Supervised Restricted Boltzmann Machines

Tu Dinh Nguyen, Dinh Phung, Viet Huynh, Trung Le
Center for Pattern Recognition and Data Analytics,
Deakin University, Australia.
{tu.nguyen, dinh.phung, viet.huynh, trung.l}@deakin.edu.au.

Abstract

We propose in this paper the *supervised restricted Boltzmann machine* (sRBM), a unified framework which combines the versatility of RBM to simultaneously learn the data representation and to perform supervised learning (i.e., a nonlinear classifier or a nonlinear regressor). Unlike the current state-of-the-art classification formulation proposed for RBM in (Larochelle *et al.*, 2012), our model is a *hybrid* probabilistic graphical model consisting of a distinguished *generative* component for data representation and a *discriminative* component for prediction. While the work of (Larochelle *et al.*, 2012) typically incurs no extra difficulty in inference compared with a standard RBM, our discriminative component, modeled as a directed graphical model, renders MCMC-based inference (e.g., Gibbs sampler) very slow and unpractical for use. To this end, we further develop scalable variational inference for the proposed sRBM for both classification and regression cases. Extensive experiments on real-world datasets show that our sRBM achieves better predictive performance than baseline methods. At the same time, our proposed framework yields learned representations which are more discriminative, hence interpretable, than those of its counterparts. Besides, our method is probabilistic and capable of generating meaningful data conditioning on specific classes – a topic which is of current great interest in deep learning aiming at data generation.

1 INTRODUCTION

Restricted Boltzmann machine (RBM) is an important generative model that is capable of learning representations from data. It has been successfully applied to diverse data types including: images (Hinton and Salakhutdinov, 2006), mixed low-level features of images (Nguyen *et al.*, 2013c), text (Salakhutdinov and Hinton, 2009a) and medical data (Nguyen *et al.*, 2013a). In these tasks, the RBMs can serve as either fast feature extractors or building blocks to provide a good parameter initialization for deep architectures (Hinton and Salakhutdinov, 2006; Salakhutdinov and Hinton, 2009b). In both cases, one often uses a two-stage pipeline framework that has the RBMs followed by another supervised learning method (a classifier or a regressor). The first stage is to train the RBMs using an unsupervised learning algorithm with efficient MCMC-based learning procedures (Hinton, 2002). Then the supervised learning algorithm is built upon the representations or parameters learned by RBMs to further carry out a discriminative training in the second stage.

These two-stage approaches, however, suffer from two key drawbacks. First, they require to tune hyperparameters for both models (RBM and the supervised model) which is computationally expensive and time consuming. Second, representations learned by RBM may effectively support generating the data, but there is no guarantee that they would be useful for further fitting a supervised model, as they have not seen any label during the unsupervised training. In other words, the models miss the opportunity to use labeled data to 'regularize' the learned representations. Thus separating the feature learning phase from the discriminative training could be suboptimal.

Our solution is to construct a supervised restricted Boltzmann machine (sRBM) – a unified framework that leverages the successful architecture of RBM to simultaneously learn the data representation as a feature extractor and to perform prediction as a nonlinear classifier or a nonlinear regressor. The sRBM uses a hidden layer to associate the outcome with input data in order to model their joint distribution. The resulting *hybrid* probabilistic graphical model consists of two components: an RBM with undirected connections of hidden and visible units, and a predictive model with directed connections from hidden units to outcome variable. The RBM component has generative capability of representing data whilst the discriminative part enables

the model to perform prediction by itself. Combining generative and discriminative models allows one to leverage the strength from each other. More precisely, the generative component plays a role as a regularization for discriminative part, at the same time the discriminative part utilizes label statistical information to drive the generative one to learn more expressive representations. Besides, the combined framework facilitates the model selection since no additional hyperparameters from the separate prediction module must be tuned. However, the introduction of the discriminative component into the model makes inferences become much more challenging where standard Gibbs sampler could be very slow for practical use. To this end, we further develop variational inference for the proposed sRBM for both classification and regression.

In addition to predictive capacity, the sRBM offers a principled way to generate data based on the specific labels or classes, and to disentangle the data and labels via the learned latent representations and embedding connection matrix – a capability that is desirable in many cases, especially in natural language processing (Mikolov *et al.*, 2013). As a generative model, once our model has been learned, we can select and fix the labels and then alternatively sample hidden and visible variables using MCMC to obtain the data corresponding to the labels. For disentangling capability, recall that, in the standard RBM, each unit in the hidden layer can act as a feature detector, and together, all the hidden units form a distributed, and discrete representation of data (Nguyen *et al.*, 2013b). In sRBM, they are also linked to the labels via directed connections parameterized by a weight matrix. Due to the *explaining away* effect (Coates and Ng, 2011; Bengio *et al.*, 2013), these hidden units must compete with each other to explain the labels. Thus, the learned representations and weight matrix associating latent factors to the labels are driven to discriminate the labels.

We quantitatively and qualitatively demonstrate the capacity of our proposed model through comprehensive experiments on three tasks – classification, regression and data generation using three real-world datasets: handwritten digits, newsgroup documents, and CT scan images. Our primary target is to verify the predictive and generative capabilities of sRBM, and the effectiveness of learned representations and the embedding discovered by the model. The experimental results show that our method achieves better predictive performance than the baselines. At the same time, its learned representations and embedding matrices are more discriminative than those of the ClassRBM and standard RBM, and the data generated by the proposed model are meaningful and appropriate to respective classes.

In short, our contributions are: (i) a novel unified RBM-based model that can acts as a complete supervised model, thus there is no need to tune additional hyperparameters for the separate predictor; (ii) the derivation of variational inference for the proposed model for both classification and regression cases; and (ii) a comprehensive evaluation of the effectiveness of our method on three learning tasks of three applications: image recognition, text classification and location prediction for CT images.

2 RELATED WORK

We first describe the restricted Boltzmann machine (RBM) for unsupervised learning representation. An RBM is a bipartite undirected graphical model in which the bottom layer contains observed variables called visible units and the top layer consists of latent *representational variables*, known as hidden units (Freund and Haussler, 1994). Two layers are fully connected but there is no connection within layers. The hidden units can capture the latent factors not presented in the observations. As a matter of convention in the literature of RBM, we shall use the term "unit" and "random variable" interchangeably.

More formally, assume a binary RBM with N visible units and K hidden units, let $\mathbf{v}$ denote the set of visible variables: $\mathbf{v} = [\mathrm{v}_1, \mathrm{v}_2, ..., \mathrm{v}_\mathrm{N}]^\top \in \{0,1\}^\mathrm{N}$ and $\mathbf{h}$ indicate the set of hidden ones: $\mathbf{h} = [\mathrm{h}_1, \mathrm{h}_2, ..., \mathrm{h}_\mathrm{K}]^\top \in \{0,1\}^\mathrm{K}$. The RBM assigns an energy function for a joint configuration over the state $(\mathbf{v}, \mathbf{h})$ as:

$$E(\mathbf{v}, \mathbf{h}; \psi) = -\left(\mathbf{a}^\top \mathbf{v} + \mathbf{b}^\top \mathbf{h} + \mathbf{v}^\top \mathbf{W} \mathbf{h}\right) \quad (1)$$

where $\psi = \{\mathbf{a}, \mathbf{b}, \mathbf{W}\}$ is the set of parameters. $\mathbf{a} = [\mathrm{a}_n]_\mathrm{N} \in \mathbb{R}^\mathrm{N}$, $\mathbf{b} = [\mathrm{b}_k]_\mathrm{K} \in \mathbb{R}^\mathrm{K}$ are the biases of hidden and visible units respectively; and $\mathbf{W} = [\mathrm{w}_{nk}]_{\mathrm{N}\times\mathrm{K}} \in \mathbb{R}^{\mathrm{N}\times\mathrm{K}}$ represents the weights connecting the hidden and visible units. The model admits a Boltzmann distribution (also known as Gibbs distribution) as follows:

$$p(\mathbf{v}, \mathbf{h}; \psi) = \exp\{-E(\mathbf{v}, \mathbf{h}; \psi) - A(\psi)\} \quad (2)$$

where $A(\psi)$ is the log-partition function. Since the network has no intra-layer connections, units in one layer become conditionally independent given the other layer. Thus the conditional distributions over visible and hidden units are factorized as:

$$p(\mathbf{h} \mid \mathbf{v}; \psi) = \prod_{k=1}^{\mathrm{K}} p(\mathrm{h}_k \mid \mathbf{v}) \quad (3)$$

$$p(\mathbf{v} \mid \mathbf{h}; \psi) = \prod_{n=1}^{\mathrm{N}} p(\mathrm{v}_n \mid \mathbf{h}) \quad (4)$$

There have been recent approaches that attempt to incorporate label information into the standard RBM (McCallum *et al.*, 2006; Schmah *et al.*, 2009; Li *et al.*, 2015). The main differences from our method are: (i) such methods still require separate classifiers, and (ii) without sharing parameters, they fail to directly capture the latent similarity between classes. Our idea is to focus on the self-contained framework for prediction, which does not need to rely on

an additional supervised algorithm. The model introduced in (Yang *et al.*, 2007) and the current state-of-the-art classification formulation proposed for RBM (ClassRBM) in (Larochelle *et al.*, 2012) are closely related to ours, that couple the label to input features of RBMs. These models, however, only support classification whilst our model can perform both classification and regression. Moreover, in these models, the label is considered an additionally observed variable that links to hidden units using undirected connections. On the other hand, our proposed model uses directed connections to construct a discriminative modeling of the label given hidden layer (cf. Fig. 1b). This structure enhances the discriminative latent representations and predictive performance, allowing for better prediction results.

More specifically, our proposed model differs substantially from the ClassRBM from three crucial points: model representation, inference scheme and model expressiveness. For model representation and inference, although one might construct the *moral* graph of our hybrid model, resulting in a similar undirected graphical model (as shown in Fig. 1c), they are technically different since one cannot convert an undirected form of the ClassRBM to our model. As a consequence, the ClassRBM can still be viewed as a standard RBM with some nodes being designated as label variables, hence can still be learned with standard techniques for RBM. Whereas, our directed link has resulted in a technical challenge during the inference process, hence our contribution in the variational inference techniques.

Furthermore, from the conceptual point of view, after being moralized, in both our sRBM and ClassRBM, the data features and label interact indirectly through the hidden layer, thus this layer would play an important role in capturing the relationship between features and label. The difference is that our proposed model now contains the connections among hidden units, which enables hidden units to capture more complex structures than those of the ClassRBM where these connections are not modeled. This view has indeed been considered in semi-restricted Boltzmann machines (Salakhutdinov, 2009), a wider class of RBMs.

In terms of model expressiveness, our sRBM implicitly models the correlations among hidden units, thus it provides flexible capacity to capture a wider class of distributions as discussed in (Salakhutdinov, 2009). In particular, it can model *nonlinear* interactions between label and hidden units (cf. Eq. (7)), whilst all interactions in the ClassRBM are *linear*. Another advantage is that our model, when using the first-order Taylor series, offers more freedom to choose the conditional distribution $p\left(\mathrm{y} \mid \mathbf{h}\right)$ due to appealing approximation in Eq. (11). This is a promising feature and opens room for further extensions of our approach to cover broader ranges of applications such as multi-task and multi-modal learning (Srivastava and Salakhutdinov, 2012), as well as incorporating different discriminative architectures such as Lasso (Tibshirani, 1996).

In the deep learning literature, the RBMs can be interpreted as stochastic neural networks, pretrained and then stacked layer-by-layer as a building block for deep architectures such as deep belief nets (DBNs) (Hinton and Salakhutdinov, 2006) and deep Boltzmann machines (DBMs) (Salakhutdinov and Hinton, 2009b). These deep models, however, are either trained not completely as generative models (e.g., DBNs are often fine-tuned as a standard feedforward neural nets after pretraining), and known to be extremely difficult to train (e.g., training DBMs requires a very careful weights initialization procedure, i.e., layer-by-layer pretraining followed by weights halving trick). By contrast, our sRBM still enjoys its versatility on efficient learning and inference of standard RBMs where the effective learning algorithm can be used.

Lastly, while we do not claim our main contribution in data generation, we would like to point out its connection to recent important and active research direction in constructing data generators epitomized by two current state-of-the-art density-based approaches, namely the variational autoencoder (VAE) (Kingma and Welling, 2014) and generative adversarial nets (GAN) (Goodfellow *et al.*, 2014). While these methods are still unsupervised, our sRBM provides a conditional probabilistic generator where data can be generated based on specific labels or classes.

3 PROPOSED SUPERVISED RBMs

We now present our main contribution – the supervised RBM (sRBM) that adds to the standard RBM an outcome variable y associated with each data point. This variable is generated from the hidden units via directed connections. The model now consists of two components: an RBM with undirected connections of joint distribution $p\left(\mathbf{v}, \mathbf{h}\right)$ that has generative capability of representing data, and a predictive model with directed connections of conditional distribution $p\left(\mathrm{y} \mid \mathbf{h}\right)$ that allows the model to perform prediction on its own right. These components form a *hybrid* probabilistic graphical model of sRBM as illustrated in Fig. 1b.

3.1 MODEL REPRESENTATION

More formally, the target variable represents the target label $\mathrm{y} \in \{1, 2, ...\mathrm{C}\}$ in classification problems or the response value $\mathrm{y} \in \mathbb{R}$ in regression tasks. Without loss of generality, the outcome follows a conditional distribution of exponential family that has the probability density as below:

$$p\left(\mathrm{y} \mid \mathbf{h}; \boldsymbol{\theta}\right) = t\left(\mathrm{y}\right) \exp\left\{\boldsymbol{\theta}^{\top} \boldsymbol{\phi}\left(\mathrm{y}, \mathbf{h}\right) - B\left(\boldsymbol{\theta}, \mathbf{h}\right)\right\} \quad (5)$$

wherein $\boldsymbol{\theta}$ denotes the natural or canonical parameters; $\boldsymbol{\phi}\left(\mathrm{y}, \mathbf{h}\right)$ refers to sufficient statistics; and $B\left(\boldsymbol{\theta}, \mathbf{h}\right)$ is the log-partition or cumulant function. The function $t\left(\mathrm{y}\right)$ is independent of the parameter $\boldsymbol{\theta}$. The function $B\left(\boldsymbol{\theta}, \mathbf{h}\right)$ en-

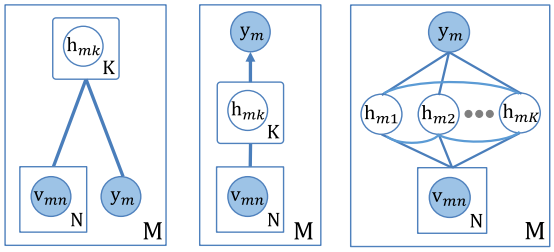

Figure 1: Graphical illustrations of (a) ClassRBM, (b) sRBM, and (c) the moralization form of sRBM. The shaded nodes represents observed variables.

sures $p(\mathrm{y} \mid \mathbf{h}; \boldsymbol{\theta})$ is a proper density, thus:

$$B(\boldsymbol{\theta}, \mathbf{h}) = \log \int_{\mathrm{y}} t(\mathrm{y}) \exp\left[\boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathrm{y}, \mathbf{h})\right] \mathrm{dy} \tag{6}$$

Our aim is to jointly model the data and the outcome in order to learn alternative data representations that simultaneously explain the data and predict the outcome for future unlabeled data. Multiplicatively combining two probability densities in Eq. (5) and Eq. (2), we obtain the joint distribution of sRBM that is also an exponential family distribution: $p(\mathbf{v}, \mathbf{h}, \mathrm{y}; \psi, \boldsymbol{\theta}) = p(\mathrm{y} \mid \mathbf{h}; \boldsymbol{\theta})\, p(\mathbf{v}, \mathbf{h}; \psi) =$

$$e^{\log t(\mathrm{y}) + \boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathrm{y}, \mathbf{h}) - B(\boldsymbol{\theta}, \mathbf{h}) - E(\mathbf{v}, \mathbf{h}; \psi) - A(\psi)} \tag{7}$$

3.2 INFERENCE

As in any other graphical models, inference is a key task in sRBM. Suppose that the model parameters have been fully specified, there are various inference tasks to be performed in a sRBM. What we present next are the most typical ones: expectation over hidden posterior and prediction.

3.2.1 Expectation over Hidden Posterior

Our aim is to compute the expectation over the hidden posterior: $\mathbb{E}_{p(\mathbf{h} \mid \mathbf{v}, \mathrm{y})}$, which requires a sum over an exponential space of hidden units. Due to the *explaining away* effect, the hidden units become conditionally dependent given the response y. Therefore the conditional distributions over hidden units are no longer factorized as in Eq. (3) of the standard RBM, resulting in an intractable inference. To overcome this shortcoming, we must resort to approximation methods. In what follows we propose two approximation approaches: Gibbs sampling and variational inference.

Gibbs sampling. Let $\mathbf{h}_{\neg k}$ denote the state of all hidden units except the k-th one. The conditional distribution of a single hidden unit is:

$$p(\mathrm{h}_k \mid \mathbf{h}_{\neg k}, \mathbf{v}, \mathrm{y}) \propto p(\mathrm{h}_k, \mathbf{h}_{\neg k} \mid \mathbf{v}, \mathrm{y}) \propto p(\mathrm{y} \mid \mathbf{h})\, p(\mathrm{h}_k \mid \mathbf{v})$$

Thus sampling from the posterior distribution of h_k can be performed using:

$$p(\mathrm{h}_k = 1 \mid \mathbf{h}_{\neg k}, \mathbf{v}, \mathrm{y}) \propto p(\mathrm{y} \mid \mathrm{h}_k = 1, \mathbf{h}_{\neg k})\, p(\mathrm{h}_k = 1 \mid \mathbf{v})$$
$$p(\mathrm{h}_k = 0 \mid \mathbf{h}_{\neg k}, \mathbf{v}, \mathrm{y}) \propto p(\mathrm{y} \mid \mathrm{h}_k = 0, \mathbf{h}_{\neg k})\, p(\mathrm{h}_k = 0 \mid \mathbf{v})$$

in which the state of a hidden unit being active or inactive given the visible units is:

$$p(\mathrm{h}_k = 1 \mid \mathbf{v}) = \mathrm{sig}\left(\mathrm{b}_k + \mathbf{v}^\top \mathbf{w}_{\bullet k}\right) \tag{8}$$
$$p(\mathrm{h}_k = 0 \mid \mathbf{v}) = \mathrm{sig}\left(-\mathrm{b}_k - \mathbf{v}^\top \mathbf{w}_{\bullet k}\right) \tag{9}$$

where $\mathrm{sig}(x) = 1/1+e^{-x}$ is logistic sigmoid function. We refer to supplementary material for full derivations.

Variational inference. When performing inference in large models or on moderately-sized datasets, the Gibbs sampler can become extremely slow as it must sequentially iterate over every single hidden unit. We hereby choose a faster method – variational inference.

In variational methods (Jordan *et al.*, 1998), the true posterior distribution $p(\mathbf{h} \mid \mathbf{v}, \mathrm{y})$ is approximated by a variational distribution $q(\mathbf{h}; \boldsymbol{\mu})$ with $\boldsymbol{\mu} = [\mu_1, \mu_2, ..., \mu_{\mathrm{K}}]^\top$ is the vector of variational parameters. These parameters are learned in order to maximize the following evidence lower bound (ELBO):

$$\begin{aligned}\mathcal{L}(\psi, \boldsymbol{\theta}, \boldsymbol{\mu}) &= \mathbb{E}_{q(\mathbf{h})}\left[\log p(\mathbf{v}, \mathbf{h}, \mathrm{y})\right] - \mathbb{E}_{q(\mathbf{h})}\left[\log q(\mathbf{h})\right] \\ &= \mathbb{E}_{q(\mathbf{h})}\left[\boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathrm{y}, \mathbf{h}) - B(\boldsymbol{\theta}, \mathbf{h}) - E(\mathbf{v}, \mathbf{h}; \psi)\right] \\ &\quad + \log t(\mathrm{y}) - \mathbb{E}_{q(\mathbf{h})}\left[\log q(\mathbf{h})\right] - A(\psi)\end{aligned}$$

Using a naive mean-field approximation, we choose a variational distribution that is fully factorized into K Bernoulli distributions as: $q(\mathbf{h}; \boldsymbol{\mu}) = \prod_{k=1}^{\mathrm{K}} q(\mathrm{h}_k; \mu_k)$ in which μ_k denotes the probability $q(\mathrm{h}_k = 1)$. The ELBO now reads (c.f. supplementary material for the full derivation): $\mathcal{L}(\psi, \boldsymbol{\theta}, \boldsymbol{\mu}) =$

$$\begin{aligned}&\sum\nolimits_{k=1}^{\mathrm{K}} \theta_k \left[\mu_k \phi(\mathrm{y}, \mathrm{h}_k = 1) + (1 - \mu_k)\, \phi(\mathrm{y}, \mathrm{h}_k = 0)\right] \\ &- \mathbb{E}_{q(\mathbf{h})}\left[B(\boldsymbol{\theta}, \mathbf{h})\right] + \sum\nolimits_{k=1}^{\mathrm{K}} \mu_k \left(\mathrm{b}_k + \sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk}\right) \\ &+ \mathrm{K}\left(\sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{a}_n \mathrm{v}_n\right) + \log t(\mathrm{y}) - A(\psi) \\ &- \sum\nolimits_{k=1}^{\mathrm{K}} \left[\mu_k \log \mu_k + (1 - \mu_k) \log(1 - \mu_k)\right]\end{aligned} \tag{10}$$

As the model parameters $\psi, \boldsymbol{\theta}$ are fixed, three terms $\mathrm{K}\left(\sum_{n=1}^{\mathrm{N}} \mathrm{a}_n \mathrm{v}_n\right)$, $\log t(\mathrm{y})$ and $A(\psi)$ are constant. Thus we can ignore them in this maximization process. Only the log-partition function $B(\boldsymbol{\theta}, \mathbf{h})$ cannot be decomposed into individual functions of each hidden unit, rendering its expectation intractable. We approximate this expectation using two strategies: the *first-order* and the *second-order*

Taylor series approximations. The first-order approximation evaluated at the first moment $\boldsymbol{\mu} = \mathbb{E}_{q(\mathbf{h})}[\mathbf{h}]$ reads:

$$\mathbb{E}_{q(\mathbf{h})}[B(\boldsymbol{\theta}, \mathbf{h})] \approx \mathbb{E}_{q(\mathbf{h})}\left[B(\boldsymbol{\theta}, \boldsymbol{\mu}) + (\mathbf{h} - \boldsymbol{\mu})^\top \nabla_{\mathbf{h}} B(\boldsymbol{\theta}, \mathbf{h})\right]$$
$$= B(\boldsymbol{\theta}, \boldsymbol{\mu}) + \mathbb{E}_{q(\mathbf{h})}[\mathbf{h} - \boldsymbol{\mu}]^\top \nabla_{\mathbf{h}} B(\boldsymbol{\theta}, \mathbf{h}) = B(\boldsymbol{\theta}, \boldsymbol{\mu}) \quad (11)$$

wherein $\nabla_{\mathbf{h}} B(\boldsymbol{\theta}, \mathbf{h})$ denotes the derivative w.r.t $\mathbf{h}$ and then evaluated at $\mathbf{h} = \boldsymbol{\mu}$, thus it is independent with $q(\mathbf{h})$ and can be taken out from the expectation in the second step. Note that we have used $\mathbb{E}_{q(\mathbf{h})}[\mathbf{h} - \boldsymbol{\mu}]^\top = 0$ in the last step. The result offers an interesting property that the expectation can be approximated by the function itself, evaluated at the mean. This allows us more freedom to choose the conditional distribution $p(\mathrm{y} \mid \mathbf{h})$.

Assuming that $B(\boldsymbol{\theta}, \mathbf{h})$ is a twice differentiable function of $\mathbf{h}$, the second-order approximation evaluated at $\boldsymbol{\mu}$ is:

$$\mathbb{E}_{q(\mathbf{h})}[B(\boldsymbol{\theta}, \mathbf{h})]$$
$$\approx B(\boldsymbol{\theta}, \boldsymbol{\mu}) + \mathbb{E}_{q(\mathbf{h})}\left[\frac{1}{2}(\mathbf{h} - \boldsymbol{\mu})^\top \mathbf{H}[B(\boldsymbol{\theta}, \boldsymbol{\mu})](\mathbf{h} - \boldsymbol{\mu})\right]$$
$$\overset{(a)}{=} B(\boldsymbol{\theta}, \boldsymbol{\mu}) + \frac{1}{2}\sum_{i=1}^{\mathrm{K}}\sum_{j=1}^{\mathrm{K}} \mathrm{H}_{ij}\mathbb{E}_{q(\mathbf{h})}[(\mathrm{h}_i - \mu_i)(\mathrm{h}_j - \mu_j)]$$
$$= B(\boldsymbol{\theta}, \boldsymbol{\mu}) + \frac{1}{2}\sum_{i=1}^{\mathrm{K}} \mathrm{H}_{ii}\mu_i(1 - \mu_i) \quad (12)$$

where $\mathrm{H}_{ij} \triangleq \partial_{\mathrm{h}_i}\partial_{\mathrm{h}_j} B(\boldsymbol{\theta}, \boldsymbol{\mu})$ denotes the second-order derivative of $B(\boldsymbol{\theta}, \mathbf{h})$ evaluated at $\mathbf{h} = \boldsymbol{\mu}$, thus H_{ij} is constant w.r.t $\mathbf{h}$ and can be taken out from the expectation as in step (a). We refer to supplementary material for full derivations.

Substituting Eq. (11) into the ELBO in Eq. (10), and then taking the derivative with respect to (w.r.t) variational parameter, we obtain: $\nabla_{\mu_k}\mathcal{L} =$

$$\theta_k[\phi(\mathrm{y}, \mathrm{h}_k = 1) - \phi(\mathrm{y}, \mathrm{h}_k = 0)] + \mathrm{b}_k + \sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk}$$
$$- \nabla_{\mu_k}\mathbb{E}_{q(\mathbf{h})}[B(\boldsymbol{\theta}, \mathbf{h})] - [\log\mu_k - \log(1 - \mu_k)] \quad (13)$$

Since there is no closed-form solution to compute μ_k by setting the gradient to zero, we update the variational parameters by iterating at the following fixed point:

$$\mu_k \leftarrow \mathrm{sig}(\theta_k[\phi(\mathrm{y}, \mathrm{h}_k = 1) - \phi(\mathrm{y}, \mathrm{h}_k = 0)]$$
$$+ \mathrm{b}_k + \sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk} - \nabla_{\mu_k}\mathbb{E}_{q(\mathbf{h})}[B(\boldsymbol{\theta}, \mathbf{h})]) \quad (14)$$

3.2.2 Prediction

The next goal is to predict the outcome given the input data: $p(\mathrm{y} \mid \mathbf{v})$. This conditional probability density can be derived as follows:

$$p(\mathrm{y} \mid \mathbf{v}) = \sum\nolimits_{\mathbf{h}} p(\mathrm{y} \mid \mathbf{h})\, p(\mathbf{h} \mid \mathbf{v}) = \mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[p(\mathrm{y} \mid \mathbf{h})]$$

As can be seen from the graphical model of sRBM, the hidden units are conditionally independent given observed visible units and unobserved outcome variables. Thus the model realizes the hidden factorization (see Eq. (3)) as in the standard RBM. Furthermore, according to Eq. (8) and Eq. (9), the probability of being active of each hidden unit also follows a Bernoulli distribution. Hence, the distribution $p(\mathbf{h} \mid \mathbf{v})$ plays the same role as the one of variational distribution $q(\mathbf{h})$ in Section 3.2.1. This allows us to use the similar approximation approach as the first-order approximation in Eq. (11), that is:

$$p(\mathrm{y} \mid \mathbf{v}) = \mathbb{E}_{p(\mathbf{h}|\mathbf{v})}[p(\mathrm{y} \mid \mathbf{h})] \approx p(\mathrm{y} \mid \boldsymbol{\mu})$$

in which μ_k is the mean of distribution $q(\mathrm{h}_k; \mu_k) \approx p(\mathrm{h}_k \mid \mathbf{v})$.

Interestingly, the predictive inference is reminiscent of the forward pass of an ordinary feedforward neural network. In particular, it could be implemented by a single layer neural network with sigmoid hidden units and softmax (for classification) or identity (for regression) output neurons. This also suggests an approach to pretrain deep models that contain the sRBM as the top layer of a building block of RBMs.

3.3 PARAMETER ESTIMATION

In this section we present how to estimate the parameters of sRBM from training data. Following the learning in the standard RBM, the sRBM also aims to maximize the log-likelihood of data: $\log p(\mathbf{v}, \mathrm{y}) = \log\sum_{\mathbf{h}} p(\mathbf{v}, \mathbf{h}, \mathrm{y})$, but the data now include the features and outcome instead of the features only. The parameters are then updated in a gradient ascent fashion as (cf. supplementary material for more details):

$$\psi \leftarrow \psi + \eta\left(\mathbb{E}_{\tilde{p}}[\nabla_\psi E(\mathbf{v}, \mathbf{h})] - \mathbb{E}_p[\nabla_\psi E(\mathbf{v}, \mathbf{h})]\right) \quad (15)$$
$$\boldsymbol{\theta} \leftarrow \boldsymbol{\theta} + \eta\left(\mathbb{E}_{\tilde{p}}[\boldsymbol{\phi}(\mathrm{y}, \mathbf{h})] - \mathbb{E}_{\tilde{p}}[\nabla_{\boldsymbol{\theta}} B(\boldsymbol{\theta}, \mathbf{h})]\right) \quad (16)$$

for a learning rate $\eta > 0$. $\tilde{p}(\mathbf{h} \mid \mathbf{v}, \mathrm{y})$ denotes the data distribution, and $p(\mathbf{v}, \mathbf{h})$ the model distribution of RBM part in sRBM. It is intractable to compute both two expectations exactly as it requires the sum over exponential space. We choose a stochastic method known as contrastive divergence (CD) (Hinton, 2002) which runs short Markov chains started from the data to approximate the model expectation.

For the data expectation, we use the approximation variational inference of $\mathbb{E}_{p(\mathbf{h}|\mathbf{v},\mathrm{y})}$ as described in Section 3.2. The mean-field update rule in Eq. (14) depends on the forms of sufficient statistics $\boldsymbol{\phi}(\mathrm{y}, \mathbf{h})$ and log-partition function $B(\boldsymbol{\theta}, \mathbf{h})$. For multiclass classification, the parameter $\boldsymbol{\theta}$ is a $\mathrm{K} \times \mathrm{C}$ matrix, and the probability density $p(\mathrm{y} \mid \mathbf{h}; \boldsymbol{\theta}, \boldsymbol{\beta})$ in Eq. (5) is given by:

$$\exp\left\{\boldsymbol{\theta}_{\cdot\mathrm{y}}^\top \mathbf{h} + \beta_{\mathrm{y}} - \log\sum\nolimits_c \exp\left(\boldsymbol{\theta}_{\cdot c}^\top \mathbf{h} + \beta_c\right)\right\}$$

in which $\boldsymbol{\theta}_{\bullet c}$ is the c-th column, and we have further introduced the bias β_c for the c-th class, hence:

$$\boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathrm{y}, \mathbf{h}) \triangleq \boldsymbol{\theta}_{\mathrm{y}\bullet}^\top \mathbf{h} + \beta_{\mathrm{y}}$$
$$B(\boldsymbol{\theta}, \mathbf{h}) = \log \sum\nolimits_{c=1}^{\mathrm{C}} \exp\left(\boldsymbol{\theta}_{\bullet c}^\top \mathbf{h} + \beta_c\right)$$

The update rule in Eq. (14) when using the first-order approximation in Eq. (11) now reads:

$$\mu_k \leftarrow \mathrm{sig}\left(\theta_{k\mathrm{y}} + \mathrm{b}_k + \sum_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk} - \frac{\sum_c \theta_{kc} e^{\left(\boldsymbol{\theta}_{\bullet c}^\top \boldsymbol{\mu} + \beta_c\right)}}{\sum_l e^{\boldsymbol{\theta}_{\bullet l}^\top \boldsymbol{\mu} + \beta_l}}\right)$$

The derivation for the second-order in Eq. (12) is more complicated, thus we refer to the supplementary material.

For regression problem, the parameter $\boldsymbol{\theta}$ is a K-dimensional vector, and the outcome variable follows the following Gaussian distribution: $p(\mathrm{y} \mid \mathbf{h}; \boldsymbol{\theta}, \beta) =$

$$\frac{1}{\sqrt{2\pi}} \exp\left[-\frac{1}{2\sigma^2}\mathrm{y}^2 + \frac{\boldsymbol{\theta}^\top \mathbf{h} + \beta}{\sigma^2}\mathrm{y} - \frac{\left(\boldsymbol{\theta}^\top \mathbf{h} + \beta\right)^2}{2\sigma^2} - \log \sigma\right]$$

with the standard deviation σ and bias β, thus:

$$\boldsymbol{\theta}^\top \boldsymbol{\phi}(\mathrm{y}, \mathbf{h}) \triangleq -\mathrm{y}^2 / \left(2\sigma^2\right) + \mathrm{y}\left(\boldsymbol{\theta}^\top \mathbf{h} + \beta\right) / \sigma^2$$
$$B(\boldsymbol{\theta}, \mathbf{h}) = \left(\boldsymbol{\theta}^\top \mathbf{h} + \beta\right)^2 / \left(2\sigma^2\right) + \log \sigma$$

Fixing $\sigma = 1$, the update rule in Eq. (14) when using the first-order approximation in Eq. (11) now reads:

$$\mu_k \leftarrow \mathrm{sig}\left[\theta_k \mathrm{y} + \mathrm{b}_k + \sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk} - \theta_k\left(\boldsymbol{\theta}^\top \boldsymbol{\mu} + \beta\right)\right]$$

, and using the second-order approximation in Eq. (12) is:

$$\mu_k \leftarrow \mathrm{sig}[\theta_k \mathrm{y} + \mathrm{b}_k + \sum\nolimits_{n=1}^{\mathrm{N}} \mathrm{v}_n \mathrm{w}_{nk} - \theta_k\left(\boldsymbol{\theta}^\top \boldsymbol{\mu} + \beta\right) - \frac{1}{2}\theta_k^2 (1 - 2\mu_k)]$$

We refer to the supplementary material[1] for the pseudo-code of learning parameters for sRBM using CD-1. Once the model is fully specified, the new representation of an input data can be achieved by computing the posterior vector $\hat{\mathbf{h}} = \left(\hat{\mathrm{h}}_1, \hat{\mathrm{h}}_2, ..., \hat{\mathrm{h}}_{\mathrm{K}}\right)$, where $\hat{\mathrm{h}}_k$ is shorthand for $\hat{\mathrm{h}}_k = p(\mathrm{h}_k = 1 \mid \mathbf{v})$ in Eq. (8).

4 EXPERIMENTS

In this section, we examine the predictive and generative capabilities of sRBM, and the effectiveness of discriminative representations and semantically related features discovered by the proposed model on three tasks – classification, regression and data generation. For classification, we use identical experimental setups to those of (Larochelle *et al.*, 2012) in order to directly compare our method with the current state-of-the-art – ClassRBM – and other baselines. Here we consider the ClassRBM with generative training objective since it is the most relevant approach to our model that is also learned in a generative fashion. We also would like to note that we do not compare with deep neural nets because our focus is on generative models.

Datasets. We use three datasets: MNIST[2], 20 Newsgroups (Mitchell, 1997) and CT slices obtained from UCI repository[3]. After ordinal preprocessing steps (e.g., normalizing, rescaling), they are not exactly binary data. Following the previous work (Hinton and Salakhutdinov, 2006), we treat their feature values as empirical probabilities on which the RBM-based models are naturally applied. Since the empirical expectations $\mathbb{E}_{\tilde{p}}[\bullet]$ in Eq. (15) and Eq. (16) require the probability $p(\mathbf{v})$, the normalized intensity is a good approximation.

Methods and baselines. We create two versions of our approach: sRBM using the first-order approximation (sRBM-1st) and the second-order (sRBM-2nd). For classification performance comparison, we employ 5 baselines: ClassRBM with generative training strategy (Larochelle *et al.*, 2012), k-nearest neighbors (kNN, where $k = 1$, with cosine similarity measures) and support vector machine (SVM) directly on the hidden posteriors of sRBM (sRBM+kNN, sRBM+SVM), ClassRBM (ClassRBM+kNN, ClassRBM+SVM) and RBM (RBM+kNN, RBM+SVM). We obtain ClassRBM code[4] that reproduces the results of (Larochelle *et al.*, 2012). For regression, we use: ridge regression (RR) – linear regression with ℓ_2-norm regularization, and RR on top of other methods, similar to the setup for classification.

Hyperparameter settings. For RBM-based models, mapping parameters are randomly drawn from $\mathcal{N}(0, 0.01)$, and biases are set to zeros. Learning is terminated after a number of epochs over training set, that was specified using early stopping based on the error of validation set in every interval of 15 epochs. For a fair comparison, we empirically tune the learning rate of each model: $\eta \in \{0.1, 0.05, 0.01, 0.005, 0.001\}$ using cross-validation for the best result on validation part.

4.1 IMAGE AND TEXT CLASSIFICATION

In the first task, we replicate the experimental settings in Sections 7.1 and 7.2 in (Larochelle *et al.*, 2012). In particular, we use MNIST and the "bydate" version of 20 Newsgroups datasets to validate the predictive performance of our proposed model on image and text data. The MNIST consists of $50,000$ training; $10,000$ validation; and $10,000$ testing 28×28 grayscale handwritten digit images whose

[1] https://tund.github.io/papers/tu_etal_uai17_srbm_supp.pdf

[2] http://yann.lecun.com/exdb/mnist/.

[3] https://archive.ics.uci.edu/ml/datasets.html.

[4] https://github.com/skaae/rbm_toolbox

pixel values are then normalized into the range $[0, 1]$. The 20 Newsgroups consists of $9,578$ training; $1,691$ validation; and $7,505$ testing documents. We first extract $5,000$ most frequent words based on their counts in the training part, then use their appearances as binary bag-of-word features for each document.

The number of hidden units is set to $6,000$ for MNIST and to $1,000$ for 20 Newsgroups. Once again, we note that these numbers are identically set to those in (Larochelle *et al.*, 2012). The results for different numbers of hidden units are provided in the supplementary material. Overall, the larger hidden layers yield better classification results, which is plausible since the hidden units need to capture both data and labels. This finding is also consistent with that of the ClassRBM.

Table 1 reports the classification errors on testing set for all methods. For handwritten recognition, our proposed model significantly outperforms all baselines, based on the fact that on MNIST, differences of more than 0.1% are statistically significant. This result, to the best of our knowledge, is the highest among those of standalone RBMs trained in a generative fashion. For document classification, our model, with 5-fold dimensionality reduction from $5,000$ words to $1,000$ hidden units, also achieves the best result.

Table 1: Classification errors (%) on testing sets.

	MNIST	20 Newsgroups
RBM+kNN	3.03	56.15
RBM+SVM	1.76	41.79
ClassRBM+kNN	2.98	57.80
ClassRBM+SVM	1.68	40.88
ClassRBM	3.39	24.9
sRBM+kNN	2.94	55.89
sRBM+SVM	1.42	38.43
sRBM-1st	2.27	24.1
sRBM-2nd	**2.21**	**23.2**

Discriminative representations. One could question that whether the discriminative representations learned by sRBM would have more advantages. The first argument is these representations, learned with guidance from labels, could straightforwardly translate into better predictive performance without the need for well-trained classifiers. Looking at the classification errors of sRBM+kNN, ClassRBM+kNN, and RBM+kNN in Table 1, we believe that the learned representations of sRBM would be more naturally discriminative than those of other RBMs. The results on MNIST are much better than those on 20 newsgroups because the MNIST is a clean, well-preprocessed dataset, whilst 20 newsgroups is a more difficult text dataset in higher dimensional space.

For further comparison, we quantitatively compute the pairwise cosine similarities between $6,000$-dimensional hidden posteriors: $\mathrm{c}_{ij} = \mathrm{cosine}(\hat{\mathbf{h}}_i, \hat{\mathbf{h}}_j)$ of data points $(\mathbf{v}_i, \mathrm{y}_i)$ and $(\mathbf{v}_j, \mathrm{y}_j)$. The similarity between a data sample and the rest is: $\mathrm{c}^\star_i = {}^1\!/\!_{\mathrm{M}} \left[\sum_{\forall j|\mathrm{y}_j=\mathrm{y}_i} \mathrm{c}_{ij} - \sum_{\forall j|\mathrm{y}_j\neq\mathrm{y}_i} \mathrm{c}_{ij}\right]$ where M is the number of data points. The average similarities over all samples of sRBM, ClassRBM, and RBM are $\mathbf{-0.02}$, $\mathbf{-0.36}$, and $\mathbf{-0.61}$, which demonstrates that our proposed model has significantly higher power of disentangling data from different classes. For quality demonstration, we project $5,000$ hidden posteriors of sRBM onto 2D using t-SNE[5] (Van der Maaten and Hinton, 2008). Fig. 2 depicts the distribution of where class information is only used for visual labeling. The separation is over satisfactory.

Figure 2: An illustration of 10 digit classes from $5,000$ MNIST images. t-SNE projection performs on higher representations of images mapped by sRBM.

Topic modeling. The next aspect is whether the sRBM effectively models the topics, so that it can obtain better classification performance. Here we examine the weight connections between the hidden and output layers. This offers a method to embed discrete topics (classes) in a continuous space using $\boldsymbol{\theta}_{\boldsymbol{\cdot}t} \in \mathbb{R}^{\mathrm{K}}$, i.e., the t-th column of weight matrix $\boldsymbol{\theta}$. A topic is now represented by a vector of K dimensions. This embedding is closely related to the word embedding that has found numerous applications in natural language processing (Mikolov *et al.*, 2013).

To assess the correlation between topics, we compute pairwise similarities between embedded vectors: $\mathrm{s}_{ij} = \mathrm{sig}(\boldsymbol{\theta}_{\boldsymbol{\cdot}i}^\top \boldsymbol{\theta}_{\boldsymbol{\cdot}j})$ and present them in Fig. 3. It can be seen that the sRBM effectively disentangles different topics, whilst still retains a certain degree of similarity for semantically related topics. More precisely, the bounding squares in the figure indicate groups of topics such as computer (comp.*), science (sci.*) and politics (talk.politics.*), or secondary topics such as sports (rec.sports.*) and other recreational activities (rec.autos and rec.motorcycles). Interestingly, the model can also recognize the resemblance of some separated groups (alt.atheism, soc.religion.christian and talk.*) that are indeed semantically related.

Similar results are obtained by the ClassRBM as shown in

[5]Note that the t-SNE does not perform clustering, it only reduces the dimensionality into 2D for visualization while still strives to preserve the local properties of the data.

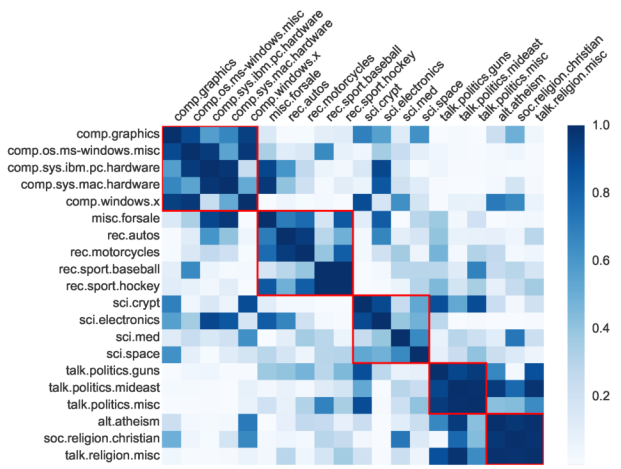

Figure 3: Similarities among weight vectors of the sRBM associated with 20 Newsgroups.

Fig. 3 in (Larochelle *et al.*, 2012)). For quantitative comparison, we compute the correlation of a topic with the rest as: $\mathrm{s}_i^\star = {}^1\!/\!_{\mathrm{T}-1} \sum_{j \neq i} \mathrm{s}_{ij}$. The result is favorable toward sRBM with the average correlation over all topics of $\mathbf{0.314}$ that is lower than $\mathbf{0.399}$ of ClassRBM. This, once again, demonstrates that our proposed method disentangles different topics better. The gap is much smaller than that of similarities between hidden posteriors, which can be reflected by the fact that the improvement in classification performances on 20 Newsgroups is much less significant than the one on MNIST dataset (cf. Table 1).

For document modeling task on 20 Newsgroups dataset, we analyze how our proposed model captures words that are coherent in a topic by examining the weight matrices $\boldsymbol{\theta}$ and $\mathbf{W}$. The entry of column $\mathbf{w}_{\boldsymbol{\cdot}k}$ reflects the association strength of a particular word to the latent factor k, and $\boldsymbol{\theta}_{\boldsymbol{\cdot}c}$ the strength of a latent factor to the topic c. We first specify top 100 hidden units with the largest weight for each topic c, then aggregate (by summing) the associated word-to-hidden weight vectors. This reveals the positive contribution of the words to each newsgroup via the hidden layer. Fig. 4 illustrates top 8 words per topic, in descending order of their aggregated association strength, discovered by our model. The chart shows that the words under each feature are semantically related in a coherent way.

4.2 LABEL-DRIVEN DATA GENERATION

As a generative model, it is desirable that our proposed method can generate meaningful data after training. In this experiment, we analyze the capability of sRBM in drawing images and words in documents for specified handwritten digit and news topics. To generate a data sample, we first fix a label y, then randomly initialize $\hat{\mathbf{h}} \in [0, 1]^{\mathrm{K}}$. Next we alternatively sample the data $\hat{\mathbf{v}} \sim p(\mathbf{v} \mid \hat{\mathbf{h}})$ and the hidden units $\hat{\mathbf{h}} \sim p(\mathbf{h} \mid \mathbf{v}, \mathrm{y})$ using 100 separate Markov chains, then collect data $\hat{\mathbf{v}}$ at the 1000^{th} sampling step. For handwritten digits, we generate 10 images for each digit. For news groups, we choose to generate documents (bags of words) for three topics: *rec.autos*, *sci.space* and *comp.sys.mac.hardware*. For each group, we generate 100 documents, and then select 50 most frequent words with their frequencies.

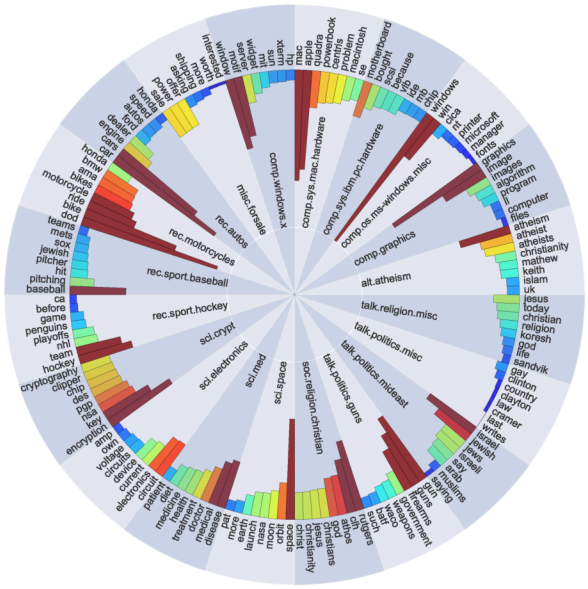

Figure 4: Topics associated with top 8 words discovered by sRBM from 20 Newsgroups dataset. The bar height and color relatively represent the aggregated association strength of a word. (Best viewed in colors and refer to the supplementary material for larger plot).

Fig. 5 (left) and Fig. 6 show images of handwritten digits and top words of three news topics generated by the sRBM. In general, our model can draw correct images on each row from top to bottom for each digit from 0 to 9, and generate words that are semantically related in a coherent way for each topic. In particular, it can capture factors of variation from the data, i.e., digits are sampled with clearly different styles. For reference, we further show the images generated by VAE in Fig. 5 (middle) and GAN in Fig. 5 (right). Here we do not make any qualitative and quantitative comparison since we do not claim our main contribution in data generation.

4.3 LOCATION PROGNOSIS FOR CT IMAGES

In the last application, the aim is to predict the relative location for computed tomography (CT) slice. The dataset consists of $53,500$ CT images captured from 74 different patients (Graf *et al.*, 2011). Two histograms in polar space are extracted from each CT slice, forming a 385-dimensional feature vector. The response variable is the relative location of an image on the axial axis with the value in $[0, 180]$, and then is rescaled into $[0, 1]$.

We divide the dataset into three disjoint parts: 80% for training, 5% for validation and 15% for testing with 42800, 2675 and 8025 data samples respectively. The validation

Figure 5: Handwritten digit images generated by sRBM (left) for 10 classes (each row is generated conditionally on the digit labels from 0 to 9), VAE (middle), and GAN (right). Images generated by VAE and GAN are taken from original work – (Kingma and Welling, 2014) and (Salimans *et al.*, 2016).

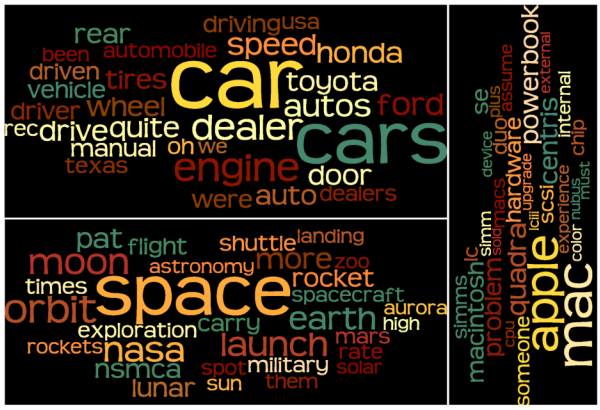

Figure 6: Words generated by sRBM for (*rec.autos*, *sci.space*, *comp.sys.mac.hardware*) news groups.

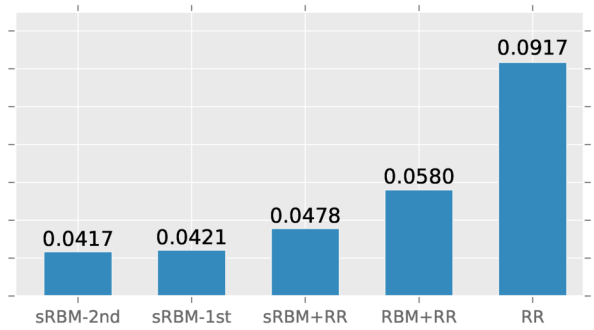

Figure 7: Regression errors (%) on testing data of CT slice dataset (lower is better).

set is used for hyperparameter tuning step wherein the number of hidden units K is varied in $\{100, 200, 500, 1000\}$. In general, we observe that the larger K improves the performance of our method, thus we use $\mathrm{K} = 1,000$. Note that we normalize the responses to zero mean and unit variance before training the sRBM. In testing phase, the sRBM maps the prediction values back into the original space.

We use root mean square error (RMSE) to measure the regression results. Fig. 7 shows that our proposed sRBM-1st achieves a regression error that is 54%, 27%, and 12% better than those of ridge regression (RR), RBM+RR, and sRBM+RR respectively, and the second-order version (sRBM-2nd) slightly outperforms the first-order (sRBM-1st). This, again, demonstrates that the unified learning of sRBM results in more discriminative representations and better prediction scores than two-stage framework of unsupervised models followed by predictive models.

5 CONCLUSION

We have introduced a novel self-contained prediction model – supervised restricted Boltzmann machine (sRBM) that leverages the powerful RBM to simultaneously learn data representations and perform supervised learning. Different from the current state-of-the-art classification formulation of RBM – ClassRBM (Larochelle *et al.*, 2012), our model is a *hybrid* probabilistic graphical model with a generative component for data representation and a discriminative component for both classification and regression.

We have further developed variational inference for sRBM to tackle the challenging inference posed by the introduction of the discriminative component. We evaluated the effectiveness of our model using comprehensive experiments on three real-world applications. The results confirm that the integration of the discriminative component into RBM: (i) significantly improves the classification performance of the ClassRBM; (ii) renders more discriminative representations; (iii) generates meaningful data driven by classes; and (iv) achieves better regression results.

ACKNOWLEDGMENT. This work was partially supported by the Australian Research Council (ARC) Discovery Grant Project DP150100031.

References

Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 35(8):1798–1828, 2013.

Adam Coates and Andrew Y Ng. The importance of encoding versus training with sparse coding and vector quantization. In *The 28th International Conference on Machine Learning (ICML)*, pages 921–928, 2011.

Yoav Freund and David Haussler. Unsupervised learning of distributions on binary vectors using two layer networks. Technical Report UCSC-CRL-94-25, University of California Santa Cruz (UCSC), CA, USA, 1994.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems (NIPS)*, pages 2672–2680. 2014.

Franz Graf, Hans-Peter Kriegel, Matthias Schubert, Sebastian Pölsterl, and Alexander Cavallaro. 2d image registration in ct images using radial image descriptors. In *Medical Image Computing and Computer-Assisted Intervention*, pages 607–614. Springer, 2011.

Geoffrey Hinton and Ruslan Salakhutdinov. Reducing the dimensionality of data with neural networks. *Science*, 313(5786):504–507, 2006.

Geoffrey E. Hinton. Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8):1771–1800, 2002.

Michael I Jordan, Zoubin Ghahramani, Tommi S Jaakkola, and Lawrence K Saul. *An introduction to variational methods for graphical models*. Springer, 1998.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. In *The International Conference on Learning Representations (ICLR)*, Banff, 2014.

Hugo Larochelle, Michael Mandel, Razvan Pascanu, and Yoshua Bengio. Learning algorithms for the classiffcation restricted boltzmann machine. *Journal of Machine Learning Research*, pages 643–669, 2012.

Xin Li, Feipeng Zhao, and Yuhong Guo. Conditional restricted boltzmann machines for multi-label learning with incomplete labels. In *Artificial Intelligence and Statistics (AISTATS)*, pages 635–643, 2015.

A. McCallum, C. Pal, G. Druck, and X. Wang. Multi-conditional learning: Generative/discriminative training for clustering and classification. In *The National Conference on AI*, volume 1, pages 433–439, 2006.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. *Neural Information Processing Systems (NIPS)*, pages 3111–3119, 2013.

Thomas M Mitchell. Machine learning. *Artificial Intelligence*, 1997. (citation for 20 newsgroups dataset).

Tu Dinh Nguyen, Truyen Tran, Dinh Phung, and Svetha Venkatesh. Latent patient profile modelling and applications with mixed-variate restricted boltzmann machine. In *The 17th Pacific-Asia Conference on Knowledge Discovery and Data Mining (PAKDD)*, volume 7818, pages 123–135, Gold Coast, Australia, 2013.

Tu Dinh Nguyen, Truyen Tran, Dinh Phung, and Svetha Venkatesh. Learning parts-based representations with nonnegative restricted Boltzmann machine. In *Proceedings of the 5th Asian Conference on Machine Learning (ACML)*, volume 29, pages 133–148, Canberra, Australia, November 13–15 2013.

Tu Dinh Nguyen, Truyen Tran, Dinh Phung, and Svetha Venkatesh. Learning sparse latent representation and distance metric for image retrieval. In *Proceedings of IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6, USA, 2013.

R. Salakhutdinov and GE Hinton. Replicated softmax: an undirected topic model. In *The 23rd Neural Information Processing Systems (NIPS)*, pages 1607–1614, Vancouver, Canada, December 7–10 2009.

Ruslan Salakhutdinov and Geoffrey Hinton. Deep Boltzmann machines. In *The 12th International Conference on Artificial Intelligence and Statistics (AISTATS)*, volume 5, pages 448–455, USA, 2009.

Ruslan Salakhutdinov. *Learning deep generative models*. PhD thesis, Ph.D. Thesis, University of Toronto, 2009.

Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *Neural Information Processing Systems (NIPS)*, pages 2226–2234, 2016.

T.a Schmah, G.E.a Hinton, R.S.a Zemel, S.L.b Small, and S.c Strother. Generative versus discriminative training of rbms for classification of fmri images. In *Advances in Neural Information Processing Systems 21 (NIPS)*, pages 1409–1416, 2009.

Nitish Srivastava and Ruslan Salakhutdinov. Multimodal learning with deep Boltzmann machines. In *The 26th Neural Information Processing Systems (NIPS)*, volume 3, pages 2222–2230, USA, December 3–6 2012.

Robert Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (Methodological)*, pages 267–288, 1996.

L. Van der Maaten and G. Hinton. Visualizing data using t-SNE. *The Journal of Machine Learning Research (JMLR)*, 9:2579–2625, 2008.

J.a Yang, Y.a b Liu, E.P.a Xing, and A.G.a Hauptmann. Harmoniurn models for semantic video representation and classification. In *The 7th SIAM International Conference on Data Mining (SDM)*, pages 378–389, 2007.