Essay - Cognition and Computation

Alexandre da Rocha Rodrigues (2039952)

February 12, 2022

1 Introduction

The EMNIST [1] Balanced dataset is derived from the NIST Special Database 19. It has 47 classes: digits (0-9), uppercase letters (A-Z), some lowercase letters (a,b,d,e,f,g,h,n,q,r,t). Consists in a total of 131600 samples, 112800 being for training and 18800 for testing. It provides a consistent and fair classification, with equal number of samples for each class, despite being sufficiently challenging.

A possible neural network used to model and classify these samples is the Deep Belief Network. It consists in a composition of Restricted Boltzmann Machines, the hidden layer of one serves as visible layer of the next.

A RBM is generative stochastic neural network used to learn the probability distribution over its inputs. This implies that the activation level of a neuron is a representation of a probability. The input is present in a set of visible neurons. It can extract statistical structure information in a set of hidden neurons. We use maximum likelihood learning to update the weights aiming for a accurate top-down reconstruction. The most probable configurations of hidden and visible neurons are specified by the energy function based on the connection weights.

These characteristics of RBMs allows us to use a stack of them, a DBN, to learn multiple levels of representation and thus better hidden representations that can serve as examples of a class. The DBN is trained with the constrastive divergence algorithm, reducing the discrepancy between the real and the learned probability distribution.

A Perceptorn is a simple neural network model with a set of inputs and a single binary output. We can hereby use a set of 47 Perceptrons, one corresponding to each class, to classify the samples, using the final hidden layer of the DBN as their inputs.

Feedforward Neural Networks are multiple layers of Perceptrons, having one or more hidden layers. In this case, we use the Rectified Linear Unit activation function to avoid the weakening of the error gradient.

2 Hyper Parameters

The parameters used can be found in the following table.

Parameter	**Value**
Hidden Layers Sizes	$1000, 1000$
Gibbs Sampling Steps (k)	1
Learning Rate	0.1
Initial Momentum	0.9
Final Momentum	0.9
Weight Decay	0.00001

Table 1: Hyper Parameters of the Deep Belief Network

These parameters were isolated and tested, i.e. starting from the 1st lab parameters, I changed only one of the parameters and found the best value, did this for all parameters and use the best of each in this final combination. This, obviously does not guarantee the best results, because the parameters can be the best when isolated but not the best combination of all of them. We can although consider these values as a good approximation to the best case. The objective was to reduce the average reconstruction error in the DBN training phase.

Enabling learning rate decay decreased the average reconstruction error but the weight representations were not as expected, due to a very small gradient, so it will be disable in this case. The Xavier weights initialization is also disabled because it did not produce any measurable performance improvement.

To serve as comparison we can use Feed Forward Neural Networks, one with 1 hidden layer and another with 2 hiddden layers, all of size 1000.

The Perceptron and FFNNs were trained in the same way, using an stochastic gradient descent optimizer with learning rate of 0.05 and the cross entropy loss function, for 1000 epochs.

3 Results

DBN	$474s$
Perceptron	$453s$
FFNN with 1 layer	$932s$
FFNN with 2 layers	$927s$

Table 2: Training Time

To achieve a training time close to the sum of the DBN and Perceptron training time, the FFNN with 2 hidden layers was trained in 60 epochs and the one with 1 hidden layer was trained in 150 epochs.

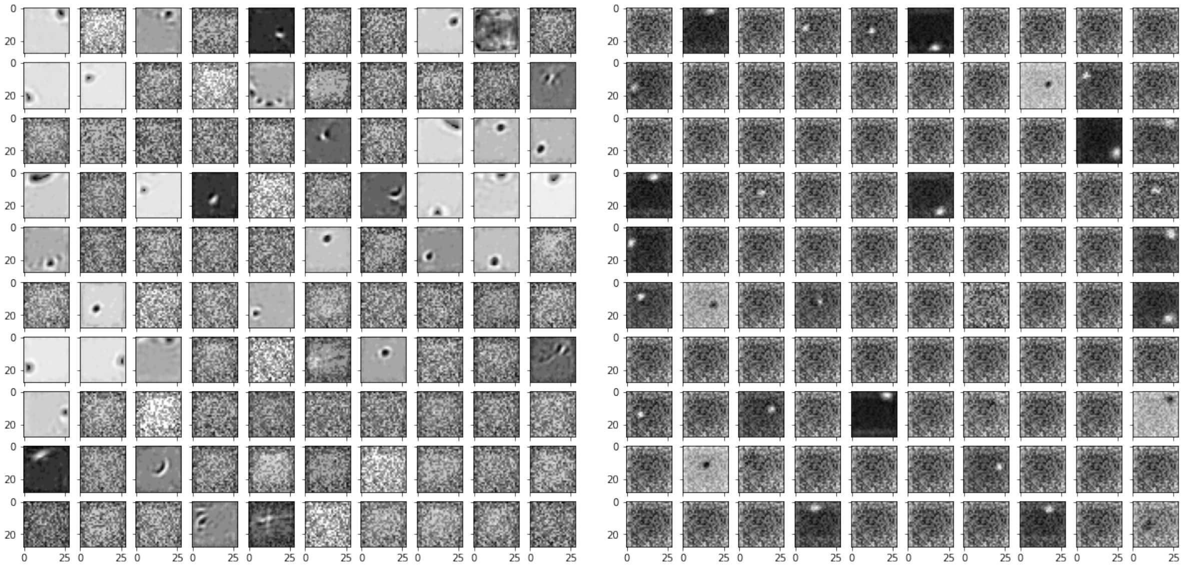
3.1 Accuracy

DBN	61.95%
FFNN with 1 layer	30.76%
FFNN with 2 layers	14.37%

Table 3: Accuracy

The DBN accuracy is a very good result, given that the EMNIST article [1] shows an 64% result for a OPIUM-based model with hidden layer size of 1000. The FFNNs produce lower accuracy as expected, the addition of a second hidden layer significantly decreased accuracy.

3.2 Representations



(a) Representation of the 1st hidden layer (b) Representation of the 2nd hidden layer

Figure 1: Representations of the DBN hidden layers

These results can be interpreted as the parts of the images that are selected by each neuron of the hidden layer. The 2nd hidden layer representation has no significantly noticeable feature selection, only some differences between each weight.

3.3 Dendrograms

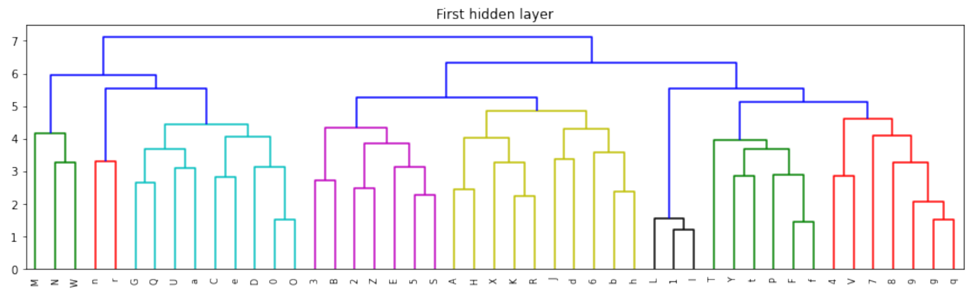


Figure 2: Dendrogram for 1st hidden layer

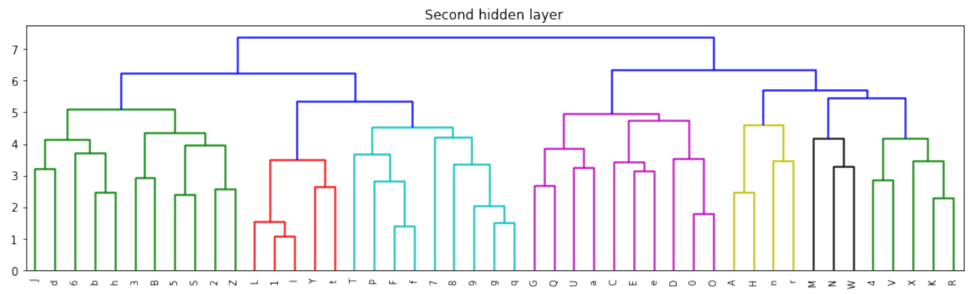


Figure 3: Dendrogram for 2nd hidden layer

These figures shows that the both hidden representation evaluates "l" and "1" as the most similar symbols, "L" is the next most similar to both. Other similar symbols are: "F" and "f", "g" and "q". The easiest distinctions for the model to make are represented as the first branching when reading from top to bottom. The second hidden representation shows more uniform branching. As an example, we can see "n" and "r" are only part of 3 chunks in the first hidden representation, on the second, the minimum is 4 chunks.

3.4 Confusion Matrix

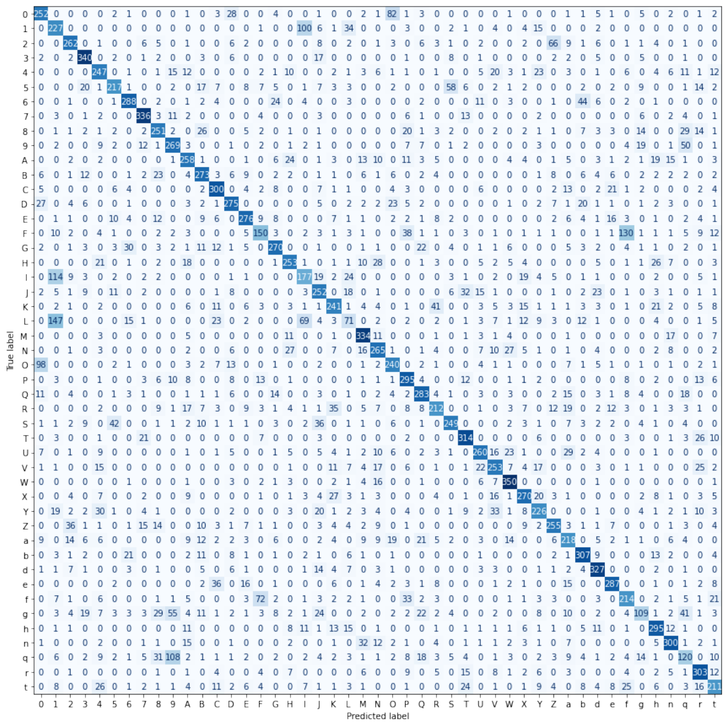


Figure 4: Confusion Matrix

The confusion matrix shows the correspondence between the predicted label and the true label. This allows us to check model errors and thus what cinfuses the model. The ideal confusion matrix would have only nonzeros in the diagonal.

The highest values outside the diagonal are the most similar symbols found by the model. These results are equivalent to the dendograms, largest values in the confusion matrix correspond to lower clades in the dendograms.

The largest values are for: "L" labeled as "1", "F" labeled as "f", "I" as "1" and "q" as "9".

3.5 Robustness to Noise

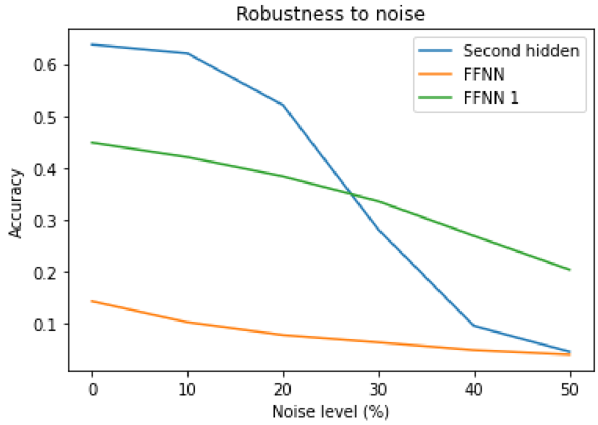


Figure 5: Robustness to noise

The FFNN with 2 hidden layers is generally worst. The DBN has the best accuracy until 30% noise level. The FFNN clearly are less affected by noise. ....

4 Conclusion

We...

References

[1] G. Cohen, S. Afshar, J. Tapson, and A. van Schaik, *EMNIST: An extension of MNIST to handwritten letters*, 2017. arXiv: 1702.05373.