Artificial Neural Networks and Deep Architectures, DD2437

Short report on lab assignment 4

Deep Belief Nets (DBNs)

Saly Al Masri, Widad Farhat and Xiaolei Liu

September 1, 2018

1 Main Objectives and Scope of the Assignment

Our major goals in the assignment were

- to implement and train a Restricted Boltzmann Machine (RBM) for image reconstruction,
- to analyze the impact of hidden unit variations on reconstruction loss and stability,
- to extend RBMs into a Deep Belief Network (DBN) for classification and generative modeling.

The scope of this lab was limited to unsupervised pretraining using contrastive divergence, without fine-tuning. Assumptions included fixed hyperparameters and the MNIST dataset as the primary benchmark for evaluation.

2 Methods

The assignment was implemented using Python, utilizing NumPy for numerical computations and Matplotlib for visualization. The MNIST dataset served as the benchmark, and contrastive divergence (CD-1) was employed for training the RBM and DBN models. Reconstruction loss was used to assess model stability, while classification accuracy evaluated DBN performance. No fine-tuning was applied, and all models were trained with fixed hyperparameters.

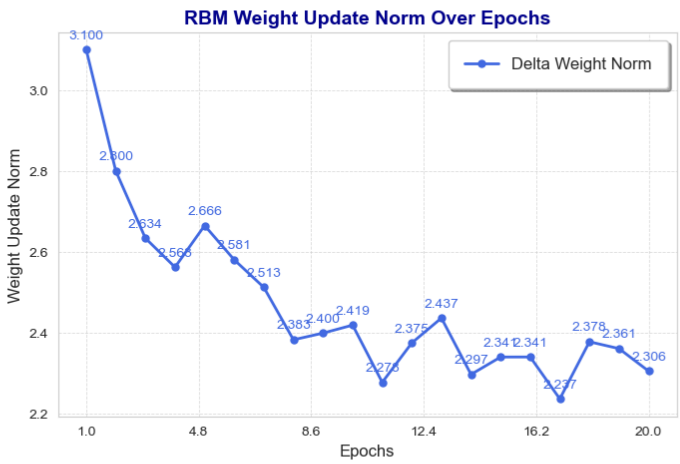
3 Results and discussion - Part I: RBM for recognising MNIST images

3.1 Monitoring and Measuring Stability in RBM Training

Reconstruction Loss as a Stability Indicator A well-trained RBM typically exhibits a rapid decrease in reconstruction loss during the early stages of training, followed by a gradual reduction until the loss plateaus. This stabilization suggests that the model has reached a steady state.

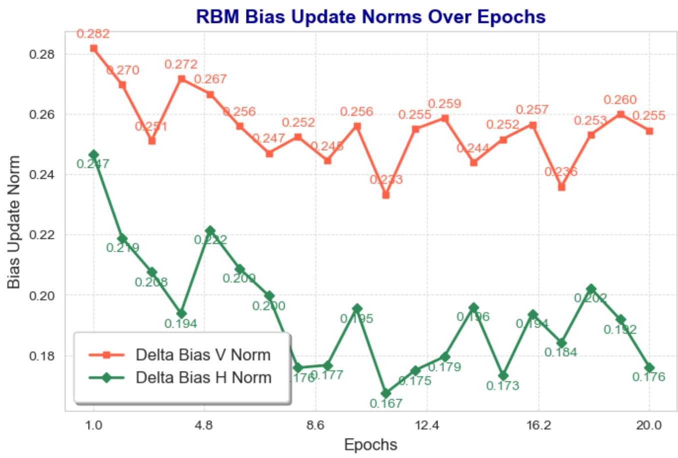
Weight and Bias Update Norms as Stability Metrics The graphs illustrate the evolution of weight and bias update norms across training epochs, offering valuable insights into RBM stability.

Weight Update Norms The weight update norm initially starts high (around 3.1) and gradually decreases, stabilizing around 2.3 after 20 epochs. The initial rapid decline followed by smaller fluctuations suggests that the weight updates are converging, indicating learning stabilization.



Figur 1: Enter Caption

Bias Update Norms The visible bias update norm starts around 0.28, fluctuates slightly, and stabilizes near 0.25. The hidden bias update norm begins at 0.22 and decreases to approximately 0.176, showing minor fluctuations. Although the bias updates exhibit more variability than weight updates, they still follow an overall decreasing trend, confirming that the biases are also stabilizing.

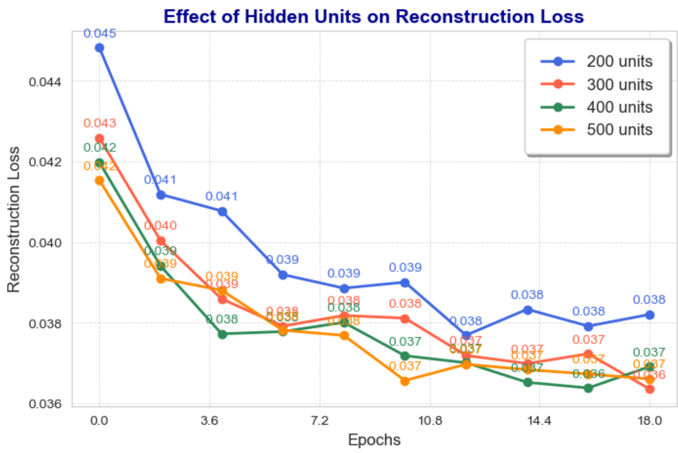


Figur 2: Bias updates over Epoches

3.2 Effect of Hidden Units on Reconstruction Loss

The number of hidden units in a Restricted Boltzmann Machine (RBM) significantly affects its ability to learn and reconstruct input data. The graph illustrates how reconstruction loss changes when the number of hidden units decreases from **500 to 200**, providing clear numerical evidence of the impact on performance. The results show that 500 hidden units consistently produce the lowest reconstruction loss, demonstrating the best capability to learn and represent the input data. As the number of hidden units is reduced to 400, the performance remains similar but with slightly higher loss values. When the hidden units are further decreased to 300, there is a moderate increase in reconstruction loss, and with 200 hidden units, the loss is at its highest, indicating a greater struggle in reconstructing the input accurately.

Across all configurations, reconstruction loss decreases sharply in the first few epochs and stabilizes around epoch 10–15 with small fluctuations. However, the model with 200 hidden units exhibits greater instability, suggesting a less reliable convergence. At the initial epoch, the reconstruction loss is highest for 200 hidden units (0.045) and lowest for 500 hidden units (0.042). By epochs 10–18, the loss stabilizes at approximately 0.037 for 500 hidden units and 0.038 for 200 hidden units, highlighting a persistent performance gap. The model with 200 hidden units shows a slower and less stable decline in reconstruction loss, reinforcing the idea that reducing hidden units limits learning capacity.



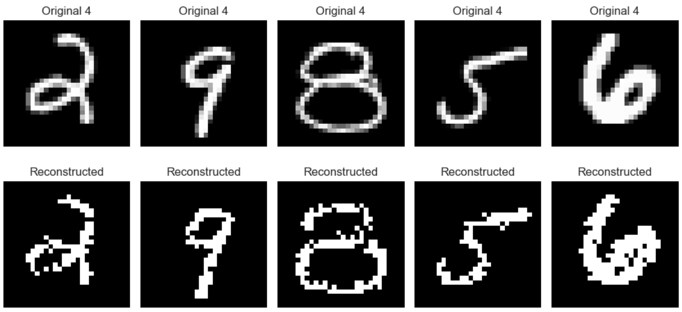
Figur 3: Hidden Units

The findings confirm that reducing the number of hidden units from 500 to 200 results in increased reconstruction loss and degraded model performance. While all models eventually stabilize, those with fewer hidden units show higher reconstruction errors and greater fluctuations. These observations underscore the importance of selecting an appropriate number of hidden units to ensure efficient learning and stable convergence in RBMs.

3.3 Hidden Unit Activations and Feature Learning

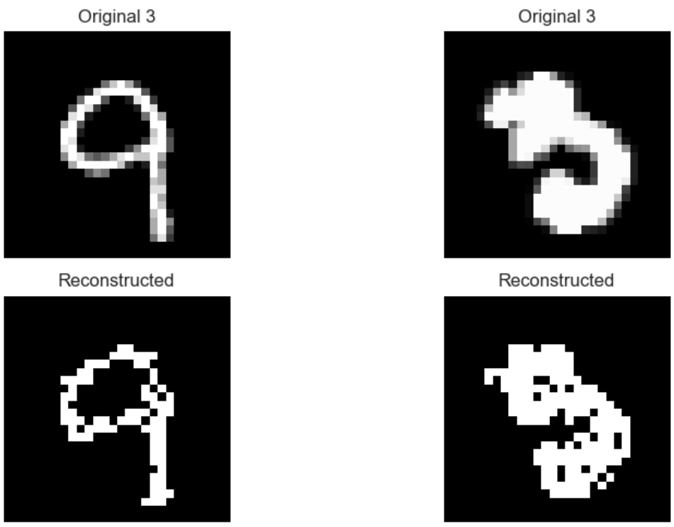
To understand how hidden units process input data, activations were analyzed for a batch of 10 MNIST test images. The results indicate that most hidden units activate across multiple images, confirming that the RBM extracts meaningful features consistently. Some units rarely activate, suggesting specialization in detecting certain patterns such as curves, strokes, and digit edges.

To verify feature learning, receptive fields, which represent the weights connecting hidden to visible units, were examined. Before training, the weights appeared random with no meaningful structure. After training, they transformed into recognizable digit components, including loops for digits like '0, '6,' and '9,' vertical strokes for '1,' and curved edges for other numbers. Some receptive fields exhibited localized activation, indicating that hidden units learn to specialize in specific digit segments rather than full digits.



Figur 4: Enter Caption

To quantitatively assess reconstruction quality, Mean Squared Error (MSE), Structural Similarity Index (SSIM), and Pixel Difference percentage were computed for two samples. Sample 1 had an MSE of 0.03747, an SSIM of 0.7352, and a pixel difference of 5.3181%, while Sample 2 had an MSE of 0.05458, an SSIM of 0.6783, and a pixel difference of 7.4875%. Lower MSE values indicate better reconstruction quality, with Sample 1 performing better than Sample 2. Higher SSIM values indicate greater structural similarity to the original images, reinforcing that Sample 1 outperforms Sample 2. Similarly, lower pixel differences suggest fewer visual discrepancies, with Sample 1 exhibiting a smaller reconstruction error than Sample 2.



Figur 5: Enter Caption

These findings confirm that quantitative metrics effectively measure reconstruction fidelity, while receptive fields and hidden activations provide insights into how the RBM learns and represents features. Improving training strategies or adjusting hidden unit configurations could further enhance reconstruction performance.

4 Results and discussion - Part II: Towards deep networks - greedy layer-wise pretraining

4.1 Implementation Details

In the construction of deep belief network, one fairly confusing part is to determine the usage of binary samples and probabilistic representations relatively. In this section we aim at a relief explanation on how and in which situation we choose from the two.
During the cd1 training process of RBMs, the two-round gibson sampling is applied. In the first round, we sample the hidden nodes for the following calculation, while other midterm variants are represented in probability, and this also applies to the loss calculation. When pre-training in DBN a greedy layer-wise way, we always apply the binary samples of the output of last layer as the input of the next layer.
Concerning the discrimination process of DBN, we apply binary samples in all of the stages, which by experiments yields the best classification accuracy than other settings.

4.2 DBN pre-training

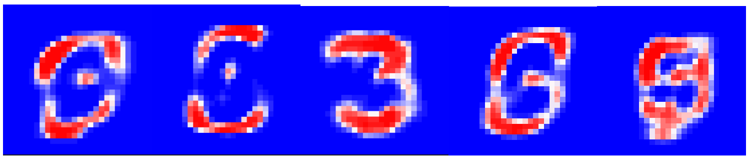
In the training of DBN, we totally follow the parameter settings in the original code, which includes training epochs, learning rate and momentum etc. The reconstruction loss on full trainset is followed. It is worth noting that in the

layer/epochs	base	midium	top
1	3086.13	2701.79	2640.77
5000	896.46	1801.19	1756.66
10000	796.87	1735.68	1502.68

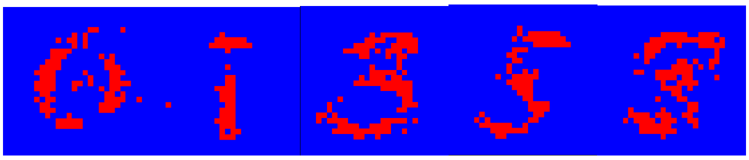
higher layers the descent of loss is not as ideal as in base layer, which raises concerns on the stability when stacking even more layers, especially taking into consideration that the distortion caused by non-optimal transformation of the formal layers may be accumulated and amplified in deep architecture.
After training, the train and test accuracy of the whole network are tested and the results are 84.28% and 84.21% respectively. There is no obvious difference in accuracy in training and test dataset, which implies the generalisation of model is fine. It is also worth noting that this performance is highly depended on the implementation of recognition method, and introducing probabilistic representation may actually harm the accuracy.

4.3 DBN generating

To generate images of assigned labels, we apply two different strategies and observe the difference. The first applies probabilistic representations all way through the Gibson sampling and top-down generating process, while the second binary samples. Due to the technical difficulties adding .gif files to latex, only static images are shown here. the .gif files will be covered in the presentation.



(a) 0 in prob. (b) 1 in prob. (c) 3 in prob. (d) 5 in prob. (e) 8 in prob.



(f) 0 in bin. (g) 1 in bin. (h) 3 in bin. (i) 5 in bin. (j) 8 in bin.

Figur 6: probabilistic or binary setting denotes the type of representation used in generating process

The difference between two settings is obvious. When applying probabilistic representations, the patterns always converge quickly. However, the danger of stucking in fake patterns is prominent. On the contrary, the random nature of binary samples makes the figure hard to converge, while also lead them out from spurious patterns. This phenomenon is best shown through the contraction between figures of number 1 in the two settings, where the spurious attractor is actually an 'average' or a prototype of all numbers.

5 Conclusion

This study explored the training and application of Restricted Boltzmann Machines (RBMs) and Deep Belief Networks (DBNs) for image recognition and generative modeling. Through the implementation of an RBM, we analyzed reconstruction loss as a measure of stability and observed the impact of reducing hidden units on performance. The results indicated that a larger number of hidden units led to lower reconstruction loss and better feature representation.

In extending RBMs to DBNs, we demonstrated the effectiveness of greedy layer-wise pretraining in improving classification accuracy. The DBN achieved a stable generalization performance, reinforcing the value of unsupervised pretraining. Furthermore, the generative capabilities of DBNs were examined, revealing that probabilistic representations facilitate smooth convergence but may lead to spurious patterns, whereas binary sampling maintains randomness at the cost of slower convergence.