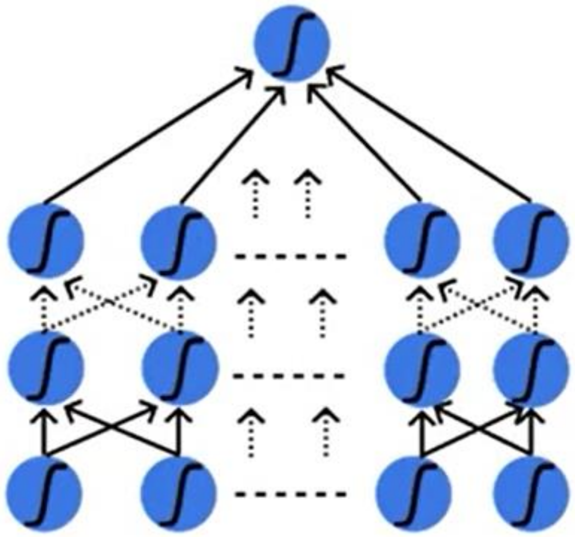
राष्ट्रीय प्रौद्योगिकी संस्थान सिक्किम
NATIONAL INSTITUTE OF TECHNOLOGY SIKKIM

Deep Learning:
Feed Forward Neural Network

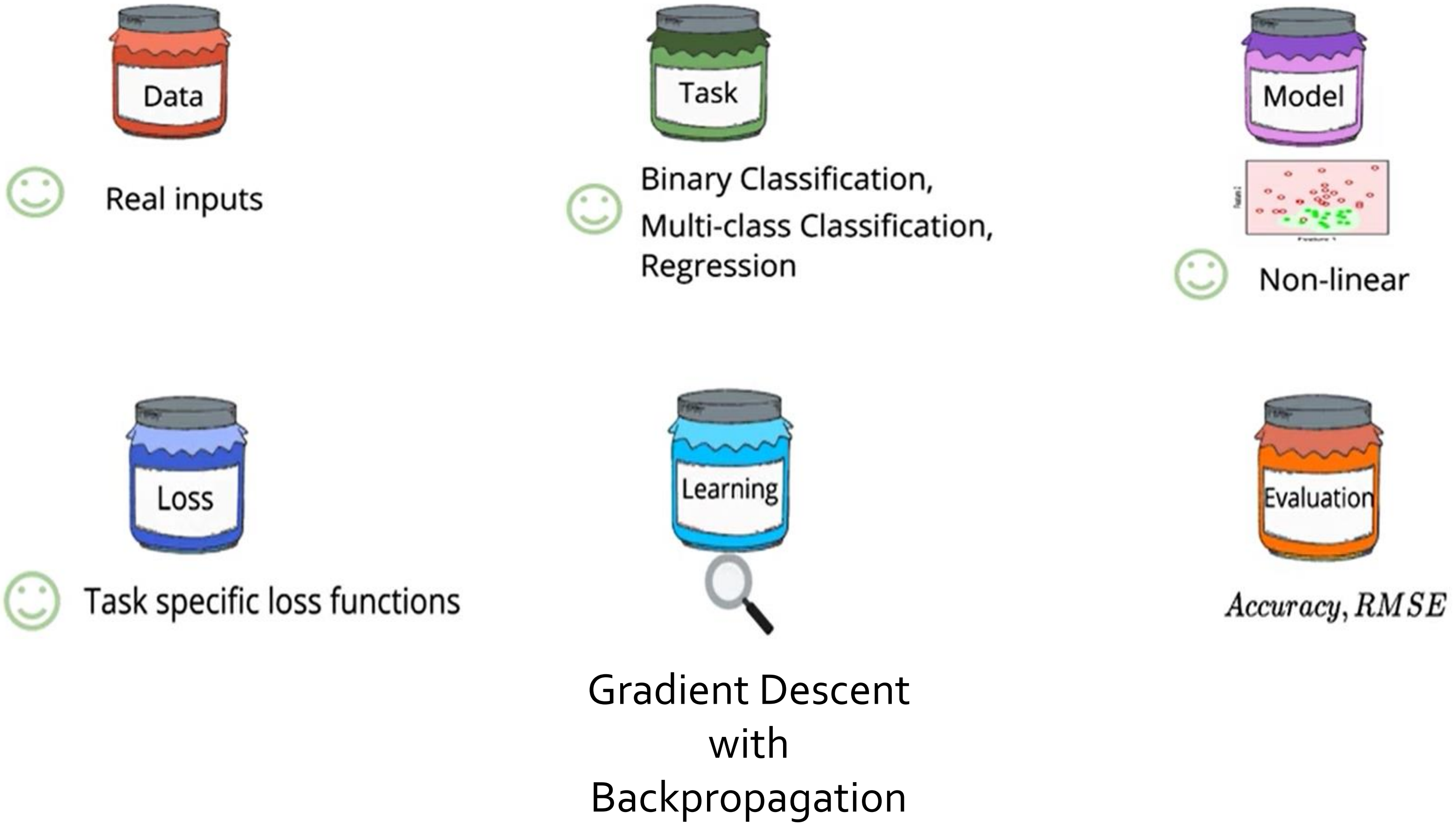
Course Instructor:
Dr. Bam Bahadur Sinha
Assistant Professor
Computer Science & Engineering
National Institute of Technology
Sikkim

Complex Functions



Data
Task
Model
Loss
Learning
Evaluation
$w = w + \eta \frac{\partial L}{\partial w}$

The Roadmap ahead



☺ Real inputs

☺ Binary Classification, Multi-class Classification, Regression

☺ Non-linear

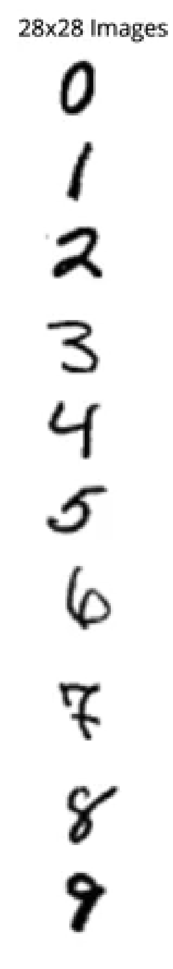
☺ Task specific loss functions

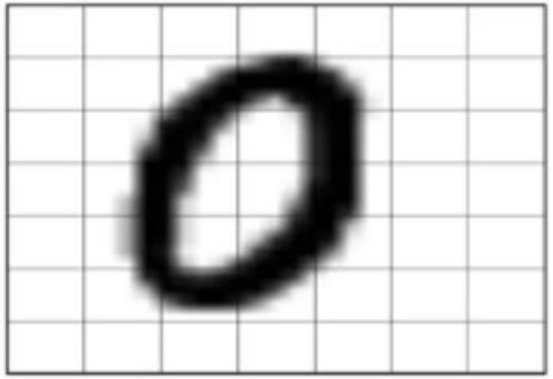
Gradient Descent
with
Backpropagation

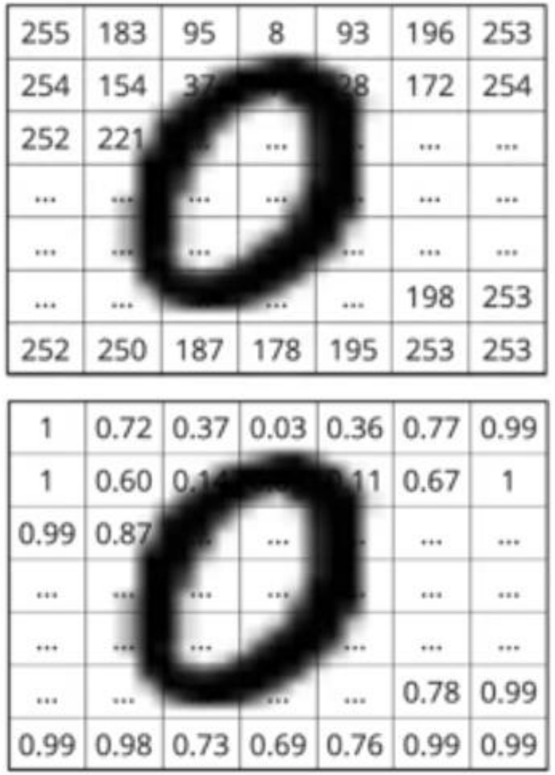
$Accuracy, RMSE$

Data & Task

Data: MNIST Images







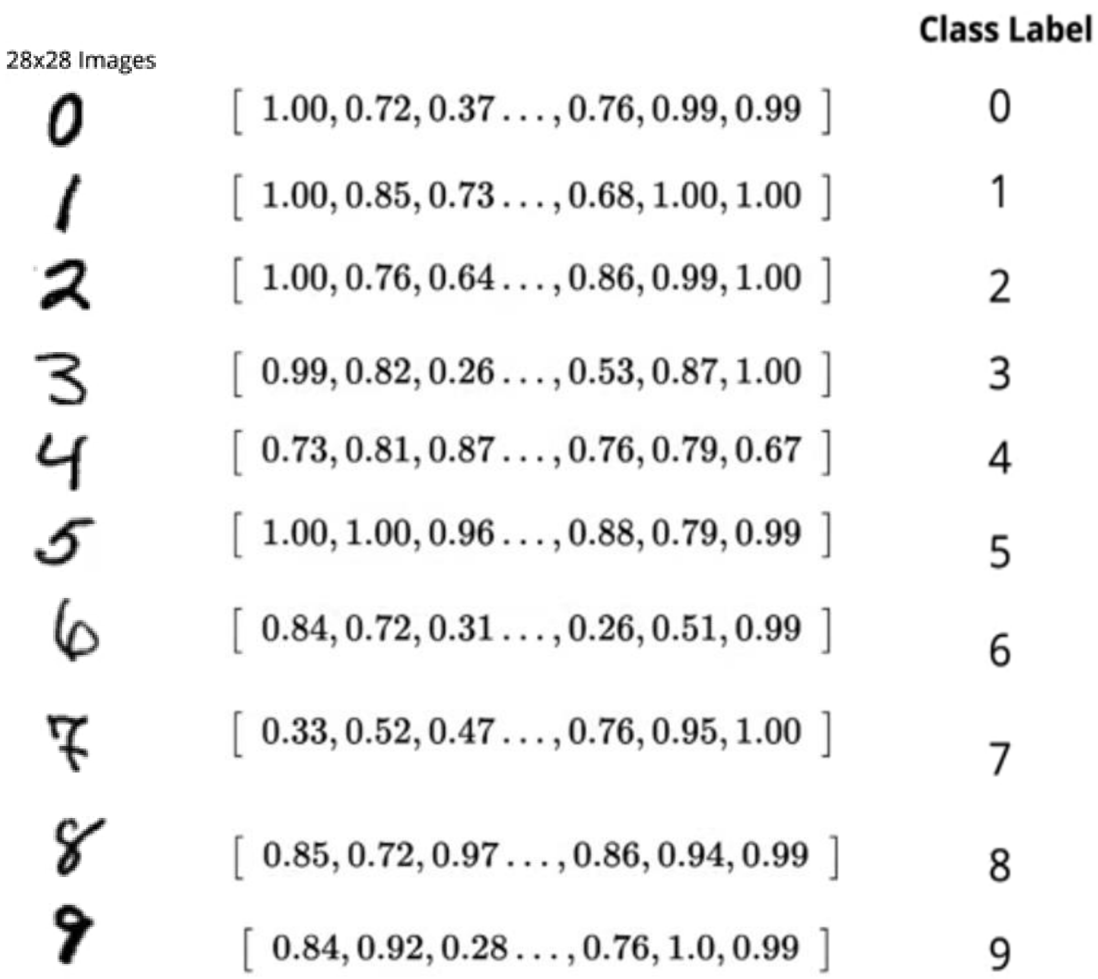
255	183	95	8	93	196	253
254	154	37	[illegible]	28	172	254
252	221	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	[illegible]	...	...	198	253
252	250	187	178	195	253	253

1	0.72	0.37	0.03	0.36	0.77	0.99
1	0.60	0.14	[illegible]	0.11	0.67	1
0.99	0.87	...	...	...	...	...
...	...	...	...	...	...	...
...	...	...	...	...	...	...
...	...	[illegible]	...	...	0.78	0.99
0.99	0.98	0.73	0.69	0.76	0.99	0.99

How can we represent MNIST images as a Vector?

Data and Task

Data: MNIST Images



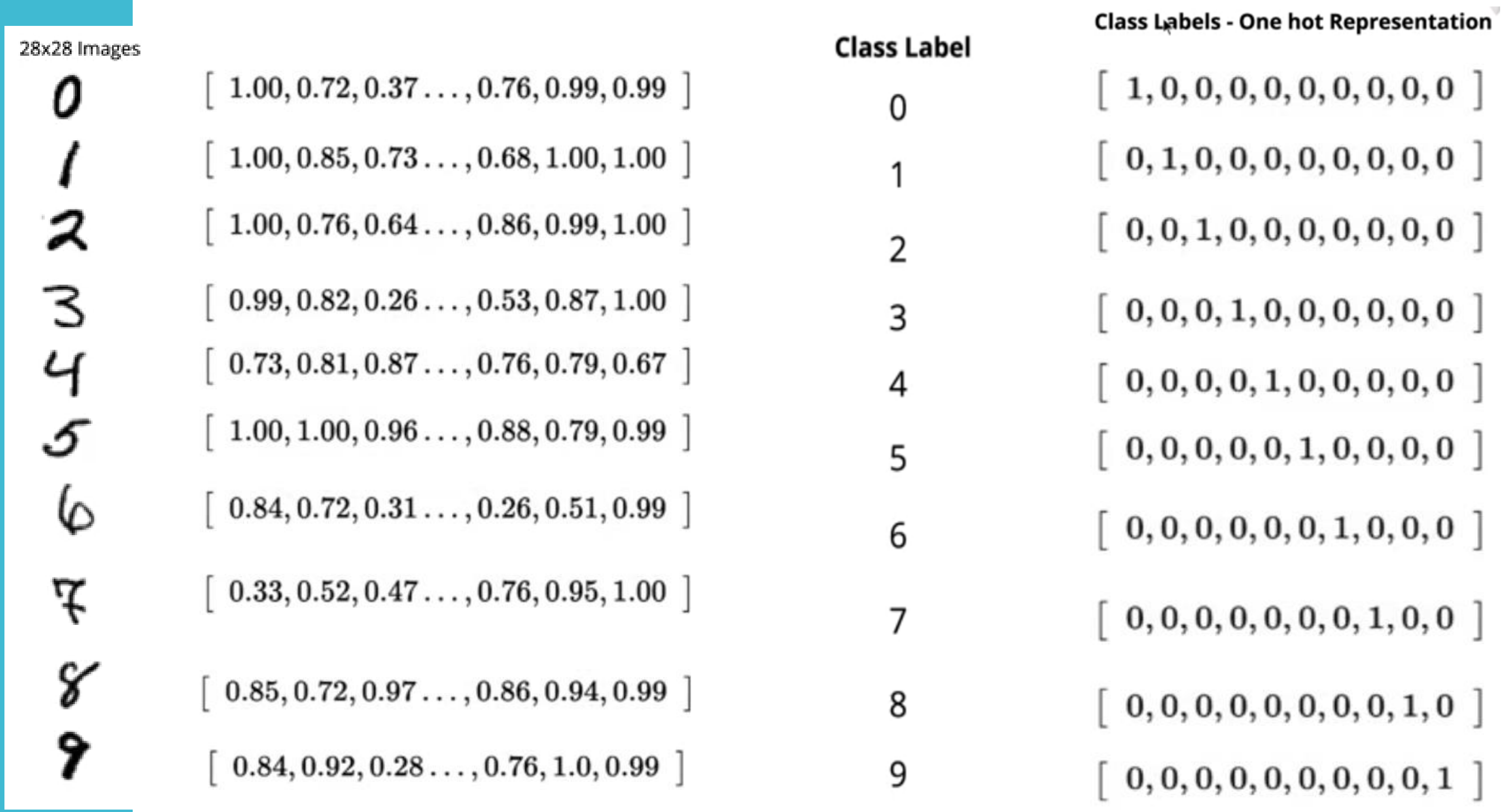
28x28 Images		Class Label
0	$[\ 1.00, 0.72, 0.37 \ldots, 0.76, 0.99, 0.99\ ]$	0
1	$[\ 1.00, 0.85, 0.73 \ldots, 0.68, 1.00, 1.00\ ]$	1
2	$[\ 1.00, 0.76, 0.64 \ldots, 0.86, 0.99, 1.00\ ]$	2
3	$[\ 0.99, 0.82, 0.26 \ldots, 0.53, 0.87, 1.00\ ]$	3
4	$[\ 0.73, 0.81, 0.87 \ldots, 0.76, 0.79, 0.67\ ]$	4
5	$[\ 1.00, 1.00, 0.96 \ldots, 0.88, 0.79, 0.99\ ]$	5
6	$[\ 0.84, 0.72, 0.31 \ldots, 0.26, 0.51, 0.99\ ]$	6
7	$[\ 0.33, 0.52, 0.47 \ldots, 0.76, 0.95, 1.00\ ]$	7
8	$[\ 0.85, 0.72, 0.97 \ldots, 0.86, 0.94, 0.99\ ]$	8
9	$[\ 0.84, 0.92, 0.28 \ldots, 0.76, 1.0, 0.99\ ]$	9

Class labels can be represented as one hot vectors

How can we represent MNIST images as a Vector?

Data and Task

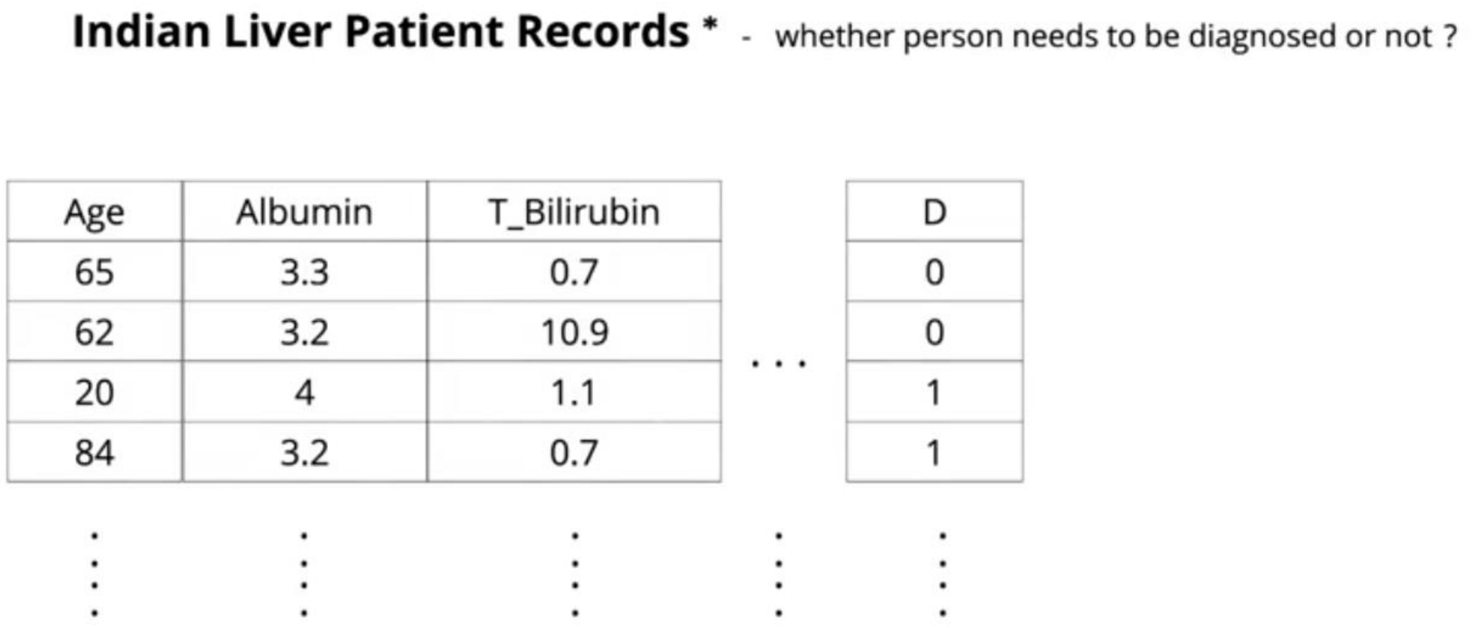
Data: MNIST Images
Task: MCC



28x28 Images		Class Label	Class Labels - One hot Representation
0	$[\,1.00, 0.72, 0.37\ldots, 0.76, 0.99, 0.99\,]$	0	$[\,1,0,0,0,0,0,0,0,0,0\,]$
1	$[\,1.00, 0.85, 0.73\ldots, 0.68, 1.00, 1.00\,]$	1	$[\,0,1,0,0,0,0,0,0,0,0\,]$
2	$[\,1.00, 0.76, 0.64\ldots, 0.86, 0.99, 1.00\,]$	2	$[\,0,0,1,0,0,0,0,0,0,0\,]$
3	$[\,0.99, 0.82, 0.26\ldots, 0.53, 0.87, 1.00\,]$	3	$[\,0,0,0,1,0,0,0,0,0,0\,]$
4	$[\,0.73, 0.81, 0.87\ldots, 0.76, 0.79, 0.67\,]$	4	$[\,0,0,0,0,1,0,0,0,0,0\,]$
5	$[\,1.00, 1.00, 0.96\ldots, 0.88, 0.79, 0.99\,]$	5	$[\,0,0,0,0,0,1,0,0,0,0\,]$
6	$[\,0.84, 0.72, 0.31\ldots, 0.26, 0.51, 0.99\,]$	6	$[\,0,0,0,0,0,0,1,0,0,0\,]$
7	$[\,0.33, 0.52, 0.47\ldots, 0.76, 0.95, 1.00\,]$	7	$[\,0,0,0,0,0,0,0,1,0,0\,]$
8	$[\,0.85, 0.72, 0.97\ldots, 0.86, 0.94, 0.99\,]$	8	$[\,0,0,0,0,0,0,0,0,1,0\,]$
9	$[\,0.84, 0.92, 0.28\ldots, 0.76, 1.0, 0.99\,]$	9	$[\,0,0,0,0,0,0,0,0,0,1\,]$

How can we represent MNIST images as a Vector?

Data: Indian Liver Patient Records
Task: Binary Classification



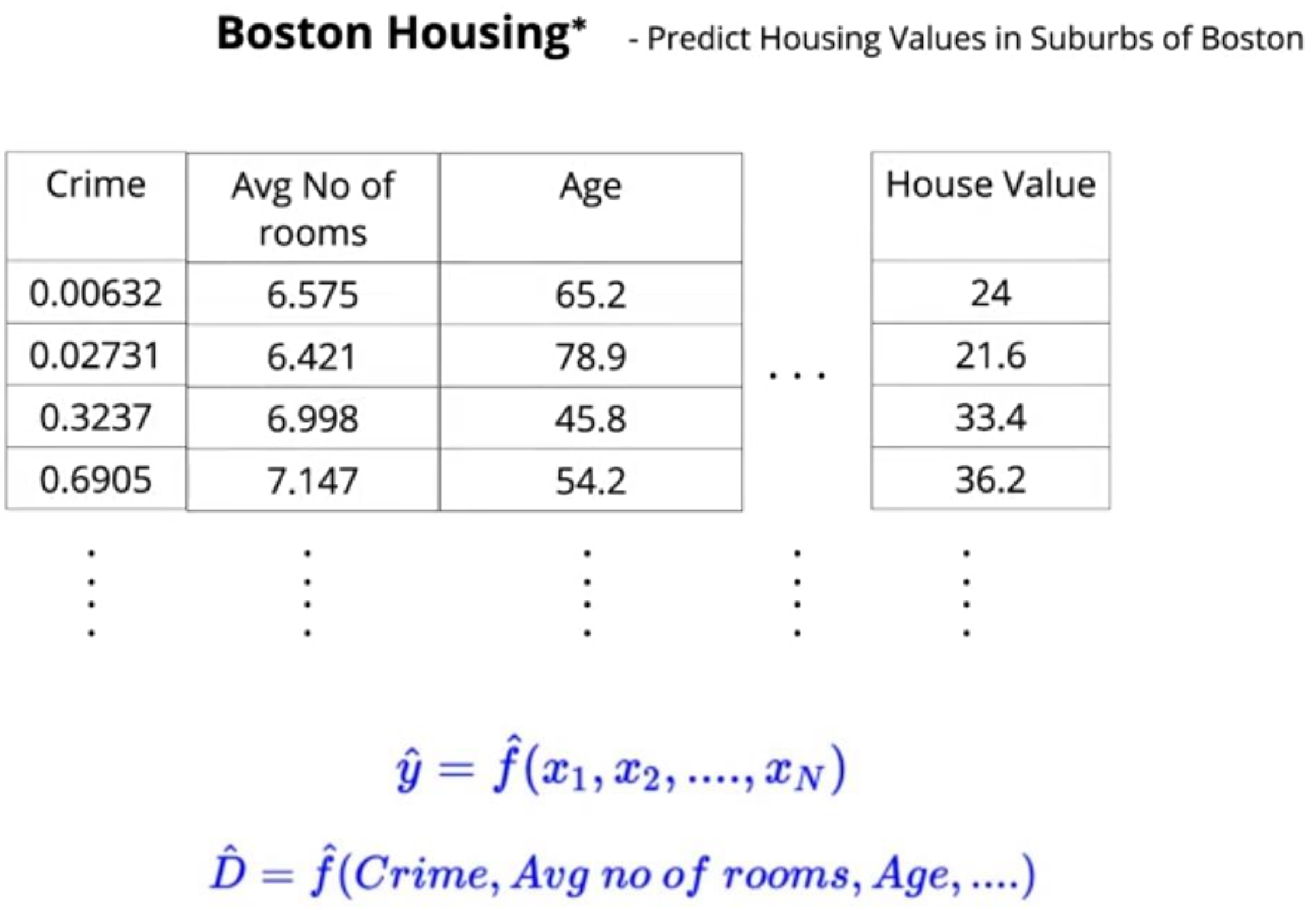
Indian Liver Patient Records * - whether person needs to be diagnosed or not ?

Age	Albumin	T_Bilirubin		D
65	3.3	0.7		0
62	3.2	10.9	. . .	0
20	4	1.1		1
84	3.2	0.7		1
⋮	⋮	⋮	⋮	⋮

Reference: https://www.kaggle.com/datasets/uciml/indian-liver-patient-records

Data: Boston Housing

Task: Regression



Boston Housing* - Predict Housing Values in Suburbs of Boston

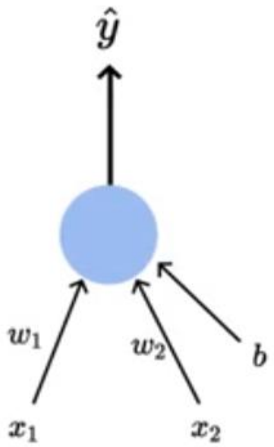
Crime	Avg No of rooms	Age		House Value
0.00632	6.575	65.2		24
0.02731	6.421	78.9	. . .	21.6
0.3237	6.998	45.8		33.4
0.6905	7.147	54.2		36.2
⋮	⋮	⋮	⋮	⋮

$$\hat{y} = \hat{f}(x_1, x_2, ..., x_N)$$

$$\hat{D} = \hat{f}(Crime, Avg\ no\ of\ rooms, Age, ...)$$

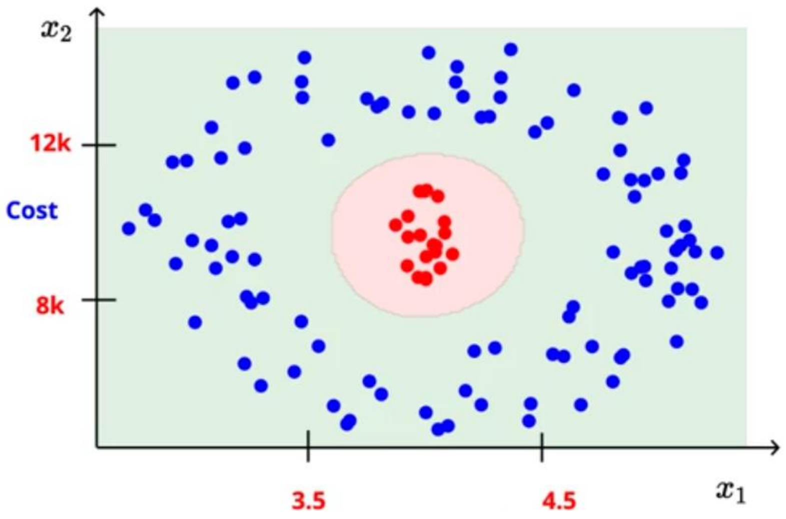
Reference: https://www.kaggle.com/datasets/altavish/boston-housing-dataset

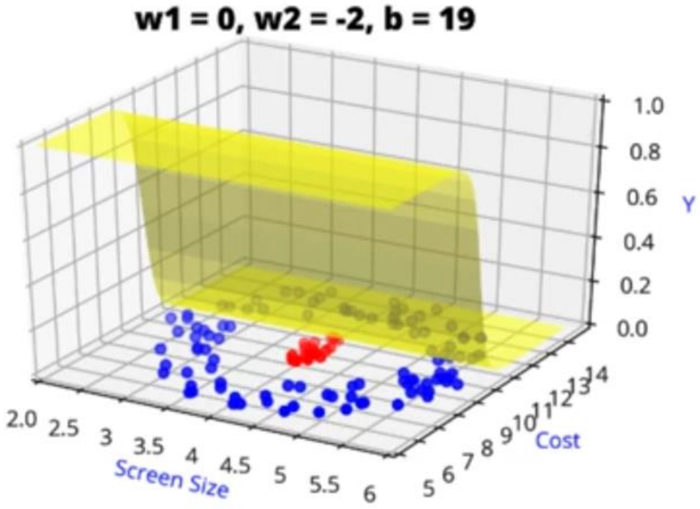
Model: How to build complex functions using Deep Neural Network?



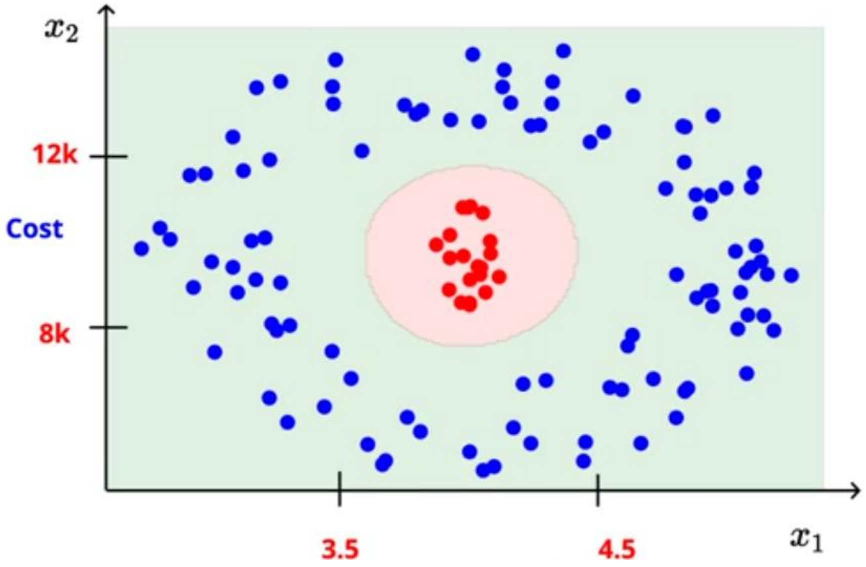
$$\hat{y} = f(x_1, x_2)$$

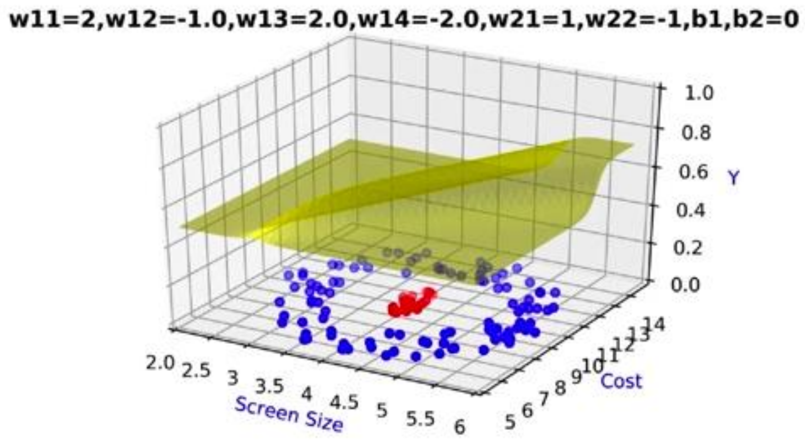
$$\hat{y} = \frac{1}{1+e^{-(w_1 * x_1 + w_2 * x_2 + b)}}$$

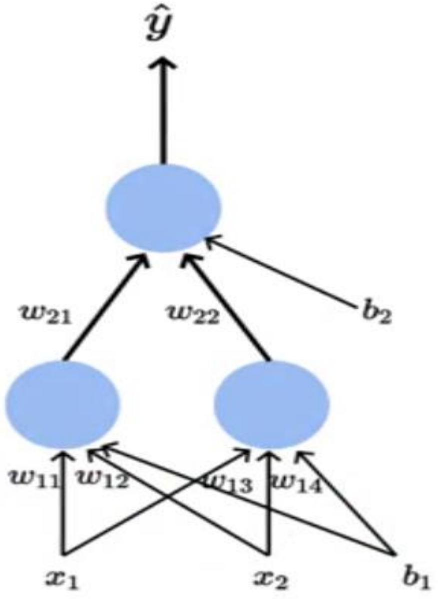



w1 = 0, w2 = -2, b = 19

Model: How to build complex functions using Deep Neural Network?





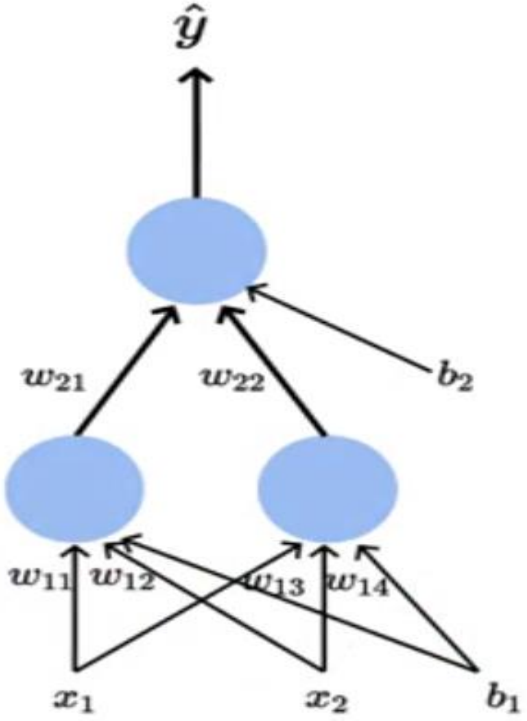


$$h_1 = f_1(x_1, x_2) \qquad h_1 = \frac{1}{1+e^{-(w_{11}*x_1+w_{12}*x_2+b_1)}}$$

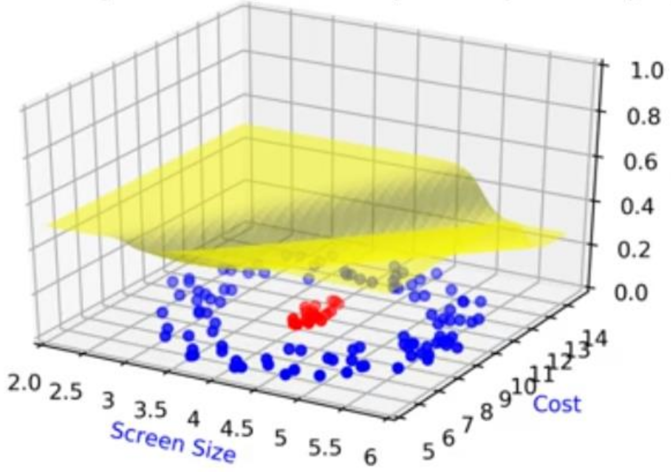
$$h_2 = f_2(x_1, x_2) \qquad h_2 = \frac{1}{1+e^{-(w_{13}*x_1+w_{14}*x_2+b_1)}}$$

$$\hat{y} = g(h_1, h_2) \qquad \hat{y} = \frac{1}{1+e^{-(w_{21}*h_1+w_{22}*h_2+b_2)}}$$

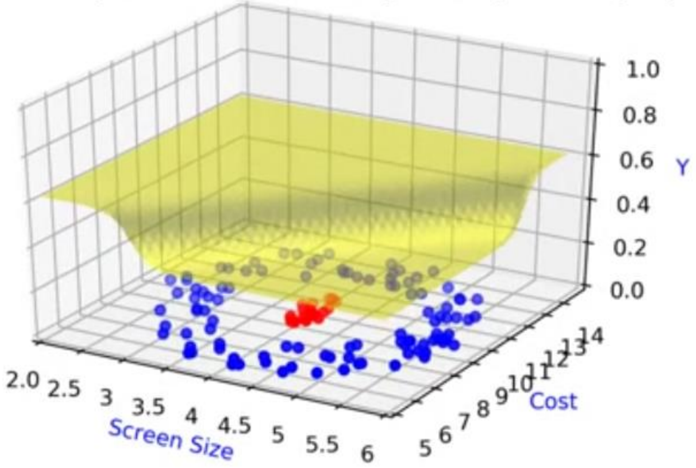
$$= \frac{1}{1+e^{-(w_{21}*(\frac{1}{1+e^{-(w_{11}*x_1+w_{12}*x_2+b_1)}})+w_{22}*(\frac{1}{1+e^{-(w_{13}*x_1+w_{14}*x_2+b_1)}})+b_2)}}$$



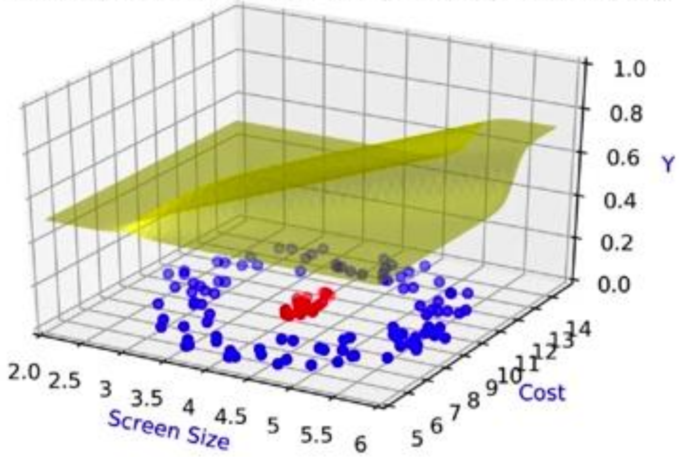
w11=-2,w12=1.0,w13=-1.5,w14=1.5,w21=1,w22=-1,b1,b2=0



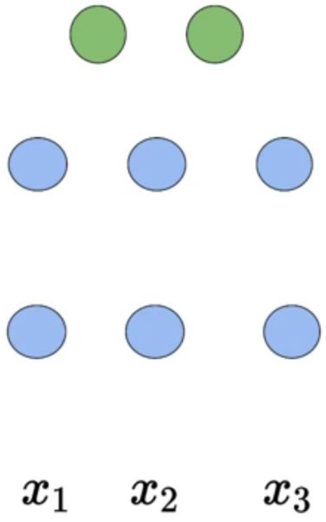
w11=0,w12=0.0,w13=2.0,w14=-2.0,w21=1,w22=-1,b1,b2=0

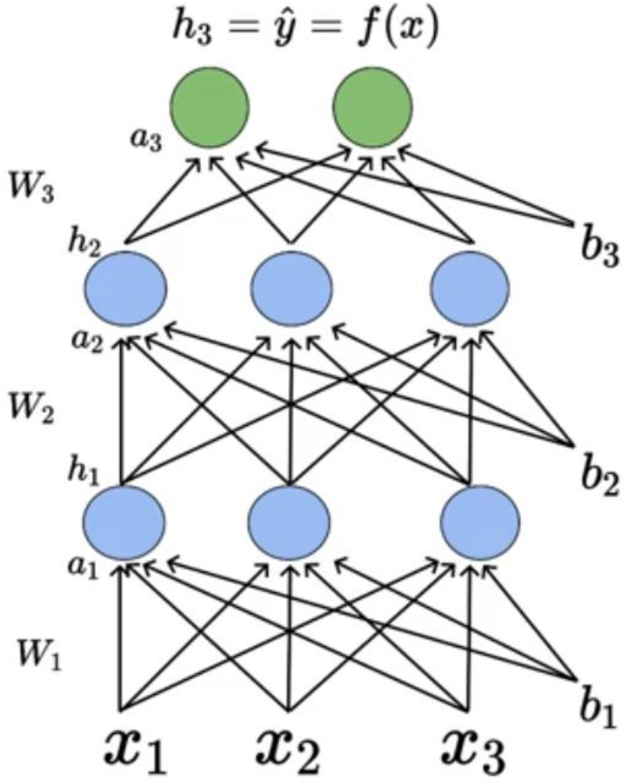


w11=2,w12=-1.0,w13=2.0,w14=-2.0,w21=1,w22=-1,b1,b2=0



Model





Model

- The pre-activation at layer 'i' is given by

$$a_i(x) = W_i h_{i-1}(x) + b_i$$

- The activation at layer 'i' is given by

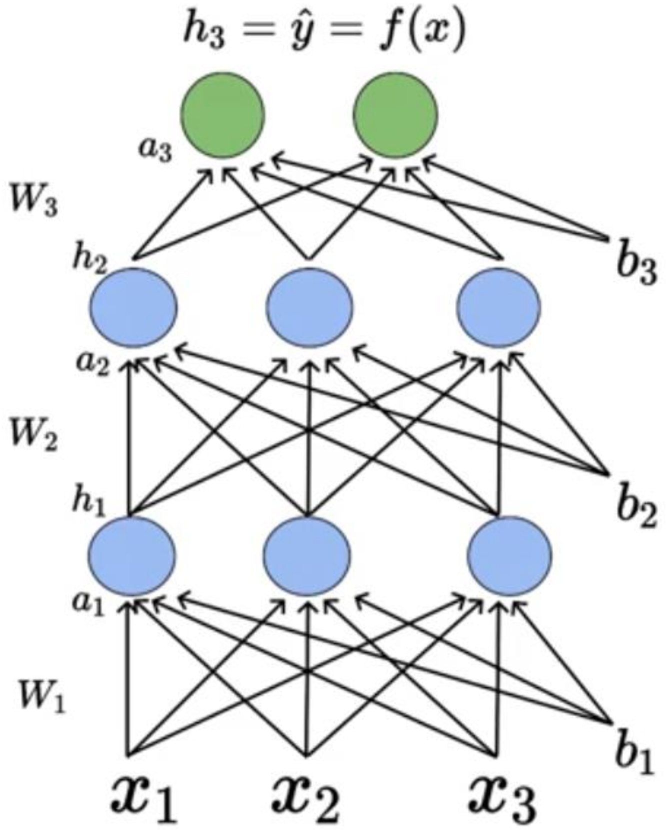
$$h_i(x) = g(a_i(x))$$

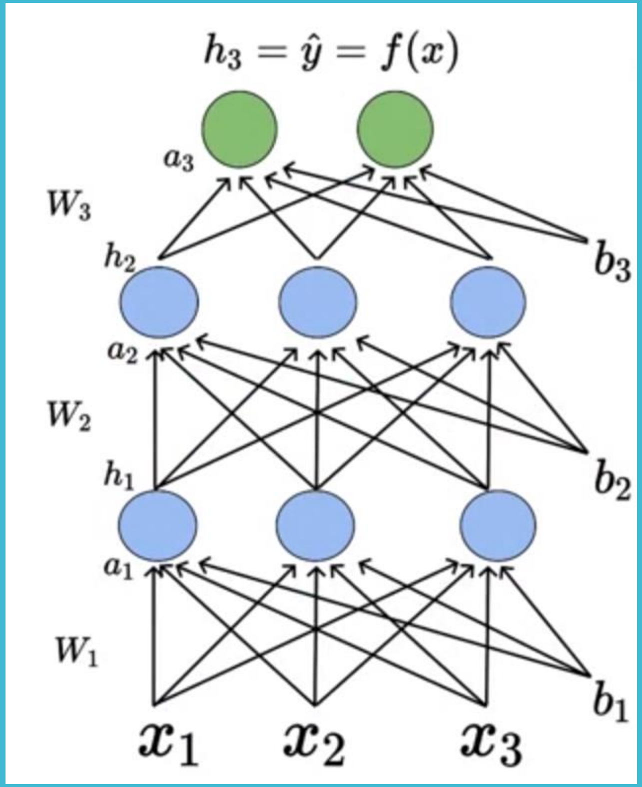
where 'g' is called as the activation function

- The activation at output layer 'L' is given by

$$f(x) = h_L = O(a_L)$$

where 'O' is called as the output activation function





$$W_1 = \begin{bmatrix} w_{1\,1\,1} & w_{1\,1\,2} & . & . & . & w_{1\,1\,99} & w_{1\,1\,100} \\ w_{1\,2\,1} & w_{1\,2\,2} & . & . & . & w_{1\,2\,99} & w_{1\,2\,100} \\ . & . & . & . & . & . & . \\ . & . & . & . & . & . & . \\ w_{1\,10\,1} & w_{1\,10\,2} & . & . & . & w_{1\,10\,99} & w_{1\,10\,100} \end{bmatrix} \qquad X = \begin{bmatrix} x_1 \\ x_2 \\ . \\ . \\ x_{100} \end{bmatrix}$$

$$a_{1\,1} = w_{1\,1\,1} * x_1 + w_{1\,1\,2} * x_2 + w_{1\,1\,3} * x_3 + ... + w_{1\,1\,100} * x_{100} + b_{11}$$

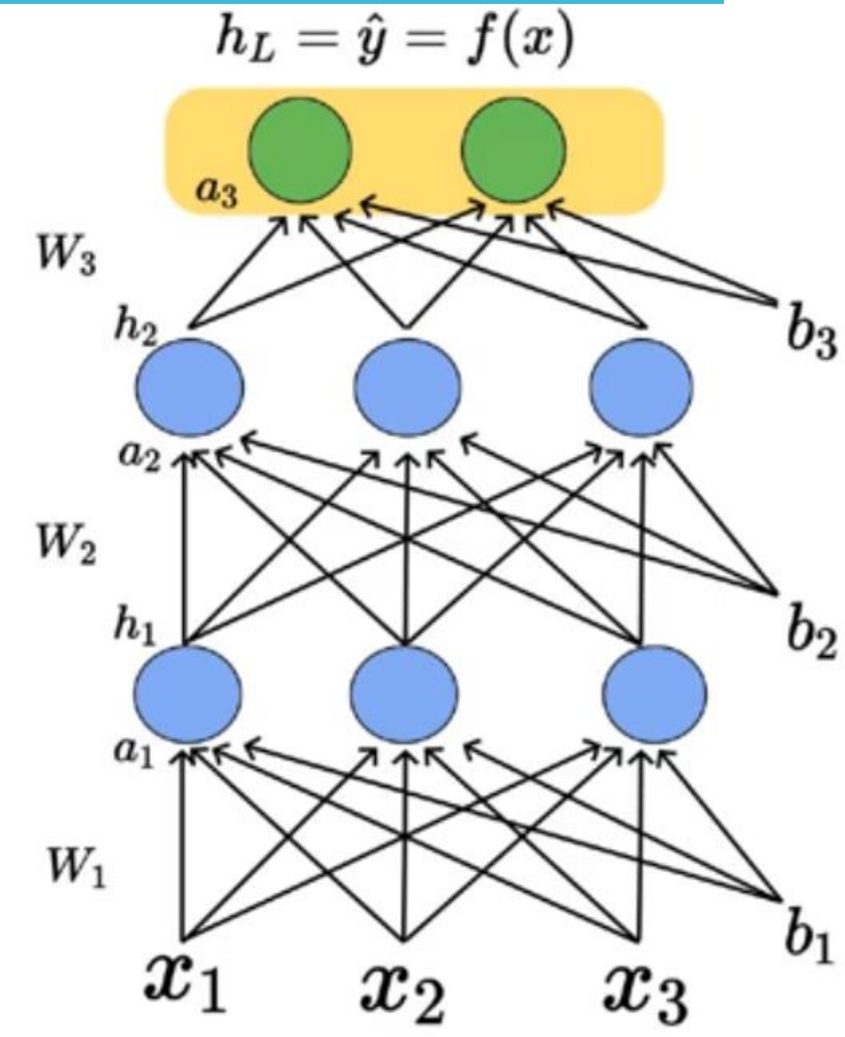
$$a_{1\,2} = w_{1\,2\,1} * x_1 + w_{1\,2\,2} * x_2 + w_{1\,2\,3} * x_3 + ... + w_{1\,2\,100} * x_{100} + b_{12}$$

$$\begin{matrix} . & . & . \\ . & . & . \end{matrix}$$

$$a_{1\,10} = w_{1\,10\,1} * x_1 + w_{1\,10\,2} * x_2 + w_{1\,10\,3} * x_3 + ... + w_{1\,10\,100} * x_{100} + b_{1,10}$$

$$a_1 = W_1 * x + b$$

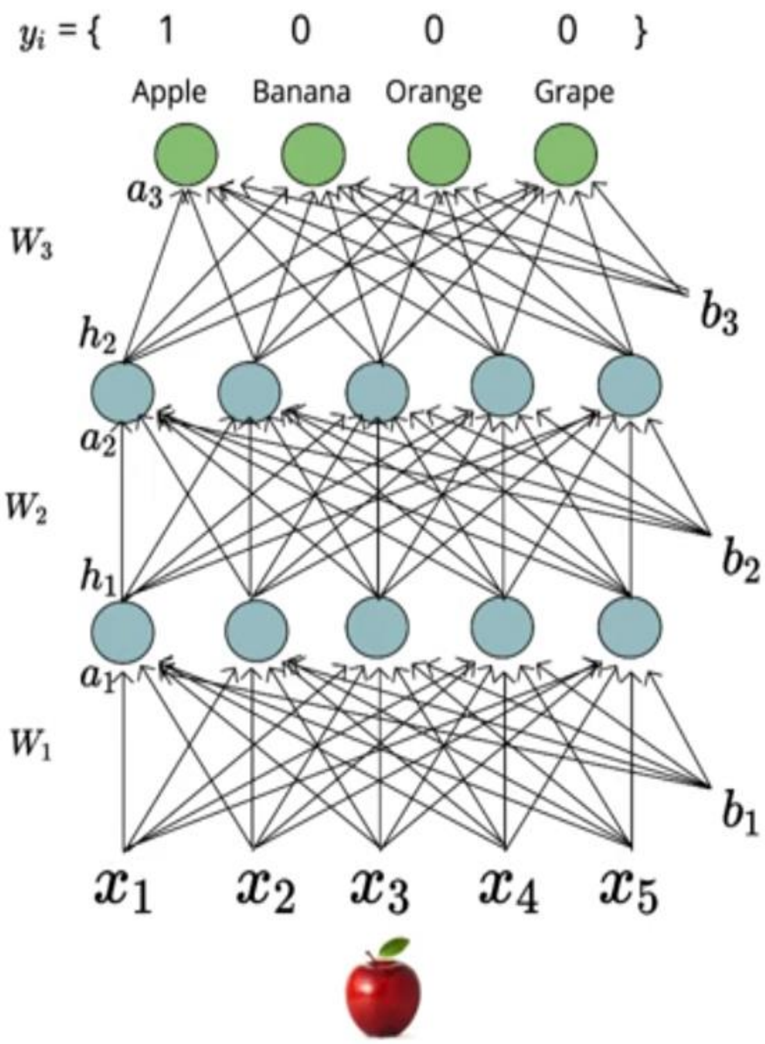
$$h_1 = g(a_1)$$

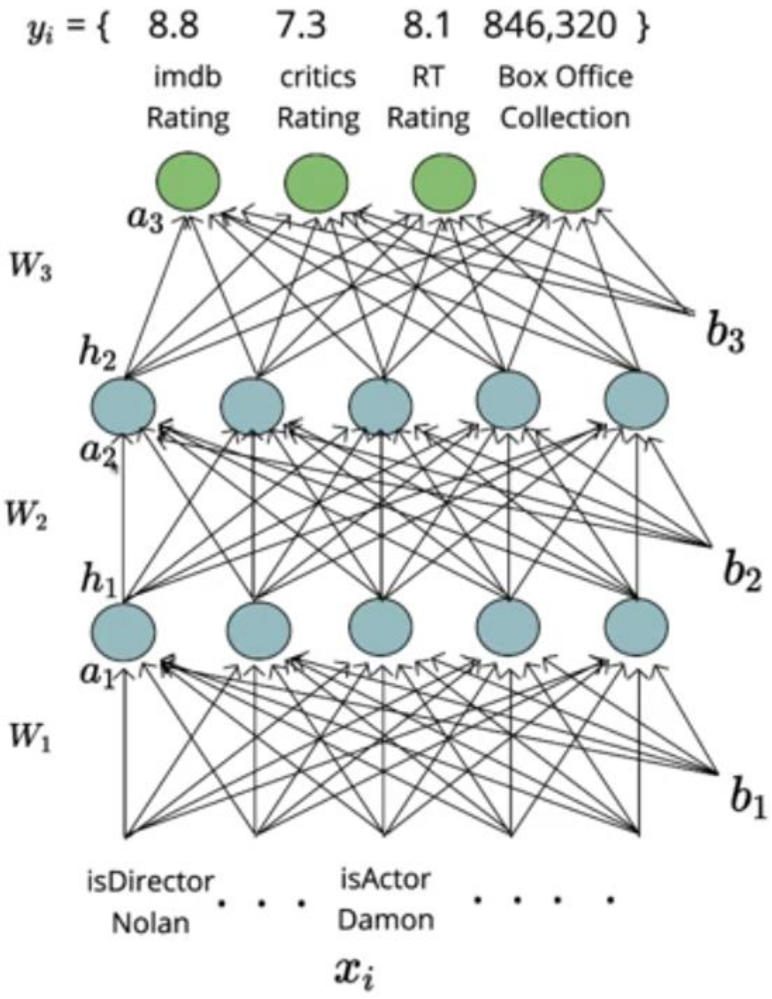


$$\hat{y} = f(x) = O(W_3 g(W_2 g(W_1 x + b_1) + b_2) + b_3)$$

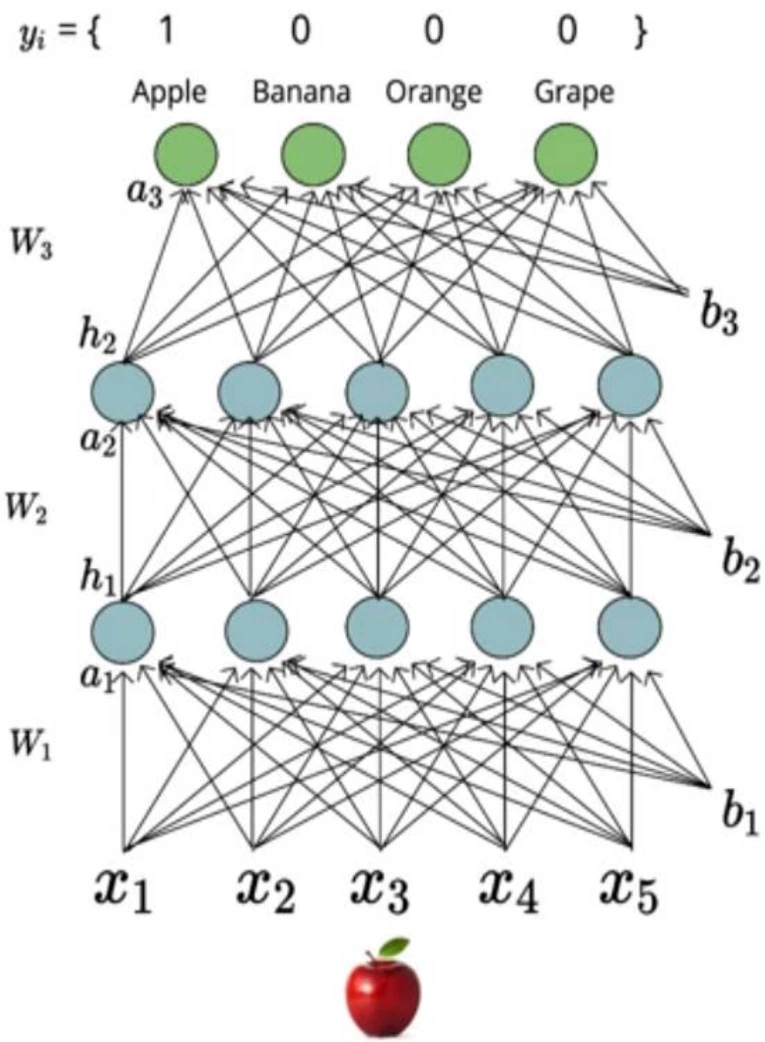
How do we decide the output layer?

Output Activation function is chosen depending on the task at hand (can be a softmax, linear)





Output Activation function is chosen depending on the task at hand (can be a softmax, linear)

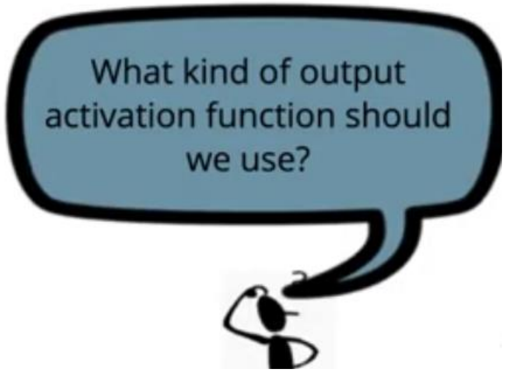


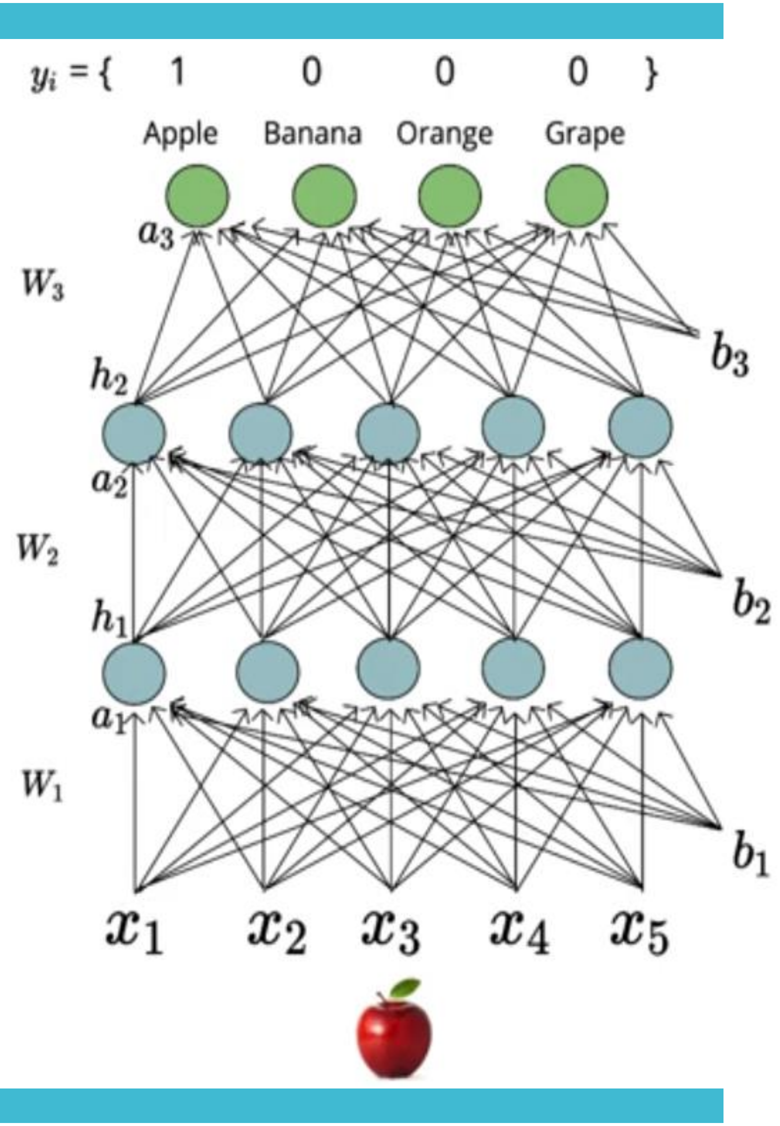
True Output :

$\hat{y}$ = { 1, 0, 0, 0 }

Predicted Output :

$\hat{y}$ = { 0.64, 0.03, 0.26, 0.07 }





$$W_3 = \begin{bmatrix} w_{3\,1\,1} & w_{3\,1\,2} & . & . & . & w_{3\,1\,10} \\ w_{3\,2\,1} & w_{3\,2\,2} & . & . & . & w_{3\,2\,10} \\ w_{3\,3\,1} & w_{3\,3\,2} & . & . & . & w_{3\,3\,10} \\ w_{3\,4\,1} & w_{3\,4\,2} & . & . & . & w_{3\,4\,10} \end{bmatrix} \qquad h_2 = \begin{bmatrix} h_{2\,1} \\ h_{2\,2} \\ . \\ . \\ h_{2\,10} \end{bmatrix}$$

$$a_{3\,1} = w_{3\,1\,1} * h_{2\,1} + w_{3\,1\,2} * h_{2\,2} + w_{3\,1\,3} * h_{2\,3} + ... + w_{3\,1\,10} * h_{2\,10} + b_{31}$$

$$a_{3\,2} = w_{3\,2\,1} * h_{2\,1} + w_{3\,2\,2} * h_{2\,2} + w_{3\,2\,3} * h_{2\,3} + ... + w_{3\,2\,10} * h_{2\,10} + b_{32}$$

$$a_{3\,3} = w_{3\,3\,1} * h_{2\,1} + w_{3\,3\,2} * h_{2\,2} + w_{3\,3\,3} * h_{2\,3} + ... + w_{3\,3\,10} * h_{2\,10} + b_{33}$$

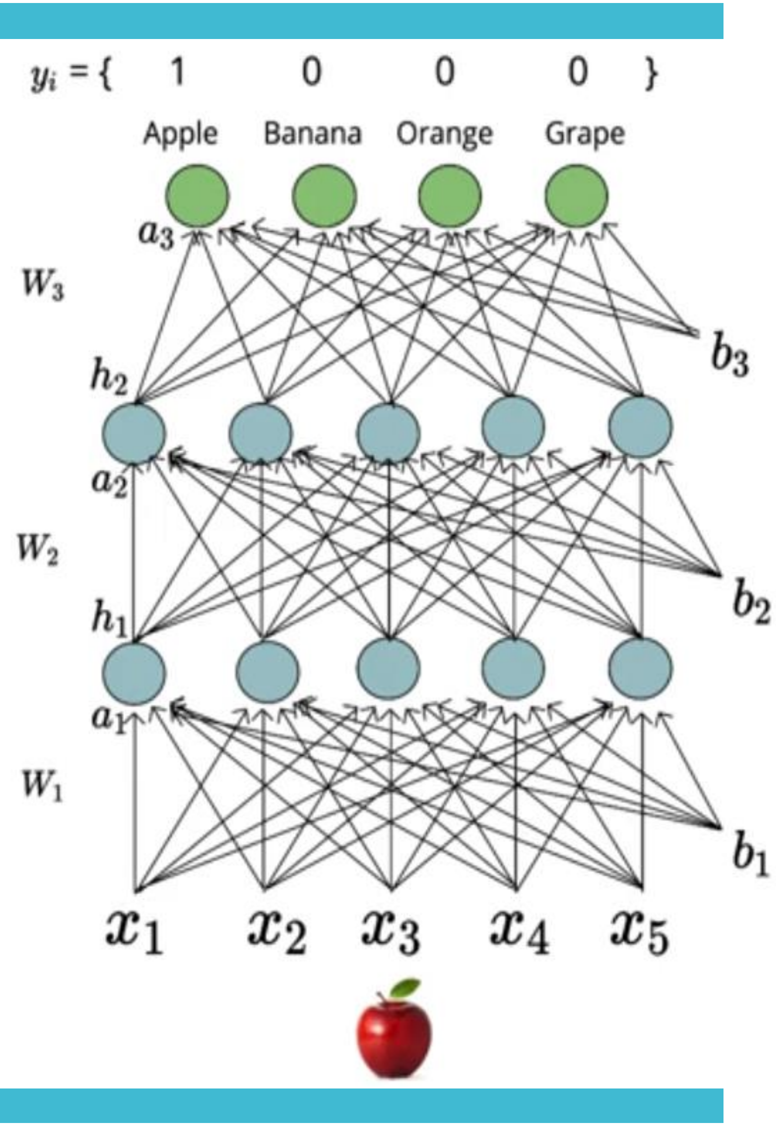
$$a_{3\,4} = w_{3\,4\,1} * h_{2\,1} + w_{3\,4\,2} * h_{2\,2} + w_{3\,4\,3} * h_{2\,3} + ... + w_{3\,4\,10} * h_{2\,10} + b_{34}$$

$$a_3 = W_3 * h_2 + b_3$$

$$\hat{y}_1 = O(a_{31}) \qquad \hat{y}_2 = O(a_{32}) \qquad \hat{y}_3 = O(a_{33}) \qquad \hat{y}_4 = O(a_{34})$$

What is the output layer for classification problems?

$$Say\ a_3 = [\,3\ \ 4\ \ 10\ \ 3\,]$$



Output Activation Function has to be chosen such that output is probability

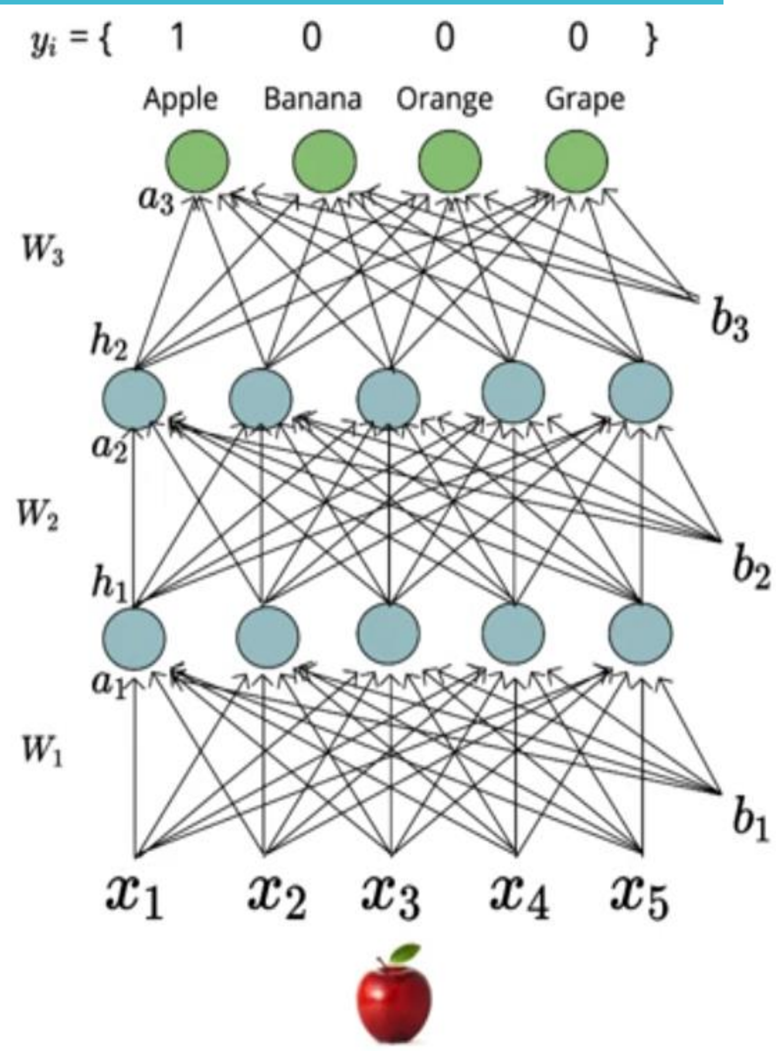
$$\hat{y}_1 = \frac{3}{(3+4+10+3)} = 0.15$$

$$\hat{y}_2 = \frac{4}{(3+4+10+3)} = 0.20$$

$$\hat{y}_3 = \frac{10}{(3+4+10+3)} = 0.50$$

$$\hat{y}_4 = \frac{3}{(3+4+10+3)} = 0.15$$

What is the output layer for classification problems?



$$Say\ for\ other\ input\ a_3 = [\,7 \quad -2 \;\; 4 \;\; 1\,]$$

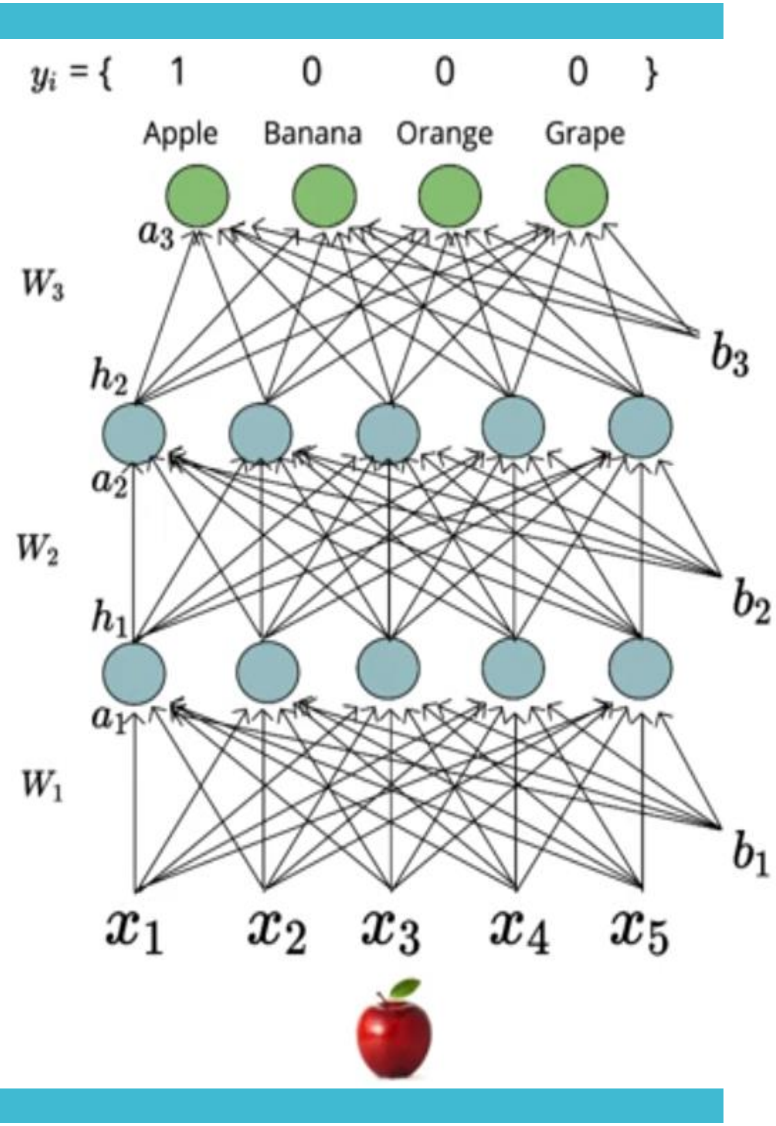
$$\hat{y}_1 = \frac{7}{(7 + (-2) + 4 + 1)} = 0.70$$

$$\hat{y}_2 = \frac{-2}{(7 + (-2) + 4 + 1)} = -0.20$$ ✖

$$\hat{y}_3 = \frac{4}{(7 + (-2) + 4 + 1)} = 0.40$$

$$\hat{y}_4 = \frac{1}{(7 + (-2) + 4 + 1)} = 0.10$$

What is the output layer for classification problems?



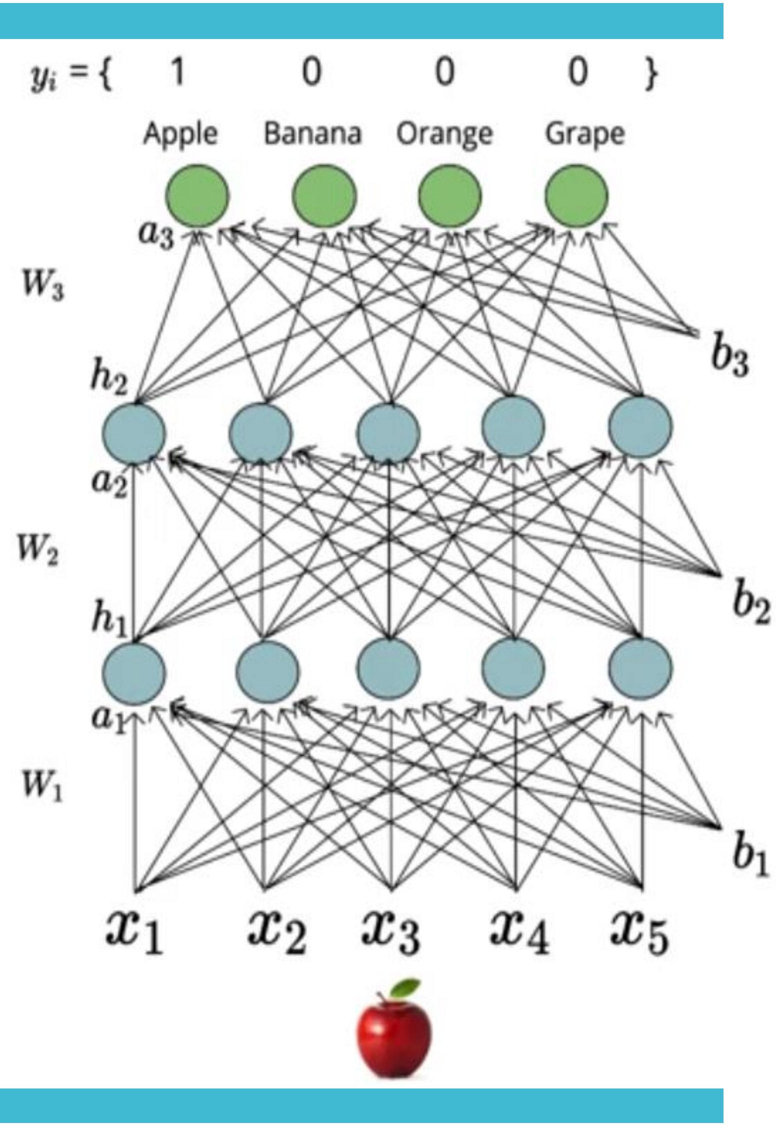
$$h = [\ h_1\ \ h_2\ \ h_3\ \ h_4\ ]$$

$$softmax(h) = [softmax(h_1)\ \ softmax(h_2)\ \ softmax(h_3)\ \ softmax(h_4)]$$

$$softmax(h) = \left[\frac{e^{h_1}}{\sum_{j=1}^{4} e^{h_j}} \quad \frac{e^{h_2}}{\sum_{j=1}^{4} e^{h_j}} \quad \frac{e^{h_3}}{\sum_{j=1}^{4} e^{h_j}} \quad \frac{e^{h_4}}{\sum_{j=1}^{4} e^{h_j}} \right]$$

$softmax(h_i)$ is the i^{th} element of softmax output

What is the output layer for classification problems?



$$a_1 = W_1 * x \quad \mathbf{+ b1} \qquad h_1 = g(a_1)$$

$$a_2 = W_2 * h_1 \quad \mathbf{+ b2} \qquad h_2 = g(a_2)$$

$$a_3 = W_3 * h_2 \quad \mathbf{+ b3} \qquad \hat{y} = softmax(a_3)$$

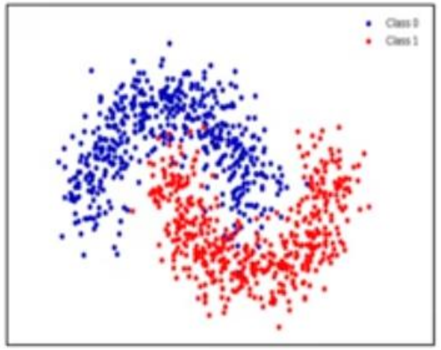
What is the output layer for classification problems?

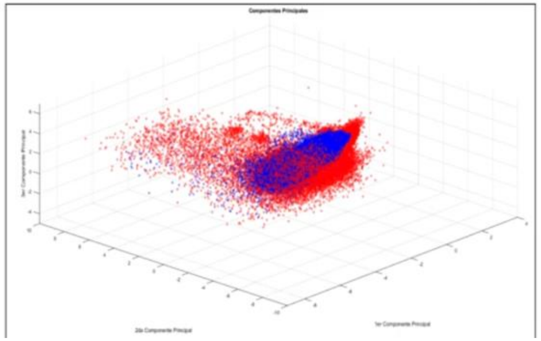
How would you deal with extreme non-linearity?

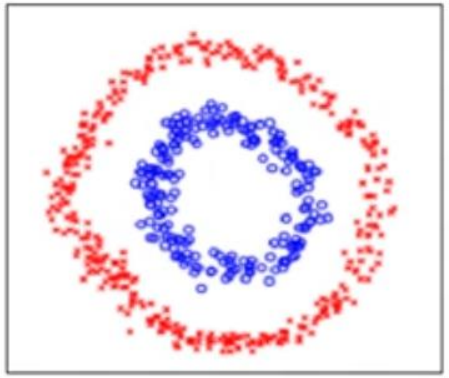
x_1	x_2
-	-
-	-
-	-

x_1	x_2	x_3
-	-	-
-	-	-
-	-	-

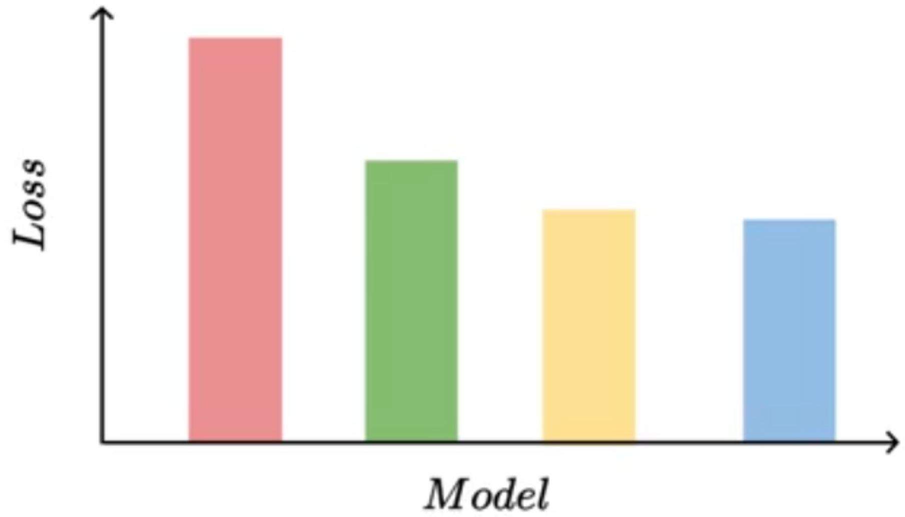
x_1	x_2	x_3	x_4
-	-	-	-
-	-	-	-
-	-	-	-

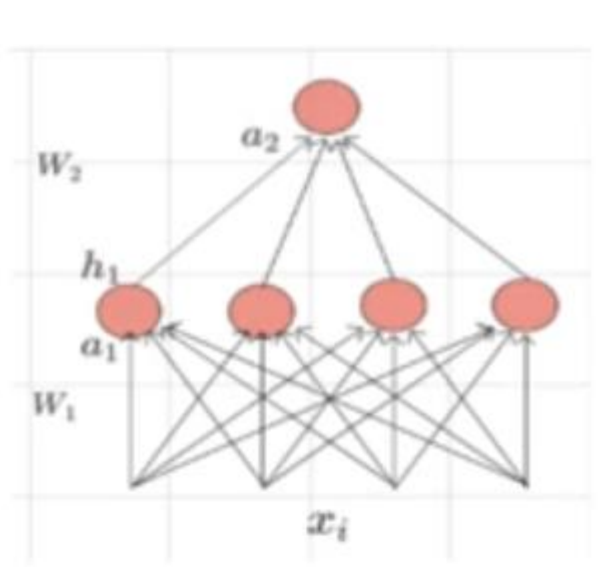


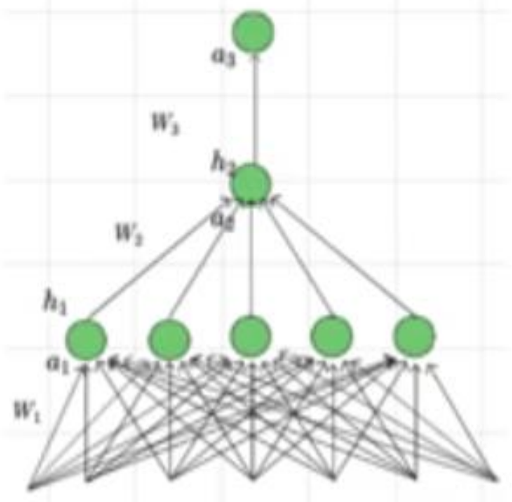


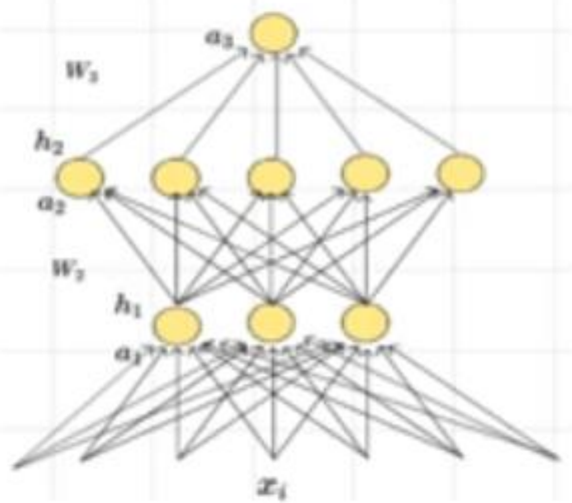


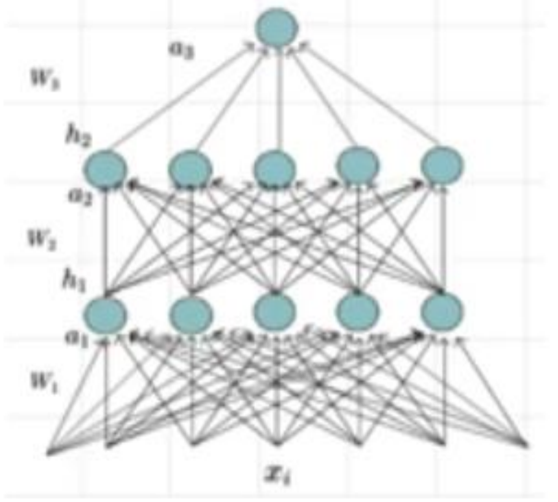
How would you deal with extreme non-linearity?




a_2
W_2
h_1
a_1
W_1
x_i


a_3
W_3
h_2
a_2
W_2
h_1
a_1
W_1


a_3
W_3
h_2
a_2
W_2
h_1
a_1
x_i


a_3
W_3
h_2
a_2
W_2
h_1
a_1
W_1
x_i