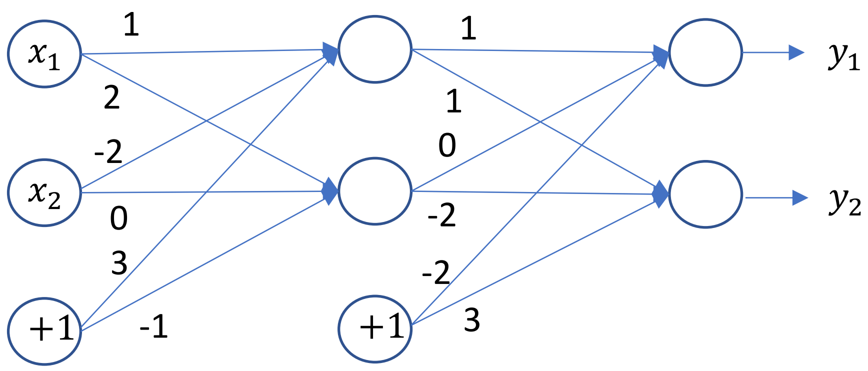

Figure 1

1. The two-layer feedforward perceptron network shown in figure 1 has weights and biases initialized as indicated and receives 2-dimensional inputs (x_1, x_2). The network is to respond with $\boldsymbol{d}_1 = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$ and $\boldsymbol{d}_2 = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ for input patterns $\boldsymbol{x}_1 = \begin{pmatrix} 1.0 \\ 3.0 \end{pmatrix}$ and $\boldsymbol{x}_2 = \begin{pmatrix} -2.0 \\ -2.0 \end{pmatrix}$, respectively.

   Analyse a single feedforward and feedback step for gradient decent learning of the two patterns by doing the following:

   (a) Find the weight matrix $\boldsymbol{W}$ to the hidden-layer and weight matrix $\boldsymbol{V}$ to the output-layer, and the corresponding biases.
   (b) Calculate the synaptic input $\boldsymbol{z}$ and output $\boldsymbol{h}$ of the hidden-layer, and the synaptic input $\boldsymbol{u}$ and output $\boldsymbol{y} = (y_1, y_2)$ of the output layer.
   (c) Find the mean square error cost J between the outputs and targets.
   (d) Calculate the gradients $\nabla_{\boldsymbol{u}} J$ and $\nabla_{\boldsymbol{z}} J$ at the output-layer and the hidden-layer, respectively.
   (e) Compute the new weights and biases.
   (f) Write a program to continue iterations until convergence and find the final weights and biases.

   Assume a learning rate of 0.05.

   Repeat above (a) – (f) for stochastic gradient decent learning.

2. A feedforward neural network with one hidden layer is to perform the following classification:

class	inputs
A	$(1.0, 1.0), (0.0, 1.0)$
B	$(3.0, 4.0), (2.0, 2.0)$
C	$(2.0, -2.0), (-2.0, -3.0)$

The network has a hidden layer consisting of three perceptrons and a softmax output layer.

Initialize the weights $\boldsymbol{W}$ and biases $\boldsymbol{b}$ to the hidden layer, and the weights $\boldsymbol{V}$ and biases $\boldsymbol{c}$ to the output layer as follows:

$$\boldsymbol{W} = \begin{pmatrix} -0.10 & 0.97 & 0.18 \\ -0.70 & 0.38 & 0.93 \end{pmatrix}, \boldsymbol{b} = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}$$

$$\boldsymbol{V} = \begin{pmatrix} 1.01 & 0.09 & -0.39 \\ 0.79 & -0.45 & -0.22 \\ 0.28 & 0.96 & -0.07 \end{pmatrix}, \boldsymbol{c} = \begin{pmatrix} 0.0 \\ 0.0 \\ 0.0 \end{pmatrix}$$

Show one iteration of gradient descent learning and plot the learning curves until convergence at a learning rate $\alpha = 0.1$.

Determine the weights and biases at convergence.

Find the class labels predicted by the trained network for patterns:

$\boldsymbol{x}_1 = \begin{pmatrix} 2.5 \\ 1.5 \end{pmatrix}$ and $\boldsymbol{x}_2 = \begin{pmatrix} -1.5 \\ 0.5 \end{pmatrix}$.

3. Design a deep neural network consisting of two ReLU hidden layers to approximate the following function:

$$\phi(x, y) = 0.8x^2 - y^3 + 2.5xy$$

for $-1.0 \leq x, y \leq 1.0$.

Use 10 neurons and 5 neurons at first and second hidden layers, respectively, and a linear neuron at the output layer.

(a) Divide the input space equally into square regions of size 0.25×0.25 and use grid points as data points to learn the function ϕ.

(b) Train the network using gradient decent learning at learning rate $\alpha = 0.01$ and plot the learning curve (mean square error vs. iterations) and the predicted data points.

(c) Compare the behaviour of learning by plotting learning curves at rates $\alpha = 0.005, 0.001, 0.01,$ and 0.05.