Parshvanath Charitable Trust's

A. P. SHAH INSTITUTE OF TECHNOLOGY

(Approved by AICTE New Delhi & Govt. of Maharashtra, Affiliated to University of Mumbai)
(Religious Jain Minority)

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING
(ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

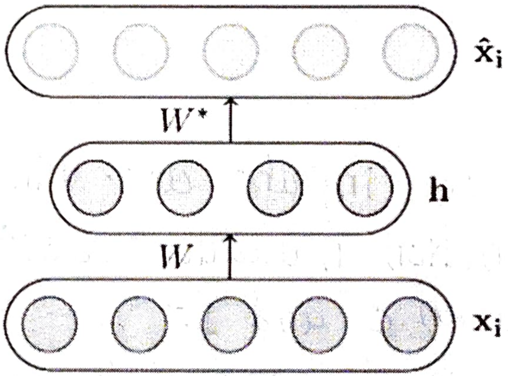
Autoencoders

An autoencoder is a special type of feed forward neural network which does the following:

- Encodes its input x_i into hidden representation h.
- Decodes the input again from this hidden representation.

The model is trained to minimize a certain loss function which will ensure that x_i(hat) is close to x_i

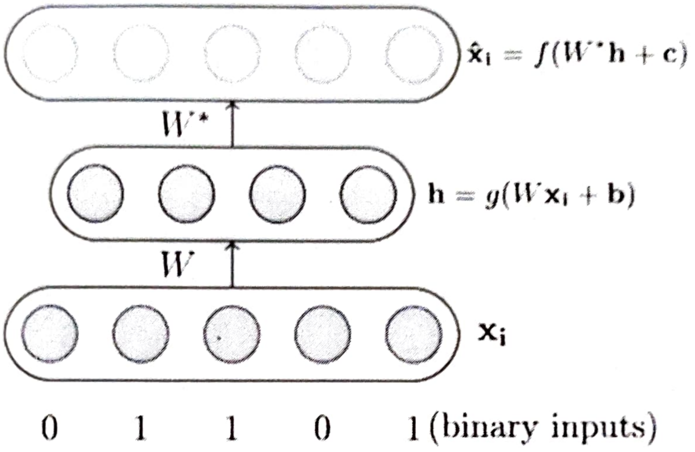
$(\hat{x}_i \approx x)$



$$h = g(Wx_i + b)$$
$$\hat{x}_i = f(W^*h + c)$$

- If you compute a hidden representation h, which is smaller than your original data, and from that hidden representation, if you are able to reconstruct x, then that would mean that this hidden representation captures everything that is required.

Binary Input:



- suppose all our inputs are binary.
- Logistic function naturally restricts all outputs to be between 0 and 1.
- Hence it is most appropriate for the decoder.

$$\therefore \hat{x}_i = \text{logistic}(W^*h + c)$$

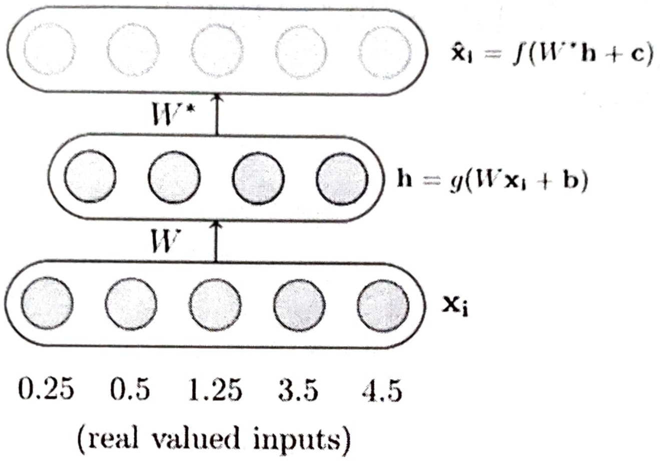
- Loss function you may use here is cross entropy loss function.

Binary cross entropy $= -[y \log(P) + (1-y)\log(1-P)]$

y: y is true label (1 for +ve, 0 for −ve)

p: p is probability of the positive class

Real Valued Input:


$\hat{\mathbf{x}}_i = f(W^*\mathbf{h} + \mathbf{c})$
W^*
$\mathbf{h} = g(W\mathbf{x}_i + \mathbf{b})$
W
$\mathbf{x}_i$
0.25
0.5
1.25
3.5
4.5
(real valued inputs)

- Suppose all our inputs are real.
- Appropriate function for decoder would be,

$$\hat{x}_i = W^* h + c$$

- Tanh and logistic will not be appropriate, as it will restrict $\hat{x}_i$ to lie between [0, 1] & [-1, 1] whereas $\hat{x}_i \in R$ (real number)
- g will be typically chosen as sigmoid function.
- You may use mean squared error for loss function.

$$\min_{W, W^*, c, b} \frac{1}{m} \sum_{i=1}^{m} (\hat{x}_i - x_i)^T (\hat{x}_i - x_i)$$

- We can then train the autoencoder just like a regular feedforward network using backpropagation.