Biasing Restricted Boltzmann Machines to Manipulate Latent Selectivity and Sparsity

Hanlin Goh, Nicolas Thome, Matthieu Cord

Biasing Restricted Boltzmann Machines to Manipulate Latent Selectivity and Sparsity

Hanlin Goh*, **Nicolas Thome, Matthieu Cord**
Laboratoire d'Informatique de Paris 6
UPMC – Sorbonne Universités, Paris, France
{hanlin.goh,nicolas.thome,matthieu.cord}@lip6.fr

Abstract

This paper proposes a modification to the restricted Boltzmann machine (RBM) learning algorithm to incorporate inductive biases. These latent activation biases are ideal solutions of the latent activity and may be designed either by modeling neural phenomenon or inductive principles of the task. In this paper, we design activation biases for sparseness and selectivity based on the activation distributions of biological neurons. With this model, one can manipulate the selectivity of individual hidden units and the sparsity of population codes. The biased RBM yields a filter bank of Gabor-like filters when trained on natural images, while modeling handwritten digits results in filters with stroke-like features. We quantitatively verify that the latent representations assume the properties of the activation biases. We further demonstrate that RBMs biased with selectivity and sparsity can significantly outperform standard RBMs for discriminative tasks.

1 Introduction

Restricted Boltzmann machines (RBMs) [17] are generative neural networks that form distributed representations capturing the latent structure of input data by approximating the maximum likelihood (ML) objective. However, given the nature of a task, learning based on ML alone may not be the most desirable approach. To manipulate the representations during learning, suitable inductive biases [13] should be incorporated so that an RBM can inherit certain desirable representational properties. The inductive biases are a priori assumptions about the nature of the target function and should be exploited to facilitate the learning process. In this paper, we will introduce a simple modification to incorporate generic inductive biases into RBM learning. With this, we design the biases to manipulate the selectivity and sparsity of latent representations. Through a series of experiments, we quantitatively show the importance of having both selectivity and sparsity in our representations.

1.1 Restricted Boltzmann Machines

An RBM consists of a layer of I visible units $\mathbf{v}$ and a layer of J hidden units $\mathbf{h}$. The layers are linked via symmetric weighted connections $\mathbf{W} \in \mathbb{R}^{I \times J}$. Additionally, each visible v_i and hidden h_j unit receives input from a bias – c_i and b_j respectively. For an RBM with binary units, the activation probabilities of units in one layer are computed based on the states of the opposite layer, fed through a sigmoid activation function $sigm(\cdot)$:

$$\Pr\left(h_j \mid \mathbf{v}\right) = sigm\left(b_j + \sum_{i=1}^{I} w_{ij} v_i\right), \quad (1) \qquad \Pr\left(v_i \mid \mathbf{h}\right) = sigm\left(c_i + \sum_{j=1}^{J} w_{ij} h_j\right). \quad (2)$$

*Hanlin Goh is also with the Institute for Infocomm Research, A*STAR, Singapore and the Image & Pervasive Access Lab, CNRS UMR 2955, Singapore – France.

An energy function defines the negative log probability of a configuration of states $(\mathbf{v}, \mathbf{h})$:

$$-\log \Pr(\mathbf{v}, \mathbf{h}) = E(\mathbf{v}, \mathbf{h}) = -\sum_{i=1}^{I}\sum_{j=1}^{J} v_i w_{ij} h_j - \sum_{i=1}^{I} c_i v_i - \sum_{j=1}^{J} b_j h_j. \tag{3}$$

By modifying the parameters $\mathbf{W}$, $\mathbf{b}$ and $\mathbf{c}$, the energy of samples from the data distribution can be decreased, while raising the energy of reconstructions that the network prefers to real data.

1.2 RBM Learning Using Contrastive Divergence

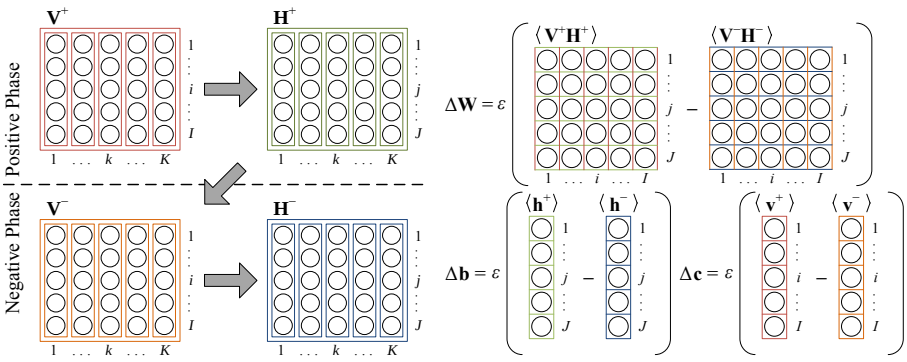
To train an RBM, one employs the contrastive divergence (CD) learning algorithm [8] to approximate the ML of the data and update parameters $\mathbf{W}$, $\mathbf{b}$ and $\mathbf{c}$. The RBM learning algorithm with one iteration of stochastic Gibbs sampling (CD-1) is described in Figure 1. Given a set of K training examples, $\mathbf{V}^+ \in \mathbb{R}^{I \times K}$ and $\mathbf{H}^+ \in \mathbb{R}^{J \times K}$ are visible and hidden states resulting from sampling from the data distribution, while $\mathbf{V}^- \in \mathbb{R}^{I \times K}$ and $\mathbf{H}^- \in \mathbb{R}^{J \times K}$ are reconstructed states. The parameters are updated using the following update rules:

$$\Delta w_{ij} = \varepsilon\left(\langle v_i^+ h_j^+ \rangle - \langle v_i^- h_j^- \rangle\right), \tag{4}$$

$$\Delta b_j = \varepsilon\left(\langle h_j^+ \rangle - \langle h_j^- \rangle\right), \tag{5}$$

$$\Delta c_i = \varepsilon\left(\langle v_i^+ \rangle - \langle v_i^- \rangle\right), \tag{6}$$

where ε is the learning rate and $\langle \cdot \rangle$ is defined as the average over the set of K examples. The activation probabilities of v_i and h_j may be used in place of their binary states for parameter updates (see [7]). This process, known as Rao-Blackwellization [1], results in an estimator with lower variance [18]. During parameter updates we will adopt this convention.



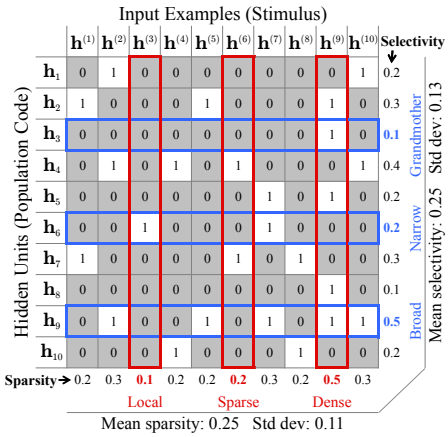
(a) Visible and hidden unit activations. (b) Parameter updates.

Figure 1: The RBM learning algorithm (CD-1). (a) In the positive phase, hidden units $\mathbf{H}^+$ are activated based solely on inputs $\mathbf{V}^+$. The negative phase involves the reconstruction of $\mathbf{V}^-$ from $\mathbf{H}^+$ and subsequently $\mathbf{H}^-$ from $\mathbf{V}^-$. (b) The parameters $\mathbf{W}$, $\mathbf{b}$ and $\mathbf{c}$ are updated based on the gradients from the positive and negative phases.

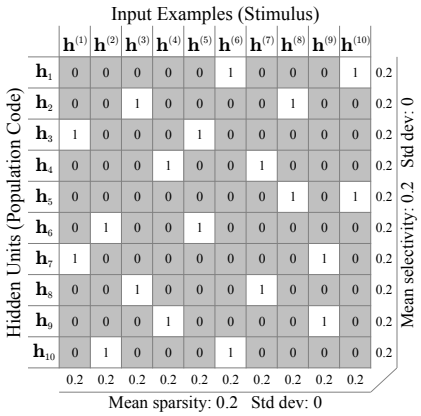
1.3 Motivation of Inducing Both Selectivity and Sparsity

Selectivity and sparsity are important properties of neural coding. Neural activity can be characterized in the lifetime or population domains [21], resulting in the properties of selectivity and sparsity respectively (Figure 2(a)). Selectivity is the property of a *single neuron* given a *series of stimuli* over time. In contrast, sparsity is defined across a *population of neurons* given a *single stimulus* [4]. The two properties are not necessarily correlated [21] and are related only by their average values [4].

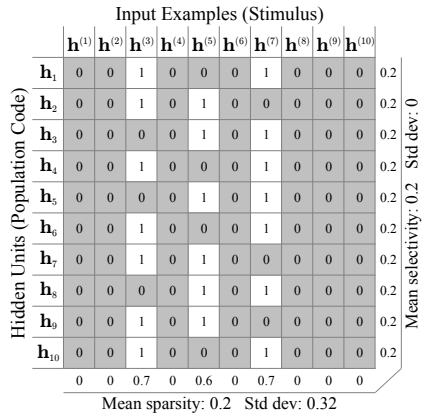
Narrowly selective neurons do not guarantee the generation of sparse population codes (Figure 2(c)). Likewise, sparse codes may be induced from neurons that may not be highly selective (Figure 2(d)). We can visually observe from Figure 2(b) that when the code is both highly selective and sparse, there is activation diversity in the population and across samples. Sparsity causes lateral inhibition and encourages competition between hidden units, while selectivity prevents over-dominance by any individual unit. When this occurs, the standard deviation is low for selectivity in the population and sparsity across examples.



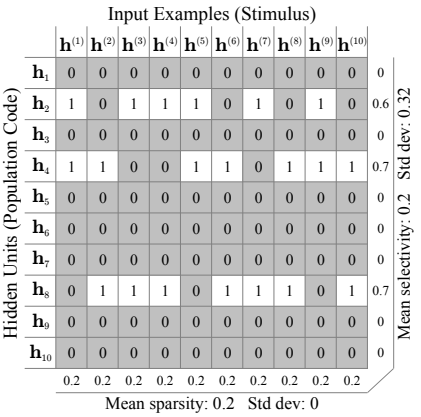
(a) Distinguishing selectivity and sparsity.



(b) Narrowly selective sparse codes.



(c) Narrow selectivity but not sparse coded.



(d) Sparsely coded but not narrowly selective.

Figure 2: Examples of binary unit activations $h_j^{(k)}$ relating selectivity and sparsity. (a) Selectivity is a property of a single unit, given the set of input samples. Sparsity is a property of the population of hidden units, given one input example. The two properties are related only by their average values. (Terminology adapted from [4]) (b) An example of units with narrow selectivity and sparse population codes. (c) When units have narrow selectivity, the population may not induce sparse codes. (d) Likewise, sparse activations are not necessarily the product of units with narrow selectivity.

2 RBM Learning with Latent Activation Biasing

An RBM does not explicitly consider the nature of the task or that it may be part of a more complex network, such as deep belief nets (DBN) [9] or other hierarchical architectures. If we have a priori knowledge of the desired latent representational properties for a particular task, such as selectivity and sparsity, how can this information be included during RBM learning? For this purpose, we extend existing work on regularizing RBMs [12, 14] and design a more generic model to incorporate any inductive principle as latent activation biases during RBM learning.

2.1 RBMs with Selective Regularization

There are several ways to achieve selectivity in RBMs.[1] Lee et al. [12] proposed to couple the ML approximation of contrastive divergence with a regularization term that penalizes non-selective hidden units. Similarly, Nair & Hinton [14] used the cross-entropy measure between the actual and desired distributions to compute the penalty. In both cases, the additional update is a penalty

[1] In the literature, the terms "sparsity" and "selectivity" are sometimes used interchangeably, sometimes leading to confusion [4]. We use "selectivity" to describe the activity of an individual unit across examples and "sparsity" to describe the activity of a population in response to a single example.

proportional to $q - p$, where p is the target selectivity of each unit and q is the observed selectivity. The selectivity of a hidden unit is computed by a process of averaging its activation across training examples. A term η is included to scale the learning of the regularizer.

For a hidden unit to be selective, it should respond strongly to only a few examples and have low activation probabilities for the other examples. However, these methods merely regularize the learning such that the activation probabilities are low on the average. Even when the selectivity objective p is satisfied, the unit's activation may not be selective (for example, $h_j^{(k)+} = p, \forall k \in K$). As such, the hidden units need to be stochastic and binary. In the case where the activations are selective but the average is higher than p, the penalty term penalizes all activations equally to bring the average down. By penalizing the high activations, one sends a signal for it to be less selective. Furthermore, since the regularizer considers only selectivity and not sparsity, we may get units that lack differentiation between each other. A population of hidden units that respond selectively but similarly to a few examples will still satisfy the regularization objectives individually (see Figure 2(c)). We propose to have a more precise and fine-grained control of the regularization process to overcome these issues.

2.2 Precise Biasing of Latent Activations

To have a more precise control of the regularization, we realize the target p as a spatiotemporal matrix $\mathbf{P} \in \mathbb{R}^{J \times K}$, where each element $p_j^{(k)} \in [0, 1]$ is a latent activation bias encoding the desired activation of h_j in response to input example k. Each row $\mathbf{p}_j$ represents the desired temporal activation sequence of h_j, while a column $\mathbf{p}^{(k)}$ is the ideal population code of hidden units given example k. More generally, $\mathbf{P}$ can be designed based on any inductive principle, not just selectivity. Inspired by Nair & Hinton [14], we adopted the cross-entropy error between the desired and actual activation probabilities for the penalty term. We now pose the following optimization problem:

$$\underset{\{\mathbf{W},\mathbf{c},\mathbf{b}\}}{\arg\min} \; -\sum_{k=1}^{K} \log \sum_{\mathbf{h}} \Pr\left(\mathbf{v}^{(k)}, \mathbf{h}^{(k)}\right) + \lambda \sum_{j=1}^{J} p_j^{(k)} \log h_j^{(k)+} + \left(1 - p_j^{(k)}\right) \log\left(1 - h_j^{(k)+}\right), \quad (7)$$

The averaged update for w_{ij} and b_j across K examples can then be modified and rearranged:

$$\Delta w_{ij} \;=\; \varepsilon\left(\left\langle v_i^+ h_j^+ \right\rangle - \left\langle v_i^- h_j^- \right\rangle\right) - \eta \left\langle v_i^+ \left(h_j^+ - p_j\right)\right\rangle = \varepsilon\left(\left\langle v_i^+ s_j \right\rangle - \left\langle v_i^- h_j^- \right\rangle\right), \quad (8)$$

$$\Delta b_j \;=\; \varepsilon\left(\left\langle h_j^+ \right\rangle - \left\langle h_j^- \right\rangle\right) - \eta \left\langle h_j^+ - p_j \right\rangle = \varepsilon\left(\left\langle s_j \right\rangle - \left\langle h_j^- \right\rangle\right). \quad (9)$$

Here, we let

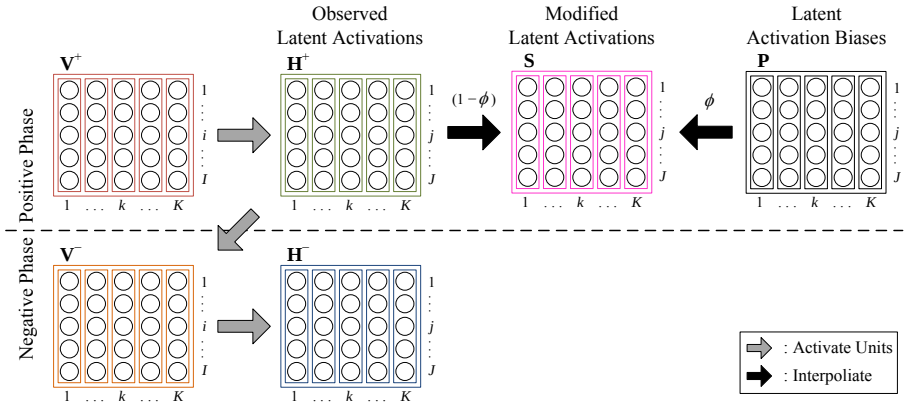
$$s_j^{(k)} = \phi p_j^{(k)} + (1 - \phi)\, h_j^{(k)+}, \quad (10)$$

where $\phi = \dfrac{\eta}{\varepsilon}$ is a hyperparameter. This modified algorithm is illustrated in Figure 3.

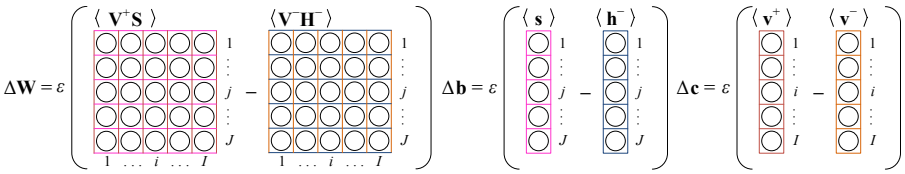
The influences of $p_j^{(k)}$ and $h_j^{(k)+}$ are interpolated by ϕ. If ϕ is constrained to be between 0 and 1, then $0 \leq s_j^{(k)} \leq 1$ and $s_j^{(k)}$ can be seen as the revised activation probability of $h_j^{(k)+}$. Because the biases are directly the desired activation probabilities of the hidden units given an example, Rao-Blackwellization can still be employed and we do not need to assume hidden units to be binary. When $\phi = 0$ or if the activation bias is met (i.e. $p_j^{(k)} = h_j^{(k)+}$), the parameter update equations simplify to those of the original contrastive divergence algorithm. Comparing Equations (8) and (9) with the original updates (Equations (4) and (5)), we observe an asymmetry in the new updates, which generates a learning signal that is guided by the latent activation biases. Note that biasing is only done in the training step.

3 Designing Latent Activation Biases for Selectivity and Sparsity

To manipulate the selectivity and sparsity of the representations, we turn to biology for inspiration. From neuronal recordings, it was found that the activity distributions for both selectivity and sparsity are positively skewed with heavy tails, such as the exponential and gamma distributions [5]. By adapting the activation probabilities of hidden units to fit such distributions in the lifetime (rows) or population (columns) domains, we can model their latent activity biases $\mathbf{P}$.



(a) Visible and hidden unit activations and inclusion of latent activation biases.



(b) Modified parameter updates.

Figure 3: The modified RBM learning algorithm with biased latent activations. (a) In the positive phase, hidden units $\mathbf{H}^+$ are re-activated as $\mathbf{S}$ with additional influences from latent activation biases $\mathbf{P}$ interpolated by ϕ. (b) When updating parameters $\mathbf{W}$ and $\mathbf{b}$, the modified activation $\mathbf{S}$ replaces $\mathbf{H}^+$, only the positive phase. $\Delta \mathbf{c}$ is unmodified from the original algorithm.

3.1 Modeling Selectivity and Sparsity

We transform the latent activations to fit desired distributions. Let $\mathbf{h} \in \mathbb{R}^N$ be either row $\mathbf{h}_j^+$ for selectivity or column $\mathbf{h}^{(k)+}$ for sparsity. The latent activation bias p_n for h_n is computed as

$$p_n = \left(rank\left(h_n, \mathbf{h}\right)\right)^{(1/\mu)-1}. \tag{11}$$

where $rank\left(h_n, \mathbf{h}\right)$ assigns a value from 0 to 1 based on the rank of h_n in $\mathbf{h}$, with smallest given a value of 0 and the largest with 1. The target mean $0 < \mu < 1$ creates the power-law expression such that when $\mu < 0.5$, the distribution will be positively skewed. Instead of merely getting the RBM to have low average activations [12, 14], we bias individual activations based on positively skewed distributions so that only a few activations are high while most remain low. This is more precise.

Unlike methods that produce stochastic spiking activations [14, 16], our method does not depend on the order of the inputs. It also preserves the ranking between activation probabilities. Moreover, it can be applied in both the lifetime and population domains to achieve both selectivity and sparsity.

3.2 Combining Selectivity and Sparsity

Because selectivity and sparseness are not highly correlated Willmore & Tolhurst [21], we need to combine the latent activation biases in both domains. We take $\mathbf{H}^+$ and progressively induce column-wise sparsity followed by row-wise selectivity with Equation (11). We use the same μ since the properties will produce the same mean values. This yields activation biases $\mathbf{P}$ that are consistent in both domains, such as the example shown in Figure 2(b).

4 Experiments

One of the advantages of RBMs is the ability to find correlations between input units, even though there are no direct connections between them. As a result, they are very suited to model the spatial-appearance of images. We study the properties and performance of our method using two image data sets, namely the Kyoto natural images data set [2] and the MNIST handwritten digits database [11].

4.1 Visualization: Modeling Natural Images

We trained RBMs biased with selectivity and sparsity to efficiently represent natural image patches. The training data consists of 100,000 randomly sampled patches of size 14×14 pixels from the Kyoto natural images data set [2]. The result a set of Gabor-like filters (Figure 4), which is consistent with other related methods [3, 12, 15, 16, 19]. Additionally, as a gauge of the generality of these filters, we adopted a transfer learning framework and reconstructed similar-sized patches from the MNIST data set. We obtained a 43% decrease in the reconstruction error as compared to the standard RBM, probably because these edge detectors are localized and some resemble handwritten strokes.

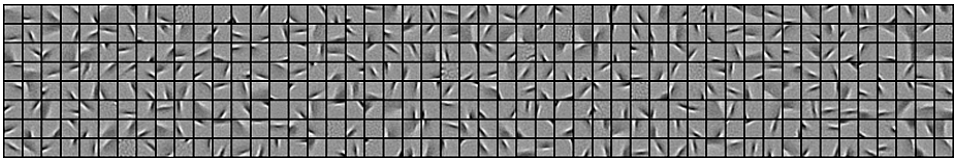


Figure 4: An example of a filter bank learned by an RBM with selectivity and sparsity biases. The filters are Garbor-like with varying orientation, spatial location and spatial frequency.

4.2 Evaluation: Modeling Handwritten Digits

The MNIST database [11] consists of 60,000 training and 10,000 test images of handwritten digits of size 28×28 pixels. For the data set, we used the training set to train RBMs with 1000 hidden units, while the test set is used for evaluation.When target mean μ of the activation biases was set sufficiently low, the learned filters appear to encode localized handwritten strokes (Figure 5).

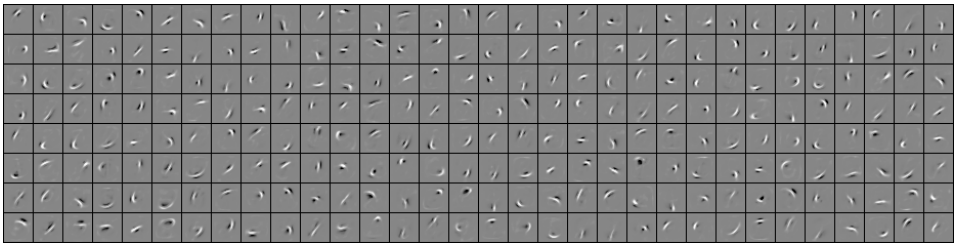


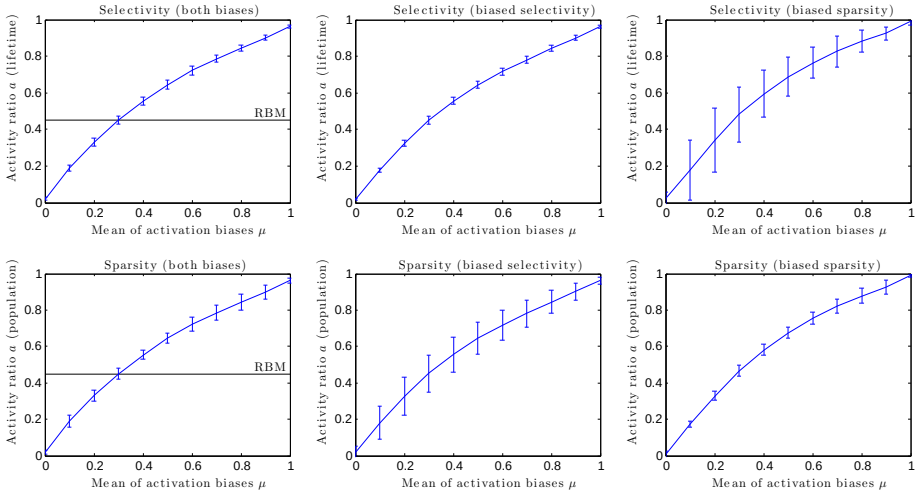
Figure 5: Examples of filters learned by the biased RBM when trained on handwritten digits.

Selectivity and sparsity are transferred from biases to parameters. The selectivity and sparsity of a set of activation probabilities can be measured using the activity ratio [20], which is a quantitative measure of length of the tail of the distribution of activations. It can be computed using either the continuous-valued activation probabilities or the binary activation states. Given a set of activations $\mathbf{x} \in \mathbb{R}^N$ in the lifetime or population domain, the activity ratio $a \in [0, 1]$ is defined as

$$a = \left(\tfrac{1}{N} \textstyle\sum_{n=1}^{N} x_n \right)^2 \Big/ \tfrac{1}{N} \textstyle\sum_{n=1}^{N} x_n^2, \tag{12}$$

where a value nearer 0 indicates either a more selective unit or a sparser population code. Using input data from the test set we activated the hidden units with the standard sigmoid activation function. We varied the degree of selectivity and sparsity by changing μ and measured the activity ratios in both the lifetime and population domains. As illustrated in Figure 6(a), as μ is decreased, both activity ratios also decrease. The properties of selectivity and sparsity in the latent activation biases are transferred to the learned parameters, which induces latent representations with those properties.

Benefits of manipulating both selectivity and sparsity. Biasing the RBM with only selectivity or only sparsity, we analyzed their activity ratios in relation to μ. Biasing only for selectivity results in activity ratios with a large spread in the population domain (Figure 6(b)). Likewise, biasing only for sparsity results in a high variations in the lifetime domain (Figure 6(c)). We found that this result is due to activations being consistently high or low in the non-biased domain. We refer to Figure 2 for a toy example of this phenomenon and Section 1.3 for a discussion. Increasing the selectivity and sparsity explicitly improves the diversity between hidden units in their own domains, but does not explicitly induce diversity in the other domain. Both selectivity and sparsity are required to reduce the number of overly active or silent units, and superfluously or inadequately represented examples.

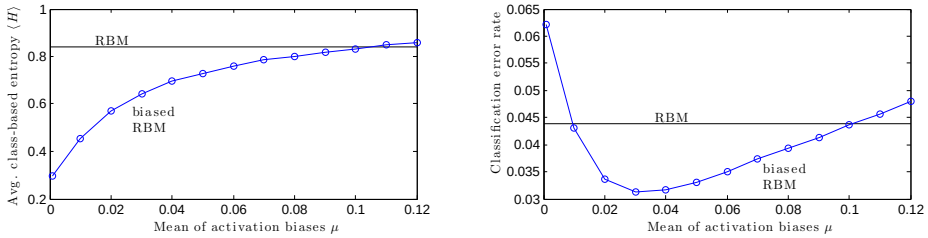


(a) Both selectivity and sparsity (b) Selectivity only (c) Sparsity only

Figure 6: Analysis of activity ratio with respect to the target mean μ of the activation biases. (a) Both activity ratios decrease in relation to μ, showing a transfer in representational properties from the biases to the actual latent activations. As a reference, the activity ratios for the standard RBM are plotted. (b) By biasing only for selectivity, the spread of activation ratios in the population domain is high. (c) In contrast, biasing only for sparsity results in high variations in lifetime activity ratios.

Biasing RBMs improves discriminative performance. While many deep learning methods use MNIST as a benchmarking data set, we adopt a shallow architecture to analyze the performance gain of a biased RBM over the standard RBM. For each hidden unit, the activation with respect to each class is totaled and normalized across classes. We then computed the Shannon entropy of each hidden unit and finally averaged it across the population. This metric $\langle H \rangle$ gives us an indication of the level of class-based discrimination of the hidden units, where lower $\langle H \rangle$ values signify fewer the number of classes each unit encodes. We also trained a simple multinomial logistic regression classifier from the activations of the hidden layer (without backpropogating the features) and computed the classification error rate. Since there are 10 classes, one for each digit and roughly uniformly distributed, we consider that for a unit to be selective, it should respond to less than 10% of the samples. Hence, we conducted our study in the range of $0.001 \leq \mu \leq 0.12$.

From Figure 7(a), we observe a clear monotonic relationship of $\langle H \rangle$ with respect to μ. When μ is lowered, a unit responds to fewer examples, so if examples from the same class have similar appearances, then it is more likely that these examples belong to the same class, thus lowering $\langle H \rangle$. Figure 7(b) shows that the relation between the classification error and μ is no longer monotonic. The model has poor generalization when μ nears 0 as units encode individual examples too specifically, just like "grandmother cells" [6]. For this data set, the biased RBM achieves better result than the standard RBM in the approximate range of $0.01 \leq \mu \leq 0.1$. At its minimum, the biased RBM significantly outperforms the standard RBM by 29% (1.25% improvement in error rate). Interestingly, the classification performance of this simple semi-supervised method without fine-tuning is comparable to other reported results using similarly simple yet completely supervised approaches [11].



(a) Class-based entropy (b) Classification error

Figure 7: Discriminative performance of RBMs biased with selectivity and sparsity. (a) $\langle H \rangle$ varies monotonically with μ. (b) Classification error is minimum when μ is low, but not at the lowest. There is a range of μ whereby biasing the RBM improves generalization performance.

5 Conclusions

Selectivity and sparsity are important properties of neural coding. In this paper, we introduced a modification to the RBM learning algorithm that can incorporate generic latent activation biases to guide the learning process. The RBM parameters are encouraged to encode the representational properties defined by these biases. The activation biases are designed based on prior knowledge of the task or nature of the problem. Here, we described the activation biases to manipulate latent selectivity and sparsity. We quantitatively verified that selectivity and sparsity were indeed encoded after training and presented the benefits of selectivity and sparsity for modeling handwritten digits. We are currently working on modeling other latent activation biases that are inspired from neural phenomenon, such as topographical organization [10] and explicitness [4].

Acknowledgments

We thank Joo-Hwee Lim for fruitful discussions. This work was supported in part by the Embassy of France in Singapore under the Merlion 2009 Project and PhD program.

References

[1] Blackwell, D. (1947). Conditional expectation and unbiased sequential estimation. *Ann. Math. Stat.*, 18.

[2] Doi, E., Inui, T., Lee, T.-W., Wachtler, T., & Sejnowski, T. J. (2003). Spatiochromatic receptive field properties derived from information-theoretic analyses of cone mosaic responses to natural scenes. *Neural Computation*, 15.

[3] Doi, E. & Lewicki, M. S. (2005). Sparse coding of natural images using an overcomplete set of limited capacity units. In *NIPS 17*.

[4] Földiák, P. (2009). Neural coding: non-local but explicit and conceptual. *Current Biology*, 19(19).

[5] Franco, L., Rolls, E. T., Aggelopoulos, N. C., & Jerez, J. M. (2007). Neuronal selectivity, population sparseness, and ergodicity in the inferior temporal visual cortex. *Biological Cybernetics*, 96(6).

[6] Gross, C. G. (2002). Genealogy of the "grandmother cell". *Neuroscientist*, (8).

[7] Hinton, G. (2010). *A practical guide to training restricted Boltzmann machines*. Technical Report UTML TR 2010–003, Department of Computer Science, University of Toronto.

[8] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural Computation*, 14(8).

[9] Hinton, G. E., Osindero, S., & Teh, Y.-W. (2006). A fast learning algorithm for deep belief networks. *Neural Computation*, 18(7).

[10] Hyvärinen, A. & Hoyer, P. O. (2001). A two-layer sparse coding model learns simple and complex cell receptive fields and topography from natural images. *Vision Research*, 41(18).

[11] LeCun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11).

[12] Lee, H., Ekanadham, C., & Ng, A. (2008). Sparse deep belief net model for visual area V2. In *NIPS 20*.

[13] Mitchell, T. M. (1980). *The Need for Biases in Learning Generalizations*. Technical Report CBM-TR-117, Departament of Computer Science, Rutgers University.

[14] Nair, V. & Hinton, G. (2009). 3D object recognition with deep belief nets. In *NIPS 22*.

[15] Olshausen, B. A. & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583).

[16] Ranzato, M., Poultney, C., Chopra, S., & LeCun, Y. (2007). Efficient learning of sparse representations with an energy-based model. In *NIPS 19*.

[17] Smolensky, P. (1987). Information processing in dynamical systems: Foundations of harmony theory. In *Parallel Distributed Processing: Volume 1: Foundations*.

[18] Swersky, K., Chen, B., Marlin, B., & de Freitas, N. (2010). A tutorial on stochastic approximation algorithms for training restricted boltzmann machines and deep belief nets. In *Information Theory and Applications (ITA) Workshop*.

[19] Teh, Y. W., Welling, M., Osindero, S., & Hinton, G. E. (2004). Energy-based models for sparse overcomplete representations. *Journal of Machine Learning Research*, 4.

[20] Treves, A. & Rolls, E. T. (1991). What determines the capacity of autoassociative memories in the brain? *Network: Computation in Neural Systems*, 2(4).

[21] Willmore, B. & Tolhurst, D. J. (2001). Characterizing the sparseness of neural codes. *Network: Computation in Neural Systems*, 12(3).