Stochastic Gradient Descent Learning:

Gradient descent learning is based on identifying the minimum value of a function where the algorithm works with the network on one layer. The stochastic quality of the algorithm makes it neutral by randomly selecting the samples instead of an iterative process. The batch contains the total number of the sample where normal Gradient Descent choose the sample iteratively while Stochastic Gradient Descent picks the samples randomly. The learning decision surface is recognised where it makes the non-linear decision surface simpler. The non-linear decision surface is transformed into a linear hyperplane which simplifies the process and reduces the computational complexity. The mathematical annotation of Gradient Descent is given below:

$$(x(n), d_n) \text{ Or,}$$

$$\left((x_{1n}, \dots x_{mn}), d_n\right)$$

The equation defines the input values i.e. $(x_{1n}, \dots x_{mn})$ where d_n is the desired output. The learning rate of the equation is identified by μ.

Neural network learning rate:

During the implementation of neural networks when the weights of the nodes are adjusted learning rate is also specified. Learning rate affect the magnitude of the movement while it also reduces the value of loss function in neural networks. The purpose of applying the learning rate is to minimise the computational cost of neural networks. Furthermore, the parameters that the learning rate requires are the weights and biases. The values of weights and biases are fluctuated to adjust to the optimal value where the cost of the learning is the minimum.

$$W(k+1) = W(k) + \mu . \Delta W$$

The learning rate of the neural network is specified by μ in the above equation. ΔW are the weights assigned to the nodes. The updating speed of the parameters defined in the equation depends on the learning rate. The updating process is more frequent when opting for the high learning rate where the learning speed of the algorithm increases. With the progress in learning the where the training of the model progresses learning rate decreases. Furthermore, the convergence of the algorithm is slower when the specified learning rate is lower as compared to the high learning rate.

Multi-layered feed-forward neural network:

The neural networks are comprised of layers with an input and output layer that maps the input data points and output data points respectively while there can be multiple hidden layers in the middle of input and output layers. The structure of a multi-layer feed-forward network is based on an interconnected perceptron where computations are performed in a single layer to map the data points from input to output. In feedforward, only the output layer contains the activation calculation function where the structure of a simpler neural network can contain only two-layer i.e. input and output. Furthermore, the flow of data is one-directional in the multi-layered feed-forward network from the input layer towards the output layer.

Hidden nodes in a neural network:

The node or neurons that exist in between the input and output layer are the hidden nodes in a neural network. Hidden nodes are not accessed directly and weights are assigned to the nodes which directs them t=to output function through an activation function. The hidden layers can be considered a layer of function with mathematical calculation intended for specific results.

Output nodes in a neural network:

The aggregation of calculations performed by input and hidden nodes is presented in the form of output nodes. The response of the network is gauged using the output layer.

Backpropagation rule:

There are two types of network rules that are confused with each other i.e. feed-forward network and backpropagation network where both are famous approaches to revise and renew the weights in the network. In the backpropagation technique, the gradient descent principle is computed for every node that is the derivative intended to update the value of the weights. The backpropagation contributes to updating the value of the weights that reflect in squared loss and error functions of the algorithm. Apart from that, feed-forward consists of two segments i.e. forward and backward phases. The input is feed to the network in the forward's phase which leads to the calculation of the squared loss function. Later to optimise the values of the weights in the network backpropagation is used to redefine the optimal weights to reduce the value of the loss. The main task of the backpropagation is a calculation of the first derivative for every weight in the function.

The objective function of the multi-layer neural network:

The process of training the machine learning model reduces the total value of mean squared effort in the multi-layer neural network. The number of instances X(n) in the training set of network N defines the objective function as:

$$E(W) = \frac{1}{2N}\sum_{n}\sum_{j}(d_j(n) - o_j(n))2$$

D(n) is the objective function where $o_j(n)$ is the output of the nodes.

Momentum term in neural network weight updating:

The values of the previous round are used to update the values for the current round where the momentum term function in the neural network provides the values of the previous round.

$$\Delta w(K) = \propto \Delta w(K-1) + \mu\delta(n)x(n)$$

$\Delta w(K-1)$ represents the values of the former round where the second part of the equation $\mu\delta(n)x(n)$ demotes the process of updating the weight. Apart from that, the weights of the ongoing round are defined using the momentum constraint where 0≤α<1.

Question no. 2:

The objective function of the single-layer neural network of the provided figure can be defined as:

$$f(w_0, w_1, w_2) = \frac{1}{2} \times [d(n) - w_0 - w_1x_1(n) - w_2x_2(n)]2$$

To update the weights of the neural network gradient descent rule is followed which is explained below:

$$\frac{\partial f(w_0, w_1, w_2)}{\partial w_2} = \frac{\partial \frac{1}{2} \times [d(n) - w_0 - w_1x_1(n) - w_2x_2(n)]2}{\partial w_2}$$

$$= \frac{1}{2} \times 2[d(n) - w_0 - w_1x_1(n) - w_2x_2(n)] \times (-x_2(n))$$

$$= -[d(n) - w_0 - w_1x_1(n) - w_2x_2(n)]x_2(n)$$

The learning rate μ is defined with the equation for updating the weights for w2.

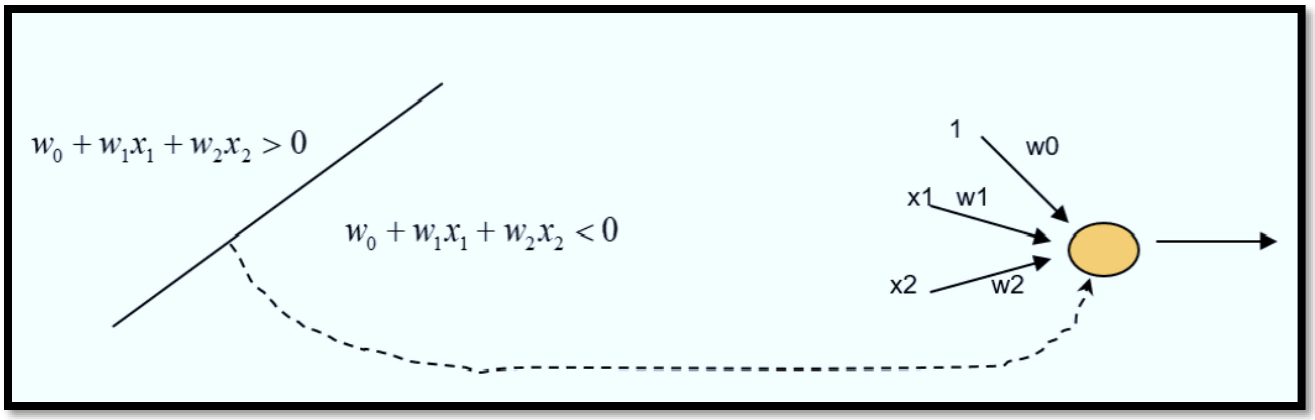
$$\Delta w_2 = -\mu\frac{\partial f(w)}{\partial w_2}$$

$$= \mu[d(n) - w_0 - w_1x_1(n) - w_2x_2(n)]x_2(n)$$

Question no.3:

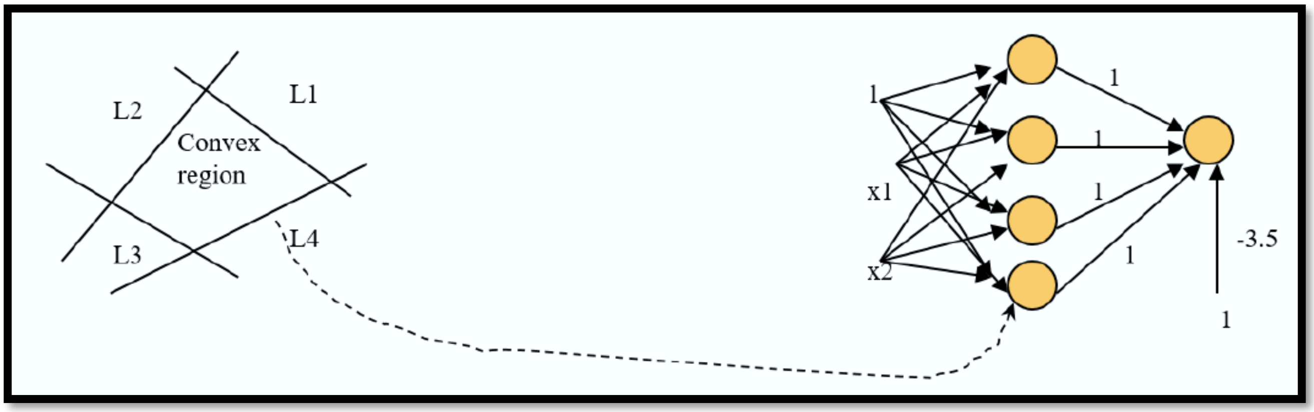
(a) Single-layer network:

In the single-layer network, the position of the point influences the network classification whether the point is up or down the line.



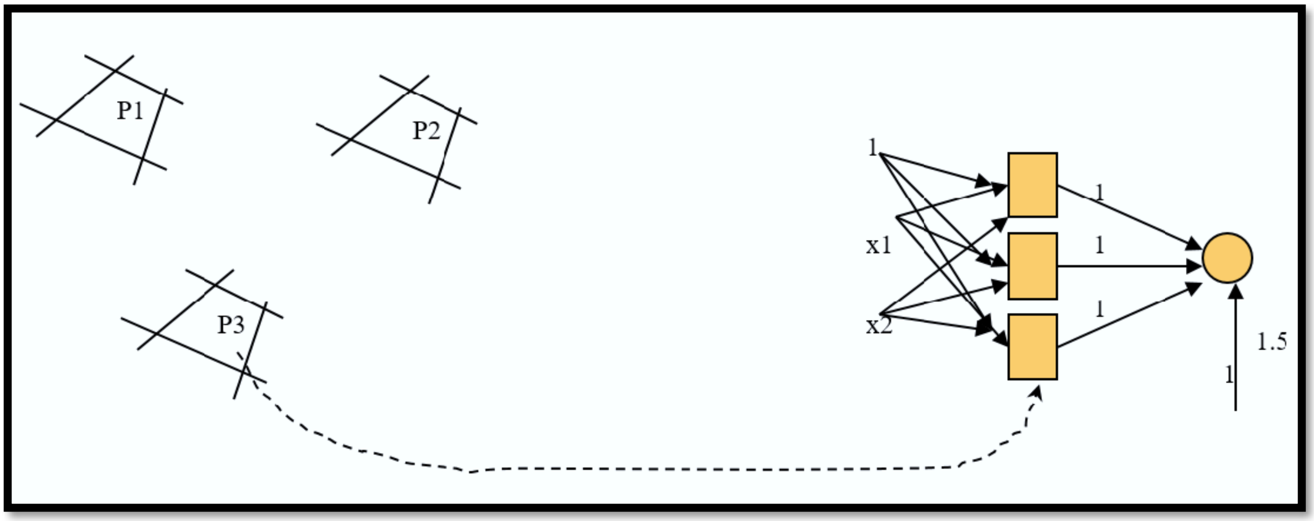
(b) Single hidden layer:

The convex region is considered where the position of the point is identified with respect to whether the point lies inside or outside of the convex region. The hidden node assumes a boundary line around the region to classify the data points.



(c) Neural network with two hidden layers:

The neural network with two hidden lines distributes the region into three parts by assuming three convex regions.



In the above figure, each box represents one convex region while the classification is performed based on the convex region.

Question 4:

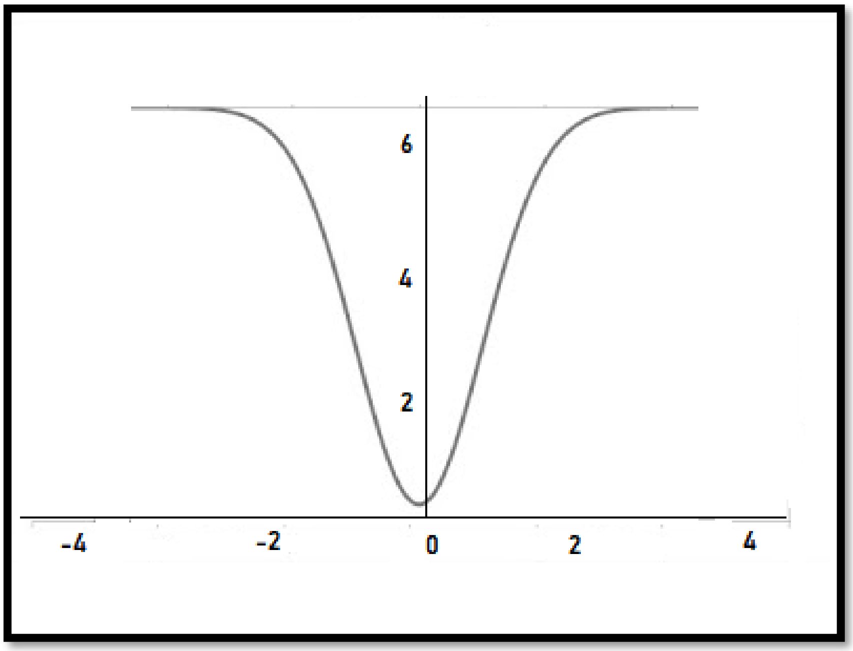


Figure 1: Quadratic function y=x2

The gradient descent calculates the derivative and here is the first derivative equation:

$$\frac{dy}{dx} = \frac{d(x^2)}{dx} = 2.x$$

The gradient at the point can be calculated by:
When x=1 the value of the derivative will be 2x1=2
When x=3 the value of the derivative will be 2x3=6
Similarly, when the gradient descent rule is applied the derivative can be computed for the next values of x using:

$$x(k+1) = x(k) + \mu(-gradient)$$

Considering the learning rate, the values are calculated using the gradient descent learning rule:

$$x(k+1) = x(k) + 0.1 \times (-6)$$

$$= 3.0 - 0.6$$

$$= 2.4$$

The derivation shows the next value which is 2.4.

Question.5:
A= 1.0
Learning rate µ=0.2
The sigmoid function is typically used for function activation.

$$\varphi(v_j) = O_j = \frac{1}{1+e^{-a\times j}}\ with\ a > 0$$

$$O_a(I_1) = \varphi\left(1+1+0.5 = \frac{1}{1+e^{-1\times 0-5}}\right) = 0.9241$$

$$O_a(I_1) = \varphi\left(1 + 1 + 0.5 = \frac{1}{1 + e^{-1\times 0-5}}\right) = 0.9241$$

$$O_c(I_1) = \varphi(1.0 + 1.0\ O_a(I_1) + 1.0\ O_b(I_1)$$

$$= \varphi(1.0 + 0.9241 + 0.941)$$

$$= \frac{1}{1 + e^{-1\times 2.8482}} = 0.9452$$

$$O_a(I_2) = \varphi(1 + 0 + 1)$$
$$\frac{1}{1 + e^{-1\times 2}} = 0.88079$$

$$O_b(I_2) = \varphi(1 + 0 + 1)$$
$$\frac{1}{1 + e^{-1\times 2}} = 0.88079$$
$$O_c I_2 = \varphi(1.0 + 1.0\ O_a(I_2) + 1.0\ O_b(I_2))$$

$$= \varphi(1.0 + 0.88079 + 0.88079)$$
$$\frac{1}{1 + e^{-1\times 2.7615}} = 0.9405$$

$$O_b(I_3) = \varphi(1 + 0.5 + 0.5)$$
$$\frac{1}{1 + e^{-1\times 2}} = 0.88079$$
$$O_b(I_3) = \varphi(1 + 0.5 + 0.5)$$
$$\frac{1}{1 + e^{-1\times 2}} = 0.88079$$
$$O_c I_3 = \varphi(1.0 + 1.0\ O_a(I_3) + 1.0\ O_b(I_3))$$
$$= \varphi(1.0 + 0.88079 + 0.88079)$$

$$\frac{1}{1 + e^{-1\times 2.7615}} = 0.9405$$

Mean square error:

$$E(W) = \frac{1}{2N}\sum_n \sum_j (d_j(n) - o_j(n))2$$
$$= \frac{1}{2\times 3}[(1 - 0.9452)^2 + (1 - 0.9452)^2 + (1 - 0.9452)^2]$$
$$= \frac{1}{6}[(0.003 + 0.8845 + 0.0035)]$$
$$= \frac{0.891}{6} = 0.1485$$

Updating weight:

$$\int c = a \times O_c(I_1).\,(1 - O_c I_1).\,(1 - O_c I_1)$$
$$= 1 \times 0.9452(1 - 0.9452)(1 - 0.9452)$$

$$= 0.9452 \times 0.0548 \times 0.0548$$
$$= 2.838473408E - 03 \; or \; 0.00283847$$

$$\int a = a \times O_a(I_1).(1 - O_a I_1).\int c$$
$$= 1 \times 0.9241(1 - 0.9241)0.2 \times 0.00283847$$
$$= 1 \times 0.9241 \times 0.0759 \times 0.2 \times 0.00283847$$
$$= 3.981759733E - 05$$

$$\int b = a \times O_b(I_1).(1 - O_b I_1).\int c$$
$$= 1 \times 0.9241(1 - 0.9241)0.2 \times 0.00283847$$
$$= 1 \times 0.9241 \times 0.0759 \times 0.2 \times 0.00283847$$
$$= 3.981759733E - 05$$

$$w_{0a} = w_{0a} + \mu.\int a\,.1$$
$$= 1 + 0.2 \times 3.981759733E - 05$$

$$= 1 + 7.963519466E = 06$$

$$= 1.000007964$$

$$w_{1a} = w_{1a} + \mu.\int a\,.1$$
$$= 1 + 0.2 \times 3.981759733E - 05$$

$$= 1 + 7.963519466E = 06$$

$$= 1.000007964$$

$$w_{2a} = w_{2a} + \mu.\int a\,.0.5$$
$$= 1 + 0.2(3.981759733E - 05) \times 0.2$$

$$= 1.0000039$$

$$w_{0b} = w_{0b} + \mu.\int b\,.1$$
$$= 1 + 0.2 \times 3.981759733E - 05$$

$$= 1.000007964$$

$$w_{1b} = w_{1b} + \mu . \int b\,.\,1$$
$$= 1 + 0.2 \times 3.981759733E - 05$$

$$= 1.000007964$$
$$w_{2b} = w_{2b} + \mu . \int b\,.\,1$$
$$= 1 + 0.2 \times 3.981759733E - 05 \times 0.2$$

$$= 1.0000039$$

Question 6:

Given:

$$t_1 = (-1, 0.5)$$

$$t_1 = (1, -0.5)$$

Gaussian RBF kernel j=1

Instance index	$\boldsymbol{\varphi 1}()$	$\boldsymbol{\varphi 2}()$	**label**
1	$r = \lvert\lvert x_1 - t_1 \rvert\rvert = 2.062$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{2.062^2}{2})}$=0.119	$r = \lvert\lvert x_1 - t_2 \rvert\rvert = 1.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.5^2}{2})}$=0.325	1
2	$r = \lvert\lvert x_2 - t_1 \rvert\rvert = 2.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{2.5^2}{2})}$=0.044	$r = \lvert\lvert x_2 - t_2 \rvert\rvert = 0.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{0.5^2}{2})}$=0.882	1
3	$r = \lvert\lvert x_3 - t_1 \rvert\rvert = 1.803$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.803^2}{2})}$=0.197	$r = \lvert\lvert x_3 - t_2 \rvert\rvert = 1.118$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.118^2}{2})}$=0.535	1
4	$r = \lvert\lvert x_4 - t_1 \rvert\rvert = 1.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.5^2}{2})}$=0.325	$r = \lvert\lvert x_1 - t_1 \rvert\rvert = 2.062$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{2.062^2}{2})}$=0.119	0
5	$r = \lvert\lvert x_5 - t_1 \rvert\rvert = 0.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{0.5^2}{2})}$=0.882	$r = \lvert\lvert x_5 - t_2 \rvert\rvert = 2.5$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{2.5^2}{2})}$=0.044	0
6	$r = \lvert\lvert x_6 - t_1 \rvert\rvert = 1.118$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.118^2}{2})}$=0.535	$r = \lvert\lvert x_6 - t_2 \rvert\rvert = 1.803$ $e^{\left(-\frac{r^2}{2\sigma^2}\right)} = e^{(-\frac{1.803^2}{2})}$=0.197	0

Multi-quadratics RBF kernel c=1

Instance number	$\boldsymbol{\varphi 1}()$	$\boldsymbol{\varphi 2}()$	**label**
1	$r = \lvert\lvert x_1 - t_1 \rvert\rvert$ $= 2.062$ $\sqrt{r^2 + c^2} = \sqrt{2.062^2 + 1}$ $= 2.291$	$r = \lvert\lvert x_1 - t_2 \rvert\rvert$ $= 1.5$ $\sqrt{r^2 + c^2} = \sqrt{1.5^2 + 1}$ $= 1.803$	1
2	$r = \lvert\lvert x_2 - t_1 \rvert\rvert$ $= 2.5$ $\sqrt{r^2 + c^2} = \sqrt{2.5^2 + 1}$	$r = \lvert\lvert x_2 - t_2 \rvert\rvert$ $= 0.5$ $\sqrt{r^2 + c^2} = \sqrt{0.5^2 + 1}$	1

	$= 2.693$	$= 1.118$	
3	$r = \|\|x_3 - t_1\|\|$ $= 1.803$ $\sqrt{r^2 + c^2} = \sqrt{1.803^2 + 1}$ $= 2.061$	$r = \|\|x_3 - t_2\|\|$ $= 1.118$ $\sqrt{r^2 + c^2} = \sqrt{1.118^2 + 1}$ $= 1.5$	0
4	$r = \|\|x_4 - t_1\|\|$ $= 1.5$ $\sqrt{r^2 + c^2} = \sqrt{1.5^2 + 1}$ $= 1.803$	$r = \|\|x_4 - t_2\|\|$ $= 2.062$ $\sqrt{r^2 + c^2} = \sqrt{2.062^2 + 1}$ $= 2.291$	0
5	$r = \|\|x_5 - t_1\|\|$ $= 0.5$ $\sqrt{r^2 + c^2} = \sqrt{0.5^2 + 1}$ $= 1.118$	$r = \|\|x_5 - t_2\|\|$ $= 2.5$ $\sqrt{r^2 + c^2} = \sqrt{2.5^2 + 1}$ $= 2.692$	0
6	$r = \|\|x_6 - t_1\|\|$ $= 1.118$ $\sqrt{r^2 + c^2} = \sqrt{1.118^2 + 1}$ $= 1.5$	$r = \|\|x_6 - t_2\|\|$ $= 1.803$ $\sqrt{r^2 + c^2} = \sqrt{1.803^2 + 1}$ $= 2.062$	0

Inverse multi-quadratic kernel c=1

Instance number	$\boldsymbol{\varphi 1}()$	$\boldsymbol{\varphi 2}()$	**label**
1	$r = \|\|x_1 - t_1\|\|$ $= 2.062$ $\frac{1}{\sqrt{x^2 + c^2}} = 0.436$	$r = \|\|x_1 - t_2\|\|$ $= 1.5$ $\frac{1}{\sqrt{1.5^2 + 1^2}} = 0.555$	1
2	$r = \|\|x_2 - t_1\|\|$ $= 2.5$ $\frac{1}{\sqrt{2.5^2 + 1}} = 0.371$	$r = \|\|x_2 - t_2\|\|$ $= 0.5$ $\frac{1}{\sqrt{0.5^2 + 1^2}} = 0.894$	1
3	$r = \|\|x_3 - t_1\|\|$ $= 1.803$ $\frac{1}{\sqrt{1.803^2 + 1^2}} = 0.485$	$r = \|\|x_3 - t_2\|\|$ $= 1.118$ $\frac{1}{\sqrt{1.118^2 + 1^2}} = 0.667$	1
4	$r = \|\|x_4 - t_1\|\|$ $= 1.5$ $\frac{1}{\sqrt{1.5^2 + 1^2}} = 0.555$	$r = \|\|x_4 - t_2\|\|$ $= 2.062$ $\frac{1}{\sqrt{2.062^2 + 1^2}} = 0.436$	0
5	$r = \|\|x_5 - t_1\|\|$ $= 0.5$ $\frac{1}{\sqrt{0.5^2 + 1^2}} = 0.894$	$r = \|\|x_5 - t_2\|\|$ $= 2.5$ $\frac{1}{\sqrt{2.5^2 + 1^2}} = 0.371$	0
6	$r = \|\|x_6 - t_1\|\|$ $= 1.118$ $\frac{1}{\sqrt{1.118^2 + 1^2}} = 0.667$	$r = \|\|x_6 - t_2\|\|$ $= 1.803$ $\frac{1}{\sqrt{1.803^2 + 1^2}} = 0.485$	0

Hyper spheric RBF kernel c= 1.5

Instance number	$\boldsymbol{\varphi 1}()$	$\boldsymbol{\varphi 2}()$	Label
1	$r = \|\|x_1 - t_1\|\|$ $= 2.062$ $r \leq c? \rightarrow 0$	$r = \|\|x_1 - t_2\|\|$ $= 1.5$ $r \leq c? \rightarrow 0$	1
2	$r = \|\|x_2 - t_1\|\|$ $= 2.5$ $r \leq c? \rightarrow 0$	$r = \|\|x_2 - t_2\|\|$ $= 0.5$ $r \leq c? \rightarrow 0$	1
3	$r = \|\|x_3 - t_1\|\|$ $= 1.803$ $r \leq c? \rightarrow 0$	$r = \|\|x_3 - t_2\|\|$ $= 1.118$ $r \leq c? \rightarrow 0$	1
4	$r = \|\|x_4 - t_1\|\|$ $= 1.5$ $r \leq c? \rightarrow 0$	$r = \|\|x_4 - t_2\|\|$ $= 2.062$ $r \leq c? \rightarrow 0$	0
5	$r = \|\|x_5 - t_1\|\|$ $= 0.5$ $r \leq c? \rightarrow 0$	$r = \|\|x_5 - t_2\|\|$ $= 2.5$ $r \leq c? \rightarrow 0$	0
6	$r = \|\|x_6 - t_1\|\|$ $= 1.118$ $r \leq c? \rightarrow 0$	$r = \|\|x_6 - t_2\|\|$ $= 1.803$ $r \leq c? \rightarrow 0$	0

Question no .7

Given:

σ=1

Two centres

$$t_1\ (0.1, 0.1)$$
$$t_2\ (0.9, 0.9)$$

Gaussian RBF function $\varphi(r) = \exp\left(-\frac{r^2}{2\sigma^2}\right)$

$\boldsymbol{x_1}$	$\boldsymbol{x_2}$	y
0	0	-1
1	0	1
0	1	1
1	1	-1

Solution:

The respective formula will be used:

$$\varphi_1(||x - t_1||) = e^{\frac{-|x-t_1|2}{2\sigma}}$$
$$\varphi_2(||x - t_2||) = e^{\frac{-|x-t_2|2}{2\sigma}}$$

Instances	$\boldsymbol{\varphi_1(\|\|x - t_1\|\|)}$	$\boldsymbol{\varphi_2(\|\|x - t_2\|\|)}$
(0,0)	$e^{\frac{-\|x-t_1\|2}{2\sigma}}$	$e^{\frac{-\|x-t_2\|2}{2\sigma}}$

	$= e^{\frac{-\|0-0.1_1\|2}{2\times 1}}$	$e^{\frac{-\|0-0.9_2\|2}{2}}$
	$= 0.99$	0.66

Similarly for other instances:

(1,0)	**0.66**	**0.66**
(1,0)	0.66	0.66
(1,1)	0.66	0.99

Converted instances

σ			D(desired 0/ps)
1	0.99	0.66	-1
1	0.66	0.66	1
1	0.66	0.66	1
1	0.66	0.99	-1

Pseudo inverse matrix

-6.174	6.674	6.674	6.674	-6.174
5.568	-4.651	-4.651	-4.651	3.734
3.734	-4.651	-4.651	-4.651	5.568

Weight W=σ^fd:

W=[25.696, -18.604, -18.604]^t

Now we will validate using calculated weights

Instances	φ_1	φ_2	Y=$w_0 + w_1\varphi_1 + w_2\varphi_2$	Class
(0,01)	0.99	0.66	-1.001	-1
(1,0)	0.66	0.66	0.9999	1
(0,1)	0.66	0.66	0.9999	1
(1,1)	0.66	0.99	-1.0001	-1

Question 8:

Given:

Multi quadratic RBF function:

$$\varphi(r) = \sqrt{r^2 + c^2}$$

C=0.5

$$t_1 = (0.1,0.1), t_2 = (0.9, 0.9), t_3 = (0.5, 0.5)$$

x_1	x_2	y
0	0	-1
1	0	1
0	1	1
1	1	-1

Solution:

$$\varphi_1(||x-t_1||)=\sqrt{(||x-t_1||)^2+c^2}$$
$$\varphi_2(||x-t_2||)=\sqrt{(||x-t_2||)^2+c^2}$$
$$\varphi_3(||x-t_3||)=\sqrt{(||x-t_3||)^2+c^2}$$

Instances	$\boldsymbol{\varphi_1(\|\|x-t_1\|\|)}$	$\boldsymbol{\varphi_2(\|\|x-t_2\|\|)}$	$\boldsymbol{\varphi_3(\|\|x-t_3\|\|)}$
(0,0)	$\sqrt{(0-0.1)^2+(0.5)^2}$ $=0.519$	$=1.367$	$=0.866$
(1,0)	**1.034**	**1.034**	**0.866**
(0, 1)	1.034	1.034	0.866
(1,1)	1.367	0.519	0.866

Converted instances:

$\boldsymbol{\varphi}$				**Desired 0/p**
1	0.519	1.367	0.866	-1
1	1.034	1.034	0.866	1
1	1.034	1.034	0.866	1
1	1.367	0.519	0.866	-1

Pseudo inverse matrix:

3.253	**-2.967**	**-2.967**	**3.253**
-3.341	2.751	2.751	-2.162
-2.162	2.751	2.751	-3.341
2.817	-2.569	-2.569	2.817

Weight W= $\varphi^t d$:

$$W=[-12.441, 11.007, 11.007, -10.774]^T$$

Now validation using calculated weights

Instances	$\boldsymbol{\varphi_1}$	$\boldsymbol{\varphi_2}$	$\boldsymbol{\varphi_3}$	$\boldsymbol{y = w_0 + w_1\varphi_1 + w_2\varphi_2 + w_3\varphi_3}$	**Class**
(0, 0)				-1.0	-1
(1, 0)				1.0	1
(0, 1)				1.0	1
(1,1)				-1.0	-1