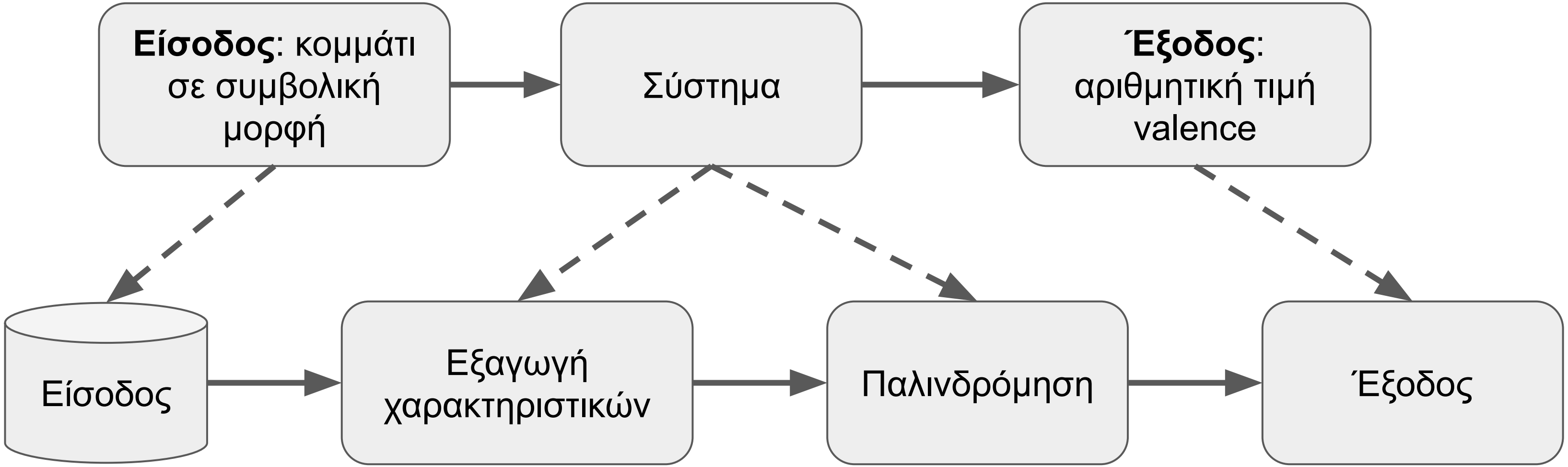
Εισαγωγή στα Τεχνητά Νευρωνικά Δίκτυα

Από τη γραμμική και λογιστική παλινδρόμηση, στα Τεχνητά Νευρωνικά Δίκτυα

Παλινδρόμηση - παράδειγμα για τη γενική διαδικασία

Στόχος: Η δημιουργία ενός συστήματος που παίρνει ως είσοδο ένα κομμάτι μουσικής σε συμβολική μορφή και μας λέει πόσο valence έχει



Π.χ.:

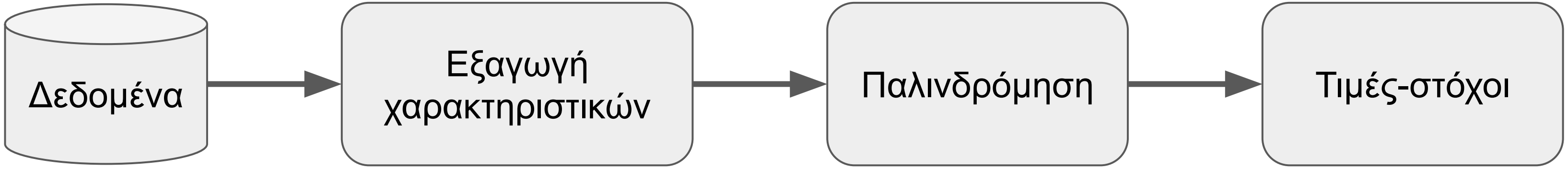
Κομμάτι από EMOPIA

Relative pitch class profile + rhythm features 16 διαστάσεις

Γραμμική παλινδρόμηση (πως ρυθμίζουμε τις παραμέτρους;)

Τιμή valence

Παλινδρόμηση: εκπαίδευση (ρυθμιση παραμέτρων)



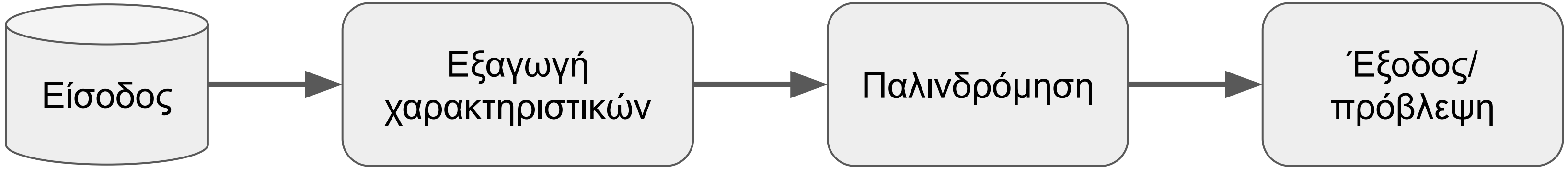
Π.χ.: EMOPIA

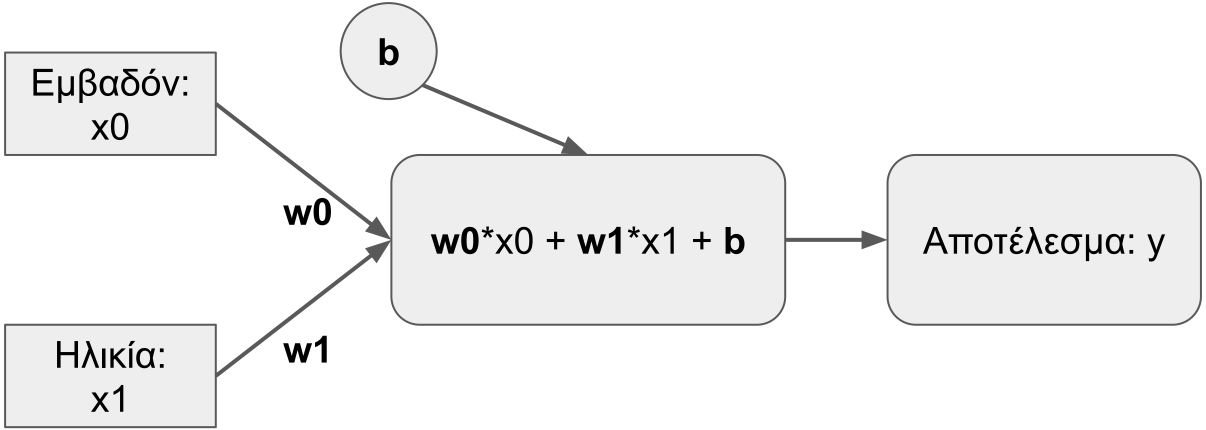
Relative pitch class profile
+ rhythm features
16 διαστάσεις

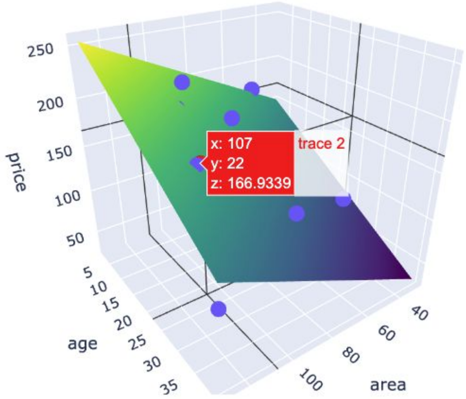
Γραμμική
παλινδρόμηση

Τιμή valence

Παλινδρόμηση: παράδειγμα με τα σπίτια

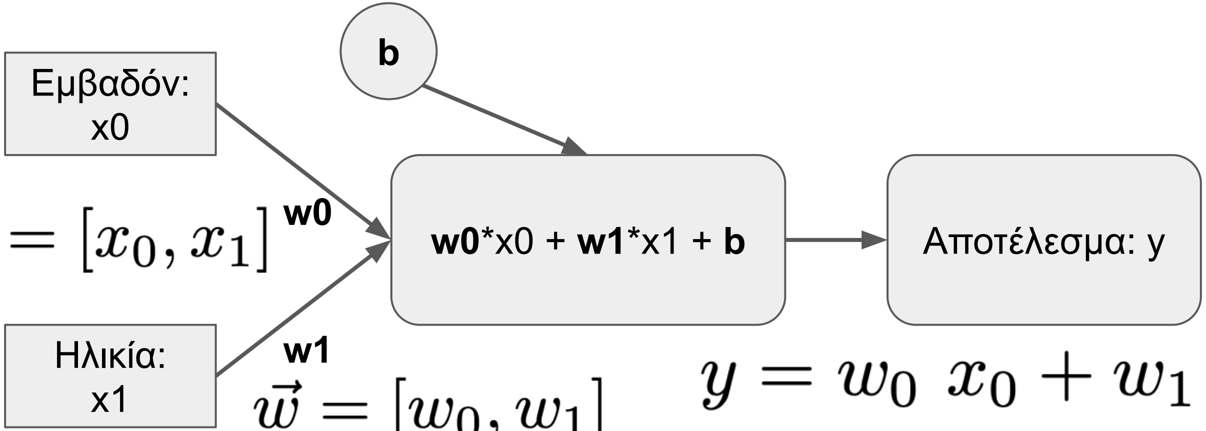






Στόχος εκπαίδευσης: μάθε τις βέλτιστες τιμές **w0, w1** και **b** για να υπάρχουν οι ελάχιστες δυνατές απώλειες σε ένα σύνολο “εκπαίδευσης”**.** Δηλαδή να “χωρέσει” βέλτιστα το υπερεπίπεδο. Μετά κάνουμε προβλέψεις που πέφτουν πάνω στο υπερεπίπεδο.

Μαθηματικοί συμβολισμοί



$$\vec{x} = [x_0, x_1]$$

$$\vec{w} = [w_0, w_1]$$

$$y = w_0\ x_0 + w_1\ x_1 + b$$

Συμβολισμός αθροίσματος: $$y = \sum_{i=0}^{1} w_i\ x_i + b$$

Συμβολισμός εσωτερικού γινομένου: $$y = \langle \vec{w}, \vec{x} \rangle + b$$

Εσωτερικό γινόμενο (dot ή inner product) και άλλα…

$$y = \boxed{\sum_{i=0}^{1} w_i \; x_i} + b$$

Το εσωτερικό γινόμενο είναι αριθμός (και τελικά αθροίζεται με τον αριθμό b για να μας δώσει μια απόφαση για το y).

$$y = \boxed{\langle \vec{w}, \vec{x} \rangle} + b$$

Στα κόκκινα πλαίσια συμβολίζεται το ίδιο πράγμα με διαφορετικό συμβολισμό: το εσωτερικό γινόμενο

$$y = \boxed{|\vec{w}| \; |\vec{x}| \; cos(\theta(\vec{w}, \vec{x}))} + b$$

Η τελευταία έκφραση, με το συνημίτονο, είναι πολύ σημαντική για διαισθητική αντίληψη. Θα το μελετήσουμε αργότερα…

Υπάρχουν και άλλα ήδη γινομένων;
Ναι! Μπορούμε να ορίσουμε ένα γινόμενο όπως σχεδόν όπως θέλουμε…

Π.χ. **Σημαντικό**: Γινόμενο Hadamard (ανά-στοιχείο ή element-wise) - προφέρεται “Ανταμάρ”.

$$\vec{a} = [a_0, a_1, a_2]$$
$$\vec{b} = [b_0, b_1, b_2]$$
$$\vec{a} \odot \vec{b} = [a_0 \; b_0, a_1 \; b_1, a_2 \; b_2]$$

Αλλα παραδείγματα: Διανυσματικό (cross), Εξωτερικό (outer), ευθύ (direct) κ.α.

https://colab.research.google.com/drive/1a58zohIsq1WI0Z3-5yIMj8BcCXO-0hYF?usp=sharing

Είδη αθροίσματος και πολλαπλασιασμού πινάκων που θα μας απασχολήσουν κυρίως

$$\vec{a} = [a_0, a_1, a_2]$$
$$\vec{b} = [b_0, b_1, b_2]$$

Διανυσματικό άθροισμα

$$\vec{a} + \vec{b} = [a_0 + b_0, a_1 + b_1, a_2 + b_2]$$

Ευθύ γινόμενο
Γινόμενο Hadamard

Διανύσματα

Πίνακες

$$A + B = C \to C_{ij} = A_{ij} + B_{ij}$$

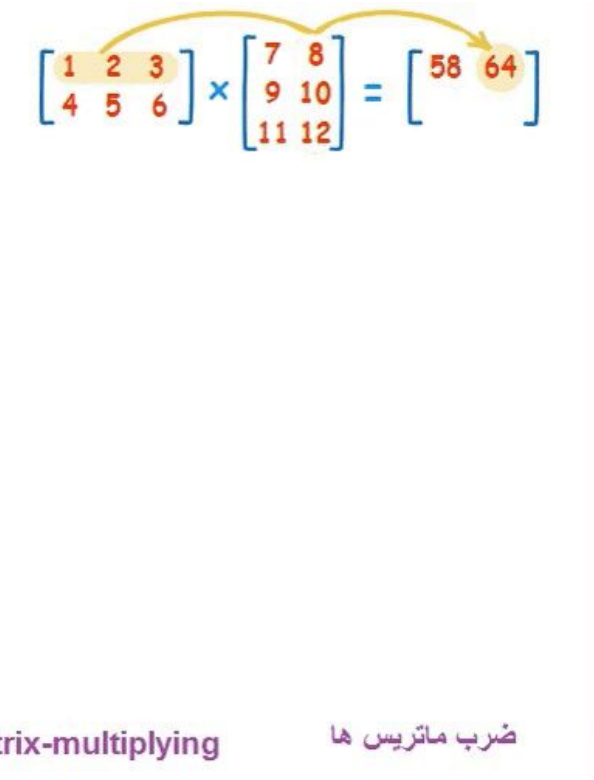
Άθροισμα πινάκων: οι πίνακες πρέπει να έχουν ίδιες διαστάσεις, πχ NxM. Το αποτέλεσμα θα είναι πίνακας ίδιων διαστάσεων και το κάθε στοιχείο του θα είναι το άθροισμα των αντίστοιχων στοιχείων στους αρχικούς πίνακες.

$$A \odot B = C \to C_{ij} = A_{ij}\ B_{ij}$$

Υπάρχει επίσης η έννοια του γινομένου Hadamard (element-wise) σε πίνακες, δηλαδή όπως γινόμενο αντίστοιχων στοιχείων του πίνακα.

Υπάρχει κι η “κλασική” έννοια του πολλαπλασιασμού πινάκων που φαίνεται στην επόμενη διαφάνεια…

“Κλασικός” πολλαπλασιασμός σε πίνακες



Η “κλασική” έννοια του πολλαπλασιασμού πινάκων:

- NxK επί KxM δίνει έναν πίνακα: NxM.
- Ο πρώτος δίνει τις γραμμές (N) και ο δεύτερος τις στήλες (M).
- Πρέπει οι “εσωτερικές” τους διαστάσεις να είναι ίδιες (K).
- Στο αποτέλεσμα, το στοιχείο στην i γραμμή και j στήλη είναι το εσωτερικό γινόμενο της γραμμής i του πρώτου και της στήλης j του δεύτερου.

$$Y = A\ X \qquad y_{ij} = \langle \vec{a_{i:}}, \vec{x_{:j}} \rangle$$

$\vec{a_{i:}}$ is the array of the i-th row (all columns)

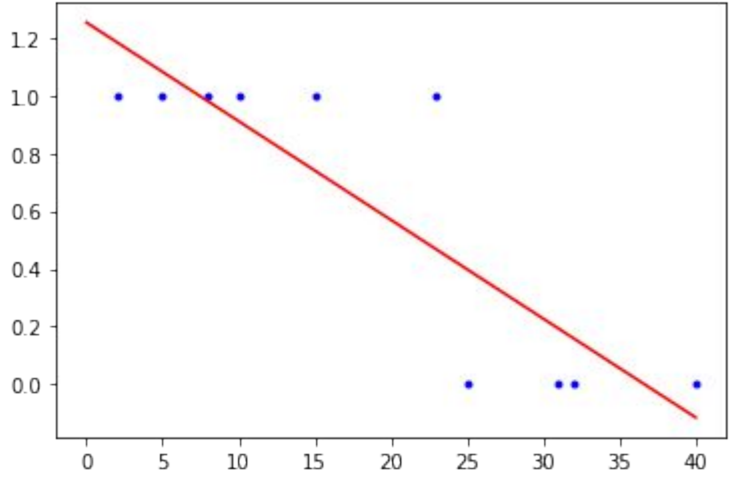
$\vec{x_{:j}}$ is the array of the j-th column (all rows)

https://commons.wikimedia.org/wiki/File:Matrix_multiplication_.gif

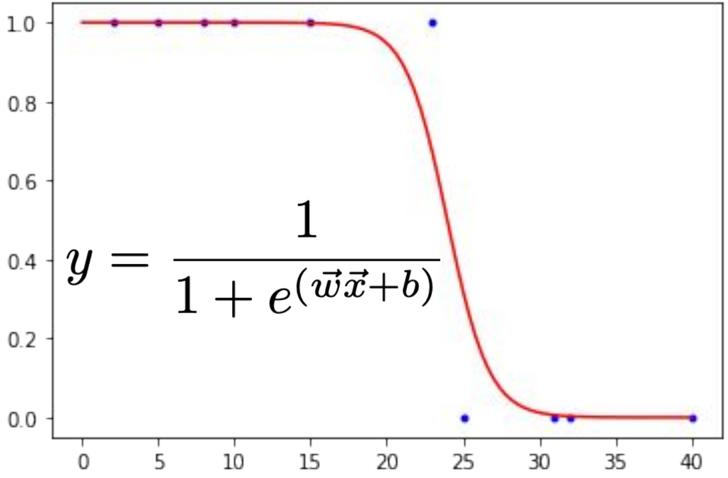
Κατηγοριοποίηση και λογιστική παλινδρόμηση

Αν η επιθυμητή έξοδος δεν είναι τιμή του valence, αλλά μία κατηγορία;
Π.χ. Αν είναι χαρούμενο ή λυπητερό κομμάτι;

Σε τέτοιου τύπου δεδομένα, η γραμμική παλινδρόμηση δεν “χωράει” βέλτιστα.

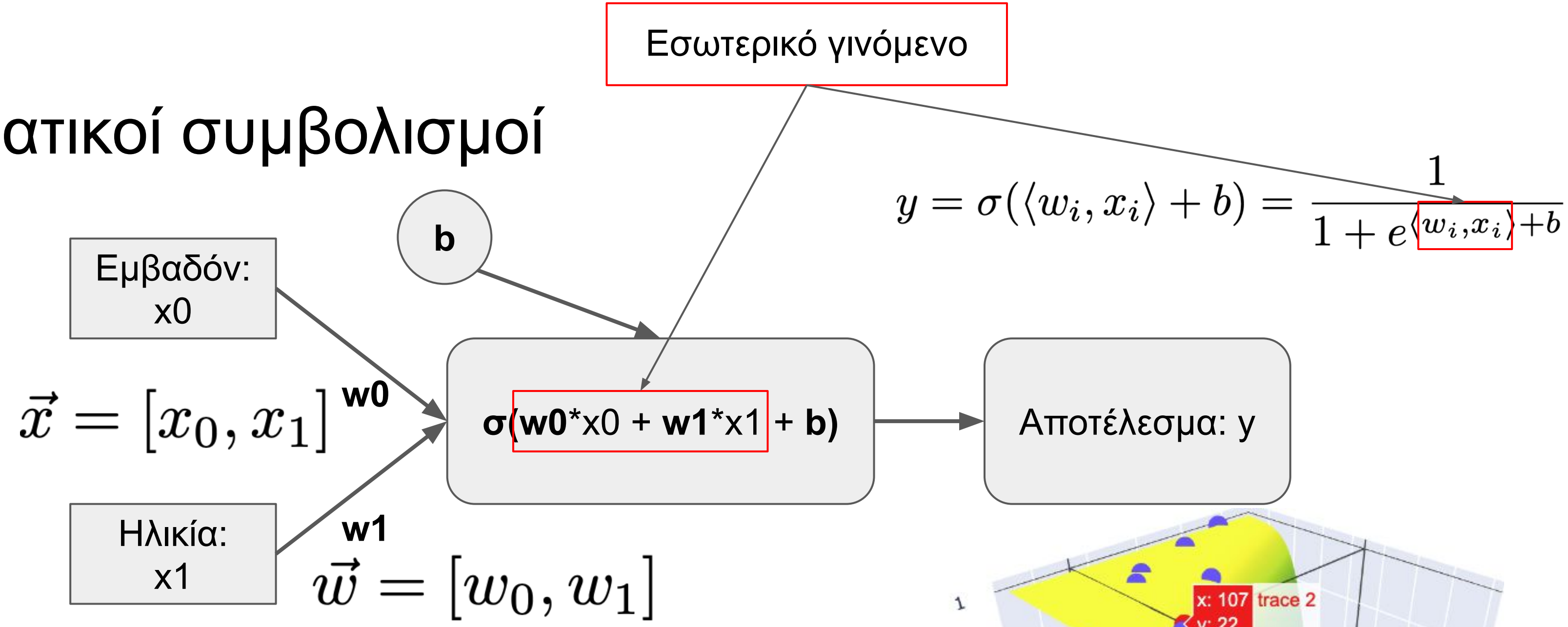


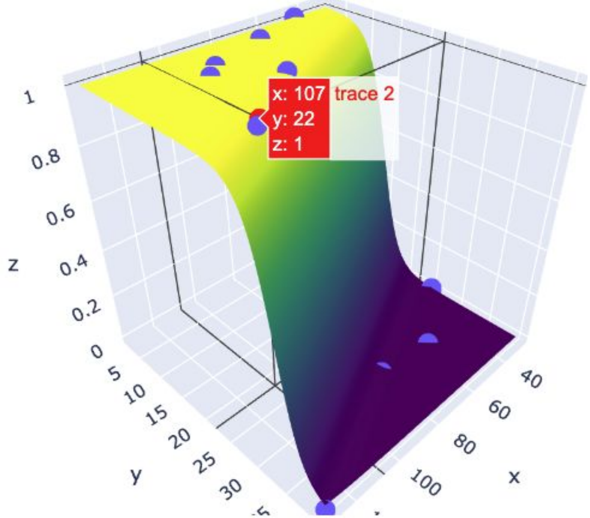
Όμως η λογιστική συνάρτηση προσαρμόζεται βέλτιστα.



$$y = \frac{1}{1 + e^{(\vec{w}\vec{x}+b)}}$$

Μαθηματικοί συμβολισμοί

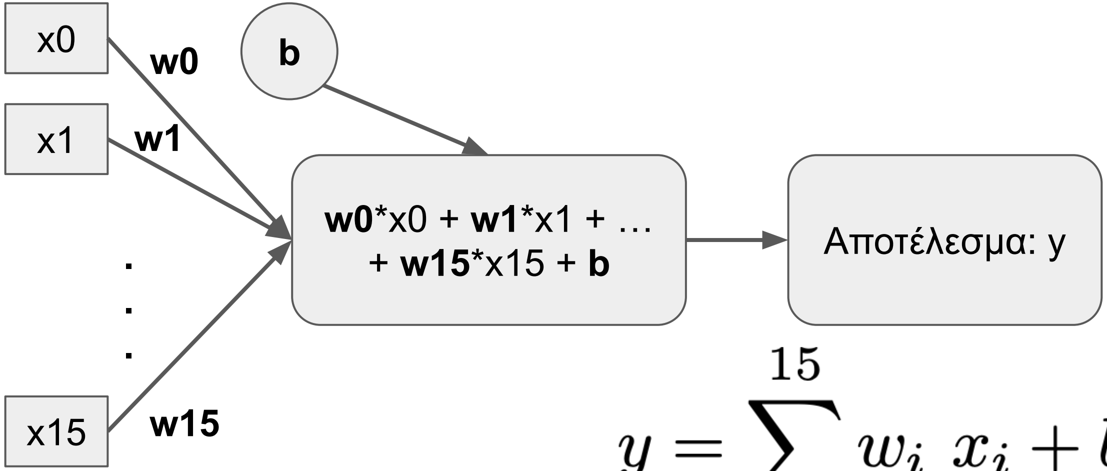




Στόχος εκπαίδευσης- παραμένει ο ίδιος: μάθε τις βέλτιστες τιμές **w0, w1** και **b** για να υπάρχουν οι ελάχιστες δυνατές απώλειες σε ένα σύνολο "εκπαίδευσης"**.** Δηλαδή να "χωρέσει" βέλτιστα το υπερεπίπεδο. Μετά κάνουμε προβλέψεις που πέφτουν πάνω στο υπερεπίπεδο.

Επιστροφή στο παράδειγμα με το EMOPIA

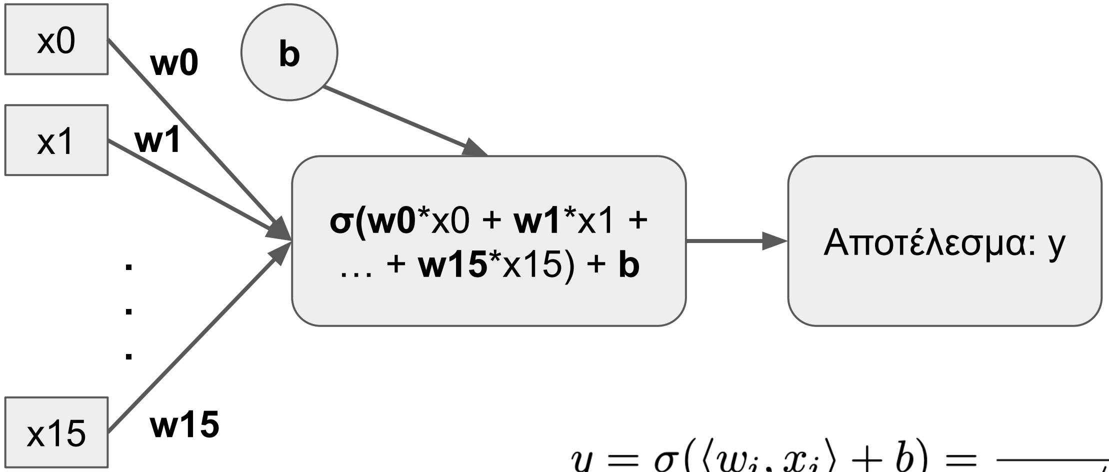
Γραμμική παρεμβολή:
Εκπαίδευση: βρες κατάλληλες τιμές για τα wi και για το b, έτσι ώστε να υπολογίζεται βέλτιστα η προσαρμογή στο σύνολο δεδομένων εκπαίδευσης.
Σύστημα: υπολόγισε τιμή valence για δεδομένη είσοδο.



$$y = \sum_{i=0}^{15} w_i \; x_i + b$$

Επιστροφή στο παράδειγμα με το EMOPIA

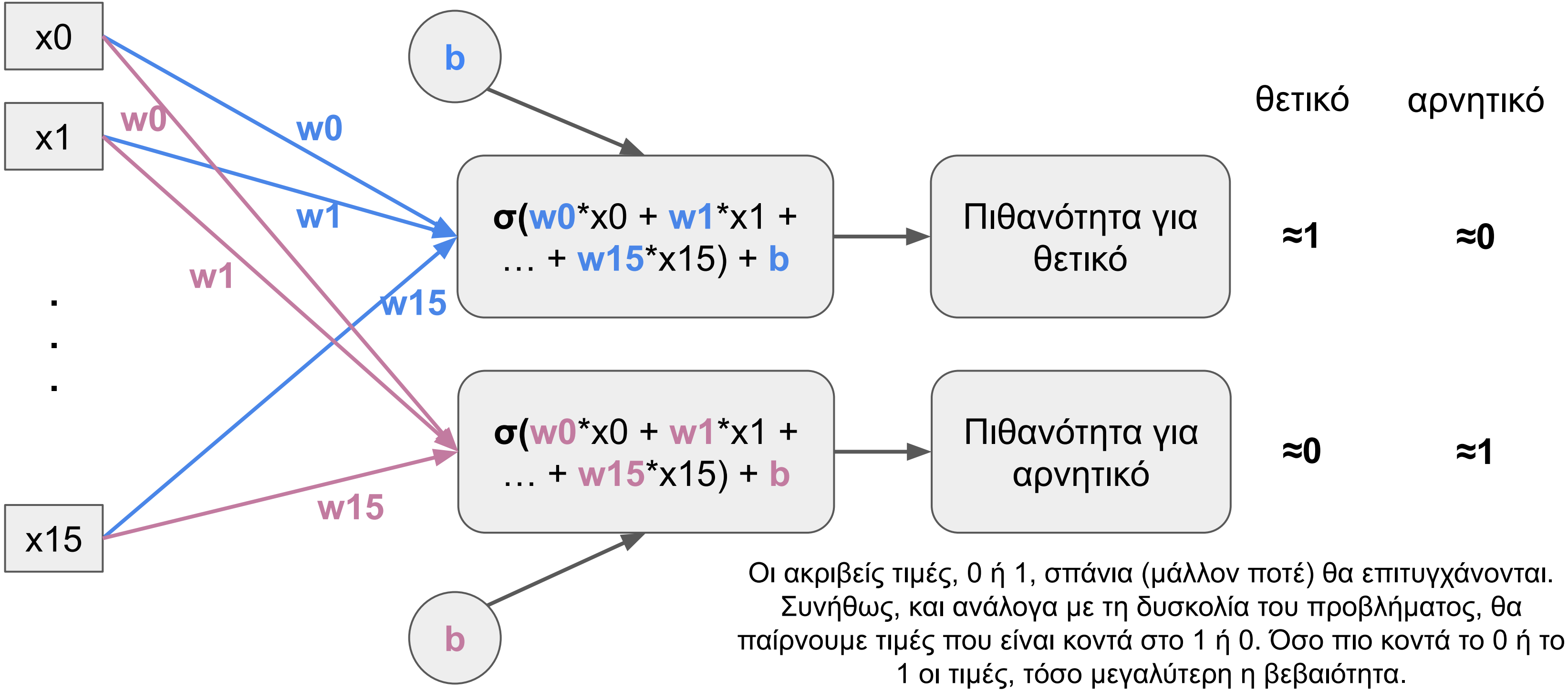
Λογιστική παλινδρόμηση για κατηγοριοποίηση:
Εκπαίδευση: βρες κατάλληλες τιμές για τα wi και για το b, έτσι ώστε να υπολογίζεται βέλτιστα η προσαρμογή στο σύνολο δεδομένων εκπαίδευσης.
Σύστημα: υπολόγισε αν το valence είναι αρνητικό ή θετικό (0/1) για δεδομένη είσοδο.



$$y = \sigma(\langle w_i, x_i \rangle + b) = \frac{1}{1 + e^{\langle w_i, x_i \rangle + b}}$$

Επιστροφή στο παράδειγμα με το EMOPIA

Άλλη προσέγγιση: Διαφορετικές έξοδοι για το αρνητικό και το θετικό πρόσημο του valence.



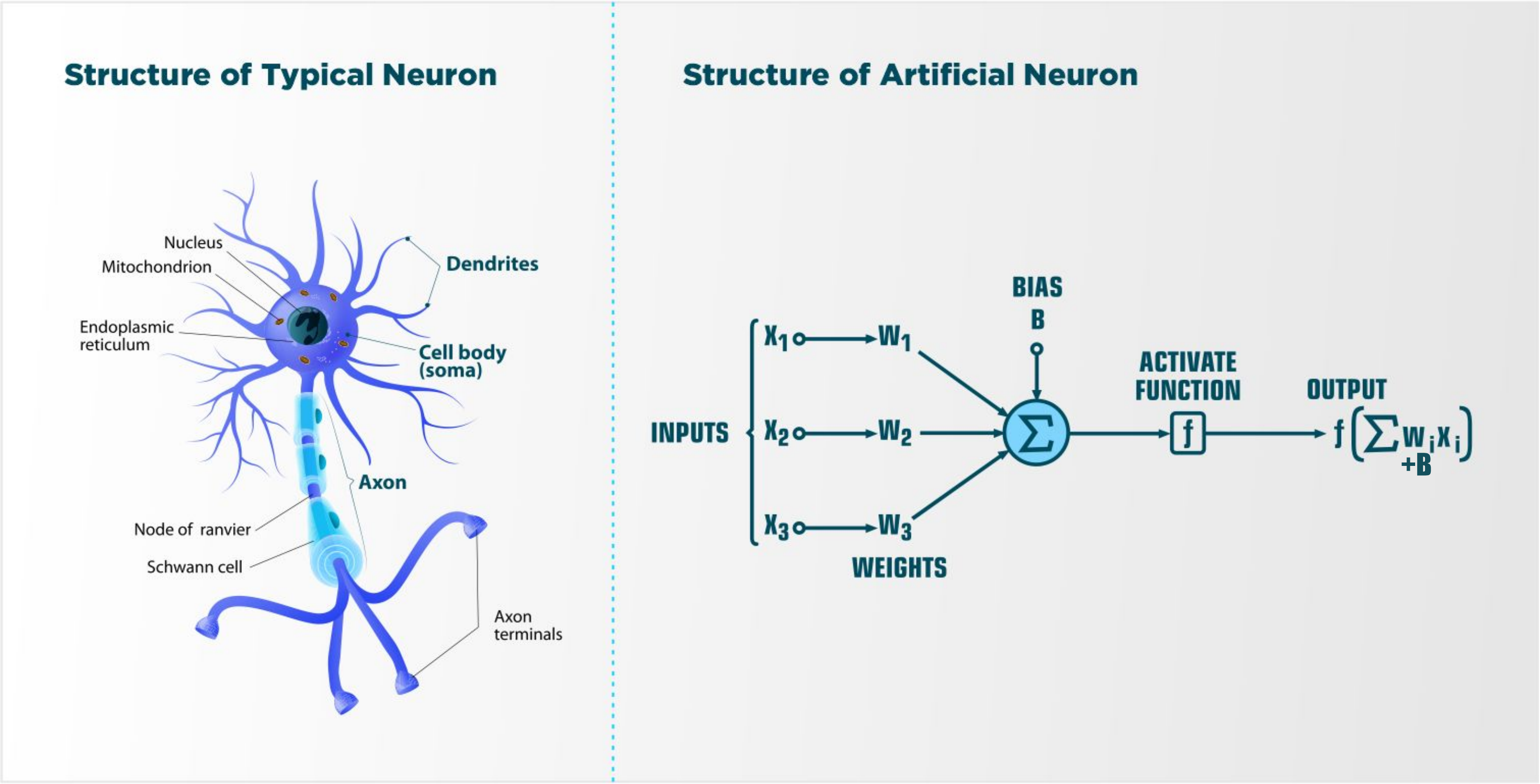
Οι ακριβείς τιμές, 0 ή 1, σπάνια (μάλλον ποτέ) θα επιτυγχάνονται. Συνήθως, και ανάλογα με τη δυσκολία του προβλήματος, θα παίρνουμε τιμές που είναι κοντά στο 1 ή 0. Όσο πιο κοντά το 0 ή το 1 οι τιμές, τόσο μεγαλύτερη η βεβαιότητα.

Αναπαράσταση σε πίνακες

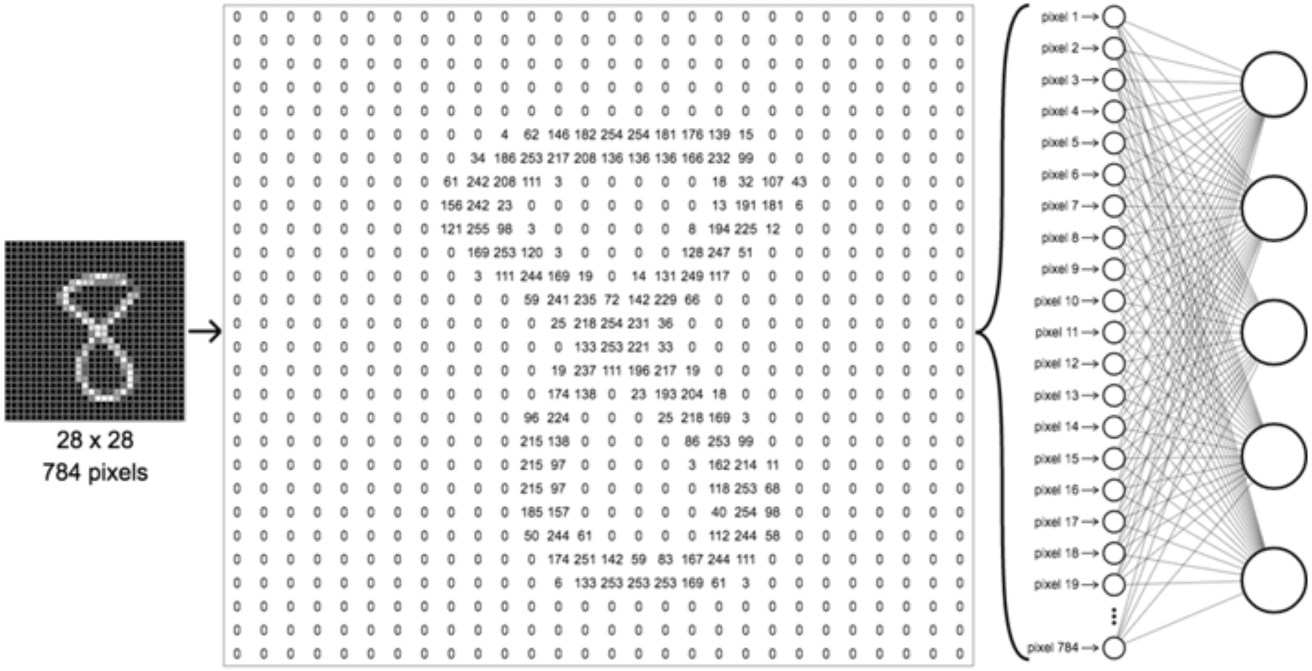
$$\underbrace{\begin{bmatrix} w0 & w1 & \cdots & w15 \\ w0 & w1 & \cdots & w15 \end{bmatrix}}_{2x16} \cdot \underbrace{\begin{bmatrix} x0 \\ x1 \\ \vdots \\ x15 \end{bmatrix}}_{16x1} + \underbrace{\begin{bmatrix} b \\ b \end{bmatrix}}_{2x1} \longrightarrow \sigma(\,.\,)$$

Στη σειγμοειδή: Δύο είσοδοι - δύο έξοδοι (μία για το θετικό και μία για το αρνητικό valence).

Τεχνητά νευρωνικά δίκτυα



Πρόβλεψη χειρόγραφων αριθμητικών χαρακτήρων



Πρόβλεψη για 0

Πρόβλεψη για 1

Πρόβλεψη για 2

.
.
.