

# ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

## Νευρωνικά Δίκτυα και Ευφυή Υπολογιστικά Συστήματα 2011-2012

### 1<sup>η</sup> Εργαστηριακή Άσκηση: «Μελέτη των πολυεπίπεδων Perceptrons και εφαρμογή σε προβλήματα ταξινόμησης εικόνας»

## 1 Εισαγωγή

Τα πολυεπίπεδα δίκτυα εμπρόσθιας τροφοδότησης (Multilayer Perceptrons) αποτελούν μια θεμελιώδη κατηγορία νευρωνικών δικτύων. Ένα τέτοιο δίκτυο αποτελείται από ένα σύνολο αισθητήρων που αποτελούν το επίπεδο εισόδου, ένα ή περισσότερα κρυμμένα επίπεδα στα οποία γίνονται οι διάφοροι υπολογισμοί και ένα επίπεδο εξόδου. Το πρότυπο που παρουσιάζεται στο επίπεδο εισόδου διαδίδεται κατά την εμπρόσθια διεύθυνση, επίπεδο προς επίπεδο. Ένας διαδεδομένος αλγόριθμος που εφαρμόζεται για την εκπαίδευση πολυεπίπεδων δικτύων εμπρόσθιας τροφοδότησης είναι ο αλγόριθμος οπίσθιας διάδοσης (Back Propagation).

Τα χαρακτηριστικά του πολυεπίπεδου perceptron είναι τα εξής:

- Κάθε νευρώνας του δικτύου περιλαμβάνει μία μη γραμμική (διαφορίσιμη) συνάρτηση ενεργοποίησης.
- Το δίκτυο αποτελείται από ένα ή περισσότερα κρυμμένα επίπεδα νευρώνων, τα οποία δεν αποτελούν είσοδο ή έξοδο και του παρέχουν τη δυνατότητα μάθησης.
- Το δίκτυο έχει μεγάλο βαθμό συνεκτικότητας, που καθορίζεται από τις συνάψεις του.

Ο συνδυασμός των παραπάνω χαρακτηριστικών, μαζί με την ικανότητα της μάθησης μέσω της εκπαίδευσης προσδίδει στο δίκτυο μεγάλη υπολογιστική ισχύ. Στη σχεδίαση ενός νευρωνικού δικτύου υπεισέρχονται πολλά θέματα αρχιτεκτονικής στα οποία δεν υπάρχει συγκεκριμένη λύση. Αντίθετα ο χρήστης πρέπει να τα αντιμετωπίσει στηριζόμενος στην προσωπική του εμπειρία.

Ο σκοπός της άσκησης αυτής είναι η κατασκευή και η μελέτη ενός Multilayer Perceptron (MLP), που θα εκπαιδευτεί με τον κλασικό αλγόριθμο Back Propagation για την επίλυση ενός προβλήματος ταξινόμησης τμημάτων εικόνων, με βάση κάποια χαρακτηριστικά χαμηλού επιπέδου.

## 2 Μελέτη ενός Νευρωνικού Δικτύου για την ταξινόμηση τμημάτων εικόνων

Ένα από τα πιο δημοφιλή προβλήματα στο χώρο της σημασιολογικής ανάλυσης πολυμεσικού υλικού αποτελεί η ταξινόμηση τμημάτων εικόνων με βάση το

περιεχόμενο. Μια συνηθισμένη διαδικασία για την επίτευξη αυτού του σκοπού έχει ως εξής. Αρχικά ένας αλγόριθμος τμηματοποίησης επεξεργάζεται την εικόνα και τη χωρίζει σε περιοχές με παρόμοια χαρακτηριστικά χαμηλού επιπέδου. Τα τμήματα αυτά ταξινομούνται χειροκίνητα σύμφωνα με το περιεχόμενο τους σε κάποιες κατηγορίες. Με αυτό το τρόπο δημιουργείται ένα σύνολο εκπαίδευσης (χαρακτηριστικά και επιθυμητές αποκρίσεις) το οποίο χρησιμοποιείται από ένα νευρωνικό δίκτυο για την ταξινόμηση των τμημάτων της εικόνας.

Οι πίνακες που θα χρησιμοποιηθούν για την εκπαίδευση του δικτύου σας έχουν τα εξής περιεχόμενα:

- Κάθε στήλη του πίνακα TrainData αντιστοιχεί στο τμήμα μίας εικόνας ενώ κάθε γραμμή περιέχει τα χαρακτηριστικά χαμηλού επιπέδου (που αντιστοιχούν σε MPEG-7 Descriptors) του τμήματος αυτού.
- Κάθε στήλη του πίνακα TrainDataTargets αντιστοιχεί στο τμήμα μίας εικόνας ενώ κάθε γραμμή αντιστοιχεί σε μία από τις κατηγορίες «Άνθρωπος», «Άμμος», «Κτήριο», «Πεζοδρόμιο», «Θάλασσα», «Ουρανός», «Κύμα», «Φυτό», «Γρασίδι», «Δέντρο», «Κορμός», και «Χώμα». Στην περίπτωση που το τμήμα μιας εικόνας ανήκει σε μία από τις κατηγορίες τότε η αντίστοιχη γραμμή θα έχει την τιμή 1 ενώ οι υπόλοιπες γραμμές θα πάρουν την τιμή 0.

Τα δεδομένα των πινάκων TrainData, TrainDataTargets θα χρησιμοποιηθούν για την εκπαίδευση του νευρωνικού δικτύου ενώ τα δεδομένα των πινάκων TestData, TestDataTargets θα χρησιμοποιηθούν ως μέτρο των επιδόσεων του νευρωνικού δικτύου. Προφανώς τα δεδομένα των πινάκων TestData και TestDataTargets έχουν αντίστοιχη δομή με τα δεδομένα των πινάκων TrainData και TrainDataTargets.

## 2.1 Προεπεξεργασία των δεδομένων

**Προσοχή!!!** Να αποθηκεύσετε τον κώδικα της άσκησης είτε σε ένα είτε σε περισσότερα m-files.

**Βήμα 1** Να φορτώσετε τα δεδομένα στο workspace του MATLAB (με την εντολή load dataset) και να τα τροποποιήσετε κατάλληλα ώστε το σύνολο που θα χρησιμοποιήσετε για εκπαίδευση να περιέχει περίπου τον ίδιο αριθμό τμημάτων εικόνων από κάθε κατηγορία.

Υπόδειξη: Χρησιμοποιείτε την εντολή sum στον πίνακα TrainDataTargets για να βρείτε πόσα τμήματα εικόνας περιέχει κάθε κατηγορία. Με την εντολή bar τυπώστε τον αριθμό των τμημάτων εικόνας που ανήκουν σε κάθε κατηγορία.

Υπόδειξη: Για να κρατήσετε περίπου τον ίδιο αριθμό από κάθε κατηγορία μπορείτε να φτιάξετε έναν πίνακα index ο οποίος θα περιέχει τις στήλες που επιθυμείτε να κρατήσετε ή να διαγράψετε. Στη συνέχεια εργαστείτε όπως περιγράφεται στην παράγραφο 3 του τυπολογίου.

**Βήμα 2** Προεπεξεργαστείτε κατάλληλα τα δεδομένα ώστε να διευκολύνετε την εκπαίδευση του δικτύου σας.

Υπόδειξη: Για την προεπεξεργασία των δεδομένων, να χρησιμοποιήσετε τις συναρτήσεις removeconstantrows, processpca.

Υπόδειξη: Προσοχή κάθε μετασχηματισμός στα δεδομένα TrainData θα πρέπει να συνοδεύεται από τον αντίστοιχο μετασχηματισμό στα δεδομένα TestData (αντίστοιχα κάθε μετασχηματισμός στα δεδομένα TrainDataTargets θα πρέπει να συνοδεύεται από τον αντίστοιχο μετασχηματισμό στα δεδομένα TestDataTargets).

## 2.2 Μελέτη της Αρχιτεκτονικής του Νευρωνικού Δικτύου

- Βήμα 3 Να επιλέξετε κατάλληλη συνάρτηση ενεργοποίησης με βάση τον τρόπο με τον οποίο κανονικοποιήσατε τα δεδομένα (θεωρείστε ότι τα δεδομένα είναι κανονικοποιημένα έτσι ώστε το ελάχιστο να έχει την τιμή -1 και το μέγιστο την τιμή 1).
- Βήμα 4 Να κατασκευάσετε ένα Νευρωνικό Δίκτυο MLP. Για την δημιουργία του νευρωνικού δικτύου χρησιμοποιείτε την συνάρτηση `newff`. Αρχικά να χρησιμοποιήσετε εκπαίδευση με Early Stopping χρησιμοποιώντας το 80% των δεδομένων για εκπαίδευση και το 20% των δεδομένων επαλήθευση (`trainRatio=0.8`, `valRatio=0.2`, `testRatio=0`).
- Βήμα 5 Να μελετήσετε την επίδραση του αριθμού των νευρώνων στην απόδοση του νευρωνικού δικτύου. Να ξεκινήσετε με 1 κρυμμένο επίπεδο και 5 νευρώνες και στη συνέχεια αυξήστε τον αριθμό τους (όχι πάνω από 30), μέχρι να πετύχετε το καλύτερο δυνατό αποτέλεσμα. Να πειραματιστείτε με την προσθήκη δεύτερου κρυμμένου επιπέδου.
- Βήμα 6 Να πειραματιστείτε με διαφορετικές συναρτήσεις εκπαίδευσης της MATLAB: `traindx`, `trainlm`, `trainrp`, `trainbfg`, `traincgb`.
- Βήμα 7 Χρησιμοποιώντας τον καταλληλότερη συνάρτηση εκπαίδευσης και αριθμό νευρώνων ανά επίπεδο να εξετάσετε:
- Τις επιδόσεις του νευρωνικού δικτύου όταν χρησιμοποιείται ο αλγόριθμος μάθησης απότομης καθόδου (Gradient descent, `learngd`) και ο αλγόριθμος μάθησης απότομης καθόδου με προσθήκη του όρου ορμής (Gradient descent with momentum, `learngdm`).
  - Τις επιδόσεις του νευρωνικού δικτύου όταν χρησιμοποιείται και όταν δεν χρησιμοποιείται σύνολο επαλήθευσης (validation set).
  - Στην περίπτωση εκπαίδευσης του νευρωνικού δικτύου χωρίς validation set να εξετάσετε την ικανότητα του δικτύου για γενίκευση τροποποιώντας τον αριθμό των εποχών του (επικεντρωθείτε σε αριθμό εποχών ο οποίος να είναι κοντά στις εποχές που σταμάτησε η εκπαίδευση του νευρωνικού δικτύου όταν είχατε validation set).
- Βήμα 8 Με βάση τις παρατηρήσεις σας στα παραπάνω ερωτήματα, να επιλέξετε τις κατάλληλες παραμέτρους για να κατασκευάσετε ένα δίκτυο που να συνδυάζει απλότητα με αποτελεσματικότητα και μεγάλη ικανότητα γενίκευσης.

## 2.3 Αξιολόγηση της απόδοσης του νευρωνικού

Να μελετήσετε το πώς η αρχιτεκτονική του δικτύου επηρεάζει την επίδοση του ταξινομητή ανά κατηγορία. Για την επίδοση του ταξινομητή χρησιμοποιήστε τα ακόλουθα μέτρα στο σύνολο εκπαίδευσης και στο σύνολο ελέγχου:

$$Accuracy = \frac{\text{Σωστά ταξινομημένα τμήματα}}{\text{Συνολικός αριθμός τμημάτων}}$$

$$Precision = \frac{\text{Σωστά ταξινομημένα τμήματα στην κατηγορία}}{\text{Ταξινομημένα τμήματα στην κατηγορία}}$$

$$Recall = \frac{\text{Σωστά ταξινομημένα τμήματα στην κατηγορία}}{\text{Σύνολο τμημάτων στην κατηγορία}}$$

Αν θέλετε να μετατρέψετε την εκτίμηση του νευρωνικού σας σε ένα πίνακα με τιμές 0 και 1 και να τον συγκρίνετε με τις επιθυμητές εξόδους χρησιμοποιήστε τα χαρακτηριστικά της συνάρτησης `max`. Για μεγαλύτερη ευκολία σας δίνουμε τη συνάρτηση `eval_Accuracy_Precision_Recall` η οποία έχει σαν ορίσματα την έξοδο του νευρωνικού δικτύου και την επιθυμητή έξοδο και επιστρέφει τις τιμές `accuracy`, `precision`, `recall`. Την συνάρτηση αυτή θα την τρέξετε ως εξής:

[accuracy precision recall]=

`eval_Accuracy_Precision_Recall(nnOutput,desiredOutput);`

Όπου το `nnOutput` είναι η έξοδος του νευρωνικού δικτύου για το `TrainData` (`TestData`) και το `desiredOutput` είναι ο πίνακας `TrainDataTargets` (`TestDataTargets`).

### 3 Ερωτήματα

- Ερώτημα 1 Τι αποτέλεσμα θα είχε στο δίκτυο μας η χρήση ενός συνόλου δεδομένων που θα περιείχε πολλά παραπάνω τμήματα από κάποιες κατηγορίες σε σύγκριση με τις υπόλοιπες;
- Ερώτημα 2 Θα ήταν δυνατή η εκπαίδευση του δικτύου χωρίς την προεπεξεργασία των δεδομένων και γιατί;
- Ερώτημα 3 Να εξηγήσετε την επιλογή της συνάρτησης ενεργοποίησης που επιλέξατε.
- Ερώτημα 4 Για το Βήμα 5 να συγκρίνετε τα αποτελέσματα για κάθε νευρωνικό και να προσπαθήσετε να αιτιολογήσετε τις διαφορές που τυχόν παρατηρήσατε.
- Ερώτημα 5 Τι παρατηρείται όταν χρησιμοποιείται ο αλγόριθμος μάθησης απότομης καθόδου (`Gradient descent`, `learngd`) και τι όταν χρησιμοποιείται ο αλγόριθμος μάθησης απότομης καθόδου με προσθήκη του όρου ορμής (`Gradient descent with momentum`, `learngdm`).
- Ερώτημα 6 Να εξηγήστε τι επιτυγχάνει η μέθοδος “Early Stopping” με τη βοήθεια κατάλληλου διαγράμματος.
- Ερώτημα 7 Τι παρατηρείτε όταν ο αριθμός των εποχών είναι πολύ μικρός και τι όταν είναι πολύ μεγάλος;
- Ερώτημα 8 Τι παρατηρείτε ως προς την απόδοση του ταξινομητή σε κάποιες κατηγορίες; Που πιστεύεται ότι οφείλεται αυτό και τι θα μπορούσατε να κάνετε ώστε να έχετε βέλτιστη απόδοση σε όλες τις κατηγορίες.