On the Problem of Local Minima in Back-propagation

M. Gori and A. Tesi

Presented by Jonathon Smereka

Motivation

- "Back-Propagation fails to Separate Where Perceptrons Succeed" (Brady et al, 1989)
  - Rosenblatt's perceptrons will converge given linearly separable data
  - Give examples where given linearly separable data back-propagation fails
- This paper proves that this is not the case for all Multilayered Models

Back-propagation Overview

- Forward pass will give Neural Network output

  $l =$ layer index, $t =$ specific pattern

  $n(l) =$ # of neurons at layer l $(l = 1,2, \ldots L)$

  $i(l) =$ neuron index for layer $l = 1 \ldots n(l)$

  $x_{i(l)}(t) =$ neuron $i(l)$'s output when pattern t is presented

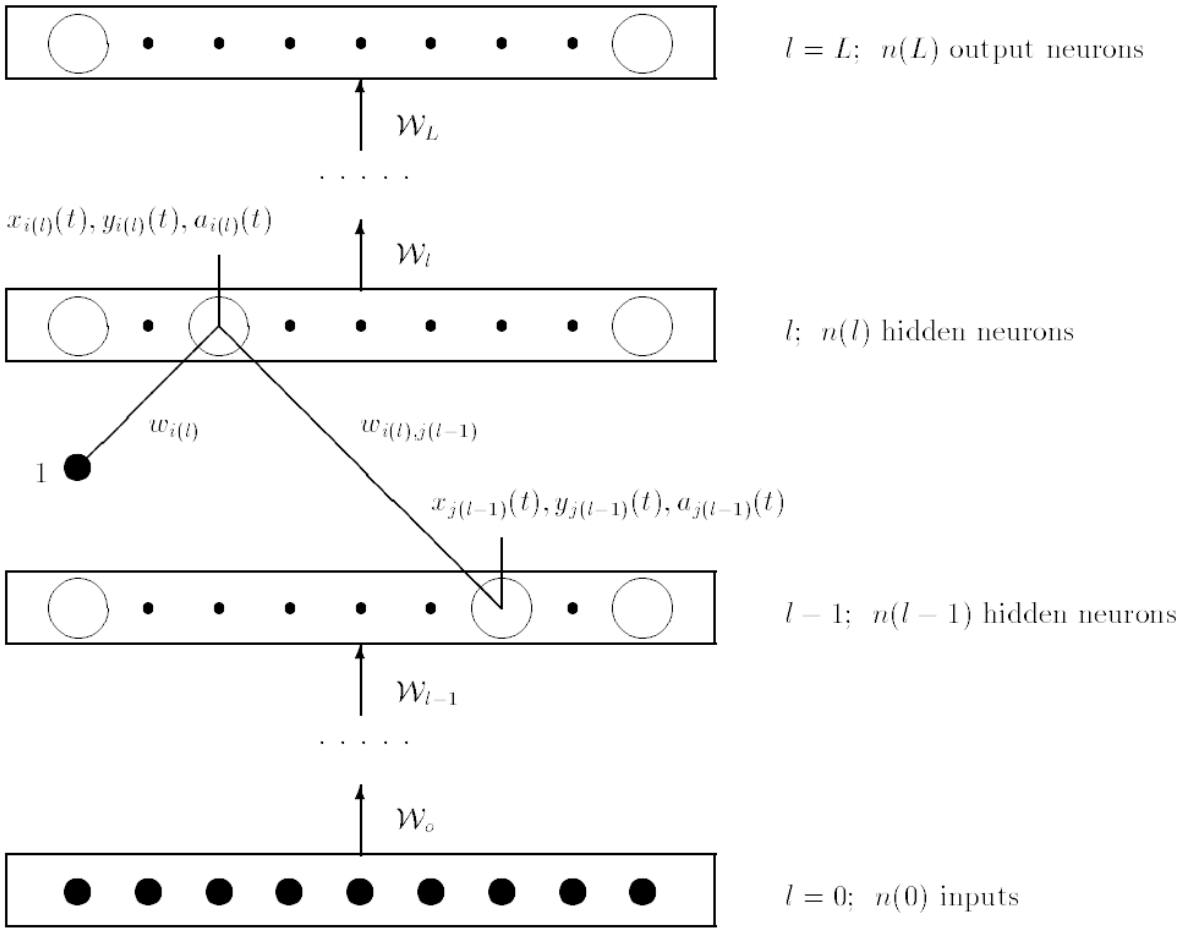
  $a_{i(l)}(t) =$ neuron $i(l)$'s activation when pattern t is presented

- Compute to update weights: $g(j(l-1), i(l)) = \frac{\partial E_T}{\partial w_{i(l),j(l-1)}}$

  $w_{i(l),j(l-1)} =$ weight of link between neurons $i(l)$ and $j(l-1)$

  $E_T = \frac{1}{2}\sum_{t=1}^{T} E_t = \frac{1}{2}\sum_{t=1}^{T}\sum_{j=1}^{L}\left(d_j(t) - x_{j(L)}(t)\right)^2 =$ cost index

Multilayered Network



Note that: $y_{i(l)}(t) = \frac{\partial E_T}{\partial a_{i(l)}(t)}$ = neuron $i(l)$'s delta error from pattern t

Theorem #1

$E_T \to 0$ if the MLN and it's associated learning environments satisfy the following:

1. Pyramid structure: $n(l+1) \leq n(l)$ for $l = 1, \dots L-1$
2. Weight matrix is full row rank (W_l for $l = 1, \dots L-1$)
3. Input patterns are linearly separable

Theorem 1.3

Input patterns are linearly separable

- Ensures that the gradient for the input layer is solvable, i.e. $G_0 = X_0^T Y_1$, $Y_1 \to 0$ if $G_0 = 0$
- The input data matrix X_0^T is required to be full rank so that the null space is only the empty set
- Limits the number of input neurons ($n(0)$) to the number of available patterns (T)

Theorem 1.1 & 1.2

Pyramid structure & Weight matrix is full row rank

- Assuming that $G_0 = 0$ implies $Y_1 \to 0$
- If W_l is full row rank then the pseudo-inverse can be calculated as $W_l^{\dagger} = W_l^T \left[W_l W_l^T \right]^{-1}$

  $\tilde{Y}_l = Y_{l+1} W_l \Rightarrow Y_{l+1} = \tilde{Y}_l W_l^{\dagger}$

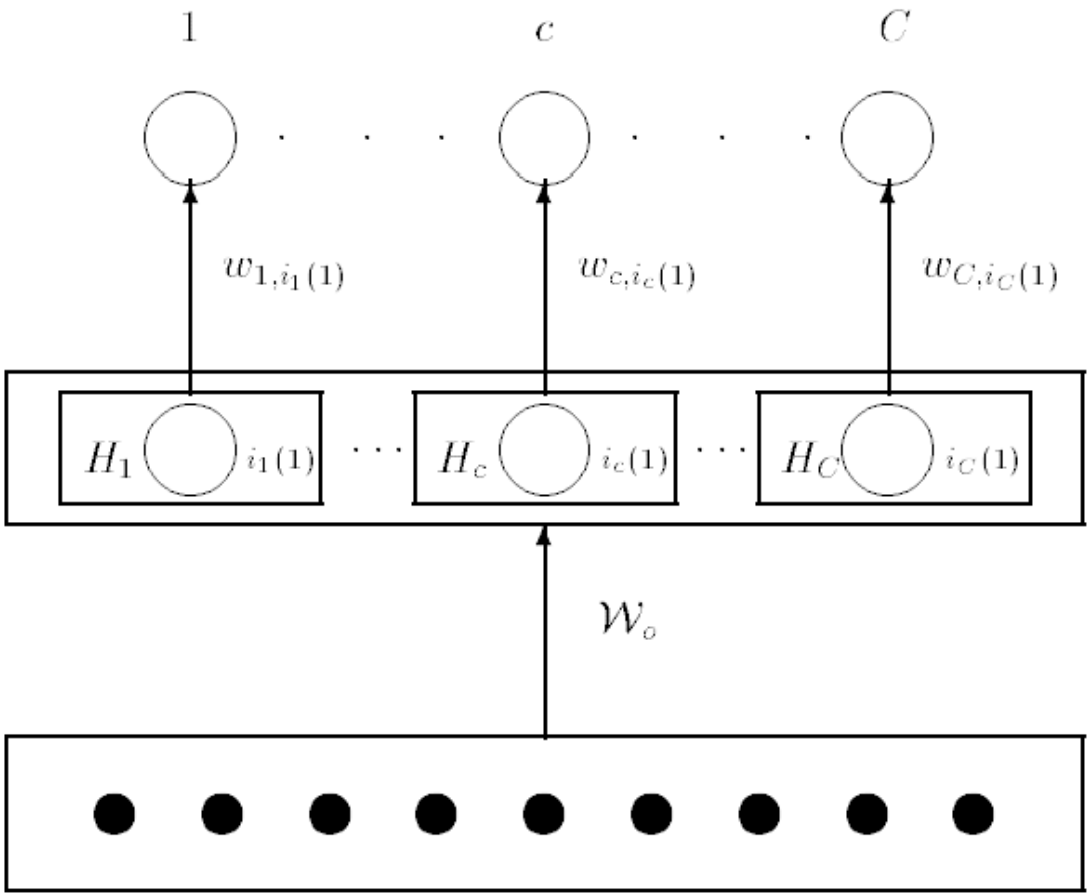
  $\therefore Y_L \to 0$ implies the $G_l = 0$ for each layer (using asymptotic target values)

Theorem #2

Gradient descent leads to absolute minimum ($E_T \rightarrow 0$), given the following conditions:

- Network has one hidden layer with C outputs (# of classes) with full connections from inputs to the subdivided hidden layers
- Exclusive coding is used for the output (for each class)
- All patterns are linearly-separable

Theorem #2 Network



$l = 2$; C output neurons

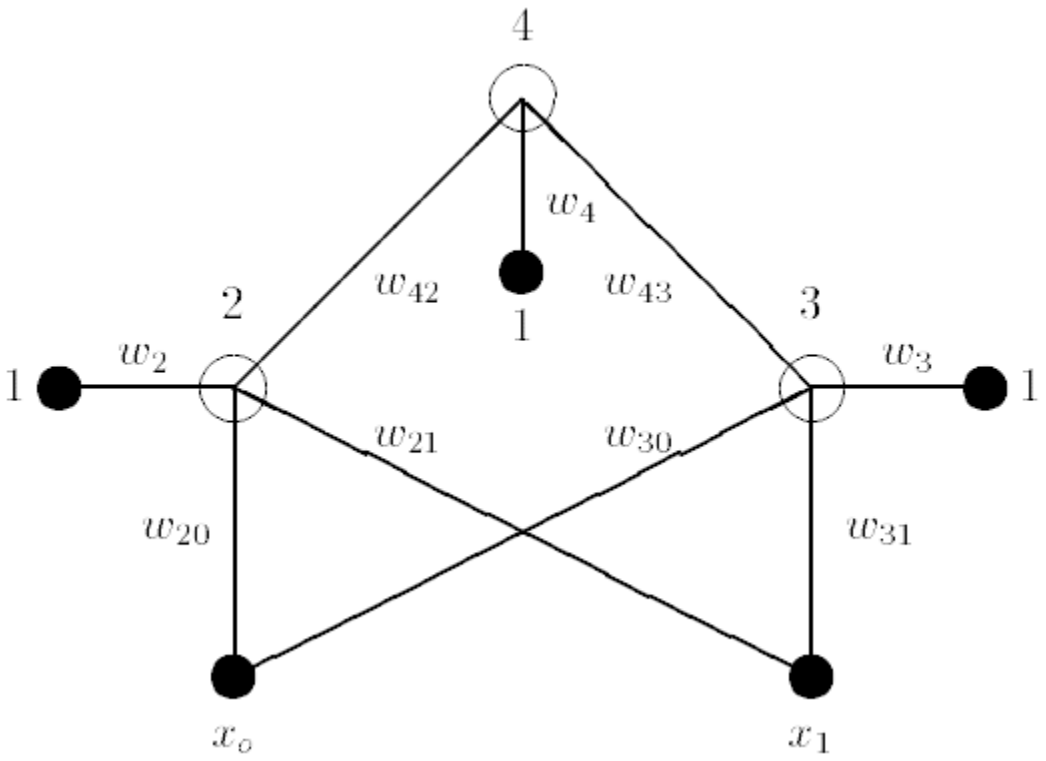
$l = 1$; C sub-layers H_c with $n_c(1)$ neurons

$l = 0$; $n(0)$ inputs

Theorem #2

- Simple multilayered net that is guaranteed to have exclusive coding for the C classes
  - Thus the delta error will have the same sign for all patterns of the same class
- Linearly separable patterns (C hyperplanes exist that return the same sign for all patterns of the same class)
  - Thus $Y_1 = 0$ if $G_0 = 0$

XOR Example

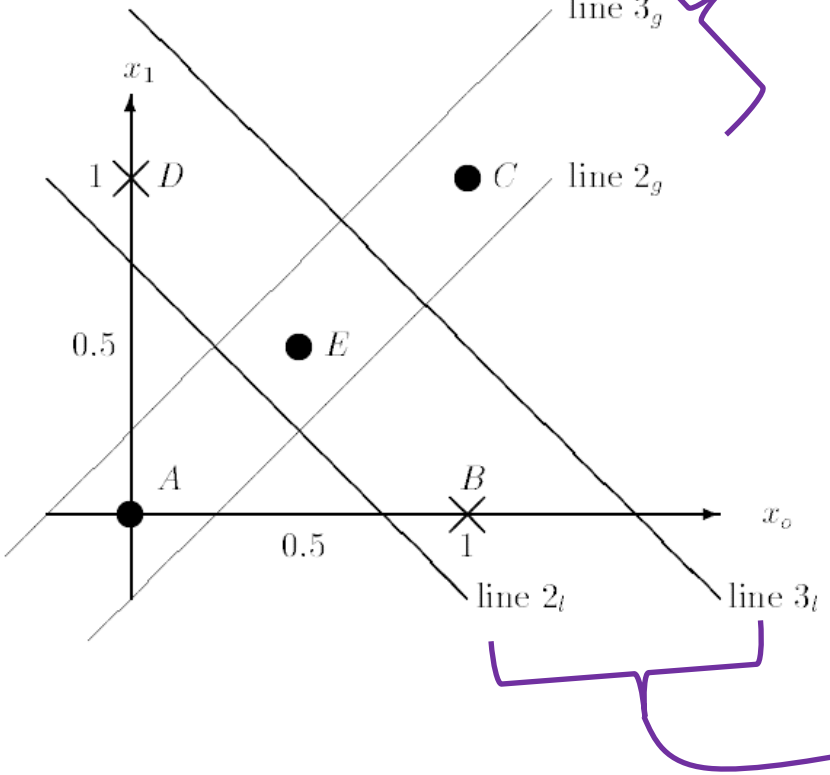


Pattern	x_o	x_1	Target
A	0	0	0
B	1	0	1
C	1	1	0
D	0	1	1
E	0.5	0.5	0

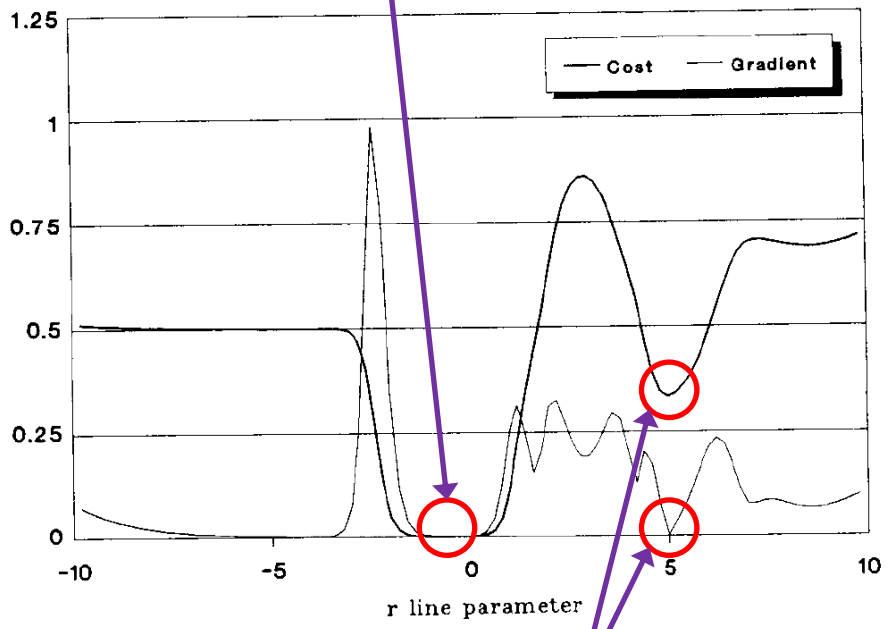
Network meets the requirements of Theorem #2, but the data is not linearly separable

Pattern	x_0	x_1	Target
A	0	0	0
B	0	1	1
C	1	1	0
D	0	1	1
E	0.5	0.5	0

XOR Example



Global minima ($G = 0, E_T = 0$)



Local minima ($G = 0, E_T \neq 0$)

Even though it is possible to perfectly learn the input patterns, BP can get stuck in local minima

Conclusions

- For linearly separable patterns, BP will converge to the global optimum
- When not linearly separable, BP can get stuck in local optima
  - Initializing the weights is particularly important in this case
  - May be useful to adjust when weights are updated (after a subset of the whole learning environment has been presented)