Learning Discriminative Sufficient Statistics Score Space for Classification

Xiong Li[1,2], Bin Wang[1], Yuncai Liu[1], and Tai Sing Lee[2]

Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China
Computer Science Department, Carnegie Mellon University, Pittsburgh 15213, USA
flit.lee@gmail.com, {binwang,whomliu}@sjtu.edu.cn, tai@cs.cmu.edu

Abstract. Generative score spaces provide a principled method to exploit generative information, e.g., data distribution and hidden variables, in discriminative classifiers. The underlying methodology is to derive measures or score functions from generative models. The derived score functions, spanning the so-called score space, provide features of a fixed dimension for discriminative classification. In this paper, we propose a simple yet effective score space which is essentially the sufficient statistics of the adopted generative models and does not involve the parameters of generative models. We further propose a discriminative learning method for the score space that seeks to utilize label information by constraining the classification margin over the score space. The form of score function allows the formulation of simple learning rules, which are essentially the same learning rules for a generative model with an extra posterior imposed over its hidden variables. Experimental evaluation of this approach over two generative models shows that performance of the score space approach coupled with the proposed discriminative learning method is competitive with state-of-the-art classification methods.

Keywords: generative score space, sufficient statistics, discriminative learning, classification.

1 Introduction

Probabilistic generative models and discriminative models are two complementary [1] and important paradigms in machine learning. Generative models are designed to model data distribution, particularly good at dealing with missing data and structured data, e.g., tree structure data or sequences with variable length. They seek to explain data in terms of hierarchical models with hidden variables. These hidden variables encode higher order information related to observed data that could be informative in the identification of data samples. Further, generative models can be used to construct classifier by means of the maximum a posteriori (MAP) decision rule, resulting in naive Bayes or MAP classifier. However, generative models in general are inferior to discriminative classifiers [2,3] which are designed to directly capture the decision boundaries among different classes. Discriminative classifiers can adapt to complex data

H. Blockeel et al. (Eds.): ECML PKDD 2013, Part III, LNAI 8190, pp. 49–64, 2013.

using furnished or learned kernel similarity. The feature spaces underlying the kernels are generally implicit.

To integrate the capabilities of generative and discriminative models, several schemes [4,5,6,7] have been proposed. Among them, generative score spaces [8,9,10,11,12] provide necessary explicit feature mappings required in many practical applications [13,8] and is the focus of this paper. These explicit feature mappings or score functions are derived from the generative models of the data distribution. Their values, i.e., features, are then delivered to discriminative classifiers to perform classification. While score spaces have shown promising performance in a variety of challenging applications [14,8,15], discriminative learning approaches [16,6,17,18,19] which can exploit label information in general perform better and still furnish state-of-the-art performance.

In this paper, we propose a score space method with an effective score space and a discriminative learning approach. The score space is spanned by the sufficient statistics of an adopted generative model, and is called sufficient statistics score space (SS). Its score function is a function over random variables, which is distinct from earlier methods [8,10,11] in which the scores are functions over random variables and model parameters. We propose a discriminative learning approach to learn the score space by subjecting the classifier over score space to margin constraints. The simple form of the score function results in simple learning rules, which are the same as those for the generative models but with a discriminative posterior imposed over the hidden variables. This posterior in fact introduces a mechanism to generate a more suitable score space for classification.

Further, we will establish the following properties of the score space: (1) the classification error of a zero-loss linear classifier over the score space is at least as low as that of a MAP classifier; (2) the MAP estimation of the linear classifier weights implied in our discriminative learning approach results in an expression of classifier weights that are equal to the weights of the linear SVMs classifiers over the discriminative score space; (3) the discriminative learning approach favors generative models with less hidden variables.

2 Related Works

2.1 Generative Score Spaces

Generative score space [11,12,20,14,8,10] is a class of methods developed to exploit information provided by generative models for discriminative classification. Score functions or feature mappings are functions defined over the observed data, and the hidden variables and parameters of the generative models. The spaces spanned by the score functions are called score spaces or feature spaces.

The score functions generally are measures over generative models. Fisher score (FS) [11] derives score functions by measuring how model parameters affect the log likelihood. Let $\mathbf{x} \in \mathbb{R}^D$ be the observed variable and $P(\mathbf{x} \,|\, \theta)$ be its marginal distribution parameterized by a vector θ, the i-th component of FS is the differential with respect to the parameter θ_i,

$$\Phi_i(\mathbf{x}, \theta) = \nabla_{\theta_i} \log P(\mathbf{x} \,|\, \theta)$$

Table 1. Summary of related discriminative learning approaches

Methods	Feature Mapping	Dis. Learn. Criterion
FKL [19]	"partially" explicit	1-NN
LM-HMM [6]	-	large-margin
disHMM [16]	-	min. hinge-loss
Med-LDA [17]	topic variable	max-margin
disLDA [18]	-	conditional max. likelihood

Free energy score space (FESS) [8] is based on the measures on how well a data point fits random variables. The resulting score functions are the summation terms of the log likelihood function. Posterior divergence (PD) [10] derives a set of comprehensive measures that are connected to both FS and FESS. Another variant class of these methods derives the score function based on class-conditional models, with a model trained for each class, seeking to utilize the label information. The score functions in [20] are log likelihood functions. TOP kernel (TK) [12] extends FS to operate on the MAP discriminant function instead of the log likelihood function. FS was operated on class-conditional models in [14]. These score spaces, working with classifiers, combine and integrate the capabilities from generative and discriminative models, with competitive results in a variety of challenging tasks [14,8,15] such as image recognition. However, these methods learn score spaces and the classifier separately, and might not fully exploit and utilize the label information.

2.2 Discriminative Learning

Several discriminative learning approaches [16,6,17,18,19] have been proposed to exploit the capabilities of generative models and discriminative models simultaneously. Gales et al. [21] comprehensively reviewed the discriminative learning approaches for speech recognition. Table 1 provides a summary of these approaches. Although several discriminative learning criteria are involved, margin based criteria [6,17] exhibit highly competitive performance.

Fisher kernel learning (FKL) [19] is most related to our approach. It proposed a discriminative learning method for Fisher kernel by minimizing the error rate of 1-nearest neighbor (1-NN) classifiers. We observed that, when the learned kernel or score space working with SVMs or its variants, the potential of this method can be further exploited. A potential improvement for this method is to replace the error measure of the 1-NN classifier with the error measure or some criteria of a classifier that will be used to perform classification.

3 Sufficient Statistics Score Space

We here describe how to formulate the sufficient statistics score space, starting from the variational lower bound of generative models. The idea is to decompose the log likelihood into parameter-based parts and variable-based parts.

3.1 Variational Inference of Exponential Family

We consider a general case where $P(\mathbf{x};\theta)$ is a hierarchical generative model. Let $P(\mathbf{x},\mathbf{h};\theta)$ be its joint distribution with a set of hidden variables $\mathbf{h}$ and the parameter vector θ. In this case, it is usually difficult to obtain the close form of $P(\mathbf{x};\theta)$ since the integration is usually intractable. A practical method is to resort to the lower bound of $\log P(\mathbf{x};\theta)$. We here use the lower bound given by variational inference [22], for sample $\mathbf{x}^t$,

$$\log P(\mathbf{x}^t;\theta) \geq \mathrm{KL}(Q(\mathbf{h}^t) \,\|\, P(\mathbf{x}^t,\mathbf{h}^t;\theta)) = \mathcal{F}^t(Q,\theta), \tag{1}$$

where $\mathbf{h}^t$ indicates that it depends on $\mathbf{x}^t$ [8]; $Q(\mathbf{h}^t)$ is the approximate distribution of the real posterior $P(\mathbf{h}^t \,|\, \mathbf{x}^t,\theta)$; $\mathcal{F}^t(Q,\theta)$ is the negative free energy function or the lower bound of $\log P(\mathbf{x}^t;\theta)$. It is worth noting that, the approximation of the real posterior $P(\mathbf{h}^t \,|\, \mathbf{x}^t,\theta)$ using $Q(\mathbf{h}^t)$ and of the the real log likelihood $\log P(\mathbf{x}^t;\theta)$ using the lower bound $\mathcal{F}^t(Q,\theta)$ is often satisfied. In fact, the approximation error can be zero since $Q(\mathbf{h}^t)$ exactly equals to $P(\mathbf{h}^t \,|\, \mathbf{x}^t,\theta)$ and $\mathcal{F}^t(Q,\theta)$ exactly equals to $\log P(\mathbf{x}^t;\theta)$ when using exact inference. Learning generative models based on the variational lower bound can be expressed as,

$$\max_{Q,\theta} \sum\nolimits_t \mathcal{F}^t(Q,\theta) = \max_{Q,\theta} \sum\nolimits_t -\mathrm{KL}(Q(\mathbf{h}^t) \,\|\, P(\mathbf{x}^t,\mathbf{h}^t;\theta)) \tag{2}$$

An assumption here, as also made in most probabilistic generative models [23], is that the joint distribution $P(\mathbf{x},\mathbf{h};\theta)$ of a generative model belongs to the exponential family, written as [23],

$$P(\mathbf{x},\mathbf{h};\theta) = \exp\{\alpha(\theta)^T T(\mathbf{x},\mathbf{h}) + A(\theta)\} \tag{3}$$

where $\alpha(\theta)$ is a vector-valued function; $T(\mathbf{x},\mathbf{h})$ is the vector of sufficient statistics over $\mathbf{x}$ and $\mathbf{h}$; $A(\theta)$ is a scalar function. Since $P(\mathbf{x},\mathbf{h}) = P(\mathbf{x} \,|\, \mathbf{h})P(\mathbf{h})$, $P(\mathbf{h})$ also belongs to exponential family, $P(\mathbf{h};\theta_h) = \exp\{\alpha(\theta_h)^T T(\mathbf{h}) + A(\theta_h)\}$. As was done in [24], we assume that, for a sample $\mathbf{x}^t$, the approximate posterior $Q(\mathbf{h}^t)$ shares the same form as $P(\mathbf{h};\theta_h)$, but with different parameters,

$$Q(\mathbf{h}^t) = \exp\{\alpha(\theta_h^t)^T T(\mathbf{h}^t) + A(\theta_h^t)\} \tag{4}$$

where θ_h^t is a vector of parameters and depends on the sample $\mathbf{x}^t$. Substituting Eqs. (3) and (4) into Eq. (1), it can be verified that,

$$\begin{aligned}
\mathcal{F}^t(Q,\theta) &= \mathrm{E}_{Q(\mathbf{h}^t)}[\alpha(\theta)^T T(\mathbf{x}^t,\mathbf{h}^t) + A(\theta) - \alpha(\theta_h^t)^T T(\mathbf{h}^t) - A(\theta_h^t)] \\
&= \mathrm{E}_{Q(\mathbf{h}^t)}[\alpha(\theta)^T T(\mathbf{x}^t,\mathbf{h}^t) - \mathbf{1}^T \mathrm{diag}(\alpha(\theta_h^t)) T(\mathbf{h}^t) - A(\theta_h^t) + A(\theta)] \\
&= \alpha(\theta)^T \mathrm{E}_{Q(\mathbf{h}^t)}[T(\mathbf{x}^t,\mathbf{h}^t)] - \mathbf{1}^T \mathrm{diag}(\alpha(\theta_h^t)) \mathrm{E}_{Q(\mathbf{h}^t)}[T(\mathbf{h}^t)] - A(\theta_h^t) + A(\theta)] \\
&= \eta^T \mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t,\mathbf{h}^t)] = \eta^T \Phi(\mathbf{x}^t)
\end{aligned} \tag{5}$$

where $\eta = (\alpha(\theta)^T, -\mathbf{1}^T, -1, A(\theta))^T$ only depends on parameter θ; $\phi^t(\mathbf{x}^t,\mathbf{h}^t)$ is a function over $\mathbf{x}^t$, $\mathbf{h}^t$ and θ_h^t, depending on $\mathbf{x}^t$,

$$\phi(\mathbf{x}^t,\mathbf{h}^t) = (T(\mathbf{x}^t,\mathbf{h}^t)^T, (\mathrm{diag}(\alpha(\theta_h^t)) T(\mathbf{h}^t))^T, A(\theta_h^t), 1)^T \tag{6}$$

Note that $\mathbf{h}^t$ and θ_h^t depend on the specific sample $\mathbf{x}^t$. Therefore they reflect some attributes or encode some information related to $\mathbf{x}^t$. $\Phi(\mathbf{x}^t)$ is the score function or feature mapping, taking the following form,

$$\Phi(\mathbf{x}^t) = \mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t, \mathbf{h}^t)] \tag{7}$$

The function $\Phi(\mathbf{x}^t)$ is termed as sufficient statistics score function since its main components are sufficient statistics $T(\mathbf{x}, \mathbf{h})$ and $T(\mathbf{h})$. $\mathcal{F}^t(Q, \theta)$ is decomposed into the linear combination of η which depends on all training samples and the score function $\Phi(\mathbf{x}^t)$ which depends on the sample $\mathbf{x}^t$.

The above formulation is based on the variational inference in Eq. (1) and the approximate posterior in Eq. (4). The approximation works well when the real log likelihood are intractable [8,10], and equals to the real log likelihood exactly when using exact inference. The derived score function in Eq. (7) is compatible with other inference methods as we can estimate the posterior (Eq. (4)) using the outputs of those methods, e.g., using the samples drawn by Gibbs sampling.

3.2 Error Rate Comparison with MAP Classification

As score spaces typically work with linear classifiers, [12] proposed a method to analyze the classification error of a linear classifier $y = \mathrm{sign}(\mathbf{w}^T\Phi(\mathbf{x}) + b)$ where $\mathbf{w} \in \mathbb{R}^d$ is the weight and $b \in \mathbb{R}$ is the bias. We assume that $\mathbf{w}$ and b are learned by an optimal learning algorithm on a sufficiently large training set. Letting $\Psi(a)$ be the zero-one loss function that outputs 1 if $a > 0$ and 0 otherwise, the classification error can be expressed as,

$$R(\Phi) = \min_{\mathbf{w},b} \mathrm{E}_{\mathbf{x},y}\Psi[-y(\mathbf{w}^T\Phi(\mathbf{x}) + b)],$$

where $\mathrm{E}_{\mathbf{x},y}$ denotes the expectation over the true distribution. Note that $R(\Phi)$ is exactly the test error if the test set and the training set share the same distribution. We assume this condition holds, as was done in [12,8,10].

Previous works [12,8,10] have shown that, in the case that the model is trained using samples from the positive class and the log likelihood $\log P(\mathbf{x} \mid y = +1)$ is available, the error rate $R(\Phi)$ of a linear classifier operating on the score space is at least as low as the error rate $R(\lambda)$ of the MAP classifier,

$$\begin{aligned} R(\lambda) &= \mathrm{E}_{\mathbf{x},y}\Psi[-y(P(y = +1 \mid \mathbf{x}) - \frac{1}{2})] = \mathrm{E}_{\mathbf{x},y}\Psi[-y(\log P(y = +1 \mid \mathbf{x}) - \log\frac{1}{2})] \\ &= \mathrm{E}_{\mathbf{x},y}\Psi[-y(\log P(\mathbf{x} \mid y = +1) - \log\frac{1}{2})] = \mathrm{E}_{\mathbf{x},y}\Psi[-y(\eta^T\Phi(\mathbf{x}) - \log\frac{1}{2})] \\ &\geq \min_{\mathbf{w},b} \mathrm{E}_{\mathbf{x},y}\Psi[-y(\mathbf{w}^T\Phi(\mathbf{x}) + b)] = R(\Phi) \end{aligned}$$

In the case that the exact log likelihood might be intractable, as shown in Eq. (5), we resort to the lower bound $\mathcal{F}_{+1}(\mathbf{x})$ and $\mathcal{F}_{-1}(\mathbf{x})$ for a pair of models θ_{+1} and θ_{-1} that are respectively trained using the positive samples and negative samples, and accordingly resort to the free energy test [8]. That is,

$\hat{y} = \text{sign}(\mathcal{F}_{+1}(\mathbf{x}) - \mathcal{F}_{-1}(\mathbf{x}))$. Applying the formulation in Eq. (5), then we have $\mathcal{F}_{+1}(\mathbf{x}) = \eta_{+1}^T \Phi_{+1}(\mathbf{x})$ and $\mathcal{F}_{-1}(\mathbf{x}) = \eta_{-1}^T \Phi_{-1}(\mathbf{x})$. We accordingly define the score function over a pair of models as $\Phi(\mathbf{x}) = (\Phi_{+1}(\mathbf{x})^T, \Phi_{-1}(\mathbf{x})^T)^T$. The above inequality $R(\Phi) \leq R(\lambda)$ still holds,

$$\begin{aligned} R(\lambda) &= \mathrm{E}_{\mathbf{x},y} \Psi[-y(\mathcal{F}_{+1}(\mathbf{x}) - \mathcal{F}_{-1}(\mathbf{x}))] \\ &= \mathrm{E}_{\mathbf{x},y} \Psi[-y(\eta_{+1}^T \Phi_{+1}(\mathbf{x}) - \eta_{-1}^T \Phi_{-1}(\mathbf{x}))] \\ &\geq \min_{\mathbf{x},y} \mathrm{E}_{\mathbf{x},y} \Psi[-y(\mathbf{w}^T \Phi(\mathbf{x}) + b)] = R(\Phi) \end{aligned}$$

The above justifications also hold for [11,12,8,10] because $\mathcal{F}(\mathbf{x}, \theta)$ can be expressed as a linear combination of any of the score functions.

4 Learning Discriminative Score Space

To exploit label information, we propose a discriminative learning method that learns score space as well as generative models under the classification margin constraints of a linear classifier in the score space.

4.1 The Learning Problem

First we will use a probabilistic classifier because of its compatibility with probabilistic generative models. Let $\mathbf{x}$ be the input data and $y \in \{-1, +1\}$ be the output label; $\mathcal{S} = \{(\mathbf{x}^t, y^t)\}_t$ be the training set whose samples are indexed by t. Let $\mathbf{x}$ be the augmented sample $(\mathbf{x}^T, 1)^T$; $\mathbf{w}$ be the weight including the bias; γ^t be the desired margin for the sample $\mathbf{x}^t$. The classifier subject to margin constraint is given by [4],

$$\min_{Q(\mathbf{w})Q(\gamma^t)} \mathrm{KL}(Q(\mathbf{w})Q(\gamma^t) \,\|\, P(\mathbf{w})P(\gamma^t)) \tag{8}$$

$$\text{s.t. } \mathrm{E}_{Q(\mathbf{w})}[y^t \mathbf{w}^T \mathbf{x}^t] \geq \mathrm{E}_{Q(\gamma^t)}[\gamma^t], \ \forall\, t, \tag{9}$$

where $P(\mathbf{w})$ and $Q(\mathbf{w})$ are the prior and posterior for the weight respectively; $P(\gamma^t)$ and $Q(\gamma^t)$ are the prior and posterior for the margin respectively. The margin γ^t is specified for $\mathbf{x}^t$. This formulation allows for a tunable and flexible margin, which functions in a way similar to the soft margin in SVMs.

Now we have shown the objective functions of generative models (Eq. (1)) and the classifier (Eqs (8) and (9)). Learning discriminative score space subject to margin constrains means we need to maximize Eq. (1) and minimize Eq. (8) simultaneously, subject to Eq. (9). The learning problem can be expressed as,

$$\min_{\mathcal{Q},\theta} \sum\nolimits_t \underbrace{\mathrm{KL}(Q(\mathbf{h}^t) \,\|\, P(\mathbf{x}^t, \mathbf{h}^t; \theta))}_{\mathrm{KL}_\theta \text{ (generative)}} + \xi \underbrace{\mathrm{KL}(Q(\mathbf{w})Q(\gamma^t) \,\|\, P(\mathbf{w})P(\gamma^t))}_{\mathrm{KL}_\mathbf{w} + \mathrm{KL}_\gamma \text{ (discriminative)}} \tag{10}$$

$$\text{s.t. } \mathrm{E}_\mathcal{Q}\big[y^t \mathbf{w}^T \phi(\mathbf{x}^t, \mathbf{h}^t) - \gamma^t\big] \geq 0, \ \forall\, t \tag{11}$$

where $\mathcal{Q} = \{Q(\mathbf{h}^t), Q(\gamma^t), Q(\mathbf{w})\}$. The first term in Eq. (10) is the objective function for the generative model as in Eq. (2), where $P(\mathbf{x}, \mathbf{h}; \theta)$ is the joint

distribution and $Q(\mathbf{h}^t)$ is the approximate posterior. The second term of Eq. (10) and the constraint Eq. (11) form the objective function of the classifier, where $P(\gamma^t)$ and $P(\mathbf{w})$ are priors on the margins and the weights respectively. $\xi > 0$ is a weight that tunes the balance between the generative model and the classifier.

4.2 Inference and Parameter Estimation

The quantities to be estimated in the objective function Eq. (10) and Eq. (11) include $Q(\mathbf{h}^t)$, $Q(\gamma^t)$, $Q(\mathbf{w})$ and θ. To estimate these quantities, we first specify the priors $P(\mathbf{w})$ and $P(\gamma^t)$. Similar to that in [4], we set the priors,

$$P(\mathbf{w}) = \mathcal{N}(0, \mathrm{I}), \tag{12}$$

$$P(\gamma^t) = ce^{-c(a-\gamma^t)} \text{ for } \gamma^t \leq a. \tag{13}$$

where a, c are two parameters to be specified. The learning problem in Eq. (10) and Eq. (11) takes the exact form of posterior regularization [25], and in principle can be solved using EM-like procedures [25,26]. In optimization [26], to estimate $Q(\mathbf{h}^t)$, $Q(\gamma^t)$, $Q(\mathbf{w})$ and θ, we alternatively solve sub-problems with respect to some of these quantities while keeping the others fixed in each pass. The solution of θ will benefit from the form of $\Phi(\mathbf{x})$ because $\Phi(\mathbf{x})$ and the constraints Eq. (11) are not related to θ.

Posterior $Q(\mathbf{h}^t)$ of Hidden Variables. By fixing quantities $Q(\gamma^t)$ and θ, the solution of $Q(\mathbf{h}^t, \mathbf{w})$ takes the following form [25,4],

$$Q(\mathbf{h}^t, \mathbf{w}) \propto P(\mathbf{x}^t, \mathbf{h}^t; \theta) P(\mathbf{w}) \cdot \exp\left\{\sum\nolimits_t \lambda^t \left[y^t \mathbf{w}^T \phi^t - \mathrm{E}_{Q(\gamma^t)}[\gamma^t]\right]\right\}, \tag{14}$$

where $\phi^t = \phi(\mathbf{x}^t, \mathbf{h}^t)$, and λ^t is the Lagrange multiplier for the t-th inequality of Eq. (11). Note that $\mathbf{w}$ follows a normal prior in Eq. (12), making the integration $Q(\mathbf{h}^t) = \int Q(\mathbf{h}^t, \mathbf{w}) d\mathbf{w}$ tractable,

$$Q(\mathbf{h}^t) \propto \underbrace{P(\mathbf{x}^t, \mathbf{h}^t; \theta)}_{\propto P(\mathbf{h}^t | \mathbf{x}^t, \theta)} \underbrace{\exp\left\{\sum\nolimits_t \lambda^t \mathrm{E}_{Q(\gamma^t)}[\gamma^t] - \frac{1}{2} \sum\nolimits_{t,t'} \lambda^t \lambda^{t'} y^t y^{t'} (\phi^t)^T \phi^{t'}\right\}}_{\propto \text{ discriminative posterior}} \tag{15}$$

This formula shows that the posterior of hidden variables is proportional to the product of (1) the joint distributions of the naive generative model and (2) an exponential term that is derived from the classifier and favors large margin. When $\mathbf{h}$ is a set of discrete variables, the posterior and $\mathrm{E}_{Q(\mathbf{h}^t)}[\phi^t]$ are straightforward to compute; when $\mathbf{h}$ is a set of continuous variables, without an analytical solution in most cases, we resort to estimate the expectation $\mathrm{E}_{Q(\mathbf{h}^t)}[\phi^t]$ by,

$$\mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t, \mathbf{h}^t)] \approx \frac{1}{n} \sum_{i=1}^{n} \phi(\mathbf{x}^t, \mathbf{h}_i^t), \tag{16}$$

where $\mathbf{h}_i^t$ is the i-th sample of all the n samples drawn from the posterior $Q(\mathbf{h}^t)$. Gibbs-rejection sampling [27] can be very effective in drawing samples from Eq. (15). A sample $\mathbf{h}_i^t$ drawn from $P(\mathbf{x}^t, \mathbf{h}^t; \theta)$ will be accepted or rejected based on the exponential term.

Posterior $Q(\gamma^t)$ of Margins. By fixing the quantities $Q(\mathbf{h}^t)$ and θ, the posterior $Q(\gamma^t, \mathbf{w})$ can be solved, in the same way as in the solution of $Q(\mathbf{h}^t, \mathbf{w})$. We compute the posterior $Q(\gamma^t) = \int Q(\gamma^t, \mathbf{w}) d\mathbf{w}$ as,

$$\begin{aligned} Q(\gamma^t) &\propto \int P(\gamma^t) \exp\left\{\lambda^t \mathrm{E}_{Q(\mathbf{h}^t)}\left[y^t \mathbf{w}^T \phi^t - \gamma^t\right]\right\} d\mathbf{w} \\ &\propto \exp\left\{-\left(c - \lambda^t\right)\left(a - \gamma^t\right)\right\}, \end{aligned} \tag{17}$$

For the exponential distribution $P(\gamma^t) = ce^{-c\gamma^t}$ with $\gamma^t \geq 0$ (Eq. (13)), the mean of γ^t is $\mathrm{E}_{P(\gamma^t)}[\gamma^t] = c^{-1}$. The expected margin can be similarly derived,

$$\mathrm{E}_{Q(\gamma^t)}[\gamma^t] = a - \left(c - \lambda^t\right)^{-1}. \tag{18}$$

which adapts to samples, for example, by taking negative values for incorrect classification, which essentially implements a soft-margin.

Lagrange Multipliers $\boldsymbol{\lambda} = \{\lambda^1, \lambda^2, ..., \lambda^N\}$. Every Lagrange multiplier here corresponds to an inequality constraint. Fixing $Q(\mathbf{h}^t)$, $Q(\gamma^t)$ and θ leads to,

$$Q(\mathbf{w}) = \frac{1}{Z(\lambda)} P(\mathbf{w}) \exp\left\{\sum\nolimits_t \lambda^t \mathrm{E}_{Q(\mathbf{h}^t, \gamma^t)}\left[y^t \mathbf{w}^T \phi^t - \gamma^t\right]\right\},$$

where $Z(\lambda) = \int Q(\mathbf{w}) d\mathbf{w}$ is the partition function. Then $\lambda \geq 0$ is obtained by maximizing the objective function $J_\lambda = -\log Z(\lambda)$. Using the same integration in Eq. (15), we have,

$$J_\lambda = \sum_t \lambda^t \mathrm{E}_{Q(\gamma^t)}[\gamma^t] - \frac{1}{2} \sum_{t,t'} \lambda^t \lambda^{t'} y^t y^{t'} \mathrm{E}_{Q(\mathbf{h}^t)}[\phi^t]^T \mathrm{E}_{Q(\mathbf{h}^{t'})}[\phi^{t'}]. \tag{19}$$

This is a standard quadratic programming problem, which can be efficiently solved. It differs from the dual form of SVMs because of the extra weight $\mathrm{E}_Q[\gamma^t]$.

Parameters θ of Generative Models. In the objective function Eq. (10) and the constraint Eq. (11), only the term KL_θ depend on the parameter θ. So minimizing the objective function with respect to θ equals to minimizing KL_θ with respect to θ, not subjecting to any inequality constraint. The resulting update rules for θ are the *same* as those for the original generative models.

The learning procedure of the proposed method is summarized in Algorithm 1. The output is the parameter θ of a generative model. Given the generative model trained by Algorithm 1, we are now equipped to compute score functions for test samples. The procedure constructing discriminative score space is summarized in Algorithm 2.

4.3 Classifier Learning Rules

Given the discriminatively learned generative models and score spaces, there are two ways to obtain classifiers over the score spaces: (1) train classifiers on the

Algorithm 1. Discriminative learning of generative models

input: training data set $\mathcal{S} = \{(\mathbf{x}^t, y^t)\}_{t=1}^N$
initialize parameters $\hat{\theta}, \mathbf{u}, \lambda$
repeat
  for $t = 1$ to N **do**
    sample $\{\mathbf{h}_i^t\}_i$ from $Q(\mathbf{h}^t)$ (Eq. (15))
    estimate $\mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t, \mathbf{h}^t)]$ (Eq. (16))
    compute $\mathrm{E}_{Q(\gamma^t)}[\gamma^t]$ (Eq. (18))
  end for
  update λ (Eq. (19))
  update $\hat{\theta}$ with $\{\mathbf{h}_i^t\}_{ti}$ using the rules of the original generative models
until convergence
output: $\hat{\theta}$

Algorithm 2. Construct discriminative sufficient statistics score spaces

input: generative model $\hat{\theta}$ and input data set $\{(\mathbf{x}^t, y^t)\}_{t=1}^{N_t}$
for $t = 1$ to N_t **do**
  sample $\{\mathbf{h}_i^t\}_i$ from $Q(\mathbf{h}^t)$ (Eq. (15))
  estimate $\Phi(\mathbf{x}^t) = \mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t, \mathbf{h}^t)]$ (Eq. (16))
end for
output: $\{\Phi(\mathbf{x}^t)\}_{t=1}^{N_t}$

score spaces using any standard method; (2) estimate SVMs like classifiers using the quantities produced by Algorithm 1. We will now present the details of (2).

The learning problem in Eq. (10) and Eq. (11) already includes a linear classifier with the following decision rule,

$$\hat{y} = \mathrm{sign}(\mathrm{E}_{Q(\mathbf{w})}[\mathbf{w}^T \Phi(\mathbf{x})])$$

To estimate a classifier based on the quantities produced by Algorithm 1, we just need to estimate the weight $\mathbf{w}$. First, we specify the posterior of the classifier $\mathbf{w}$ to be a Gaussian distribution with unit covariance matrix [28],

$$Q_s(\mathbf{w}) = N(\mathbf{u}, \mathrm{I}). \tag{20}$$

where $\mathbf{u}$ is the mean to be estimated from training data. Considering the above specification for $Q_s(\mathbf{w})$ and the specification for $P(\mathbf{w})$ (Eq. (12)), it can be verified that $\mathrm{KL}_{\mathbf{w}} = \mathrm{KL}(Q(\mathbf{w}) \,\|\, P(\mathbf{w})) = \frac{1}{2}\mathbf{u}^T\mathbf{u}$. This means that minimizing $\mathrm{KL}_{\mathbf{w}}$ in Eq. (10) encourages $\mathbf{u}$ to have a short length. Under the above specifications, the solution of $\mathbf{w}$ takes the following form.

Proposition 1. *Let $\Phi(\mathbf{x}) = \mathrm{E}_{Q(\mathbf{h})}[\phi(\mathbf{x}, \mathbf{h})]$ be the score function derived from (Algorithm 2) the discriminatively trained generative models (Algorithm 1). With the specification in Eq. (20), the maximum a posteriori (MAP) estimation of $\mathbf{w}$ in Eq. (10) takes the same form as the solution of the linear SVMs equipped with the score function $\Phi(\mathbf{x})$.*

Proof. The solution of $Q(\mathbf{w})$ can be expressed as,

$$\begin{aligned} Q(\mathbf{w}) &= \frac{1}{Z} P(\mathbf{w}) \exp\left\{ \sum\nolimits_t \lambda^t \mathrm{E}_{Q(\mathbf{h}^t, \gamma^t)} \left[y^t \mathbf{w}^T \phi^t - \gamma^t \right] \right\} \\ &= \frac{1}{Z} P(\mathbf{w}) \exp(\alpha^T \mathbf{w} - \beta), \end{aligned} \tag{21}$$

where $Z = \int P(\mathbf{w}) \exp(\alpha^T \mathbf{w} + \beta) d\mathbf{w}$ is the partition function to ensure $Q(\mathbf{w})$ being a probabilistic distribution; $\alpha = \sum_t \lambda^t y^t \mathrm{E}_{Q(\mathbf{h}^t)}[\phi^t]$ and $\beta = \mathrm{E}_{Q(\gamma^t)}[\gamma^t]$. Considering the specification $Q_s(\mathbf{w})$ in Eq. (20), the MAP estimation $\hat{\mathbf{w}}$ of $\mathbf{w}$ satisfies $\hat{\mathbf{w}} = \mathrm{E}_{Q(\mathbf{w})}[\mathbf{w}] = \mathbf{u}$ and can be determined by minimizing the I-projection between the specified posterior Eq. (20) and the derived posterior Eq. (21),

$$\begin{aligned} &\min_{\mathbf{u}} \mathrm{KL}\Big[Q_s(\mathbf{w}) \,\|\, \frac{1}{Z} P(\mathbf{w}) \exp(\alpha^T \mathbf{w} + \beta) \Big] \\ &= \min_{\mathbf{u}} \mathrm{E}_{Q_s(\mathbf{w})}\Big[\log Q_s(\mathbf{w}) - \log \frac{1}{Z} P(\mathbf{w}) \exp(\alpha^T \mathbf{w} + \beta) \Big] \\ &= \min_{\mathbf{u}} \mathrm{E}_{Q_s(\mathbf{w})}\Big[\mathbf{w}^T \mathbf{u} - \frac{1}{2} \mathbf{u}^T \mathbf{u} - (\alpha^T \mathbf{w} + \beta) \Big] + \log Z \\ &= \min_{\mathbf{u}} \Big[\frac{1}{2} \mathbf{u}^T \mathbf{u} - \alpha^T \mathbf{u} - \beta \Big] + \log Z. \end{aligned}$$

where Z does not depend on $\mathbf{u}$. Letting $\frac{\partial \mathrm{KL}}{\partial \mathbf{u}} = 0$, we has an analytical solution,

$$\hat{\mathbf{w}} = \mathbf{u} = \alpha = \sum_t \lambda^t y^t \mathrm{E}_{Q(\mathbf{h}^t)}[\phi(\mathbf{x}^t, \mathbf{h}^t)]. \tag{22}$$

This is equivalent to the solution of linear SVMs [2].

5 Experiments

We experimented with two generative models in the proposed framework in the context of classification. As shown in Section 4.2 and Algorithm 1, we only need to specify the feature mapping Φ for each adopted generative model. In each experiment, we compare (1) the proposed sufficient statistics (SS) score space which learns the score spaces (including generative models) and the discriminative classifiers separately, under no discriminative constraint; (2) the discriminative learning of SS subject to margin constraints (MSS), as proposed in Section 4; (3) Fisher score (FS) method [11]; (4) free energy score space (FESS) method [8] and other state-of-the-art methods. Here, we omit the comparison with other hybrid methods [5,10] due to the space limitation. For each problem, we repeatedly test 20 rounds. In each round, training and test sets are formed by random sampling from the dataset.

The MSS approach is proposed in the setting of binary classification. It is straightforward to extend it to multi-class classification problems, by splitting each multi-class problem into several binary problems and combine the MSS features separately learned from each of the binary classification problems. The

Table 2. Summary of classification accuracy (%) on sequence datasets. Discrete HMMs are used to model the distribution of sequences. SS is the baseline version of the proposed method without using discriminative learning.

Class	C	LM-HMM	FKL [19]	FS [11]	FESS [8]	SS	MSS
Character	20	94.26	**95.71**	95.20	93.99	93.55	95.62
Hill Valley	2	58.71	54.00	63.60	55.41	53.39	**65.68**
Jap. Vowel	9	92.26	**96.16**	88.93	90.63	91.26	93.40
Hand Move.	15	78.10	75.22	79.11	79.89	78.00	**82.22**
Promoter Gene	2	67.92	69.81	63.38	65.77	65.35	**74.23**
Junction Gene	3	58.64	57.05	58.71	58.78	58.86	**65.37**
Protein Kinase	3	72.18	74.15	73.24	72.65	73.53	**78.53**
SCOP Protein	7	**64.75**	60.96	64.17	64.12	64.24	64.64
Chicken Shape	5	77.64	79.83	79.63	80.26	79.58	**83.36**

parameters ($\xi = 1$ (Eq. (10)), $a = 1$ and $c = 6$ (Eq. (13)), the number of topics M of LDA (Section 5.2), the number of hidden states K of HMMs (Section 5.1)) used in the following experiments are chosen through an *offline* cross validation method, i.e., the parameters are chosen using cross validation on a dataset and then applied to all datasets. The reasons of using offline rather that online method are that (1) online cross validation for 5 parameters are computationally very expensive; (2) offline method produces satisfied performance.

For score spaces FS, FESS and SS, we use the same scheme as [8], i.e., train a generative model for each class and combine the features obtained from these models. This scheme is empirically validated to be more effective than the score space derived from one generative model of all samples. For all score space methods (FS, FESS, SS, MSS), we use linear SVMs (libsvm toolbox [29]) as the classifier. For localized multiple kernel learning (LMKL) [3], Fisher kernel learning (FKL) [19] and FESS, we use the authors' implementations, which can be downloaded from their websites. FS-HMMs, FS-LDA, LM-HMMs [6] and the proposed methods are implemented by ourselves.

5.1 Sequence Recognition: Hidden Markov Models

In the first experiment, we learn the score space for sequence recognition with hidden Markov models (HMMs) [30] as the generative model. Let $\mathbf{x}$ be the sequence with length $L_{\mathbf{x}}$. We here consider the discrete case where $\mathbf{x}^l$ is a vector of binary indicators of states at position l along the sequence, i.e., $x_k^l = 1$ if the k-th of the K possible observed states is selected at position l. $\mathbf{q}^l$ is the binary indicator for hidden states, where $q_i^l = 1$ if the i-th of the M possible hidden states is selected at position l. The joint distribution is given by,

$$P(\mathbf{x}, \mathbf{q}; \theta) = \prod_{i=1}^{M} \pi_i^{q_i^0} \cdot \prod_{l=0}^{L_{\mathbf{x}}-1} \prod_{i,j=1}^{M,M} a_{ij}^{q_i^l q_j^{l+1}} \cdot \prod_{l=0}^{L_{\mathbf{x}}} \prod_{i,k=1}^{M,K} b_{ik}^{q_i^l x_k^l}$$

where $\theta = \{\pi_i, a_{ij}, b_{ik}\}_{ijk}$. Let $\hat{\pi} = \{\hat{\pi}_i\}_i$, $\hat{\mathbf{a}} = \{\hat{a}_{ij}\}_{ij}$ and $\hat{\mathbf{b}} = \{\hat{b}_{ik}\}_{ik}$ respectively be the initial, state transition and emission probabilities of the approximate posterior. The score function is $\Phi(\mathbf{x}) = \mathrm{E}_{Q(\mathbf{q})}[\phi(\mathbf{x}, \mathbf{q})]$, where,

$$\phi(\mathbf{x}, \mathbf{q}) = \mathrm{vec}\left(\left\{ q_i^0, \sum_{l=0}^{L_{\mathbf{x}}-1} q_i^l q_j^{l+1}, \sum_{l=0}^{L_{\mathbf{x}}} q_i^l x_k^l, \right.\right.$$
$$\left.\left. q_i^0 \log \hat{\pi}_i, \sum_{l=0}^{L_{\mathbf{x}}-1} q_i^l q_j^{l+1} \log \hat{a}_{ij}, \sum_{l=0}^{T_{\mathbf{x}}} q_i^l x_k^l \log \hat{b}_{ik} \right\}_{i,k}\right).$$

Given the hidden states of the input sequence inferred with the Baum-Welch algorithm [31], it is easy to estimate the posterior probabilities, i.e. initial, transition, and emission probabilities conditioned on $\mathbf{x}$. Using the sampling distribution in Eq. (15), we are able to draw examples of hidden states and re-estimate their posterior. The quantity $\mathrm{E}_{Q(\mathbf{z})}[\cdot]$ can be computed effectively since $\mathbf{z}$ is a discrete variable.

We compare the performance of SS and MSS against that of FS, FESS, FKL [19] and large margin HMM (LM-HMM) [6]. The number of hidden states M is set to be $M = 3$ for MSS and $M = 10$ for FS, FESS and SS based on cross-validation performance as shown in Fig. (1). For FKL and LM-HMM, we chose M from $2, 5, 10$ using offline cross validation. We randomly select 50% samples for training and the rest for testing. The learned score space is evaluated on 9 sequence datasets where SCOP protein is obtained from ASTRAL database with similar sequences reduced by a E-value threshold of 10^{-25}; the chicken piece shape dataset is collected by [32]; the rest are obtained from UCI database. For FS, FESS, SS and MSS, the datasets with continuous values are quantized to state sequences for the discrete HMMs, i.e., 8 states for chicken piece shape and 20-40 states for other datasets. For FKL, we use continuous HMMs for continuous data and discrete HMMs which is implemented by configuring the graph of MRF for state sequence data.

The results are reported in Table 2. Our SS's performance is competitive against FS and FESS, even though it does not utilize the parameters of generative models as was done in FS and FESS. Our MSS outperforms other methods in 8 of the 13 experiments. The improvement of MSS over SS is brought about by the discriminative learning paradigm. The comparison between MSS against LM-HMM or FKL is particularly worth noting because both LM-HMM and FKL are methods that learn generative models and discriminative models jointly. We should also note that the data representation used in FKL is slightly different with that used in MSS. That is, FKL uses continuous data on the first 4 datasets while MSS quantities them into discrete data, and thus its performance might suffer from this quantization. Further, MSS is effective for a small number of hidden states and thus is more efficient to train, even with limited samples. We will discuss these issues more in Section 5.3.

5.2 Image Recognition: Latent Dirichlet Allocation

We also evaluate the framework when Latent Dirichlet Allocation (LDA) [34] is used as its generative model in the context of image scene recognition. In this

Table 3. Classification accuracy (%) on OT, Scene-15 and UIUC-sports datasets

Dataset	C	PHOW [33]	Med-LDA[17]	FS [11]	FESS [8]	SS	MSS
OT	8	87.21	89.50	86.42	88.89	88.25	**90.36**
Scene-15	15	79.83	81.05	78.68	81.92	79.25	**83.64**
UIUC-sports	8	80.04	82.79	79.91	82.18	80.34	**84.67**

task, visual words are used to represent images as is typically done in computer vision. The LDA version in [35] is used to model the distribution of visual words, with each topic associated with a particular distribution. We sample topic variables using collapsed Gibbs sampling [35], and reject examples according to the rule stipulated in Eq. (15). Differing from [35], we update model parameters α and β in each iteration. In order to make it compatible with [35], we only use word and topic to construct feature mapping $\Phi(w_{\cdot}^{d})$,

$$\phi = \mathrm{vec}\left(\left\{z_{dn}^{k}, w_{n}^{d} z_{dn}^{k}, 1\right\}_{n,k}\right),$$

where w and z denote word and topic respectively. d, n and k index image, word and topic respectively. For FS [11] and FESS [8], we extract features from the trained LDA model and use them with linear SVM.

The OT scene dataset, Scene-15 dataset and UIUC-sports dataset are used for evaluation. They contains 8, 15 and 8 categories respectively. For each image, dense SIFT descriptors [36] are extracted from 20×20 grid patches over 4 scales. The descriptors are clustered (using K-mean on randomly selected descriptors) into 200 visual words in a code book. An image is represented by a histogram of the frequency of the observed visual words. The number of topics is set to $K = 10$ for MSS and $K = 50$ for FS, FESS and SS throughout the experiment. For OT, Scene-15 and UIUC-sports datasets, 5%, 100, 70 images per category are randomly selected as training set and the rest as test set in each test.

The evaluation results are reported in Table 3. PHOW [33] is a state-of-the-art feature descriptor for scene recognition, and Med-LDA [17] is a discriminative learning approach for LDA that has been shown to be superior to disLDA [18]. MSS outperforms all compared methods on all three datasets. The performance of SS is slightly inferior to FESS but slightly better than FS, which indicates that even though SS does not fully exploit the model parameters, it still captures rich generative information. We also evaluate the feature mapping as a function of the number of samples and topics, as shown in Fig. 2, and show that MSS works well with few topics (hidden variables).

5.3 Computational Efficiency

The proposed discriminative learning method is an iterative process, involving the inference step and the parameter estimation step, where the parameter estimation is slower because it needs to solve a quadratic programming and update

the generative model. Learning can be greatly sped up in the following way. Instead of cycling through the steps T times, we can pre-train the generative models (e.g. with T iterations) and then launch Algorithm 1 for a few iterations, about 10 iterations empirically.

The performance of a score space, to some extent, depends on the number of hidden variables (e.g., the number of topics in LDA and the number of hidden states in HMMs) in the generative model used [10,19]. We investigate this dependency by evaluating the method's performance as a function of the number of hidden variables. We evaluate the score space using two classification schemes: (1) multi-class classification; (2) splitting the multi-class problem into a group of binary classification problems and averaging their results. We find that the two methods of evaluation share very similar trend, and report only experimental results based on (2) in Fig. 1 (HMMs), and Fig. 2 (LDA). Overall, we found the proposed method works well with generative models with a few hidden variables, fewer than other methods required, which also makes our method more efficient. In addition, as shown in Fig. 1 and Fig. 2, MSS's performance over other methods is robust against the percentage of total samples used as training samples.

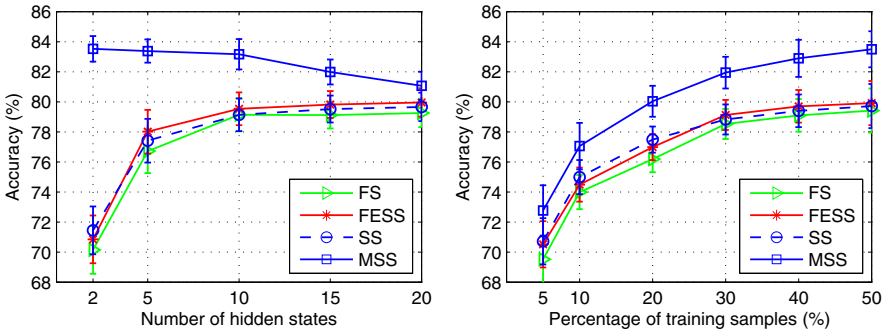

Fig. 1. Accuracy (%) w.r.t. the number of hidden states (HMMs) and the percentage of training samples on Chicken data

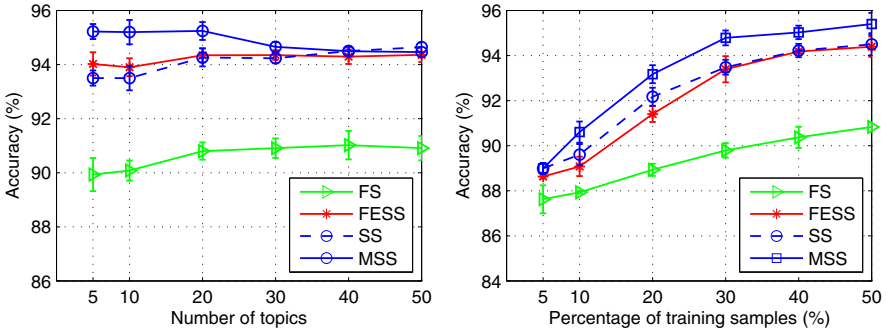

Fig. 2. Accuracy (%) w.r.t. the number of topics (LDA) and the percentage of training samples on OT (City vs rest)

Empirically, The MSS-HMM and MSS-LDA can be faster than MAP-HMM and MAP-LDA in part because the number of topics or hidden variables are smaller.

6 Conclusions

In this paper, we derive a new score space (SS) by decomposing the lower bound of the log likelihood into a linear combination of two parts. The first part is related to model parameters while the second part is related data samples. The second part, based mainly on sufficient statistics, provides the score functions to span the score space. This decomposition allows us to develop a computationally tractable method to learn score space discriminatively, subject to margin constraints of a classifier over the score space. We provide an EM-like algorithm for inference and learning, where the posterior introduced by discriminative factors (margin constraints) feed-back discriminative information to tune the score space. This method works well with a small number of hidden variables, which makes inference and learning fast and efficient. We show that this approach is competitive against other state-of-the-art methods in a variety of datasets.

Acknowledgement. Thanks NSF CISE IIS 0713206, 973 Program 2011CB302203, NSFC 60833009, NSFC 60975012 for support, and to Liwei Wang, Jiaya Jia and Zhuowen Tu for helpful discussions. This work was done when X. Li visited the Computer Science Department at CMU with support from the Chinese Scholarship Council.

References

1. Ng, A., Jordan, M.: On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. In: NIPS (2002)
2. Vapnik, V.: The nature of statistical learning theory. Springer (2000)
3. Gönen, M., Alpaydin, E.: Localized multiple kernel learning. In: ICML (2008)
4. Jaakkola, T., Meila, M., Jebara, T.: Maximum entropy discrimination. TR AITR-1668, MIT (1999)
5. Raina, R., Shen, Y., Ng, A., McCallum, A.: Classification with hybrid generative/discriminative models. In: NIPS (2004)
6. Sha, F., Saul, L.: Large margin hidden markov models for automatic speech recognition. In: NIPS (2007)
7. Li, X., Zhao, X., Fu, Y., Liu, Y.: Bimodal gender recognition from face and fingerprint. In: CVPR (2010)
8. Perina, A., Cristani, M., Castellani, U., Murino, V., Jojic, N.: Free energy score spaces: using generative information in discriminative classifiers. IEEE Trans. on PAMI 34(7), 1249–1262 (2012)
9. Li, X., Lee, T., Liu, Y.: Stochastic feature mapping for pac-bayes classification. arXiv:1204.2609 (2012)
10. Li, X., Lee, T., Liu, Y.: Hybrid generative-discriminative classification using posterior divergence. In: CVPR (2011)
11. Jaakkola, T., Haussler, D.: Exploiting generative models in discriminative classifiers. In: NIPS (1999)

12. Tsuda, K., Kawanabe, M., Ratsch, G., Sonnenburg, S., Muller, K.: A new discriminative kernel from probabilistic models. Neural Computation 14(10), 2397–2414 (2002)
13. Vedaldi, A., Zisserman, A.: Efficient additive kernels via explicit feature maps. IEEE Trans. on PAMI 34(3), 480–492 (2012)
14. Holub, A.D., Welling, M., Perona, P.: Hybrid generative-discriminative visual categorization. International Journal of Computer Vision 77(1-3), 239–258 (2008)
15. Chatfield, K., Lempitsky, V., Vedaldi, A., Zisserman, A.: The devil is in the details: an evaluation of recent feature encoding methods. In: BMVC (2011)
16. Collins, M.: Discriminative training methods for hidden markov models: theory and experiments with perceptron algorithms. In: EMNLP, pp. 1–8 (2002)
17. Zhu, J., Ahmed, A., Xing, E.P.: Medlda: maximum margin supervised topic models for regression and classification. In: ICML, pp. 1257–1264 (2009)
18. Lacoste-Julien, S., Sha, F., Jordan, M.: Disclda: Discriminative learning for dimensionality reduction and classification. In: NIPS (2008)
19. der Maaten, L.V.: Learning discriminative fisher kernels. In: ICML, pp. 217–224 (2011)
20. Smith, N., Gales, M.: Speech recognition using svms. NIPS 14, 1197–1204 (2001)
21. Gales, M.J.F., Watanabe, S., Fosler-Lussier, E.: Structured discriminative models for speech recognition: an overview. IEEE Signal Processing Magazine 29(6), 70–81 (2012)
22. Neal, R., Hinton, G.: A new view of the em algorithm that justifies incremental, sparse and other variants. Learning in Graphical Models, 355–368
23. Wainwright, M.J., Jordan, M.I.: Graphical Models, Exponential Families, and Variational Inference. Now Publishers Inc., Hanover (2008)
24. Jordan, M., Ghahramani, Z., Jaakkola, T., Saul, L.: An introduction to variational methods for graphical models. Machine Learning 37(2), 183–233 (1999)
25. Graça, J., Ganchev, K., Taskar, B.: Expectation maximization and posterior constraints. In: NIPS (2007)
26. Friedman, J., Hastie, T., Tibshirani, R.: The Elements of Statistical Learning. Spinger (2008)
27. Gilks, W.R., Wild, P.: Adaptive rejection sampling for gibbs sampling. Applied Statistics, 337–348 (1992)
28. Langford, J.: Tutorial on practical prediction theory for classification. JMLR 6(1), 273–306 (2006)
29. Chang, C.C., Lin, C.J.: Libsvm: a library for support vector machines. ACM Transactions on Intelligent Systems and Technology 2(3), 27 (2011)
30. Rabiner, L.: A tutorial on hidden Markov models and selected applications in speech recognition. Proceeding of the IEEE 77(2), 257–286 (1989)
31. Baum, L., Petrie, T., Soules, G., Weiss, N.: A maximization technique occurring in the statistical analysis of probabilistic functions of markov chains. The Annals of Mathematical Statistics 41(1), 164–171 (1970)
32. Mollineda, R., Vidal, E., Casacuberta, F.: Cyclic sequence alignments: approximate versus optimal techniques. IJPRAI 16, 291–299 (2002)
33. Vedaldi, A., Gulshan, V., Varma, M., Zisserman, A.: Multiple kernels for object detection. In: ICCV (2009)
34. Blei, D., Ng, A., Jordan, M.: Latent dirichlet allocation. JMLR 3, 993–1022 (2003)
35. Griffiths, T., Steyvers, M.: Finding scientific topics. PNAS 101(suppl. 1), 5228–5235 (2004)
36. Lowe, D.: Distinctive image features from scale-invariant keypoints. IJCV 60(2), 91–110 (2004)