Deep Learning with Stacked AEs & RBMs

DD2437 - Artificial Neural Networks & Deep Architectures - Lab 4

Niels Agerskov agerskov@kth.se
Lukas Bjarre lbjarre@kth.se
Gabriel Carrizo gabcar@kth.se

This lab will examine two different artificial neural network structures, Auto Encoders (AE) and Restricted Boltzmann Machines (RBM). Their effectiveness in a learning task and the effect of the layer depth of the models will be tested and evaluated.

1 Feature learning

In this first task shallow versions of both models are trained as benchmarks for the later deeper versions. The dataset used is a subset of the MNIST dataset containing 28×28 images of handwritten digits from 0 to 9 together with correct labels of the written digit. All the pixel values has for simplicity's sake been converted to binary values via simple thresholding. A total of 10000 images are used from the dataset, which has been further subdivided into a training set of size 8000 and a validation set of size 2000.

1.1 Hidden unit size

The input size hyperparameter for both of the models is decided by the shape of the data. In our case we require $28 \times 28 = 784$ input nodes, one for each image pixel. We do however have a choice in the number of hidden units, n_{h}.

Both models were trained with $n_{\text{h}} = 50, 75, 100, 150$ hidden units. The error curves on the validation set during the training are displayed in fig. 1 for the RBM, and in fig. 2 for the AE using Stochastic Gradient Descent (SGD). However, the AE clearly converges to the same values no matter the number of hidden units. Complementary to using SGD to train the AE ADADELTA was also used, which error curve can be seen in fig. 3. ADADELTA shows a similar improvement given more hidden units as the error curves for the RBM. The errors are also on one order of magnitude smaller compared to the errors using SGD, which is why the ADADELTA trained AE is only used furthermore.

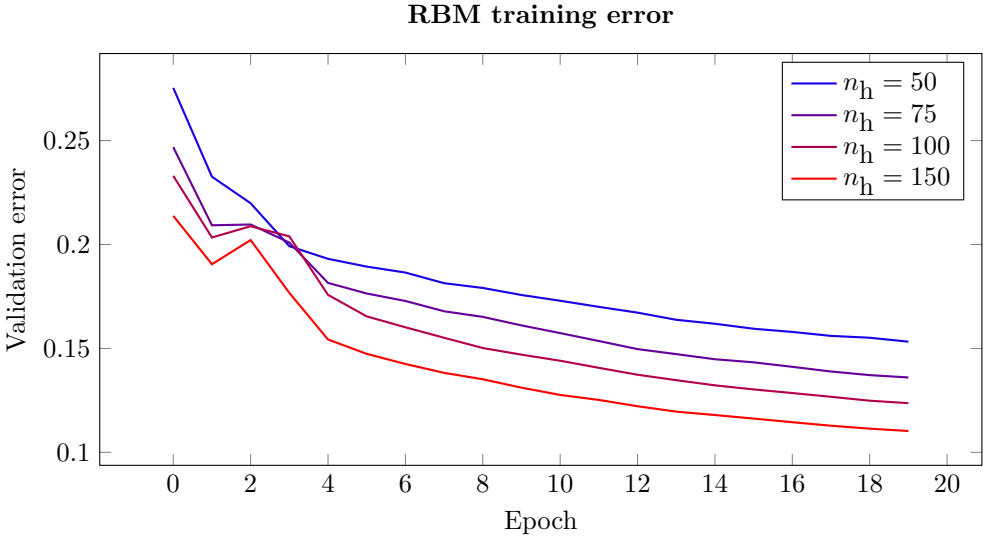


Figure 1: Error curves on the validation set for the RBM.

The quality of the models can be seen in fig. 4, where one image of each class have been used to get both models recalled versions.

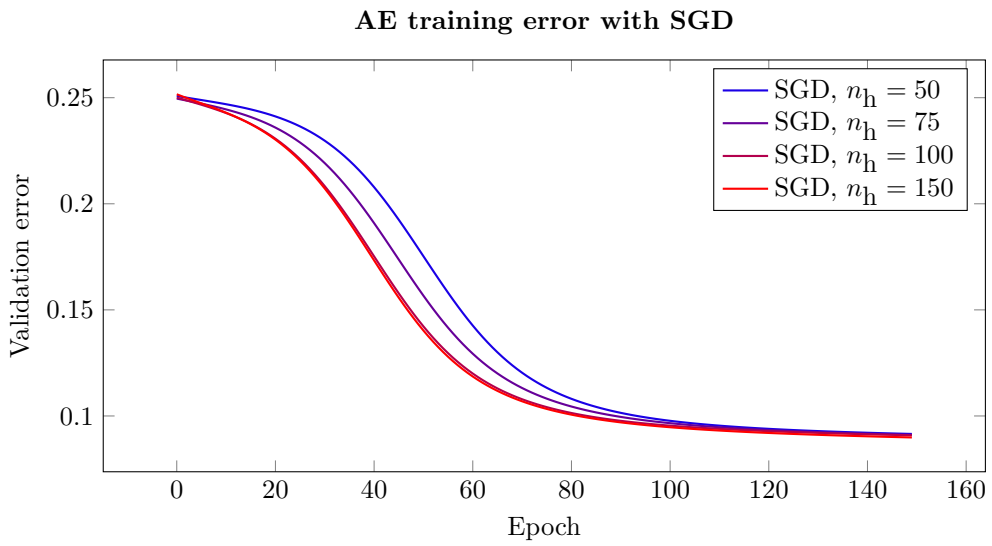


Figure 2: Error curves on the validation set for the AE using SGD.

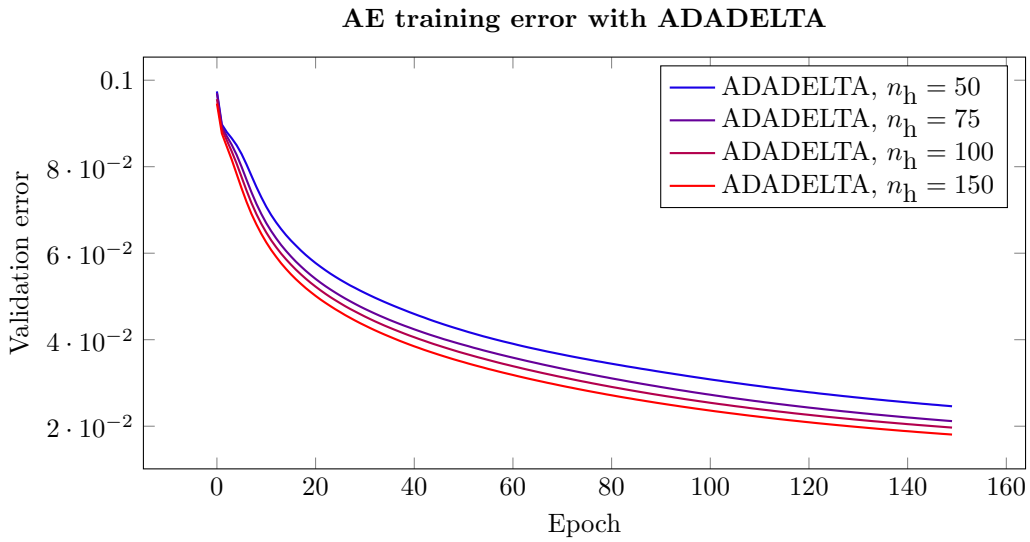
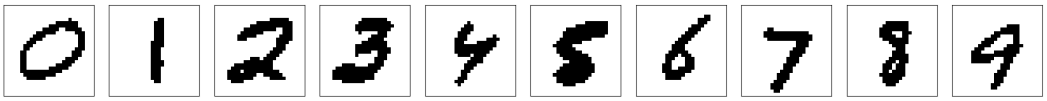


Figure 3: Error curves on the validation set for the AE using ADADELTA.

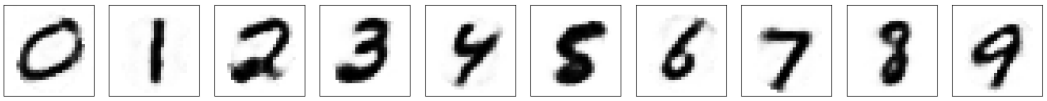
1.2 Learned features

The learned weights of each model can be examined to get an idea what and how the models are learning. By reshaping the weight vectors back into 28×28 grids each hidden units weights can be represented as images where each pixel value corresponds to the strength of that weight from the given pixel to the hidden unit. Plots for these are found in fig. 6. The AE models seems to in general have better trained hidden nodes as more of the components seems to have learned a specific shape instead of something that resembles white noise.

2 Deep architectures



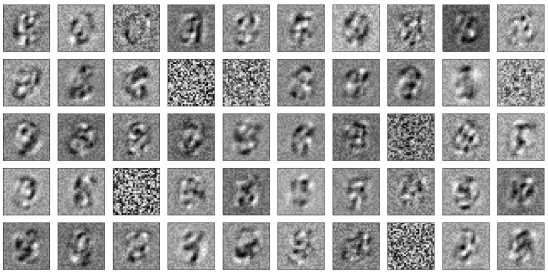
(a) Input patterns



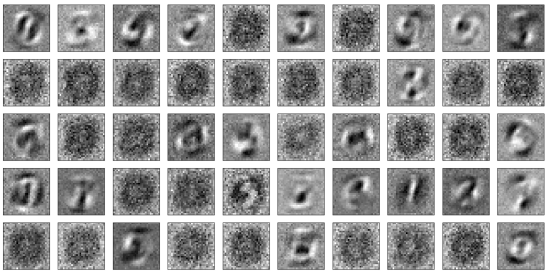
(b) Recalled patterns of the AE, 150 hidden units

(c) Recalled patterns of the RBM, 150 hidden units

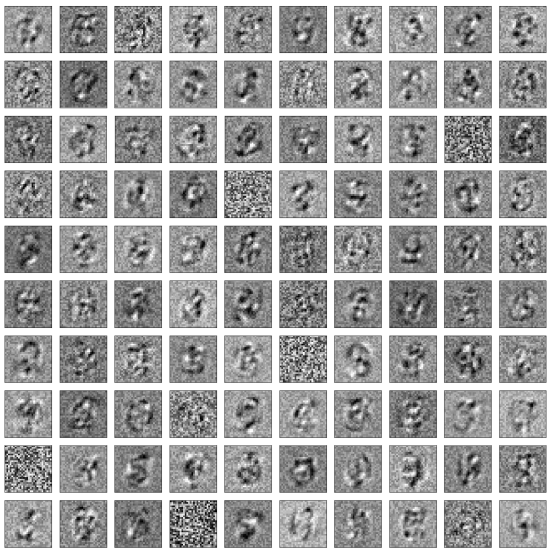
Figure 4: Recalled images from the models.



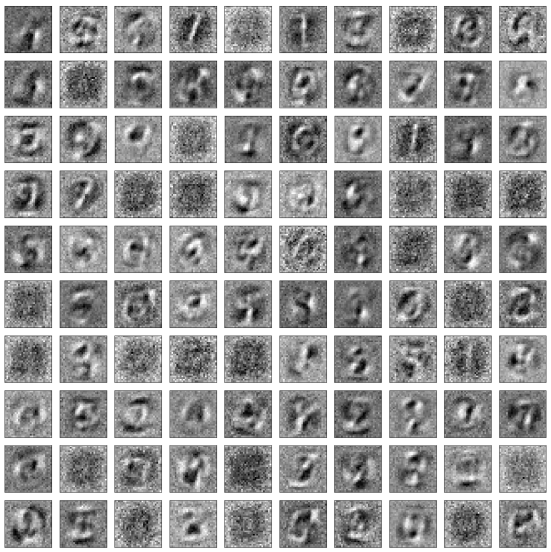
(a) AE weights, 50 hidden units.



(c) RBM weights, 50 hidden units.



(b) AE weights, 100 hidden units.

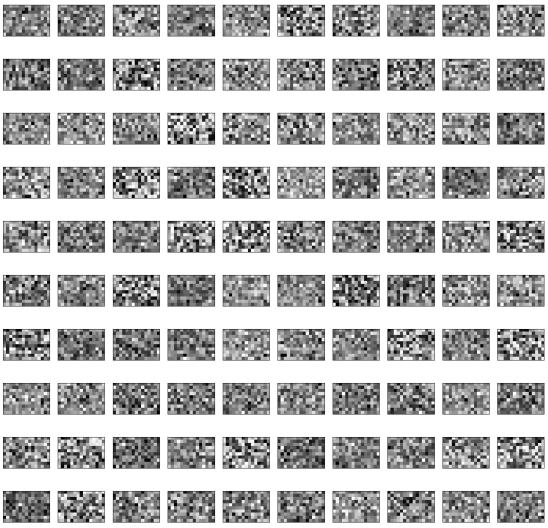


(d) RBM weights, 100 hidden units.

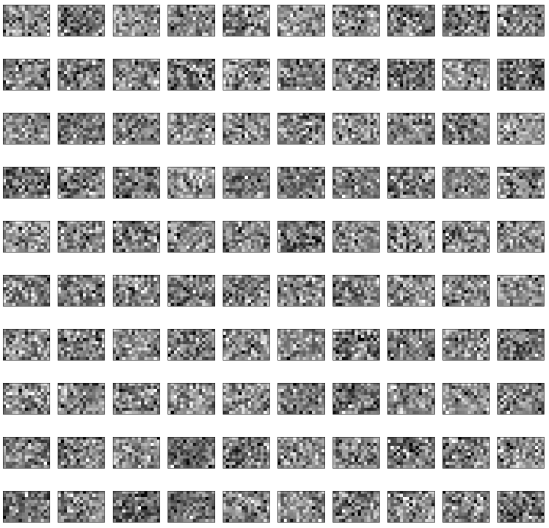
Figure 5: Image representations of the hidden weights in the AE and RBM models for 50 and 100 hidden units.

Table 1: Accuracy of different deep architectures

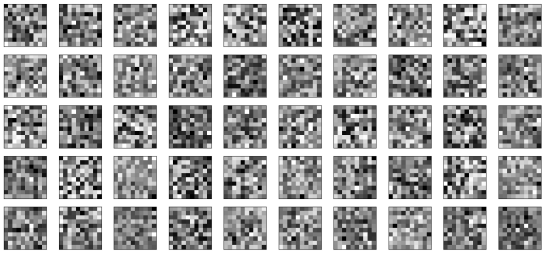
Layers	DBN	SAE
0	0.939	
1	0.937	0.937
2	0.907	0.943
3	0.859	0.866



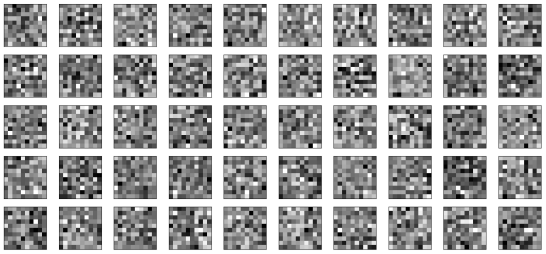
(a) Stacked AE, layer 2 weights.



(c) DBN, layer 2 weights.

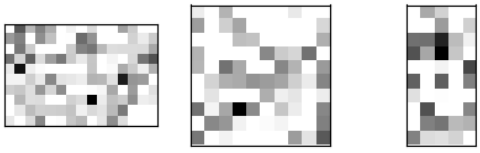


(b) Stacked AE, layer 3 weights.

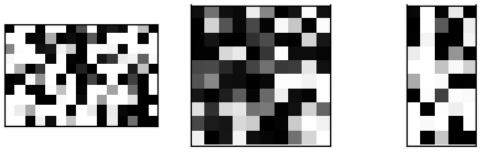


(d) DBN, layer 3 weights.

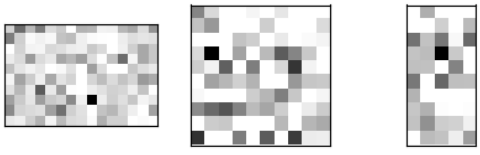
Figure 6: Image representations of the hidden weights in the AE and RBM models for 50 and 100 hidden units.



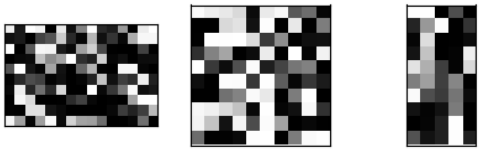
(a) Activations of hidden units in SAE for a 0 digit.



(b) Activations of hidden units in DBN for a 0 digit.



(c) Activations of hidden units in SAE for a 8 digit.



(d) Activations of hidden units in DBN for a 8 digit.