**Assignment 7**

**Title:** a) To implement a simple linear regressor using single perceptron

**Theory:**

* **Linear Regression:**

Linear regression is an approach for modeling relationship between a scalar dependent variable y and one or more independent variables. When there is single independent variable it is called simple linear regression and when there are more than one independent variables it is called multiple linear regression. Linear regression is a statistical procedure for predicting the value of a dependent variable from an independent variable when the relationship between the variables can be described with linear model given by y = m \* x + c.

* **Perceptron as Linear regressor:**

Consider a single perceptron with one input ‘x’ with weight w and bias ‘b’. Then output ‘y’ of the perceptron is given by,

y = w \* x + b

If we compare this equation with equation of line y = m \* x + c it is clear that ‘w’ represents slope of line and bias ‘b’ represents intercept. If we train this perceptron with a training set of samples (x, t) having approximate linear relationship, at the end of the training phase we get ‘w’ and ‘b’ which represents best fit line through the given set of data points.

Thus we can use a perceptron for simple linear regression.

w b

x y = w\*x + b

Figure 1:Perceptron for Linear Regression

**Algorithm:**

**Step 0:** Create data of 100 samples (x, t) which have approximately linear relationship and plot t vs x. (t = mx + c + random number)

**Step 1:** Initialize weight ‘w’ and bias ‘b’ (Set to zero or small random values). Initialize learning rate ‘eta’. Set maximum number of iterations to be used.

**Step 2:** While number of iterations is not zero, do steps 3-9.

**Step 3:** Initialize MSE (Mean Square Error) for current iteration to zero.

**Step 4**: For each training pair, do steps 5-8.

**Step 5**: Compute net output of neuron by,

**y( i ) = w \* x( i ) + b**;

**Step 6**: Compute error signal as difference between target t( i ) and output y( i ).

**e = t( i )** - **y( i )**

**Step 7**: Update weight w and bias b.

**w = w + eta \* e \* x( i )**

**b = b + eta \* e**

**Step 8**: Accumulate MSE for all training pairs.

**MSE( i ) = MSE( i ) + e \* e**

**Step 9**: Decrement number of iterations counter. If number of iterations is not zero, go back to Step 3.

**Step 10**: Plot line with slope w and intercept b.

**FAQ’S:**

1. How a perceptron can be used for linear regression?
2. Implement two input AND gate using perceptron.
3. What are limitations of Perceptron?