UNIVERSITY OF PATRAS
ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΑΤΡΩΝ

Εργαστηριακή Άσκηση Μέρος Α΄ Πρόβλεψη Alzheimer's με Χρήση Νευρωνικών Δικτύων

Μαριάνθη Θώδη
AM:1084576

Μάρτιος 2025

Contents

1 Γενικά

Στόχος της παρούσας άσκησης είναι η ανάπτυξη ενός τεχνητού νευρωνικού δικτύου για την πρόβλεψη της νόσου Alzheimer, βασισμένο σε ιατρικά δεδομένα ασθενών. Η διαδικασία ξεκινά με την

απαραίτητη προεπεξεργασία των δεδομένων, ώστε να διασφαλιστεί η ορθή και αποτελεσματική εκπαίδευση του μοντέλου, αποφεύγοντας μεροληψίες που μπορεί να προκύψουν λόγω διαφορετικών κλιμάκων στα χαρακτηριστικά.Στη συνέχεια, σχεδιάζεται και εκπαιδεύεται το νευρωνικό , με στόχο να μπορεί να ταξινομεί τα δείγματα σε δύο κατηγορίες: παρουσία ή απουσία alzheimer. Ακολουθεί πειραματισμός με διαφορετικές αρχιτεκτονικές και παραμέτρους , ώστε να μελετηθεί η επίδρασή τους στην απόδοση του συστήματος.Τέλος, πραγματοποιείται βελτιστοποίηση των υπερπαραμέτρων εκπαίδευσης, με στόχο την βελτίωση της ακρίβειας και της ικανότητας του μοντέλου να γενικεύει σε νέα δεδομένα.

- Μπορείτε να δείτε το project στο **GitHub** εδώ: Alzheimer's Prediction Using Neural Networks
- Λόγω του ότι η εκτέλεση των δικτύων πραγματοποιήθηκε πολλές φορές με επαναλαμβανόμενα τρέξιματα, είναι πιθανό τα αποτελέσματα να παρουσιάζουν μικρές διαφοροποιήσεις. Αυτή η παρατήρηση οφείλεται στην τυχαιότητα που εισάγεται κατά τη διάρκεια της εκπαίδευσης και της αξιολόγησης των μοντέλων.Επιπλέον, για να εμφανιστούν τα γραφήματα και τα αποτελέσματα, θα πρέπει να εκτελέσετε το πρόγραμμα, καθώς αυτά παράγονται δυναμικά κατά την εκτέλεση.

2 A1 : Προεπεξεργασία και Προετοιμασία δεδομένων

Για την προετοιμασία των δεδομένων για εκπαίδευση , είναι σημαντικό να εφαρμοστούν κατάλληλες τεχνικές προεπεξεργασίας. Αυτές εξασφαλίζουν ότι το μοντέλο μαθαίνει αποτελεσματικά και δεν γίνεται biased από τις διαφορές στις κλίμακες των χαρακτηριστικών. Οι τεχνικές που πρόκειται να εφαρμοστούν σε ένα σύνολο δεδομένων εξαρτώνται από τις ιδιότητες του ίδιου του συνόλου δεδομένων και το πρόβλημα που έχουμε να αντιμετωπίσουμε.

2.1 Centering & Scaling

Το centering μετατοπίζει τα δεδομένα έτσι ώστε να είναι κεντραρισμένα γύρω από το μηδέν, βοηθώντας στην εξάλειψη προκατάληψης που μπορεί να προκύψει από τις διαφορές στους μέσους όρους των χαρακτηριστικών. Από την άλλη, το scaling προσαρμόζει το εύρος των δεδομένων, εξασφαλίζοντας ότι όλα τα χαρακτηριστικά έχουν παρόμοια μεγέθη, αποτρέποντας τα χαρακτηριστικά με μεγάλες τιμές να κυριαρχούν στα υπόλοιπα. Αυτές οι τεχνικές κάνουν το μοντέλο να αντιμετωπίζουν όλα τα χαρακτηριστικά ισότιμα.

2.2 Dataset

Το σύνολο δεδομένων περιλαμβάνει κυρίως αριθμητικές τιμές που σχετίζονται με παράγοντες υγείας, δημογραφίας και γνωστικής κατάστασης, και μπορεί να χρησιμοποιηθεί για την πρόβλεψη της νόσου alzheimer. Είναι ένα σχετικά μικρό σύνολο δεδομένων και περιέχει features με συνεχείς τιμές, όπως η ηλικία, το BMI και τα επίπεδα χοληστερόλης, καθώς και binary χαρακτηριστικά, όπως το κάπνισμα και η κατανάλωση αλκοόλ. Η στήλη DoctorInCharge είναι το μόνο κατηγορικό χαρακτηριστικό. Επιπλέον, δεν υπάρχουν missing values ή outliers στα δεδομένα,διασφαλίζοντας ότι δεν χρειάζεται χειρισμός για αυτά. Συνολικά, το σύνολο δεδομένων συνδυάζει συνεχείς, binary και ordinal; αριθμητικές μεταβλητές, καθιστώντας το κατάλληλο για την πρόβλεψη της νόσου alzheimer.

2.3 Μεθόδοι

Παρακάτω αναφέρονται οι βασικές μέθοδοι προεπεξεργασίας που κληθήκαμε να εξετάσουμε.

2.3.1 Centering

Η κεντροποίηση είναι μια τεχνική κατά την οποία από κάθε τιμή ενός χαρακτηριστικού αφαιρείται ο μέσος όρος του, ώστε η νέα μέση τιμή να γίνει μηδέν. Με αυτόν τον τρόπο μειώνεται το bias από μη μηδενικά means και τα χαρακτηριστικά αποκτούν πιο ομοιόμορφη κλίμακα. Είναι χρήσιμη σε μεθόδους όπως το PCA.

$$X_{\text{centered}} = X - \mu$$

2.3.2 Normalization

Η κανονικοποίηση κλιμακώνει τις τιμές των χαρακτηριστικών σε μια καθορισμένη κλίμακα, συνήθως [0, 1] ή [-1, 1], χωρίς να αλλοιώνει τις σχετικές σχέσεις μεταξύ τους. Είναι ιδιαίτερα χρήσιμη σε νευρωνικά δίκτυα, καθώς λειτουργεί καλά με συναρτήσεις ενεργοποίησης όπως sigmoid ή tanh. Με αυτόν τον τρόπο, όλα τα χαρακτηριστικά αντιμετωπίζονται με τον ίδιο τρόπο και αποφεύγεται η επίδραση ακραίων τιμών που θα μπορούσαν να επηρεάσουν το μοντέλο.

$$X_{\text{scaled}} = \frac{X - X_{\min}}{X_{\max} - X_{\min}}$$

2.3.3 Standardization

Η τυποποίηση μετατρέπει τα δεδομένα έτσι ώστε κάθε χαρακτηριστικό να έχει μέση τιμή 0 και τυπική απόκλιση 1. Υποθέτει ότι τα δεδομένα ακολουθούν κανονική κατανομή. Αυτή η μέθοδος είναι χρήσιμη σε μοντέλα που βασίζονται σε αποστάσεις ή variance και διασφαλίζει ότι όλα τα χαρακτηριστικά συνεισφέρουν ισότιμα στο μοντέλο.

$$X_{\text{standardized}} = \frac{X - \mu}{\sigma}$$

2.3.4 One Hot Encoding

Δημιουργεί μια δυαδική στήλη για κάθε κατηγορία, όπου το "1" δηλώνει την παρουσία της συγκεκριμένης κατηγορίας. Αυτή η τεχνική είναι χρήσιμη για δεδομένα κατηγορικά unordered, καθώς αποφεύγει την υπόθεση για οποιαδήποτε σειρά μεταξύ των κατηγοριών και είναι απαραίτητη για τη μετατροπή κατηγορηματικών χαρακτηριστικών σε μορφή που να μπορεί να επεξεργαστεί το μοντέλο. Για ένα σύνολο κατηγοριών c1, c2, . . . , cn, το χαρακτηριστικό x κωδικοποιείται ως:

$$x_{\text{one-hot}} = \begin{bmatrix} 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{bmatrix}$$

2.4 Cross Validation

Η cross validation είναι μια τεχνική που χρησιμοποιείται για να αξιολογήσουμε πόσο καλά μπορεί ένα μοντέλο να προβλέψει νέα, άγνωστα δεδομένα. Γίνεται με τον διαχωρισμό του dataset σε μικρότερα υποσύνολα (folds). Το μοντέλο εκπαιδεύεται σε κάποια από αυτά και δοκιμάζεται στο υπόλοιπο. Αυτή η διαδικασία επαναλαμβάνεται πολλές φορές, κάθε φορά με διαφορετικό υποσύνολο για έλεγχο.Στο

τέλος, υπολογίζουμε τον μέσο όρο των αποτελεσμάτων από όλες τις επαναλήψεις, ώστε να έχουμε μια πιο αξιόπιστη εικόνα της απόδοσης του μοντέλου. Χρησιμοποιούμε αυτή τη μέθοδο γιατί μειώνει τον κίνδυνο το μοντέλο να "μάθει απ' έξω" τα δεδομένα εκπαίδευσης και να μην αποδίδει σωστά στα νέα (overfitting).

2.5 Απαντήσεις

Στην περίπτωση αυτής της εργασίας, το centering από μόνο του δεν είναι αρκετό, καθώς απλώς μετατοπίζει τα δεδομένα ώστε να έχουν μέση τιμή μηδέν, χωρίς όμως να αντιμετωπίζει τις διαφορές στην κλίμακα ή τη διακύμανση μεταξύ των χαρακτηριστικών. Αυτό που χρειαζόμαστε είναι είτε normalization είτε standardization, οι οποίες όχι μόνο κεντράρουν τα δεδομένα, αλλά τα κλιμακώνουν σε ένα κοινό εύρος. Αυτό διασφαλίζει ότι κανένα χαρακτηριστικό δεν θα έχει μεγαλύτερη επιρροή στο μοντέλο λόγω των διαφορών στην κλίμακά του.Όσον αφορά την κωδικοποίηση one-hot, αυτή δεν είναι απαραίτητη στην περίπτωση αυτή, καθώς τα κατηγορικά δεδομένα είναι ήδη αριθμητικά κωδικοποιημένα, γεγονός που επιτρέπει στο μοντέλο να τα επεξεργαστεί απευθείας χωρίς να απαιτείται η δημιουργία επιπλέον δυαδικών στηλών.

Για το ερώτημα A1, επέλεξα να αξιολογήσω τις μεθόδους normalization και standardization σε ένα μοντέλο logistic regression, προκειμένου να εξετάσω την απόδοση του μοντέλου στην πρόβλεψη του alzheimer πριν την τροφοδότηση του νευρωνικού με το dataset,να λάβω μία πρώτη ένδειξη. Από τις μετρικές που χρησιμοποιούνται για την αξιολόγηση του μοντέλου — accuracy, precision, recall, f1, roc — οι πιο σημαντικές για το συγκεκριμένο πρόβλημα είναι οι recall, f1, και roc. Η recall είναι κρίσιμη, καθώς δείχνει πόσους ασθενείς με Alzheimer καταφέρνει το μοντέλο να εντοπίσει σωστά, ενώ η f1 score μας προσφέρει μια ένδειξη της ισορροπίας μεταξύ precision και recall. Η roc αξιολογεί τη συνολική απόδοση του μοντέλου, λαμβάνοντας υπόψη διαφορετικά επίπεδα κατωφλίου,όπου το κατώφλι καθορίζει τη πιθανότητα πάνω από την οποία το μοντέλο ταξινομεί ένα δείγμα ως "θετικό".

Σημειώνεται ότι πριν γίνει οποιαδήποτε εκπαίδευση, τα δεδομένα χωρίζονται αρχικά σε δύο σύνολα: 80% για εκπαίδευση και 20% για τελικό hold-out test και οι παραπάνω μετρικές για την απόδοση του μοντέλου ,υπολογίζονται μέσω 5 fold cross validation(80%).

Παρακάτω παρατηρώ ότι τα αποτελέσματα με βάση τα confusion matrices για την τυποποίηση (standardization) και την κανονικοποίηση (normalization) στη λογιστική παλινδρόμηση διαφέρουν ως προς τις απόλυτες τιμές, ωστόσο οι μετρικές απόδοσης παραμένουν παρόμοιες. Αυτό υποδηλώνει ότι, παρά τις διαφορές στους αριθμούς του πίνακα, η συνολική απόδοση του μοντέλου δεν επηρεάζεται από τη μέθοδο προεπεξεργασίας,συνεπώς αυθαίρετα επιλέγω να χρησιμοποιήσω standardization.

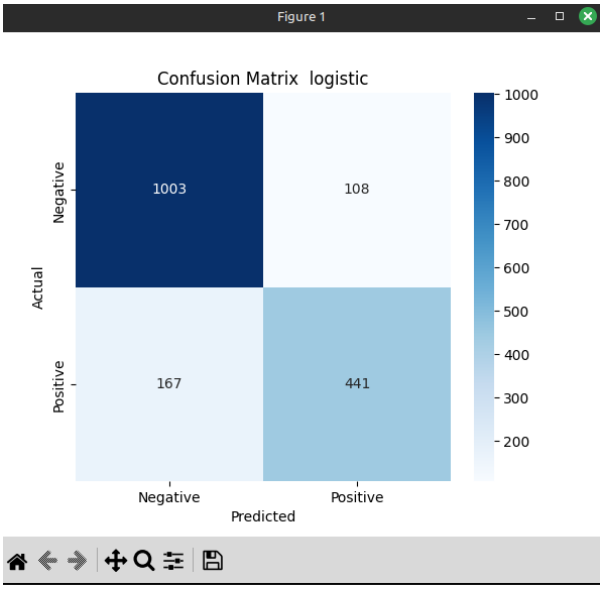

Figure 1: Confusion matrix με normalization

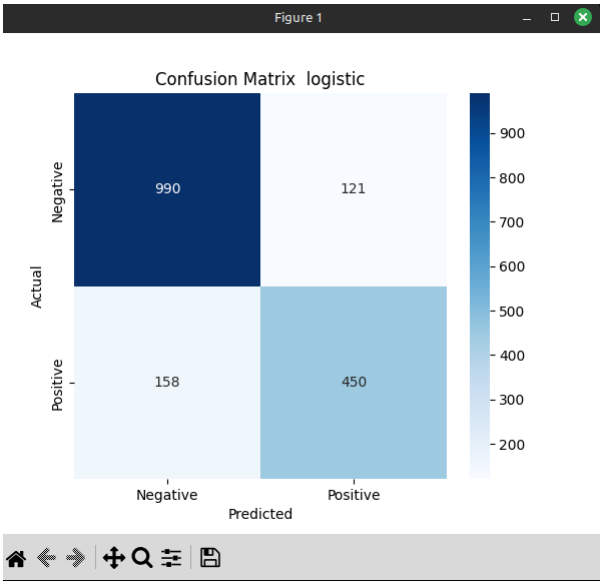

Figure 2: Confusion matrix με standardization

3 Overall Performance

Όλες οι μετρικές μαζί αποσκοπούν στο να αξιολογήσουν πόσο καλά το μοντέλο εκτελεί τις προβλέψεις του και πόσο καλά μπορεί να αποδώσει σε νέα, άγνωστα δεδομένα, ενώ ελέγχουν την ταχύτητα εκπαίδευσης και τον κίνδυνο overfitting.

- **Accuracy:** Αυτή είναι η συνολική ποσοστιαία αναλογία σωστών προβλέψεων που έκανε το μοντέλο. Όσο υψηλότερη είναι, τόσο καλύτερη είναι η απόδοση του μοντέλου.
- **Curve Score:** Αυτό το σκορ δείχνει πόσο καλά ταιριάζουν η ακρίβεια εκπαίδευσης και η ακρίβεια επικύρωσης κατά τη διάρκεια της εκπαίδευσης. Υπολογίζεται από τη διαφορά μεταξύ της τελικής ακρίβειας εκπαίδευσης και επικύρωσης. Όσο μεγαλύτερο είναι το σκορ, τόσο λιγότερο υπερβολική είναι η εκμάθηση (overfitting).
- **Epochs Score:** Αυτό μετράει πόσο γρήγορα το μοντέλο συγκλίνει, δηλαδή, πόσες εποχές (επανάληψη εκπαίδευσης) χρειάστηκαν για να φτάσει σε μια σταθερή ή βέλτιστη απόδοση. Όσο μεγαλύτερο το σκορ, τόσο πιο γρήγορα έμαθε το μοντέλο.
- **Overfit Penalty:** Αυτή είναι μια ποινή που εφαρμόζεται όταν το μοντέλο υπερεκπαιδεύεται στα δεδομένα εκπαίδευσης, κάτι που φαίνεται από τη μεγάλη διαφορά μεταξύ της ακρίβειας εκπαίδευσης και της ακρίβειας επικύρωσης. Όσο μεγαλύτερη είναι η διαφορά, τόσο μεγαλύτερη είναι η ποινή και αυτό μειώνει το τελικό σκορ.
- **Total Score:** Αυτό είναι το τελικό συνδυασμένο σκορ, που υπολογίζεται από την ακρίβεια, το σκορ καμπύλης, το σκορ εποχών και την ποινή υπερβολικής μάθησης. Δίνει μια ισχυρή εικόνα της απόδοσης του μοντέλου, λαμβάνοντας υπόψη τόσο την ακρίβεια όσο και το πόσο καλά γενικεύει σε άγνωστα δεδομένα. Όσο μεγαλύτερο το συνολικό σκορ, τόσο καλύτερο είναι το μοντέλο.

4 A2 : Επιλογή Αρχιτεκτονικής

Το ερώτημα αυτό στοχεύει στην ανάπτυξη και αξιολόγηση ενός τεχνητού νευρωνικού δικτύου για την ταξινόμηση δεδομένων σε δύο κλάσεις, χρησιμοποιώντας τον αλγόριθμο οπισθοδιάδοσης (backpropagation). Σκοπός είναι να πειραματιστούμε με διάφορες παραμέτρους του μοντέλου, όπως ο αριθμός των νευρώνων στο κρυφό επίπεδο, οι συναρτήσεις ενεργοποίησης και η συνάρτηση κόστους, προκειμένου να βρούμε την καλύτερη απόδοση. Η αξιολόγηση του μοντέλου θα γίνει μέσω 5 fold cross validation, ώστε να παρέχει μία εκτίμηση της απόδοσης και να κατανοήσουμε καλύτερα τις παραμέτρους που καθορίζουν την κατασκευή και τη διαμόρφωση της αρχιτεκτονικής του δικτύου.

4.1 Backpropagation

Backpropagation είναι ένας αλγόριθμος βελτιστοποίησης που χρησιμοποιείται για να ελαχιστοποιήσει το σφάλμα (ή την απώλεια) σε ένα νευρωνικό δίκτυο ενημερώνοντας τα βάρη μέσω της μεθόδου της gradient descent. Περιλαμβάνει τον υπολογισμό της παραγώγου της συνάρτησης απώλειας ως προς κάθε βάρος εφαρμόζοντας τον κανόνα αλυσίδας. Ξεκινώντας από το επίπεδο εξόδου, το σφάλμα διαδίδεται προς τα πίσω μέσα στο δίκτυο, υπολογίζοντας την παράγωγο της απώλειας για κάθε βάρος. Αυτή η παράγωγος χρησιμοποιείται για να ενημερωθούν τα βάρη στην κατεύθυνση που μειώνει το σφάλμα. Ενημέρωση των βαρών δίνεται από τον τύπο:

$$w_{ij} = w_{ij} - \eta \cdot \frac{\partial L}{\partial w_{ij}}$$

Όπου:

- w_{ij} είναι το βάρος μεταξύ του i-ου και του j-ου νευρώνα,
- η είναι το ποσοστό εκμάθησης (learning rate),
- $\frac{\partial L}{\partial w_{ij}}$ είναι η παράγωγος της συνάρτησης απώλειας L ως προς το βάρος w_{ij}.

Η διαδικασία αυτή συνεχίζεται για κάθε layer μέχρι να ενημερωθούν όλα τα βάρη.

4.2 Loss Function

Μία loss function μετρά το πόσο αποκλίνουν οι προβλέψεις ενός μοντέλου από τις πραγματικές τιμές. Παράγει έναν αριθμό που εκφράζει το μέγεθος του σφάλματος του μοντέλου. Κατά τη διάρκεια της εκπαίδευσης, το μοντέλο αξιοποιεί αυτή την τιμή ώστε να προσαρμόσει τα βάρη του με στόχο τη βελτίωση των προβλέψεών του. Όσο μικρότερη είναι η απώλεια, τόσο πιο ακριβές θεωρείται το μοντέλο. Παρακάτω παρουσιάζονται ορισμένες βασικές συναρτήσεις , οι οποίες χρησιμοποιούνται ανάλογα με το είδος του προβλήματος που επιλύει το δίκτυο.

4.2.1 Cross Entropy

Η συνάρτηση απώλειας cross entropy χρησιμοποιείται όταν το μοντέλο προβλέπει πιθανοτήτες. Λειτουργεί καλά με έξοδο sigmoid και είναι ιδιαίτερα αποτελεσματική για δυαδική ταξινόμηση((alzheimer και μη alzheimer). Αυτή η συνάρτηση επιβάλλει αυστηρές ποινές για σίγουρες λανθασμένες προβλέψεις, πράγμα που βοηθά το μοντέλο να μάθει πιο αποτελεσματικά.

Η λογαριθμική φύση της cross entropy σημαίνει ότι η ποινή αυξάνεται εκθετικά καθώς η προβλεπόμενη πιθανότητα απομακρύνεται από την αληθινή τιμή. Αυτό αναγκάζει το μοντέλο να βελτιώσει την εμπιστοσύνη του στις σωστές προβλέψεις, ενώ αποθαρρύνει την εμπιστοσύνη όταν κάνει λανθασμένες προβλέψεις.

$$\mathrm{CE}(y, \hat{y}) = -\left[y \cdot \log(\hat{y}) + (1 - y) \cdot \log(1 - \hat{y})\right]$$

Όπου:

- $y \in \{0, 1\}$: true label
- $\hat{y} \in (0, 1)$: predicted probability

4.2.2 MSE(Mean Squared Error)

Το mse μετρά τη μέση τετραγωνική διαφορά μεταξύ των προβλεπόμενων και των πραγματικών τιμών. Υπολογίζει πόσο μακριά είναι οι προβλέψεις του μοντέλου, με τα μεγαλύτερα σφάλματα να έχουν μεγαλύτερο αντίκτυπο λόγω του τετραγωνισμού των διαφορών. Το mse λειτουργεί καλά για μοντέλα παλινδρόμησης όπου ο στόχος είναι η πρόβλεψη συνεχών τιμών.

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

όπου y_i είναι η πραγματική (true) τιμή, $\hat{y}_i$ είναι η προβλεπόμενη τιμή από το μοντέλο, και n είναι ο αριθμός των δεδομένων.

4.2.3 Accuracy

Η ακρίβεια είναι η αναλογία των σωστών προβλέψεων προς το σύνολο των προβλέψεων. Δίνει μια πρακτική έννοια για το πόσο καλό είναι το μοντέλο στο classification.

$$\mathrm{Accuracy} = \frac{\text{Correct Predictions}}{\text{Total Predictions}}$$

4.2.4 Απαντήσεις

Για το μοντέλο πρόβλεψης της άνοιας, θεωρώ ότι η επιλογή του cross entropy είναι η καταλληλότερη. Δεδομένου ότι πρόκειται για πρόβλημα δυαδικής ταξινόμησης και το μοντέλο εξάγει πιθανότητες (δηλαδή, αν ο ασθενής έχει ή όχι άνοια), αποτελεί την πιο αποτελεσματική συνάρτηση κόστους. Μετρά τη διαφορά μεταξύ της πραγματικής τιμής και της προβλεπόμενης πιθανότητας, επιβάλλοντας μια ισχυρή ποινή όταν το μοντέλο κάνει λανθασμένες προβλέψεις, κάτι που ενισχύει τη διαδικασία μάθησης και βοηθά το μοντέλο να προσαρμοστεί γρηγορότερα.

Από την άλλη, το MSE δεν είναι κατάλληλο για την παρούσα περίπτωση, καθώς αντιμετωπίζει το πρόβλημα ως περίπτωση παλινδρόμησης, όπου οι προβλέψεις είναι συνεχείς τιμές και όχι πιθανότητες. Το MSE δεν επιβάλλει ποινές στις λανθασμένες προβλέψεις με την ίδια ένταση, γεγονός που ενδέχεται να περιορίσει την απόδοση του μοντέλου και να μειώσει την ταχύτητα εκπαίδευσης.

Ενώ η ακρίβεια (accuracy) μπορεί να είναι χρήσιμη για την αξιολόγηση του μοντέλου μετά την εκπαίδευση, δεν είναι κατάλληλη για χρήση κατά τη διάρκεια της εκπαίδευσης, καθώς δεν παρέχει "κατευθύνσεις" για να καθοδηγήσει την προσαρμογή των βαρών του μοντέλου. Ειδικότερα, η ακρίβεια δεν είναι παραγωγίσιμη και επομένως, δεν μπορεί να βοηθήσει το μοντέλο να βελτιωθεί μέσω της διαδικασίας gradient descent, κάτι που καθιστά τη χρήση της αναποτελεσματική για τη φάση εκπαίδευσης.

Για όλους αυτούς τους λόγους, επιλέγω το cross entropy, καθώς είναι η καταλληλότερη loss function για το συγκεκριμένο πρόβλημα και θα συμβάλει στην πιο αποτελεσματική εκπαίδευση του μοντέλου.

4.3 Ρυθμός Νευρώνων στο Επίπεδο Εξόδου

Για τη δυαδική κατηγοριοποίηση, το επίπεδο εξόδου αποτελείται από ένα μόνο νευρώνα. Αυτό συμβαίνει επειδή ο στόχος είναι να προβλεφθεί μία από τις δύο κατηγορίες , οι οποίες μπορούν να αναπαρασταθούν από μια μοναδική τιμή . Το μοντέλο θα εξάγει μια πιθανότητα, όπου:

- Μια τιμή κοντά στο 0 αντιπροσωπεύει μία κατηγορία ("χωρίς άνοια").
- Μια τιμή κοντά στο 1 αντιπροσωπεύει την άλλη κατηγορία ("με άνοια").

4.4 Επιλογή Συνάρτησης Ενεργοποίησης για Hidden Layers

Παρακάτω αναλύονται οι πιθανές συναρτήσεις ενεργοποίσης για τους κρυφούς νευρώνες :

4.4.1 Tanh

Η συνάρτηση tanh μετατρέπει την είσοδο σε τιμές μεταξύ -1 και 1. Αυτό σημαίνει ότι είναι κεντραρισμένη στο μηδέν, διευκολύνοντας τη διαδικασία μάθησης, εξασφαλίζοντας ότι τα gradients έχουν θετικές και αρνητικές τιμές.

$$\tanh(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$

Με εύρος εξόδου: $[-1, 1]$

4.4.2 SiLU

Η silu είναι μια συνάρτηση που συνδυάζει τα πλεονεκτήματα των sigmoid και γραμμικής. Η συνάρτηση είναι ομαλή, μη μονοτονική, και η έξοδος πολλαπλασιάζεται με την sigmoid της εισόδου, γεγονός που αποτρέπει την εξαφάνιση των gradient.

$$\text{SiLU}(x) = x \cdot \sigma(x) = x \cdot \frac{1}{1 + e^{-x}}$$

Με εύρος εξόδου: $(0, \infty)$

4.4.3 ReLU

Η συνάρτηση relu είναι μια συνάρτηση ενεργοποίησης που επιστρέφει την είσοδο απευθείας αν είναι θετική, διαφορετικά επιστρέφει το 0.

$$\text{ReLU}(x) = \max(0, x)$$

Με εύρος εξόδου: $[0, \infty)$

4.4.4 Απαντήσεις

Η συνάρτηση relu είναι πιο κατάλληλη για την πρόβλεψη του αλτσχάιμερ λόγω της ικανότητάς της να μετριάζει το πρόβλημα της εξαφάνισης gradient και να εξασφαλίζει ότι τα βάρη παραμένουν μεγάλα για τις ανανεώσεις του backpropagation . Επιπλέον,ο απλός υπολογισμός της μειώνει τον χρόνο εκπαίδευσης σε σύγκριση με τις υπόλειπες συναρτήσιες.
Στην συνέχεια η συνάρτηση tanh έχει το πρόβλημα της εξαφάνισης των gradient, όταν η είσοδος είναι είναι ένας μεγάλος θετικός ή αρνητικός αριθμός, η έξοδός της πλησιάζει το 1 ή το -1, κάνοντάς την παράγωγο πολύ μικρή. Αυτό οδηγεί σε μικρά gradients κατά την οπισθοδιάδοση, γεγονός που προκαλεί κακές ανανεώσεις των βαρών
Τέλος , η συνάρτηση silu είναι λίγο πιο αργή από τη relu, καθώς περιλαμβάνει μια συνάρτηση sigmoid, η οποία απαιτεί τον υπολογισμό μιας εκθετικής συνάρτησης, κάτι που απαιτεί περισσότερο χρόνο και υπολογιστική ισχύ σε σχέση με τη απλή λειτουργία κατωφλίου που χρησιμοποιεί relu.

4.5 Επιλογή Συνάρτησης για Νευρώνα Εξόδου

4.5.1 Sigmoid

Η συνάρτηση sigmoid μετατρέπει οποιαδήποτε πραγματική τιμή εισόδου σε ένα εύρος μεταξύ 0 και 1, όπου η έξοδος μπορεί να ερμηνευτεί ως η πιθανότητα της θετικής κατηγορίας. Στη συνέχεια, εφαρμόζεται ένα κατώφλι (συνήθως 0,5) για να προσδιοριστεί το decision bounary για το διαχωριμό των κατηγοριών.

$$\sigma(x) = \frac{1}{1 + e^{-x}}$$

Όπου:

- x είναι η είσοδος στον νευρώνα (σταθμισμένο άθροισμα από το προηγούμενο layer)
- $\sigma(x) \in (0, 1)$

4.5.2 Linear

Η γραμμική συνάρτηση απλά περνά την είσοδο χωρίς καμία τροποποίηση. Εξάγει πραγματικές τιμές χωρίς να εφαρμόζει περιορισμούς .

$$f(x) = x$$

4.5.3 Softmax

Η συνάρτηση softmax είναι μια επέκταση της συνάρτησης sigmoid που μετατρέπει τις τιμές εξόδου (logits) σε πιθανότητες για όλες τις δυνατές κατηγορίες, εξασφαλίζοντας ότι το άθροισμα των πιθανοτήτων είναι 1. Η έξοδος κάθε νευρώνα αντιπροσωπεύει την πιθανότητα το εισερχόμενο δεδομένο να ανήκει σε μια συγκεκριμένη κατηγορία.

$$\text{Softmax}(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$$

Όπου:

- z_i είναι το input για την κατηγορία i(logits)
- Η έξοδος είναι ένα διάνυσμα πιθανοτήτων που αθροίζονται στο 1

4.5.4 Απαντήσεις

Για binary classification, όπως η πρόβλεψη του αλτσχάιμερ, η συνάρτηση sigmoid είναι η πιο κατάλληλη επιλογή για το output layer, καθώς παράγει μια πιθανότητα μεταξύ 0 και 1, κάτι που είναι ιδανικό για την ερμηνεία αν ένα άτομο έχει ή όχι αλτσχάιμερ.Χρησιμοποιεί ένα threshold, αν η πιθανότητα είναι πάνω από πχ 0,5, προβλέπεται "έχει αλτσχάιμερ", και αν είναι κάτω από 0,5, "δεν έχει αλτσχάιμερ".Από την άλλη, η γραμμική συνάρτηση ενεργοποίησης δεν είναι κατάλληλη, γιατί μπορεί να παράγει οποιονδήποτε πραγματικό αριθμό (όχι μια πιθανότητα). Η συνάρτηση softmax, από την άλλη πλευρά, είναι σχεδιασμένη για multiclass classification, κάτι που δεν είναι απαραίτητο εδώ, καθώς έχουμε μόνο δύο αποτελέσματα, καθιστώντας τη sigmoid την πιο αποδοτική επιλογή για binary classification προβλήματα , όπως η πρόβλεψη του αλτσχάιμερ.

4.6 Κριτήριο τερματισμού

Early stopping είναι μια τεχνική κανονικοποίησης που χρησιμοποιείται κατά τη διάρκεια της εκπαίδευσης του μοντέλου για να αποφευχθεί η υπερεκπαίδευση . Ο κύριος στόχος της είναι να εξασφαλίσει ότι το μοντέλο μαθαίνει αποτελεσματικά χωρίς να απομνημονεύει τα δεδομένα , κάτι που θα μπορούσε να βλάψει την ικανότητά του να 'γενικεύει'. Το καλύτερο μοντέλο είναι εκείνο με τη χαμηλότερη validation loss που παρατηρείται κατά τη διάρκεια της εκπαίδευσης.

Αν η απώλεια επικύρωσης δεν βελτιώνεται για έναν καθορισμένο αριθμό εποχών (patience), η εκπαίδευση διακόπτεται. Συγκεκριμένα, αν η διαφορά στην validation loss μεταξύ διαδοχικών εποχών είναι μικρότερη από ένα όριο (ε) για P διαδοχικές εποχές, η εκπαίδευση θα σταματήσει στην εποχή t.

Τέλος η πρόβλεψη ασθενειών όπως ο αλτσχάιμερ ,απαιτεί μοντέλα που να μπορούν να γενικεύσουν καλά σε νέους ασθενείς και δεδομένα, και το early stopping βοηθάει στην αποφυγή της υπερεκπαίδευσης, που μπορεί να οδηγήσει σε κακή απόδοση σε νέα δεδομένα.

$$\text{Αν } \forall t \in [t-P, t],\ L_{\text{val}}(t) - L_{\text{val}}(t-1) < \epsilon, \text{ τότε η εκπαίδευση σταματά στην εποχή } t.$$

Όπου:

- L_{val} είναι η validation loss

4.7 Πειραματισμός για Αριθμό Νευρώνων Κρυφού Επιπέδου

4.7.1 Διαδικασία

Ο πρώτος μέρος του κώδικα αντιστοιχεί σε tuning για τον καλύτερο αριθμό κρυφών νευρώνων για την εκπαίδευση ενός δικτύου. Εφαρμόζεται *cross-validation*, εκπαιδεύοντας ένα ξεχωριστό μοντέλο για κάθε fold και συλλέγοντας μετρήσεις απόδοσης όπως ακρίβεια, loss κλπ. Κατά τη διάρκεια του training, εφαρμόζεται *early stopping* για να αποφευχθεί η υπερπροσαρμογή και καταγράφεται πόσες εποχές έτρεξε το μοντέλο και πόσο χρόνο διήρκησε. Μετά την αξιολόγηση των μοντέλων (4 συνολικά), ο κώδικας επιλέγει την καλύτερη απόδοση βάσει της μέσης validation accuracy.

4.7.2 Ανάλυση

Στο συγκεκριμένο πείραμα εξετάζεται η επίδραση του αριθμού νευρώνων στο κρυφό επίπεδο του νευρωνικού δικτύου στην απόδοση του μοντέλου. Δοκιμάζονται διαφορετικές επιλογές για το πλήθος των νευρώνων $H_1 \in \{16, 21, 32, 64\}$.

Γενική παρατήρηση: Καθώς αυξάνεται ο αριθμός των νευρώνων στο κρυφό επίπεδο, παρατηρείται βελτίωση τόσο στην ακρίβεια όσο και στη συνάρτηση απώλειας (CE Loss). Αυτό οφείλεται στο γεγονός ότι περισσότεροι νευρώνες συνεπάγονται μεγαλύτερη ικανότητα του δικτύου να μάθει πιο πολύπλοκα patterns (*non-linearity*). Παρά την εφαρμογή του *early stopping*, δεν παρατηρείται κάποια σημαντική αλλαγή στην εκπαίδευση. Αυτό συμβαίνει διότι το *early stopping* ενεργοποιείται μόνο όταν το μοντέλο αρχίζει να αποδίδει χειρότερα, αλλά η προσθήκη περισσότερων νευρώνων κάνει το μοντέλο πιο σταθερό, οπότε δεν παρατηρείται καμία πτώση στην απόδοση που να προκαλεί την πρόωρη διακοπή.

- **CE loss και accuracy ανά H_1:** Καθώς ο αριθμός των νευρώνων στο κρυφό επίπεδο αυξάνεται από $H_1 = 16$ σε $H_1 = 64$ υπό συνθήκες **τυποποίησης**, παρατηρούνται σημαντικές αλλαγές τόσο στη συνάρτηση απώλειας cross-entropy (CE) όσο και στην ακρίβεια. Οι μεταβολές αυτές αντανακλούν την ικανότητα του μοντέλου να μαθαίνει πιο σύνθετα πρότυπα.

- **Από $H_1 = 16$ σε $H_1 = 21$:**

  - Η CE loss μειώνεται από **0.5681** σε **0.5253**, δηλαδή βελτίωση **7.53%**.
  - Η ακρίβεια αυξάνεται από **72.37%** σε **74.99%**, αύξηση **2.62%**.

  Η αύξηση αυτή επιτρέπει στο μοντέλο να αποτυπώνει καλύτερα βασικά χαρακτηριστικά των δεδομένων, μειώνοντας τόσο τα σφάλματα όσο και τις λανθασμένες ταξινομήσεις.

- **Από $H_1 = 21$ σε $H_1 = 32$:**

  - Η CE loss παραμένει σχεδόν σταθερή (από **0.5253** σε **0.5249**).
  - Η ακρίβεια μειώνεται ελαφρώς από **74.99%** σε **74.46%**.

  Η στασιμότητα αυτή ενδέχεται να οφείλεται στο ότι το μοντέλο πλησιάζει σε ένα βέλτιστο επίπεδο για την πολυπλοκότητα των δεδομένων, οδηγώντας σε φθίνουσες αποδόσεις. Πιθανώς επίσης να εμφανίζονται τα πρώτα σημάδια υπερπροσαρμογής (*overfitting*).

- **Από $H_1 = 32$ σε $H_1 = 64$:**

- Η CE loss μειώνεται από **0.5249** σε **0.4891**, βελτίωση **6.83%**.
- Η ακρίβεια αυξάνεται από **74.46%** σε **77.54%**, αύξηση **4.13%**.

Το άλμα αυτό δείχνει πως, μόλις ξεπεραστεί ένα συγκεκριμένο όριο, οι επιπλέον νευρώνες επιτρέπουν στο δίκτυο να βελτιώνει τη γενίκευση και την ακρίβεια.

- **Συνολικά (από $H_1 = 16$ σε $H_1 = 64$):**
  - Η CE loss μειώνεται συνολικά κατά **13.92%**.
  - Η ακρίβεια αυξάνεται συνολικά κατά **7.14%**.
- **CE loss , accuracy over epochs:** Στην ανάλυση αυτή, εξετάσαμε τις τιμές της training και validation CE loss, καθώς και της accuracy κατά τη διάρκεια των εποχών, χρησιμοποιώντας τη μέθοδο *k fold cross validation*. Η πρόωρη διακοπή (*early stopping*) δεν εφαρμόστηκε, καθώς δεν παρατηρήθηκαν ενδείξεις που να απαιτούν τη χρήση της. Η training loss μειωνόταν σταθερά σε όλα τα folds, ενώ η validation loss δεν αυξήθηκε ποτέ, υποδεικνύοντας ότι το μοντέλο γενικεύει καλά σε δεδομένα που δεν έχει δει.
  - Η training accuracy αυξάνεται συνεχώς σε όλα τα folds, υποδηλώνοντας ότι το μοντέλο μαθαίνει αποτελεσματικά κατά τη διάρκεια της εκπαίδευσης. Η validation accuracy παρουσίασε τυπικές διακυμάνσεις, αλλά γενικά βελτιώθηκε, χωρίς απότομες πτώσεις, υποδεικνύοντας την απουσία *overfitting*. Ως αποτέλεσμα, η χρήση του *early stopping* δεν ήταν αναγκαία.
- **Αύξηση από 16 σε 21 Νευρώνες:**
  - Η training accuracy βελτιώνεται, καθώς η αυξημένη χωρητικότητα επιτρέπει στο μοντέλο να μάθει πιο σύνθετα χαρακτηριστικά των δεδομένων. Η training loss μειώνεται, καθώς το μοντέλο εντοπίζει καλύτερα τα μοτίβα των δεδομένων. Η validation accuracy επίσης βελτιώνεται, με το μοντέλο να γενικεύει καλύτερα σε δεδομένα που δεν έχει δει.
- **Αύξηση από 21 σε 32 Νευρώνες:**
  - Η training accuracy συνεχίζει να αυξάνεται, αλλά με πιο αργό ρυθμό. Η training loss μειώνεται, ωστόσο η μείωση γίνεται πιο αργή. Η validation accuracy παρουσιάζει μια πιο περιορισμένη αύξηση, ενώ η validation loss οριακά σταθεροποιείται, υποδεικνύοντας ότι το μοντέλο πλησιάζει στο βέλτιστο επίπεδο εκπαίδευσης και η πρόσθετη χωρητικότητα δεν έχει σημαντικό αντίκτυπο στην απόδοση.
- **Αύξηση από 32 σε 64 Νευρώνες:**
  - Με την αύξηση από 32 σε 64 νευρώνες, η training accuracy αυξάνεται ξανά και η training loss μειώνεται πιο γρήγορα, λόγω της αυξημένης χωρητικότητας και οι validation accuracy , loss βελτιώνονται επίσης.

Τα αποτελέσματα αυτά δείχνουν ότι, ενώ οι αρχικές αυξήσεις στον αριθμό νευρώνων αποφέρουν αύξηση της απόδοσης του μοντέλου, μπορεί να υπάρξει περίοδος στασιμότητας πριν συνεχιστεί η βελτίωση.

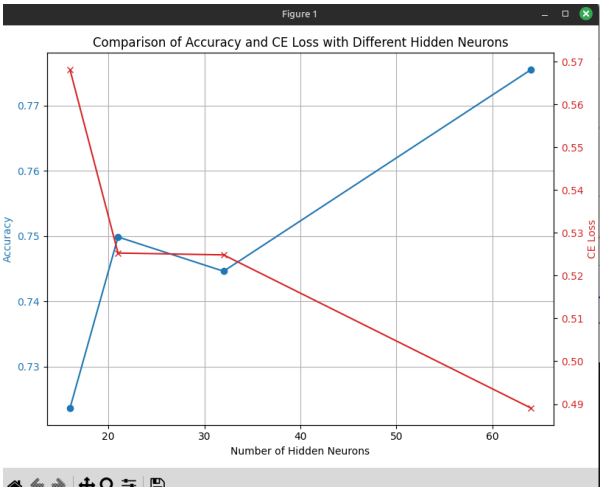

Figure 3: CE Loss, Accuracy vs Epochs

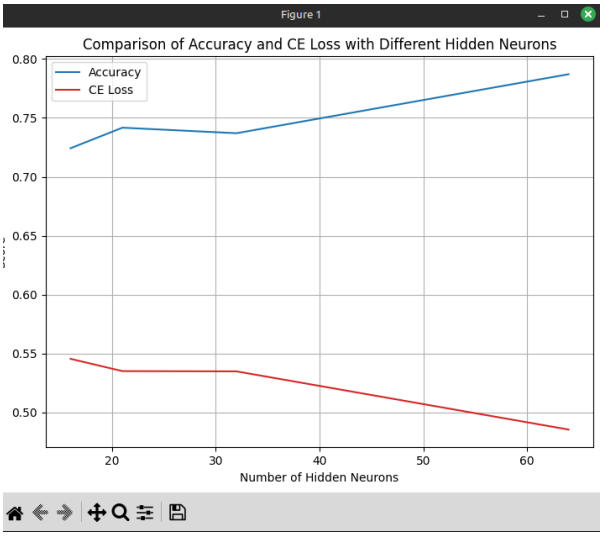

Figure 4: CE Loss, Accuracy vs Hidden Neurons

4.7.3 Συμπεράσματα

Από τα αποτελέσματα προκύπτει ότι η αύξηση των νευρώνων στο κρυφό επίπεδο βελτιώνει την ακρίβεια και μειώνει την CE loss. Το βέλτιστο αποτέλεσμα επιτυγχάνεται με $H_1 = 64$, με αύξηση της ακρίβειας κατά 4.13% και μείωση της CE loss κατά 6.83%. Παρά την εφαρμογή του early stopping, δεν παρατηρήθηκε υπερπροσαρμογή, δείχνοντας αποδοτική γενίκευση.

Αριθμός νευρώνων στο κρυφό επίπεδο	**CE loss**	**MSE**	**Acc**
H1 = 16	0.5681	0.1917	0.7237
H1 = 21	0.5253	0.1742	0.7499
H1 = 32	0.5249	0.1744	0.7446
H1 = 64	0.4891	0.1589	0.7754

Table 1: Αριθμός νευρώνων στο κρυφό επίπεδο και τις αντίστοιχες τιμές για ce loss, mse και acc με standarization

5 A3 : Μεταβολές στον ρυθμό εκπαίδευσης και σταθεράς ορμής

Ο στόχος του ερωτήματος A3 είναι να βελτιστοποιήσουμε την απόδοση του μοντέλου μας δοκιμάζοντας διαφορετικές τιμές για τον ρυθμό εκπαίδευσης και τη σταθερά ορμής. Μέσα από τη χρήση cross validation, επιδιώκουμε να βρούμε τους καλύτερους συνδυασμούς αυτών των υπερπαραμέτρων που οδηγούν στη γρηγορότερη και πιο σταθερή σύγκλιση του μοντέλου.Παρακάτω παρατίθεται μία μικρή ανάλυση των δύο υπερπαραμέτρων.

5.1 Ορμή

Η ορμή (momentum) στα νευρωνικά δίκτυα είναι μια τεχνική που βοηθά το μοντέλο να μαθαίνει πιο γρήγορα και σταθερά. Αντί να ενημερώνει τα βάρη μόνο με βάση την τρέχουσα κλίση (gradient), η ορμή προσθέτει μια "μνήμη" από τις προηγούμενες ενημερώσεις. Έτσι, όταν η κατεύθυνση της κλίσης είναι σταθερή για μερικές εποχές, η ορμή βοηθά το μοντέλο να συνεχίσει με μεγαλύτερη ταχύτητα και να αποφύγει τις μικρές, άσκοπες κινήσεις, κάνοντάς το πιο αποδοτικό στην εκπαίδευση.

1. **Gradient Descent χωρίς ορμή** : Στην τυπική κλίση κατάβασης, η ενημέρωση βάρους βασίζεται μόνο στην τρέχουσα κλίση στο βήμα ∩

$$w_{t+1} = w_t - \eta \cdot \nabla L(w_t)$$

Όπου:

- w_t: Το βάρος στο βήμα t
- α: Ρυθμός εκπαίδευσης
- $\nabla L(w_t)$: Η κλίση της συνάρτησης απώλειας στο βήμαt

2. **Gradient Descent με ορμή** : Με ορμή, η ενημέρωση βάρους περιλαμβάνει έναν όρο "ταχύτητα" που λαμβάνει υπόψη τις προηγούμενες κλίσεις. Ο όρος v_t αναπαριστά την τρέχουσα ταχύτητα, η οποία συνδυάζει τις προηγούμενες ενημερώσεις και το τρέχον gradient για να καθοδηγήσει την επόμενη ενημέρωση παραμέτρων. Το v_{t-1} είναι η προηγούμενη ταχύτητα, η οποία αποθηκεύει πληροφορίες από την τελευταία ενημέρωση και παρακολουθεί την ορμή του μοντέλου. Το β είναι ένας παράγοντας κλίμακας μεταξύ 0 και 1, ο οποίος ελέγχει πόσο επηρεάζει η προηγούμενη ταχύτητα την ενημέρωση. Ένα μεγαλύτερο β σημαίνει μεγαλύτερη εξάρτηση από τις προηγούμενες ενημερώσεις, ενώ ένα μικρότερο β δίνει μεγαλύτερο βάρος στο τρέχον gradient. Το $\nabla L(w_t)$ είναι το gradient της loss function, που δείχνει πόσο πρέπει να αλλάξουν οι παράμετροι του μοντέλου για να μειωθεί το σφάλμα. Το $(1 - \beta)$ καθορίζει πόσο επηρεάζει το τρέχον gradient την ενημέρωση.

$$v_t = \beta \cdot v_{t-1} + (1 - \beta) \cdot \nabla L(w_t)$$

Ο τύπος ενημερώνει τις παραμέτρους του μοντέλου χρησιμοποιώντας την υπολογισμένη ταχύτητα. Αυτό προσαρμόζει το μοντέλο στην κατεύθυνση που μειώνει το σφάλμα.

$$w_{t+1} = w_t - \eta \cdot v_t$$

Όπου:

- v_t: Ο όρος ταχύτητα την χρονική στιγμή t
- β: Ο συντελεστής ορμής, ο οποίος ελέγχει πόση από την προηγούμενη ταχύτητα "θυμάται"
- $\nabla L(w_t)$: Η κλίση της loss function στο βήμα t
- w_{t+1}: Το ενημερωμένο βάρος στο βήμα $t + 1$
- η: Ρυθμός εκπαίδευσης

Τώρα η ορμή πάντα είναι < 1 διότι ο όρος ταχύτητας v_t είναι ουσιαστικά ένα σταθμισμένο άθροισμα προηγούμενων κλίσεων όπου ο κανόνας ενημέρωσης σχηματίζει μια γεωμετρική σειρά, όπου τα προηγούμενα gradients έχουν βάρος που μειώνεται εκθετικά, ανάλογα με την τιμή του β.

$$v_t = \sum_{k=0}^{t} \beta^k \cdot \nabla L(w_{t-k})$$

Η παράμετρος β ελέγχει πόσες από τις προηγούμενα gradients απομνημονεύονται.Αν β=1, τότε β^k=1, και τα βάρη δεν μειώνονται.Σε αυτή την περίπτωση, το άθροισμα γίνεται άπειρο και δεν συγκλίνει.

5.2 Ρυθμός Εκπαίδευσης

Ο ρυθμός εκπαίδευσης ελέγχει πόσο μεγάλο είναι κάθε βήμα κατά την ενημέρωση των βαρών του μοντέλου κατά τη διάρκεια της εκπαίδευσης. Εάν το ηείναι μικρό, οι ενημερώσεις είναι αργές και σταθερές, αλλά η εκπαίδευση διαρκεί περισσότερο. Εάν είναι πολύ μεγάλο, οι ενημερώσεις μπορεί να είναι πολύ μεγάλες, με αποτέλεσμα το μοντέλο να πηδήξει και να μην καταλήξει ποτέ στην καλύτερη λύση.

$$w_{t+1} = w_t - \eta \cdot \nabla L(w_t)$$

5.3 Πειραματισμός για Ρυθμό Εκπαίδευσης και Σταθερά Ορμής

5.3.1 Διαδικασία

Το δεύτερο μέρος του κώδικα αφορά τη βελτιστοποίηση του τρόπου με τον οποίο μαθαίνει το νευρωνικό δίκτυο, δοκιμάζοντας διαφορετικούς συνδυασμούς ρυθμού εκμάθησης (learning rate – δηλαδή το μέγεθος του κάθε βήματος ενημέρωσης των βαρών) και ορμής (momentum – δηλαδή πόσο τα προηγούμενα gradients επηρεάζουν το τρέχον βήμα εκμάθησης). Οι συνδυασμοί αυτοί εξετάζονται αφού έχει πρώτα επιλεγεί ο αριθμός των κρυφών νευρώνων στο προηγούμενο στάδιο.

Χρησιμοποιώντας cross validation, κάθε ζεύγος παραμέτρων αξιολογείται με βάση την απόδοσή του στα δεδομένα εκπαίδευσης. Ο καλύτερος συνδυασμός επιλέγεται με βάση ένα σύνολο από μετρικές, όπως η validation και training accuracy(overfitting) ,ο μέσος αριθμός εποχών μέχρι τη σύγκλιση , καθώς και η ακρίβεια.

Αυτό το βήμα διασφαλίζει ότι το μοντέλο μαθαίνει αποτελεσματικά, βελτιστοποιώντας την απόδοσή του χωρίς να υπερεκπαιδεύεται, εξασφαλίζοντας έτσι καλύτερη γενίκευση σε άγνωστα δεδομένα.

Γενική παρατήρηση: Ένα χαμηλό learning rate ($\eta = 0.001$) οδηγεί σε αργή αλλά σταθερή μάθηση με σταθερή ακρίβεια και απώλεια, μειώνοντας την υπερπροσαρμογή αλλά χρειάζεται περισσότερο χρόνο για να συγκλίνει. Ένα $\eta = 0.05$ δίνει γρήγορη σύγκλιση και υψηλή ακρίβεια χωρίς σημαντική υπερπροσαρμογή, ειδικά με m = 0.2. Learning rates όπως $\eta = 0.1$ επιταχύνουν την εκπαίδευση αλλά προκαλούν αστάθεια και overfitting, ειδικά με υψηλό momentum. Το momentum βοηθά στη σταθεροποίηση της διαδικασίας, με υψηλότερο επιταχύνει τη σύγκλιση με κίνδυνο overfitting όμως.

5.3.2 Ανάλυση

Πραγματοποιήθηκε ανάλυση σε 6 διαφορετικά μοντέλα με διαφορετικους συνδιασμούς από learning rate και momentum ώστε να πετύχει η βελτιστοποίηση των υπερπαραμέτρων του δικτύου .Δοκιμάζονται διάφοροι συνδυασμοί των παραπάνω παραμέτρων μέσω 5 fold cross validation, ώστε να εντοπιστεί εκείνος που προσφέρει την υψηλότερη ακρίβεια. Ο στόχος είναι να διασφαλιστεί η σταθερή και αποδοτική εκπαίδευση του , βελτιώνοντας τη γενίκευση και την απόδοσή του στα δεδομένα.

- **Learning Rate $\eta = 0.001$:** Αυτή η χαμηλή τιμή οδηγεί σε σταθερή αλλά αργή μάθηση. Το μοντέλο βελτιώνεται σταδιακά με την πάροδο του χρόνου χωρίς απότομες αλλαγές στην απόδοση, γεγονός που δείχνει σταθερή σύγκλιση, αλλά με κόστος βραδύτερης σύγκλισης, περίπου 100 εποχές.

- **Momentum $\mu = 0.20$ ή $\mu = 0.60$:** Όταν το momentum είναι $\mu = 0.2$, οι ενημερώσεις βάρους είναι πιο σταθερές, με CE Loss 0.4585 και Accuracy 0.8040, επιτυγχάνοντας πιο αργή αλλά αξιόπιστη βελτίωση. Ωστόσο, απαιτούνται περισσότερες εποχές για σύγκλιση. Με $\mu = 0.6$, παρατηρούμε ταχύτερη σύγκλιση, με CE Loss 0.4173 και Accuracy 0.8266, αλλά και μεγαλύτερες διακυμάνσεις στην απόδοση, γεγονός που μπορεί να μειώσει τη σταθερότητα του μοντέλου.
- **Training Accuracy/Loss vs. Validation Accuracy/Loss:** Η ακρίβεια στο training set αυξάνεται σταθερά, και η ακρίβεια στο validation set ακολουθεί παρόμοιο μοτίβο, με λίγες διακυμάνσεις. Το μικρό κενό ανάμεσα στην απώλεια εκπαίδευσης και επικύρωσης δείχνει ότι το μοντέλο γενικεύει καλά και δεν απομνημονεύει απλώς τα δεδομένα.

- **Learning Rate $\eta = 0.05$:** Ο ρυθμός μάθησης $\eta = 0{,}05$ επιτρέπει ταχύτερη και αποτελεσματικότερη σύγκλιση σε σχέση με το $\eta = 0.001$, διασφαλίζοντας σταθερότητα και ομαλή πρόοδο προς το ελάχιστο χωρίς υπερβολικές διακυμάνσεις.

  - **Momentum $\mu = 0.20$ ή $\mu = 0.60$:** Για momentum $\mu = 0.20$, η απώλεια είναι 0,3894 και η ακρίβεια 0,8359, ενώ για $\mu = 0.60$, η απώλεια είναι 0,3956 και η ακρίβεια 0,8336. Η μικρή διαφορά υποδεικνύει ότι και οι δύο τιμές προσφέρουν αποτελεσματική μάθηση χωρίς υπερπροσαρμογή. Το $\mu = 0.20$ προσφέρει πιο ομαλή σύγκλιση, ενώ το $\mu = 0.60$ επιτρέπει ταχύτερη σύγκλιση, αν και προκαλεί μεγαλύτερες διακυμάνσεις.
  - **Training Accuracy/Loss vs. Validation Accuracy/Loss:** Η ακρίβεια επικύρωσης και η απώλεια αυξάνεται/μειώνεται μέχρι το early stopping και μετά παραμένει οριακά σταθερή.

- **Learning Rate** $\eta = 0.1$**:** Ένας υψηλός ρυθμός εκμάθησης όπως $\eta = 0.1$ επιτρέπει στο μοντέλο να συγκλίνει γρήγορα μέσω μεγάλων ενημερώσεων βάρους. Ενώ αυτό έχει ως αποτέλεσμα την ταχεία μάθηση κατά τις πρώτες εποχές, οδηγεί σε *overfitting* και αστάθεια καθώς προχωρά το training, ιδιαίτερα όταν συνδυάζεται με υψηλότερη ορμή.

- **Momentum** $\mu = 0.20$ **ή** $\mu = 0.60$**:**

  - Και οι δύο τιμές οδηγούν σε γρήγορη σύγκλιση.
  - Το $\mu = 0.20$ επιτυγχάνει ελαφρώς καλύτερη απόδοση με **accuracy = 0.8377** και **CE Loss = 0.3810**.
  - Το $\mu = 0.60$ έχει **accuracy = 0.8365** και **CE Loss = 0.3872**, εισάγοντας περισσότερη ταλάντωση λόγω της υψηλότερης τιμής της μ.

- **Training Accuracy/Loss vs. Validation Accuracy/Loss:**

  - Η ακρίβεια του training set αυξάνεται απότομα και σταθεροποιείται νωρίς, δείχνοντας ότι το μοντέλο μαθαίνει γρήγορα τα training data.
  - Η ακρίβεια επικύρωσης παρουσιάζει διακυμάνσεις, υποδεικνύοντας πιθανή υπερπροσαρμογή.
  - Η CE loss μειώνεται γρήγορα κατά τις αρχικές εποχές, αλλά αρχίζει να αυξάνεται μετά από πρόωρη διακοπή.
  - Αυτά τα μοτίβα επιβεβαιώνουν ότι, ενώ το μοντέλο μαθαίνει γρήγορα, δυσκολεύεται να γενικεύσει αποτελεσματικά σε νέα δεδομένα.

Σύστημα Βαθμολόγησης: Για να αξιολογηθεί αντικειμενικά η ποιότητα κάθε συνδυασμού υπερπαραμέτρων, εφαρμόστηκε σύστημα βαθμολόγησης με βάση τις εξής μετρικές:

- **Ακρίβεια (βάρος: 0.4)**
- **Καμπύλη Επικύρωσης (βάρος: 0.4)** – Μικρότερο χάσμα overfitting δίνει υψηλότερη βαθμολογία.
- **Εποχές μέχρι σύγκλιση (βάρος: 0.2)** – Ταχύτερη σύγκλιση επιβραβεύεται.
- **Ποινές**: Αν το χάσμα υπερεκπαίδευσης υπερβεί το 0.1, εφαρμόζεται γραμμική ποινή. Αν υπερβεί το 0.3, το σκορ μηδενίζεται.

Συνολική Σύγκριση Συνδυασμών: Η παρακάτω σύνοψη παρουσιάζει τα σκορ κάθε συνδυασμού παραμέτρων:

- **LR=0.001, M=0.2:** Σκορ = 0.6403 – Πολύ χαμηλό overfitting, αλλά εξαιρετικά αργή εκπαίδευση.
- **LR=0.001, M=0.6:** Σκορ = 0.6582 – Βελτιωμένη ακρίβεια αλλά και πάλι αργή σύγκλιση.
- **LR=0.05, M=0.2: Σκορ = 0.6758 (Καλύτερο)** – Ισορροπία ανάμεσα σε ακρίβεια, overfitting και ταχύτητα.
- **LR=0.05, M=0.6:** Σκορ = 0.6670 – Γρήγορη εκπαίδευση, ελαφρώς χειρότερη ακρίβεια.
- **LR=0.1, M=0.2:** Σκορ = 0.6753 – Υψηλή ακρίβεια, ελαφρώς χειρότερη καμπύλη από το 0.05/0.2.
- **LR=0.1, M=0.6:** Σκορ = 0.6485 – Υψηλή ποινή λόγω υπερεκπαίδευσης.

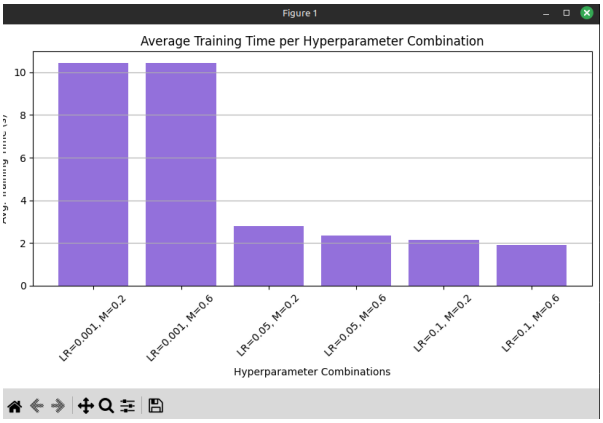

Figure 5: CE Loss, Accuracy vs Epochs

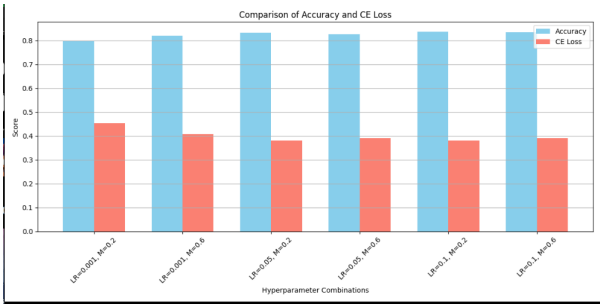

Figure 6: CE Loss, Accuracy vs Hidden Neurons

5.3.3 Συμπεράσματα

Ο συνδυασμός **learning rate = 0.05 και momentum = 0.2** αναδεικνύεται ως ο βέλτιστος, προσφέροντας την καλύτερη ισορροπία μεταξύ ακρίβειας, αποδοτικής εκπαίδευσης και ελαχιστοποίησης του φαινομένου της υπερεκπαίδευσης.

Table 2: Αποτελέσματα για διαφορετικές τιμές των η και m με standardization

η	m	CE Loss	MSE	Acc
0.001	0.20	0.4550	0.1452	0.7981
0.001	0.60	0.4074	0.1272	0.8208
0.050	0.20	0.3809	0.1175	0.8342
0.050	0.60	0.3907	0.1215	0.8266
0.100	0.20	0.3804	0.1173	0.8365
0.100	0.60	0.3906	0.1205	0.8348

6 Α4 : Ομαλοποίηση

Ο σκοπός του Α4 ερωτήματος είναι να εφαρμόσουμε μια μέθοδο ομαλοποίησης (regularization) στο νευρωνικό για να αποτρέψουμε το overfitting και να βελτιώσουμε την ικανότητά του να γενικεύει σε νέα δεδομένα. Αυτό σημαίνει ότι θέλουμε το δίκτυο να μην "μαθαίνει" υπερβολικά τα δεδομένα εκπαίδευσης, αλλά να είναι ικανό να αποδίδει καλά και σε νέα.

6.1 Overfitting

Η υπερπροσαρμογή συμβαίνει όταν ένα μοντέλο λειτουργεί καλά σε δεδομένα εκπαίδευσης αλλά αποτυγχάνει με νέα δεδομένα. Για να το εντοπίσουμε, χρησιμοποιούμε cross validation όπου,χωρίζει τα δεδομένα σε ένα σύνολο εκπαίδευσης και ένα σύνολο δοκιμών. Το μοντέλο εκπαιδεύεται στο σετ εκπαίδευσης και δοκιμάζεται στο σετ δοκιμής. Εάν υπάρχει μεγάλο χάσμα απόδοσης μεταξύ των δύο, το μοντέλο μπορεί να είναι υπερπροσαρμόζεται στα δεδομένα εκπαίδευσης

6.2 Regularization

Οι τεχνικές regularization βοηθούν στην αποφυγή του overfitting προσθέτοντας μία ποινή στην πολυπλοκότητα του μοντέλου (προσθέτοντας επιπλέον όρους στη loss function). Αυτό αποθαρρύνει το μοντέλο από το να προσαρμόζεται υπερβολικά στα δεδομένα εκπαίδευσης, βοηθώντας το να γενικεύει καλύτερα σε νέα.Παρακάτω αναλύονται οι δύο τεχνικές regularization .

6.2.1 L1 Lasso

Η L1 regularization χρησιμοποιείται όταν υπάρχουν πολλά χαρακτηριστικά (features) που μπορεί να είναι άσχετα. Προσθέτει μία ποινή στη loss function ίση με το άθροισμα των απόλυτων τιμών των βαρών του μοντέλου. Αυτό ενθαρρύνει το μοντέλο να μειώσει ορισμένα βάρη ακριβώς στο μηδέν, ουσιαστικά αφαιρώντας μη σημαντικά χαρακτηριστικά και καθιστώντας το μοντέλο πιο αραιό (sparse). Συνεπώς, η L1 regularization είναι χρήσιμη για αυτόματη επιλογή χαρακτηριστικών (feature selection). Ωστόσο, μπορεί να απορρίψει χρήσιμα χαρακτηριστικά όταν αυτά είναι highly correlated.

$$J(\theta) = \text{Loss Function} + \lambda \sum_{i=1}^{n} |\theta_i|$$

6.2.2 L2 Ridge

Η L2 Regularization μειώνει τα βάρη του μοντέλου προς το μηδέν, αλλά δεν τα μηδενίζει, σε αντίθεση με την L1 regularization. Βοηθά στην αποφυγή του overfitting επιβάλλοντας ποινή σε μεγάλα coefficients, καθιστώντας την χρήσιμη όταν όλα τα features θεωρούνται σημαντικά. Ωστόσο, είναι λιγότερο αποτελεσματική στη μείωση της πολυπλοκότητας του μοντέλου σε σχέση με την L1 και δεν είναι ανθεκτική σε outliers λόγω της τετραγωνικής ποινής στα βάρη. Η Ridge regularization ελέγχει τα μεγάλα βάρη, αλλά διατηρεί όλα τα features στο μοντέλο.

$$J(\theta) = \text{Loss Function} + \lambda \sum_{i=1}^{n} \theta_i^2$$

Όπου:

- $J(\theta)$: Η συνολική συνάρτηση κόστους(είναι ο μέσος όρος όλων των απωλειών στο σύνολο δεδομένων)
- Loss Function : Loss function (χρησιμοποιείται κατά τη διάρκεια της εκπαίδευσης για τον υπολογισμό του σφάλματος για μια πρόβλεψη)
- λ : Είναι regularization παράμετρος (υπερπαράμετρος που ελέγχει την ισχύ)
- θ_i : Παράμετροι μοντέλου (βάρη,bias)
- n : Αριθμός χαρακτηριστικών

6.2.3 Απαντήσεις

Στην περίπτωση πρόβλεψης της ασθένειας Alzheimer χρησιμοποιώντας ιατρικά δεδομένα, η L2 regularization είναι πιθανότατα η καλύτερη επιλογή. Δεδομένου ότι τα ιατρικά δεδομένα συνήθως περιλαμβάνουν πολλά χαρακτηριστικά, όπως βιοδείκτες, ιστορικό ασθενούς και άλλες κλινικές μετρήσεις, είναι σημαντικό να αποφεύγεται η απόρριψη οποιασδήποτε ενδεχομένως πολύτιμης πληροφορίας. Η L2 regularization βοηθά στην αποφυγή του overfitting επιβάλλοντας ποινή σε μεγάλα βάρη χωρίς να εξαλείφει τα χαρακτηριστικά εντελώς, διασφαλίζοντας ότι όλα τα δεδομένα λαμβάνονται υπόψη. Σε αντίθεση με τα δεδομένα εικόνας, όπου κάποια χαρακτηριστικά μπορεί να είναι άσχετα και να απορριφθούν (όπως στην περίπτωση αραιών δεδομένων), τα ακατέργαστα ιατρικά δεδομένα περιέχουν συνήθως πληθώρα πολύτιμων πληροφοριών που δεν πρέπει να αγνοηθούν. Επομένως, η L2 regularization παρέχει μια πιο κατάλληλη προσέγγιση για τη διατήρηση της ακεραιότητας όλων των χαρακτηριστικών, ενώ βελτιώνει την ικανότητα του μοντέλου να γενικεύει.Τέλος Η L2 στα κρυφά επίπεδα και στο επίπεδο εξόδου βοηθά στην αποφυγή overfitting, διατηρώντας τα χαρακτηριστικά πιο απλά και εμποδίζοντας το μοντέλο να απομνημονεύει τα δεδομένα εκπαίδευσης. Στο επίπεδο εξόδου, εξασφαλίζει ότι η τελική πρόβλεψη δεν είναι υπερβολικά ευαίσθητη στα δεδομένα εκπαίδευσης, βελτιώνοντας τη γενίκευση του μοντέλου.

6.3 Πειραματισμός για Διάφορες τιμές για Regularization Coeffients

6.3.1 Διαδικασία

Η διαδικασία tuning of regularization εστιάζει στην εύρεση του βέλτιστου coefficient (λ) εκτελώντας cross validation σε διάφορες τιμές κανονικοποίησης. Ο στόχος είναι να μειωθεί το overfitting και να βελτιωθεί η ικανότητα γενίκευσης του μοντέλου, διασφαλίζοντας την καλή απόδοση σε νεα δεδομένα με δεδομένα βέλτιστα learning rate,momentum και αριθμό κρυφό νευρώνων.

Γενική παρατήρηση: Μια γενική παρατήρηση από τα αποτελέσματα είναι ότι η $\lambda = 0.01$ προσφέρει την καλύτερη ισορροπία μεταξύ πρόληψης του overfitting, ακρίβειας (0.8604) και γρήγορης σύγκλισης. Επίσης, το $\lambda = 0.001$ επιτυγχάνει καλή απόδοση (0.8412), χωρίς να προκαλεί overfitting, αλλά η σύγκλιση είναι πιο αργή. Από την άλλη, για $\lambda = 0.0001$ παρατηρούμε ότι η ακρίβεια (0.8400) παραμένει υψηλή, αλλά με περισσότερη διακύμανση, που υποδεικνύει μια τάση προς overfitting.

6.3.2 Ανάλυση

Ο κώδικας εκτελεί cross validation για κάθε υποψήφια τιμή και αξιολογεί την απόδοση χρησιμοποιώντας ένα σύνθετο σκορ που λαμβάνει υπόψη την ακρίβεια στα δεδομένα επικύρωσης, τη συμπεριφορά υπερπροσαρμογής και την αποδοτικότητα σύγκλισης.Υπολογίζεται ένα αυστηρό πρόστιμο overfitting με βάση το τετράγωνο της διαφοράς μεταξύ της τελικής ακρίβειας στο training και στο validation σύνολο, ενώ εφαρμόζεται και ένα ήπιο πρόστιμο όταν αυτό το χάσμα ξεπερνά το όριο των 0.1.Το σύνθετο σκορ ενσωματώνει αυτό το πρόστιμο μαζί με τη συνολική ακρίβεια και τον αριθμό των εποχών που χρειάστηκε για να συγκλίνει το μοντέλο .Ρυθμίσεις που εμφανίζουν έντονη υπερπροσαρμογή (χάσμα > 0.3) απορρίπτονται αυτόματα όπως και στο A3. Η καλύτερη τιμή κανονικοποίησης είναι αυτή που πετυχαίνει το υψηλότερο σύνθετο σκορ, ισορροπώντας ανάμεσα στη γενίκευση, τη σταθερότητα του μοντέλου και την αποδοτικότητα της εκπαίδευσης .

- **Regularization $\lambda = 0.0001$:** Το μοντέλο επιτυγχάνει υψηλή ακρίβεια (0.8400), αλλά χρειάζεται ελαφρώς περισσότερες εποχές για να συγκλίνει. Το μικρό χάσμα υπερπροσαρμογής (0.0794)

υποδηλώνει μέτρια υπερπροσαρμογή. Το συνολικό σκορ των 0.6779 αντανακλά καλή απόδοση, αν και όχι την βέλτιστη. Αρχίζει με χαμηλή ακρίβεια, αλλά βελτιώνεται με την πάροδο του χρόνου, και στο τέλος παραμένει σχεδόν σταθερή, ενώ η απώλεια CE μειώνεται, αλλά υπάρχει περισσότερη διακύμανση, υποδεικνύοντας overfitting.

- **Regularization λ = 0.001:** Το μοντέλο πετυχαίνει τη δεύτερη υψηλότερη ακρίβεια (0.8412) και το υψηλότερο σκορ καμπύλης (0.3796), υποδεικνύοντας ελάχιστο overfitting. Συγκλίνει γρήγορα (σκορ εποχών = 0.0179) με μικρό χάσμα υπερπροσαρμογής (0.0237). Το συνολικό σκορ των 0.6868 αντανακλά την καλύτερη ισορροπία μεταξύ ακρίβειας, σύγκλισης και υπερπροσαρμογής. Η ακρίβεια αυξάνεται γρήγορα και στη συνέχεια σταθεροποιείται, με την απώλεια να μειώνεται σταθερά, δείχνοντας καλή γενίκευση.

- **Regularization λ = 0.01:** Το μοντέλο πετυχαίνει την υψηλότερη ακρίβεια (0.8604) και σκορ καμπύλης (0.3537), υποδεικνύοντας ότι το μοντέλο συγκλίνει με τον καλύτερο τρόπο χωρίς να υποφέρει από υπερπροσαρμογή. Συγκλίνει γρήγορα (σκορ εποχών = 0.0331) και το χάσμα υπερπροσαρμογής είναι πολύ μικρό (0.0177). Το συνολικό σκορ των 0.5964, αν και χαμηλότερο σε σχέση με άλλες περιπτώσεις, υποδεικνύει την καλύτερη γενίκευση του μοντέλου σε αυτό το επίπεδο κανονικοποίησης. Η ακρίβεια αυξάνεται σταθερά και παραμένει υψηλή, με την απώλεια να μειώνεται συνεχώς.

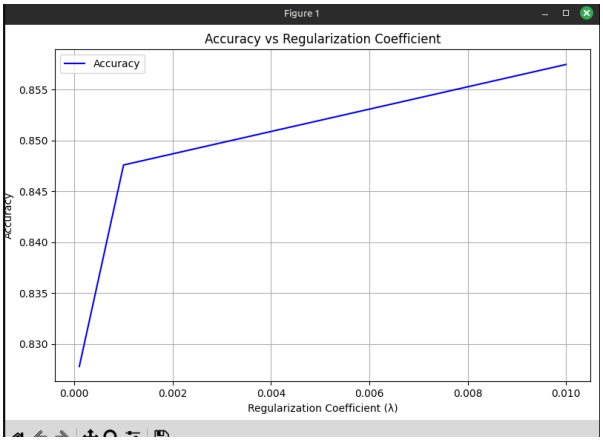

Figure 7: Accuracy vs λ

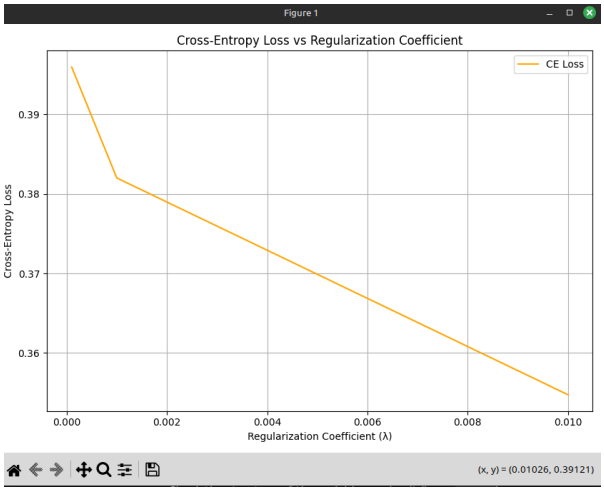

Figure 8: CE Loss vs λ

6.3.3 Συμπεράσματα

Βασισμένο στην ανάλυση των αποτελεσμάτων για διαφορετικές τιμές κανονικοποίησης, η καλύτερη απόδοση παρατηρήθηκε με συντελεστή κανονικοποίησης **0.01**. Αυτή η ρύθμιση πέτυχε την υψηλότερη ακρίβεια (0.8604), με χαμηλή απώλεια CE (0.3537), αλλά το συνολικό σκορ ήταν 0.5964, δείχνοντας την καλύτερη ισορροπία μεταξύ πρόληψης του overfitting και καλή απόδοση. Η τιμή λ = 0.001 πέτυχε την επόμενη καλύτερη ακρίβεια (0.8412) και συνολικό σκορ 0.6868, με μικρό χάσμα υπερπροσαρμογής, δείχνοντας γενίκευση.Η τιμή λ = 0.0001 πέτυχε χαμηλότερη ακρίβεια (0.8400) και μεγαλύτερη απώλεια CE (0.3831), υποδεικνύοντας τάση προς overfitting. Συνολικά, η τιμή λ = 0.01 φαίνεται να προσφέρει την καλύτερη ισορροπία μεταξύ ακρίβειας και γενίκευσης.

Table 3: Αποτελέσματα για διαφορετικές τιμές του συντελεστή λ

Συντελεστής λ	CE Loss	MSE	Acc
0.0001	0.3831	0.1188	0.8400
0.0010	0.3796	0.1162	0.8412
0.0100	0.3537	0.1068	0.8604

7 A5 : Βαθύ Νευρωνικό Δίκτυο

7.0.1 Διαδικασία

Το tuning των κρυφών επιπέδων και των νευρώνων σε ένα νευρωνικό γίνεται δοκιμάζοντας διαφορετικούς συνδυασμούς βάθους και μεγέθους για να βρούμε τι λειτουργεί καλύτερα για το μοντέλο. Σε αυτή τη διαδικασία, δοκιμάζουμε διάφορες ρυθμίσεις π.χ. διαφορετικό αριθμό επιπέδων (όπως 2 ή 3) και διαφορετικό αριθμό νευρώνων σε κάθε επίπεδο (συχνά με βάση το πόσα χαρακτηριστικά έχει η είσοδος). Για κάθε τέτοια ρύθμιση, εκπαιδεύουμε και αξιολογούμε το μοντέλο χρησιμοποιώντας cross-validation, για να δούμε πόσο καλά αποδίδει, με βάση κυρίως την ακρίβεια και το overfitting. Στο τέλος, επιλέγουμε τον συνδιασμό με την καλύτερη απόδοση. Αυτό μας βοηθά να φτιάξουμε ένα δίκτυο που να είναι σχετικά πολύπλοκο ώστε να μαθαίνει καλά, χωρίς είναι overfitted .

Γενική παρατήρηση: Από τα αποτελέσματα βλέπουμε ότι η χρήση 2 κρυφών επιπέδων γενικά αποδίδει καλύτερα από τη χρήση 3, με λίγο μεγαλύτερη ακρίβεια και λιγότερα σημάδια overfitting. Όσο αυξάνεται ο αριθμός των νευρώνων, η απόδοση βελτιώνεται μέχρι ένα σημείο — συγκεκριμένα, η καλύτερη ήταν 2 επίπεδα με 21 νευρώνες το καθένα. Πέρα από αυτό, η προσθήκη περισσότερων νευρώνων ή επιπέδων δεν βοήθησε ιδιαίτερα και μερικές φορές μείωσε την απόδοση, κάνοντας το μοντέλο πιο περίπλοκο χωρίς να βελτιώνεται η ακρίβεια.

7.0.2 Ανάλυση

Η ανάλυση των αποτελεσμάτων περιλαμβάνει την αξιολόγηση διαφορετικών συνδιασμών κρυφών στρωμάτων και νευρώνων σε ένα δίκτυο. Κάθε συνδιασμος δοκιμάζεται για την ακρίβειά της, τη σταθερότητα εκπαίδευσης (Curve Score) και την αποδοτικότητα όσον αφορά τις εποχές που απαιτούνται για τη σύγκλιση. Ένας βασικός παράγοντας είναι το συνολικό σκορ, το οποίο συνδυάζει αυτά τα κριτήρια και περιλαμβάνει ποινές για υπερπροσαρμογή (overfitting). Οι καλύτερες ρυθμίσεις είναι αυτές με μεγαλύτερη ακρίβεια και μικρότερη υπερπροσαρμογή, ενώ οι ρυθμίσεις με μεγαλύτερο χρόνο εκπαίδευσης ή μικρότερη σταθερότητα είναι λιγότερο ευνοϊκές. Συγκρίνοντας αυτά τα κριτήρια, εντοπίζουμε την καλύτερη αρχιτεκτονική δικτύο.

- **Hidden Neurons = [16, 16], Layers = 2 :** Αυτός ο συνδυασμός, με 2 επίπεδα και 16 κρυφούς νευρώνες σε κάθε στρώμα, πέτυχε καλή ακρίβεια 85.05%, υποδεικνύοντας ισχυρή απόδοση του μοντέλου. Εμφάνισε σταθερή εκπαίδευση με σκορ καμπύλης 0.8534 και συγκλίνε γρήγορα, απαιτώντας μόνο 0.0178 σκορ εποχών για να εκπαιδευτεί αποδοτικά. Το μοντέλο δεν υπερπροσαρμόστηκε, όπως φαίνεται από την ποινή overfitting μηδέν και τη μικρή διαφορά 0.0419 μεταξύ της απόδοσης εκπαίδευσης και επικύρωσης,δείχνοντας έναν καλό συνδυασμό μεταξύ ακρίβειας και αποδοτικότητας εκπαίδευσης.

- **Hidden Neurons = [16, 16, 16], Layers = 3 :** Αυτός ο συνδυασμός πέτυχε ακρίβεια 85.28%, λίγο υψηλότερη από το προηγούμενο μοντέλο, με σταθερό σκορ καμπύλης 0.8485,

υποδεικνύοντας συνεπή απόδοση κατά τη διάρκεια της εκπαίδευσης. Το σκορ των εποχών αυξήθηκε σε 0.0207, υποδεικνύοντας ότι χρειάστηκε λίγο περισσότερο χρόνο για να συγκλίνει. Παρόλο που η ποινή overfitting παραμένει μηδέν, η διαφορά αυξήθηκε σε 0.0564, υποδεικνύοντας ελαφρώς μεγαλύτερο κίνδυνο υπερπροσαρμογής σε σχέση με το μοντέλο των 2 επίπεδα. Το συνολικό σκορ μειώθηκε ελαφρώς σε 0.6847, λίγο κάτω από το προηγούμενο καλύτερο σκορ των 0.6851.

- **Hidden Neurons = [21, 21, 21], Layers = 3 :** Αυτός ο συνδυασμός πέτυχε ακρίβεια 85.51%, ελαφρώς καλύτερη από τα προηγούμενα μοντέλα με 16 νευρώνες, αν και η διαφορά είναι μικρή. Το σκορ καμπύλης παρέμεινε σταθερό στο 0.8467, και το σκορ των εποχών ήταν χαμηλό στο 0.0182, υποδεικνύοντας γρήγορη εκπαίδευση. Η ποινή overfitting παρέμεινε μηδενική, με μια μικρή διαφορά 0.0563 μεταξύ της απόδοσης εκπαίδευσης και επικύρωσης. Το συνολικό σκορ 0.6844 είναι ελαφρώς χαμηλότερο από το καλύτερο μοντέλο των 16 νευρώνων και 2 επίπεδα.

- **Hidden Neurons = [32, 32], Layers = 2 :** Αυτός ο συνδυασμός πέτυχε ακρίβεια 85.40%, ελαφρώς χαμηλότερη από το μοντέλο με 2 επίπεδα και 21 νευρώνες. Το σκορ καμπύλης έπεσε στο 0.8438, δείχνοντας μια μικρή μείωση στη σταθερότητα κατά τη διάρκεια της εκπαίδευσης. Το σκορ των εποχών αυξήθηκε στο 0.0213, υποδεικνύοντας ότι χρειάστηκε λίγο περισσότερο χρόνο για να εκπαιδευτεί. Παρόλο που η ποινή overfitting παρέμεινε μηδενική, η διαφορά αυξήθηκε σε 0.0663, υποδεικνύοντας ελαφρώς μεγαλύτερο κίνδυνο υπερπροσαρμογής. Το συνολικό σκορ 0.6834 είναι χαμηλότερο από το καλύτερο μοντέλο με 2 επίπεδα και 21 νευρώνες.

- **Hidden Neurons = [32, 32, 32], Layers = 3 :** Αυτός ο συνδυασμός πέτυχε ακρίβεια 85.11%, ελαφρώς χαμηλότερη από τα μοντέλα με 16 και 21 νευρώνες. Το σκορ καμπύλης παραμένει ικανοποιητικό στο 0.8452, αν και όχι τόσο υψηλό όσο τα προηγούμενα μοντέλα, ενώ το σκορ των εποχών είναι το χαμηλότερο μέχρι στιγμής, στο 0.0171, υποδεικνύοντας γρήγορη σύγκλιση. Η ποινή υπερπροσαρμογής παραμένει μηδενική, με μια μικρή διαφορά 0.0646, δείχνοντας ελάχιστη overfittinh. Ωστόσο, το συνολικό σκορ 0.6819 είναι χαμηλότερο από το καλύτερο σκορ των 0.6851, πράγμα που σημαίνει ότι αυτός ο συνδυασμός αποδίδει χειρότερα συνολικά.

- **Hidden Neurons = [64, 64], Layers = 2 :** Αυτός ο συνδυασμός πέτυχε ακρίβεια 84.64%, ελαφρώς καλύτερη από πριν, αλλά το σκορ καμπύλης έπεσε στο 0.8328, δείχνοντας προβλήματα σταθερότητας κατά τη διάρκεια της εκπαίδευσης. Το σκορ των εποχών αυξήθηκε στο 0.0220, πράγμα που σημαίνει ότι το μοντέλο χρειάστηκε περισσότερο χρόνο για να εκπαιδευτεί. Παρόλο που η ποινή overfitting παραμένει μηδενική, η διαφορά αυξήθηκε σε 0.0848, υποδεικνύοντας υψηλότερο κίνδυνο overfitting. Με συνολικό σκορ 0.6761, αυτή είναι η χειρότερη απόδοση μέχρι στιγμής.

- **Hidden Neurons = [64, 64, 64], Layers = 3 :** Αυτός ο συνδυασμός πέτυχε τη χαμηλότερη ακρίβεια μέχρι στιγμής, 84.06%, με περαιτέρω πτώση στο σκορ καμπύλης στο 0.8188, υποδεικνύοντας κακή σταθερότητα εκπαίδευσης. Το σκορ των εποχών, 0.0200, δείχνει ότι χρειάστηκε λίγο περισσότερο χρόνο για να συγκλίνει. Η ποινή υπερπροσαρμογής αυξήθηκε σε 0.0243, με μεγαλύτερη διαφορά 0.1217, δείχνοντας σημαντική overfitting. Με συνολικό σκορ 0.6434, αυτό είναι το μοντέλο με την χειρότερη απόδοση.

7.0.3 Συμπέρασμα

Από τα πειράματα που πραγματοποιήθηκαν, το βέλτιστο μοντέλο ήταν αυτό με δύο κρυφά επίπεδα και 16 νευρώνες ανά επίπεδο, το οποίο πέτυχε το υψηλότερο συνολικό σκορ 0.6851. Η αύξηση των επιπέδων ή των νευρώνων δεν είχε ουσιαστικά θετική επίδραση στην απόδοση και, σε ορισμένες περιπτώσεις, οδήγησε σε μειωμένη σταθερότητα και χαμηλότερη ακρίβεια. Παρόλο που το overfitting ήταν ελάχιστο σε αρκετές περιπτώσεις, τα μοντέλα με τρία επίπεδα και περισσότερους νευρώνες παρουσίασαν αυξημένο κίνδυνο υπερπροσαρμογής.

Η γενική απόδοση του καλύτερου μοντέλου ήταν εξαιρετική, καθώς παρουσίασε καλή ισορροπία στην ακρίβεια, την ευαισθησία, την ανάκληση και το F1 score. Σε αντίθεση, το μοντέλο με 1 κρυφό επίπεδο και 64 νευρώνες, αν και είχε καλύτερη ακρίβεια, δυσκολευόταν να γενικεύσει και να καταγράψει τα θετικά περιστατικά με την ίδια αποδοτικότητα. Αυτό το μοντέλο εμφάνισε υψηλότερη ακρίβεια, αλλά θυσίασε την ανάκληση, χάνοντας κάποια σημαντικά περιστατικά. Η υπερπροσαρμογή του μεγαλύτερου μοντέλου είχε ως αποτέλεσμα ελαφρώς χαμηλότερη απόδοση σε δεδομένα που δεν είχε συναντήσει κατά τη διάρκεια της εκπαίδευσης.

Όσον αφορά τη διάταξη των κρυφών στρωμάτων, συχνά αποδεικνύεται αποτελεσματικό να αρχίζουμε με περισσότερους νευρώνες στο πρώτο στρώμα και να μειώνουμε τον αριθμό τους στα επόμενα επίπεδα. Αυτή η προσέγγιση επιτρέπει στο μοντέλο να αποτυπώσει τα γενικά χαρακτηριστικά των δεδομένων και στη συνέχεια να τα βελτιώσει στα κατώτερα επίπεδα. Ωστόσο, ο αριθμός των στρωμάτων και των νευρώνων πρέπει να προσαρμόζεται στην πολυπλοκότητα των δεδομένων. Ενώ η προσθήκη περισσότερων στρωμάτων μπορεί να βελτιώσει την απόδοση μέχρι ένα σημείο, η υπερβολική αύξηση της πολυπλοκότητας μπορεί να οδηγήσει σε υπερπροσαρμογή και να μειώσει τη γενίκευση. Συνολικά, τα απλά μοντέλα φαίνεται να αποδίδουν καλύτερα σε λιγότερο πολύπλοκα δεδομένα, καθώς η υπερβολική πολυπλοκότητα μπορεί να περιορίσει την απόδοση.

Hidden Neurons	Hidden Layers	CE Loss	MSE	Accuracy
16	2	0.351338	0.106501	0.850493
16	3	0.353616	0.107298	0.852826
21	2	0.352798	0.106803	0.855143
21	3	0.352525	0.106220	0.853988
32	2	0.348108	0.105523	0.851076
32	3	0.367718	0.113005	0.846420
64	2	0.359495	0.109514	0.847003
64	3	0.375543	0.114387	0.840610

Table 4: Απόδοση μοντέλου για διαφορετικούς αριθμούς νευρώνων και κρυφών επιπέδων

8 Βιβλιογραφία

References

[1] Tomasz Szandała, *Review and Comparison of Commonly Used Activation Functions for Deep Neural Networks*, Διαθέσιμο εδώ, 2021.

[2] Chigozie E. Nwankpa,Winifred l.Ijomah ,Anthony, Gachagan ,Stephen Marshal , *Activation Functions: Comparison of Trends in Practice and Research for Deep Learning*, Διαθέσιμο εδώ, 2020.

[3] Houmem Slimi,Ala Balti ,Sabeur Abid , Mounir Sayadi , *A combinatorial deep learning method for Alzheimer's disease classification-based merging pretrained networks*, Διαθέσιμο εδώ, 2024. Houmem Slimi,Ala Balti ,Sabeur Abid , Mounir Sayadi