# APR Ejercicios Tema 5

Vladyslav Mazurkevych

---

**1) En el formato de las transparencias 5.27 y 5.28 (El Algoritmo BackProp):**

El BackProp al ser un algoritmo que depende de la técnica del descenso por gradiente, tiene el problema de ser muy propenso a terminar en un mínimo local con mucha facilidad, con lo que para minimizar estos hechos, se usan varias variantes del algoritmo que le permiten no quedarse atrapado en un mínimo local sino seguir intentando caer en el global o, en su defecto, a otro mínimo local mejor.

**→ Escribir el algoritmo BackProp batch con momentum.**

El BackProp padece de convergencia lenta, así como para evitar que se quede en un mínimo local fácilmente pudiendo haber otras opciones mejores, se añade una 'inercia' o 'momentum' con un peso $0 \leq v < 1$ que actuará sobre el descenso del gradiente con una idea similar a la que tenemos en la física clásica, lo que le permitirá al algoritmo seguir un poco más del punto mínimo donde quedaría estancado en un BackProp por defecto.

***Entrada:*** Topología, pesos iniciales $\theta_{ij}^{l}$ $1 \leq l \leq L$, $1 \leq i \leq M_l$ , $0 \leq j \leq M_{l-1}$, factor de aprendizaje $\rho$, condiciones de convergencia, N datos de entrenamiento S, además ahora tenemos el momentum, con lo que momentum $0 \leq v < 1$ .

***Salida:*** Pesos de las conexiones que minimizan el error cuadrático medio de S.

*Mientras no se cumplan las condiciones de convergencia:*

*Para* $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$, *inicializar* $\Delta\theta_{ij}^{l} = 0$

*Para cada muestra de entrenamiento (x,t)* $\in S$

*Desde la capa de entrada a la de salida (l = 0, ..., L):*

*Para* $1 \leq i \leq M_l$, *si l = 0 entonces* $s_i^0 = x_i$ *sino calcular* $\phi_i^l$ *y* $s_i^l = g(\phi_i^l)$

*Desde la capa de salida a la de entrada (l = L, ..., 1):*

*Para cada nodo* $(1 \leq i \leq M_l)$

*Calcular si l==L :* $\delta_i^l = g'(\phi_i^l)(t_{ni} - s_i^L)$

*Ó en otro caso:* $\delta_i^l = g'(\phi_i^l)\left(\sum_r \delta_r^{l+1}\theta_{ri}^{l+1}\right)$

*Para cada peso* $\theta_{ij}^{l}$ $(0 \leq j \leq M_{l-1})$ *calcular*

$$\Delta\theta_{ij}^{l} = v\Delta\theta_{ij}^{l} + \rho\,\delta_i^l\, s_j^{l-1}$$

*Para* $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$, *actualizar pesos* $\theta_{ij}^{l} = \theta_{ij}^{l} + \frac{1}{N}\Delta\theta_{ij}^{l}$

**→ Escribir el algoritmo BackProp incremental con momentum.**

***Entrada:*** Topología, pesos iniciales $\theta_{ij}^{l}$ $1 \leq l \leq L$, $1 \leq i \leq M_l$ , $0 \leq j \leq M_{l-1}$, factor de aprendizaje $\rho$, condiciones de convergencia, N datos de entrenamiento S, además ahora tenemos el momentum, con lo que momentum $0 \leq v < 1$ .

# APR Ejercicios Tema 5

Vladyslav Mazurkevych

---

***Salida:*** Pesos de las conexiones que minimizan el error cuadrático medio de S.

Mientras no se cumpla nlas condiciones de convergencia:

*Para cada muestra de entrenamiento (x,t)* $\in S$

*Desde la capa de entrada a la de salida (l = 0, ..., L):*

*Para* $1 \leq i \leq M_l$*, si l = 0 entonces* $s_i^0 = x_i$ *sino calcular* $\phi_i^l$ *y* $s_i^l = g(\phi_i^l)$

*Desde la capa de salida a la de entrada (l = L, ..., 1):*

*Para cada nodo* $(1 \leq i \leq M_l)$

*Calcular si l==L :* $\delta_i^l = g'(\phi_i^l)(t_{ni} - s_i^L)$

*Ó en otro caso:* $\delta_i^l = g'(\phi_i^l)(\sum_r \delta_r^{l+1} \theta_{ri}^{l+1})$

*Para cada peso* $\theta_{ij}^l$ $(0 \leq j \leq M_{l-1})$ *calcular*

$$\Delta\theta_{ij}^l = \rho\, \delta_i^l\, s_j^{l-1}$$

*Para* $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$, *actualizar pesos* $\theta_{ij}^l = v\theta_{ij}^l + \frac{1}{N} \Delta\theta_{ij}^l$

**⟶ Escribir el algoritmo BackProp batch con amortiguamiento.**

Esta variante del algoritmo pretende evitar que los pesos sean muy grandes y provoquen una parálisis de la red mediante la regularización del error cuadrático medio. Por lo tanto vamos añadir un nuevo parámetro a la ecuación para conseguir tal fin.

***Entrada:*** Topología, pesos iniciales $\theta_{ij}^l$ $1 \leq l \leq L$, $1 \leq i \leq M_l$ , $0 \leq j \leq M_{l-1}$, factor de aprendizaje $\rho$, condiciones de convergencia, N datos de entrenamiento S, además ahora tenemos añadido un factor de regularización $\lambda$.

***Salida:*** Pesos de las conexiones que minimizan el error cuadrático medio de S.

*Mientras no se cumplan las condiciones de convergencia:*

*Para* $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$, *inicializar* $\Delta\theta_{ij}^l = 0$

*Para cada muestra de entrenamiento (x,t)* $\in S$

*Desde la capa de entrada a la de salida (l = 0, ..., L):*

*Para* $1 \leq i \leq M_l$*, si l = 0 entonces* $s_i^0 = x_i$ *sino calcular* $\phi_i^l$ *y* $s_i^l = g(\phi_i^l)$

*Desde la capa de salida a la de entrada (l = L, ..., 1):*

*Para cada nodo* $(1 \leq i \leq M_l)$

*Calcular si l==L :* $\delta_i^l = g'(\phi_i^l)(t_{ni} - s_i^L)$

*Ó en otro caso:* $\delta_i^l = g'(\phi_i^l)(\sum_r \delta_r^{l+1} \theta_{ri}^{l+1})$

*Para cada peso* $\theta_{ij}^l$ $(0 \leq j \leq M_{l-1})$ *calcular*

$$\Delta\theta_{ij}^l = \rho\, \delta_i^l\, s_j^{l-1} - 2\rho\lambda\, \theta_{ij}^l$$

*Para* $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$, *actualizar pesos* $\theta_{ij}^l = \theta_{ij}^l + \frac{1}{N} \Delta\theta_{ij}^l$

# APR Ejercicios Tema 5

Vladyslav Mazurkevych

---

**→ Escribir el algoritmo BackProp incremental con amortiguamiento.**

***Entrada:*** Topología, pesos iniciales $\theta_{ij}^{l}$ $1 \leq l \leq L$, $1 \leq i \leq M_l$ , $0 \leq j \leq M_{l-1}$, factor de aprendizaje $\rho$, condiciones de convergencia, N datos de entrenamiento S, además ahora tenemos añadido un factor de regularización $\lambda$.

***Salida:*** Pesos de las conexiones que minimizan el error cuadrático medio de S.

Mientras no se cumpla nlas condiciones de convergencia:

*Para cada muestra de entrenamiento (x,t)* $\in S$

*Desde la capa de entrada a la de salida (l = 0, ..., L):*

*Para* $1 \leq i \leq M_l$*, si l = 0 entonces* $s_i^0 = x_i$ *sino calcular* $\phi_i^l$ *y* $s_i^l = g(\phi_i^l)$

*Desde la capa de salida a la de entrada (l = L, ..., 1):*

*Para cada nodo* $(1 \leq i \leq M_l)$

*Calcular si l==L :* $\delta_i^l = g'(\phi_i^l)(t_{ni} - s_i^L)$

*Ó en otro caso:* $\delta_i^l = g'(\phi_i^l)(\sum_r \delta_r^{l+1} \theta_{ri}^{l+1})$

*Para cada peso* $\theta_{ij}^l$ $(0 \leq j \leq M_{l-1})$ *actualizar:*

$$\theta_{ij}^l = -2\rho\lambda\theta_{ij}^l + \frac{\rho}{N}\delta_i^l s_j^{l-1}$$

**2) Desarrollar formalmente las ecuaciones de actualización de los pesos en el algoritmo BackProp para clasificación (transparencia 6.39)**

Antes de nada, recuérdese que la actualización de los pesos en el algoritmo BackProp se realiza durante la segunda fase, después de hacer el *´forward pass´* obtenemos la nueva aproximación del algoritmo, con lo que ahora haremos descenso por gradiente para ver hacia qué sentido descenderá nuestra ecuación para intentar conseguir el mínimo error.

Para nuestro algoritmo BackProp utilizaremos como criterio de optimización la entropía cruzada, que viene dada por la siguiente fórmula:

$$q_s(\Theta) = -\frac{1}{N}\sum_{n=1}^{N}\sum_{i=1}^{M_2} t_{ni} \log(s_i^2(x_n;\Theta)) \quad (1)$$

Donde el $t_{ni}$ representa la solución deseada por el modelo (labels) y el $s_i^2(x_n;\Theta)$ la predicción del modelo. Suponiendo que despejamos la ecuación 1, lo haríamos por capas, empezando en escribir la segunda capa que dependería de la primera, de esta forma:

$$s_i^2 = g(\phi_i^2) \qquad (2)$$

$$\phi_i^2 = \sum_{j=0}^{M_1} \theta_{ij}^2 \, s_j^1 \qquad (3)$$

Por otra parte tenemos la otra capa de la que depende

$$s_i^1 = g(\phi_j^1) \qquad (4)$$

$$\phi_j^1 = \sum_{k=0}^{Mo} \theta_{jk}^1 \, x_k^1 \qquad (5)$$

Pero si nos fijamos en el descenso por gradiente que nos piden despejar, va en función *l* que representa las capas, así que en función de *l* despejamos la ecuación 1 como sigue:

$$q_s(\Theta) = -\frac{1}{N} \sum_{i=1}^{N} \sum_{r=1}^{M_{l+1}} t_{ni} \log(g(\sum_{j=0}^{M_l} \theta_{ij}^{l+1} \, s_j^l)) \qquad (6)$$

Por lo tanto el descenso por gradiente nos quedaría como:

$$\frac{\partial q_s(\theta)}{\partial \theta_{ij}^l} = \sum_{i=1}^{N} \sum_{r=1}^{M_{l+1}} \frac{\partial q}{\partial s_r^{l+1}} \frac{\partial s_r^{l+1}}{\partial \theta_{ij}^1} = \sum_{i=1}^{N} \sum_{r=1}^{M_{l+1}} \frac{\partial q}{\partial s_r^{l+1}} \frac{\partial s_r^{l+1}}{\partial \phi_r^{1+1}} \frac{\partial \phi_r^{1+1}}{\partial s_i^l} \frac{\partial s_i^l}{\partial \phi_i^1} \frac{\partial \phi_i^1}{\partial \theta_{ij}^1} \qquad (7)$$

Vamos a suponer las siguientes sustituciones:

$$\begin{aligned} \frac{\partial q}{\partial s_r^{l+1}} \frac{\partial s_r^{l+1}}{\partial \phi_r^{1+1}} &\longrightarrow -\delta_r^{l+1} \\ \frac{\partial \phi_r^{1+1}}{\partial s_i^l} &\longrightarrow \theta_{ri}^{l+1} \\ \frac{\partial s_i^l}{\partial \phi_i^1} &\longrightarrow g'(\phi_i^l) \\ \frac{\partial \phi_i^1}{\partial \theta_{ij}^1} &\longrightarrow x_j \end{aligned} \qquad (8)$$

Simplificando nos quedaría:

$$\begin{aligned} \frac{\partial q}{\partial s_r^{l+1}} \frac{\partial s_r^{l+1}}{\partial \phi_r^{1+1}} &\longrightarrow -\delta_r^{l+1} \\ \frac{\partial \phi_r^{1+1}}{\partial s_i^l} \frac{\partial s_i^l}{\partial \phi_i^1} &\longrightarrow -\delta_r^l \\ \frac{\partial \phi_i^1}{\partial \theta_{ij}^1} &\longrightarrow x_j \end{aligned} \qquad (9)$$

Ahora veremos con qué corresponde cada parte, así pues la $\delta_r^{l+1}$ será definida después de aplicar las derivaciones como:

$$-\delta_r^{l+1} = \left(\frac{-t_{pi}}{s^{l+1}}\, g'(\phi_i^{l+1})\right) \qquad (10)$$

Una vez obtenida la definición de la $-\delta_r^{l+1}$ vamos a proseguir definiendo la derivación de la siguiente capa, con lo cual después de aplicar las derivaciones definidas en la ecuaciones 7,8 y 9 nos quedaría la $\delta_r^{l}$ definida con respecto a la ecuación 10 como:

$$-\delta_r^{l} = -(g'(\phi_i^{l}) \sum_{r=1}^{M_{l+1}} \delta_r^{l+1} \theta_{ri}^{l+1}) \qquad (11)$$

Con lo cual, sustituyendo la ecuación 7 con las correspondientes derivaciones nos quedaría:

$$\frac{\partial q_s(\theta)}{\partial \theta_{ij}^{l}} = -\sum_{i=1}^{N} \sum_{r=1}^{M_{l+1}} \left(\frac{-t_{pi}}{s^{l+1}}\, g'(\phi_i^{l+1})\right) \theta_{ri}^{l+1}\, g'(\phi_i^{l})\, x_j \qquad (12)$$

Simplificando lo anterior con las sustituciones previamente calculadas, obtenemos:

$$\frac{\partial q_s(\theta)}{\partial \theta_{ij}^{l}} = -\sum_{i=1}^{N} \delta_i^{l} x_j \qquad (13)$$

Una vez despejadas las derivadas, podemos completar nuestra nueva ecuación de descenso por gradiente, la cual tendría la fórmula como la que sigue suponiendo que está compuesta por dos capas:

$$\Delta\theta_{ij}^{l} = -\rho \frac{\partial q_s(\Theta)}{\partial \theta_{ij}^{l}} = -\rho \sum_{i=1}^{N} \delta_i^{l} x_j \qquad 1 \le l \le 2,\ 1 \le i \le M_l,\ 0 \le j \le M_{l-1} \qquad (14)$$