Mach Learn
DOI 10.1007/s10994-015-5525-9

Stochastic feature mapping for PAC-Bayes classification

Xiong Li[1,2,3] · **Bin Wang**[4] · **Yuncai Liu**[1] · **Tai Sing Lee**[3]

Received: 16 November 2013 / Accepted: 6 July 2015
© The Author(s) 2015

Abstract Hidden information derived from probabilistic generative models of data distributions can be used to construct features for discriminative classifiers. This observation has motivated the development of approaches that attempt to couple generative and discriminative models together for classification. However, existing approaches typically feed features derived from generative models to discriminative classifiers, and do not refine the generative models or the feature mapping functions based on classification results. In this paper, we propose a coupling mechanism developed under the PAC-Bayes framework that can fine-tune the generative models and the feature mapping functions iteratively to improve the classifier's performance. In our approach, a stochastic feature mapping, which is a function over the random variables of a generative model, is derived to generate feature vectors for a stochastic classifier. We construct a stochastic classifier over the feature mapping and derive the PAC-Bayes generalization bound for the classifier, for both supervised and semi-supervised learning. This allows us to jointly learn the feature mapping and the classifier by minimizing the bound with an EM-like iterative algorithm using labeled and unlabeled data. The resulting framework integrates the learning of the discriminative classifier and the generative model and allows iterative fine-tuning of the generative models, and the feedforward feature mappings based on task performance feedback. Our experiments show, in three distinct

Editors: Vadim Strijov, Richard Weber, Gerhard-Wilhelm Weber, and Süreyya Ozogur Akyüz.

Thanks NSF CISE IIS 0713206, NSFC 61403090, 973 Program 2011CB302203 for support. Xiong Li and Bin Wang have equally contributed to this work.

✉ Xiong Li
flit.lee@gmail.com

1 Department of Automation, Shanghai Jiao Tong University, Shanghai, China

2 National Computer Network Emergency Response Technical Team, Beijing, China

3 Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, USA

4 College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai, China

applications, this new framework produces a general classification tool with state-of-the-art performance.

Keywords Stochastic feature mapping · PAC-Bayes generalization bound · Hybrid generative-discriminative classification

1 Introduction

Probabilistic generative models that seek to model data distributions are adept in exploiting hidden information, in dealing with structured data (e.g. protein sequence with variable length) and in solving nonlinear classification problems by means of maximum posteriori (MAP) classifiers, while discriminative models designed to find decision boundaries among different classes based on extracted features still furnish most of the state-of-the-art tools for classification. A number of promising methods (Jaakkola and Haussler 1999; Jaakkola et al. 1999; Raina et al. 2003; McCallum et al. 2006; Li et al. 2010, 2011; Perina et al. 2012) have been developed to exploit the complementarities of these two major paradigms (Jaakkola et al. 1999; Ng and Jordan 2002). These methods can be roughly categorized into two classes based on how they couple the generative and discriminative models: methods with explicit feature mappings (Jaakkola and Haussler 1999; Perina et al. 2012; Li et al. 2011) and methods without explicit feature mappings (Jaakkola et al. 1999; Raina et al. 2003; McCallum et al. 2006). In this paper, we focus on the first class as it is more flexible and can be directly used in discriminative classifiers.

Methods with explicit feature mappings, called generative feature mapping or generative score space (Jaakkola and Haussler 1999; Perina et al. 2012; Li et al. 2011), are motivated by the two findings revealed by earlier works in the context of classification: (1) generative models can provide useful information from their parameters and variables to construct feature mappings and simultaneously transform structured data of variable length into data in a fixed dimension feature space; (2) discriminative models are effective in finding decision boundaries in such a feature space. A feature mapping is a function over the hidden variables, observed variables and model parameters. It transforms a data point into a feature vector for the classifier. While these existing methods have tried to exploit the power of the generative models in uncovering hidden information, the generative models and the classifiers in these methods are insulated from each other and the resulting feature mappings could be sub-optimal. Thus, it is desirable to develop a closed-loop coupling mechanism that allows the generative models and the feature maps to be fine-tuned by the classification performance.

PAC-Bayes theory (McAllester 1999; Seeger 2002; McAllester 2003; Langford 2006; Lacasse et al. 2006; Germain et al. 2009; Seldin et al. 2012; Tolstikhin and Seldin 2013) potentially can provide a framework to learn feature mappings and classifiers jointly, allowing the fine tuning of feature mapping. PAC-Bayes is a theory proposed to bound the generalization error of classifiers, where classifiers are learned by minimizing the generalization bound with respect to the parameters of the classifiers over the training set. Similarly, feature mappings can also be learned by minimizing the generalization bound with respect to the quantities of feature mappings.

In this paper, we propose an approach based on the PAC-Bayes theory (McAllester 1999; Seeger 2002; McAllester 2003; Langford 2006; Lacasse et al. 2006; Germain et al. 2009; Seldin et al. 2012; Tolstikhin and Seldin 2013) to integrate the complementary strengths of generative and discriminative models. First we derive a stochastic feature mapping which is a

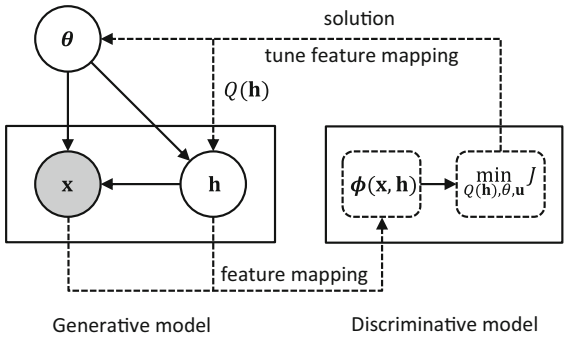

Fig. 1 A graphical illustration of the proposed approach. The generative model on the *left* provides hidden variables, data distribution and model parameters to construct a feature mapping for the discriminative model on the *right*. The classification performance of the discriminative model feeds back to tune the parameters of the generative model, which leads to the tuning of the feedforward feature mapping to improve classification performance

function over the observed and hidden variables of generative models. The feature mapping maps a data point to a stochastic feature. It is stochastic because it is constructed as the mean of multiple Gibbs samples of the generative model based on the observed data point. This is different from earlier methods (Jaakkola and Haussler 1999; Perina et al. 2012; Li et al. 2011) which map a data point to a deterministic feature. Further, we construct a Gibbs classifier to operate on the derived feature mapping, and derive a PAC-Bayes generalization bound that can be used to learn the classifier in supervised or semi-supervised manner. With the derived stochastic feature mapping and generalization bounds, learning is relatively simple. By minimizing the bound using an EM-like iterative algorithm, we obtain the analytic posterior over the hidden variables (E-step) and the set of simple update rules for the model's parameters (M-step). The derived posterior provides a bridge that allows the classifier to tune the generative models and consequently the feature mapping to improve classification performance. Our proposed framework is illustrated in Fig. 1 (Li et al. 2013).

The primary contributions of this paper are threefold:

(1) We derived a stochastic feature mapping that is effective in capturing generative (distribution and hidden) information in the data;
(2) We derived a PAC-Bayes generalization bound for the stochastic classifier over this stochastic feature mapping for both supervised and semi-supervised learning;
(3) We developed a joint learning approach to learn feature mapping and classifier by minimizing the derived bound.

Our proposed scheme offers a number of advantages over existing methods:

(1) The proposed stochastic feature mapping and its generalization bound can effectively be exploited to utilize hidden variables in the classification process, yielding state-of-the-art classification performance;
(2) The proposed method produces satisfactory performance when the 'capacity' of generative models is small, suggesting that it is efficient in both inference and learning;
(3) When the number of labeled training data is limited, the unlabeled data can be used to bootstrap the training of the classifier to improve performance.

In the remainder of this paper, we will first briefly review the related works in Sect. 2. We will then derive the feature mapping in Sect. 3. Section 4 constructs the stochastic classifier

Table 1 The notation list

Obj.	Description	Obj.	Description
$\mathbf{x}$	Input data	y	Output label
θ	Model parameter	$\mathbf{h}$	Hidden variables
$Q(\mathbf{h} \mid \mathbf{x})$	Posterior distribution	$Q(\mathbf{h})$	$\int Q(\mathbf{h} \mid \mathbf{x}) P(\mathbf{x}) d\mathbf{x}$
ϕ	Feature mapping	$\bar{\phi}$	$\bar{\phi} = \phi / \parallel \phi \parallel$
f_Q	Stochastic classifier	$\mathbf{w}$	Weight, $\mathrm{E}_Q[\mathbf{w}] = \mathbf{u}$
D	Unknown $P(\mathbf{x}, y)$	S	Training set with size $m = \lvert S \rvert$
$R_D(f_Q)$	True risk	$R_S(f_Q)$	Empirical risk
$e(f_Q)$	Risk for labeled data	$d(f_Q)$	Risk for unlabeled data
S_l	Labeled set with size $m_l = \lvert S_l \rvert$	S_u	Labeled set with size $m_u = \lvert S_u \rvert$

over the derived feature mapping, and drives the generalization bound for the classifier. In Sect. 5, we will present learning algorithms for the generative model, the feature mapping and the classifier simultaneously. In Sect. 6 we will evaluate the proposed method on three typical applications. We will conclude our contributions in Sect. 7. For readability, we have summarized the involved mathematical notations of this paper in Table 1.

2 Related works

2.1 Generative score spaces

Generative feature mapping (Jaakkola and Haussler 1999; Tsuda et al. 2002; Smith and Gales 2002; Holub et al. 2008; Perina et al. 2012; Li et al. 2011) is a class of methods that are designed to exploit the generative information for discriminative classification. Feature mappings are scores or measures computed over the generative models. They are functions over the observed variables, hidden variables, and parameters of generative models. The space spanned by a feature mapping is called as a score space or feature space.

Fisher score (FS) method (Jaakkola and Haussler 1999) derives feature mappings by measuring how a generative model's parameters affect the log likelihood of the data given the model. Let $\mathbf{x} \in \mathbb{R}^d$ be the observed variable and $P(\mathbf{x} \mid \theta)$ be its marginal distribution parameterized by a vector θ, the i-th component of the FS feature mapping is the differential with respect to the parameter θ_i,

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

Free energy score space (FESS) method (Perina et al. 2012) measures how well a data point fits random variables. The resulting feature mappings are the summation terms of log likelihood function. Posterior divergence (PD) (Li et al. 2011) derives a set of comprehensive measures that are related to both FS and FESS. These methods, working with classifiers, integrate the abilities of generative and discriminative models, and have produced very competitive performance in a variety of challenging tasks (Holub et al. 2008; Perina et al. 2012; Chatfield et al. 2011), including, for example, image recognition. However, in these methods, feature mappings and classifiers are learned independently, label information or classification performance was not fully utilized in the learning of feature mappings.

2.2 PAC-Bayes generalization bounds

PAC-Bayes (McAllester 1999; Seeger 2002; McAllester 2003; Langford 2006; Lacasse et al. 2006; Germain et al. 2009; Seldin et al. 2012; Tolstikhin and Seldin 2013) is a theory for bounding the generalization error of classifiers. A variety of PAC-Bayes generalization bounds (McAllester 1999; Seeger 2002; McAllester 2003; Langford 2006; Lacasse et al. 2006; Germain et al. 2009; Seldin et al. 2012; Tolstikhin and Seldin 2013) have been proposed for different classifiers such as deterministic classifiers, Gibbs classifiers (McAllester 1999), linear classifiers or nonlinear classifiers (e.g. Gaussian process (Seeger 2002)). Gibbs classifier, which we will use, is a stochastic classifier that usually operates under majority voting decision rules.

PAC-Bayes can bound classifiers built from different discriminative criteria, for example, the large margin criterion. The generalization bounds, derived from PAC-Bayes theory, can be expressed in two typical forms: an implicit form which bounds the difference between the empirical risk and the true risk (Seeger 2002; Langford 2006; Lacasse et al. 2006), or an explicit form which bounds the true risk directly (McAllester 2003; Germain et al. 2009). Besides, there are some tight bounds (Seldin et al. 2012; Tolstikhin and Seldin 2013) available. In this paper, we will focus on explicit bounds because they allow us to derive the analytic solution of the posteriors of hidden variables, with higher computational efficiency.

Our proposed method is related to transductive methods (Joachims 1999, 2003) which exploit both labeled data and unlabeled data for classification. Different with their methodology that explicitly infers the labels of unlabeled examples, our method instead minimizes the error rate of unlabeled examples. These methods work particularly well when the labeled training set is relatively small.

3 Stochastic feature mapping from free energy lower bound

Exploiting generative information, i.e., hidden variable, observed variable and data distribution, for discriminative classification (Jaakkola and Haussler 1999; Holub et al. 2008; Perina et al. 2012; Li et al. 2011) has shown promise in a variety of real world applications. A way to achieve this is to derive feature mapping from probabilistic generative models.

This section aims to derive a feature mapping to exploit generative information. Given a generative model with observed variable $\mathbf{x}$, hidden variable $\mathbf{h}$ and parameter θ, the *problem* is to find a feature mapping $\phi(\mathbf{x}, \mathbf{h})$ over the random variables. Our *method* is to fish out the informative components from the free energy the lower bound of log likelihood of generative models. The feature mapping takes a stochastic form rather than a deterministic form. The use of stochastic form makes it easier to derive and optimize the generalization bound. Further, the feature mapping is not an explicit function of parameters, simplifying the estimation procedure of model parameters (see Sect. 5.3).

3.1 Formulation

Let $P(\mathbf{x} \mid \theta)$ be the marginal distribution of a generative model parameterized by θ. Let $P(\mathbf{x}, \mathbf{h} \mid \theta)$ be its joint distribution where $\mathbf{h}$ is the set of hidden variables. As in Jaakkola and Haussler (1999), Perina et al. (2012) and Li et al. (2011), we choose to operate on the lower bound or negative free energy function of $\log P(\mathbf{x} \mid \theta)$ rather than $\log P(\mathbf{x} \mid \theta)$ because the lower bound of $\log P(\mathbf{x} \mid \theta)$ can be obtained even if $\log P(\mathbf{x} \mid \theta)$ itself is intractable. The lower bound is given by Jordan et al. (1999),

$$\log P(\mathbf{x} \mid \theta) \geq \mathrm{E}_{Q(\mathbf{h} \mid \mathbf{x})}[\log P(\mathbf{x}, \mathbf{h}) - \log Q(\mathbf{h} \mid \mathbf{x})] \triangleq F(\mathbf{x}, \theta) \tag{1}$$

where $Q(\mathbf{h} \mid \mathbf{x})$ is the variational approximate posterior of $P(\mathbf{h} \mid \mathbf{x})$. It is worth noting that the lower bound $F(\mathbf{x}, \theta)$ can be used here without loss of generality, because it is exactly equal to the log likelihood when $Q(\mathbf{h} \mid \mathbf{x})$ is expressive enough, i.e., $Q(\mathbf{h} \mid \mathbf{x})$ is given by exact inference.

Here, assuming that the generative model $P(\mathbf{x}, \mathbf{h} \mid \theta)$ belongs to the exponential family which covers most generative models, we arrive at the following general form,

$$P(\mathbf{x}, \mathbf{h}) = \exp\{\alpha(\theta)^T T(\mathbf{x}, \mathbf{h}) + A(\theta)\} \tag{2}$$

where θ is the vector of parameters; $T(\mathbf{x}, \mathbf{h})$ is the vector of sufficient statistics; $\alpha(\theta)$ and $A(\theta)$ are functions over parameter θ. Similarly, the prior is $P(\mathbf{h}) = \exp\{\alpha_h(\theta_h)^T T(\mathbf{h}) + A_h(\theta_h)\}$. Further, we assume that the posterior $Q(\mathbf{h} \mid \mathbf{x})$, given $\mathbf{x}$, takes the same form with its prior $P(\mathbf{h})$ but with different parameter (Jordan et al. 1999),

$$Q(\mathbf{h} \mid \mathbf{x}) = \exp\{\alpha_h(\hat{\theta}_h)^T T(\mathbf{h}) + A_h(\hat{\theta}_h)\} \tag{3}$$

Substituting the formulas of $P(\mathbf{x}, \mathbf{h})$ in Eq. (2) and $Q(\mathbf{h} \mid \mathbf{x})$ in Eq. (3) into Eq. (1), we have,

$$\begin{aligned} F(\mathbf{x}, \theta) &= \mathrm{E}_{Q(\mathbf{h} \mid \mathbf{x})}[\alpha(\theta)^T T(\mathbf{x}, \mathbf{h}) + A(\theta) - \alpha_h(\hat{\theta}_h)^T T(\mathbf{h}) - A_h(\hat{\theta}_h)] \\ &= (\alpha(\theta)^T, -\mathbf{1}^T, -1)^T \mathrm{E}_{Q(\mathbf{h} \mid \mathbf{x})}[\phi(\mathbf{x}, \mathbf{h})] + A(\theta) \end{aligned} \tag{4}$$

where $(\alpha(\theta)^T, -\mathbf{1}^T, -1)$ and $A(\theta)$ are functions over the model parameters; and the stochastic function,

$$\phi(\mathbf{x}, \mathbf{h}) = (T(\mathbf{x}, \mathbf{h})^T, (\mathrm{diag}(\alpha_h(\hat{\theta}_h))T(\mathbf{h}))^T, A_h(\hat{\theta}_h))^T \tag{5}$$

is a vector of explicit functions over $\mathbf{x}$ and $\mathbf{h}$, but not over θ. This means that $\phi(\mathbf{x}, \mathbf{h})$ will not be involved in the estimation of θ at the E-step (Sect. 5.3). The feature vector output by $\phi(\mathbf{x}, \mathbf{h})$ thus contains three groups of features. The first group comes from $T(\mathbf{x}, \mathbf{h})$, which is the sufficient statistics of the adopted generative model, based on both the hidden variables $\mathbf{h}$ and the observed variables $\mathbf{x}$. The second group of features come from $\mathrm{diag}(\alpha_h(\hat{\theta}_h))T(\mathbf{h})$, which is a score that measures how well the posterior explains the data $\mathbf{x}$. The third group of features come from $A_h(\hat{\theta}_h)$, which is a score related to the partition function of $Q(\mathbf{h} \mid \mathbf{x})$.

3.2 An illustrative example

To illustrate the above idea on feature mapping, we provide a simple example of feature mapping derived from a Gaussian mixture model with 3 mixture centers. This is illustrated in Fig. 2. Let $x \in \mathbb{R}$ be the observed variable; and the hidden variable be $\mathbf{h} = \mathbf{z} = (z_1, \ldots, z_3)^T$ which is a binary indicator vector assigning the example x to 3 mixture centers. That is, for each data point x, $\mathbf{z}$ can only be $(1, 0, 0)$, $(0, 1, 0)$, $(0, 0, 1)$, indicating which Gaussian (center) the data point is assigned to as a result of the MAP inference. In this case, the data vary along in 1D (i.e. x) and the examples from the three Gaussians are shown in the right top inset. Note that we assume there are in fact only two causes (circle vs. triangles) for the observation x. The goal is to map these data onto a new space in which the data points are easily separable into the two causes or classes.

The first two groups of features in the feature mapping ϕ in this case are:

$$\begin{aligned} T(x, \mathbf{z}) &= (z_1 x, z_1 x^2, z_1, z_2 x, z_2 x^2, z_2, z_3 x, z_3 x^2, z_3)^T \\ \mathrm{diag}(\alpha_z(\hat{\theta}_z))T(\mathbf{z}) &= (z_1 \log \hat{a}_1, z_2 \log \hat{a}_2, z_3 \log \hat{a}_3)^T \end{aligned}$$

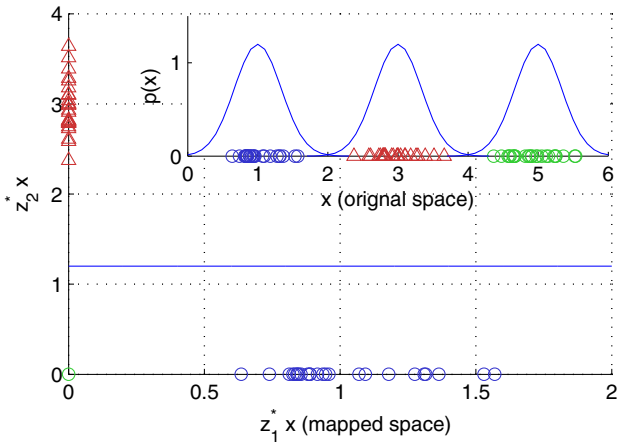

Fig. 2 An illustrative example for the proposed feature mapping. *Inset*: The raw data are generated from three Gaussian distributions, but we assume the data come from two classes (*circles* and *triangles*). *Main*: Gaussian mixture models with three mixture centers are used to model the data distribution. Each data point is to be inferred and assigned to one of the centers, which is indicated by the binary indicator vector $\mathbf{z} = (z_1, z_2, z_3)^T$. The feature mapping as described in the text and in more details in Sect. 6.2. $z_1 x$ and $z_2 x$ are two of the derived feature mapping functions which are stochastic. For illustration, we alternatively use their deterministic version $z_1^* x$, $z_1^* x$ where $\mathbf{z}^* = \max_{\mathbf{z}} P(\mathbf{z}|x)$ is given by MAP estimation. Note that, those points assigned to the third component are project to $(0, 0)$. When the raw data (*inset*), which are not linearly separable in the original space, are mapped to a new feature space spanned by these two feature mappings, they form distinct and linearly separable clusters

where $\hat{a}_i = \mathrm{E}_{Q(z)}[z_i]$ is the expectation of z_i over the posterior $Q(\mathbf{z} \mid x)$, which can be estimated by examples or taking expectation. The last group of features $A_z(\hat{\theta}_z) = 0$ because the partition function of multinomial distribution is $1 = e^0$.

Hence, each 1D data point x is mapped to a 12D feature space in this case. Figure 2 illustrates only two feature dimensions from $T(x, \mathbf{z})$, i.e. $z_1 x$ and $z_2 x$, which already produces a feature space in which the projected data points are linearly separable, greatly simplifying the classification problem.

4 Stochastic classifier and generalization bound

Given the stochastic feature mapping (Eq. 5), the *problem* of this section is to derive a generalization error bound for a stochastic classifier (Eq. 6) equipped with the feature mapping, for both supervised and semi-supervised learning. Our *method* is to decompose the risk term into two parts which are respectively for labeled data and unlabeled data. The error bound allows us to learn an effective feature mapping for classification in a discriminative manner by minimizing it with respect to the parameters of the feature mapping.

To obtain this error bound, we use a stochastic classifier over the feature mapping here. There are two reasons for our using a stochastic feature mapping and a stochastic classifier instead of a deterministic classifier: (1) the general setting of PAC-Bayes theory assumes a stochastic form which allows simple derivation of the generalization error bound; (2) the stochastic form also allows solving the resulting model in a simple algorithm.

4.1 Linear stochastic classifier over feature mapping

Let $\mathcal{X}$ be the input space consisting of an arbitrary subset of $\mathbb{R}^d$ and $\mathcal{Y} = \{-1, +1\}$ be the output space. An example is an input-output pair $(\mathbf{x}, y)$ where $\mathbf{x} \in \mathcal{X}$ and $y \in \mathcal{Y}$. With stochastic feature mapping $\phi(\mathbf{x}, \mathbf{h})$ derived in Eq. (5), we can construct a Gibbs classifier over this stochastic feature mapping:

$$f_Q = \text{sign}[\mathbf{w} \cdot \phi(\mathbf{x}, \mathbf{h})] \triangleq f_{\mathbf{w}}(\mathbf{x}, \mathbf{h}) \tag{6}$$

where $\mathbf{w} \sim Q(\mathbf{w})$ is the weight and $\mathbf{h} \sim Q(\mathbf{h})$; the posteriors $Q(\mathbf{w})$ and $Q(\mathbf{h})$ will be determined later in Sect. 5.3. A Gibbs classifier with an appropriate feature mapping ϕ is known to allow exploitation of the hidden variables in discriminative classifiers (Yu and Joachims 2009), and the PAC-Bayes bound for such a classifier can be tighter than VC bounds (Vapnik 2000).

4.2 Classification risk of stochastic classifier

In a PAC-Bayes setting (McAllester 1999), each example $(\mathbf{x}, y)$ is independently drawn from a fixed but unknown probability distribution D on $\mathcal{X} \times \mathcal{Y}$. Let $f(\mathbf{x}, \mathbf{h}) : \mathcal{X} \to \mathcal{Y}$ be any classifier with an auxiliary variable $\mathbf{h} \in \mathcal{H}$. Let $Q(f)$ be a posterior distribution over a space $\mathcal{F}$ of classifiers conditioned on the whole training set; and $Q(\mathbf{h})$ be the posterior distribution over a space $\mathcal{H}$ of hidden variables. Let $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$ be the training set whose examples are independently drawn. Consider a Gibbs classifier f_Q that first chooses a classifier f according to $Q(f)$ and a variable $\mathbf{h}$ according to $Q(\mathbf{h})$, and then classifies an example $\mathbf{x}$. The *true risk* $R_D(f_Q)$ and the *empirical risk* $R_S(f_Q)$ of this Gibbs classifier can be given by the following expressions:

$$R_D(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f)}\left[\mathrm{E}_{(\mathbf{x},y)\sim D}\,\mathrm{I}(f(\mathbf{x}, \mathbf{h}) \neq y)\right] \tag{7}$$

$$R_S(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f)}\left[\frac{1}{m}\sum\nolimits_{i=1}^{m}\mathrm{I}(f(\mathbf{x}_i, \mathbf{h}) \neq y_i)\right] \tag{8}$$

where $Q(\mathbf{h}) = \int Q(\mathbf{h} \,|\, \mathbf{x})P(\mathbf{x})d\mathbf{x}$ depends on the whole training set instead of any specific example $\mathbf{x}$; $m = |S|$ is the number of training examples; $\mathrm{I}(a)$ is the indicator function which outputs 1 if a is true and outputs 0 otherwise. $R_D(f_Q)$ and $R_S(f_Q)$ can be decomposed as follows.

Lemma 1 *Let $S = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_m, y_m)\}$ be a set of independently drawn examples. Let $f_1 \sim Q$ and $f_2 \sim Q$ be two independent and identically distributed random variables. The empirical risk $R_S(f_Q)$ in Eq. (8) and the true risk $R_D(f_Q)$ in Eq. (7) can be decomposed as follows,*

$$R_S(f_Q) = e_S(f_Q) + \frac{1}{2}d_S(f_Q)$$

$$R_D(f_Q) = e_D(f_Q) + \frac{1}{2}d_D(f_Q)$$

where

$$e_S(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f_1)Q(f_2)}\left[\frac{1}{m}\sum\nolimits_{i=1}^{m}\mathrm{I}(f_1(\mathbf{x}_i, \mathbf{h}) \neq y_i)\mathrm{I}(f_2(\mathbf{x}_i, \mathbf{h}) \neq y_i)\right]$$

$$d_S(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f_1)Q(f_2)}\left[\frac{1}{m}\sum\nolimits_{i=1}^{m}\mathrm{I}(f_1(\mathbf{x}_i, \mathbf{h}) \neq f_2(\mathbf{x}_i, \mathbf{h}))\right]$$

$$e_D(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f_1)Q(f_2)}\left[\mathrm{E}_{(\mathbf{x},y)\sim D}\,\mathrm{I}(f_1(\mathbf{x},\mathbf{h}) \neq y)\mathrm{I}(f_2(\mathbf{x},\mathbf{h}) \neq y)\right]$$

$$d_D(f_Q) = \mathrm{E}_{Q(\mathbf{h})Q(f_1)Q(f_2)}\left[\mathrm{E}_{\mathbf{x}\sim D}\left[\mathrm{I}(f_1(\mathbf{x},\mathbf{h}) \neq f_2(\mathbf{x},\mathbf{h}))\right]\right]$$

The proof of this Lemma can be found in the Appendix. Noticing that, the classifier $f = \mathrm{sign}[\mathbf{w} \cdot \phi(\mathbf{x},\mathbf{h})]$ (Eq. 6) is parameterized by $\mathbf{w}$, therefore $f_1 \sim Q(f)$ means $\mathbf{w}_1 \sim Q(\mathbf{w})$, i.e., the weight $\mathbf{w}$ of a stochastic classifier is sampled from the posterior distribution of f. e_S is a measure of the variance of the classification error, and is estimated from labeled data. d_S measures the disagreement of the classification, and is estimated from the unlabeled data.

4.3 Generalization bound for classification risk

Having defined the stochastic classifier over the feature mapping and derived the classification risks, we now proceed to derive the generalization bound for the classifier using PAC-Bayes theory. We can learn the stochastic feature mapping discriminatively and train the stochastic classifier over the feature mapping by minimizing the error bound.

In this derivation, although there are some tighter bounds (Seldin et al. 2012; Tolstikhin and Seldin 2013) available, we prefer explicit bounds for the true risk $R_D(f_Q)$, which allows an analytical derivation of the posterior Q. We choose to bound the true risk following the one-side version in McAllester (2003) and use the explicit bound in Keshet et al. (2011). Considering the measures $\mathrm{kl}(q \parallel p) = q \ln \frac{q}{p} + (1-q) \ln \frac{1-q}{1-p}$ and $\mathrm{KL}(Q \parallel P) = \mathrm{E}_Q[\log \frac{Q}{P}]$, we have the following bound.

Theorem 1 *For any distribution D over $\mathcal{X} \times \mathcal{Y}$, any space $\mathcal{F}$ of classifiers, any space $\mathcal{H}$ of hidden variables* $\mathbf{h}$ *of generative models, any distribution P over $\mathcal{F} \times \mathcal{H}$, any $\delta \in [0,1)$, $\epsilon > 0$, with probability at least $1-\delta$, the inequality holds simultaneously for all posteriors Q,*

$$\ell_D(f_Q) \leq \sup\left\{\epsilon : \mathrm{kl}(\ell_S(f_Q) \,\|\, \epsilon) \leq \frac{1}{m}\left(\mathrm{KL}_\ell(Q \,\|\, P) + \ln\frac{m+1}{\delta}\right)\right\}$$

where $\mathrm{KL}_\ell(Q \parallel P) = \alpha_\ell \mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x})) \parallel P(\mathbf{h} \,|\, \mathbf{x}))$, $m = |S|$ *where $\alpha_\ell = 1$ if $\ell(f_Q)$ is $R(f_Q)$ and $\alpha_\ell = 2$ if $\ell(f_Q)$ is $e(f_Q)$ or $d(f_Q)$.*

The proof of this theorem is summarized in the Appendix. Note that, the theorem differs from the bounds in McAllester (2003), Seeger (2002) and Lacasse et al. (2006) by the extra variable $\mathbf{h}$ introduced along with the stochastic feature mapping. This bound has a parameter ϵ and is difficult to minimize. However, in the following theorem, we can formulate this bound into a more practical bound that can be minimized directly.

Theorem 2 *For any distribution D over $\mathcal{X} \times \mathcal{Y}$, any space $\mathcal{F}$ of classifiers, any space $\mathcal{H}$ of hidden variables* $\mathbf{h}$ *of generative models, any distribution P over $\mathcal{F} \times \mathcal{H}$, any $\delta \in [0,1)$, with probability at least $1-\delta$, the inequality holds simultaneously for all posteriors Q,*

$$\ell_D(f_Q) \leq \inf_{\lambda > 1/2} \frac{1}{1-\frac{1}{2\lambda}}\left[\ell_S(f_Q) + \frac{\lambda}{m}\left(\mathrm{KL}_\ell(Q \,\|\, P) + \ln\frac{m+1}{\delta}\right)\right]$$

where $\mathrm{KL}_\ell(Q \parallel P) = \alpha_\ell \mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x})) \parallel P(\mathbf{h} \,|\, \mathbf{x}))$, $m = |S|$ *where $\alpha_\ell = 1$ if $\ell(f_Q)$ is $R(f_Q)$ and $\alpha_\ell = 2$ if $\ell(f_Q)$ is $e(f_Q)$ or $d(f_Q)$.*

The proof of the theorem can be found in the Appendix. Here we extend the bound to accommodate both labeled and unlabeled data for semi-supervised learning. Letting S_l be the labeled training set, S_u be the unlabeled training set, $S = S_u \cup S_l$, we have the following theorem.

Theorem 3 *For any distribution D over $\mathcal{X} \times \mathcal{Y}$, any space $\mathcal{F}$ of classifiers, any space $\mathcal{H}$ of hidden variables* $\mathbf{h}$ *of generative models, any distribution P over $\mathcal{F} \times \mathcal{H}$, any $\delta \in (0, 1]$, with probability at least $1 - \delta$, the inequality holds simultaneously for all posteriors Q,*

$$R_D(f_Q) \leq \inf_{\lambda_l > 1/2} \frac{1}{1 - \frac{1}{2\lambda_l}} \left[e_{S_l}(f_Q) + \frac{\lambda_l}{m_l} \left(\mathrm{KL}_e(Q \,\|\, P) + \ln \frac{m_l + 1}{\delta} \right) \right] + \inf_{\lambda_u > 1/2} \frac{1/2}{1 - \frac{1}{2\lambda_u}} \left[d_S(f_Q) + \frac{\lambda_u}{m} \left(\mathrm{KL}_d(Q \,\|\, P) + \ln \frac{m + 1}{\delta} \right) \right]$$

where $\mathrm{KL}_e = \mathrm{KL}_d = 2\mathrm{KL}(Q(f) \,\|\, P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}))$ *and* $m_l = |S_l|$, $m = |S|$.

The proof of this theorem can be found in the Appendix.

Remarks This bound allows classifiers to exploit unlabeled data, since $d_S(f_Q)$ dose not involve class label. Minimizing $d_S(f_Q)$ will contract the posteriors over the stochastic classifier and the stochastic feature space, reducing the uncertainty or ambiguity in classification and feature mappings. In the above bound, we use the $S = S_u \cup S_l$ instead of S_u to build the risk term for unlabeled data, because the labeled set S_l can be simultaneously used as the unlabeled set. Noticing that, the above semi-supervised bound is different with that in Lacasse et al. (2006) which is over the variance of the classification risk.

Also we derived a semi-supervised bound on the basis of the explicit bound proposed in Germain et al. (2009). However, in the experiments, we found that the solutions to the classifier and the generative model are difficult to find by optimization, as they are sensitive to the specification of parameters and the initial weights of the classifier (Germain et al. 2009). In the remainder of this paper, we will show that the bound derived in Theorem 3 can be minimized effectively using an EM-like algorithm and can produce generative model and classifier solutions that yield satisfied classification performance.

5 Learning and inference

Learning the stochastic feature mapping and classifier, in the sense of generalization error minimization, requires to minimize the bound in Theorem 3. This is equal to minimizing the right side of the inequality for specified λ_l and λ_u (Keshet et al. 2011). Our *method* is to optimize the bound using an EM-like iterative algorithm. To simplify the solution and improve optimization effectiveness, we specify $\lambda_u = \lambda_l$. Given the labeled training set S_l with the size $m_l = |S_l|$ and the unlabeled training set S_u with the size $m_u = |S_u|$, $S = S_u \cup S_l$ with the size $m = |S| = m_l + m_u$, the objective function can be expressed as,

$$J = e_{S_l}(f_Q) + \frac{1}{2} d_S(f_Q) + \left(\frac{\lambda_l}{m_l} + \frac{\lambda_u}{2m} \right) \mathrm{KL}_e(Q \,\|\, P) \tag{9}$$

where $\mathrm{KL}_e(Q \,\|\, P) = 2\mathrm{KL}(Q(f) \,\|\, P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}))$ which is the sum of the objective functions for the stochastic classifier (Eq. 6) and the objective function for the generative model (Eq. 1). To minimize J, we need the expressions for $\mathrm{KL}(Q(f) \,\|\, P(f))$, $\mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}))$, $e_{S_l}(f_Q)$ and $d_S(f_Q)$ which will be given in the next section.

5.1 Specification and expression

To derive the four expressions required in Eq. (9), we first need to specify the form of stochastic classifier. We consider the linear stochastic classifier in Eq. (6). In this case, $f_Q = f_{\mathbf{w}}$. Then, as were done in Langford (2006), we choose the prior of the weight $\mathbf{w}$ to be Gaussian $P(\mathbf{w}) = N(0, \mathrm{I})$ and its posterior also to be Gaussian but with a different mean, $Q(\mathbf{w}) = N(\mathbf{u}, \mathrm{I})$.

Using the above specifications of $P(\mathbf{w})$ and $Q(\mathbf{w})$, and applying the Gaussian integrals (Langford 2006), we have,

$$\mathrm{E}_{Q(\mathbf{w})}\, \mathbf{I}(f_{\mathbf{w}}(\mathbf{x}, \mathbf{h}) \neq y) = \Phi\left(y\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}, \mathbf{h})\right) \tag{10}$$

where $\Phi(a) = \int_a^\infty \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{x^2}{2}\right)dx = \frac{1}{2}\mathrm{erfc}(\frac{a}{\sqrt{2}})$; $\bar{\mathbf{u}} = \frac{\mathbf{u}}{\|\mathbf{u}\|}$ and the normalized feature $\bar{\phi} = \frac{\phi(\mathbf{x},\mathbf{h})}{\|\phi(\mathbf{x},\mathbf{h})\|}$. Further, considering Eq. (10), we have the integration:

$$\begin{aligned}\mathrm{E}_{Q(\mathbf{w}_1)Q(\mathbf{w}_2)}\, \mathrm{I}(f_{\mathbf{w}_1} \neq f_{\mathbf{w}_2}) &= \mathrm{E}_{Q(\mathbf{w}_1)Q(\mathbf{w}_2)}\, 2\mathrm{I}(f_{\mathbf{w}_1} \neq 1)\mathrm{I}(f_{\mathbf{w}_2} \neq -1) \\ &= 2\Phi\left(\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}, \mathbf{h})\right) \Phi\left(-\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}, \mathbf{h})\right)\end{aligned} \tag{11}$$

Minimizing the above risk term drives $\Phi(\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}, \mathbf{h}))$ and $\Phi(-\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}, \mathbf{h}))$ apart, reducing the classification uncertainty. Substituting Eq. (10) into $e_{S_l}(f_Q)$ and Eq. (11) into $d_S(f_Q)$, we have the following expressions,

$$\begin{aligned} e_{S_l}(f_Q) &= \frac{1}{m_l} \sum\nolimits_{i=1}^{m_l} \mathrm{E}_{Q(\mathbf{h})}\left[\mathrm{E}_{Q(\mathbf{w}_1)Q(\mathbf{w}_2)} \prod\nolimits_{k=1}^{2} \mathrm{I}(f_{\mathbf{w}_k}(\mathbf{x}_i, \mathbf{h}) \neq y_i)\right] \\ &= \frac{1}{m_l} \sum\nolimits_{i=1}^{m_l} \mathrm{E}_{Q(\mathbf{h})}\left[\Phi(y_i\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}_i, \mathbf{h})^2\right] \end{aligned} \tag{12}$$

$$\begin{aligned} d_S(f_Q) &= \frac{1}{m} \sum\nolimits_{i=1}^{m} \mathrm{E}_{Q(\mathbf{h})}\left[\mathrm{E}_{Q(\mathbf{w}_1)Q(\mathbf{w}_2)} \mathrm{I}(f_{\mathbf{w}_1}(\mathbf{x}_i, \mathbf{h}) \neq f_{\mathbf{w}_2}(\mathbf{x}_i, \mathbf{h}))\right] \\ &= \frac{2}{m} \sum\nolimits_{i=1}^{m} \mathrm{E}_{Q(\mathbf{h})}\left[\Phi(\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}_i, \mathbf{h}))\Phi(-\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}_i, \mathbf{h}))\right] \end{aligned} \tag{13}$$

Further, with the specifications of $Q(\mathbf{w})$ and $P(\mathbf{w})$, their KL divergence is,

$$\mathrm{KL}(Q(\mathbf{w}) \,\|\, P(\mathbf{w})) = \frac{1}{2}\|\mathbf{u}\|^2 \tag{14}$$

And the expression of $\mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}))$ over the training set S is,

$$\frac{1}{m} \sum\nolimits_{i=1}^{m} \mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}_i) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}_i)) \tag{15}$$

5.2 The objective function

Having the expressions for $\mathrm{KL}(Q(\mathbf{w}) \,\|\, P(\mathbf{w}))$ (Eq. 14), $e_{S_l}(f_Q)$ (Eq. 12), $d_S(f_Q)$ (Eq. 13) and $\mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \,|\, \mathbf{x}) \,\|\, P(\mathbf{h} \,|\, \mathbf{x}))$ (Eq. 15), for brevity, letting $\tilde{m}_\lambda = (\frac{\lambda_l}{m_l} + \frac{\lambda_u}{2m})^{-1}$, the objective function in Eq. (9) over the labeled training set S_l and the unlabeled training set S_u can be expressed as:

$$\begin{aligned} J &= e_{S_l}(f_Q) + \frac{1}{2}d_{S_u}(f_Q) + \frac{1}{\tilde{m}_\lambda}\mathrm{KL}_e(Q \,\|\, P) \\ &= \frac{1}{m_l} \sum\nolimits_{i=1}^{m_l} \mathrm{E}_{Q(\mathbf{h})}\left[\Phi(y_i\bar{\mathbf{u}} \cdot \bar{\phi}(\mathbf{x}_i, \mathbf{h}))^2\right] \end{aligned}$$

$$+\frac{1}{m}\sum_{i=1}^{m}\mathrm{E}_{Q(\mathbf{h})}\left[\Phi(\bar{\mathbf{u}}\cdot\bar{\phi}(\mathbf{x}_i,\mathbf{h}))\Phi(-\bar{\mathbf{u}}\cdot\bar{\phi}(\mathbf{x}_i,\mathbf{h}))\right]$$
$$+\frac{1}{2\tilde{m}_\lambda}\|\mathbf{u}\|^2+\frac{1}{\tilde{m}_\lambda m}\sum_{i=1}^{m}\mathrm{KL}(Q(\mathbf{h}\,|\,\mathbf{x}_i)\,\|\,P(\mathbf{h}\,|\,\mathbf{x}_i)) \tag{16}$$

where the first and second terms are estimated by labeled and unlabeled data respectively. Learning the classifier with the feature mapping function ϕ embedded in it is to minimize J with respect to the unknown quantities $\mathbf{u}$, θ and $Q(\mathbf{h}\,|\,\mathbf{x}_i)$, subject to $\int Q(\mathbf{h}\,|\,\mathbf{x}_i)\,d\,\mathbf{h}=1$, for fixed values of λ_l and λ_u (Keshet et al. 2011).

The unlabeled data benefit the classifier in two ways: (1) by shaping the feature space so that the mapped features are more distinct for the classifier; (2) by providing more data to train generative models. We will describe an EM-like iterative algorithm (Jordan et al. 1999) that can be used to minimize J in Eq. (16).

5.3 Inference and parameter estimation

In this section, we derive the EM-like iterative learning procedure to optimize the objective function in our proposed approach. In the first step, we fix $\mathbf{u}$ and θ, and minimize J with respect to $Q(\mathbf{h}\,|\,\mathbf{x}_i)$ (Eq. 16), subject to $\int Q(\mathbf{h}\,|\,\mathbf{x}_i)d\mathbf{h}=1$. This is a standard posterior regularization problem (Graça et al. 2007) which can be solved using the method of Lagrange multipliers. Note that, the objective functions for labeled data and unlabeled data are different. For each labeled example $\mathbf{x}_i\in S_l$, as derived in the Appendix, we have,

$$Q(\mathbf{h}|\mathbf{x}_i)\propto P(\mathbf{h}_i,\mathbf{x}_i)\exp\left\{-\frac{\tilde{m}_\lambda m}{m_l}\Phi\left(y\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)^2-\tilde{m}_\lambda\Phi\left(\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\Phi\left(-\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\right\} \tag{17}$$

where $\bar{\phi}_i$ is the short notation of $\bar{\phi}(\mathbf{x}_i,\mathbf{h})$. For each unlabeled example $\mathbf{x}_i\in S_u$, similarly we have,

$$Q(\mathbf{h}\,|\,\mathbf{x}_i)\propto P(\mathbf{h},\mathbf{x}_i)\exp\left\{-\tilde{m}_\lambda\Phi\left(\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\Phi\left(-\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\right\} \tag{18}$$

The fact that the classifier output is inside the expression for posteriors means that the generative models are being tuned when the classifier is being optimized during the minimization of the generalization bound. This tuning mechanism inhibits those examples of $\mathbf{h}$ that had lead to misclassification and promotes those with less misclassification.

In the second step, we fix $Q(\mathbf{h}\,|\,\mathbf{x}_i)$ and θ, and determine $\mathbf{u}$ (i.e., the mean of posterior $Q(\mathbf{w})$), by minimizing J with respect to $\mathbf{u}$. The gradient of J can be expressed as:

$$\frac{\partial J}{\partial\mathbf{u}}=\frac{1}{m_l}\sum_{i=1}^{m_l}\mathrm{E}_{Q(\mathbf{h}\,|\,\mathbf{x}_i)}\left[2\Phi\left(y_i\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)G(y_i\bar{\mathbf{u}}\cdot\bar{\phi}_i)\,y_i\,\bar{\phi}_i\right]\frac{\partial\bar{\mathbf{u}}}{\partial\mathbf{u}}+\frac{1}{\tilde{m}_\lambda}\mathbf{u}$$
$$+\frac{1}{m}\sum_{i=1}^{m}\mathrm{E}_{Q(\mathbf{h}\,|\,\mathbf{x}_i)}\left[G(\mathbf{u}\cdot\bar{\phi}_i)\left(\Phi(\bar{\mathbf{u}}\cdot\bar{\phi}_i)-\Phi(-\bar{\mathbf{u}}\cdot\bar{\phi}_{ik})\right)\bar{\phi}_i\right]\frac{\partial\bar{\mathbf{u}}}{\partial\mathbf{u}}$$

where $G(\cdot)$ is a gaussian function with zero-mean and unit variance; n is the number of examples drawn from $Q(\mathbf{h}\,|\,\mathbf{x}_i)$. We use *rejection sampling* to draw examples from this posterior, where $P(\mathbf{h},\mathbf{x}_i)$ can be used as the comparison function due to $\exp(\cdot)\le 1$ ($\Phi\ge 0\Rightarrow\exp(\cdot)\le 1$). First, we draw the examples of $\mathbf{h}$ from $P(\mathbf{h},\mathbf{x}_i)$ using Gibbs sampling. Second, for the drawn example $\mathbf{h}_{ik}$, we reject it if $Q(\mathbf{h}_{ik}\,|\,\mathbf{x}_i)<r_k$ and accept it otherwise, where r_k is an example randomly drawn from the uniform distribution over $[0,\,P(\mathbf{h}_{ik},\mathbf{x}_i)]$. The accepted example are the examples of $Q(\mathbf{h}\,|\,\mathbf{x}_i)$. Then we have,

$$\frac{\partial J}{\partial \mathbf{u}} \approx \frac{1}{m_l n}\sum_{i,k=1}^{m_l,n} 2\Phi\left(y_i \bar{\mathbf{u}}\cdot\bar{\phi}_{ik}\right) G(y_i\bar{\mathbf{u}}\cdot\bar{\phi}_{ik})\, y_i\, \bar{\phi}_{ik}\frac{\partial \bar{\mathbf{u}}}{\partial \mathbf{u}} + \frac{1}{\tilde{m}_\lambda}\mathbf{u} + \frac{1}{mn}\sum_{i,k=1}^{m,n} G(\mathbf{u}\cdot\bar{\phi}_{ik})\left[\Phi(\bar{\mathbf{u}}\cdot\bar{\phi}_{ik}) - \Phi(-\bar{\mathbf{u}}\cdot\bar{\phi}_{ik})\right]\bar{\phi}_{ik}\frac{\partial \bar{\mathbf{u}}}{\partial \mathbf{u}} \tag{19}$$

In the third step, we fix $Q(\mathbf{h}\,|\,\mathbf{x}_i)$, $\mathbf{u}$ and solve parameters θ. Note only the last term of Eq. (16), i.e., the objective function of the generative model, involves θ. So the update rules for θ in this joint learning model, derived by minimizing Eq. (16) with respect to θ, are the same as the update rules of the original generative model, i.e.,

$$\begin{aligned}\theta &= \max_\theta \sum_{i=1}^{m} \mathrm{KL}(Q(\mathbf{h}\,|\,\mathbf{x}_i)\,\|\,P(\mathbf{h}\,|\,\mathbf{x}_i,)) \\ &= \max_\theta \sum_{i=1}^{m} \mathrm{KL}(Q(\mathbf{h}\,|\,\mathbf{x}_i)\,\|\,P(\mathbf{x}_i,\mathbf{h}\,|\,\theta)) \\ &\approx \max_\theta \frac{1}{n}\sum_{i,k=1}^{m,n}[\log Q(\mathbf{h}_{ik}\,|\,\mathbf{x}_i) - \log P(\mathbf{x}_i,\mathbf{h}_{ik}\,|\,\theta)]\end{aligned} \tag{20}$$

The complete learning procedure of the proposed method is summarized in Algorithm 1. The classification procedure is summarized in Algorithm 2.

Algorithm 1 Inference and learning

input: data set $S = S_l \cup S_u$ where $|S| = m$
initialize $\hat{\mathbf{u}}_0$, $\delta = 0.05$, $\lambda_l = \lambda_u = 0.55$, $t = 1$ and learning rate $\gamma = 0.5$
pre-train the adopted generative model and output $\hat{\theta}_0$
repeat
  for $i = 1$ to m **do**
    sample from $Q(\mathbf{h}\,|\,\mathbf{x}_i)$ using Gibbs-rejection sampling (Eq. 17)-(18))
  end for
  $\hat{\mathbf{u}}^t \leftarrow \hat{\mathbf{u}}^{t-1} - \gamma\frac{\partial J(\hat{\theta}^{t-1})}{\partial \mathbf{u}}$ (Eq. 19)
  update $\hat{\theta}^t$ according to Eq. (20)
  $t \leftarrow t + 1$
until convergence
output: $\hat{\mathbf{u}}$, $\hat{\theta}$

Algorithm 2 Classification

input: example $\mathbf{x}_i$, parameters $\hat{\mathbf{u}}$, $\hat{\theta}$
sample $\{\mathbf{h}_{i1}, \cdots, \mathbf{h}_{in}\}$ from $Q(\mathbf{h}\,|\,\mathbf{x}_i)$ using Gibbs-rejection sampling (Eq. 18)
sample $\{\mathbf{w}_1, \cdots, \mathbf{w}_n\}$ from $Q(\mathbf{w})$
$\hat{y}_i = \max_y \sum_{k=1}^{n} \mathrm{I}(\mathrm{sign}[\mathbf{w}_k \cdot \phi(\mathbf{x}_i, \mathbf{h}_{ik})] = y)$ (majority voting using examples)
output: $\hat{y}_i$

5.4 A toy example

To demonstrate how the proposed approach works, we present a toy example using 2D synthetic data. The data points, belonging to two categories, are drawn from four Gaussian distributions. See Fig. 3a for illustration, where 'o' and '+' label two categories respectively, and color and gray markers respectively indicate training and test examples. For this is a nonlinear classification problem, we use SFM-GMM that is derived in Sect. 6.2 for demonstration, where the number of mixture centers is set to $K = 10$. The learning procedure and the classification procedure of SFM-GMM are respectively shown in Algorithms 1 and 2.

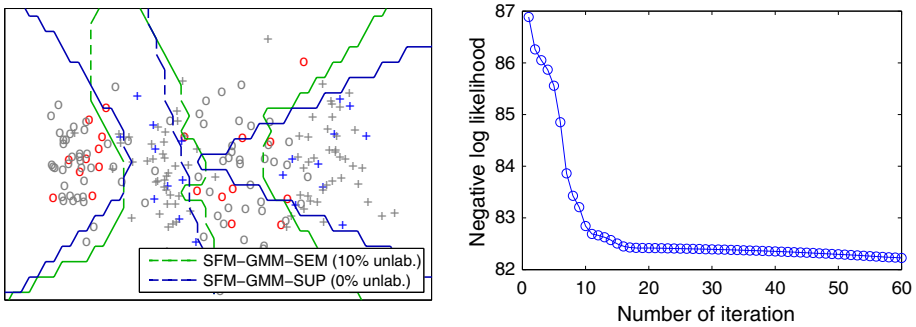

Fig. 3 Illustration of the toy example. **a** The decision bounds of the semi-supervised version (*green*) and supervised version (*blue*) of SFM-GMM. The data points are drawn from four Gaussian distributions where $\mu_1 = (10, 10)^T$, $\Sigma_1 = \text{diag}([8, 8])$, $\mu_2 = (20, 10)^T$, $\Sigma_2 = \text{diag}([10, 8])$, $\mu_3 = (30, 10)^T$, $\Sigma_3 = \text{diag}([12, 8])$ and $\mu_4 = (40, 10)^T$, $\Sigma_4 = \text{diag}([10, 8])$. **b** The negative log likelihood as the function of the number of iteration

Figure 3a visualizes the decision bounds of the supervised version (blue) and the semi-supervised version (green) of SFM-GMM, where the test accuracies are 78.13 and 81.25 % respectively. In general, both supervised and semi-supervised SFM-GMM can separate the two categories appropriately. Figure 3a presents the negative log likelihood for supervised SFM-GMM, as a function of the number of iterations. It can be found that, with the pre-trained GMM, our approach reaches convergence within about 20 iterations.

6 Experiments

In this section, we will evaluate the proposed stochastic feature mapping (SFM) and related methods empirically on general classification tasks, scene recognition and protein sequence classification respectively. We seek to demonstrate three advantages of SFM: (1) the proposed stochastic feature mapping and its generalization bound can effectively exploit information from generative models for classification, producing results that are competitive with several state-of-the-art methods; (2) SFM shows satisfactory performance when the 'capacity' of a generative model is small, meaning that SFM is efficient in inference and learning; (3) when the amount of labeled training data is small, unlabeled data can help train the generative models, resulting in improvement in performance.

6.1 Overall testing approach and evaluation strategies

For each of these multiple-class classification problems, we break it down to many binary classification problems, each of which is a *one-versus-rest classification* that distinguishes one class from all the others. For each binary problem, we test each binary classification problem on 20 random partitions, and report the average accuracy of the labeled data. For each application, we perform three groups of experiments to verify the three advantages of the proposed SFM method stated above: (1) we randomly partition the positive examples into 50 % training and 50 % test sets, and do so also for the negative examples; (2) we vary the capacity of generative models (e.g., the number of mixture centers) to evaluate how capacity affects performance; (3) in the semi-supervised scenario, we vary the percentage

of the labeled training examples and the unlabeled training examples to evaluate how and whether the unlabeled data improves the classifier performance.

For each problem, a generative model appropriate for the database has to be chosen. We used Gaussian mixture models (GMM) for the UCI datasets, latent Dirichlet allocation (LDA) for the scene dataset, and Hidden Markov models (HMM) for the protein sequence datasets. Thus, our approach is called SFM-GMM, SFM-LDA and SFM-HMM in the three different applications to indicate the generative models used.

In the three applications, we will compare the performance of our proposed approach SFM with a number of state-of-the-art classifiers, as detailed in the following list,

- **LMKL** (localized multiple kernel learning) (Gönen and Alpaydin 2008) is a state-of-the-art classifier. We use the authors' toolbox[1], where linear kernel and 2-degree polynomial kernel are chosen.
- **PBGD3** (PAC-Bayes gradient descent) (Germain et al. 2009) is a classifier also derived by minimizing a PAC-Bayes generalization bound, we implement this algorithm according to the authors' suggestions, with confidence parameter $\delta = 0.05$; C based on cross validation, and the random initial number $k = 10$.
- **SVM** (Supported Vector Machine) is a popular classifier. We use a popular toolbox libsvm (Chang and Lin 2011)[2] with a RBF kernel. The cost is set to $C = 1$, and the bandwidth parameter is chosen by cross validation around $\gamma = 1/\#\text{feature}$.
- **TSVM** (transducitve SVM) (Joachims 1999) is a state-of-the-art semi-supervised classifier. We use the toolbox[3] released by the authors, with the parameters chosen by cross validation.
- **MAP** (maximum a posteriori). Probabilistic generative models with a maximum the posteriori decision rule. The models are same with those used in FS and FESS.
- **SFM** (stochastic feature mapping, our approach). We implemented Algorithm 1. Since the solution of $\mathbf{u}$ could be trapped in local minima problem, we typically repeated the optimization $2\sim6$ times, with a different random initial point each time within the range $[-10, 10]$, to obtain a satisfactory solution. As discussed in Sect. 5, we augment the unlabeled set to $S_u \cup S_l$. The maximum iteration number is set to 20 for Experiment I and 30 for Experiments II and III.

Also, we compare our approach SFM with two feature mapping methods derived from generative models:

- **FS** (Fisher score) (Jaakkola and Haussler 1999). We implement FS-LDA and FS-HMM following the suggestions of the authors and (Chatfield et al. 2011). The parameters of generative models, i.e., the number of mixture centers, topics and hidden states, are chosen according to cross validation.
- **FESS** (free energy score space) (Perina et al. 2012). We implement FESS-LDA according to the authors' suggestion, and use the authors' toolbox for FESS-HMM[4].

6.2 Experiment I: deriving a general classification tool

In this experiment, we derive a general classification method by applying the proposed framework to a simple yet general generative model, the Gaussian mixture model. Let $\mathbf{x} \in \mathbb{R}^d$

[1] http://users.ics.aalto.fi/gonen/icml08.php.

[2] http://www.csie.ntu.edu.tw/~cjlin/libsvm/.

[3] http://svmlight.joachims.org/.

[4] http://profs.sci.univr.it/~perina/fess2.htm.

Table 2 Classification accuracy (%±std) on UCI database, with *one-versus-rest* scheme

Dataset	TSVM (Joachims 1999)	SVM (Chang and Lin 2011)	LMKL (Gönen and Alpaydin 2008)	PBGD3 (Germain et al. 2009)	SFM-GMM
Breast cancer	**96.91 ± 1.32**	96.79 ± 1.79	96.41 ± 0.97	93.98 ± 1.52	95.26 ± 0.93
Breast tissue	88.25 ± 5.74	83.37 ± 4.31	87.69 ± 5.24	88.14 ± 4.50	**89.61 ± 3.84**
Wine	95.61 ± 2.46	**97.36 ± 1.94**	95.48 ± 4.10	92.22 ± 8.56	96.11 ± 1.38
Sonar	75.29 ± 4.83	74.45 ± 3.22	80.21 ± 1.52	75.52 ± 5.70	**81.54 ± 2.93**
Credit approval	84.01 ± 1.72	84.61 ± 1.83	81.92 ± 1.41	83.53 ± 1.82	**85.06 ± 1.31**
SPECTF heart	78.27 ± 1.05	76.56 ± 2.97	80.38 ± 3.40	79.70 ± 0.65	**81.34 ± 0.46**
Libras movement	95.47 ± 2.12	91.74 ± 3.14	**96.58 ± 1.78**	94.52 ± 2.80	95.97 ± 3.12
Steel plates faults	88.60 ± 8.94	86.52 ± 9.03	**92.63 ± 8.14**	87.30 ± 8.26	90.24 ± 8.23

Bold values represent the best result on each experiment
'Credit approval' is the short of 'Australian Credit Approval' and 'Sonar' is the short of 'Connectionist Bench (Sonar, Mines vs. Rocks)'

be the observed variable; $\mathbf{z} = \{z_1, \ldots, z_K\}$ be the hidden binary indicator vector for K mixture centers; $\mathbf{a} = \{a_1, \ldots, a_K\}$ be the parameters of the approximate posterior of $\mathbf{z}$. We assume the covariance matrix be diagonal. The feature mapping of this model is,

$$\phi = (T(\mathbf{x}, \mathbf{z})^T, (\text{diag}(\alpha_z(\hat{\theta}_z))T(\mathbf{z}))^T, A_z(\hat{\theta}_z))^T$$

where,

$$T(\mathbf{x}, \mathbf{z}) = \left(z_1(\mathbf{x}^T, diag(\mathbf{x}\mathbf{x}^T)), \ldots, z_K(\mathbf{x}^T, diag(\mathbf{x}\mathbf{x}^T)), 1\right)^T$$

$$\text{diag}(\alpha_z(\hat{\theta}_z))T(\mathbf{z}) = (z_1 \log a_1, \ldots, z_K \log a_K)^T$$

and $A_z(\hat{\theta}_z) = 0$. The posterior of $\mathbf{z}$ can be easily derived from Eq. (17). The number of mixture centers is configured to $K = 4$ in these experiments, since $K = 4$ could produce satisfactory results for most datasets.

Here we select 8 datasets from UCI database for evaluation, preferring those without missing entities. The number of classes of each dataset is between 2 and 15. In each dataset, each example, such as a type of wine in the wine class, is described by a list of attributes, such as color intensity, acidity and hue for wine. The number of examples of each class varies from 14 and 673. The dimensionality is between 9 and 90. We compare our method SFM-GMM with SVM (Vapnik 2000), TSVM (Joachims 1999), LMKL (Gönen and Alpaydin 2008) and PBGD3 (Germain et al. 2009). 5 % unlabeled data is used to activate the semisupervised learning of TSVM. In each test, a dataset is randomly split to two parts, 50 % for training and the rest for test. The average results over 20 tests are reported in Table 2. It shows that SFM-GMM is adaptive to the distribution of each dataset to achieve consistent top or near the top performance for all datasets, outperforming other methods on half of the datasets.

The results of semi-supervised case are presented in Fig. 4. Figure 4a shows the classifiers' performance as a function of the number of mixture centers K. Results from three datasets using SFM-GMM are shown together with results from the state-of-the-art FS feature mapping which is however a deterministic mapping and is not tunable because the feature mapping and the classifier are learned separately. It can be observed that SFM-GMM has a significant performance gap over FS when the number of mixture centers is small (e.g.

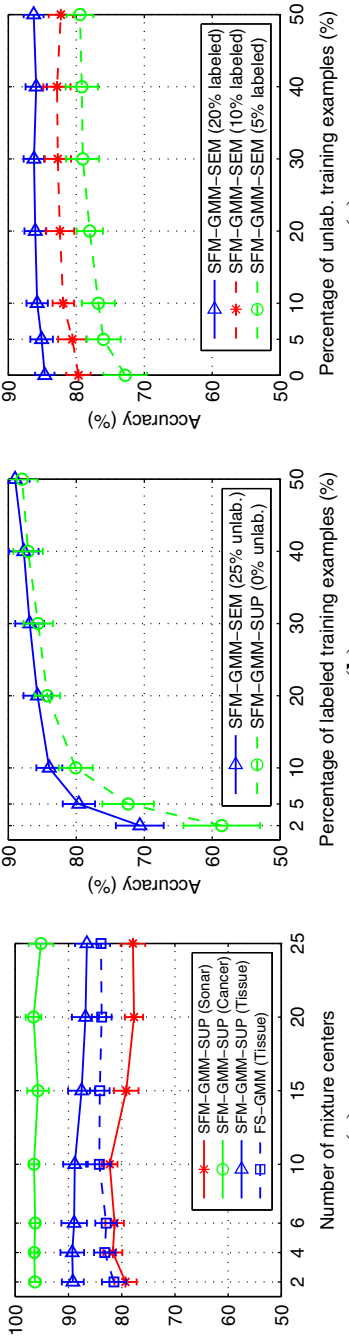

Fig. 4 Classification accuracy (%) of Tissue dataset in the UCI database, as a function of **a** the number of mixture centers or causes, where half examples from training and the rest half for test; **b** the number of training labeled examples, expressed in percentage of the total of 106 positive examples; **c** the number of unlabeled examples, expressed in percentage of the total of 106 positive examples. The examples are partitioned according two rules, (1) the size of labeled training set is equivalent to the size of test set; (2) the unlabeled training examples are selected from the remaining set after the labeled training set and test set are selected. If the remaining set is not enough, we select from the labeled training set and test set. For example, a partition with 20 % of labeled training examples and 25 % of unlabeled training examples indicates that 20 % of data are test examples and 35 % of data are not used in the experiments. 'SUP' and 'SEM' indicate supervised learning and semi-supervised learning respectively. 'unlab' is the short of unlabeled training examples

$K = 2, 4, 6$) for the 'Breast tissue' category. Also, the algorithm reaches convergence within 15 iterations. These imply that SFM-GMM is efficient computationally. Classification results for 'Sonar' and 'Breast cancer' classes are also shown to show that $K = 4$ is close to optimal for many classes in this data set.

Figure 4b demonstrates that when the amount of labeled data is small, introduction of unlabeled data yield improved performance, particularly when only $2 \sim 10$ percent of the labeled data are used in the training. When the amount of labeled data increased, the benefit of unlabeled data diminishes. Increasing the amount of unlabeled data in semi-supervised training produces performance benefit particularly when the amount of labeled data used is small, as shown in Fig. 4c. The diminished benefit of the unlabeled data when significant amount of labeled data is present in the training set is because the labeled examples have an increasing dominating effect.

This experiment shows that the proposed stochastic feature mapping and the feedback tuning mechanism in our approach could yield improvement for the general class of Gaussian mixture models for classification.

6.3 Experiment II: scene recognition using LDA

We evaluate our SFM method and compare its performance against comparable methods on a scene recognition task popular in computer vision. The distribution of a collection of visual words, typically some informative image patterns, or cluster center of image pattern descriptors, has found to be informative in this task. Such representation based on visual words is found to be relatively robust against topic variation and spatial position variation. We use latent Dirichlet allocation (LDA) (Blei et al. 2003) to model the distributions of visual words, and derive a recognition tool with our proposed framework. As in Griffiths and Steyvers (2004), we sample the topic variable using collapsed Gibbs sampling and reject examples according to the rule in Eq. (17). We fix the LDA model's parameter α and allow β (Griffiths and Steyvers 2004) to be updated. Note that α is the parameter of the distribution over the mixture of topics, or scene, and β is the parameter of the distribution over topics.

Let w, z respectively indicate word and topic, and γ be the parameter of the approximate posterior of z. The feature mapping of this model is given by Eq. (5). That is, $\phi = (T(\mathbf{w}, \mathbf{z})^T)^T, (\text{diag}(\alpha_z(\hat{\theta}_z))T(\mathbf{z}))^T, A_z(\hat{\theta}_z))^T$, where,

$$T(\mathbf{w}, \mathbf{z}) = (z_{11}, \ldots, z_{NK}, w_1 z_{11}, \ldots, w_N z_{NK})^T$$
$$\text{diag}(\alpha_z(\hat{\theta}_z))T(\mathbf{z}) = (z_{11} \log \gamma_{11}, \ldots, z_{NK} \log \gamma_{NK})^T$$

and $A_z(\hat{\theta}_z) = 0$, where n, i, k index word, term and topic respectively. For FS (Jaakkola and Haussler 1999) and FESS (Perina et al. 2012), we extract features from the trained LDA model and deliver to SVM. Cross-validation shows that the optimal number of topics for FS and FESS are both 50, and for SFM is 10 (see also Fig. 5a) for the particular scene database we will discuss next.

The OT scene dataset (Oliva and Torralba 2001) is chosen for evaluation. This dataset contains 2688 images, classified into 4 categories of artificial scenes and 4 categories of natural scenes, with $260 \sim 410$ images for each scene category. For each image, dense SIFT descriptors (Lowe 2004) are extracted from 20×20 grid patches over 4 scales. These descriptors are quantized to visual words using a code book (50 centers) obtained by clustering randomly selected descriptors. The distribution of occurrence frequency of visual words is represented as a histogram and used as an input feature for scene classification. The evaluation results are summarized in Table 3. Our results compare well with PHOW (Vedaldi et al. 2009)

Table 3 Accuracy (%±std.) of *one-versus-rest* scene recognition

SCENE	PHOW (Vedaldi et al. 2009)	LDA-MAP	FS-LDA (Jaakkola and Haussler 1999)	FESS-LDA (Perina et al. 2012)	SFM-LDA
Coast	90.66 ± 0.65	83.85 ± 0.92	90.42 ± 0.34	93.89 ± 0.46	**94.56 ± 0.61**
Forest	96.49 ± 0.39	94.94 ± 0.46	94.45 ± 0.46	97.92 ± 0.26	**98.15 ± 0.34**
Mountain	92.58 ± 0.64	84.99 ± 1.78	88.62 ± 0.50	93.29 ± 0.47	**93.97 ± 0.41**
Country	**91.38 ±0.71**	72.30 ± 1.74	87.40 ± 0.46	90.62 ± 0.33	90.81 ± 0.63
Highway	95.27 ± 0.49	81.50 ± 1.28	92.48 ± 0.22	94.67 ± 0.34	**96.18 ± 0.27**
InsideCity	93.96 ± 0.62	85.14 ± 1.74	90.79 ± 0.14	94.26 ± 0.65	**95.81 ± 0.37**
Street	93.89 ± 0.64	76.46 ± 1.23	93.76 ± 0.24	94.21 ± 0.42	**95.40 ± 0.45**
Building	94.40 ± 0.49	87.85 ± 0.55	92.83 ± 0.57	96.06 ± 0.51	**96.39 ± 0.44**

Bold values represent the best result on each experiment

which is a state-of-the-art feature transform for generating input vector for scene recognition. The results of semi-supervised learning are shown in Fig. 5, again demonstrating unlabeled examples can help classification particularly when there are few labeled examples.

Figure 5a compares the performance among the three methods (our SFM-LDA, FESS-LDA and FS-LDA) as a function of the number of topics used in the model in the binary classification of "highway" category against all other categories. The models are trained with 50 % of the labeled data, and tested with the rest. The results show that both SFM and FESS are better than FS in this case, and that SFM has a performance advantage over FESS when small number of topics are used (5–20), and their performance converge at 30 topics. Fig. 5b compares the benefit of using unlabeled data to train the models first, versus not using any unlabeled data at all. 25 % or 672 images of the dataset is used as unlabeled data, i.e. not using the label of the images. Training with unlabeled data yields significant benefit when the labeled data used is relatively small, i.e. up to 268. As more and more labeled data are used, the overall performance of the classifier continues to improve, but the benefit of training with unlabeled data disappears because the classifier relies more and more on the labeled data. Figure 5c demonstrates this trend from a different perspective.

6.4 Experiment III: protein classification using HMMs

An advantage of the stochastic feature mapping is that it can map structured input data of variable length into feature vector of a fixed dimensional feature space. To demonstrate this feature of our approach, we apply our proposed framework to remote homology recognition in molecular biology. The problem here is that given a test protein sequence, we assign it to one of the domain superfamilies defined in the SCOP (1.53) taxonomy tree according to the functions of proteins. The protein sequence data is obtained from the ASTRAL database. E-value threshold of 10^{-25} is applied to the database to reduce similar sequences. We use four labeled domain superfamilies, i.e. metabolism, information, intra-cellular processes and extra-cellular processes in our evaluation. The numbers of sequences are 804, 950, 695 and 992 respectively. Each protein sequence is a string composed of 22 distinct letters, and the string length varies from 20 to 994.

The hidden Markov model (HMM) (Rabiner 1989), a generative model that is useful for dealing with sequences with variable length, is used to model the distribution of pro-

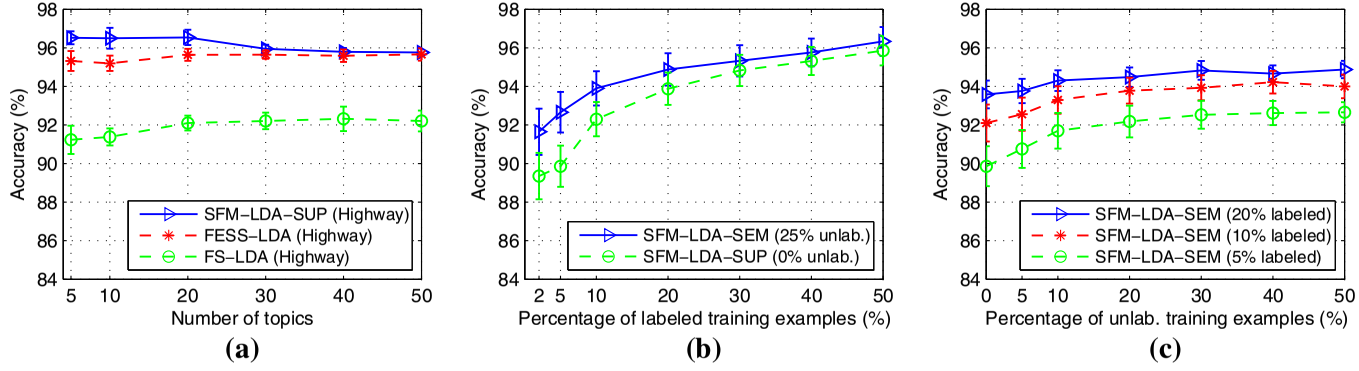

Fig. 5 Classification accuracy (%) of Highway category in the OT scene database, as a function of **a** the number of topics; **b** the number of labeled training examples, expressed in percentage of the total of 260 positive examples, and of 2428 negative examples; **c** the number of unlabeled examples, expressed in percentage of the total of 260 positive examples and 2428 negative examples. 'SUP' and 'SEM' indicate supervised learning and semi-supervised learning respectively. 'unlab' is the short of unlabeled training examples. The examples are partitioned according two rules, (1) the size of labeled training set is equivalent to the size of test set; (2) the unlabeled training examples are selected from the remaining set after the labeled training set and test set are selected. If the remaining set is not enough, we select from the labeled training set and test set. For example, a partition with 20 % of labeled training examples and 25 % of unlabeled training examples indicates that 20 % of data are test examples and 35 % of data are not used in the experiments

Table 4 Accuracy (%±std.) of *one-versus-rest* protein recognition

SUP.FAM.	2GRAM	HMM-MAP	FS-HMM (Jaakkola and Haussler 1999)	FESS-HMM (Perina et al. 2012)	SFM-HMM
# 1	78.79 ± 1.13	80.91 ± 1.53	80.03 ± 0.78	80.12 ± 0.84	$\mathbf{83.43 \pm 0.91}$
# 2	79.01 ± 0.97	80.10 ± 0.51	77.56 ± 0.64	78.96 ± 0.59	$\mathbf{84.16 \pm 0.60}$
# 3	75.19 ± 0.86	77.92 ± 0.79	73.31 ± 0.21	73.35 ± 0.41	$\mathbf{80.12 \pm 0.54}$
# 4	96.01 ± 0.33	95.10 ± 0.39	94.27 ± 0.37	$\mathbf{97.58 \pm 0.13}$	96.89 ± 0.35

Bold values represent the best result on each experiment

tein sequences. The number of output states is 22. Let $\mathbf{x}$ be the sequence with length $T_{\mathbf{x}}$; $\mathbf{x}^t$ be the binary indicator at time t, where $x_k^t = 1$ indicates that the k-th state of K possible states is selected at time t. Let $\mathbf{q}^t$ be the binary state indicator with $q_i^t = 1$ indicating the i-th state of M possible states is selected at time t. $A_{M\times M}$ denotes the transition probabilities of the approximate posterior. The feature mapping of this model is given by $\phi = (T(\mathbf{x}, \mathbf{q})^T, (\mathrm{diag}(\alpha_q(\hat{\theta}_q))T(\mathbf{q}))^T, A_q(\hat{\theta}_q))^T$, where,

$$T(\mathbf{x}, \mathbf{q}) = \left(q_1^0, \ldots, q_M^0, \sum_{t=0}^{T_{\mathbf{x}}-1} q_1^t q_1^{t+1}, \ldots, \sum_{t=0}^{T_{\mathbf{x}}-1} q_M^t q_M^{t+1}, \sum_{t=0}^{T_{\mathbf{x}}} q_1^t x_1^t, \ldots, \sum_{t=0}^{T_{\mathbf{x}}} q_M^t x_K^t \right)^T$$

$$\mathrm{diag}(\alpha_q(\hat{\theta}_q))T(\mathbf{q}) = \left(\sum\nolimits_{t=0}^{T_{\mathbf{x}}-1} q_i^t q_j^{t+1} \log A_{ij}, \ldots, \sum\nolimits_{t=0}^{T_{\mathbf{x}}-1} q_M^t q_M^{t+1} \log A_{MM} \right)^T$$

and $A_q(\hat{\theta}_q) = 0$. With the hidden states of the input sequence inferred using Baum–Welch algorithm (Baum et al. 1970), it is easy to estimate the posterior transition probabilities conditioned on $\mathbf{x}$. We can sample examples of the hidden states from the sampling distribution derived in Eq. (17) to re-estimate their posterior.

The comparative results are reported in Table 4. The number of hidden states used here is 4 for SFM-HMM and 15 for FS and FESS, which are chosen to achieve their best performing results respectively. As shown in Fig. 6a, our SFM-HMM consistently outperforms FS and FESS at any number of hidden states chosen, but the largest difference in performance gap is observed when the number of states is small. Our model can be considered more efficient as it can explain data better using fewer number of states (causes). The 2-GRAM feature is the transition probability of observed states of a sequence, i.e. $\{\frac{1}{T_c}\sum_{t=0}^{T_c-1} x_i^t x_k^{t+1}\}_{i,k}$. The difference of the performance of the first four existing methods are not significant except on superfamily #3. The results of semi-supervised learning are reported in Fig. 6, which again when there are few labeled examples in the training set, unlabeled data could help the learning of the generative models (Fig. 6b, c).

6.5 Discussions

6.5.1 Generalization bound and performance

The proposed learning approach for the stochastic feature mapping is based on the minimization of generalization bound. Even though the generalization bound is not always tight, the proposed approach shows some promising attributes. The primary reason is that its advantage comes from the exploitation of hidden variables and the feedback mechanism based on the

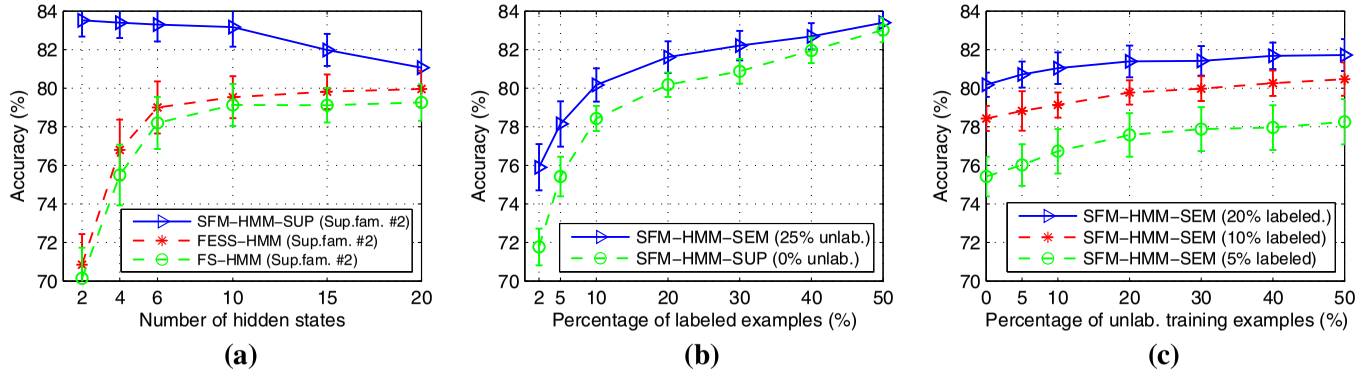

Fig. 6 Classification accuracy (%) of super-family #2 of protein sequences in the ASTRAL database, as a function of **a** the number of hidden states; **b** the number of training labeled examples, expressed in percentage of the total of 950 positive examples, and of 2491 negative examples; **c** the number of unlabeled examples, expressed in percentage of the total of 950 positive examples and 2491 negative examples. 'SUP' and 'SEM' indicate supervised learning and semi-supervised learning respectively. 'unlab' is the short of unlabeled training examples. The examples are partitioned according two rules, (1) the size of labeled training set is equivalent to the size of test set; (2) the unlabeled training examples are selected from the remaining set after the labeled training set and test set are selected. If the remaining set is not enough, we select from the labeled training set and test set. For example, a partition with 20 % of labeled training examples and 25 % of unlabeled training examples indicates that 20 % of data are test examples and 35 % of data are not used in the experiments

generalization bound, namely, tuning the generative models and feature mapping according to classification results.

6.5.2 Semi-supervised versus supervised

The above experiments also arise the comparison discussion on semi-supervised learning and supervised learning. It is worth noting that, the semi-supervised learning scheme uses the same labeled examples with the supervised learning scheme, but exploits additional unlabeled examples to train generative model and reduce the classification variance. The unlabeled examples are significantly helpful when the number of labeled examples is few, and seldom bring degeneration to the classification. Thus, in our experiments, the semi-supervised scheme usually outperforms against the supervised scheme.

7 Conclusions

This paper presents a new approach to integrate generative models and discriminative models for classification under the PAC-Bayes theoretical framework. The bridge for this integration is a stochastic feature mapping derived from the negative free energy function for exponential family models. This feature mapping is an explicit function over the hidden and observed variables, but not over the parameters of the generative models. This allows the update of the generative models to be independent of feature mapping, as if it is in a uncoupled system. This allows the SFM scheme to be easily and flexibly coupled to many types of generative models, greatly increase the flexibility of the framework. Under this framework, the generative model and the discriminative model form a close loop, with stochastic feature mapping being tuned in the feedforward path to improve the discriminative classifier, and the classification performance in the feedback path to tune the generative models. This innovation makes the classifier more flexible and adaptive, yielding state-of-the-art results in many application scenarios. Another innovation of this work is the derivation of the PAC-Bayes bound for semi-supervised learning. This allows the generative models to learn from both labeled and unlabeled data, significantly enhancing the ability of the classifier when labeled data are limited. The fact that the generative model can be optimized independent of the feature mapping allows the SFM to be coupled with a large variety of generative models, adding to the versatility of our framework.

We performed three experiments on distinct datasets from medicine, computer vision, and molecular biology and demonstrated a number of advantages offered by this framework over other existing approaches. In particular, because our method allows the fine tuning of the generative models and consequently the feature mapping function based on classification results, it is versatile and adaptive to the data. This leads to a more efficient generative model that can explain data with small capacity, as well as a more effective classifier that yields consistent state-of-the-art performance across multiple datasets. We demonstrated that when there is a limited amount of training data, this framework can capitalize on the strength of the generative models to learn from unlabeled data and tune the feature mapping to achieve better classifier performance. We further demonstrated in our applications that the SFM can be coupled to a variety of generative models, including GMM, LDA and HMM. A major remaining difficulty is the non convexity of the objective function, which can trap the solution in local minima. We have adopted a multiple initialization or seeding strategy to remedy the situation, and have achieved good results.

Nevertheless, we expect the exploitation of more robust and efficient optimization methods will likely yield better performance, and the development of incremental learning algorithm or parallel learning method could scale the approach to large dataset. Besides, coupling the proposed framework with tighter bounds is left as a future work.

Appendix

Proof of Lemma 1

Here we give the proof of the decomposition of $R_S(f_Q)$. The decomposition of the true risk $R_D(f_Q)$ can be similarly proved. Letting E_f be the abbreviation of $\mathrm{E}_{Q(f)}$, for any $y \in [-1, 1]$, we have,

$$\begin{aligned}
\mathrm{E}_{f_1,f_2}\mathrm{I}(f_1 \neq f_2) &= \mathrm{E}_{f_1,f_2}[\mathrm{I}(f_1 = y)\mathrm{I}(f_2 \neq y) + \mathrm{I}(f_1 \neq y)\mathrm{I}(f_2 = y)] \\
&= \mathrm{E}_{f_1,f_2}2[\mathrm{I}(f_1 \neq y)(1 - \mathrm{I}(f_2 \neq y))] \\
&= \mathrm{E}_{f_1,f_2}2[\mathrm{I}(f_1 \neq y) - \mathrm{I}(f_1 \neq y)\mathrm{I}(f_2 \neq y)] \\
&= 2\mathrm{E}_{f_1}[\mathrm{I}(f_1 \neq y)] - 2\mathrm{E}_{f_1,f_2}\mathrm{I}(f_1 \neq y)\mathrm{I}(f_2 \neq y)]
\end{aligned}$$

Then we have the decomposition of $\mathrm{E}_{f_1}\mathrm{I}(f_1 \neq y)$,

$$\mathrm{E}_{f_1}\mathrm{I}(f_1 \neq y) = \frac{1}{2}\mathrm{E}_{f_1,f_2}\mathrm{I}(f_1 \neq f_2) + \mathrm{E}_{f_1,f_2}\mathrm{I}(f_1 \neq y)\mathrm{I}(f_2 \neq y) \tag{21}$$

where the first term is label-independent while the second term is label-dependent. Substituting Eq. (21) into $R_S(f_Q)$ of Eq. (8) and letting f_{1i} be the abbreviation of $f_1(\mathbf{x}_i, \mathbf{h})$, then,

$$\begin{aligned}
R_S(f_Q) &= \frac{1}{m}\sum\nolimits_i \mathrm{E}_{Q(\mathbf{h})}\mathrm{E}_{f_1}\mathrm{I}(f_{1i} \neq y_i) \\
&= \frac{1}{m}\sum\nolimits_i \mathrm{E}_{Q(\mathbf{h})}\Big[\mathrm{E}_{f_1,f_2}\mathrm{I}(f_{1i} \neq y_i)\mathrm{I}(f_{2i} \neq y_i) + \frac{1}{2}\mathrm{E}_{f_1,f_2}\mathrm{I}(f_{1i} \neq f_{2i})\Big] \\
&= e_S(f_Q) + \frac{1}{2}d_S(f_Q)
\end{aligned}$$

which finishes the proof.

Proof of Theorem 1

First, if $\ell(f_Q)$ is $d(f_Q)$ (see Lemma 1), with probability at least $1-\delta$, the following inequality holds simultaneously for all posteriors Q,

$$\begin{aligned}
\mathrm{kl}(d_S(f_Q) \parallel d_D(f_Q)) &= \mathrm{kl}\left(\mathrm{E}_{Q(f_1)Q(f_2)Q(\mathbf{h})}[\hat{r}_d(f_Q)] \parallel \mathrm{E}_{Q(f_1)Q(f_2)Q(\mathbf{h})}[r_d(f_Q)]\right) \\
&\leq \mathrm{E}_{Q(f_1)Q(f_2)Q(\mathbf{h})}\left[\mathrm{kl}(\hat{r}_d(f_Q) \parallel r_d(f_Q))\right] \\
&\leq \frac{1}{m}\left(\mathrm{KL}_d(Q \parallel P) + \ln\frac{m+1}{\delta}\right)
\end{aligned}$$

where the true risk related term $r_d(f_Q) = \mathrm{E}_{(\mathbf{x},y)\sim D}\,\mathrm{I}(f_1(\mathbf{x}, \mathbf{h}) \neq f_2(\mathbf{x}, \mathbf{h}))$, the empirical risk related term $\hat{r}_d(f_Q) = \frac{1}{m}\sum_{i=1}^{m}\mathrm{I}(f_1(\mathbf{x}_i, \mathbf{h}) \neq f_2(\mathbf{x}_i, \mathbf{h}))$ and the divergence $\mathrm{KL}_d(Q \parallel P) = 2\mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \mid \mathbf{x} \parallel P(\mathbf{h} \mid \mathbf{x}))$. The first inequality is derived by applying Jensen's inequality, given the convexity of kl. The second inequality is proved in Appendix 1, holding with probability at least $1 - \delta$.

Second, for $\epsilon > 0$, following the proof in McAllester (2003) gives,

$$d_D(f_Q) \leq \sup\left\{\epsilon : \mathrm{kl}(d_S(f_Q) \,\|\, \epsilon) \leq \frac{1}{m}\left(\mathrm{KL}_d(Q \,\|\, P) + \ln\frac{m+1}{\delta}\right)\right\} \tag{22}$$

which finished the proof for $d(f_Q)$.

Third, for $e(f_Q)$ and $R(f_Q)$, using the inequalities proved in next Appendix 1 and applying the above proof, we obtain the explicit bounds for $e_D(f_Q)$ and $R_D(f_Q)$ which are summarized in Theorem 1.

Proof of inequalities for Theorem 1

First, we prove the inequality for the case that $\ell(f_Q)$ is $d(f_Q)$. The inequalities for the cases that $\ell(f_Q)$ is $R(f_Q)$ or $e(f_Q)$ can be similarly proved. Following the approach in Seeger (2002). We denote the risk related variables,

$$r_d = \mathop{\mathrm{E}}_{\mathbf{x}\sim D} \mathrm{I}(f_1(\mathbf{x}, \mathbf{h}) \neq f_2(\mathbf{x}, \mathbf{h}))$$
$$\hat{r}_d = \frac{1}{m}\sum_{i=1}^{m} \mathrm{I}(f_1(\mathbf{x}_i, \mathbf{h}) \neq f_2(\mathbf{x}_i, \mathbf{h}))$$

The variable $m\hat{r}_d$ follows the binomial distribution parameterized by (m, r_d), thus we have the following inequality,

$$\begin{aligned}\mathrm{E}_S\left[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}\right] &= \sum_{i=0}^{m}\binom{m}{i} e^{m\left[\mathrm{kl}(i/m\|r_d) + i/m\log r_d + (1-i/m)\log(1-r_d)\right]} \\ &= \sum_{i=0}^{m}\binom{m}{i} e^{-mH(i/m)} \leq m+1\end{aligned}$$

where $H(p) = -p\log p - (1-p)\log(1-p)$ is the entropy of binomial variable, and the inequality is derived by using $\exp(-mH(i/m)) \geq \binom{m}{i}$ (Seeger 2002). Noting that $f_1, f_2, \mathbf{h}$ are independently distributed, we write $P(f_1)P(f_2)P(\mathbf{h})$ as the compact form $P(f_1, f_2, \mathbf{h})$, and $Q(f_1)Q(f_2)Q(\mathbf{h})$ as $Q(f_1, f_2, \mathbf{h})$. Taking expectation over $P(f_1, f_2, \mathbf{h})$, and applying Markov's inequality, we have,

$$\Pr\left\{\mathrm{E}_{P(f_1,f_2,\mathbf{h})}\left[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}\right] > \frac{m+1}{\delta}\right\} \leq \delta$$

Therefore, for an arbitrary set of examples S, the following inequality holds with probability as least $1-\delta$,

$$\mathrm{E}_{P(f_1,f_2,\mathbf{h})}\left[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}\right] \leq \frac{m+1}{\delta} \tag{23}$$

Defining the Gibbs measure $dP_G(f_1, f_2, \mathbf{h}) = \frac{e^{m\mathrm{kl}(\hat{r}_d\|r_d)}}{\mathrm{E}_P[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}]} dP(f_1, f_2, \mathbf{h})$ (Seeger 2002), then we have the following formulas,

$$\begin{aligned}&\mathrm{KL}(Q(f_1, f_2, \mathbf{h}) \,\|\, P(f_1, f_2, \mathbf{h})) + \ln \mathrm{E}_P\left[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}\right] - \mathrm{E}_Q[m\mathrm{kl}(\hat{r}_d \,\|\, r_d)] \\ &= \int \ln\left(\frac{dQ(f_1, f_2, \mathbf{h})}{dP(f_1, f_2, \mathbf{h})} \cdot \frac{\mathrm{E}_P[e^{m\mathrm{kl}(\hat{r}_d\|r_d)}]}{e^{m\mathrm{kl}(\hat{r}_d\|r_d)}}\right) dQ(f_1, f_2, \mathbf{h}) \\ &= \mathrm{KL}(Q(f_1, f_2, \mathbf{h}) \,\|\, P_G(f_1, f_2, \mathbf{h})) \geq 0\end{aligned} \tag{24}$$

Substituting Eq. (23) into Eq. (24), we have,

$$\begin{aligned}\mathrm{E}_Q[\,\mathrm{kl}(\hat{r}_d \parallel r_d)] &\leq \frac{1}{m}\left(\mathrm{KL}(Q(f_1, f_2, \mathbf{h}) \parallel P(f_1, f_2, \mathbf{h})) + \ln \mathrm{E}_P\left[e^{m\mathrm{kl}(\hat{r}_d \parallel r_d)}\right]\right) \\ &\leq \frac{1}{m}\left(\mathrm{KL}(Q(f_1, f_2, \mathbf{h}) \parallel P(f_1, f_2, \mathbf{h})) + \ln \frac{m+1}{\delta}\right) \end{aligned} \tag{25}$$

The divergence term can be further formulated as,

$$\begin{aligned}&\mathrm{KL}(Q(f_1, f_2, \mathbf{h}) \parallel P(f_1, f_2, \mathbf{h})) \\ &= \mathrm{KL}(Q(f_1) \parallel P(f_1)) + \mathrm{KL}(Q(f_2) \parallel P(f_2)) + \mathrm{KL}(\mathrm{E}_{P(\mathbf{x})} Q(\mathbf{h}|\mathbf{x}) \parallel \mathrm{E}_{P(\mathbf{x})} P(\mathbf{h}|\mathbf{x})) \\ &\leq 2 \cdot \mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \mid \mathbf{x} \parallel P(\mathbf{h} \mid \mathbf{x})) = \mathrm{KL}_d(Q \parallel P) \end{aligned} \tag{26}$$

where the inequality is derived by applying Jensen's inequality, given the convexity of KL. Substituting Eq. (26) into Eq. (25) leads to the following inequality which holds with probability at least $1 - \delta$,

$$\mathrm{E}_Q\left[\,\mathrm{kl}(\hat{r}_d \parallel r_d)\right] \leq \frac{1}{m}\left(\mathrm{KL}_d(Q \parallel P) + \ln \frac{m+1}{\delta}\right)$$

Second, if $\ell(f_Q)$ is $e(f_Q)$, the above inequality can proved for $P(f_1, f_2, \mathbf{h})$ and $Q(f_1, f_2, \mathbf{h})$, true risk term $r_e = \mathrm{E}_{(\mathbf{x},y)\sim D}[\mathrm{I}(f_1(\mathbf{x}, \mathbf{h}) \neq y)\mathrm{I}(f_2(\mathbf{x}, \mathbf{h}) \neq y)]$, empirical term $\hat{r}_e = \frac{1}{m}\sum_{i=1}^{m}[\mathrm{I}(f_1(\mathbf{x}_i, \mathbf{h}) \neq y_i)\mathrm{I}(f_2(\mathbf{x}_i, \mathbf{h}) \neq y_i)]$. And divergence $\mathrm{KL}_e(Q \parallel P) = \mathrm{KL}_d(Q \parallel P)$. If $\ell(f_Q)$ is $R(f_Q)$, considering its definition and applying the above proof, the inequality can be proved for $P(f, \mathbf{h})$, $Q(f, \mathbf{h})$, $r_R = \mathrm{E}_{(\mathbf{x},y)\sim D}[\mathrm{I}(f(\mathbf{x}, \mathbf{h}) \neq y)]$, $\hat{r}_R = \frac{1}{m}\sum_{i=1}^{m}[\mathrm{I}(f(\mathbf{x}_i, \mathbf{h}) \neq y_i)]$,

$$\begin{aligned}&\mathrm{KL}(Q(f, \mathbf{h}) \parallel P(f, \mathbf{h})) \\ &= \mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{KL}(\mathrm{E}_{P(\mathbf{x})} Q(\mathbf{h} \mid \mathbf{x}) \parallel \mathrm{E}_{P(\mathbf{x})} P(\mathbf{h} \mid \mathbf{x})) \\ &\leq 1 \cdot \mathrm{KL}(Q(f) \parallel P(f)) + \mathrm{E}_{P(\mathbf{x})}\mathrm{KL}(Q(\mathbf{h} \mid \mathbf{x} \parallel P(\mathbf{h} \mid \mathbf{x})) = \mathrm{KL}_R(Q \parallel P) \end{aligned} \tag{27}$$

In sum, the following inequality holds with probability at least $1 - \delta$, for $\ell(f_Q)$ being either $R(f_Q)$, $d(f_Q)$ or $e(f_Q)$,

$$\mathrm{E}_Q\left[\,\mathrm{kl}(\hat{r}_\ell \parallel r_\ell)\right] \leq \frac{1}{m}\left(\mathrm{KL}_\ell(Q \parallel P) + \ln \frac{m+1}{\delta}\right) \tag{28}$$

Proof of Theorem 2

The proof follows the route in Keshet et al. (2011). For readability, we give the outline. Note that $\mathrm{kl}(p \parallel q) \geq (q-p)^2/(2q)$ when $q > p$. Thus, for $\ell_S > \ell_D$, we have the inequality $\mathrm{kl}(\ell_D \parallel \ell_S) \geq (\ell_S - \ell_D)^2/(2\ell_S)$. Theorem 1 becomes,

$$\ell_D \leq \sup\left\{\ell_D : \frac{(\ell_D - \ell_S)^2}{2\ell_D} \leq \frac{1}{m}\left(\alpha_\ell \mathrm{KL}_\ell(Q \parallel P) + \ln \frac{m+1}{\delta}\right)\right\}$$

Letting $c = \frac{1}{m}\left(\alpha_\ell \mathrm{KL}_\ell(Q \parallel P) + \ln \frac{m+1}{\delta}\right)$, applying the results in Keshet et al. (2011) gives,

$$\sup\left\{\ell_D : \frac{(\ell_D-\ell_S)^2}{2\ell_D} \le c\right\} = \sup\left\{\ell_D : \forall\lambda>0, \ell_D-\ell_S \le \frac{\ell_D}{2\lambda}+\lambda c\right\}$$
$$= \inf_{\lambda>1/2}\left(\frac{1}{1-\frac{1}{2\lambda}}\right)(\ell_S+\lambda c)$$

which finishes the proof.

Proof of Theorem 3

This theorem can be proved by applying the proof of PAC-Bound 3 in Lacasse et al. (2006). Note that, Lemma 1 has shown that $R_S(f_Q) = e_{S_l}(f_Q) + \frac{1}{2}d_S(f_Q)$. Letting $c_l = \frac{1}{m_l}\big(2\mathrm{KL}_e(Q \parallel P) + \ln\frac{m_l+1}{\delta}\big)$, $c_u = \frac{1}{m}\big(2\mathrm{KL}_d(Q \parallel P) + \ln\frac{m+1}{\delta}\big)$, and using Theorem 2, we have,

$$R_S(f_Q) = e_{S_l}(f_Q) + \frac{1}{2}d_S(f_Q)$$
$$\le \inf_{\lambda_l>1/2}\left(\frac{1}{1-\frac{1}{2\lambda_l}}\right)(e_{S_l}+\lambda_l c_l) + \frac{1}{2}\inf_{\lambda_u>1/2}\left(\frac{1}{1-\frac{1}{2\lambda_u}}\right)(d_S+\lambda_u c_u)$$

which finishes the proof.

Solution of $Q(\mathbf{h} \mid \mathbf{x}_i)$

For a labeled example $\mathbf{x}_i$, the objective function for $Q(\mathbf{h} \mid \mathbf{x}_i)$ is,

$$\min_{Q(\mathbf{h}\mid\mathbf{x}_i)} \frac{1}{m_l}\mathrm{E}_{Q(\mathbf{h}\mid\mathbf{x}_i)}\Big[\Phi(y_i\bar{\mathbf{u}}\cdot\bar{\phi}_i)^2\Big] + \frac{1}{m}\mathrm{E}_{Q(\mathbf{h}\mid\mathbf{x}_i)}\Big[\Phi(\bar{\mathbf{u}}\cdot\bar{\phi}_i)\Phi(-\bar{\mathbf{u}}\cdot\bar{\phi}_i)\Big]$$
$$+\frac{1}{\tilde{m}_\lambda m}\mathrm{E}_{Q(\mathbf{h}\mid\mathbf{x}_i)}\Big[\log Q(\mathbf{h}\mid\mathbf{x}_i) - \log P(\mathbf{x}_i,\mathbf{h}\mid\theta)\Big]$$
$$s.t. \int Q(\mathbf{h}\mid\mathbf{x}_i)\,d\,\mathbf{h} = 1$$

Using the method of Lagrange multipliers, we have,

$$Q(\mathbf{h}\mid\mathbf{x}_i) \propto P(\mathbf{x}_i,\mathbf{h})\exp\left\{-\frac{\tilde{m}_\lambda m}{m_l}\Phi\left(y\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)^2 - \tilde{m}_\lambda\Phi\left(\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\Phi\left(-\bar{\mathbf{u}}\cdot\bar{\phi}_i\right)\right\}$$

The solution of $Q(\mathbf{h} \mid \mathbf{x}_i)$ for unlabeled examples can be similarly derived.

References

Baum, L., Petrie, T., Soules, G., & Weiss, N. (1970). A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. *The Annals of Mathematical Statistics*, *41*(1), 164–171.

Blei, D., Ng, A., & Jordan, M. (2003). Latent Dirichlet allocation. *Journal of Machine Learning Research*, *3*, 993–1022.

Chang, C.-C., & Lin, C.-J. (2011). Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, *2*(3), 27.

Chatfield, K., Lempitsky, V., Vedaldi, A., & Zisserman, A. (2011). The devil is in the details: an evaluation of recent feature encoding methods. In *British machine vision conference* (pp. 76.1–76.12).

Germain, P., Lacasse, A., Laviolette, F., & Marchand, M. (2009). PAC-Bayesian learning of linear classifiers. In *International conference on machine learning* (pp. 353–360).

Gönen, M., & Alpaydin, E. (2008). Localized multiple kernel learning. In *International conference on machine learning* (pp. 352–359).
Graça, J., Ganchev, K., & Taskar, B. (2007). Expectation maximization and posterior constraints. *Advances in Neural Information Processing Systems, 20*, 569–576.
Griffiths, T. L., & Steyvers, M. (2004). Finding scientific topics. *Proceedings of the National academy of Sciences of the United States of America*, *101*(Suppl 1), 5228–5235.
Holub, A., Welling, M., & Perona, P. (2008). Hybrid generative-discriminative visual categorization. *International Journal of Computer Vision*, *77*(1), 239–258.
Jaakkola, T., & Haussler, D. (1999). Exploiting generative models in discriminative classifiers. *Advances in Neural Information Processing Systems, 11*, 487–493.
Jaakkola, T., Meila, M., & Jebara, T. (1999). Maximum entropy discrimination. *MIT Technical Report AITR-1668*
Joachims, T. (1999). Transductive inference for text classification using support vector machines. In *International conference on machine learning* (pp. 200–209). Slowenien: Bled.
Joachims, T. (2003) Transductive learning via spectral graph partitioning. In *International conference on machine learning* (pp. 290–297).
Jordan, M., Ghahramani, Z., Jaakkola, Tommi S., & Saul, Lawrence K. (1999). Introduction to variational methods for graphical models. *Machine Learning*, *37*, 183–233.
Keshet, J., McAllester, D., & Hazan, T. (2011). Pac-bayesian approach for minimization of phoneme error rate In *IEEE conference on acoustics, speech and signal processing* (pp. 2224–2227).
Lacasse, A., Laviolette, F., Marchand, M., Germain, P., & Usunier, N. (2006). Pac-bayes bounds for the risk of the majority vote and the variance of the gibbs classifier. *Advances in Neural Information Processing Systems, 19*, 769–776.
Langford, J. (2006). Tutorial on practical prediction theory for classification. *Journal of Machine Learning Research*, *6*(1), 273.
Li, X., Lee, T.S., & Liu, Y. (2011). Hybrid generative-discriminative classification using posterior divergence. In *IEEE conference on computer vision and pattern recognition* (pp. 2713–2720).
Li, X., Wang, B., Liu, Y., & Lee, T.S. (2013). Learning discriminative sufficient statistics score space for classification. In *European conference on machine learning* (pp. 49–64).
Li, X., Zhao, X., Fu, Y., & Liu, Y. (2010). Bimodal gender recognition from face and fingerprint. In *IEEE conference on computer vision and pattern recognition* (pp. 2590–2597).
Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, *60*(2), 91–110.
McAllester, D. (1999). Some pac-bayesian theorems. *Machine Learning*, *37*(3), 355–363.
McAllester, D. (2003). Simplified PAC-Bayesian margin bounds. *Learning theory and Kernel machines* (pp. 203–215). Newyork: Springer.
McCallum, A., Pal, c, Druck, G., & Wang, X. (2006). Multi-conditional learning: Generative/discriminative training for clustering and classification. *National Conference on Artificial Intelligence*, *21*(1), 433.
Ng, A.Y., & Jordan, M.I. (2002). On discriminative vs. generative classifiers: A comparison of logistic regression and naive bayes. *Advances in Neural Information Processing Systems, 14*, 841–848.
Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, *42*(3), 145–175.
Perina, A., Cristani, M., Castellani, U., Murino, V., & Jojic, N. (2012). Free energy score spaces: Using generative information in discriminative classifiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *34*(7), 1249–1262.
Rabiner, L. (1989). A tutorial on hidden Markov models and selected applications inspeech recognition. *Proceedings of the IEEE*, *77*(2), 257–286.
Raina, R., Shen, Y., Ng, A., & McCallum, A. (2003) Classification with hybrid generative/discriminative models. *Advances in Neural Information Processing Systems, 16*, 545–552.
Seeger, M. (2002). Pac-bayesian generalisation error bounds for gaussian process classification. *The Journal of Machine Learning Research*, *3*, 233–269.
Seldin, Y., Cesa-Bianchi, N., Auer, P., Laviolette, F., &Shawe-Taylor, J. (2012). Pac-bayes-bernstein inequality for martingales and its application to multiarmed bandits. In *JMLR: workshop and conference proceedings* (no. 26, pp. 98–111).
Smith, N., & Gales, M. (2002). Speech recognition using svms. *Advances in Neural Information Processing Systems*, *14*, 1197–1204.
Tolstikhin, I., & Seldin, Y. (2013). Pac-Bayes-empirical-bernstein inequality. *Advances in Neural Information Processing Systems, 26*, 109–117.
Tsuda, K., Kawanabe, M., Ratsch, G., Sonnenburg, S., & Muller, K. (2002). A new discriminative kernel from probabilistic models. *Neural Computation*, *14*(10), 2397–2414.

Vapnik, V. (2000). *The nature of statistical learning theory*. Berlin: Springer.
Vedaldi, A., Gulshan, V., Varma, M., & Zisserman, A. (2009). Multiple kernels for object detection. In *IEEE international conference on computer vision* (pp. 606–613).
Yu, C.-N.J., & Joachims, T. (2009). Learning structural svms with latent variables. In *International conference on machine learning* (pp. 1169–1176).