Intelligent Control

Artificial Neural Network (4b)

by
Dr. Nor Maniha Abdul Ghani
FKEE
normaniha@ump.edu.my

Contents

- **4.3 ANN Model**
- **4.4 ANN Learning**

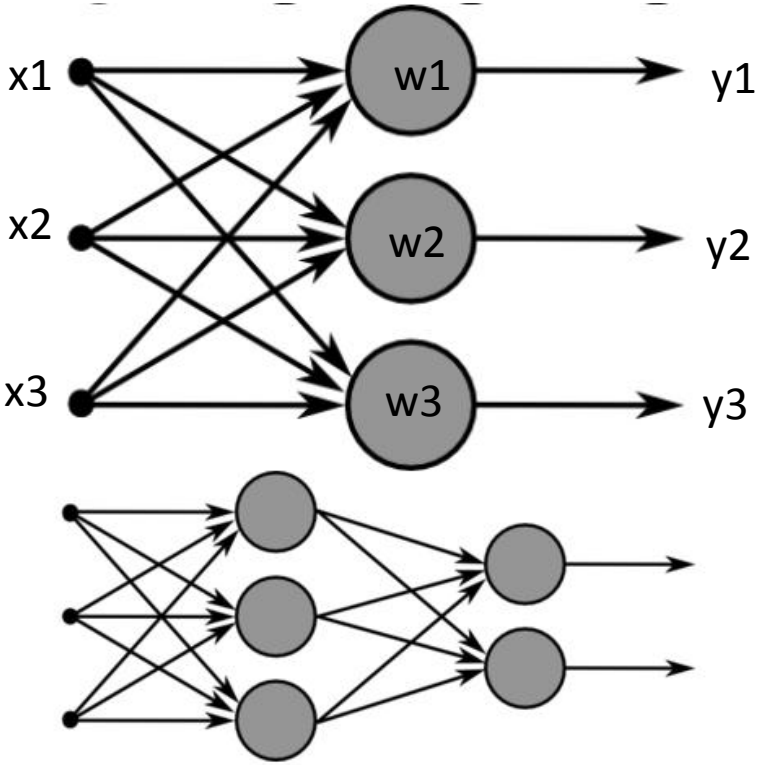
ANN Model

4.3

ANN Models

- Feed-forward networks
- Radial basis function networks
- Recurrent networks
- Echo state networks
- Hopfield networks
- Competitive model
- Self-organizing maps
- ART model
- Boltzmann machine
- Committee of machines
- Etc….

Feed-forward Network



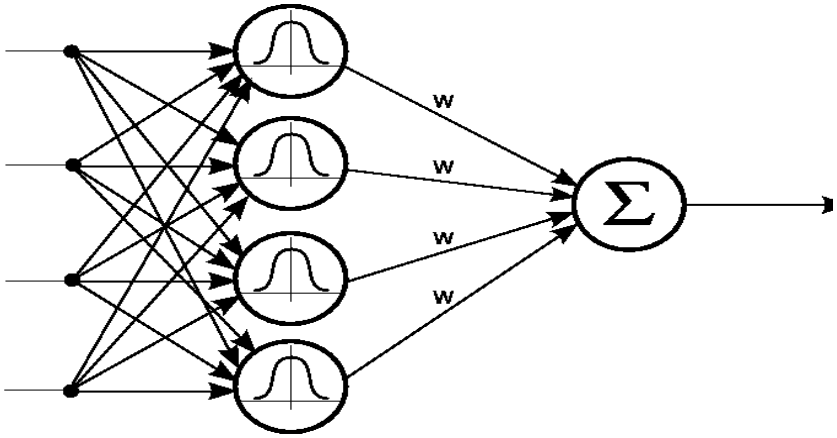
$$y_1 = x_1 w_1 + x_2 w_1 + x_3 w_1$$

$$y_2 = x_1 w_2 + x_2 w_2 + x_3 w_2$$

$$y_3 = x_1 w_3 + x_2 w_3 + x_3 w_3$$

http://en.wikibooks.org/wiki/Artificial_Neural_Networks/Feed-Forward_Networks

Radial Basis Function Networks

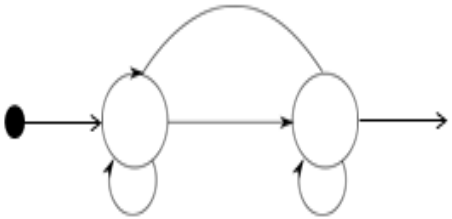


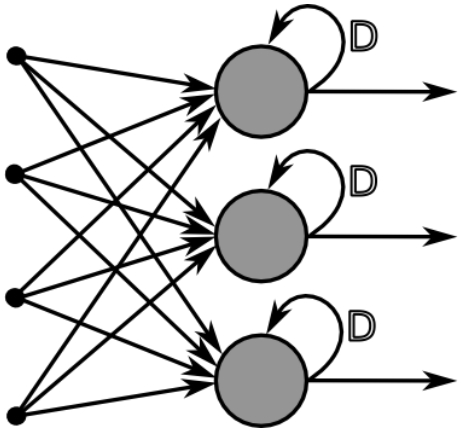
http://bio.felk.cvut.cz/biocmsms/index.php?page=neural-networks

$$\sigma(\zeta) = e^{-\beta\zeta^2}$$

- 3 layers
- Hidden layer : Radial basis activation function (RBF)
- Output layer of linear summation unit(s).
- Only the tap weights between the hidden layer and the output layer are modified during training.

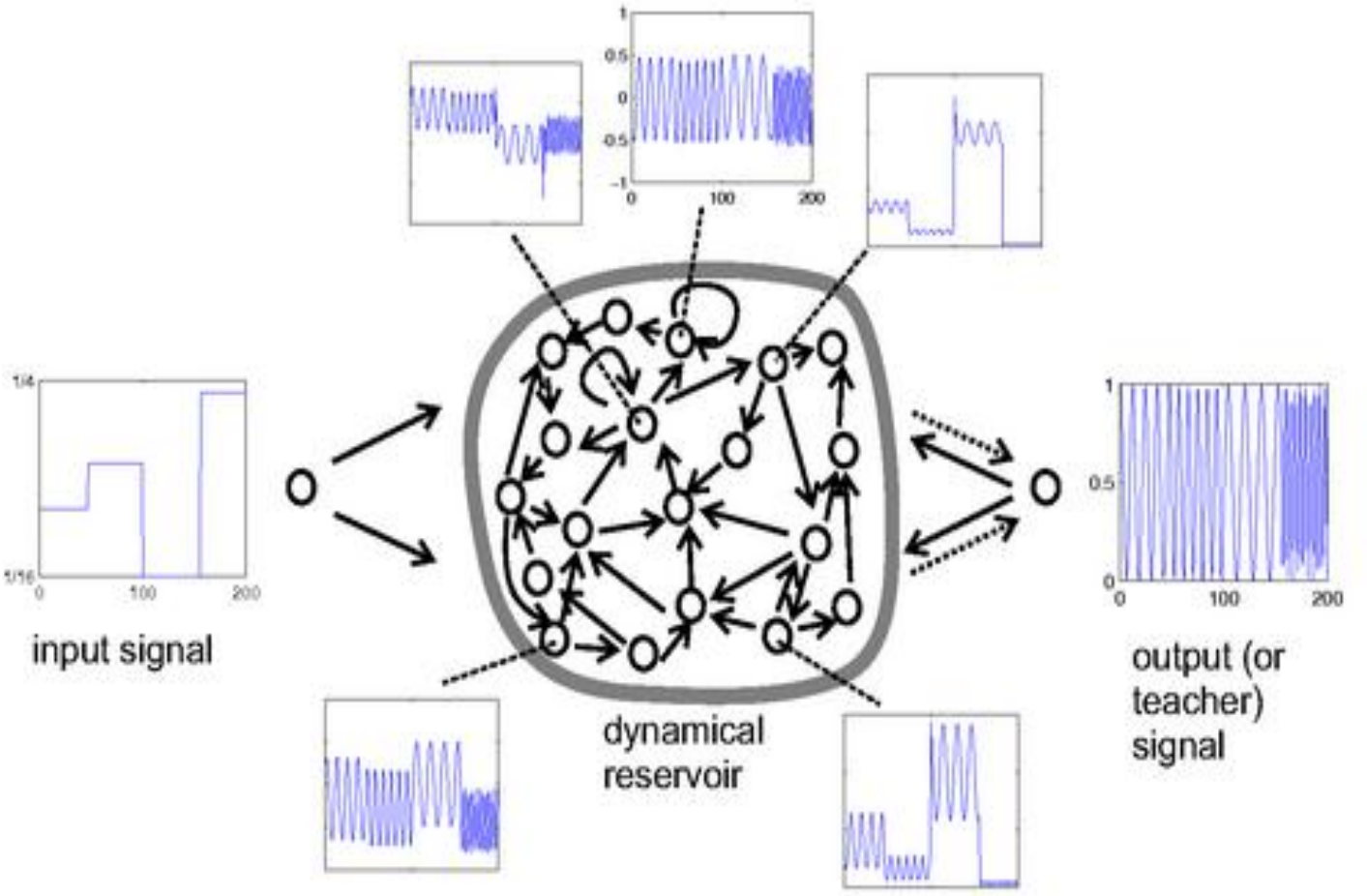
Recurrent Networks





http://en.wikibooks.org/wiki/Artificial_Neural_Networks/Recurrent_Networks

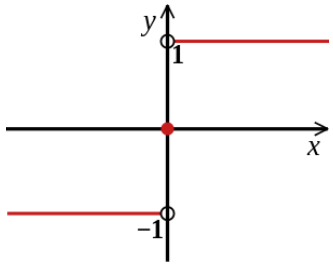
Echo State Networks



http://www.scholarpedia.org/article/Echo_state_network

Hopfield Networks

- The oldest and simplest networks.
- It utilizes a network energy function.
- The activation function of a binary Hopfield network is given by the signum function of a biased weighted sum:

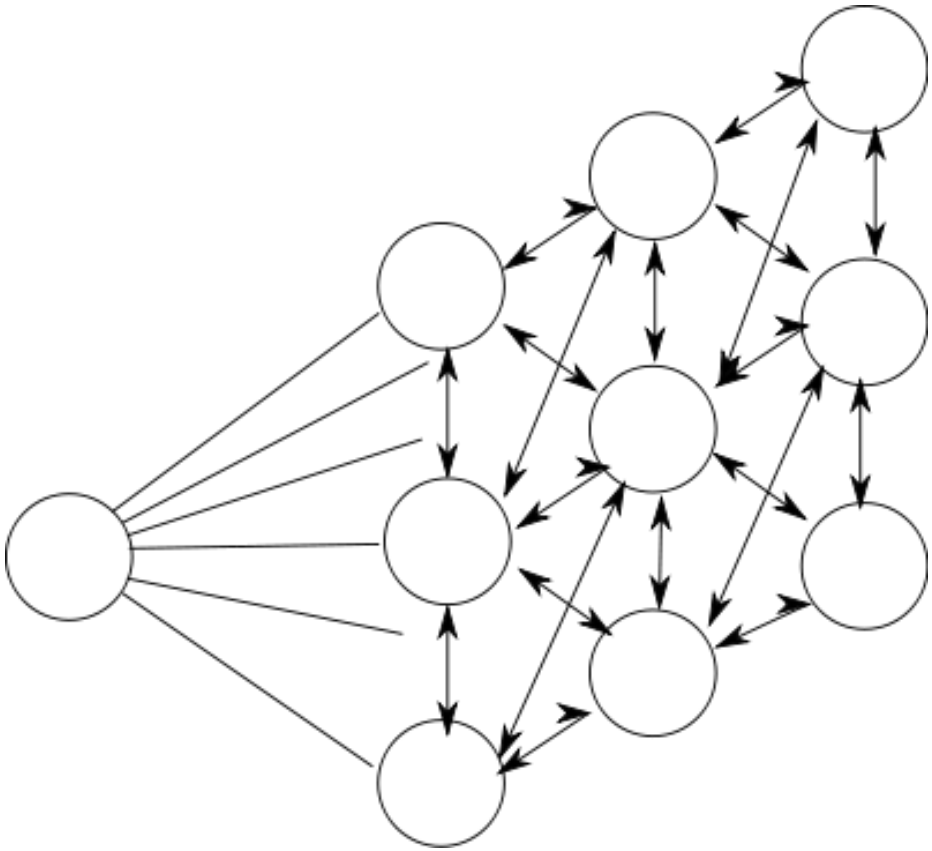


$$y_i = sgn(\zeta_i - \theta_i)$$

$$\operatorname{sgn}(x) := \begin{cases} -1 & \text{if } x < 0, \\ 0 & \text{if } x = 0, \\ 1 & \text{if } x > 0. \end{cases}$$

with ζ_i and θ_i are output of the layer and threshold applied, respectively.

Competitive Networks



http://en.wikibooks.org/wiki/Artificial_Neural_Networks/Competitive_Models

ANN LEARNING

4.4

ANN Learning

Application of ANN involves two phases:

- Learning
- Recall

Learning

-ANN is trained until the tasks has learned.

-Supervised

-Unsupervised

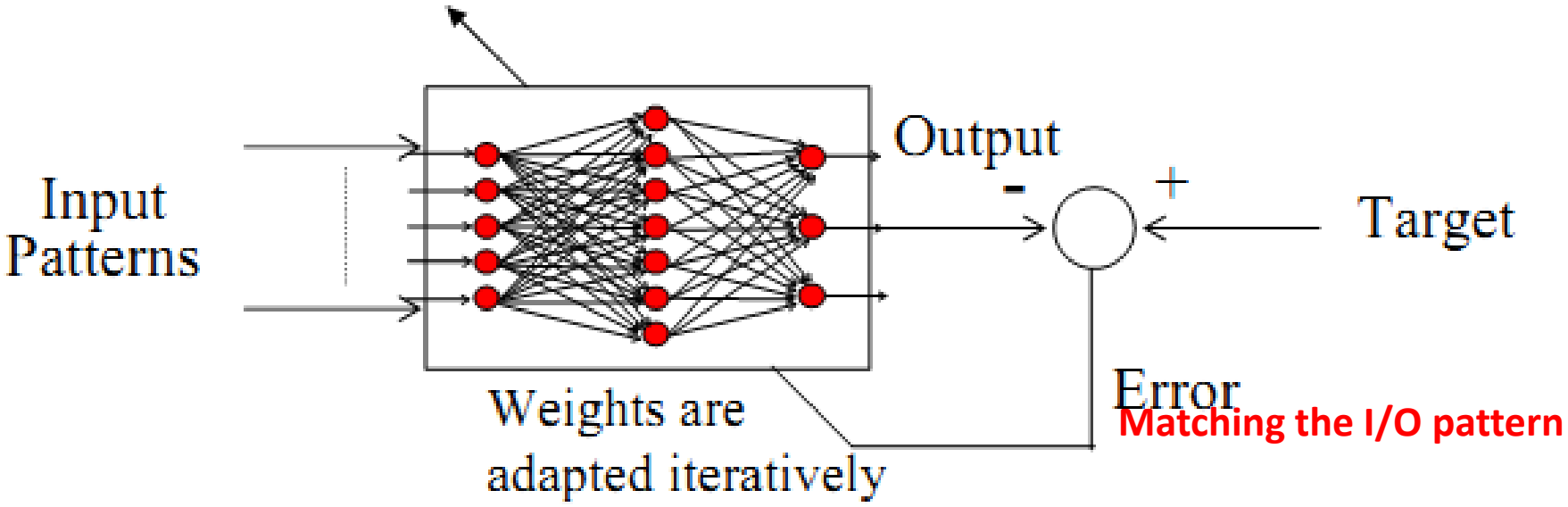
Recall

For task solving

ANN paradigm

ANN Learning algorithm	Model	Learning mode
• Backpropagation. • Competitive Learning • ART • Hopfield • Kohonen	• Feedforward • Feedback	• Supervised • Unsupervised

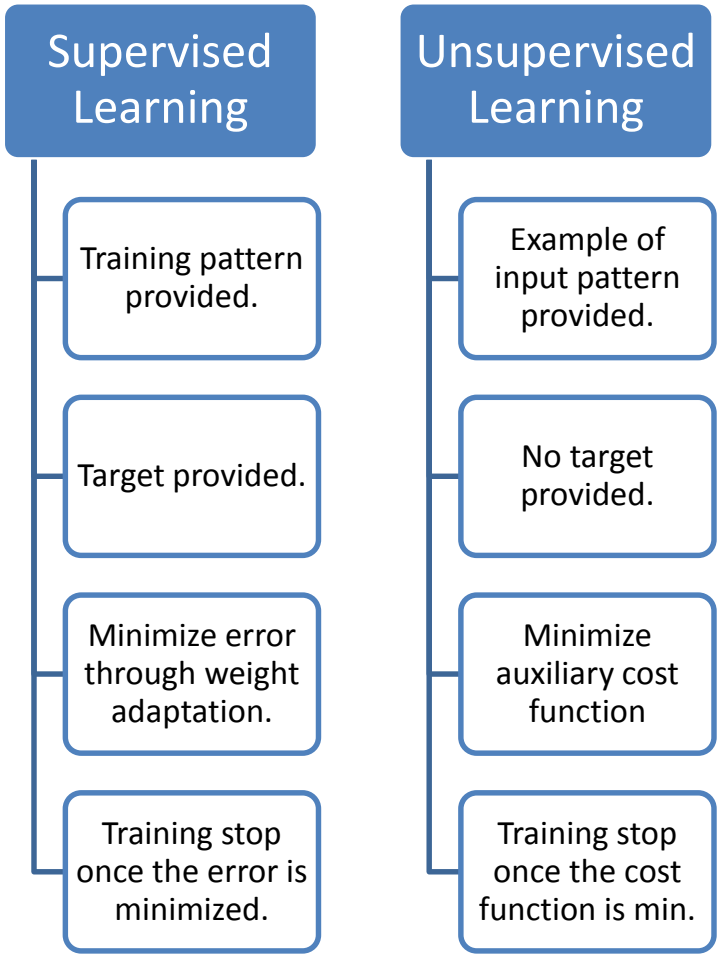
Supervised Learning



Learning configuration

http://slideplayer.com

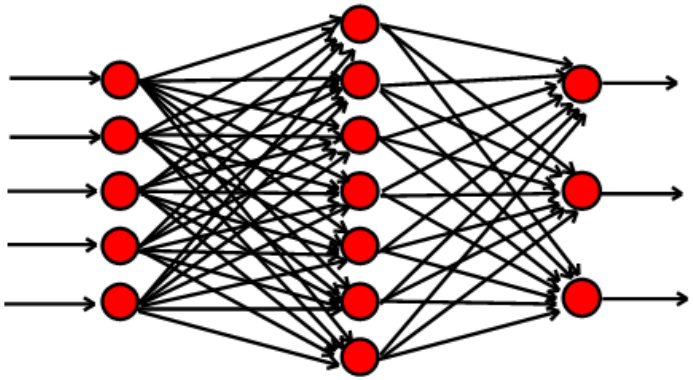
Supervised vs Unsupervised



ANN Classifications

Learning Mode	Feedforward	Feedback
Supervised	• Least Mean Square • Backpropagation • Reinforcement Learning • Fuzzy ARTMAP • GRNN	• Recurrent Backpropagation
Unsupervised	• Self-Organizing Maps • Competitive Learning • Counter Propagation	• Adaptive Resonance Theory • Fuzzy ART • Boltzmann Learning • Hopfield Network • BAM

ANN Performance



000 = MOUSE

001 = RABBIT

010 = COW

http://slideplayer.com

Basis of ANN computing idea

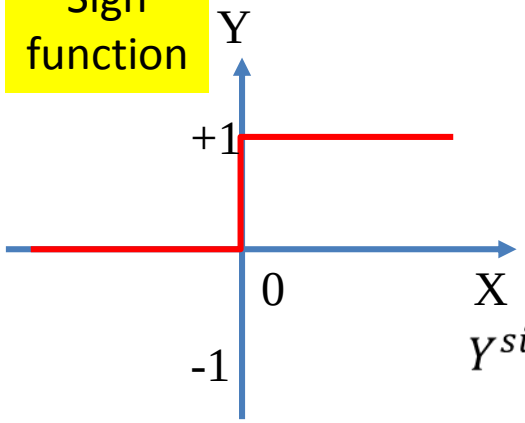
- Neuron computes the input signals and compares the result with a threshold value, θ.
- If the input is *less than* θ, then the neuron output is *-1*, otherwise *+1*.
- Hence, the following activation function(*sign* function) is used,

$$X = \sum_{i=1}^{n} x_i w_i \qquad Y = \begin{cases} +1 & if\ X \geq \theta \\ -1 & if\ X < \theta \end{cases} \qquad \Rightarrow \qquad Y = sign\left[= \sum_{i=1}^{n} x_i w_i - \theta \right]$$

Where X is the net weighted input to neuron, x_i is the i input value, w_i is the weight of input i . n is the number of neuron input and Y is the neuron output.

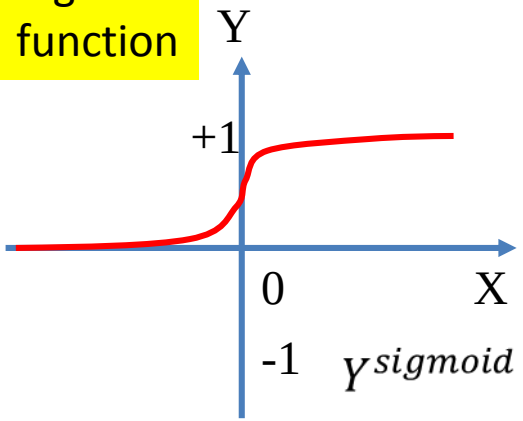
Other types of activation function

Sign function



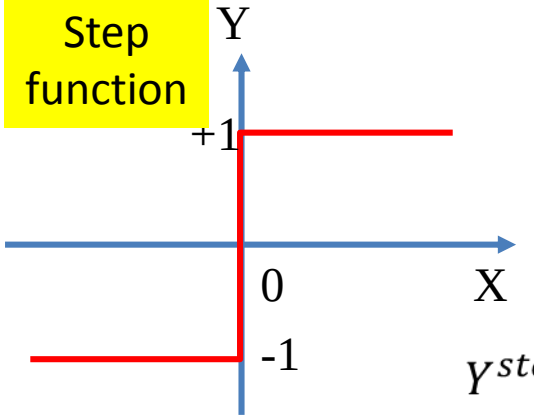
$$Y^{sign} = \begin{cases} +1, if\ X \geq 0 \\ -1, if\ X < 0 \end{cases}$$

Sigmoid function



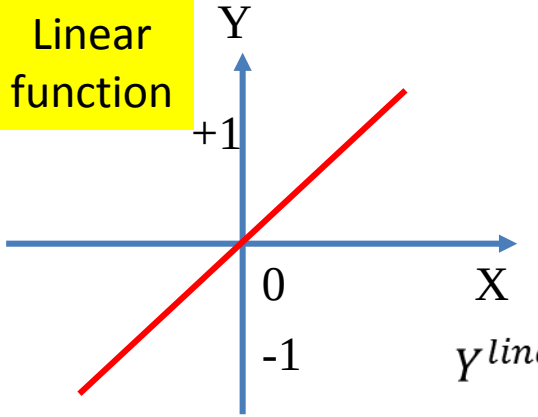
$$Y^{sigmoid} = \frac{1}{1 + e^{-X}}$$

Step function



$$Y^{step} = \begin{cases} +1, if\ X \geq 0 \\ 0, if\ X < 0 \end{cases}$$

Linear function



$$Y^{linear} = X$$

Dr. Nor Maniha Abdul Ghani

Faculty of Electrical and Electronics Engineering
Universiti Malaysia Pahang,
26600, Pekan, Pahang, Malaysia
Phone: +609-424-6087
Fax: +609-424-6000

http://fkee.ump.edu.my/index.php/en/staff-menu/articles-staff/1034-niha-main-profile