2021-11-29

Introduction to Statistical Learning,
James, Witten, Hastie, Tibshirani
(2nd ed, 2021)

Labs in R !!

Νευρωνικά Δίκτυα
Support Vector Machines
Unsupervised Learning (Cluster Analysis).
(Random Forests)

2 εργασίες (3 από 4)
Ατομικά projects: { κεφ. βιβλίου / papers
report
υλοποίηση σε δεδομένα

Neural Networks

Αναπτύγματα με συναρτήσεις βάσης

γραμμικό ως προς b.

$$\boxed{f(x) = \sum_{m=1}^{M} b_m \underline{h_m(x)}}, \quad x \in \mathbb{R}^{p+1} \qquad x_i = (1, x_{1i}, \ldots, x_{pi})$$

π.χ. $h_m(x) = x^{m-1}$

splines, etc.

$\boxed{b_1, \ldots, b_M}$: άγνωστες παράμετροι

Γενίκευση

$$h_m(x) = \sigma(a_m^T x)$$

όπου $\sigma(v)$ = activation function

$$\left(= \frac{1}{1+e^{-v}}\right) \text{ σιγμοειδής}$$

a_m^T : άγνωστες παράμετροι.

Μοντέλο

$$f(x) = \sum_{m=1}^{M} b_m \, \sigma(a_m^T x) =$$

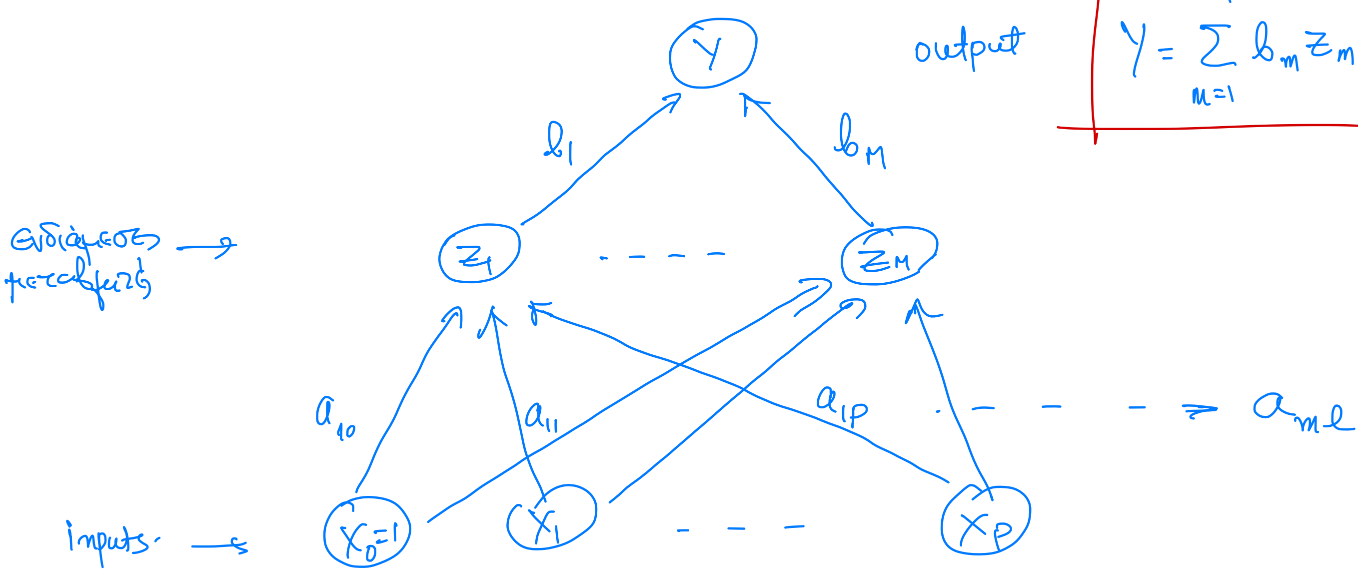
$$= \sum_{m=1}^{M} b_m \frac{1}{1+e^{-a_m^T x}} =$$

$$\boxed{f(x) = \sum_{m=1}^{M} b_m \frac{1}{1+e^{-\sum_{\ell=0}^{p} a_{m\ell} x_\ell}}}$$

μη γραμμικό ως προς παραμέτρους.

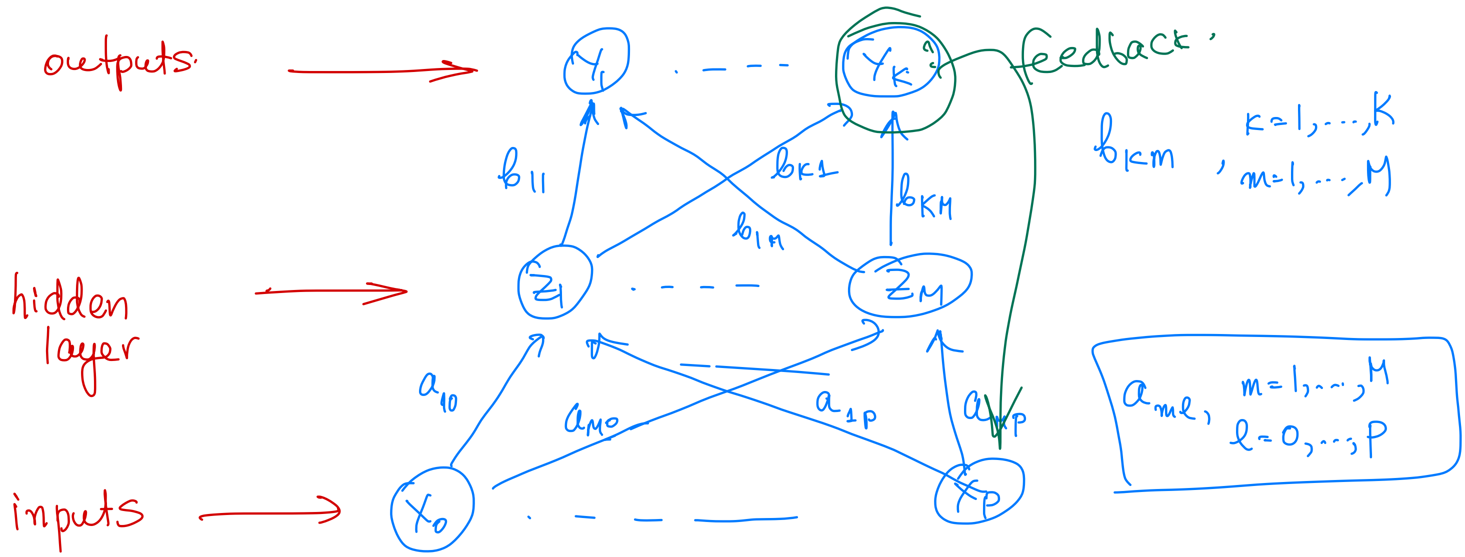
$$= \sum_m b_m z_m$$

output $\boxed{Y = \sum_{m=1}^{M} b_m z_m}$



$$z_m = \sigma\left(\sum_{\ell=0}^{p} a_{m\ell} x_\ell\right)$$

Γενικό Μοντέλο Νευρωνικού Δικτύου με 1 hidden layer



Αντίστοιχο Μοντέλο

$$Z_m = \sigma(a_m^T \cdot X) = \sigma\left(\sum_{\ell=0}^{p} a_{m\ell} X_\ell\right), \quad m=1,\dots,M, \quad \sigma = \text{activation function}$$

$$T_k = b_k^T \cdot Z = \sum_{m=0}^{M} b_{km} Z_m \quad (Z_0 = 1)$$

$$Y_k = g_k(T_k)$$

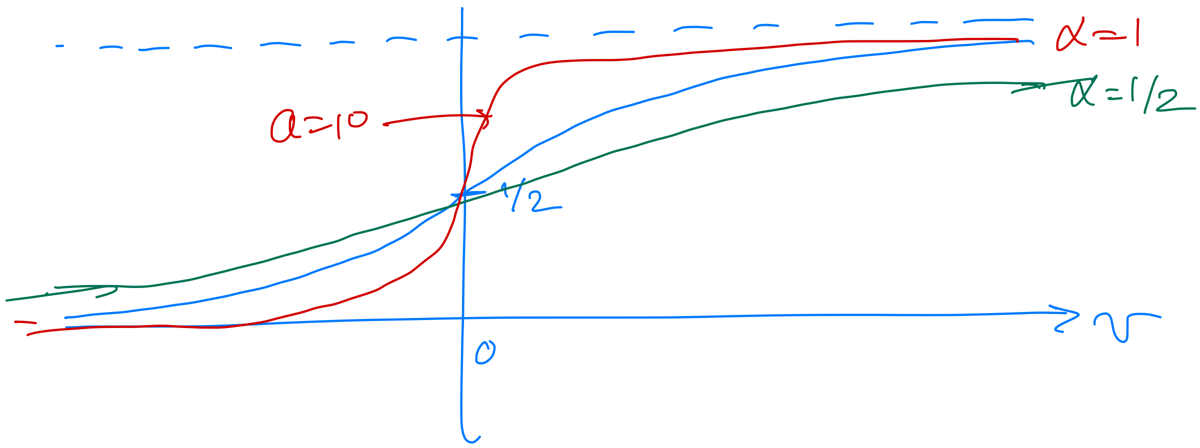
Regression $\quad Y_k = T_k$

Classification $\quad Y_k = \dfrac{e^{T_k}}{\sum_{k=1}^{K} e^{T_k}}$ "softmax" function $\quad \sum_k Y_k = 1$

Classifier: $\quad G(x) = \operatorname{argmax}_k (Y_k)$

Activation Function

$$\sigma(v) = \frac{1}{1+e^{-v}}$$



$$\sigma(\alpha v) = \frac{1}{1+e^{-\alpha v}}$$

$\alpha \to \infty$ $\quad \sigma(\alpha v) \approx \begin{cases} 0 & v<0 \\ 1 & v>0. \end{cases}$ activation

---

$\sigma(a_m^T x)$ $\qquad a_m \in \mathbb{R}^{p+1}$

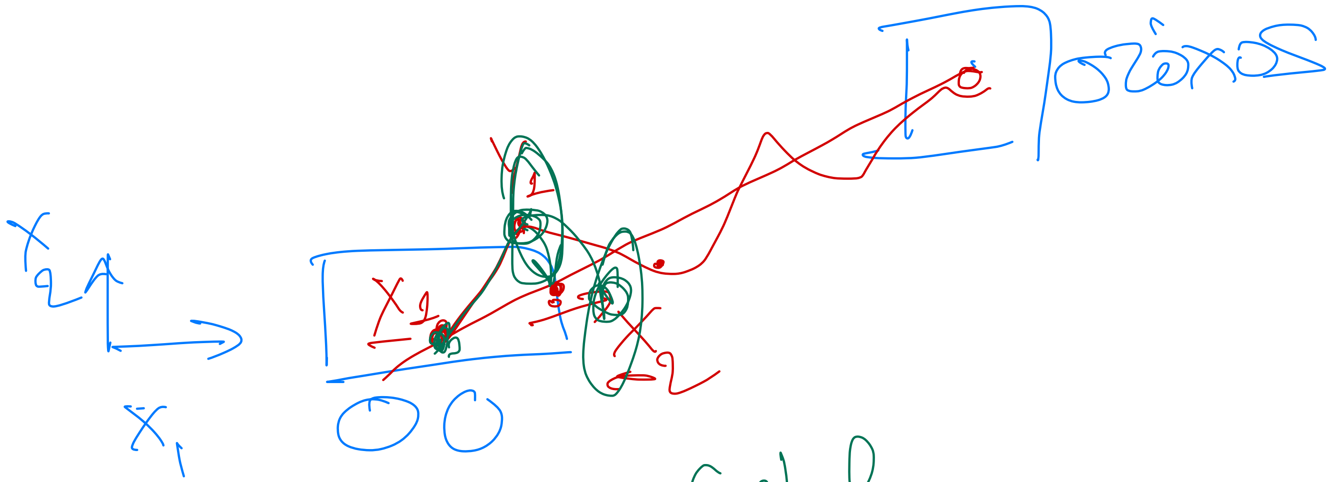
$$a_m = \|a_m\| \cdot \frac{a_m}{\|a_m\|} = \|a_m\| \cdot \tilde{a}_m$$

$$\sigma(a_m^T x) = \sigma\left(\widehat{\|a_m\|} \cdot \tilde{a}_m^T \cdot x\right)$$

Ειδ. περίπτωση

$\sigma(v) = v \Rightarrow \dots \Rightarrow$ γραμμικό μοντέλο

Neural networks με feedback



control

δυν. προγραμματισμός

reinforcement learning

Fitting Άγνωστες παράμετροι ⟹ βάρη (weights)

$\{a_{ml},\ m=1,\dots,M,\ l=0,\dots,p\}$ $\quad M\times(p+1)$

$\{a_m\ ;\ m=1,\dots,M\}$ $\quad a_m\in\mathbb{R}^{p+1}$

$\{b_k,\ k=1,\dots,K\}$, $b_k\in\mathbb{R}^{M+1}$ $\quad \theta=(a_{ml},\ b_{km})$

$Z_0=1$ $\quad \{b_{km},\ m=0,\dots,M\}$ $\quad K\times(M+1)$

$K=10$
$p=10$
$M=5$

$5\cdot 11+10\cdot 6=115$

Συνολικά $M\times(p+1)+K(M+1)$ παράμετροι

Κριτήριο $\quad \theta:=(a_{ml},\ b_{mk},\ \ l=0,\dots,p,\ m=0,\dots,M,\ k=1,\dots,K)$

Regression $\quad R(\theta)=\sum_{i=1}^{N}\sum_{k=1}^{K}\left(y_{ik}-f_k(x_i)\right)^2\ \left(+\lambda\|\theta\|^2\right)$

Classification $\quad R(\theta)=-\sum_{i=1}^{N}\sum_{k=1}^{K} y_{ik}\log f_k(x_i)=-\sum_{i=1}^{N}\log f_{g_i}(x_i)$

$\forall i\ \ \sum_k y_{ik}=1,\ \ y_{ik}\in\{0,1\}$

$\min_\theta R(\theta)$

overfitting !!

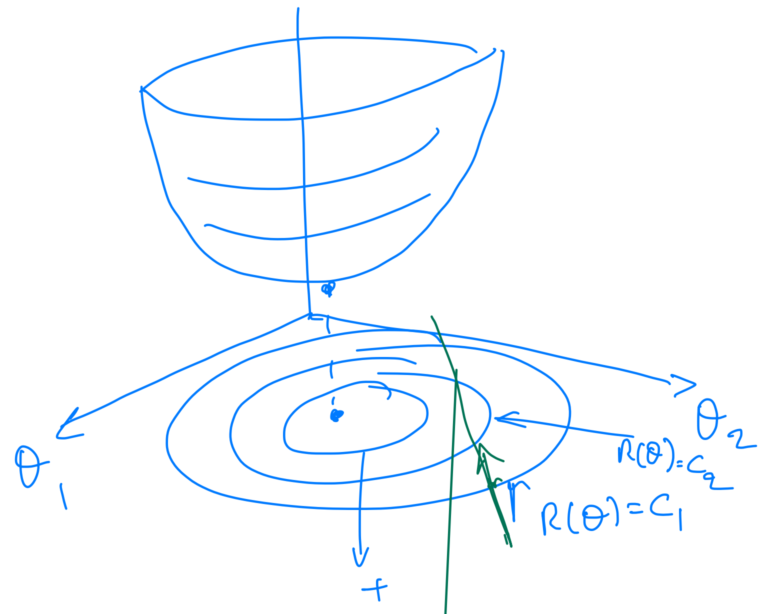
regularization $\to \lambda\sum b_{km}^2$

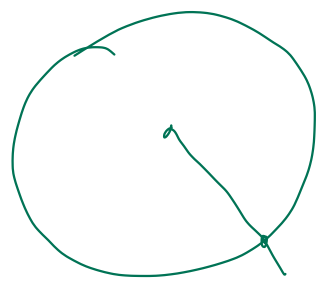
$\to$ early stopping

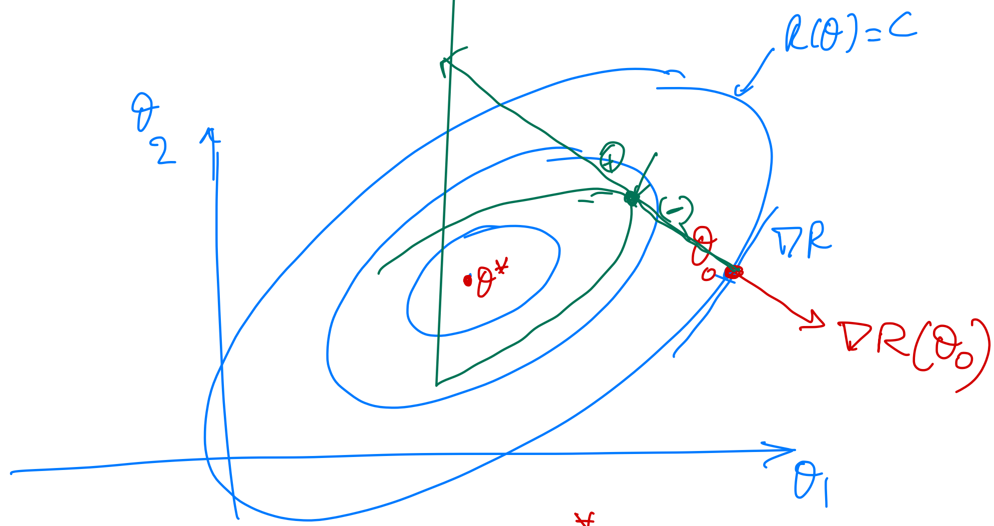
Η μέθοδος Gradient descent για ελαχιστοποίηση κυρτής συνάρτησης

Έστω $R(\theta)$, $\theta \in \mathbb{R}^M$ (εδώ $M=2$)

R: κυρτή (δηλ. $\nabla^2 R(\theta) = \left(\frac{\partial^2 R}{\partial \theta_{ij}}\right)$ θετικά ημιορισμένος)







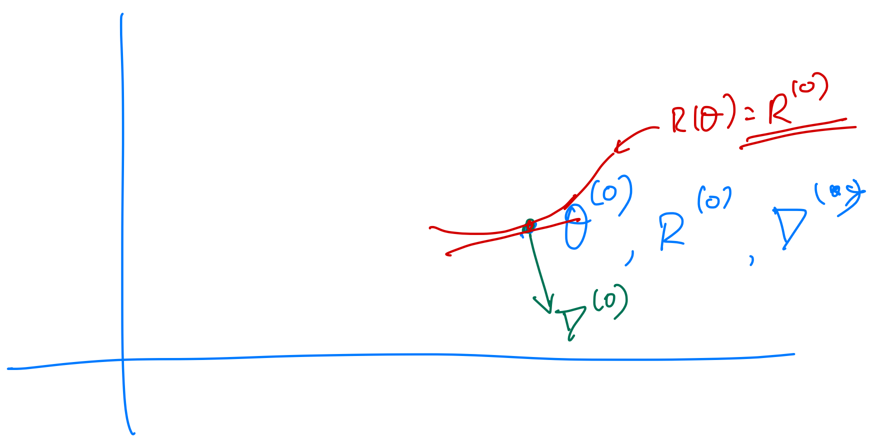
local optimization method.

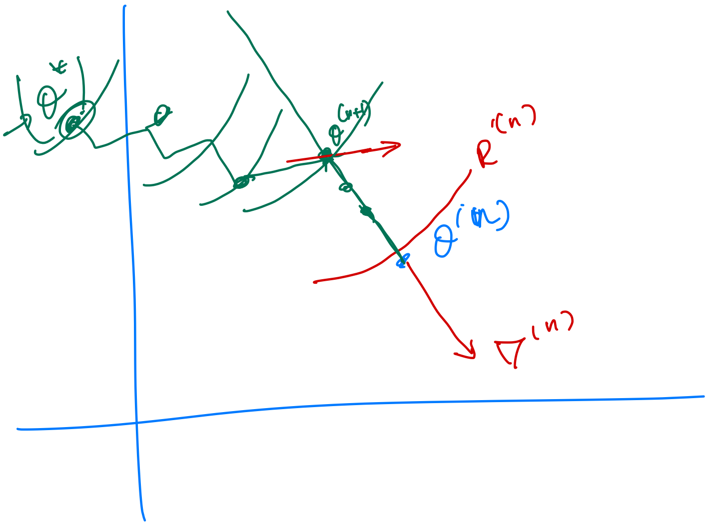
σε κάθε επανάληψη

1) Μια προσέγγιση $\theta^{(n)}$ της θ^*

2) $\left.\begin{matrix} R(\theta^{(n)}) \\ \nabla R(\theta^{(n)}) \end{matrix}\right\}$ υπολογίζονται

$\nabla R(\theta^*) = 0.$





γενική επανάληψη

$$\boxed{\theta^{(n+1)} = \theta^{(n)} - \gamma^{(n)} \nabla R(\theta^{(n)})}$$

[Newton method $x_{n+1} = x_n - \dfrac{f(x_n)}{f'(x_n)}$]

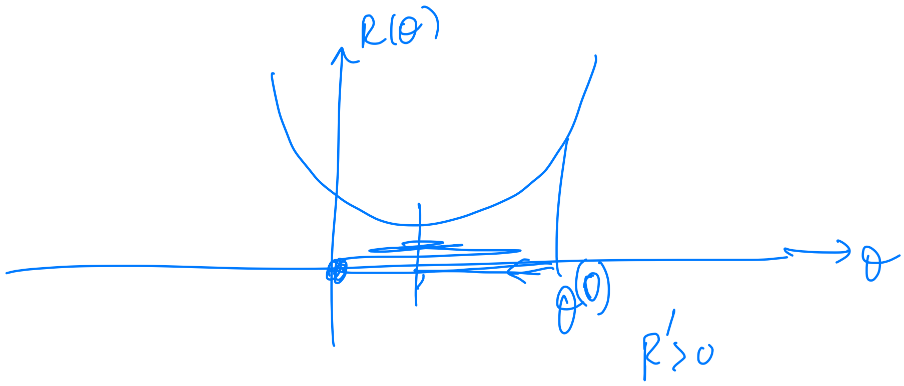
$\gamma^{(n)}$: step size (Learning rate).

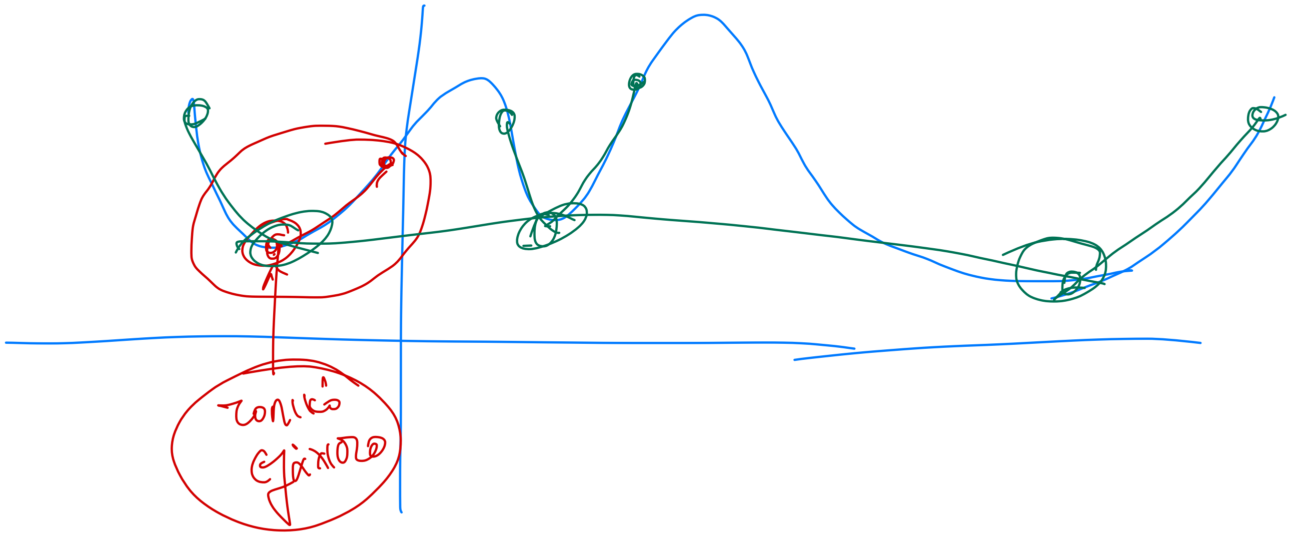
$\gamma^* = ?$

Έστω $h(\gamma) = R(\theta^{(n)} - \gamma \nabla R(\theta^{(n)}))$, $\gamma \in \mathbb{R}$ $(\gamma \geq 0)$

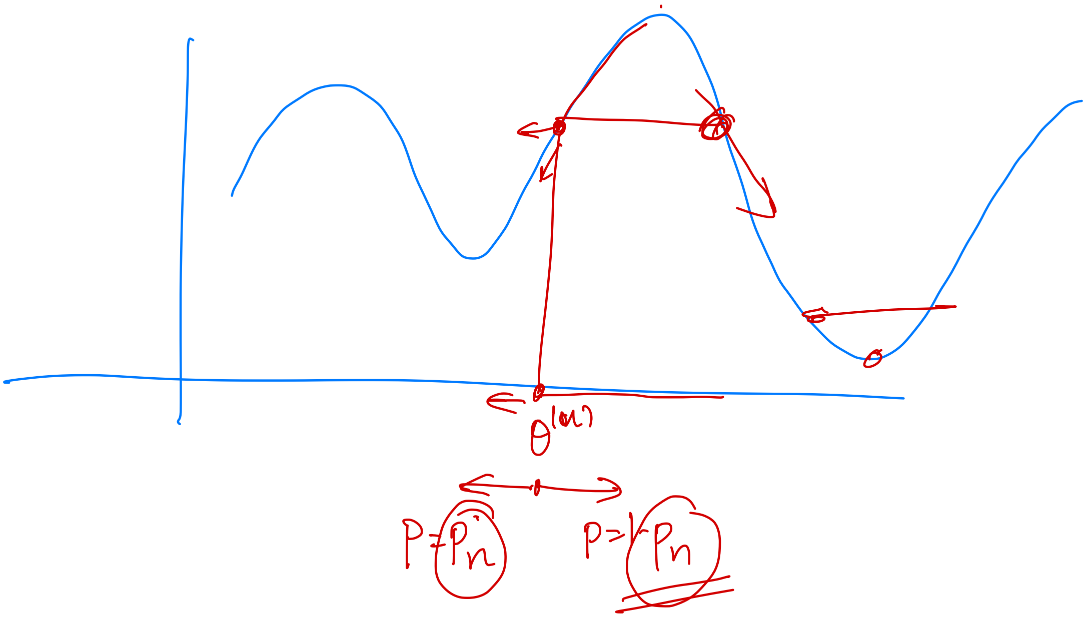
Δ.Ο. $h(\gamma)$ κυρτή $\Rightarrow$ $h'(\gamma) = 0$ $\Rightarrow$ γ^*

Μη κυρτές συναρτήσεις





Simulated annealing



Κατηγορία αλγορίθμων $\boxed{\text{stochastic approximation}}$

Fitting in a neural network - gradient descent.

$$f_k(x) = g_k(b_k^T z) = g_k\left(\sum_{m=0}^{M} b_{km} z_m\right)$$

$$z_m = \sigma(a_m^T \cdot x) = \sigma\left(\sum_{l=0}^{P} a_{ml} x_l\right)$$

$$f_k(x) = g_k\left(\sum_{m=0}^{M} b_{km}\, \sigma\left(\sum_{l=0}^{P} a_{ml} x_l\right)\right)$$

$$R(\theta) = \sum_{i=1}^{N} \sum_{k=1}^{K} \left(y_{ik} - g_k\left(\sum_{m=0}^{M} b_{km} z_{mi}\right)\right)^2$$

$$z_{mi} = \sigma\left(\sum_{l=0}^{P} a_{ml} X_{il}\right)$$

$$\theta^{(n+1)} = \theta^{(n)} - \gamma^{(n)} \cdot \underline{\underline{\nabla R(\theta^{(n)})}} \; ?$$

$$b_{km}^{(n+1)} = b_{km}^{(n)} - \gamma^{(n)} \frac{\partial R}{\partial b_{km}}(\theta^{(n)})$$

$$a_{ml}^{(n+1)} = a_{ml}^{(n)} - \gamma^{(n)} \frac{\partial R}{\partial a_{ml}}(\theta^{(n)})$$

$= ?$

$$R(\theta) = \sum_{i=1}^{N} \underbrace{\sum_{k=1}^{K} \Big(y_{ik} - \underbrace{g_k \Big(\sum_{m=0}^{M} b_{km} z_{im} \Big)}_{f_k(x_i)} \Big)^2}_{R_i(\theta)} =$$

$$= \sum_{i=1}^{N} R_i(\theta) \, ,$$

$$\frac{\partial R}{\partial b_{km}} = \sum_{i=1}^{N} \frac{\partial R_i(\theta)}{\partial b_{km}}$$

$$\frac{\partial R_i}{\partial b_{km}} = \frac{\partial}{\partial b_{km}} \Big[\big(y_{ik} - f_k(x_i) \big)^2 \Big] = \frac{\partial}{\partial b_{km}} \Big[\Big(y_{ik} - g_k \Big(\sum_{m=0}^{M} b_{km} z_{mi} \Big) \Big)^2 \Big]$$

$$= 2 \cdot \big(y_{ik} - f_k(x_i) \big) \Big(- \frac{\partial f_k(x_i)}{\partial b_{km}} \Big)$$

$b_{k0} z_{0i} + b_{k1} z_{1i} + b_{km} z_{mi} + \cdots b_{kM} z_{Mi}$

$$= -2 \big(y_{ik} - f_k(x_i) \big) \cdot g_k' \Big(\sum_{m=0}^{M} b_{km} z_{mi} \Big) \cdot \frac{\partial}{\partial b_{km}} \Big(\sum_{m=0}^{M} b_{km} z_{mi} \Big)$$

$$= \underbrace{-2 \big(y_{ik} - f_k(x_i) \big) \cdot g_k' \Big(\sum b_{km} z_{mi} \Big)}_{\delta_{ki}} \cdot z_{mi} = \delta_{ki} z_{mi}$$

$$z_i = (z_{0i}, \dots, z_{Mi})$$

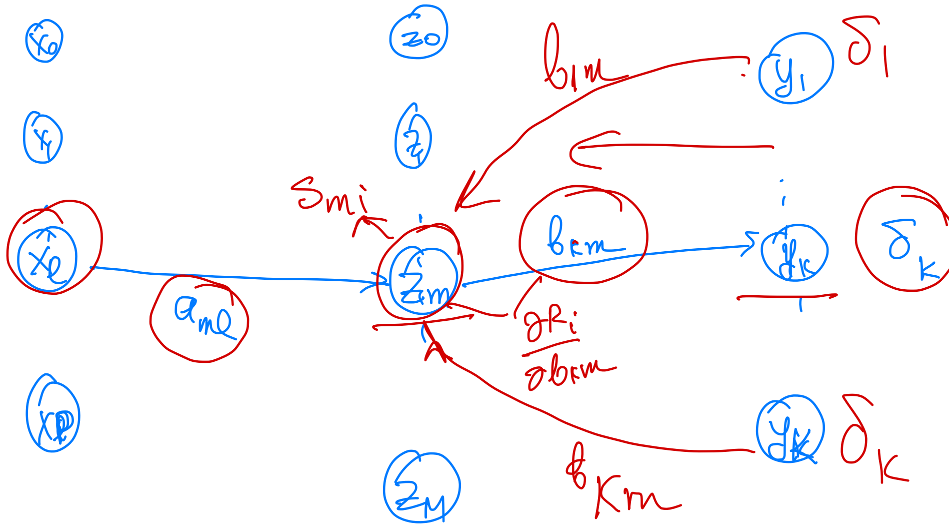
Αντίστοιχα

$$\frac{\partial R_i}{\partial a_{ml}} = - \sum_{k=1}^{K} 2 \big(y_{ik} - f_k(x_i) \big) g_k' (b_k^T z_i) \cdot b_{km} \, \sigma' (a_m^T x_i) \cdot x_{il}$$

$$\frac{\partial R_i}{\partial a_{ml}} = s_{mi} \cdot x_{il}$$

$$\boxed{s_{mi} = \sigma' (a_m^T x_i) \cdot \sum_{k=1}^{K} b_{km} \delta_{ki}}$$

back propagation



$$\delta_k = (y_k - f_k(x))\left[-2\, g_k'(\beta_k^T z)\right]$$

$$\theta^{(n)} \longrightarrow f_\theta \quad x \to z \to f_k$$

[forward pass]

$$\frac{\partial R_i}{\partial \beta_{km}} = \delta_{ki}\, z_{mi}$$

$$\frac{\partial R_i}{\partial \alpha_{ml}} = \Bigg[\overbrace{\sigma'(\alpha_m^T x_i) \cdot \sum_{k=1}^{K} \underline{\beta_{km}}\, \delta_{ki}}^{s_{mi}}\Bigg] x_{il}$$