An autoencoder is a specific type of neural network in which the output is the same as input. The input is encoded into a smaller dimension, creating a bottleneck structure, from which the output can be decoded. The compressed representation is more commonly known as the latent space representation.

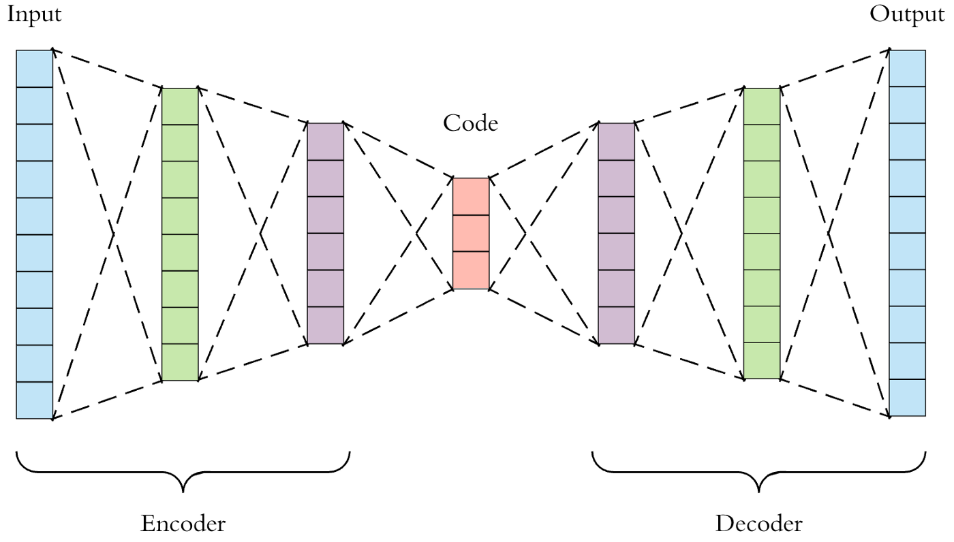


Figure 1 Typical architecture of an autoencoder

In a general concept, an autoencoder has three different parts – the encoder, code and the decoder. The encoder compresses the given input data into a lower dimensional code while the decoder decodes the code into the meaningful output of the original dimension.



Figure 2 8-3-8 autoencoder

**Simulation Setup:**

Total number of layers used = 3 (input – hidden – output)

Total number of input nodes = 8 (identity matrix of order 8)

Total number of hidden nodes = 4 (1 node for the bias)

Total number of output nodes = 8 (same as input)

Total number of iterations = 5000

Learning rate ( = 0.3

**Results:**

1. Learned Hidden Layer Representation:

|  |  |  |
| --- | --- | --- |
| Input | Hidden values | Output |
| |  | | --- | | 10000000 | | 01000000 | | 00100000 | | 00010000 | | 00001000 | | 00000100 | | 00000010 | | 00000001 | | |  |  |  | | --- | --- | --- | | 0.98 | 0.99 | 0.99 | | 0.97 | 0.02 | 0.97 | | 0.79 | 0.01 | 0.02 | | 0.02 | 0.01 | 0.68 | | 0.01 | 0.72 | 0.98 | | 0.02 | 0.27 | 0.01 | | 0.18 | 0.99 | 0.21 | | 0.99 | 0.79 | 0.02 | | |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | **0.93** | 0.03 | 0.00 | 0.00 | 0.01 | 0.00 | 0.00 | 0.03 | | 0.04 | **0.95** | 0.03 | 0.04 | 0.00 | 0.00 | 0.00 | 0.00 | | 0.00 | 0.04 | **0.94** | 0.00 | 0.00 | 0.04 | 0.00 | 0.06 | | 0.00 | 0.04 | 0.00 | **0.93** | 0.06 | 0.03 | 0.00 | 0.00 | | 0.05 | 0.00 | 0.00 | 0.04 | **0.93** | 0.00 | 0.05 | 0.00 | | 0.00 | 0.00 | 0.04 | 0.05 | 0.00 | **0.92** | 0.05 | 0.00 | | 0.02 | 0.00 | 0.00 | 0.00 | 0.05 | 0.05 | **0.92** | 0.05 | | 0.05 | 0.00 | 0.04 | 0.00 | 0.00 | 0.00 | 0.05 | **0.93** | |

1. Sum of squared errors for each output unit

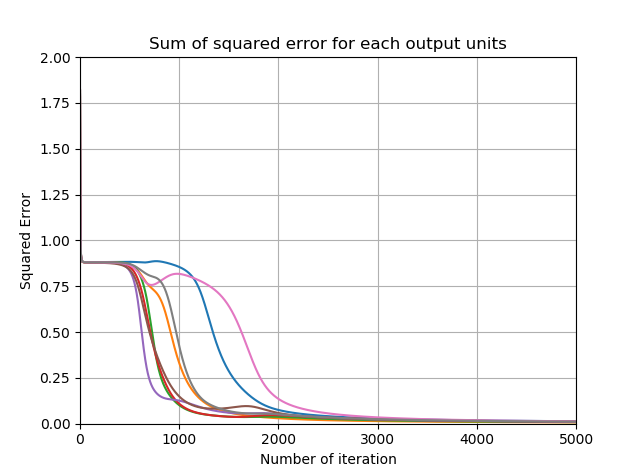


Figure 3 Sum of squared error for each output units

1. Evolution of hidden layer representation

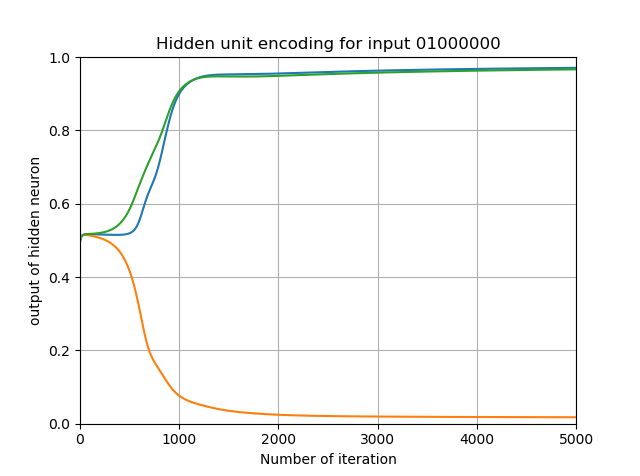


Figure 4 Three hidden unit values for input 01000000

1. Weights from inputs to one hidden unit

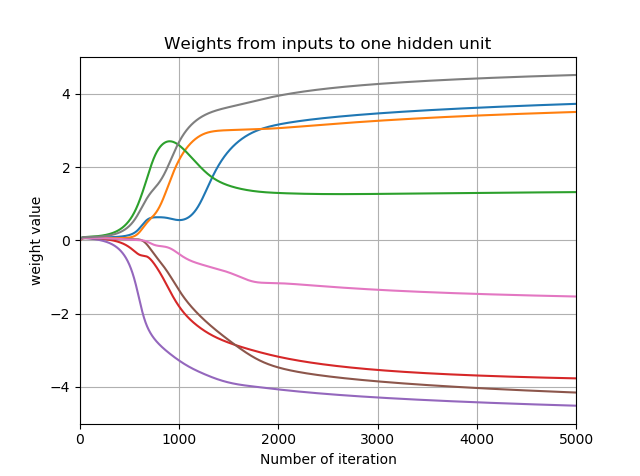


Figure 5 Weights from input to one hidden unit