\right\}$ and the shared projection matrix $\mathbf{A}$,

$$\min_{\mathbf{A}} \min_{\mathbf{w},\xi,b} \mathcal{L}(\mathbf{w}, b, \mathbf{A}) = \frac{1}{2} \sum_{k=1}^{K} \|\mathbf{w}^k\|^2 + C \sum_{i=1}^{n} \xi_{i_k}^k \quad \text{s.t.} \quad \mathbf{A}\mathbf{A}^T = I_d, \quad y_{i_k}(\mathbf{w}^k \mathbf{A} \mathbf{x}_{i_k}^k + b) \geq 1 - \xi_{i_k}^k, \quad \forall k, i_k \,. \tag{1}$$

Note that we use a rank constraint (L_0) on A, even though rank constraints often make optimization hard, and are commonly relaxed to L_1 sparsity constraints using trace regularization [1, 3, 5].

Optimizing directly over both $\mathbf{A}$ and $\mathbf{w}$ is inherently harder than standard SVM problems since the constraints $\mathbf{wA}$ are not jointly convex in $\mathbf{w}$ and $\mathbf{A}$. Instead, we derive the dual of (1) w.r.t. $\mathbf{w}$.

$$\min_{\mathbf{A}} \max_{\alpha} \qquad \mathcal{L}(\mathbf{A}, \alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{A}\mathbf{x}_i, \mathbf{A}\mathbf{x}_j) y_j \alpha_j, \quad \text{s.t.} \quad \mathbf{A}\mathbf{A}^T = I_d \tag{2}$$

where $0 \leq \alpha_i \leq C$ are the standard Lagrange coefficient. Since the primal is not convex, the duality gap is not guaranteed to vanish (weak duality) but in practice we show below that the dual provides good solutions.

The problem in (2) is a saddle point, having a global optimum if it is concave in α and convex in $\mathbf{A}$. Concavity in α is guaranteed since for any given $\mathbf{A}$, K is a positive semi-definite kernel, and a minimization over a set of concave functions is also concave. Convexity w.r.t. $\mathbf{A}$ is much harder to guarantee, hence we address two important special cases, linear and RBF kernels.

The case of linear kernel may first seem degenerated since the combination of two linear operators $\mathbf{A}$ and $\mathbf{w}$ is equivalent to a single linear operator. However, this is not the case when $\mathbf{A}$ is shared across multiple classification tasks. We rewrite Eq. (2) for linear kernels $K(\mathbf{A}\mathbf{x}_i, \mathbf{A}\mathbf{x}_j) = (\mathbf{A}\mathbf{x}_i)^T \mathbf{A}\mathbf{x}_j = Tr\left(\mathbf{x}_i^T \mathbf{A}^T \mathbf{A} \mathbf{x}_j\right)$, using a single-task notation for simlpicity. Using properties of the trace $Tr\left(\mathbf{x}_i^T \mathbf{A}^T \mathbf{A} \mathbf{x}_j\right) = Tr\left(\mathbf{A}\mathbf{x}_j \mathbf{x}_i^T \mathbf{A}^T\right)$, we have

$$\min_{\mathbf{A}} \max_{\alpha} \sum_{i=1}^{n} \alpha_i + \frac{1}{2} \sum_{y_i \neq y_j} \alpha_i Tr\left(\mathbf{A}\mathbf{x}_j \mathbf{x}_i^T \mathbf{A}^T\right) \alpha_j - \frac{1}{2} \sum_{y_i = y_j} \alpha_i Tr\left(\mathbf{A}\mathbf{x}_j \mathbf{x}_i^T \mathbf{A}^T\right) \alpha_j, \quad \text{s.t.} \quad \mathbf{A}\mathbf{A}^T = I_d. \tag{3}$$

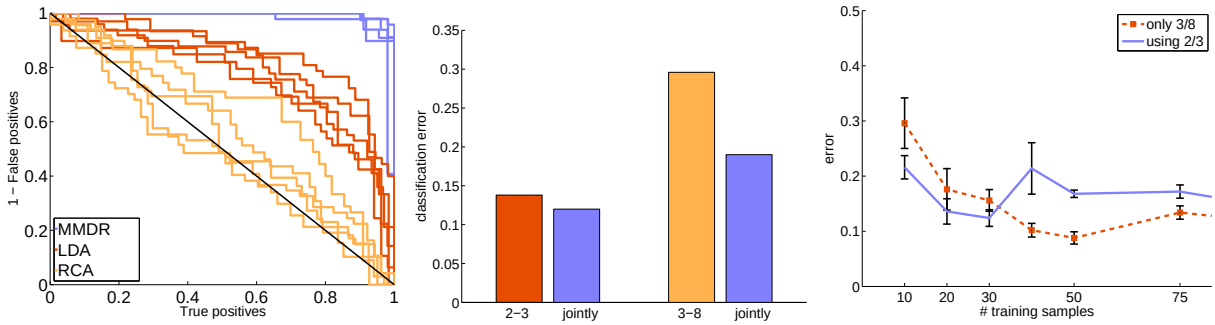

Figure 1: **(a)** Classification of MNIST digits. ROC curves for discriminating the digits 5 *vs.* 6. $d = 10$. Similar results obtained for other values of $d = 1, 2, 5, 20, 50$, and other digit pairs. **(b)** Classification accuracy when classifiers are trained separately and jointly, to discriminate (2,3) and (3,8). Bars denote mean over 5 fold cross validation, $d = 10$, using only 10 training samples from each binary task. **(c)** Classification accuracy for discriminating the digit 3 from 8, as a function of number of training samples. Dashed curve is for model trained on the digits 3 and 8 only. Solid curve used a fixed projection matrix $\mathbf{A}_{23}$ obtained from training on 100 digits of 2 and 3. $d = 10$. Error bars are standard error of the mean over 5 cross validation sets. The auxiliary data facilitiates learning new tasks with very few examples.

For any given α this is equivalent to maximizing $\max_{\mathbf{A}} \quad Tr\left(\mathbf{A}^T\left(C_w - C_b\right)\mathbf{A}\right) \quad \text{s.t.} \quad \mathbf{A}\mathbf{A}^T = I_d$, where $C_w = \sum_{i,j:y_i=y_j} \alpha_i \mathbf{x}_i \mathbf{x}_j^T \alpha_j$ is the within-class, and $C_b = \sum_{i,j:y_i \neq y_j} \alpha_i \mathbf{x}_i \mathbf{x}_j^T \alpha_j$ is the between-class empirical correlation matrix, **taken over the support vectors only**. The solution to this problem consists of the top-d non-negative eigenvectors of $C_w - C_b$. The optimal A can therefore be found efficiently using power methods such as Lanczos [6]. The dual in (2) is concave in α for every value of $\mathbf{A}$ and convex in $\mathbf{A}$ for every value of α, hence has a unique optimum, which can be found by iteratively optimizing over $\mathbf{A}$ (an eigen problem) and over α (a QP). We call this algorithm Linear-MMDR and found that it converged very quickly in practice.

We tested Linear-MMDR in three tasks, all using handwritten digit recognition (MNIST, [10]). First, Fig. 1a compares classification accuracy achieved with MMDR to two other supervised dimensionality reduction approaches: Fisher linear discriminant analysis (LDA), and RCA Relevant component analysis [7]. Second, we tested MMDR in a multi-task setup. Fig. 1b shows that learning two tasks jointly (sharing the projection $\mathbf{A}$), improves accuracy when the number of samples is small. Finaly, we tested MMDR in a transfer learning task, where a linear projection is first learned from an auxiliary task (classify 2,3), and then used to reduce the dimensionality for another task (classify 3,8). Fig. 1c shows that this procedure allows to learn a new (related) task with very few samples.

Learning optimal representation for RBF kernels is hard since nonlinearities typically make the problems inherently non-convex. We study a simlpified problem, of a diagonal projection matrix, closely related to learning the widt hof an RBF kernel. We find sufficient conditions for the problem to be convex, but these only hold for well separated probloems. In practice however, we find that there exists a convex region around the optimum, that can often be identified from the data. More details will be given in the full version.

References

[1] Y. Amit, M. Fink, N. Srebro, and S. Ullman. Uncovering shared structures in multiclass classification. *ICML*, 2007.
[2] R. Ando and T. Zhang. A framework for learning predictive structures from multiple tasks. *JMLR*, 6, 2005.
[3] A. Argyriou, T. Evgeniou, and M. Pontil. Multi-Task Feature Learning. *NIPS 19*, 2007.
[4] J. Baxter. Theoretical models of learning to learn. *Learning to Learn*, 1998.
[5] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge Univ. Press, 2004.
[6] J.W. Demmel. *Applied numerical linear algebra*. Society for Industrial and Applied Mathematics, 1997.
[7] A. Bar-Hillel et al. Learning distance functions using equivalence relations. *ICML*, 2003.
[8] K. Weinberger et al. Distance metric learning for large margin nearest neighbor classification. *NIPS*, 18, 2006.
[9] J. Goldberger, S. Roweis, G. Hinton, and R. Salakhutdinov. Neighbourhood components analysis. *NIPS*, 17, 2005.
[10] Y. LeCun and C. Cortes. *MNIST Database of Handwritten Digits*. NEC Research, 1998.
[11] S. Thrun and L. Pratt. *Learning to Learn*. Kluwer Academic Publishers, 1998.