

ΑΝΑΓΝΩΡΙΣΗ ΠΡΟΤΥΠΩΝ (PATTERN RECOGNITION)

Σέργιος Θεοδωρίδης
Κωνσταντίνος Κουτρούμπας

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Άλλοι τύποι νευρωνικών δικτύων

- Αυτοοργανούμενοι χάρτες (Self-organizing maps - SOMs)
- Αναδρομικά νευρωνικά δίκτυα (Recurrent Neural Networks): γενικής δομής, δίκτυα Hopfield, δίκτυα Hamming)
- Long short-term memory (LSTM) networks
- Convolutional neural networks (CNNs)

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Θεωρητικά αποτελέσματα

- Έστω ένα ΝΔ **δύο επιπέδων** (ένα «κρυμμένο» επίπεδο νευρώνων και το επίπεδο εξόδου με ένα νευρώνα γραμμικής συνάρτησης εξόδου)

$$\hat{g}(\mathbf{x}) = \sum_{i=1}^K \theta_{K+1,i} f(\theta_i^T \mathbf{x}) + \theta_{K+1,0}$$

K νευρ. 1ου επιπέδου
 1 γραμ. νευρ. εξόδου
($K+1$)

Θεώρημα (Cybenko, 1989):

Παρόμοιο αποτέλεσμα

Έστω

ισχύει για τα δίκτυα **RBF**.

(a) $g(\mathbf{x})$ μια **συνεχής συνάρτηση** πάνω σε συμπαγές σύνολο $S \subset \mathbb{R}^l$ και (b) $\varepsilon > 0$.

Τότε

υπάρχει $K(\varepsilon)$ και δίκτυο **δύο επιπέδων** ώστε

$$|g(\mathbf{x}) - \hat{g}(\mathbf{x})| < \varepsilon, \quad \forall \mathbf{x} \in S$$

Δεν εμφανίζεται το l

Παρατηρήσεις:

- Το **σφάλμα προσέγγισης** μειώνεται με ρυθμό $O(1/K)$.
- Το θεώρημα δε λέει **πόσο μεγάλο** πρέπει να είναι το K . Μπορεί να **απαιτείται** πολύ **μεγάλος αριθμός νευρώνων** K (**ρηχό** (shallow) ή “**παχύ**” (“fat”) δίκτυο).
- **Μία λύση:** Η χρήση δικτύων **περισσότερων επιπέδων** με **λιγότερους κόμβους** σε κάθε επίπεδο (**βαθιά** (deep) ή “**υψηλά**” (“tall”) δίκτυα).

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Η λογική πίσω από τη χρήση **πολλαπλών επιπέδων** νευρώνων:

Τα **διαδοχικά επίπεδα** παράγουν όλο και πιο **αφηρημένες αναπαραστάσεις** του προτύπου εισόδου.

1^ο επ.: Υπερεπιπεδα

2^ο επ.: περιοχές

3^ο επ.: κλάσεις

Γιατί η προσέγγιση αυτή είναι **λογική**?

Διότι **μιμείται** τον τρόπο με τον οποίο είναι **δομημένος** ο **εγκέφαλος** των **θηλαστικών**.

Έχει κάθε **«βαθιά» αρχιτεκτονική** μια **ισοδύναμη «ρηχή» αρχιτεκτονική** (με <3 επίπεδα)?

Ναι, αλλά μια **«ρηχή» αρχ.** μπορεί να περιλαμβάνει **υπερβολικά μεγαλύτερο αριθμό νευρώνων** στα (συνολικά λιγότερα) επίπεδά του, **σε σχέση** με την αντίστοιχη **«βαθιά» αρχιτεκτονική** ⇒

Ένα δίκτυο είναι **συμπαγές** αν **περιλαμβάνει σχετικά λίγες παραμέτρους**, προς **εκπαίδευση**.

Οι **«βαθιές» αρχ.** περιλαμβάνουν **σημαντικά λιγότερες παραμέτρους** από τις αντίστοιχες **«ρηχές»** ⇒

Αναμένεται οι **«βαθιές» αρχ.** να έχουν **καλύτερη ικανότητα γενίκευσης**. (δεδομένου ότι θα είναι αρκετά μεγάλες ώστε να δύνανται να λύσουν το πρόβλημα)

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Σχόλια πάνω στην εκπαίδευση των νευρωνικών δικτύων

Η εκπαίδευσή τους μπορεί να γίνει **δύσκολη**, ειδικά για ≥ 2 επίπεδα.

Για αρκετό καιρό υπήρχε η **πεποίθηση** ότι η **δυσκολία** ήταν συνέπεια της **σύγκλισης** σε ένα «**ρηχό**» **τοπικό ελάχιστο**.

Νέα αποτελέσματα δείχνουν ότι το **πρόβλημα είναι** τα **saddle points**.

Σε **χώρους υψηλής διάστασης** ο **αριθμός των saddle points πολ/ζεται**. Αυτό μπορεί να **επιβραδύνει** δραματικά τη **σύγκλιση**.

Υπό προϋποθέσεις, σε δίκτυα μεγάλου μεγέθους, **τα περισσότερα τοπικά ελάχιστα** δίνουν μικρές τιμές για τη συνάρτηση κόστους

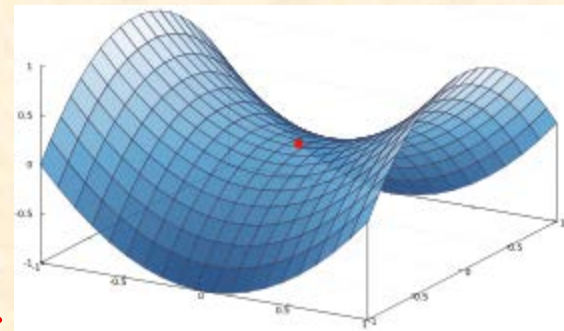
Σε δίκτυο με **ένα κρυφό επίπεδο** νευρώνων, με **ReLU** συν. εξόδου, το **σφάλμα γενίκευσης φράσσεται** από $O(\epsilon + 1/\sqrt{N}) \Rightarrow$ ϵ : bound of the sq. norm of the gradient mat.

Περισσότερα σημεία οδηγούν σε **αυξημένη ικανότητα γενίκευσης**.

Η επιτυχία των νευρ. δικτ. «**βαθίας**» αρχ. **οφείλεται**:

(a) στη **διαθεσιμότητα** συν. δεδομένων μεγ. μεγέθους

(b) στη **διαθεσιμότητα** αυξημένης **υπολογιστικής ισχύος**.



ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

- Πολύ αποδοτικές τεχνικές σε προβλήματα αναγνώρισης και ταξινόμησης εικόνας.
- Τα συστατικά μέρη τους είναι
 - (a) Ο τελεστής συνέλιξης (convolution operation)
 - (b) Μη γραμμικότητα (συνάρτηση ReLU)
 - (c) Ο τελεστής «συγκέντρωσης» (pooling - downsampling)

Περιγράφουμε πρώτα τα συστατικά και μετά τα συνδυάζουμε ώστε να πάρουμε ένα CNN.

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

(a) Ο τελεστής συνέλιξης

- Πρόκειται για ένα είδος φιλτραρίσματος, που αναδεικνύει τοπικές σχέσεις μεταξύ εικονοστοιχείων.
- Διαφορετικά φίλτρα αποκαλύπτουν διαφορετικά είδη πληροφοριών.
- Ένα φίλτρο ή πυρήνας (kernel) ή ανιχνευτής χαρακτ. (feature detector), είναι ένας μικρός πίνακας (συνήθως 3x3).
- Παράγει μια νέα εικόνα που ονομάζεται χάρτης χαρακτηριστικού (feature map) (convolved feature)
- Πώς δουλεύει:
 - Ένα φίλτρο πίνακας «ολισθαίνει» πάνω στην υπό μελέτη εικόνα.
 - Σε κάθε θέση, πραγματοποιείται ένας στοιχείο-προς-στοιχείο πολ/σμός ανάμεσα στο φίλτρο πίνακα και το τμήμα της εικόνας με το οποίο αυτό επικαλύπτεται.
 - Τα αποτελέσματα του πολ/σμού αθροίζονται προκειμένου να υπολογιστεί η τιμή του εικονοστοιχείου στο χάρτη χαρακτηριστικού (convolved feature)

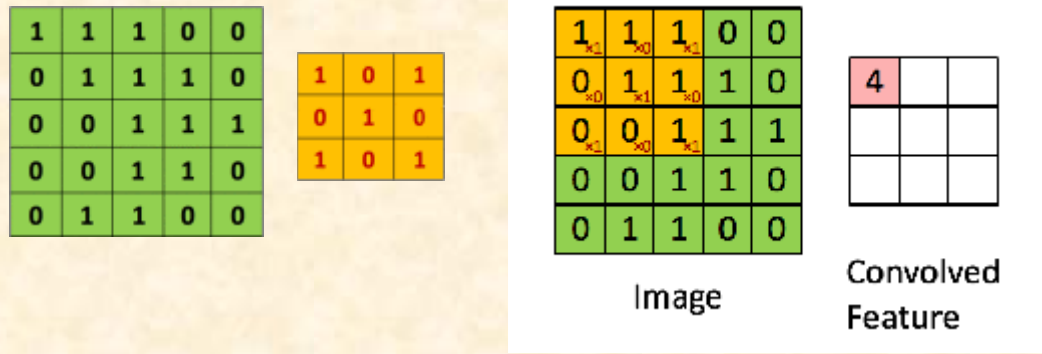
ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

(a) Ο τελεστής συνέλιξης

Παράδειγμα:

Θεωρείστε την **5x5 πράσινη** (δυαδική) **εικόνα** και το παρακάτω **3x3 φίλτρο**. Πιο κάτω επιδεικνύεται γραφικά η **διαδικασία συνέλιξης**.



Διαφορετικά φίλτρα, με την αντίστοιχη **δράση** τους εκτίθενται στα δεξιά.

Operation	Filter	Convolved Image
Identity	$\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	
Edge detection	$\begin{bmatrix} 1 & 0 & -1 \\ 0 & 0 & 0 \\ -1 & 0 & 1 \end{bmatrix}$	
	$\begin{bmatrix} 0 & 1 & 0 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$	
	$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$	
Sharpen	$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{bmatrix}$	
Box blur (normalized)	$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$	
Gaussian blur (approximation)	$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$	

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

(a) Ο τελεστής συνέλιξης

Παράδειγμα: Διαφορετικά φίλτρα δίνουν διαφορετικούς χάρτες χαρακτηριστικών.



0	0	0	0	0	0	0
0	*	*	*	*	*	0
0	*	*	*	*	*	0
0	*	*	*	*	*	0
0	*	*	*	*	*	0
0	*	*	*	*	*	0
0	0	0	0	0	0	0

Stride: Αριθμός εικονοστοιχείων που προσπέρνα ο πίνακας φίλτρου καθώς ολισθαίνει πάνω στην εικόνα (στο παραπάνω παράδειγμα $\text{stride}=1$)

•Όσο μεγαλύτερο είναι το stride , τόσο μικρότεροι είναι οι παραγόμενοι χάρτες χαρακτ.

Γέμισμα με μηδενικά (Zero padding): Το γέμισμα της εικόνας εισόδου με 0's γύρω από τα όριά της, επιτρέπει την εφαρμογή του φίλτρου και στα εικονοστοιχεία που βρίσκονται στα όρια της εικόνας.

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

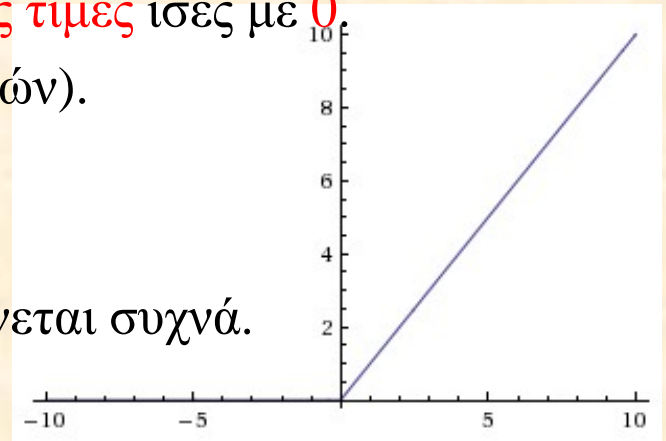
(b) Μη γραμμικότητα (συνάρτηση **ReLU**)

Εφαρμόζεται χωριστά σε κάθε εικονοστοιχείο του **χάρτη χαρακτ.** που παράχθηκε από το βήμα (a).

Στην πραγματικότητα η **ReLU** θέτει όλες τις **αρνητικές τιμές** ίσες με **0** (**επιδιορθώνει** (“**rectifies**”) τον πίνακα χαρακτηριστικών).

Μπορούν επίσης να χρησιμοποιηθούν και άλλοι τύποι μη γραμμικότητας (π.χ. σιγμοειδείς) παρότι αυτό δε γίνεται συχνά.

Παράδειγμα: Η επίδραση της ReLU



Input Feature Map



ReLU



Rectified Feature Map



$f(\cdot)$: Rectified
Linear Unit
(ReLU)

$$f(t) = t(0), \text{ if } t \geq (<) 0$$

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

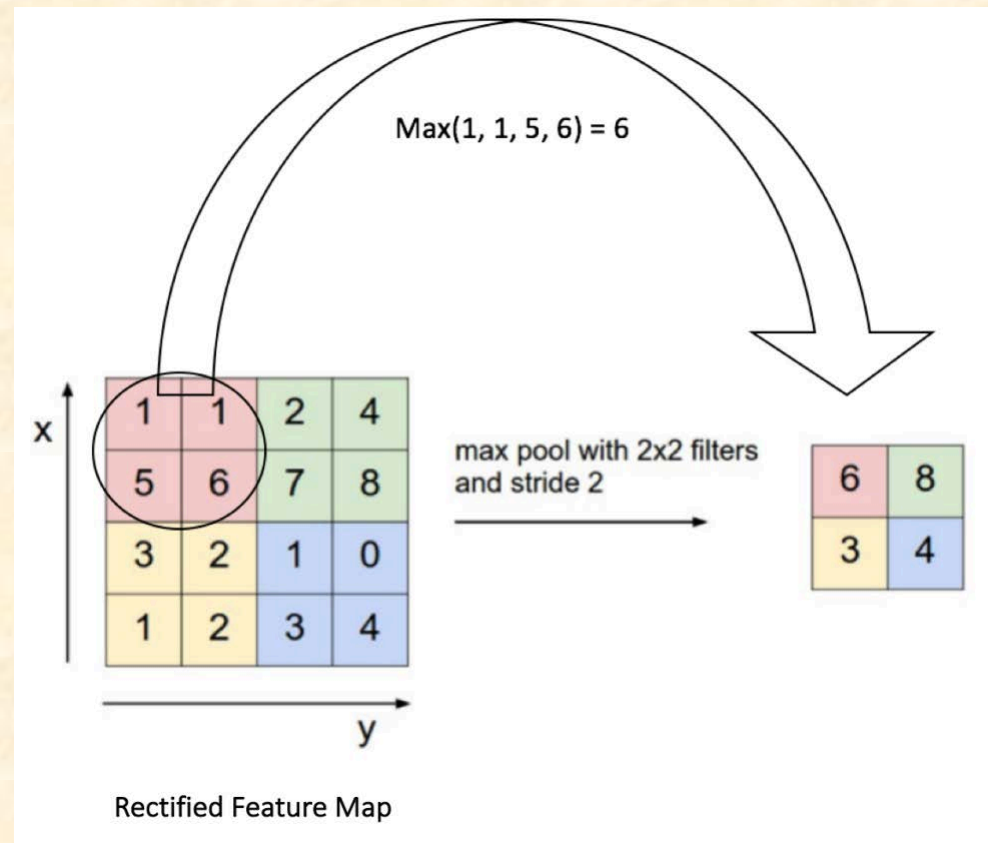
(c) Η λειτουργία **συγκέντρωσης** (pooling - downsampling)

Εφαρμόζεται πάνω στους **επιδιορθωμένους χάρτες χαρακτηριστικών**.

Μειώνει τη **διάστασή** τους **διατηρώντας** την πιο **σημαντική πληροφορία**.

Οι **τελεστές** που εφαρμόζονται μπορεί να είναι **άθροισμα** (sum), **μέγιστο** (max), **μέσος όρος** (average), etc.

Η λειτουργία αυτή φαίνεται μέσω του ακόλουθου παραδείγματος.



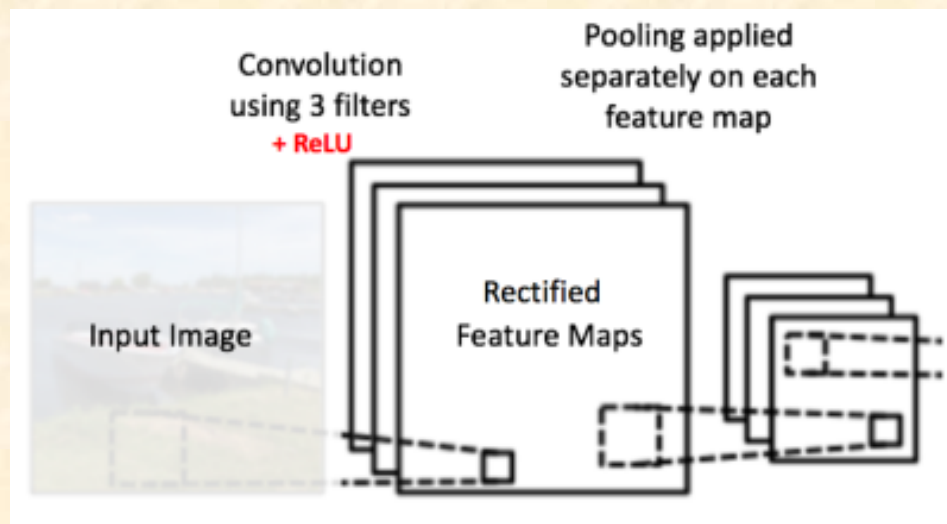
ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

Στη συνέχεια συζητάμε πώς συνδυάζονται τα παραπάνω χαρακτηριστικά

Το **πρώτο στάδιο επεξεργασίας** σε ένα **CNN**

- (a) εφαρμόζει έναν αριθμό **διαφορετικών φίλτρων** στην εικόνα εισόδου και παράγει τους αντίστοιχους χάρτες χαρακτηριστικών.
- (b) εφαρμόζει τη συνάρτηση **ReLU** στους παραγόμενους **πίνακες χαρακτηριστικών** (τις “επιδιορθώνει”)
- (c) εφαρμόζει τη λειτουργία **συγκέντρωσης** σε καθένα από τους **επιδιορθωμένους χάρτες χαρακτηριστικών**, χωριστά.



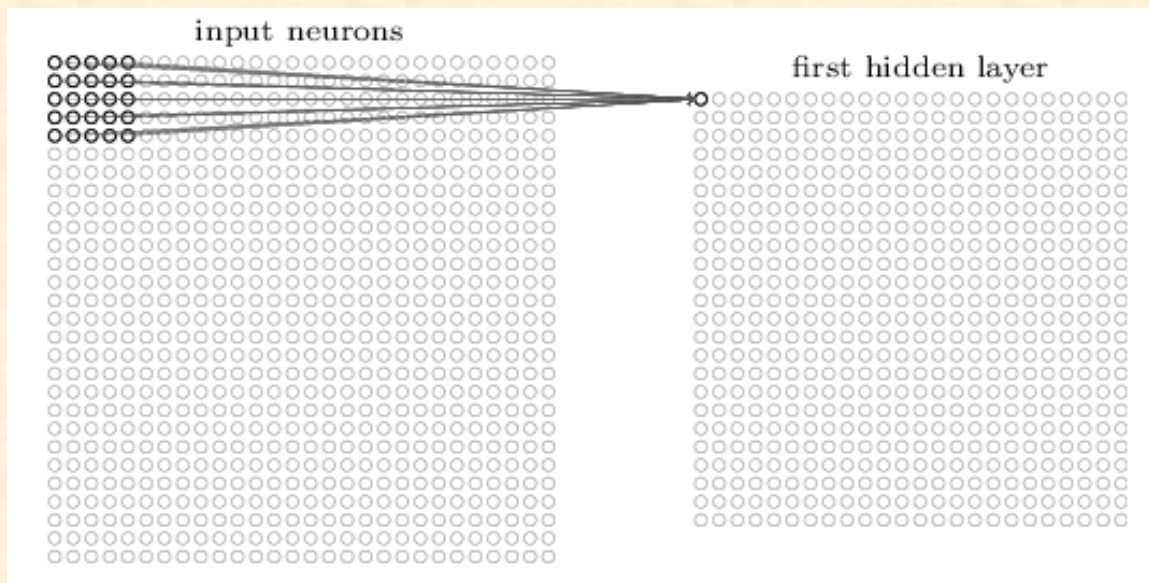
ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

Στη συνέχεια συζητάμε πώς συνδυάζονται τα παραπάνω χαρακτηριστικά

Το **πρώτο στάδιο επεξεργασίας** σε ένα CNN:

- (a) εφαρμόζει έναν αριθμό **διαφορετικών φίλτρων** στην εικόνα εισόδου και παράγει τους αντίστοιχους χάρτες χαρακτηριστικών.
- (b) εφαρμόζει τη συνάρτηση **ReLU** στους παραγόμενους **πίνακες χαρακτηριστικών** (τις “επιδιορθώνει”)
- (c) εφαρμόζει τη λειτουργία **συγκέντρωσης** σε καθένα από τους **επιδιορθωμένους χάρτες χαρακτηριστικών**, χωριστά.

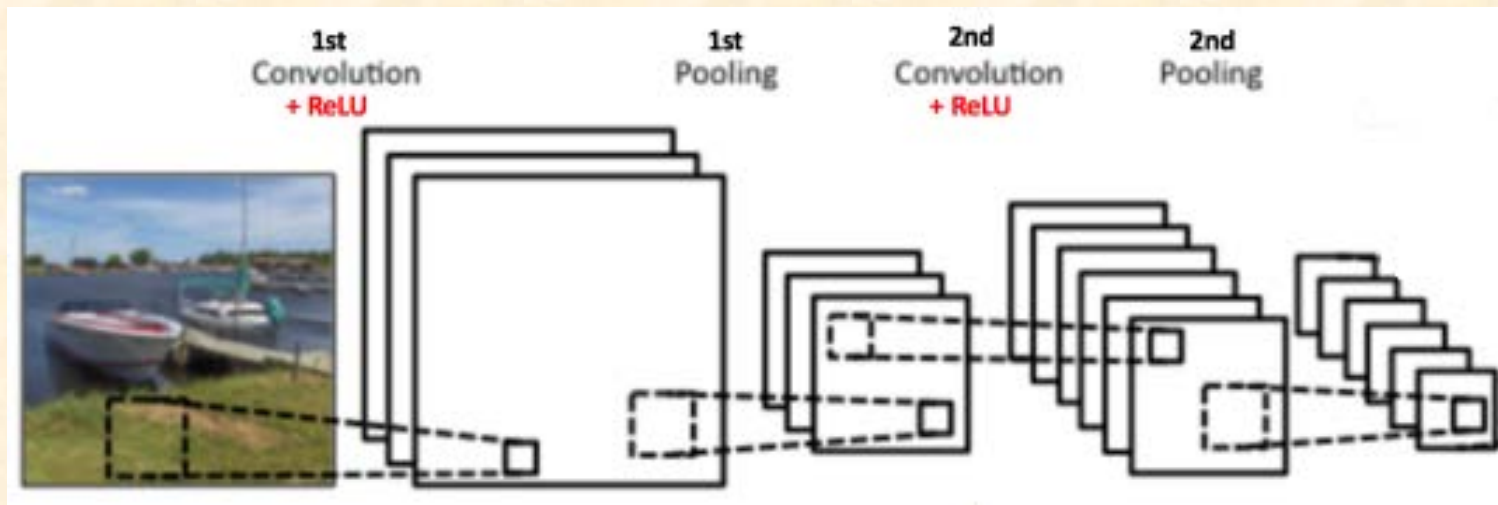


ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

Κατά το **δεύτερο στάδιο επεξεργασίας** σ' ένα **CNN** εφαρμόζεται ακριβώς η **ίδια επεξεργασία** πάνω στα **αποτελέσματα** του **πρώτου** σταδίου.

Η μόνη **διαφορά** είναι ότι η **συνέλιξη** εφαρμόζεται τώρα πάνω σε **όλους** τους **(μειωμένου μεγέθους) χάρτες χαρακτηριστικών** που παρήχθησαν από το 1^ο στάδιο επεξεργασίας.

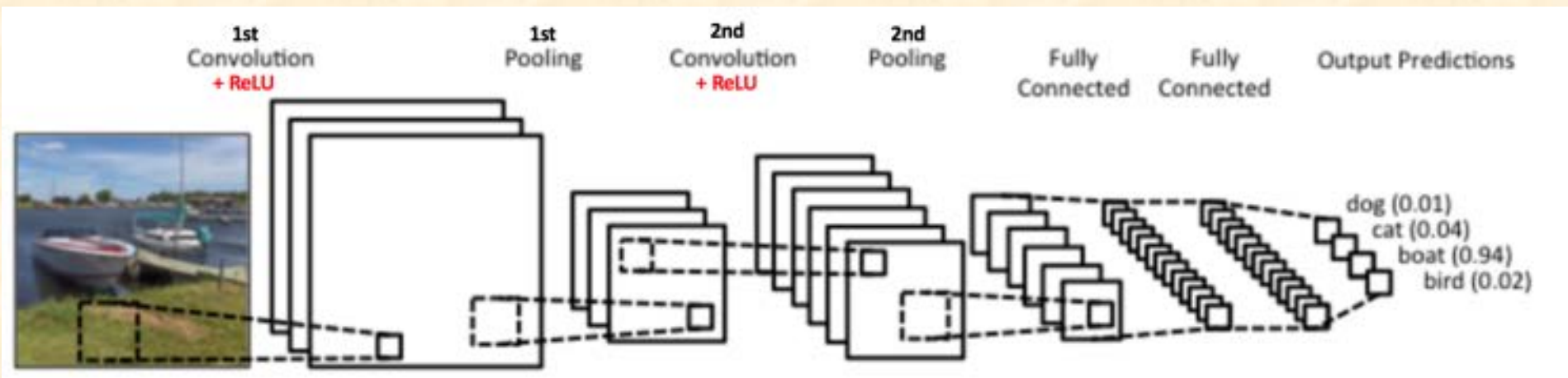


Μπορούν να εισαχθούν και **επιπλέον στάδια επεξεργασίας**, καθένα από τα οποία εφαρμόζεται στα αποτελέσματα του προηγούμενου.

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

Το **τελικό στάδιο επεξεργασίας** ενός **CNN** πραγματοποιείται από ένα δίκτυο 2 ή 3 επιπέδων, όπου κάθε νευρώνας εξόδου αντιστοιχεί σε μια κλάση.



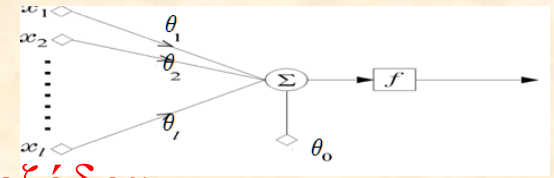
Παρατηρήσεις:

- Στην πράξη, οι **συντελεστές του φίλτρου** δεν προεπιλέγονται, αλλά **εκτιμούνται** κατά τη φάση της **εκπαίδευσης**.
- Η **εκπαίδευση ακολουθεί** τη **στρατηγική οπίσθιας διάδοσης (Back propagation)**, λαμβάνοντας υπόψιν τα **κοινά βάρη** στα συνελκτικά επίπεδα.
- Οι **νευρώνες εξόδου** του **δικτύου χρησιμοποιούν** την **softmax** ως **συνάρτηση εξόδου**. Αυτό, **εγγυάται** ότι τα **αποτελέσματα** θα βρίσκονται στο διάστημα **(0,1)**.

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

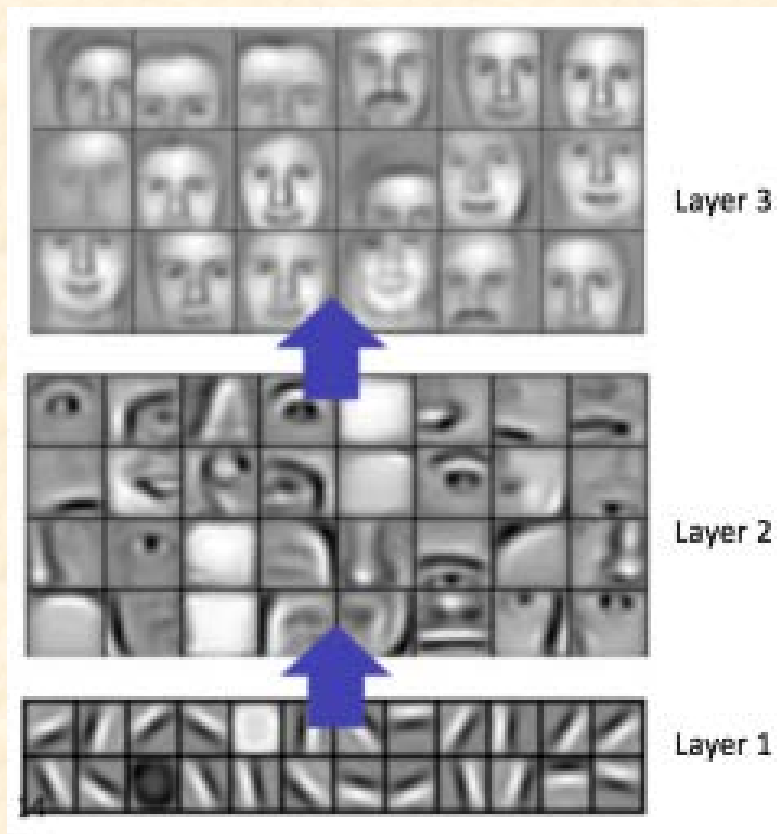
Συνάρτηση softmax:



Έστω $\mathbf{s}=[s_1, s_2, \dots, s_M]^T$ οι **έξοδοι** της **ReLU** στο **δίκτυο εξόδου**.

Το **τελικό αποτέλεσμα** των νευρώνων του επιπέδου αυτού θα είναι

$$\hat{\mathbf{y}} = [\hat{y}_1, \hat{y}_2, \dots, \hat{y}_M]^T = [e^{s_1} / \sum_{j=1}^M e^{s_j}, \dots, e^{s_M} / \sum_{j=1}^M e^{s_j}]^T$$

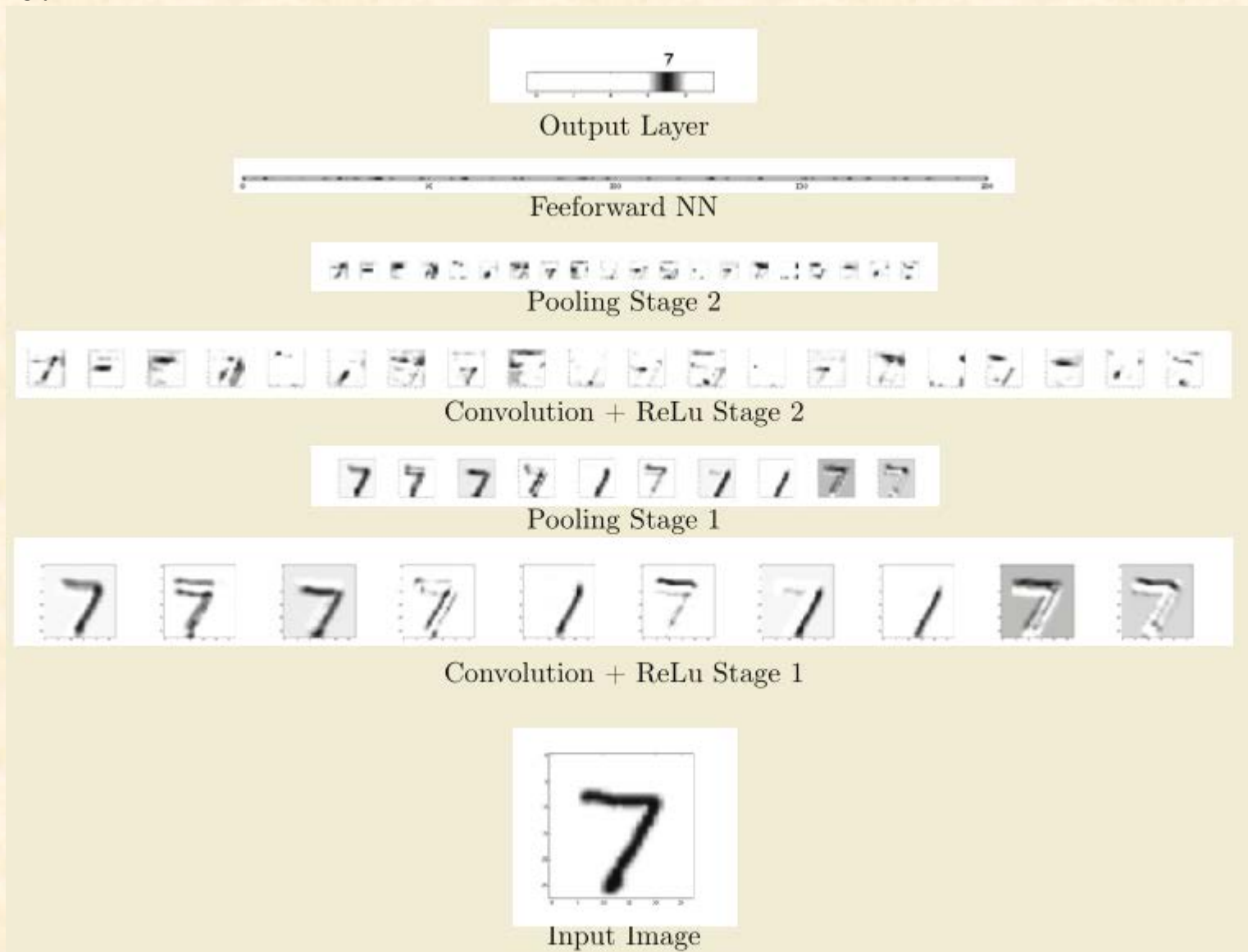


Θυμηθείτε ότι **κάθε επίπεδο** ανιχνεύει **χαρακτηριστικά** **υψηλότερου επιπέδου** από **το προηγούμενό του**.

ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Convolutional Neural Networks (CNNs)

Example:

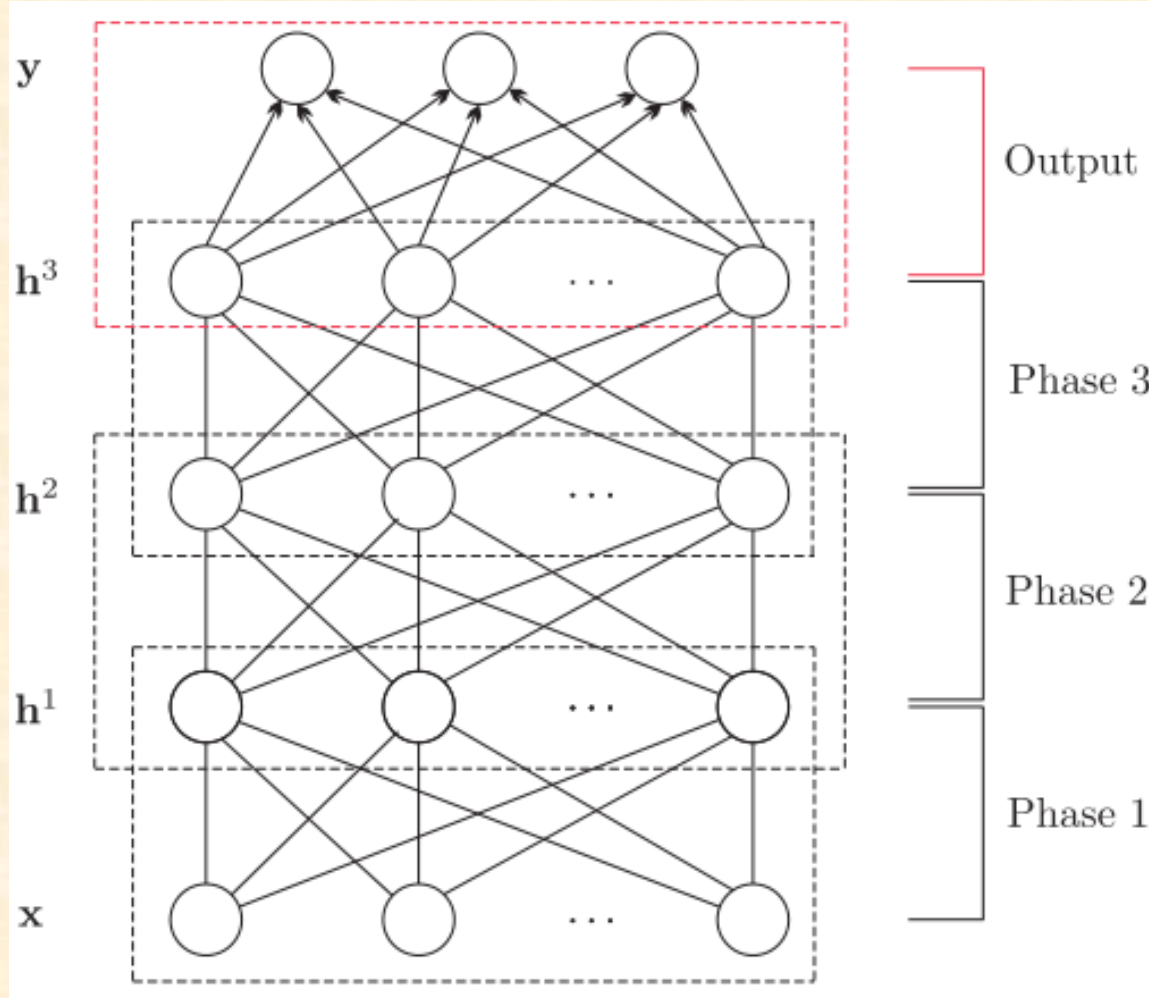


ΜΗ ΓΡΑΜΜΙΚΟΙ ΤΑΞΙΝΟΜΗΤΕΣ – ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

Εκπ/ση δικτύων «βαθιάς»
αρχιτεκτονικής

Γενική στρατηγική

- Προ-εκπαίδευσε τα **βάρη** που σχετίζονται με **κάθε κρυφό επίπεδο** ακολουθιακά με **χρήση μη επιβλεπόμενων** τεχνικών, ξεκινώντας από το πρώτο.
- Προ-εκπαίδευσε τα **βάρη** που σχετίζονται με **επίπεδο εξόδου** με **χρήση επιβλεπόμενων** τεχνικών.
- Χρησιμοποίησε τις παραπάνω τιμές των βαρών ως αρχικές τιμές και εφάρμοσε τον αλγόριθμο **BP algorithm** ώστε να προκύψουν οι **τελικές εκτιμήσεις** όλων των **βαρών** του δικτύου.



Δημοφιλής επιλογή για προ-εκπαίδευση:
Restricted Boltzmann Machines (RBMs).