ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ

Τεχνητά νευρωνικά δίκτυα και ευφυή υπολογιστικά συστήματα

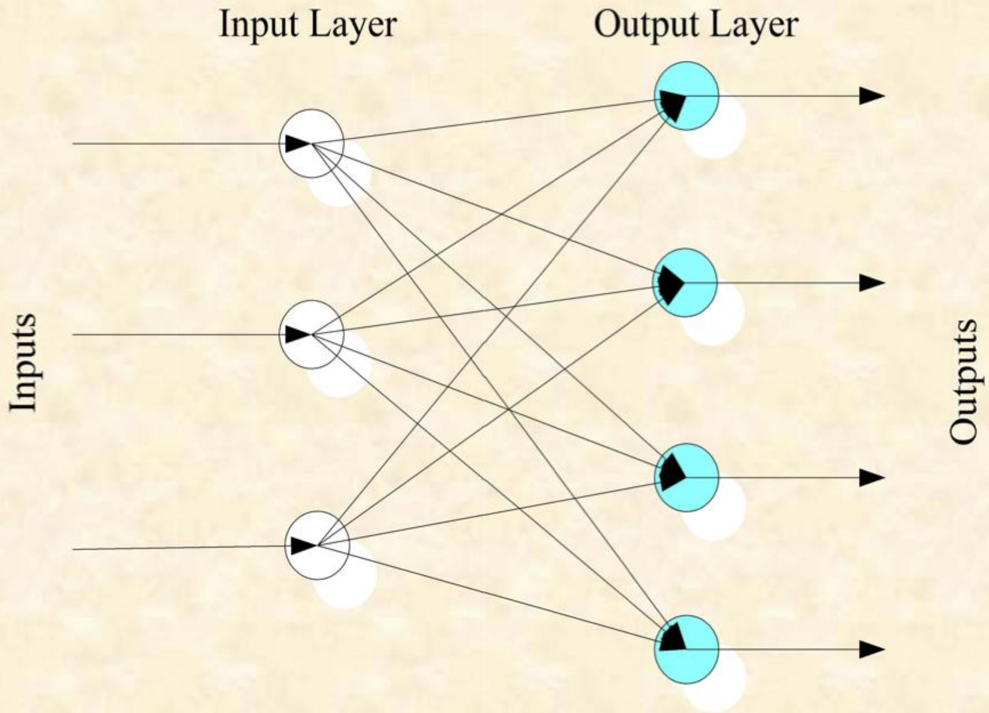
2Η ΔΙΑΛΕΞΗ - PERCEPTRONS & ΜΑΘΗΣΗ

19/10/2021

Perceptron

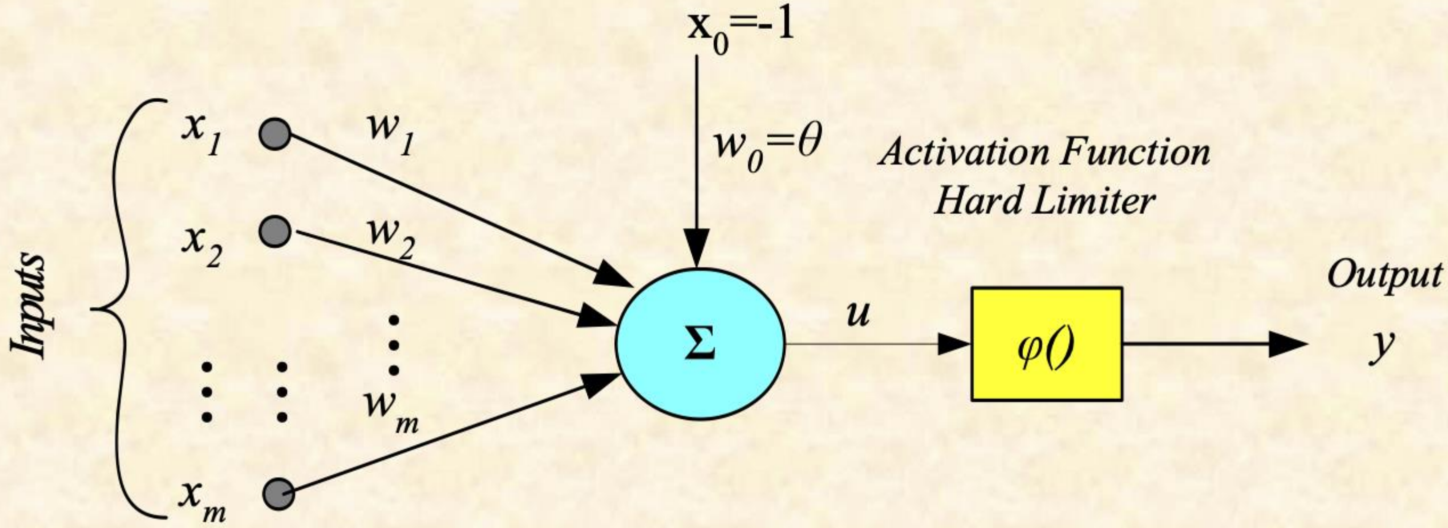
Ο αλγόριθμος μάθησης του Perceptron

- Το Perceptron είναι η απλούστερη μορφή Νευρωνικού Δικτύου, το οποίο χρησιμοποιείται για την ταξινόμηση γραμμικά διαχωριζόμενων προτύπων. Αποτελείται από ένα επίπεδο υπολογιστικών νευρώνων που ακολουθούν το μοντέλο McCulloch – Pitts.



Ο αλγόριθμος μάθησης του Perceptron

Για ευκολία, και χωρίς απώλεια της γενικότητας, στη συνέχεια της ενότητας θα μελετήσουμε ένα απλό perceptron αποτελούμενο **μόνο από ένα νευρώνα**



Ο αλγόριθμος μάθησης του Perceptron

Ο σκοπός του Perceptron είναι να ταξινομήσει σωστά το σύνολο των εισόδων (προτύπων):

$x_1, x_2, \ldots, x_m$

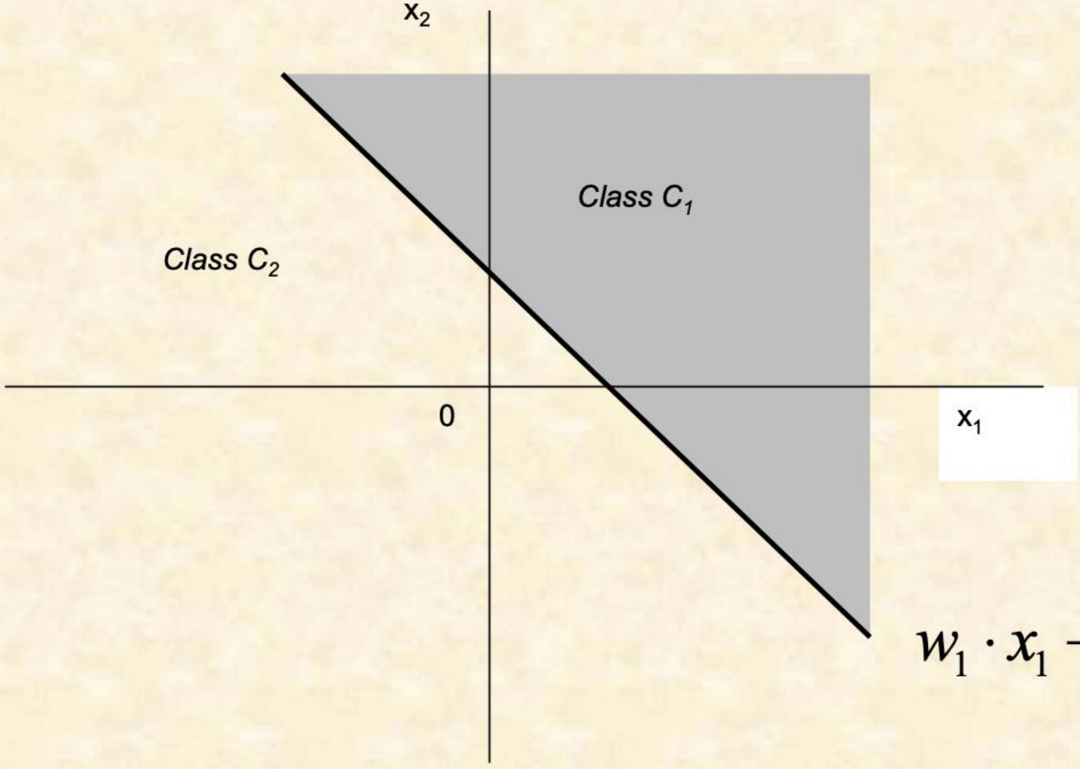
σε μία από τις δύο κλάσεις: C_1 και C_2.

Ο κανόνας απόφασης για την ταξινόμηση είναι να *αναθέτει το σημείο που αναπαριστούν τα $x_1, x_2, \ldots, x_m$ στην κλάση C_1, εάν η έξοδος y του Perceptron είναι +1 και στην κλάση C_2 εάν η έξοδος y του Perceptron είναι -1*.

Γραμμική Διαχωρισιμότητα

- Στην απλούστερη μορφή του **Perceptron** (ένας νευρώνας μόνο) υπάρχουν δύο περιοχές απόφασης που διαχωρίζονται από ένα υπερεπίπεδο το οποίο ορίζεται από τη σχέση:

$$\upsilon = \sum_{j=1}^{m} w_j x_j - \theta = 0$$



$$w_1 \cdot x_1 + w_2 \cdot x_2 - \theta = 0$$

Ο Αλγόριθμος Σύγκλισης του Perceptron για ένα Perceptron με m εισόδους και 1 υπολογιστικό νευρώνα - 1/2

Μεταβλητές και Παράμετροι

x(n) = ο πίνακας εισόδων (m + 1) x 1 = $[-1, x_1(n), x_2(n), \dots x_m(n)]^T$

w(n) = ο πίνακας των βαρών (m + 1) x 1 = $[\theta(n), \mathbf{w}_1(n), \mathbf{w}_2(n), \dots, \mathbf{w}_m(n)]^T$

θ(n) = το κατώφλι (threshold)

y(n) = η πραγματική έξοδος (actual response)

d(n) = η επιθυμητή έξοδος (desired response) – καθορίζεται από το πρόβλημα

η = η παράμετρος μάθησης (learning - rate parameter), θετική σταθερά < 1

Βήμα 1: ***Αρχικοποίηση***

Θέτουμε το διάνυσμα βαρών ίσο με το μηδενικό: **w**(0) = 0.

Βήμα 2: ***Ενεργοποίηση***

Στο χρόνο n, ενεργοποιούμε το Perceptron εφαρμόζοντας το διάνυσμα εισόδου **x**(n).

Ο Αλγόριθμος Σύγκλισης του Perceptron για ένα Perceptron με m εισόδους και 1 υπολογιστικό νευρώνα – 2/2

Βήμα 3: ***Υπολογισμός πραγματικής απόκρισης***

Υπολογίζουμε την πραγματική απόκριση (έξοδο) του Perceptron:

$$\upsilon(n) = \mathbf{w}^T(n) \cdot \mathbf{x}(n) \qquad \textbf{και} \qquad y(n) = \mathrm{sgn}(\upsilon(n)) = \begin{cases} +1, \upsilon(n) \geq 0 \\ -1, \upsilon(n) < 0 \end{cases}$$

όπου sgn(•) είναι η συνάρτηση προσήμου (sign function)

Βήμα 4: ***Προσαρμογή διανύσματος βαρών***

Προσαρμόζουμε τα βάρη του Perceptron σύμφωνα με τον κανόνα:

$$\mathbf{w}(n+1) = \mathbf{w}(n) + \eta \cdot [d(n) - y(n)] \cdot \mathbf{x}(n)$$

όπου:

$$d(n) = \begin{cases} +1, \text{ αν το } \mathbf{x}(\mathrm{n}) \text{ ανήκει στην κλάση } \mathrm{C}_1 \\ -1, \text{ αν το } \mathbf{x}(\mathrm{n}) \text{ ανήκει στην κλάση } \mathrm{C}_2 \end{cases}$$

Βήμα 5: *Αυξάνουμε το χρόνο n κατά μια μονάδα και επιστρέφουμε στο βήμα 2.*

<u>ΜΗ ΓΡΑΜΜΙΚΗ ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ ΧΩΡΙΣ ΠΕΡΙΟΡΙΣΜΟΥΣ</u>

<u>UNCONSTRAINED NON-LINEAR OPTIMIZATION</u>

ΠΡΟΒΛΗΜΑ

min $F(x) \rightarrow F'(x) = 0$

ΛΥΣΗ

Επαναληπτικές Τεχνικές

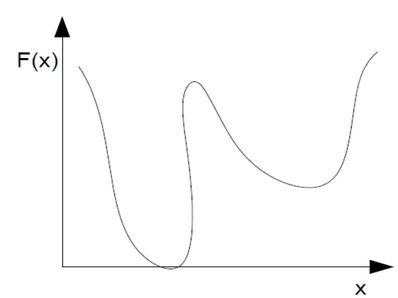
$F: R^N \rightarrow R$

$x_o \in R^N$

Α) Μέθοδος του Newton

$$x_{\kappa+1} = x_\kappa + s_\kappa^N$$

$$\underbrace{\nabla^2 F(x_\kappa)}_{Hessian} S_\kappa^N = -\lambda \nabla F(x_\kappa)$$



Σύγκλιση : Τετραγωνική

$$\left\| x_{\kappa+1} - x_* \right\| \le \beta\gamma \left\| x_\kappa - x_* \right\|^2$$

όπου,

$F^{'}(x) = 0$

β,γ: όρια της $1^{ης}$ / $2^{ης}$ παραγώγου της F

εάν $H_c = \nabla^2 F(x_c)$ θετικά ορισμένη

τότε $\nabla F(x_c)^T s_c^N = -\{\nabla F(x_c)\}^T \{\nabla f(x_c)\} \langle 0$

Άρα $F(x_c + \delta s_c^N) \langle F(x_c)$

B) Steepest Descent (βαθύτατης καθόδου)

$$x_{\kappa+1} = x_\kappa - \lambda_\kappa \nabla F(x_\kappa)$$

Σύγκλιση : Γραμμική

$$\lim_{\kappa \to \infty} \sup \left\| \frac{x_{\kappa+1} - x_*}{x_\kappa - x_*} \right\| \leq C$$

$$\left. \begin{matrix} e_{v\max} \\ e_{v\min} \end{matrix} \right| = \begin{matrix} \mu\acute{\varepsilon}\gamma\iota\sigma\tau\eta \\ \varepsilon\lambda\acute{\alpha}\chi\iota\sigma\tau\eta \end{matrix}$$ ιδιοτιμή της $\nabla^2 F(x_\kappa)$

$$C = \left[\frac{e_{v\max} - e_{v\min}}{e_{v\max} + e_{v\min}} \right]$$

If $e_{v\max} >> e_{v\min}$ $C \approx 1$ Πολύ Αργή Σύγκλιση

$e_{v\max} \approx e_{v\min}$ Καλή Σύγκλιση

Γ) Επεκτάσεις της Μεθόδου του Newton

α) $H_\kappa = \nabla^2 F(x_\kappa) + \mu_\kappa I$

όπου $\mu_k = 0$ αν $\nabla^2 F$ θετικά ορισμένη

$\mu_k > 0$ αλλού

Λύση μέσω του Cholesky Decomposition

β) Backtracking πάνω στη Newton direction (Line Search)

γ) Προσδιορισμός περιοχής όπου εμπιστευόμαστε το τοπικά υπολογισμένο μοντέλο να αναπαραστήσει τη συνάρτηση (Trust Region).

Δ) Quasi – Newton Τεχνικές
Σύγκλιση : Υπερ- Γραμμική (super linear)

$|x_{\kappa+1} - x_*| \langle\ c_\kappa\ |x_\kappa - x_*|$ εξαρτώμενη από την τιμή του c_k

Ε) Μη – Γραμμικά Ελάχιστα Τετράγωνα

Επίλυση Συστήματος μη Γραμμικών Εξισώσεων

i) Gauss-Newton (γραμμικόποιηση γύρω από το x_*)

$$x_{\kappa=+1} = x_\kappa - (J^T J)^{-1} R(x_\kappa),$$

όπου

$$F(x) = \frac{1}{2}\{R(x)\}^T R(x) = \frac{1}{2}\sum_{i=1}^{M} r_i^2(x)$$

και $\nabla F(x) = J^T(x) R(x)$

$$\nabla^2 F(x) = J^T(x) J(x) + \underbrace{S(x)}$$

ο όρος β΄ βαθμού αγνοείται

Τοπικά τετραγωνική Σύγκλιση (αν $R(x_*)=0$)

Μη συγκλίνουσα τοπικά (σε έντονα μη γραμμικά προβλήματα)

ii) Marquardt Levenberg

Χρησιμοποιούμε τον πίνακα $J^T J + \mu I$ ως προσέγγιση του Hessian.

Συναρτήσεις Σφάλματος F(x)

Sum absolute error (SAE):

$$SAE \equiv \sum_{i=1}^{P} \sum_{j=1}^{N} |e_{ji}|$$

Mean absolute error (MAE):

$$MAE \equiv \frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{N} |e_{ji}|$$

Sum squared error (SSE):

$$SSE \equiv \sum_{i=1}^{P} \sum_{j=1}^{N} |e_{ji}|^2$$

Mean squared error (MSE):

$$MSE \equiv \frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{N} |e_{ji}|^2$$

Relative absolute error (RAE):

$$RAE \equiv \frac{\sum_{i=1}^{P} \sum_{j=1}^{N} |y_{ji} - \tilde{y}_{ji}|}{\sum_{i=1}^{P} \sum_{j=1}^{N} |y_{ji} - \bar{y}_{ji}|}$$

Relative squared error (RSE):

$$RAE \equiv \frac{\sum_{i=1}^{P} \sum_{j=1}^{N} |y_{ji} - \tilde{y}_{ji}|^2}{\sum_{i=1}^{P} \sum_{j=1}^{N} |y_{ji} - \bar{y}_{ji}|^2}$$

Minkowski:

$$E_{Minkowski} \equiv \left(\sum_{i=1}^{P} \sum_{j=1}^{N} e_{ji}{}^{2} \right)^{\frac{1}{n}}$$

Logistic:

$$E_{logistic} \equiv k^2 \sum_{i=1}^{P} \sum_{j=1}^{N} \boldsymbol{ln}\left(\boldsymbol{cosh}\left(\frac{e_{ji}}{k} \right) \right)$$

Cross-entropy error:

$$E_{cross\ entropy} \equiv \frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{N} r_j \boldsymbol{log}\left(\frac{\mathbf{1}}{\tilde{y}_j} \right) = -\frac{1}{P} \sum_{i=1}^{P} \sum_{j=1}^{N} r_j \boldsymbol{log} \tilde{y}_j$$

Other distance metrics: **Mahalanobis, Jaccard measure, Tanimoto coefficient, Cosine measure, …**