

# ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

## Τμήμα Μηχανικών Παραγωγής και Διοίκησης

Εργαστήριο Σχεδιασμού & Ανάπτυξης Συστημάτων Υποστήριξης Αποφάσεων  
(ΕΡΓΑ.Σ.Υ.Α)

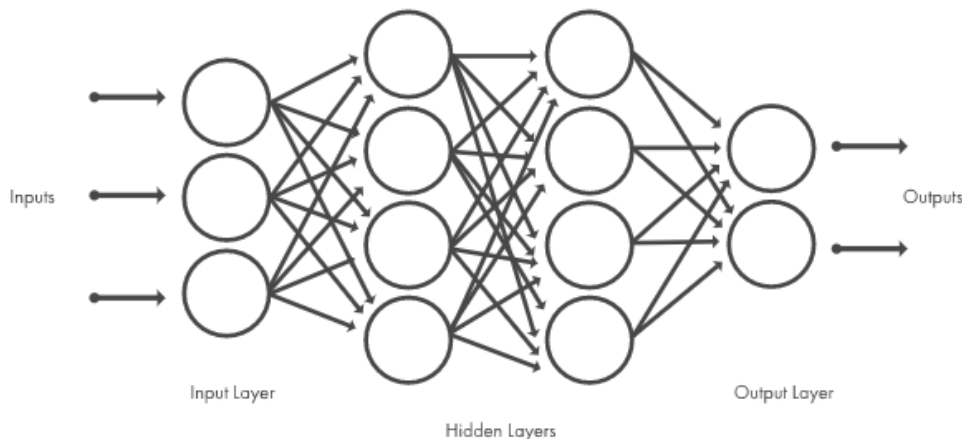
### Εισαγωγή στην Τεχνητή Νοημοσύνη (ΜΠΔ306)

Χειμερινό εξάμηνο 2020-2021

Εργαστηριακός βοηθός: Γιαννακάκη Κατερίνα

### ΕΡΓΑΣΙΑ 3<sup>η</sup>

Στόχος αυτής της εργασίας είναι η εξοικείωση με τη λειτουργία των Τεχνητών Νευρωνικών Δικτύων (ΤΝΔ) επιβλεπόμενης μάθησης (supervised learning). Η τυπική αρχιτεκτονική ενός ΤΝΔ επιβλεπόμενης μάθησης φαίνεται στο Σχήμα 1.



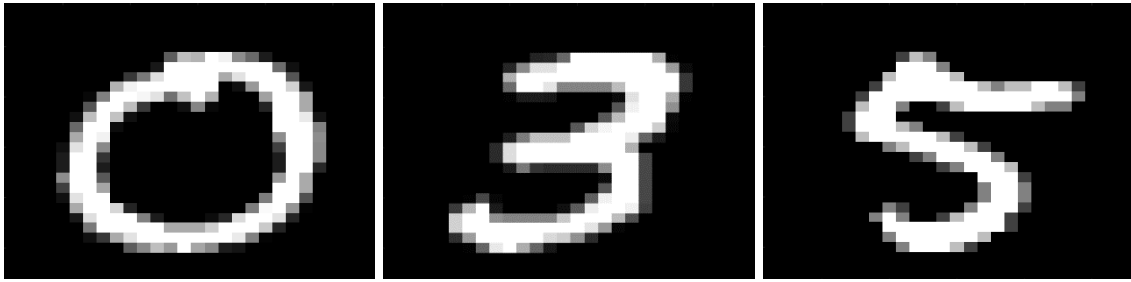
Σχήμα 1: Αρχιτεκτονική ενός νευρωνικού δικτύου επιβλεπόμενης μάθησης

Ένα ΤΝΔ συνδυάζει διάφορα επίπεδα επεξεργασίας (layers) τα οποία επεξεργάζονται τις εισόδους (inputs) οδηγώντας σε συγκεκριμένες εξόδους (outputs). Αποτελείται από ένα επίπεδο εισόδου (input layer), ένα ή περισσότερα κρυφά ενδιάμεσα επίπεδα νευρώνων (hidden layers) και ένα επίπεδο εξόδου (output layer). Τα επίπεδα διασυνδέονται μέσω νευρώνων, με κάθε επίπεδο να χρησιμοποιεί την έξοδο των νευρώνων του προηγούμενου επιπέδου ως είσοδο.

Η εργασία σχετίζεται με την αυτόματη αναγνώριση χειρόγραφων αριθμητικών ψηφίων (0 έως 9) από εικόνες<sup>1</sup> διαστάσεων 28x28 pixels, μέσω της κατηγοριοποίησής τους σε μια από

<sup>1</sup> Τα δεδομένα προέρχονται από τη βάση δεδομένων "[The MNIST database of handwritten digits](https://www.nist.gov/special-interests/multimedia/mnist/)"  
LeCun, Yann, et al. "Gradient-based learning applied to document recognition." *Proceedings of the IEEE* 86.11 (1998): 2278-2324.

10 κλάσεις. Κάθε κλάση αντιστοιχεί σε ένα ψηφίο: 0, 1, 2,..., 9. Χαρακτηριστικές εικόνες από τα δεδομένα φαίνονται στο παρακάτω σχήμα.



Σχήμα 2: Χαρακτηριστικές εικόνες από τις εικόνες ψηφίων που μελετώνται

Το ΤΝΔ που θα δημιουργηθεί θα δέχεται ως είσοδο τα pixels των εικόνων διαστάσεων 28x28, ανεπτυγμένων σε διάνυσμα μεγέθους 784 (28x28=784 pixels), και θα δίνει στην έξοδο την πρόβλεψή του για την κάθε είσοδο. Οι 10 δυνατές κλάσεις, που αντιστοιχούν στα 10 ψηφία, είναι κωδικοποιημένες ως εξής (One-Hot Encoding):

Ψηφίο	Διάνυσμα κλάσης
0	1 0 0 0 0 0 0 0 0 0
1	0 1 0 0 0 0 0 0 0 0
2	0 0 1 0 0 0 0 0 0 0
3	0 0 0 1 0 0 0 0 0 0
4	0 0 0 0 1 0 0 0 0 0
5	0 0 0 0 0 1 0 0 0 0
6	0 0 0 0 0 0 1 0 0 0
7	0 0 0 0 0 0 0 1 0 0
8	0 0 0 0 0 0 0 0 1 0
9	0 0 0 0 0 0 0 0 0 1

Μπορείτε να δείτε μια εικόνα αν μετατρέψετε ένα διάνυσμα των δεδομένων σε πίνακα 28x28, χρησιμοποιώντας τις συναρτήσεις `transpose()`, `reshape()` και `imagesc()`, όπως παρακάτω:

```
imagesc(transpose(reshape(x, [28 28]))); colormap(gray);
```

όπου `x` ένα παράδειγμα από τον πίνακα δεδομένων.

Για την υλοποίηση της άσκησης χρησιμοποιήστε το *Neural Network Pattern Recognition wizard* του MATLAB (`nprtool`) ώστε να εξαγάγετε τον αρχικό κώδικα, και επί αυτού να κάνετε τις **απαραίτητες τροποποιήσεις** (θα χρειαστεί να αλλάξετε, να προσθέσετε και να αφαιρέσετε κώδικα) για να απαντήσετε στα παρακάτω ερωτήματα.

Ακολουθήστε τα βήματα και συντάξτε μια αναφορά στην οποία θα περιγράφετε συνοπτικά και ουσιαστικά τις ενέργειες που πραγματοποιήσατε και θα απαντάτε στις ερωτήσεις που θα συναντήσετε παρακάτω. Επίσης, συμπληρώστε το αρχείο “resultsXXXXXXXXXX.xlsx” όπου ζητείται (**αφού αντικαταστήσετε τα X με τον αριθμό μητρώου σας**).

**ΠΡΟΣΟΧΗ** αν χρειαστεί να κάνετε εκχώρηση τιμών σε μεταβλητές που έχετε ήδη χρησιμοποιήσει και περιέχουν ήδη κάποιες τιμές. Ίσως να είναι καλύτερο να τις “αδειάσετε” πρώτα, π.χ. `x = []` ;

1. Χρησιμοποιώντας τη συνάρτηση `load` να φορτώσετε στο περιβάλλον εργασίας του MATLAB το αρχείο `.mat` που σας δίνεται (`MNISTdataset.mat`), το οποίο περιέχει τα δεδομένα που θα χρησιμοποιηθούν για την εκπαίδευση και για την αξιολόγηση του δικτύου. Το `MNISTdataset.mat` περιέχει δυο μεταβλητές, την `dataX` που περιέχει τις μεταβλητές του προβλήματος και την `dataY` που περιέχει τις μεταβλητές κλάσης (εξόδους-στόχους) του προβλήματος. **Πόσους νευρώνες εισόδου και πόσους νευρώνες εξόδου θα έχει το ΤΝΔ;**
2. Ανοίξτε το Neural Network Pattern Recognition wizard του MATLAB και φορτώστε τα δεδομένα εκπαίδευσης.
3. Με το Neural Network Pattern Recognition wizard δημιουργήστε ένα ΤΝΔ με 1 κρυφό επίπεδο και 10 νευρώνες στο κρυφό επίπεδο. Προχωρήστε μέχρι το τέλος του οδηγού και εξάγετε τον κώδικα στο κυρίως παράθυρο του MATLAB. **Τροποποιήστε** αυτόν τον κώδικα όπου χρειάζεται για να εφαρμόσετε και να απαντήσετε τα επόμενα ερωτήματα.
4. Επιλέξτε τις παρακάτω παραμέτρους για το ΤΝΔ:
  - Συνάρτηση εκπαίδευσης: standard Gradient Descent backpropagation (`traingd`)
  - Μέθοδος υπολογισμού σφάλματος δικτύου: Mean squared error
  - Ρυθμός μάθησης (learning rate): 0.25
  - Αποδεκτή απόκλιση εξόδου του δικτύου από την έξοδο-στόχο:  $10^{-8}$
  - Αποδεκτός αριθμός διαδοχικών epochs που δεν παρατηρείται βελτίωση κατά το validation: 10
  - Ποσοστό δεδομένων που θα αποτελέσουν το training set: 60%
  - Ποσοστό δεδομένων που θα αποτελέσουν το validation set: 10%
  - Συνάρτηση ενεργοποίησης νευρώνων κρυφού επιπέδου: Tan-Sigmoid (`tansig`)
  - Συνάρτηση ενεργοποίησης νευρώνων επιπέδου εξόδου: Log-Sigmoid (`logsig`)

#### *Μελέτη της επίδρασης του αριθμού κύκλων εκπαίδευσης*

5. Εκπαιδεύστε το δίκτυο (χρησιμοποιώντας τη συνάρτηση `train()` για μέγιστο αριθμό κύκλων εκπαίδευσης (epochs): 10, 50, 100, 200, 400, 600, 800, 1000. Για κάθε αριθμό κύκλων εκπαίδευσης να εκπαιδεύσετε το δίκτυο 5 φορές με τα δεδομένα εκπαίδευσης (training set). Μετά να δοκιμάσετε/αξιολογήσετε καθένα από τα 5 μοντέλα που προέκυψαν από την εκπαίδευση χρησιμοποιώντας τα δεδομένα του test set και τις συναρτήσεις `net()`, `vec2ind()`, `sum()`, `numel()` για να υπολογίσετε το ποσοστό των εξόδων που διαφέρουν από τις εξόδους-στόχους (`percentErrors`). Οι δείκτες των παραδειγμάτων που αποτέλεσαν το training set δίνονται από το `tr.trainInd` και οι δείκτες των παραδειγμάτων που αποτέλεσαν το test set δίνονται από το `tr.testInd`. Η απόδοση (accuracy) κάθε μοντέλου προκύπτει από τον τύπο:  $1 - \text{percentErrors}$ .

Συμπληρώστε τα κελιά του αρχείου “resultsXXXXXXXXXX.xlsx” που αντιστοιχούν στο ερώτημα 5.

**Για ποιον αριθμό κύκλων εκπαίδευσης παρατηρείτε τη χειρότερη απόδοση του δικτύου και για ποιον την καλύτερη; Πώς δικαιολογείτε τα αποτελέσματα αυτά;**

Συνοδέψτε τις απαντήσεις με **γραφήματα** που δείχνουν τις μέσες αποδόσεις σε συνάρτηση με τον αριθμό των κύκλων εκπαίδευσης. Μπορείτε να δημιουργήσετε τα γραφήματα στο matlab ή στο excel, ή με όποιο άλλο τρόπο θέλετε. Σε κάθε περίπτωση να τα συμπεριλάβετε στην αναφορά σας.

#### *Μελέτη της επίδρασης του αριθμού νευρώνων κρυφού επιπέδου*

6. Δημιουργήστε δίκτυο με 1 κρυφό επίπεδο και 5, 10, 20, 50, 100, 300 νευρώνες κρυφού επιπέδου και εκπαιδεύστε το με τις παραμέτρους που περιγράφονται στο βήμα 4, χρησιμοποιώντας 500 κύκλους εκπαίδευσης (epochs). Για κάθε διαφορετικό αριθμό νευρώνων κρυφού επιπέδου να εκπαιδεύσετε το δίκτυο 5 φορές και να υπολογίσετε την μέση απόδοση του εκπαιδευμένου μοντέλου με βάση το test set. Συμπληρώστε τα κελιά του αρχείου “resultsXXXXXXXXXX.xlsx” που αντιστοιχούν στο ερώτημα 6.

**Για ποιο αριθμό νευρώνων κρυφού επιπέδου παρατηρείτε καλύτερη απόδοση του δικτύου και για ποιο χειρότερη; Πώς δικαιολογείται το αποτέλεσμα αυτό;**

Συνοδέψτε τις απαντήσεις με **γραφήματα** που δείχνουν τις μέσες αποδόσεις σε συνάρτηση με τον αριθμό νευρώνων κρυφού επιπέδου. Μπορείτε να δημιουργήσετε τα γραφήματα στο matlab ή στο excel, ή με όποιο άλλο τρόπο θέλετε. Σε κάθε περίπτωση να τα συμπεριλάβετε στην αναφορά σας.

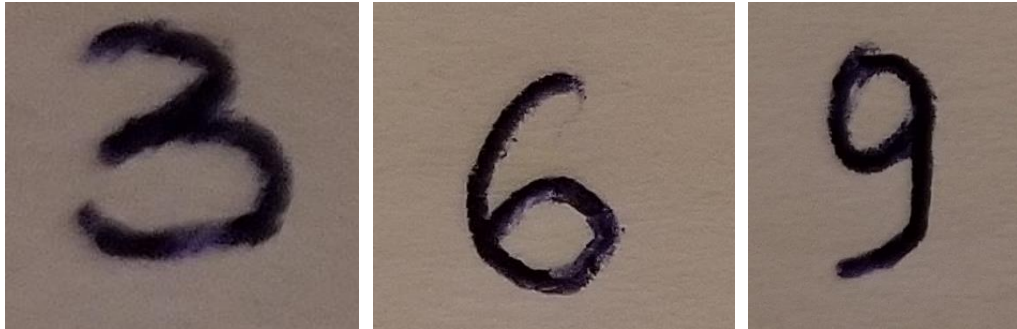
#### *Χρήση εκπαιδευμένου μοντέλου με δικά σας δεδομένα*

7. Δημιουργήστε δίκτυο με 2 κρυφά επίπεδα, με 110 νευρώνες στο 1<sup>ο</sup> κρυφό επίπεδο και 80 νευρώνες στο 2<sup>ο</sup> κρυφό επίπεδο και εκπαιδεύστε το χρησιμοποιώντας τις παρακάτω παραμέτρους:

- Συνάρτηση εκπαίδευσης: standard Gradient Descent backpropagation (traingd)
- Μέθοδος υπολογισμού σφάλματος δικτύου: Mean squared error
- Ρυθμός μάθησης (learning rate): 0.2
- Μέγιστος αριθμός κύκλων εκπαίδευσης: 2000
- Αποδεκτή απόκλιση εξόδου του δικτύου από την έξοδο-στόχο:  $10^{-6}$
- Αποδεκτός αριθμός διαδοχικών epochs που δεν παρατηρείται βελτίωση κατά το validation: 6
- Ποσοστό δεδομένων που θα αποτελέσουν το training set: 70%
- Ποσοστό δεδομένων που θα αποτελέσουν το validation set: 0%
- Συνάρτηση ενεργοποίησης νευρώνων κρυφού επιπέδου: Tan-Sigmoid (tansig)
- Συνάρτηση ενεργοποίησης νευρώνων επιπέδου εξόδου: Log-Sigmoid (logsig)

Γράψτε σε ένα χαρτί 3 ψηφία σε κάποια απόσταση μεταξύ τους, και φωτογραφίστε το κάθε ψηφίο ξεχωριστά (εναλλακτικά μπορείτε να τα σκανάρετε αν έχετε τη δυνατότητα). Φροντίστε στη φωτογραφία να περιλαμβάνεται μόνο 1 ψηφίο και τίποτα άλλο, και το ψηφίο να πιάνει όλη την εικόνα (μπορείτε είτε να πλησιάσετε όσο

περισσότερο γίνεται κατά τη λήψη της φωτογραφίας, είτε να κάνετε crop στην εικόνα μετά. Ενδεικτικές φωτογραφίες με τα ψηφία 3, 6 και 9:



Μεταφέρετε τις εικόνες στον υπολογιστή σας και αποθηκεύστε τις σε έναν φάκελο. Χρησιμοποιείτε τη συνάρτηση `imgPreprocessing()` που περιλαμβάνεται στο αρχείο της εργασίας, βάζοντας ως όρισμα τη διαδρομή (path) της εικόνας. Για παράδειγμα, αν η μια εικόνα έχει όνομα `img1.jpg` και βρίσκεται στον φάκελο `D:\myfiles\projectAI`, τότε για να χρησιμοποιήσετε τη συνάρτηση θα γράψετε:

```
img = imgPreprocessing('D:\myfiles\projectAI\img1.jpg');
```

Το array `img` που επιστρέφει η συνάρτηση, είναι πλέον σε μορφή τέτοια που μπορεί να μπει ως είσοδος σε ένα νευρωνικό δίκτυο. Στο δίκτυο που εκπαιδεύσατε δώστε ως είσοδο κάθε μια από τις εικόνες σας και δείτε τι θα επιστρέψει ως έξοδο το νευρωνικό δίκτυο. Καταφέρνει να αναγνωρίσει σωστά τα ψηφία που του δίνετε; Μπορείτε, αν θέλετε, να δοκιμάσετε και με τα υπόλοιπα ψηφία.

Σημείωση 1: Μπορείτε αν θέλετε να χρησιμοποιήσετε τις συναρτήσεις `confusion()` για να υπολογίσετε το confusion matrix και το ποσοστό λάθος κατηγοριοποιήσεων και `plotconfusion()` για να δημιουργήσετε σε γράφημα τον confusion matrix.

Σημείωση 2: Στον κώδικά σας μπορείτε αν θέλετε να χρησιμοποιήσετε το `%%` για να δημιουργήσετε ενότητες κώδικα (sections), και με το κουμπί “Run Section” να τρέχετε κάθε φορά μόνο ένα κομμάτι του κώδικα, ένα section.

Η άσκηση είναι **ατομική** και **υποχρεωτική**.

**Να παραδώσετε μέσω eclass ένα συμπιεσμένο αρχείο** (π.χ. .rar) το οποίο να περιέχει:

- την αναφορά σας
- το αρχείο “resultsXXXXXXXXXX.xlsx” συμπληρωμένο
- το αρχείο κώδικα που γράψατε στο MATLAB (.m) και
- τις εικόνες των ψηφίων που χρησιμοποιήσατε στο ερώτημα 7.

Το όνομα του συμπιεσμένου αρχείου να είναι το επίθετό σας με λατινικούς χαρακτήρες, ο αριθμός μητρώου σας και η λέξη “asgmt03”, στη μορφή:

PAPADAKIS\_2018000000\_asgmt03