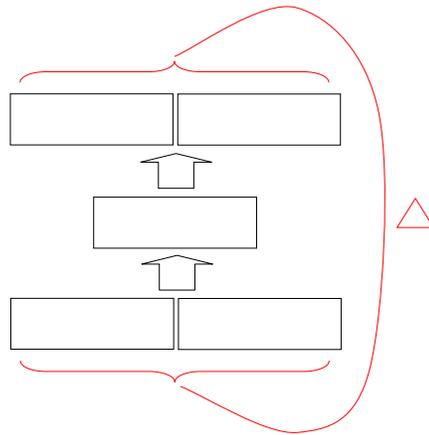


Just a quick summary of the algorithm:

Consider the sequence:  $S = a b c d e f g h i a b c x y z a b c \dots$



Remember  $\Delta$  ONLY determines what will be in the input layer on the next cycle!! It DOES NOT determine whether or not to backprop.

- $i$  and  $a$  are on input
- $i$  and  $a$  are fed forward to the output layer to produce  $\hat{i}$  and  $\hat{a}$  on the output layer and  $H$  on the hidden layer
- The output  $\hat{i}$  and  $\hat{a}$  is compared to  $i$  and  $a$ . An error value  $\Delta$  is calculated between  $[\hat{i}, \hat{a}]$  (the output of the network) and  $[i, a]$  (the desired output).
- **Since  $i$  and  $a$  are both items from  $S$ , a backprop pass is done.** Whenever the two items on input are items from  $S$ , a backprop pass is done.
- Depending on the size of  $\Delta$  one of two things may happen:
  - If  $\Delta$  is large (above CRITERION), then  $a$  is moved into the left-hand side of the input and the next item in the sequence,  $b$ , is put into the right-hand side.
  - If  $\Delta$  is small (at or below CRITERION), then  $H$  is put into the left-hand side of the input and the next item in the sequence,  $b$ , is put into the right-hand side.
- Whatever is on input is fed through to the output:
  - An error value  $\Delta$  is calculated between what is on the output of the network and what is on input.
  - If the both of the inputs were from  $S$ , a backprop pass is done.
  - If the left-hand input is  $H$ , a backprop pass is done with  $p = 0.25$ .