

Algoritmo BACKPROP

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l
 $1 \leq l \leq L, 1 \leq i \leq M_l, 1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Mientras no converja

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

Para toda muestra de aprendizaje $(\mathbf{x}, \mathbf{t}) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$: Calcular $\phi_i^l(\mathbf{x})$ y $s_i^l(\mathbf{x}) = g(\phi_i^l(\mathbf{x}))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(\mathbf{x}) = \begin{cases} g'(\phi_i^l(\mathbf{x})) (t_i - s_i^L(\mathbf{x})) & \text{si } l == L \\ g'(\phi_i^l(\mathbf{x})) (\sum_r \delta_r^{l+1}(\mathbf{x}) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$: Calcular: $\Delta\theta_{ij}^l += \delta_i^l(\mathbf{x}) s_j^{l-1}(\mathbf{x})$ Fin-para

Fin-para cada nodo

Fin-para cada muestra

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$, Actualizar pesos: $\theta_{ij}^l += \rho \Delta\theta_{ij}^l$ Fin-para

Fin-mientras

Algoritmo BACKPROP incremental

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l $1 \leq l \leq L$, $1 \leq i \leq M_l$, $1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Mientras no converja

Para toda muestra de aprendizaje $(x, t) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$, calcular $\phi_i^l(x)$ y $s_i^l(x) = g(\phi_i^l(x))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(x) = \begin{cases} g'(\phi_i^l(x)) (t_i - s_i^L(x)) & \text{si } l == L \\ g'(\phi_i^l(x)) (\sum_r \delta_r^{l+1}(x) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$, calcular: $\Delta\theta_{ij}^l = \delta_i^l(x) s_j^{l-1}(x)$ Fin-para

Fin-para

Para $1 \leq l \leq L$, $1 \leq i \leq M_l$, $0 \leq j \leq M_{l-1}$: actualizar $\theta_{ij}^l += \rho \Delta\theta_{ij}^l$ Fin-para

Fin-para

Fin-mientras

Algoritmo BACKPROP (batch) con momentum

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l

$1 \leq l \leq L, 1 \leq i \leq M_l, 1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$, inicializar $\Delta \bar{\theta}_{ij}^l = 0$ Fin-para

/* $\Delta \bar{\theta}_{ij}^l$: almacenamiento de los incrementos para la iteración siguiente */

Mientras no converja

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

Para toda muestra de aprendizaje $(\mathbf{x}, \mathbf{t}) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$: Calcular $\phi_i^l(\mathbf{x})$ y $s_i^l(\mathbf{x}) = g(\phi_i^l(\mathbf{x}))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(\mathbf{x}) = \begin{cases} g'(\phi_i^l(\mathbf{x})) (t_i - s_i^L(\mathbf{x})) & \text{si } l == L \\ g'(\phi_i^l(\mathbf{x})) (\sum_r \delta_r^{l+1}(\mathbf{x}) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$ calcular: $\Delta \theta_{ij}^l += \delta_i^l(\mathbf{x}) s_j^{l-1}(\mathbf{x})$ Fin-para

Fin-para cada nodo

Fin-para toda muestra

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$:

Calcular $\Delta \bar{\theta}_{ij}^l = \rho \Delta \theta_{ij}^l + \nu \Delta \bar{\theta}_{ij}^l$ /* Incremento para ser usado en la iteración siguiente */

Actualizar pesos: $\theta_{ij}^l += \Delta \bar{\theta}_{ij}^l$

Fin-para

Fin-mientras

Algoritmo BACKPROP (incremental) con momentum

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l

$1 \leq l \leq L, 1 \leq i \leq M_l, 1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$, inicializar $\Delta \bar{\theta}_{ij}^l = 0$ Fin-para

/* $\Delta \bar{\theta}_{ij}^l$: almacenamiento de los incrementos para la iteración siguiente */

Mientras no converja

Para toda muestra de aprendizaje $(\mathbf{x}, \mathbf{t}) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$, calcular $\phi_i^l(\mathbf{x})$ y $s_i^l(\mathbf{x}) = g(\phi_i^l(\mathbf{x}))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(\mathbf{x}) = \begin{cases} g'(\phi_i^l(\mathbf{x})) (t_i - s_i^L(\mathbf{x})) & \text{si } l == L \\ g'(\phi_i^l(\mathbf{x})) (\sum_r \delta_r^{l+1}(\mathbf{x}) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$ calcular: $\Delta \theta_{ij}^l = \delta_i^l(\mathbf{x}) s_j^{l-1}(\mathbf{x})$ Fin-para

Fin-para

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$,

Calcular $\Delta \bar{\theta}_{ij}^l = \rho \Delta \theta_{ij}^l + \nu \Delta \bar{\theta}_{ij}^l$ /* Incrém. para ser usado en la iteración siguiente */

Actualizar pesos: $\theta_{ij}^l += \Delta \bar{\theta}_{ij}^l$

Fin-para

Fin-para toda muestra

Fin-mientras

Algoritmo BACKPROP (batch) con amortiguamiento

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l $1 \leq l \leq L, 1 \leq i \leq M_l, 1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Mientras no converja

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

Para toda muestra de aprendizaje $(\mathbf{x}, \mathbf{t}) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$, calcular $\phi_i^l(\mathbf{x})$ y $s_i^l(\mathbf{x}) = g(\phi_i^l(\mathbf{x}))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(\mathbf{x}) = \begin{cases} g'(\phi_i^l(\mathbf{x})) (t_i - s_i^L(\mathbf{x})) & \text{si } l == L \\ g'(\phi_i^l(\mathbf{x})) (\sum_r \delta_r^{l+1}(\mathbf{x}) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$ calcular: $\Delta\theta_{ij}^l += \delta_i^l(\mathbf{x}) s_j^{l-1}(\mathbf{x})$ Fin-para

Fin-para

Fin-para toda muestra

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$,

Actualizar pesos: $\theta_{ij}^l += \rho \Delta\theta_{ij}^l + 2 \rho \lambda \theta_{ij}^l$

Fin-para

Fin-mientras

Algoritmo BACKPROP (incremental) con amortiguamiento

Entrada: Una topología, datos de entrenamiento S , un factor de aprendizaje ρ , pesos iniciales θ_{ij}^l

$1 \leq l \leq L, 1 \leq i \leq M_l, 1 \leq j \leq M_{l-1}$ y condiciones de convergencia

Salidas: Pesos de las conexiones que minimizan el error cuadrático medio de S

Método:

Mientras no converja

Para toda muestra de aprendizaje $(\mathbf{x}_n, \mathbf{t}_n) \in S$

Desde la capa de entrada a la capa de salida ($l = 1, \dots, L$):

Para $1 \leq i \leq M_l$, calcular $\phi_i^l(\mathbf{x})$ y $s_i^l(\mathbf{x}) = g(\phi_i^l(\mathbf{x}))$ Fin-para

Desde la salida a la entrada ($l = L, \dots, 1$),

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

Calcular $\delta_i^l(\mathbf{x}) = \begin{cases} g'(\phi_i^l(\mathbf{x})) (t_i - s_i^L(\mathbf{x})) & \text{si } l == L \\ g'(\phi_i^l(\mathbf{x}_n)) (\sum_r \delta_r^{l+1}(\mathbf{x}_n) \theta_{ri}^{l+1}) & \text{en otro caso} \end{cases}$

Para $0 \leq j \leq M_{l-1}$, calcular $\Delta\theta_{ij}^l = \delta_i^l(\mathbf{x}) s_j^{l-1}(\mathbf{x})$ Fin-para

Fin para

Para $1 \leq l \leq L, 1 \leq i \leq M_l, 0 \leq j \leq M_{l-1}$,

Actualizar pesos: $\theta_{ij}^l += \rho \Delta\theta_{ij}^l + 2 \rho \lambda \theta_{ij}^l$

Fin-para

Fin-para toda muestra

Fin-mientras