Sheet 4

November 15, 2015

Assignment 19

Assignment 20

Given an array N with n elements and a function called *random* that.

Randomize elements

1: **for** $i = 0$ `such that` $i < n$ **do**
2: $i \leftarrow i + 1$
3: $swap \leftarrow random(0, n)$
4: $temp \leftarrow N[swap]$
5: $N[swap] \leftarrow N[i]$
6: $N[i] \leftarrow temp$
7: **end for**

Assignment 21

Backpropagation of errors (BP) is an algorithm used to modify the weights in a Technical Neural Network (TNN) in order to improve the learning of the network.

After the computation of the output values, we calculate the difference between our teacher value ${}^p\hat{y}_m$ and each output unit ${}^p y_m$. We obtain the sum of all and apply to it an error function:

$$ {}^pE = \frac{1}{2} \sum_{m=1}^{M} ({}^p\hat{y}_m - {}^p y_m)^2 $$

Now, is possible to calculate δ values. This calculation is different for the neurons located in the output layer:

$$ \delta_m = (\hat{y}_m - y_m) \cdot f'(net_m) $$

and the neurons located in the hidden layers:

$$\delta_h = (\sum_{k=1}^{k} \delta_k w_{hk}) \cdot f'(net_h)$$

We must be warned that $\delta_k w_{hk}$ are referring to values in the next layer. The way we calculate the weight changes is using this formula:

$$\Delta w_{ij} = \eta \cdot \delta w_{ij} \cdot out_i$$

We iterate in this process until we reach the input layer. Finally, we update the current weights w_{ij} adding to them the changed weights Δw_{ij}:

$$\Delta w_{ij} = w_{ij} + \Delta w_{ij}$$

Assignment 22

Assignment 23