

# Neural network notes

Marco Marini

November 16, 2015

## Contents

<b>1</b>	<b>Generale</b>	<b>1</b>
<b>2</b>	<b>Funzione di costo</b>	<b>2</b>
<b>3</b>	<b>Back-propagation</b>	<b>3</b>

### Abstract

This document contains notes about neural network regarding the Temporal Differential reinforcement algorithms.

## 1 Generale

Siano data una rete neurale formata da  $N$  strati di neuroni. Ogni strato  $i = 1 \dots N$  è composto da  $s_i$  neuroni.

Il primo strato

$$H_{1i} = x_i$$

rappresenta i valori d'ingresso della rete. Per utilità indicheremo i segnali fissi di polarizzazione con

$$H_{i0} = 1$$

Ogni neurone elabora segnali afferenti dal proprio livello e produrrà un segnale uscita verso i livelli successivi. Prendiamo in esame una rete NLR (Non Linear regression)

La funzione di trasferimento di ogni neurone nascosto  $i < N$  è

$$\begin{aligned} H_{(i+1)j} &= \frac{1}{1+e^{-Z_{ij}}} \\ Z_{ij} &= \sum_{k=0}^{s_i} w_{ijk} H_{ik} \end{aligned} \tag{1}$$

la funzione di trasferimento dello strato di uscita  $N$  è

$$H_{Nj} = \sum_{k=0}^{s_{N-1}} w_{(N-1)jk} H_{(N-1)k} \tag{2}$$

## 2 Funzione di costo

Sia  $y_j$  l'uscita attesa per il neurone  $j$ , definiamo l'errore del neurone con

$$\delta_{(N-1)j} = y_j - H_{Nj} \quad (3)$$

La funzione costo della rete è definita come

$$J = \begin{cases} \frac{1-\alpha}{2} \sum_{i=1}^{s_N} \delta_{(N-1)i}^2 + \frac{\alpha}{2} \sum_{i,j,k} w_{ijk}^2, & k \geq 1 \\ \frac{1-\alpha}{2} \sum_{i=1}^{s_N} \delta_{(N-1)i}^2, & k = 0 \end{cases}$$

dove  $\alpha$  è il fattore di regolarizzazione della rete.

Calcoliamo  $\nabla J$

$$\frac{\partial}{\partial w_{ijk}} J = \begin{cases} (1-\alpha) \sum_{r=1}^{s_N} \delta_{(N-1)r} \frac{\partial}{\partial w_{ijk}} \delta_{(N-1)r} + \alpha w_{ijk}, & k \geq 1 \\ (1-\alpha) \sum_{r=1}^{s_N} \delta_{(N-1)r} \frac{\partial}{\partial w_{ijk}} \delta_{(N-1)r}, & k = 0 \end{cases}$$

Per la (3) abbiamo

$$\frac{\partial}{\partial w_{ijk}} \delta_{(N-1)r} = -\frac{\partial}{\partial w_{ijk}} H_{Nr}$$

quindi

$$\frac{\partial}{\partial w_{ijk}} J = \begin{cases} -(1-\alpha) \sum_{r=1}^{s_N} \delta_{(N-1)r} \frac{\partial}{\partial w_{ijk}} H_{Nr} + \alpha w_{ijk}, & k \geq 1 \\ -(1-\alpha) \sum_{r=1}^{s_N} \delta_{(N-1)r} \frac{\partial}{\partial w_{ijk}} H_{Nr}, & k = 0 \end{cases}$$

Per ora trascuriamo gli effetti di regolarizzazione della rete

Per la (2) abbiamo

$$\frac{\partial}{\partial w_{(N-1)jk}} H_{Nj} = H_{(N-1)k}$$

$$\frac{\partial}{\partial w_{(N-1)jk}} H_{Nr} = 0, \quad r \neq j$$

quindi

$$\frac{\partial}{\partial w_{(N-1)jk}} J = -\delta_{(N-1)j} H_{(N-1)k}$$

Per la (2) e  $i \leq N-2$  abbiamo

$$\begin{aligned} \frac{\partial J}{\partial w_{ijk}} &= -\sum_r \delta_{(N-1)r} \frac{\partial}{\partial w_{ijk}} \left[ \sum_s w_{(N-1)rs} H_{(N-1)s} \right] = \\ &= -\sum_{r,s} \delta_{(N-1)r} w_{(N-1)rs} \frac{\partial}{\partial w_{ijk}} H_{(N-1)s} \end{aligned}$$

Dalla (1) abbiamo

$$\frac{\partial}{\partial w_{ijk}} H_{(i+1)j} = \frac{\partial}{\partial Z_{ij}} H_{(i+1)j} \frac{\partial}{\partial w_{ijk}} Z_{ij} = H_{(i+1)j} (1 - H_{(i+1)j}) H_{ik}$$

Quindi

$$\frac{\partial}{\partial w_{(N-2)jk}} J = - \sum_r \delta_{(N-1)r} w_{(N-1)rj} H_{(N-1)j} (1 - H_{(N-1)j}) H_{(N-2)k}$$

### 3 Back-propagation

Definiamo ora

$$\delta_{(N-2)j} = \sum_r \delta_{(N-1)r} w_{(N-1)rj} H_{(N-1)j} (1 - H_{(N-1)j})$$

Avremo allora

$$\frac{\partial}{\partial w_{(N-2)jk}} J = -\delta_{(N-2)j} H_{(N-2)k}$$

Induttivamente ricaviamo per  $i \leq N - 2$

$$\delta_{ij} = \sum_k \delta_{(i+1)k} w_{(i+1)kj} H_{(i+1)j} (1 - H_{(i+1)j})$$

mentre in generale

$$\frac{\partial}{\partial w_{ijk}} J = -\delta_{ij} H_{ik}$$

Definiamo ora il tensore di back-propagation come

$$B_{ijk} = w_{(i+1)kj} H_{(i+1)j} (1 - H_{(i+1)j}), \quad i \leq N - 2$$

L'errore sui livelli intermedi è dato da

$$\delta_{ij} = \sum_{k=1}^{s_i} \delta_{(i+1)k} B_{ijk}, \quad i = 1 \dots N - 2$$

Reintroducendo la regolarizzazione otteniamo

$$\frac{\partial}{\partial w_{ijk}} J = \begin{cases} -(1 - \alpha) \delta_{ij} H_{ik} + \alpha w_{ijk}, & k \geq 1 \\ -(1 - \alpha) \delta_{ij} H_{ik}, & k = 0 \end{cases}$$

La correzione dei pesi della rete per ridurre l'errore (gradient descent) sarà quindi

$$\Delta w_{ijk} = -\eta \frac{\partial}{\partial w_{ijk}} J = \begin{cases} \eta [(1 - \alpha) \delta_{ij} H_{ik} - \alpha w_{ijk}], & k \geq 1 \\ \eta (1 - \alpha) \delta_{ij} H_{ik}, & k = 0 \end{cases}$$

con  $\eta$  fattore di apprendimento.