Tikhonov-Type Regularization for Restricted Boltzmann Machines

KyungHyun Cho, Alexander Ilin and Tapani Raiko

Department of Information and Computer Science
Aalto University School of Science, Finland
{firstname.lastname@aalto.fi}

Abstract. In this paper, we study a Tikhonov-type regularization for restricted Boltzmann machines (RBM). We present two alternative formulations of the Tikhonov-type regularization which encourage an RBM to learn a smoother probability distribution. Both formulations turn out to be combinations of the widely used weight-decay and sparsity regularization. We empirically evaluate the effect of the proposed regularization schemes and show that the use of them could help extracting better discriminative features with sparser hidden activation probabilities.

Keywords: Restricted Boltzmann Machine, Tikhonov Regularization

1 Introduction

Restricted Boltzmann machines (RBM) play an important role in deep learning. In many deep neural networks each layer of the network is pre-trained as if it were an RBM, and it has been empirically shown to facilitate training the whole network (see, e.g., [10, 8]).

It is common to use a stochastic gradient method for training RBMs. Both contrastive divergence learning [12] and approximate maximum-likelihood learning (see, e.g., [21]), two of the most popular learning methods, are based on the stochastic gradient method.

One important research direction in using the stochastic gradient method for RBMs is to design a regularization term. For instance, one of the most naive, but widely-used, regularization methods called weight-decay regularizes the growth of parameters in order to avoid overfitting and to stabilize learning. Another completely different regularization technique introduced in [14] forces learning to result in an RBM that gives sparser hidden activations given visible data.

Along this line of research, we investigate a Tikhonov-type regularization (see, e.g., [1, 9]) by which we refer to regularizing the derivative of either an approximating function or a function related to it. In this paper, two different formulations of the Tikhonov-type regularization for RBMs are derived. We found that both formulations appear as a combination of weight-decay and sparsity regularization, and present an empirical evaluation on their effect in training RBMs.

Recently, a form of Tikhonov-type regularization was successfully applied to auto-encoders, which are closely related to RBMs [22], in [18, 17] to explicitly encourage hidden variables to be invariant to (small) deformation of input representations. It was done by regularizing the squared derivative of a latent variable with respect to an input variable, which makes it a modified form of Tikhonov-type regularization.

2 Restricted Boltzmann Machines

The restricted Boltzmann machine is a stochastic neural network with a bipartite structure such that each visible neuron is connected to all the hidden neurons and each hidden neuron is connected to all the visible ones [20].

We define a log-probability assigned to a given visible vector $\mathbf{v}$ by an RBM as:

$$\log p(\mathbf{v} \mid \boldsymbol{\theta}) = f(\mathbf{v} \mid \boldsymbol{\theta}) + \sum_{j=1}^{N_h} \log \left(1 + \exp \left(c_j + \sum_{i=1}^{N_v} w_{ij} \frac{v_i}{\sigma^2} \right) \right) - \log Z(\boldsymbol{\theta}), \quad (1)$$

where $\mathbf{v}$ and $\mathbf{h}$ are a column vector and a binary column vector representing the state of the visible and hidden neurons, and parameters $\boldsymbol{\theta} = (\mathbf{W}, \mathbf{b}, \mathbf{c}, \boldsymbol{\sigma})$ include weights $\mathbf{W} = [w_{ij}]_{N_v \times N_h}$, biases $\mathbf{b} = [b_i]_{N_v \times 1}$, $\mathbf{c} = [c_j]_{N_h \times 1}$ and standard-deviations $\boldsymbol{\sigma} = [\sigma_i]_{N_v \times 1}$. N_v and N_h are the numbers of visible and hidden neurons, respectively. $Z(\boldsymbol{\theta})$ denotes the normalizing constant which is intractable and it can be calculated by summing exponentially many terms.

Function $f(\mathbf{v} \mid \boldsymbol{\theta})$ in (1) indicates a contribution of visible neurons' biases to an energy of an RBM. $f(\mathbf{v} \mid \boldsymbol{\theta})$ together with σ_i decides whether a visible neuron may have a binary value or a continuous real value.

When $f(\mathbf{v} \mid \boldsymbol{\theta}) = \mathbf{b}^\top \mathbf{v}$, it requires a visible neuron to have either 0 or 1, making a standard binary RBM [20]. In this case each σ_i is set to 1. On the other hand, each visible neuron can have a continuous real value when $f(\mathbf{v} \mid \boldsymbol{\theta}) = -\sum_{i=1}^{N_v} \frac{(v_i - b_i)^2}{2\sigma_i^2}$, and each σ_i can either be set to a pre-defined value or learned [2, 10]. We call this model a Gaussian-Bernoulli RBM (GRBM).

Given a training set $\{\mathbf{v}^{(n)}\}_{n=1}^{N}$ an RBM can be trained by maximizing log-likelihood. The maximization is usually done by the stochastic gradient method, and in this paper, we use the recently introduced method of the enhanced gradient [4] together with parallel tempering [3, 7].

2.1 Regularization for Restricted Boltzmann Machines

There are a number of regularization techniques that are widely used.

One most widely used technique is *weight-decay* regularization. Training an RBM with the weight-decay regularization maximizes the following objective function:

$$\mathcal{L}(\boldsymbol{\theta}) - \frac{\beta_w}{2} \sum_{ij} w_{ij}^2, \quad (2)$$

where $\mathcal{L}(\boldsymbol{\theta})$ and β_w are the log-likelihood function and the regularization constant, respectively [1].

[1] The weight-decay may be applied to visible and hidden biases, as we have done in this paper.

Another widely used technique, called *sparsity regularization*, regularizes the average activation probability of each hidden neuron. An RBM trained using the sparsity regularization is commonly referred to as sparse RBM (sRBM) [14]. Sparse RBMs have been popular due to the fact that an RBM with low average hidden activation probabilities can extract better discriminative features than non-regularized RBMs (see, e.g., [16, 5]). In [14], the sRBM was introduced by modifying the objective function to

$$\mathcal{L}(\boldsymbol{\theta}) - \frac{\beta_s}{2} \sum_{j=1}^{N_h} \left(\rho - \frac{1}{N} \sum_{n=1}^{N} p(h_j \mid \mathbf{v}^{(n)}, \boldsymbol{\theta}) \right)^2 , \tag{3}$$

where ρ and β_s are a target average activation of each hidden neuron and the regularization constant, respectively.

3 Tikhonov Regularization for Restricted Boltzmann Machines

In this section, we present two possible formulations of the Tikhonov-type regularization for RBMs. We refer to them as **TYPE-1** and **TYPE-2** formulations, respectively.

3.1 TYPE-1 and TYPE-2 Regularizations

One basic approach of the Tikhonov-type regularization is to minimize

$$\frac{\beta}{2} \mathbb{E}_{p(\mathbf{v})} \left[\|\nabla_{\mathbf{v}} y(\mathbf{v})\|^2 \right] , \tag{4}$$

when the task is to approximate some function $y(\mathbf{v})$ (see, e.g., [1, 9]) of inputs $\mathbf{v}$. Here, β is a regularization parameter. $p(\mathbf{v})$ can be defined by a set of training samples or be approximated by a probabilistic model.

Intuitively, by minimizing the derivative of the approximating function, Eq. (4) keeps the function smooth around training samples or around regions of high probability. In other words, it makes function $y(\mathbf{v})$ more invariant to (small) deformations of $\mathbf{v}$.

Under this intuition, it is natural to use as the approximating function $y(\mathbf{v})$ the probability density function $p(\mathbf{v})$ learned by an RBMs. Thus, the RBM model distribution is regularized to be smoother.

After replacing $y(\mathbf{v})$ with Eq. (1), we get the following **TYPE-1** regularization term:

$$J_1 = \frac{\beta}{2} \mathbb{E}_{p(\mathbf{v})} \left[\sum_{i=1}^{N_v} \left(\frac{\partial}{\partial v_i} \log p(\mathbf{v} \mid \boldsymbol{\theta}) \right)^2 \right] \approx \frac{\beta}{2N} \sum_{n=1}^{N} \sum_{i=1}^{N_v} \left(\frac{\partial f(\mathbf{v}^{(n)} \mid \boldsymbol{\theta})}{\partial v_i} + \sum_{j=1}^{N_h} \frac{w_{ij}}{\sigma_i^2} h_j^{(n)} \right)^2 ,$$

where $\mathbf{v}^{(n)}$ is either an empirical sample or a sample drawn from the model distribution[2] and $h_j^{(n)}$ is a short-hand notation for $p(h_j = 1 \mid \mathbf{v}^{(n)}, \boldsymbol{\theta})$.

[2] In the experiments, we used the samples from the model distribution which are readily available when computing the gradients.

Instead, we may formulate the Tikhonov-type regularization by regularizing the derivative of another function. From Eq. (1) it is apparent that an RBM is a special-case of product-of-experts models [12], which implies that a probability given to $\mathbf{v}$ by an RBM consists of contributions from experts which, in the case of RBMs, are hidden neurons. Hence, it is reasonable to regularize each contribution of a hidden neuron by minimizing the derivative of the logarithmic conditional probability distribution of each hidden neuron $\log p(h_j \mid \mathbf{v}, \boldsymbol{\theta})$[3].

The **Type-2** Tikhonov regularization is then formulated to minimize the following term:

$$J_2 = \frac{\beta}{2} \mathbb{E}_{p(\mathbf{v})} \left[\sum_{i=1}^{N_v} \sum_{j=1}^{N_h} \left(\frac{\partial}{\partial v_i} \log h_j^{(n)} \right)^2 \right] \approx \frac{\beta}{2N} \sum_{n=1}^{N} \sum_{i=1}^{N_v} \sum_{j=1}^{N_h} \left(\frac{w_{ij}}{\sigma_i^2} h_j^{(n)} \right)^2 . \quad (5)$$

In the case of a standard binary RBM, it is easy to see that the derivatives in both formulations are not well-defined as $\mathbf{v}$ is a binary vector. However, we can simply assume that $p(\mathbf{v} \mid \boldsymbol{\theta})$ has a domain of $\mathbb{R}^{N_v}$ instead, which is obviously followed by $p(h_j \mid \mathbf{v}, \boldsymbol{\theta})$ having the same domain[4].

The idea behind this choice is that a probability distribution defined by a binary RBM is constructed by taking values of all $\mathbf{v}$ such that each component of $\mathbf{v}$ is restricted to be either 0 or 1. Hence, we make the distribution defined by the RBM smoother by smoothing another continuous distribution with the same probability density function.

It is easy to see that both **Type-1** and **Type-2** can be seen as a combination of the weight-decay and sparsity regularizations. Both terms decrease when the absolute l^2-norm of each weight and the average activation probability of each hidden neuron decrease.

3.2 Optimization

A straightforward way to train an RBM with one of the two types of the Tikhonov-type regularization is to optimize the regularization term together with the log-likelihood. However, this approach makes it difficult to utilize the enhanced gradient which has been shown to perform better than the traditional gradient [4]. Hence, we follow the approach introduced in [14]. At each iteration, following the normal stochastic update of the parameters using the enhanced gradient we update the parameters w_{ij}, b_i and c_j again according to one of the regularization terms computed with the current minibatch.

4 Experiments

In this section we try to see the effect of the proposed regularization. From here on we refer to an RBM trained using the proposed Tikhonov regularization as a regularized RBM (rRBM). Note that in this paper we only focus on a standard RBM which constraints each visible neuron to be binary. *rRBMt1* and *rRBMt2* indicate the **Type-1**

[3] As noted earlier, a similar idea has been applied to auto-encoders in [18, 17].

[4] This assumption does not need to be made in case of RBMs with continuous visible neurons.

and **Type-2** regularization, respectively. Additionally, we tested the weight-decay and sparsity regularization techniques in order to see how they perform differently compared to the proposed Tikhonov-type regularization. They are denoted by *wRBM* and *sRBM*, respectively.

We take a look at three metrics that can explain the effect of the proposed regularization term. We trained RBMs with 500 hidden neurons on two different data sets which are the handwritten digits (MNIST) [13] and the Caltech-101 Silhouettes [15]. As they have been quite well studied previously, we can easily compare to results obtained by other researchers.

Firstly, log-probabilities of test samples are checked. It may happen that the log-probabilities become larger for the rRBMs, as smoothing could potentially decrease the peaks around training samples systematically resulting in higher probability being assigned to nearby test samples.

Secondly, we consider classification accuracies using the learned features from an RBM, which indirectly suggests how discriminative extracted features are. In order to see how discriminative features were, we did not fine-tune the already trained weights of RBMs.

Additionally, in order to see how the proposed scheme biases a resulting model we check the average hidden activation probabilities given test samples. It can be expected that rRBMs will achieve higher sparsity.

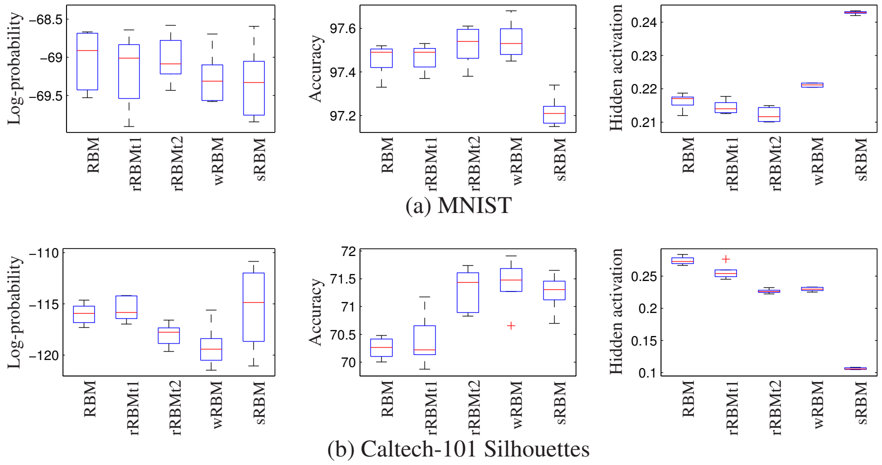
For each data set we chose a regularization constant β through validation. We grid-searched from 2^{-8} to 2^{-20} and estimated log-probabilities of validation samples. For each grid point five RBMs were trained for a small number of epochs with different random initializations, we considered their medians. Starting from a large β we logarithmically decreased it until the log-probabilities of validation samples stopped increasing or decreasing significantly. Then, we chose the largest β with the converged performance and sparsity.

We followed the same validation strategy to choose β_w for RBMs trained using the weight-decay regularization. For sparse RBMs, we chose the target sparsity ρ through validation. ρ was grid-searched from 2^{-1} to 2^{-8}, and ρ with the best log-probabilities was chosen. The regularization constant β_s was fixed to the inverse of the target sparsity, as recommended by [14].

Finally, we chose 2^{-16} and 2^{-17} for MNIST with the **Type-1** and **Type-2**, respectively. For Caltech-101 Silhouettes, 2^{-17} was chosen for both formulations of the Tikhonov-type regularization. β_w for the weight-decay was chosen to be 2^{-14} and 2^{-11} for MNIST and Caltech-101 Silhouettes. 2^{-2} and 2^{-5} were chosen to be the target sparsity ρ for MNIST and Caltech-101 Silhouettes, respectively.

For initializing parameters, we followed the strategy recommended in [11]. Each weight w_{ij} was drawn from a zero-mean normal distribution with its variance $\frac{1}{\sqrt{N_v+N_h}}$. Visible biases $\mathbf{b}$ were set according to the training samples, and hidden biases $\mathbf{c}$ were initialized to negative values (-4) in order to encourage sparse hidden activation probabilities.

We independently trained RBMs five times with different parameters initializations. Each RBM was trained for 200 epochs and 3000 epochs for MNIST and Caltech-101 Silhouettes, which amount to about 93,800 and 99,000 updates, respectively.



(a) MNIST

(b) Caltech-101 Silhouettes

Fig. 1. Log-probabilities, classification accuracies and average hidden activation probabilities of test samples computed from the RBMs trained on MNIST and Caltech-101 Silhouettes with the proposed Tikhonov regularization schemes (*rRBMt1* and *rRBMt2*) and without it (*RBM*).

Log-probabilities were computed using a normalizing constant estimated using the AIS [19]. A simple logistic classifier was trained on hidden activation probabilities to compute classification accuracies. We used parallel tempering to sample from the model distribution [7, 3], and used the enhanced gradient and the adaptive learning rate, with both an initial learning rate and an upper-bound set to 0.1, proposed in [2]. For each experiment we decreased the learning rate proportionally to the inverse of the number of updates for the last half of training.

4.1 Result

In Fig. 1, we see the log-probabilities and the classification accuracies of the test samples and the average activation probabilities of the hidden neurons given the test samples.

The most obvious difference between the non-regularized RBM and the rRBMs is the lower average hidden activation probabilities given test samples [5]. As discussed previously the proposed regularization schemes resulted in a model with sparser hidden activation probabilities. It is also noticeable that **TYPE-2** tends to bias a resulting model to have sparser hidden activation probabilities even compared to the RBMs trained using the **TYPE-1** regularization or the RBMs trained with the weight-decay.

A general trend of extracting better discriminative features could be observed when the RBMs were regularized with either the **TYPE-1** or **TYPE-2** schemes. It was especially obvious with the **TYPE-2** regularization while the use of the **TYPE-1** formulation gave only marginal improvement over the non-regularized RBMs.

[5] Inconsistently high or low average hidden activation probabilities achieved by the sparse RBMs are due to the fact that the target sparsity ρ was chosen by the validation to be as high as $2^{-2} = 0.25$ for MNIST and as low as $2^{-5} = 0.0312$ for Caltech-101 Silhouettes.

On the other hand, it could be observed that the proposed regularization schemes were not able to improve the resulting models' generative performance. In the case of MNIST, it could be seen that the better discriminative performance was achieved with slight degradation in the log-probabilities of test samples. However, in the case of Caltech-101 Silhouettes, we could observe that better generative models were learned using the **TYPE-1** scheme. It indirectly suggests that smoothing the overall probability distribution (1) could potentially improve the generalization of the model by removing highly peaked probability mass on training samples, while smoothing a contribution from each hidden neuron does not necessarily help.

5 Discussion

We have presented two possible types of the Tikhonov-type regularization for training RBMs in this paper. Both the **TYPE-1** and **TYPE-2** schemes prefer an RBM to learn a smoother probability distribution by minimizing the derivatives of either log-probability distribution or log-conditional distribution of hidden neurons. It was shown that both types were formulated as a combination of the weight-decay and sparsity regularizations which are widely used when training RBMs.

The experiments showed that both types were able to extract better discriminative features with sparser hidden activation probabilities while marginally sacrificing the generative capability of the resulting RBMs. The trend was more visible with the **TYPE-2** scheme, while it was not so apparent with the **TYPE-1** regularization. However, we were not able to see any significant performance improvement over other conventional regularization techniques with binary RBMs.

We noticed through the validation step of the experiments that the regularization constant β needs to be carefully chosen. Too large β overly simplified the distribution learned by an RBM and failed to give a good fit of a training distribution. More thorough investigation in choosing an appropriate regularization constant will need to be done. Regardlessly, the proposed formulations of the Tikhonov regularization reduce the number of hyper-parameters from at least three (ρ, β_s, and β_w) to one (β) against using both the weight-decay and the sparse regularization together.

We tested the proposed regularization schemes with a standard, binary RBM only. Both schemes, however, are not restricted to an RBM, but applicable to any other variant that can explicitly sum out hidden variables. One such variant is a GRBM which replaces a binary visible neuron with a continuous, real-valued visible neuron. Another possibility is a recently introduced spike-and-slab RBM [6]. It is natural to test the proposed method with this model as a next step in future work.

References

1. Bishop, C.M.: Pattern Recognition and Machine Learning (Information Science and Statistics). Springer, 1st ed. 2006. corr. 2nd printing edn. (Oct 2007)
2. Cho, K., Ilin, A., Raiko, T.: Improved Learning of Gaussian-Bernoulli Restricted Boltzmann Machines. In: Proceedings of the Twentith International Conference on Artificial Neural Networks. ICANN 2011 (2011)

3. Cho, K., Raiko, T., Ilin, A.: Parallel tempering is efficient for learning restricted boltzmann machines. In: Proceedings of the International Joint Conference on Neural Networks (IJCNN 2010). Barcelona, Spain (July 2010)
4. Cho, K., Raiko, T., Ilin, A.: Enhanced Gradient and Adaptive Learning Rate for Training Restricted Boltzmann Machines. In: Proceedings of the Twenty-seventh International Conference on Machine Learning. ICML 2011 (2011)
5. Coates, A., Lee, H., Ng, A.Y.: An analysis of single-layer networks in unsupervised feature learning. In: AISTATS (2011)
6. Courville, A., Bergstra, J., Bengio, Y.: Unsupervised models of images by spike-and-slab rbms. In: Getoor, L., Scheffer, T. (eds.) Proceedings of the 28th International Conference on Machine Learning (ICML-11). pp. 1145–1152. ICML '11, ACM, New York, NY, USA (June 2011)
7. Desjardins, G., Courville, A., Bengio, Y., Vincent, P., Delalleau, O.: Parallel Tempering for Training of Restricted Boltzmann Machines. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. pp. 145–152 (2010)
8. Erhan, D., Bengio, Y., Courville, A., Manzagol, P.A., Vincent, P., Bengio, S.: Why does unsupervised pre-training help deep learning? Journal of Machine Learning Research (2010)
9. Haykin, S.: Neural Networks: A Comprehensive Foundation (2nd Edition). Prentice Hall, 2 edn. (July 1998)
10. Hinton, G.E., Salakhutdinov, R.R.: Reducing the Dimensionality of Data with Neural Networks. Science 313(5786), 504–507 (July 2006)
11. Hinton, G.: A Practical Guide to Training Restricted Boltzmann Machines. Tech. rep., Department of Computer Science, University of Toronto (2010)
12. Hinton, G.E.: Training products of experts by minimizing contrastive divergence. Neural Comput. 14, 1771–1800 (August 2002)
13. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-Based Learning Applied to Document Recognition. In: Proceedings of the IEEE. vol. 86, pp. 2278–2324 (1998)
14. Lee, H., Ekanadham, C., Ng, A.: Sparse deep belief net model for visual area V2 pp. 873–880 (2008)
15. Marlin, B.M., Swersky, K., Chen, B., de Freitas, N.: Inductive Principles for Restricted Boltzmann Machine Learning. In: Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics. pp. 509–516 (2010)
16. Ngiam, J., Khosla, A., Kim, M., Nam, J., Lee, H., Ng, A.Y.: Multimodal deep learning. In: International Conference on Machine Learning (ICML). Bellevue, USA (June 2011)
17. Rifai, S., Dauphin, Y.N., Vincent, P., Bengio, Y., Muller, X.: The manifold tangent classifier. In: Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., Weinberger, K. (eds.) Advances in Neural Information Processing Systems 24, pp. 2294–2302 (2011)
18. Rifai, S., Vincent, P., Muller, X., Glorot, X., Bengio, Y.: Contractive auto-encoders: Explicit invariance during feature extraction. In: Getoor, L., Scheffer, T. (eds.) Proceedings of the 28th International Conference on Machine Learning (ICML-11). pp. 833–840. ICML '11, ACM, New York, NY, USA (June 2011)
19. Salakhutdinov, R.: LEARNING DEEP GENERATIVE MODELS. Ph.D. thesis, University of Toronto (2009)
20. Smolensky, P.: Information processing in dynamical systems: foundations of harmony theory. In: Parallel distributed processing: explorations in the microstructure of cognition, vol. 1: foundations, pp. 194–281. MIT Press, Cambridge, MA, USA (1986)
21. Tieleman, T.: Training restricted Boltzmann machines using approximations to the likelihood gradient. In: Proceedings of the 25th international conference on Machine learning. pp. 1064–1071. ICML '08, ACM, New York, NY, USA (2008)
22. Vincent, P.: A connection between score matching and denoising autoencoders. Neural Computation 23(7), 1661–1674 (Jul 2011)