Introduction to Neural Netwroks (CPE 520)
HW#13: Dec. 14, 2021

Q.1: Find the expression for the Inverse for the following linear matrix equations (systems):

(a): $\mathbf{y}_{n\times1} = \mathbf{A}_{n\times m}\mathbf{x}_{m\times1}$ where n < m and rank=n

(b): $\mathbf{y}_{n\times1} = \mathbf{A}_{n\times m}\mathbf{x}_{m\times1}$ where n > m and rank=m

Q.2: (a) What is the difference between Ridge and LASSO regressions?
(b) Write the expressions for them.

Q.3: Convolve the following kernels:

$$\begin{bmatrix}1\\2\\2\\1\end{bmatrix} * \begin{matrix}1 & 1 & 1\end{matrix}$$

Q.4: A number of DNN architectures use 1x1 convolution, what does it mean and what is its purpose?

Q.5: What is the difference between learning update term and Momentum term in DNN.

Q.6: Why we need to do regularization on weights in DNN?

Q.7: Give examples of different pooling methods used in DNN.

Q.8: What is Global Pooling.

Q.9: What is the difference between logistic regression and Softmax?

Q.10: Write the **hinge loss** expression and in which classifier it is usually used.

Q.11: The common regularization in machine learning are based on L2 or L1 norm on the weights, write the expressions for these two norms.

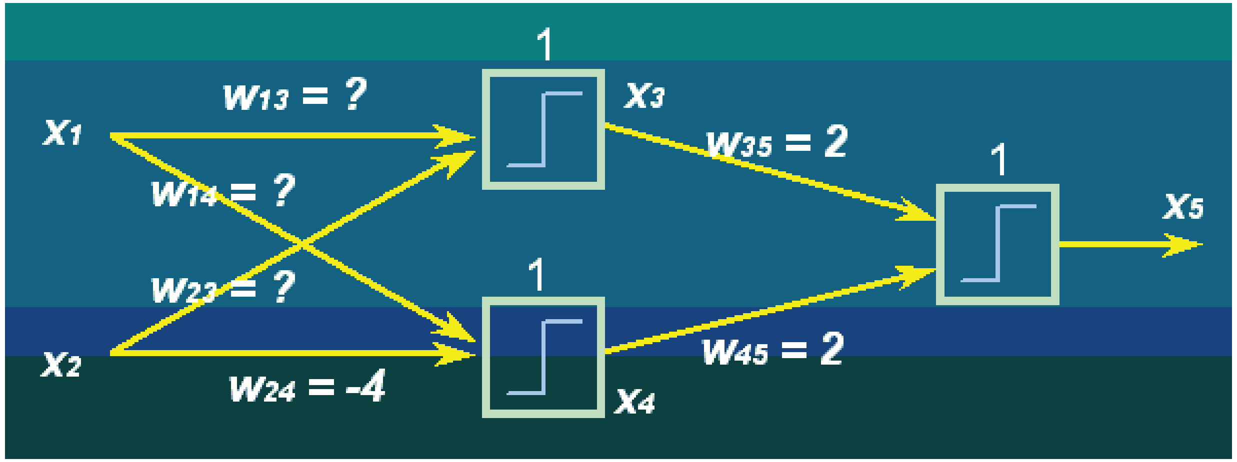
Q.12: In DNN we use auto-encoders, what are the usefulness & applications of developing autoencoders?

Q13: Describe the Kohonen Feature Maps.

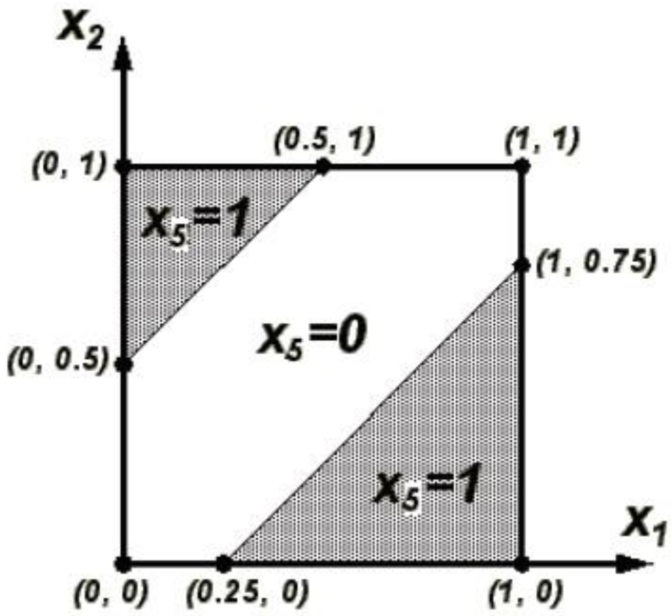
Q.14: Describe how learning vector quantization works.

Q.15: Give two examples of unsupervised deep neural network architectures.

Q.16: An 2-2-1 MLP has the following configuration:



Suppose that the above MLP has the following decision boundary for XOR problem:



Find the missing weights of the MLP.

Q.17: Consider sigmoid function $f(NET_j)$:

$$f(NET_j) = \frac{1}{1+e^{-\lambda NET_j}}$$

Find its gradient.

Q.18: What is kernel function? What is the kernel trick?

Q.19: Given 2-dimensional vectors $\mathbf{x}_i = [x_1, x_2]^T$ for i=1,2, …,N , let the kernel be a polynomial $k(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i^T \mathbf{x}_j)^2$

show that $k(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$ where $\phi(\mathbf{x}_i) = (x_1^2, \sqrt{2}x_1 x_2, x_2^2)^T$