

11 / 5 / 2025

ΥΠΟΛΟΓΙΣΤΙΚΗ ΝΟΗΜΟΣΥΝΗ

ΑΣΚΗΣΗ 3

THE SIX-BIT PARITY PROBLEM

ΕΡΓΑΣΤΗΡΙΑΚΕΣ ΑΣΚΗΣΕΙΣ ΤΕΧΝΗΤΩΝ

ΝΕΥΡΩΝΙΚΩΝ ΔΙΚΤΥΩΝ

2024-2025

Ηλίας Ξανθόπουλος 58545

Πίνακας Περιεχομένων

Πίνακας Περιεχομένων.....	2
1. Εισαγωγή	3
2. Περιγραφή Κώδικα	3
2.α. Δημιουργία Dataset (<i>create_dataset.m</i>)	3
2.β. Σχεδίαση Heatmap (<i>pred_heatmap.m</i>)	3
2.γ. Κύριο Πρόγραμμα (<i>parity_nn.m</i>)	4
3. Σχολιασμός βέλτιστου δικτύου	5
4. Αξιολόγηση Βέλτιστου Δικτύου.....	6

Το παρόν report όπως και οι κώδικες της υλοποίησης είναι ανεβασμένα στο GitHub στο repository: https://github.com/iliaxant/NN_6-bit_parity_problem

1. Εισαγωγή

Η άσκηση που επιλύεται με την χρήση Τεχνητών Νευρωνικών Δικτύων (ΝΔ) είναι η “Άσκηση 3 - The six-bit parity problem”. Η υλοποίηση αποτελείται από 4 .m αρχεία MATLAB, εκ των οποίων το *parity_nn.m* αποτελεί το κύριο script, ενώ τα υπόλοιπα περιλαμβάνουν τις βοηθητικές/custom συναρτήσεις. Στα επόμενα κεφάλαια πρώτα περιγράφεται σύντομα το περιεχόμενο των προαναφερθέντων αρχείων MATLAB και μετά αναλύεται το δίκτυο που επιλέγεται για την επίλυση του προβλήματος.

Σχόλιο: Από τα 4 .m αρχεία, θα περιγραφούν όλα εκτός του *time_calc.m*, διότι το συγκεκριμένο αφορά μια συνάρτηση η οποία δεν παίζει κανέναν ρόλο στην επίλυση του προβλήματος. Το μόνο που κάνει είναι να αλλάζει το format στο οποίο προβάλλεται ο συνολικός χρόνος εκπαίδευσης, ώστε να είναι ξεκάθαρος ο χρόνος που πέρασε.

2. Περιγραφή Κώδικα

2.α. Δημιουργία Dataset (*create_dataset.m*)

Η συνάρτηση αυτή αναλαμβάνει την δημιουργία του dataset που περιγράφει πλήρως το πρόβλημα.

Το 6-bit parity problem ισοδυναμεί με μια λογική συνάρτηση η οποία δέχεται 6 bit εισόδου και επιστρέφει 1 bit εξόδου. Επομένως τα δεδομένα εκπαίδευσης δεν είναι τίποτα άλλο από έναν πίνακα αληθείας που περιγράφει την λογική συνάρτηση, με το λογικό ‘1’ στην έξοδο να αντιστοιχεί στην κλάση “Even” (class 1) και το λογικό ‘0’ στην κλάση “Odd” (class 2).

2.β. Σχεδίαση Heatmap (*pred_heatmap.m*)

Η συνάρτηση αυτή αναλαμβάνει την δημιουργία ενός heatmap κατά την αξιολόγηση του δικτύου με σκοπό την οπτικοποίηση των αποτελεσμάτων και της ακρίβειας.

Το σχεδιαζόμενο διάγραμμα απεικονίζει τις δυαδικές τιμές των διαφόρων μεγεθών με είσοδο το καθένα από τα testing αντικείμενα κατά την αξιολόγηση του δικτύου. Τα μεγέθη τα οποία αναπαριστώνται είναι οι 6 εισοδοί του δικτύου, η μία πρόβλεψη / έξοδος, η επιθυμητή έξοδος και το αν η πρόβλεψη και το επιθυμητό αποτέλεσμα είναι ίσα.

Αυτό το διάγραμμα είναι πολύ χρήσιμο για την γρήγορη επισκόπηση της επίδοσης του δικτύου, όπως και για τον εντοπισμό των δεδομένων στα οποία αποτυγχάνει το δίκτυο.

2.γ Κύριο Πρόγραμμα (*parity_nn.m*)

Το *parity_nn.m* αποτελεί τον εγκέφαλο της υλοποίησης και χωρίζεται σε 4 μέρη:

a. Ρυθμίσεις και δημιουργία δεδομένων εκπαίδευσης:

Σε αυτήν την ενότητα αξιοποιείται η συνάρτηση *create_dataset.m* για την δημιουργία του dataset, τυπώνονται σημαντικές πληροφορίες για το πρόβλημα και επιτρέπεται στον χρήστη να επιλέξει ορισμένες ρυθμίσεις εκτέλεσης. Αυτές οι ρυθμίσεις περιλαμβάνουν την αρχικοποίηση της γεννήτριας τυχαίων αριθμών (για την αναπαραγωγή συγκεκριμένων αποτελεσμάτων ή τυχαίων), την επιλογή του ποσοστού των δεδομένων που θα αξιοποιηθεί για τη εκπαίδευση και συνεπώς και για την αξιολόγηση (το οποίο θα λειτουργήσει μόνο κάνοντας `uncomment` τα αντίστοιχα κομμάτια κώδικα) και την επιλογή της λειτουργίας εκπαίδευσης. Πιο συγκεκριμένα, θέτοντας την μεταβλητή *train_mode* ίση με 1 εκπαιδεύεται και αξιολογείται μόνο το καλύτερο από τα δίκτυα που εξετάστηκαν (το οποίο είναι και το ζητούμενο της εργαστηριακής άσκησης), ενώ θέτοντας την ίση με 0, εκπαιδεύονται και αξιολογούνται όλα τα δίκτυα που έχουν οριστεί.

b. Ορισμός δομής και παραμέτρων δικτύων:

Το μέρος αυτό είναι υπεύθυνο για την κατασκευή των νευρωνικών δικτύων που πρόκειται να εξεταστούν και για την ρύθμιση των υπερπαραμέτρων εκπαίδευσης. Κατά την κατασκευή των δικτύων ορίζονται χαρακτηριστικά όπως είναι το πλήθος των επιπέδων και των νευρώνων, οι συναρτήσεις ενεργοποίησης όλων των επιπέδων και η μέθοδος αρχικοποίησης των βαρών και του bias των νευρώνων. Η ρύθμιση των παραμέτρων training αφορά μεγέθη όπως η μέθοδος εκπαίδευσης, ο μέγιστος αριθμός εποχών, η συνάρτηση σφάλματος και το όριο της τιμής της που πρέπει να ξεπεραστεί για τερματισμό και η ελάχιστη βαθμίδα που απαιτείται για να συνεχίσει η εκπαίδευση (αυτή η παράμετρος δεν έχει ισχύ για όλες τις μεθόδους εκπαίδευσης, παρά μόνων αυτές που χρησιμοποιούν την βαθμίδα).

c. Εκπαίδευση Νευρωνικών Δικτύων:

Σε αυτό το block εκπαιδεύονται τα ΝΔ του προηγούμενου μέρους με βάση τις προαναφερθείσες παραμέτρους εκπαίδευσης. Στο τέλος της εκπαίδευσης κάθε δικτύου προβάλλονται αποτελέσματα όπως είναι ο συνολικός χρόνος και εποχές εκπαίδευσης, η αιτία τερματισμού και χαρακτηριστικές τιμές της καλύτερης εποχής. Επίσης προβάλλεται και το GUI αποτελεσμάτων του *nprtool*, αλλά μόνο για το τελευταίο δίκτυο που εκπαιδεύτηκε.

d. Αξιολόγηση μοντέλων:

Το τελευταίο κομμάτι του κυρίου προγράμματος εκτελεί το testing των Νευρωνικών Δικτύων είτε πάνω σε όλα τα δεδομένα εκπαίδευσης, είτε πάνω στο ποσοστό που έχει οριστεί στο πρώτο μέρος του κυρίου προγράμματος. Στο τέλος της αξιολόγησης τυπώνονται οι έξοδοι των μοντέλων σε όλα τα αντικείμενα testing, το πλήθος των σωστών προβλέψεων και η ακρίβεια (accuracy) του μοντέλου. Επιπλέον προβάλλονται

και χρήσιμα διαγράμματα όπως είναι το confusion matrix, ένα heatmap μέσω της συνάρτησης του *pred_heatmap.m* και το διάγραμμα σφάλματος συναρτήσει των εποχών εκπαίδευσης (παρόλο που τυπώνεται στο testing, αφορά την εκπαίδευση).

3. Σχολιασμός βέλτιστου δικτύου

Στην συγκεκριμένη υλοποίηση έχουν εξεταστεί 6 διαφορετικές δομές Νευρωνικών Δικτύων, τόσο με διαχωρισμό δεδομένων σε training και testing data, όσο και χωρίς. Από αυτά, το “net_6” χωρίς data splitting είναι αυτό με την καλύτερη επίδοση. Η επιλογή αυτή έγινε μόνο με βάση τον συνολικό χρόνο και εποχές εκπαίδευσης, καθώς από άποψη ακρίβειας και οι 6 δομές είναι το ίδιο ικανοποιητικές (χωρίς data split, όλες 100% accuracy).

Πιο λεπτομερώς, το data splitting εν τέλει δεν εφαρμόζεται και δεν λαμβάνεται υπόψη λόγω της ίδιας της φύσης του προβλήματος. Όπως προαναφέρθηκε, η λύση αντιστοιχεί σε μια λογική συνάρτηση/πύλη. Για να εκπαιδευτεί ένα δίκτυο ώστε να προσομοιώνει την λειτουργία αυτής της συνάρτησης, πρέπει να εκπαιδευτεί πάνω σε όλο τον πίνακα αληθείας της. Εάν αγνοηθεί οποιοδήποτε τμήμα αυτού του πίνακα, τότε το δίκτυο δεν γνωρίζει όλες τις πτυχές του προβλήματος και έτσι αποτυγχάνει όταν αξιολογείται πάνω στα προηγούμενως απομακρυσμένα από την εκπαίδευση τμήματα του πίνακα αληθείας.

Αυτό επιβεβαιώνεται παρατηρώντας διαδοχικές εκτελέσεις του κυρίου κώδικα με *train_mode=0* και *data_split* ίσο με π.χ. 0.8. Το accuracy όλων των δικτύων σε κάθε εκτέλεση μπορεί να διαφέρει σημαντικά από την προηγούμενη (αφού σε κάθε εκτέλεση διαχωρίζεται τυχαία το dataset) και κυμαίνεται από ~26% έως και ~76%, με κάθε φορά να αναδεικνύεται διαφορετικό δίκτυο ως το πιο αποτελεσματικό.

Αυτό το πρόβλημα ίσως λυνόταν εάν τα ψηφία εισόδου ήταν πολύ περισσότερα από 6 και επομένως υπήρχαν πολύ περισσότερα δεδομένα (αυξάνονται εκθετικά). Ωστόσο, στην συγκεκριμένη υλοποίηση τα 6 ψηφία αντιστοιχούν σε πολύ μικρό dataset (μόνο 64 σετ εισόδου και εξόδου), το οποίο μάλιστα δεν μπορεί ούτε να επαυξηθεί λόγω του πεπερασμένου πλήθους bit της εισόδου.

Ως προς την δομή του βέλτιστου δικτύου “net_6”, αποτελείται από 1 κρυφό επίπεδο 20 νευρώνων, με τα κρυφά επίπεδα να έχουν την $tansig(x) = \frac{2}{1+e^{-2x}} - 1$ συνάρτηση ενεργοποίησης και το επίπεδο εξόδου την $purelin(x) = x$. Το ένα σε πλήθος κρυφό επίπεδο δεν αποτελεί εμπόδιο, καθώς το πρόβλημα είναι αρκετά απλό ώστε να λύνεται με τον ελάχιστο αριθμό κρυφών επιπέδων, ενώ οι νευρώνες σε αυτό είναι σχετικά πολλοί, ώστε να απαιτούνται λιγότερες εποχές για σύγκλιση. Ενώ το πρόβλημα είναι δυαδικής ταξινόμησης, χρησιμοποιείται η *purelin*, η οποία είναι χαρακτηριστική συνάρτηση regression, γιατί ένα τόσο απλό πρόβλημα, και μάλιστα δυαδικής, ταξινόμησης μπορεί να προσεγγιστεί από ένα μοντέλο παλινδρόμησης μίας εξόδου.

Ανάμεσα στην Nguyen-Widrow αρχικοποίηση και στην μηδενική αρχικοποίηση των βαρών και των bias των νευρώνων επιλέγεται η δεύτερη, διότι εφαρμόζοντας και τις

δύο σε πανομοιότυπα δίκτυα, η μηδενικής αρχικοποίησης παρατηρείται να οδηγεί πιο γρήγορα σε σύγκλιση, παρόλο που η πρώτη κατανέμει τα βάρη στρατηγικά στο δίκτυο για την αποφυγή κορεσμού των νευρώνων.

Τέλος, ως προς τις υπερπαραμέτρους, οι τιμές των περισσότερων προκύπτουν πειραματικά, έτσι ώστε να τερματίζεται η εκπαίδευση γρήγορα και την κατάλληλη στιγμή (χαμηλότερο σφάλμα). Η συνάρτηση σφάλματος mse επιλέχθηκε γιατί είναι η καταλληλότερη για προβλήματα παλινδρόμησης, ενώ η μέθοδος εκπαίδευσης Levenberg-Marquardt επειδή είναι ταχύτερη συγκριτικά με άλλες.

4. Αξιολόγηση Βέλτιστου Δικτύου

Υπενθυμίζεται ότι το βέλτιστο δίκτυο “net_6” είναι το ίδιο αποτελεσματικό με τα υπόλοιπα δίκτυα, οδηγώντας σε 100% accuracy όταν εκπαιδεύεται και αξιολογείται πάνω σε όλο το dataset. Ωστόσο, αυτό πάντα προκύπτει να είναι το ταχύτερο με συνολικούς χρόνους εκτέλεσης της τάξεως των 100ms και τις εποχές να μην ξεπερνάνε κατά μέσο όρο τις 10. Παρόλου που ανάμεσα στα διάφορα δίκτυα, οι εποχές εκπαίδευσης διαφέρουν από λίγο έως σημαντικά, οι χρόνοι εκτέλεσης είναι όλοι πολύ κοντά μεταξύ τους χωρίς να ξεπερνούν ποτέ το 1sec. Επομένως, αν και επιλέχθηκε ο “net_6” ως ο συστηματικά πιο γρήγορος, και τα 6 δίκτυα είναι το ίδιο καλά στην επίλυση του προβλήματος της *6-bit parity*.