1Η εργασία του μαθήματος «Νευρωνικά Δίκτυα»

Χρυσολόγου Γεώργιος (ΑΕΜ: 10782)

Για την εκπόνηση της παρούσας εργασίας, κλήθηκα να υλοποιήσω ένα feedforward νευρωνικό δίκτυο το οποίο να εκπαιδεύεται με τον αλγόριθμο backpropagation, για να επιλύει προβλήματα κατηγοριοποίησης πολλών κλάσεων. Επέλεξα να κατασκευάσω ένα MLP δίκτυο ,χωρίς την χρήση των σχετικών frameworks (pytorch, keras κλπ), σε γλώσσα python, το οποίο εκπαίδευσα με χρήση του data set CIFAR-10.

Περιγραφή αλγορίθμου backpropagation: Ο αλγόριθμος αυτός χρησιμοποιείται για την επαναλαμβανόμενη ανανέωση των τιμών των βαρών και των biases κάθε νευρώνα του δικτύου. Οι τιμές αυτές τροποποιούνται κατάλληλα, με σκοπό την μείωση της τιμής του loss function του δικτύου και κατά συνέπεια την αύξηση του ποσοστού επιτυχίας του δικτύου στην αναγνώριση της κλάσης των δειγμάτων ελέγχου. Το κάθε βάρος και το κάθε bias ανανεώνονται σε κάθε batch δειγμάτων εκπαίδευσης με βάση την τιμή του loss function. Η τιμή αυτή δείχνει το ποσό του λάθους των τελικών προβλέψεων του δικτύου κατά την προσπάθεια εύρεσης της κλάσης κάθε δείγματος εκπαίδευσης. Καθώς τα βάρη και τα biases υπόκεινται σε αλλαγές σύμφωνα με τον backpropagation, το δίκτυο πραγματοποιεί ολοένα και ευστοχότερες προβλέψεις. Στόχος μας είναι η αύξηση του ποσοστού επιτυχίας στο validation test με το πέρας κάθε εποχής, η οποία ολοκληρώνεται κάθε φορά που το δίκτυο έχει δεχθεί ως είσοδο όλα τα δείγματα ελέγχου σε batches. Οι εξισώσεις που περιγράφουν τις αλλαγές των βαρών και των biases κάθε στρώματος, συμφώνα με τον backpropagation, είναι οι εξής:

$$W_{k+1}^{(l)} = W_k^{(l)} - \beta * \frac{\partial J}{\partial W^{(l)}}$$

$$b_{k+1}^{(l)} = b_k^{(l)} - \beta * \frac{\partial J}{\partial b^{(l)}}$$

όπου J η συνάρτηση απώλειας (loss function) και β το βήμα εκμάθησης (learning rate).

Ο δείκτης l , ο οποίος παίρνει ακέραιες τιμές από 1 εώς L, υποδεικνύει το στρώμα το οποίο αφορούν οι εξισώσεις(layer 1 θεωρείται το input layers και layer L το outer layer). Το W παραπάνω αποτελεί πίνακα διαστάσεων (πλήθος νευρώνων) x (πλήθος εισόδων) (όπου το πλήθος εισόδων είναι ίσο με το πλήθος νευρώνων του προηγούμενου layer ή το πλήθος των εισόδων του δικτύου αν πρόκειται για το input layer) με στοιχεία τις τιμές των βαρών που ενώνουν όλους τους νευρώνες του layer $l-1$ με όλους τους νευρώνες του layer l. Αντίστοιχα, το b παραπάνω αποτελεί διάνυσμα με στοιχεία τις τιμές των biases των νευρώνων του layer l.

Θεώρησα καταλληλότερη επιλογή για το πρόβλημα κατηγοριοποίησης την cross-entropy loss function, σε συνδυασμό με την softmax συνάρτηση ενεργοποίησης για το outer layer του δικτύου μου. Με βάση τις επιλογές αυτές, οι εξισώσεις για την ανανέωση των βαρών και των biases του outer layer, που αποτελείται από 10 νευρώνες και συμβολίζεται με L, με βάση τον backpropagation είναι οι εξής:

1) $J = -\sum_{i=1}^{N} d_i * \log(y_i)$, όπου N ο αριθμός των κλάσεων, d_i ίσο με 1 αν το δείγμα ανήκει στην κλάση I και 0 διαφορετικά, y_i τιμή στο διάστημα [0,1] (λόγω χρήσης της softmax) η οποία δείχνει την πιθανότητα το δείγμα να ανήκει στην κλάση i.

2) $\delta^{(L)} = y^{(L)} - d^{(L)}$, όπου δ το σφάλμα εξόδου, y το διάνυσμα εξόδου μετά την εφαρμογή της softmax και d το διάνυσμα στόχου.

3) $\frac{\partial J}{\partial W^{(L)}} = \delta^{(L)} * (a^{(L)})^T$, όπου a το διάνυσμα εισόδου του outer layer, το οποίο είναι το αντίστοιχο διάνυσμα εξόδου του προηγούμενου layer.

4) $\frac{\partial J}{\partial b^{(L)}} = \delta^{(L)}$

Συνεπώς, οι εξισώσεις ανανέωσης των βαρών και των biases διαμορφώνονται ως εξής:

$$W_{k+1}^{(L)} = W_k^{(L)} - \beta * \delta^{(L)} * (a^{(L)})^T$$

$$b_{k+1}^{(L)} = b_k^{(L)} - \beta * \delta^{(L)}$$

Όσον αφορά τα hidden layers, η τελική μορφή των εξισώσεων εξαρτάται από την επιλογή της συνάρτησης ενεργοποίησης (πραγματοποίησα δοκιμές με διαφορετικές) . Για μία τυχαία συνάρτηση ενεργοποίησης $\sigma(z)$, το σφάλμα ενός hidden layer l είναι το εξής:

$\delta^{(l)} = \left(W^{(l+1)}\right)^T * \delta^{(l+1)} * \sigma'(z^{(l)})$, όπου $z^{(l)}$ η έξοδος του layer l προτού εισαχθεί στην συνάρτηση ενεργοποίησης.

Οι εξισώσεις ανανέωσης των βαρών και των biases των hidden layers είναι ίδιες με αυτές του outer layer με την παραπάνω διαφορά του σφάλματος.

Περιγραφη κώδικα: Ο κώδικας μου, αρχικά, αποθηκεύει τα 50000 δεδομένα εκπαίδευσης του σε έναν πίνακα 50000 x 3072 με όνομα data_tr, όπου κάθε γραμμή αποτελεί ένα δείγμα εκπαίδευσης. Αντίστοιχα, αποθηκεύει τα 10000 δείγματα ελέγχου σε έναν πίνακα 10000 x 3072 με όνομα data_test, όπου κάθε γραμμή αποτελεί ένα δείγμα ελέγχου.

Κατά την διάρκεια της φάσης εκπαίδευσης, το πρόγραμμα δημιουργεί batches πολλών δειγμάτων (δοκίμασα διαφορετικά batch sizes) τα οποία εισάγονται στο δίκτυο. Σε κάθε εποχή, δημιουργούνται συνολικά 50000 / batch size, σε αριθμό, batches τα οποία αποτελούνται από ανακατεμένα δείγματα εκπαίδευσης, ώστε το δίκτυο να εκπαιδεύεται κάθε φορά με διαφορετικό συνδυασμό δειγμάτων και να αποφεύγεται η ‘’απομνημόνευση’’ τους.

Στην συνέχεια, δημιουργείται το νευρωνικό δίκτυο με αριθμό hidden layers και πλήθος νευρώνων στο κάθε layer που μπορούν εύκολα να τροποποιηθούν. Τα biases όλων των

νευρώνων αρχικοποιούνται στην τιμή 0. Όσον αφορά τα βάρη, για το outer layer επέλεξα την αρχικοποίηση Xavier, η οποία είναι κατάλληλη για την softmax συνάρτηση ενεργοποίησης, ενώ για τα hidden layers χρησιμοποίησα διαφορετικές, ανάλογα με την επιλογή της συνάρτησης ενεργοποίησης. Επέλεξα να μην συμπεριλάβω κάποιο input layer, αλλά το κάθε batch να αποτελεί είσοδο στο 1ο hidden layer.

Κατά την φάση εμπρόσθιας τροφοδότησης, το batch δειγμάτων, το οποίο είναι ένας πίνακας διαστάσεων batch size x 3072 με κάθε γραμμή του πίνακα να αντιστοιχεί σε ένα δείγμα, μέσω συναπτικών βαρών εισέρχεται στο 1ο hidden layer. Εκεί υπολογίζεται η έξοδος του layer, σύμφωνα με την εξίσωση

$z^{(1)} = (w^{(1)})^T * x + b^{(1)}$, όπου x είναι το batch των δειγμάτων.

Κατόπιν, υπολογίζεται η τελική ‘’ενεργοποιημένη’’ έξοδος του layer, μετά την είσοδο της στην συνάρτηση ενεργοποίησης, ως $y^{(1)} = \sigma(z^{(1)})$. Η έξοδος αυτή, η οποία είναι ένας πίνακας διαστάσεων (batch size) x (πλήθος νευρώνων 1ο hidden layer) αποτελεί είσοδο, μέσω των αντίστοιχων συναπτικών βαρών, για το 2ο hidden layer. Η έξοδος του νέου layer υπολογίζεται, παρόμοια με προηγουμένως, ως $y^{(2)} = \sigma\left((w^{(2)})^T * y^{(1)} + b^{(2)}\right)$ (όπου $w^{(2)}$ τα συναπτικά βάρη που ενώνουν το 1ο με το 2ο hidden layer) και αποτελεί είσοδο για το επόμενο layer. Η διαδικασία αυτή επαναλαμβάνεται για όλα τα hidden layers και το outer layer. Τελικά, η έξοδος του outer layer είναι ένας πίνακας διαστάσεων (batch size) x 10, στην οποία έχει εφαρμοστεί η συνάρτηση ενεργοποίησης softmax, και κάθε γραμμή m του πίνακα αποτελείται από 10 στοιχεία που δείχνουν την πιθανότητα το δείγμα m του batch να ανήκει σε καθεμία από τις 10 κλάσεις.

Στην συνέχεια υπολογίζεται η τιμή της cross-entropy loss function για το συνολικό batch ως ο μέσος όρος των τιμών της cross-entropy loss function για το κάθε δείγμα του batch, και ακολουθεί η φάση οπισθοδρόμησης. Κατά την φάση αυτή, ανανεώνονται τα βάρη και τα biases όλων των layers ξεκινώντας από το outer layer και σταδιακά καταλήγοντας στο 1ο hidden layer. Οι ανανεώσεις γίνονται με βάση τις εξισώσεις που ορίζει ο αλγόριθμος backpropagation, οι οποίες αναλύθηκαν παραπάνω. Στο τέλος κάθε εποχής, υπολογίζεται το loss της εποχής ως ο μέσος όρος της τιμής της cross-entropy loss function όλων των batches, καθώς και το training accuracy, το οποίο ορίζεται ως ο συνολικός αριθμός των σωστών προβλέψεων του δικτύου όσον αφορά το label κάθε δείγματος εκπαίδευσης προς τον συνολικό αριθμό των δειγμάτων εκπαίδευσης. Τέλος, πριν την έναρξη της επόμενης εποχής, πραγματοποιείται ένα validation test όπου εισάγονται ατομικά (όχι ως batches) τα 10000 δείγματα ελέγχου του data set και υπολογίζεται το test accuracy του δικτύου για την συγκεκριμένη εποχή, το οποίο ορίζεται παρόμοια με το training accuracy.

Παραδείγματα ορθής και εσφαλμένης κατηγοριοποίησης:

Τα παραδείγματα αυτά αντλήθηκαν από εξόδους του δικτύου κατά την διάρκεια εξέτασης ενός batch δειγμάτων εκπαίδευσης στην 150η εποχή της φάσης εκπαίδευσης.

Ορθή κατηγοριοποίηση:

```
Εξοδος του outer layer:
[0.0693248  0.00921437 0.06624738 0.482677   0.00661784 0.31027456
 0.00198383 0.01026698 0.02590866 0.01748458]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 4η κλάση
Το δείγμα ανήκει στην 4η κλάση
```

```
Εξοδος του outer layer:
[0.23086006 0.00264164 0.14410582 0.08604826 0.15751232 0.0454161
 0.04321065 0.00367838 0.28196099 0.00456577]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 9η κλάση
Το δείγμα ανήκει στην 9η κλάση
```

```
Εξοδος του outer layer:
[0.04596741 0.01362827 0.04364344 0.01999217 0.7132095  0.01566914
 0.02959285 0.10143853 0.00903831 0.00782038]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 5η κλάση
Το δείγμα ανήκει στην 5η κλάση
```

Παρατηρείται ότι στο 1ο παράδειγμα οι πιθανότητες που αντιστοιχούν στην 4η και στην 6η κλάση είναι πολύ υψηλότερες από τις υπόλοιπες, αλλά το δίκτυο καταλήγει στην σωστή από τις δύο. Στο 2ο παράδειγμα, υπάρχουν αρκετές κλάσεις των οποίων οι πιθανότητες έχουν σχετικά παρόμοιες τιμές. Στο 3ο παράδειγμα, η πιθανότητα που αντιστοιχεί στην 5η κλάση, η οποία είναι και η σωστή, είναι κατά πολύ μεγαλύτερη από όλες τις υπόλοιπες. Οι παρατηρήσεις αυτές ενδεχομένως οφείλονται στο γεγονός ότι κάποια δείγματα μιας κλάσης έχουν χαρακτηριστικά που τα ξεχωρίζουν κατά πολύ από τις υπόλοιπες κλάσεις, άλλα δείγματα φέρουν χαρακτηριστικά που τα καθιστούν παρόμοια με δείγματα μίας άλλης κλάσης ενώ ,τέλος, κάποια άλλα δείγματα φέρουν χαρακτηριστικά τα οποία δεν επιτρέπουν την κατηγοριοποίηση τους σε μία συγκεκριμένη κλάση με μεγάλη πιθανότητα σε σχέση με τις υπόλοιπες.(Στο data set CIFAR-10 τα χαρακτηριστικά αυτά αποτελούν οι τιμές του έντασης του κόκκινου, του πράσινου και του μπλε σε κάθε pixel μιας εικόνας-δείγματος)

Εσφαλμένη κατηγοριοποίηση:

```
Εξοδος του outer layer:
[0.27385826 0.15544748 0.01958427 0.03216589 0.0280161  0.02176467
 0.00080706 0.00675133 0.42439484 0.03721011]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 9η κλάση
Το δείγμα ανήκει στην 1η κλάση
```

```
Εξοδος του outer layer:
[0.17817463 0.03612842 0.00511423 0.02859538 0.01258344 0.00970643
 0.00233414 0.01516576 0.29597924 0.41621834]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 10η κλάση
Το δείγμα ανήκει στην 9η κλάση
```

```
Εξοδος του outer layer:
[0.00887246 0.00105367 0.19829151 0.19502375 0.1396971  0.226117
 0.19023431 0.03239299 0.00536755 0.00294965]
Το δίκτυο προβλέπει ότι το δείγμα ανήκει στην 6η κλάση
Το δείγμα ανήκει στην 4η κλάση
```

Με αντίστοιχο τρόπο, παρατηρείται ότι και στις περιπτώσεις εσφαλμένης κατηγοριοποίησης, ορισμένες φορές το δίκτυο καταλήγει σε λάθος πρόβλεψη, με την πιθανότητα της λάθος κλάσης-πρόβλεψης να είναι πολύ υψηλότερη από τις πιθανότητες των υπολοίπων κλάσεων, άλλες φορές η πρόβλεψη του περιορίζεται ανάμεσα σε δύο κλάσεις και τελικά επιλέγει την λάθος, ενώ υπάρχουν και περιπτώσεις στις οποίες οι τιμές των πιθανοτήτων που αντιστοιχούν σε πολλές κλάσεις είναι παρόμοιες.

Πραγματοποίησα πληθώρα πειραμάτων, μεταβάλλοντας τις διάφορες παραμέτρους του νευρωνικού δικτύου. Συγκεκριμένα, δοκίμασα διαφορετικό αριθμό hidden layers, διαφορετικό αριθμό νευρώνων σε κάθε hidden layer, διαφορετικά είδη και τιμές του learning rate, διαφορετικά batch sizes καθώς και διαφορετικές συναρτήσεις ενεργοποίησης στα hidden layers. Τα δεδομένα εκπαίδευσης και ελέγχου κανονικοποιήθηκαν για μεγαλύτερη ακρίβεια στα αποτελέσματα. Παρακάτω περιγράφονται και σχολιάζονται 15 από τα πειράματα αυτά.

Στα πειράματα 1-5 χρησιμοποιείται η sigmoid συνάρτηση ενεργοποίησης και τα βάρη των hidden layers αρχικοποιούνται με βάση την αρχικοποίηση Xavier. Στα πειράματα 6-10 χρησιμοποιούνται η ReLu και η αρχικοποίηση He αντίστοιχα. Τέλος, τα πειράματα 11-15 πραγματοποιούνται με χρήση της Leaky ReLu (με παράγοντα διαρροής alpha = 0.01), η οποία είναι μία παραλλαγή της ReLu και επιλέχθηκε με σκοπό την αντιμετώπιση του προβλήματος των ‘’νεκρών’’ νευρώνων που πιθανόν να προκύψει από την χρήση της ReLu, και αρχικοποίηση He.

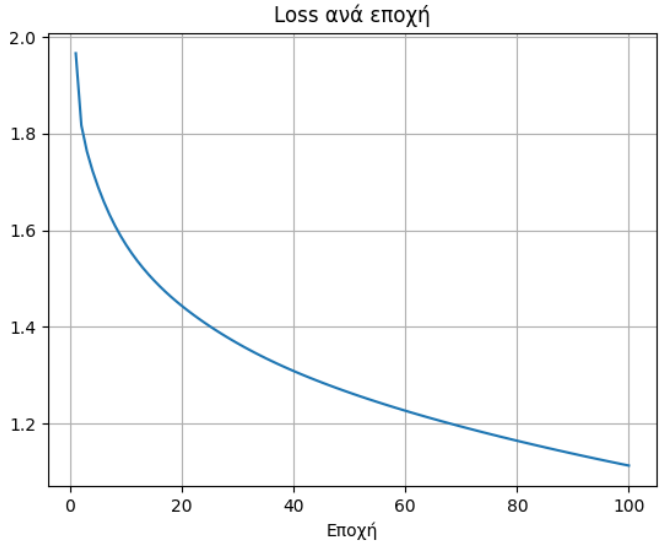
Τα δύο διαγράμματα του κάθε πειράματος αφορούν την καμπύλη της τιμής του loss function όλων των δειγμάτων εκπαίδευσης ανά εποχή καθώς και τις καμπύλες των training accuracy (πορτοκαλί) και test accuracy (μπλε) ανά εποχή.

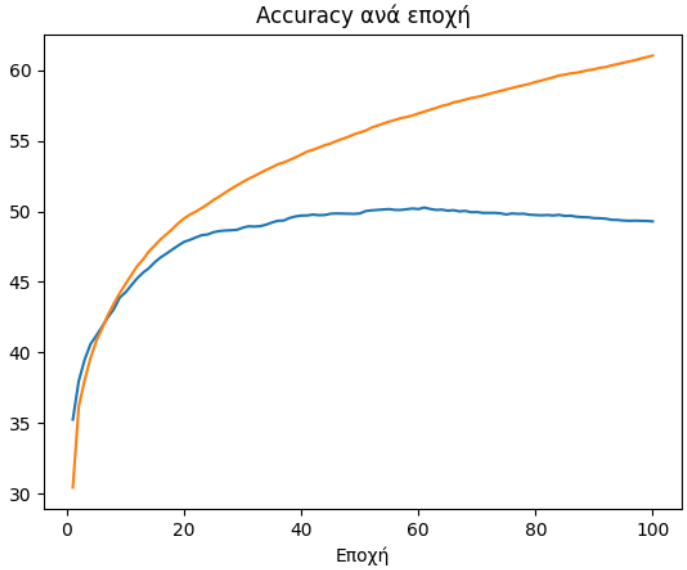
Μετά την περιγραφή κάθε πειράματος και την παράθεση των αποτελεσμάτων, ακολουθεί σχολιασμός και σύγκριση με προηγούμενα πειράματα.

Σημαντική σημείωση: Δεν μπορούμε να οδηγηθούμε σε απολύτως ασφαλή συμπεράσματα όσον αφορά την σύγκριση της ταχύτητας των διαφορετικών αρχιτεκτονικών και παραμέτρων του δικτύου, διότι ο χρόνος εκπαίδευσης τους επηρεάζεται και από άλλους παράγοντες που σχετίζονται με τον υπολογιστή. Για παράδειγμα, το ίδιο ακριβώς πείραμα απαιτεί διαφορετικούς χρόνους όταν εκτελείται δύο φορές.

Πείραμα 1:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
1	64	Σταθερό 0.05	64	sigmoid	100





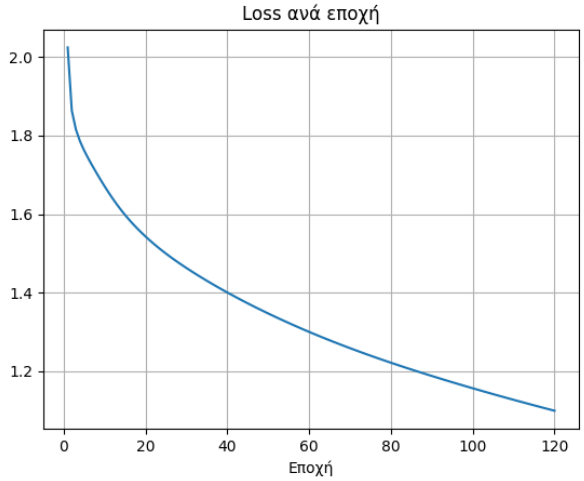
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.11	61.03	61.03	50.27	49.29	7.93

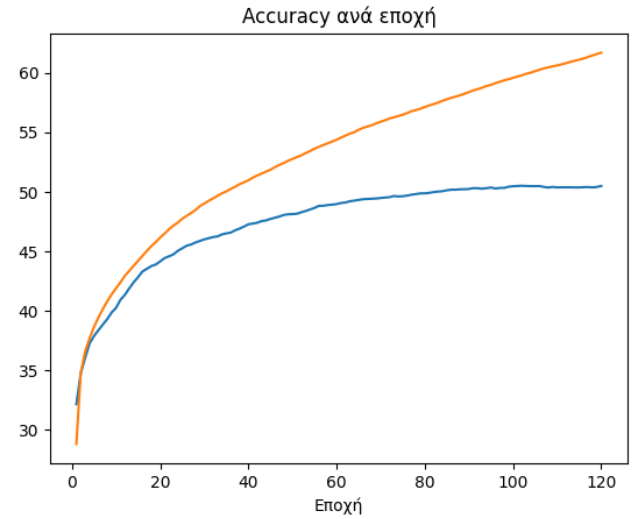
Παρατηρείται ότι η τιμή του loss συνεχώς μειώνεται, η τιμή του training accuracy συνεχώς αυξάνεται, αλλά η τιμή του test accuracy μετά το πέρας της 60^{ης} εποχής παραμένει

στάσιμη και τελικά αρχίζει να μειώνεται. Σημειώνεται ,επομένως, το φαινόμενο του underfitting, γεγονός λογικό δεδομένου ότι χρησιμοποιείται μόνο ένα hidden layer με λίγους νευρώνες.

Πείραμα 2:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
1	128	Σταθερό 0.05	128	sigmoid	120





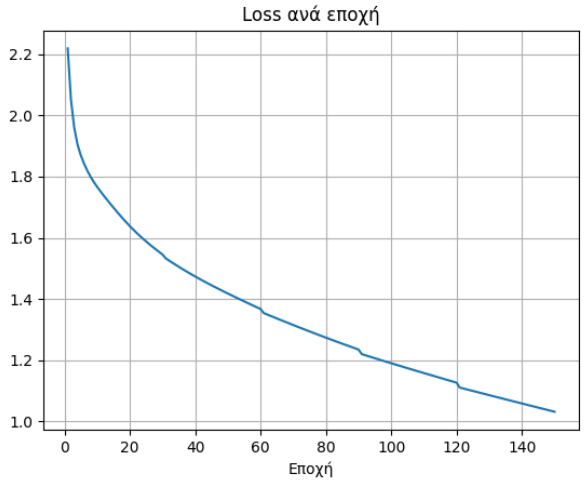
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.15	61.7	61.7	50.52	50.5	7.64

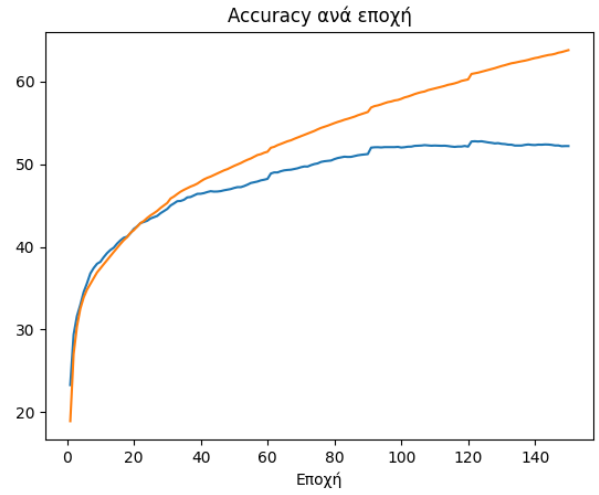
Το πείραμα αυτό, σε σχέση με το πείραμα 1 , διαφέρει μόνο ως προς τον αριθμό των νευρώνων του hidden layer και το batch size, τα οποία έχουν διπλασιαστεί. Παρατηρούμε ότι η απόδοση του δικτύου αυτού είναι ελαφρώς υψηλότερη σε σχέση με αυτή του

προηγούμενου πειράματος, ενώ δεν παρατηρείται τάση μείωσης του test accuracy μετά από κάποια εποχή.

Πείραμα 3:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
2	256,128	0.05 με πολ/σμο επί 0.9 κάθε 30 εποχές	128	sigmoid	150



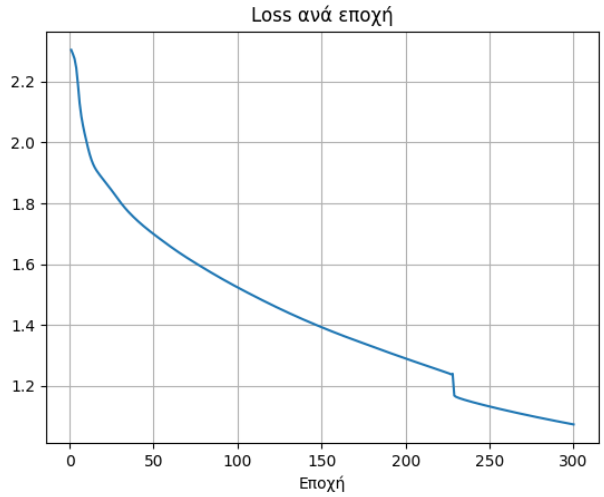


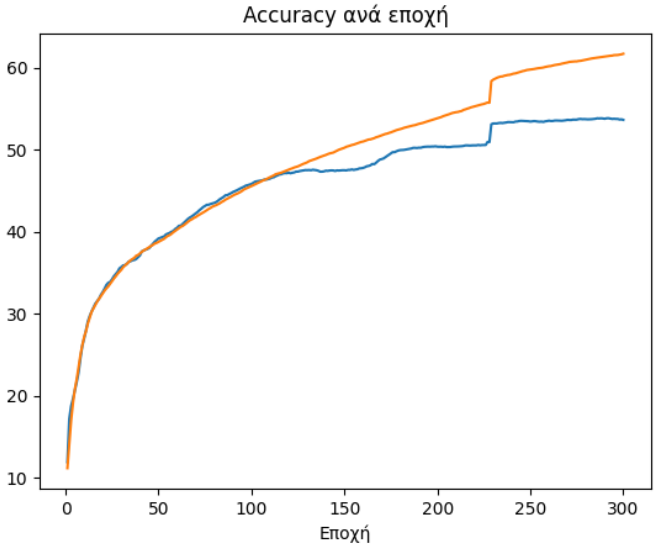
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.03	63.78	63.78	52.77	52.19	27.07

Παρατηρούμε ότι η τελική τιμή του loss, με την εισαγωγή ενός ακόμα hidden layer και την εκπαίδευση για παραπάνω εποχές, μειώθηκε αισθητά. Παράλληλα, το test accuracy αυξήθηκε κατά σχεδόν 2%.

Πείραμα 4:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	512,256,128	0.05 με πολ/σμο επί 0.5 όταν loss_of_epoch >= loss_of_previous_epoch	256	sigmoid	300



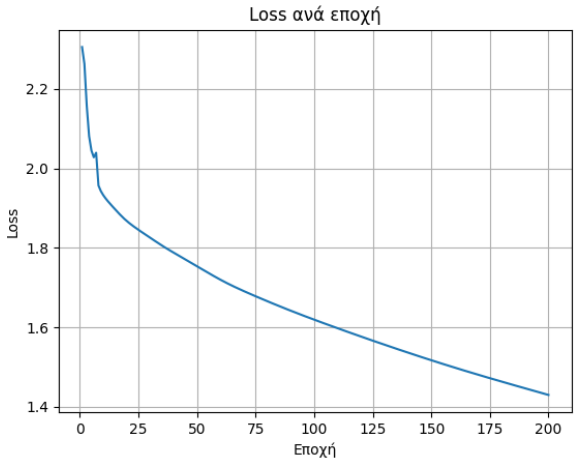


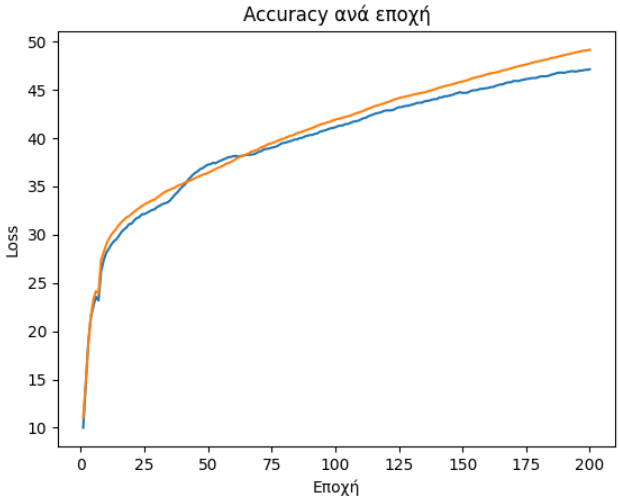
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.08	62.66	62.66	53.78	53.62	68,82

Παρατηρούμε ότι το test accuracy αυξήθηκε κατά 1% για 3 hidden layers και 300 εποχές.

Πείραμα 5:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	1024,512,256	0.03 με πολ/σμο επί 0.7 όταν loss_of_epoch >= loss_of_previous_epoch	256	sigmoid	200





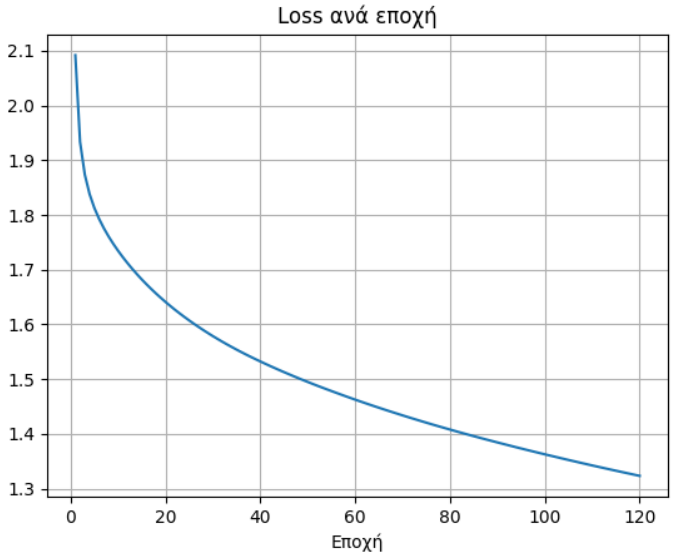
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.43	49.1	49.1	46.9	46.85	113,22

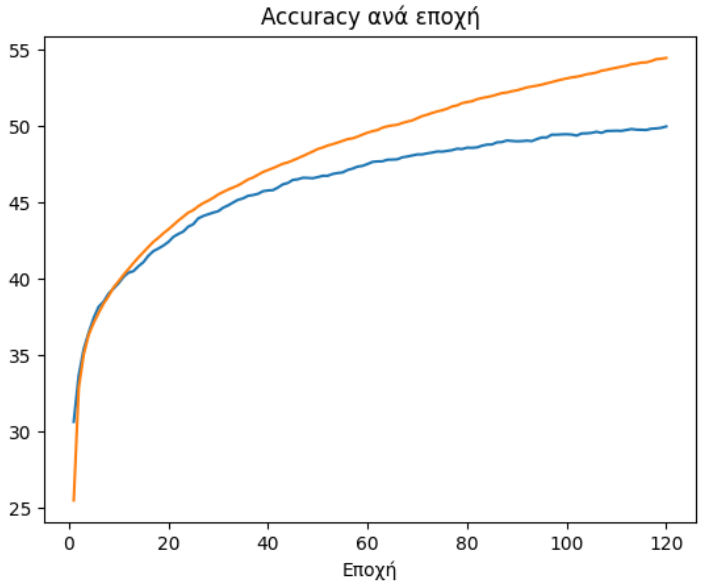
Η αύξηση του πλήθους των νευρώνων σε κάθε hidden layer, σε σχέση με το προηγούμενο πείραμα, δεν οδήγησε σε καλύτερη απόδοση και ο χρόνος εκπαίδευσης αυξήθηκε σε

μεγάλο βαθμό, παρά το γεγονός ότι ο αριθμός των συνολικών εποχών ορίστηκε μικρότερος. Ωστόσο, η μορφή της καμπύλης του test accuracy φανερώνει ότι το δίκτυο για πολλές περισσότερες εποχές πιθανότατα να πετύχαινε μεγαλύτερα ποσοστά στο test accuracy, αφού με το πέρας των 200 εποχών οι τιμές του διαγράμματος (μπλε καμπύλη) δείχνουν να έχουν ακόμη αυξητική τάση.

Πείραμα 6:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
1	128	Σταθερό 0.003	128	ReLu	120



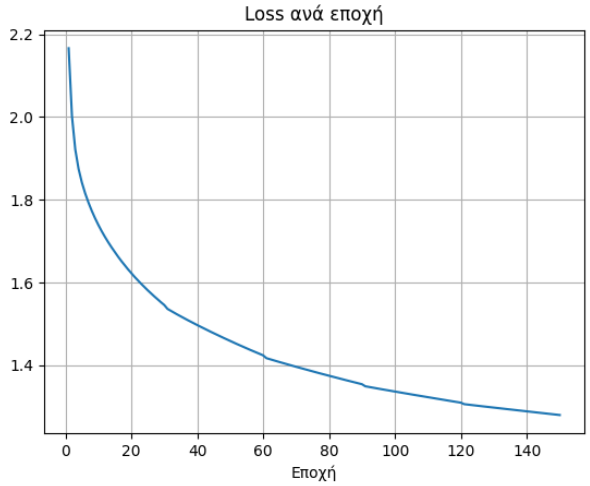


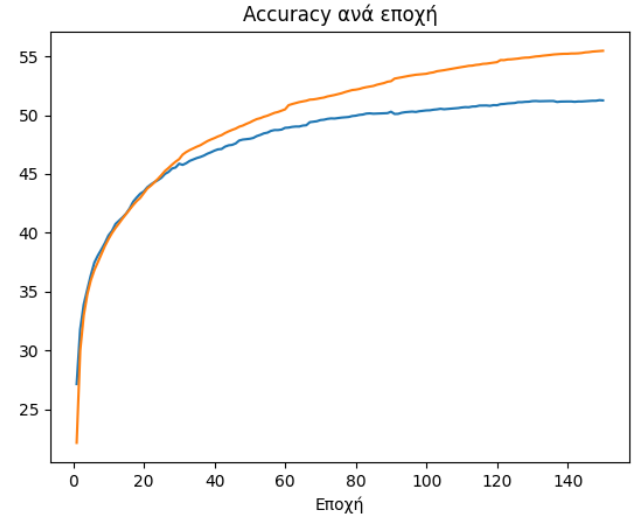
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.32	54.45	54.45	49.97	49.97	10.19

Το πείραμα αυτό έχει τις ίδιες παραμέτρους με το πείραμα 2 με μόνη διαφορά την συνάρτηση ενεργοποίησης των hidden layers. Δεν παρατηρούνται μεγάλες διαφορές όσον αφορά το test accuracy και τον χρόνο εκπαίδευσης του πειράματος με sigmoid και του πειράματος με ReLu. Υπάρχουν όμως διαφορές όσον αφορά τις τελικές τιμές του loss και του training accuracy (η επιλογή sigmoid οδηγεί σε καλύτερη απόδοση στην φάση της εκπαίδευσης).

Πείραμα 7:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
2	256,128	0.05 με πολ/σμο επί 0.9 κάθε 30 εποχές	128	ReLu	150



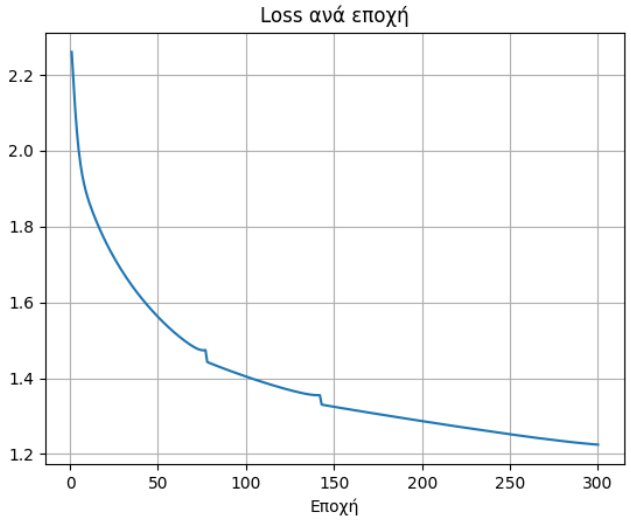


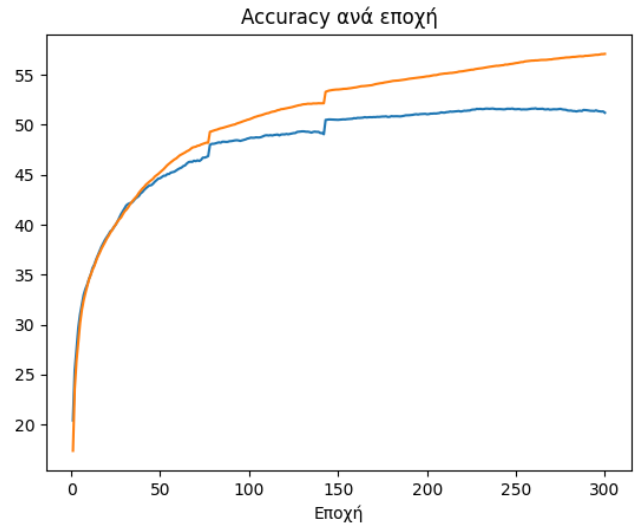
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.28	55.46	55.46	51.27	51.25	15.48

Με την αύξηση των hidden layers σε 2, το δίκτυο επιτυγχάνει μεγαλύτερο ποσοστό του test accuracy, ενώ ο χρόνος εκπαίδευσης με χρήση της ReLu είναι αισθητά μικρότερος σε σχέση με τον αντίστοιχο χρόνο με χρήση της sigmoid (για τις ίδιες υπόλοιπες παραμέτρους του δικτύου) ο οποίος υπολογίστηκε στο πείραμα 3.

Πείραμα 8:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	512,256,128	0.003 με πολ/σμο επί 0.5 όταν loss_of_epoch >= loss_of_previous_epoch	256	ReLu	300



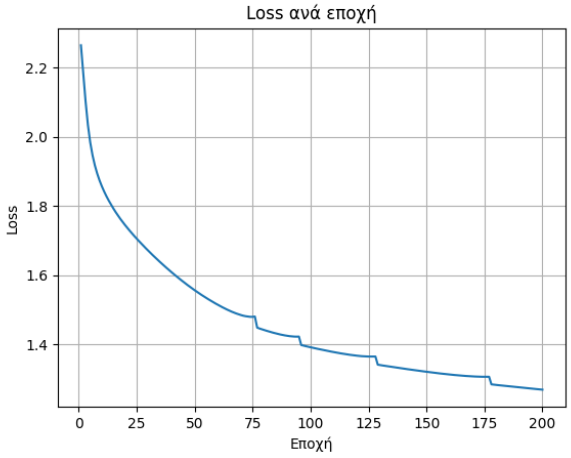


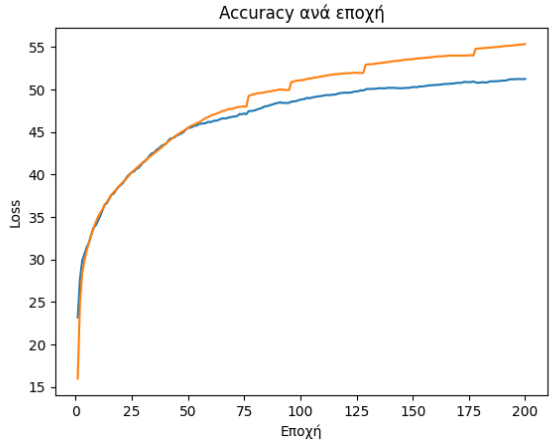
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.22	57.08	57.08	51.64	51.2	52.27

Παρατηρούμε ότι μετά το πέρας της 200^{ης} εποχής η τιμή του test accuracy παραμένει σχετικά σταθερή, οπότε το δίκτυο φαίνεται να έχει φτάσει κοντά στο υψηλότερα επίπεδα απόδοσης που δύναται να επιτύχει με βάση αυτήν την αρχιτεκτονική. Συγκρίνοντας τα πειράματα 8 και 4, παρατηρούμε ότι, ενώ η χρήση της ReLu οδηγεί σε χαμηλότερο ποσοστό test accuracy, ο χρόνος εκπαίδευσης μειώνεται αισθητά.

Πείραμα 9:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	1024,512,256	0.003 με πολ/σμο επί 0.7 όταν loss_of_epoch >= loss_of_previous_epoch	256	ReLu	200



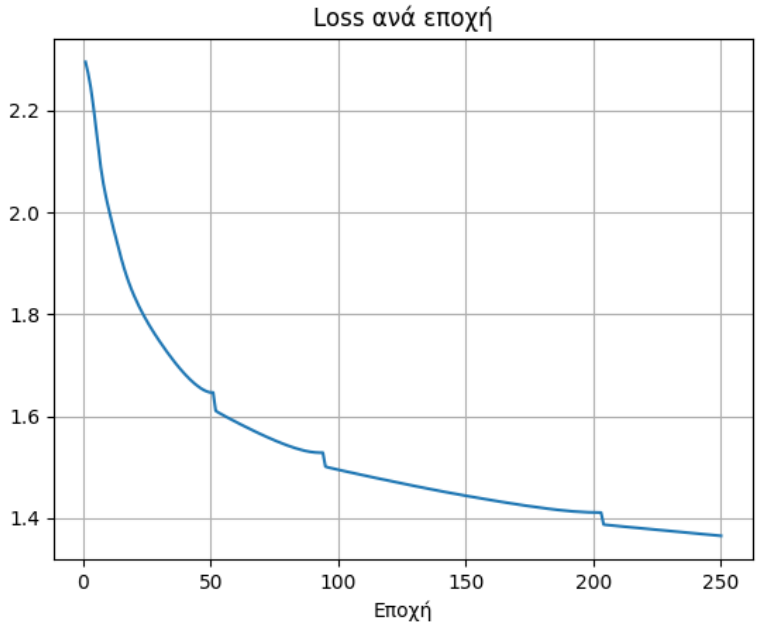


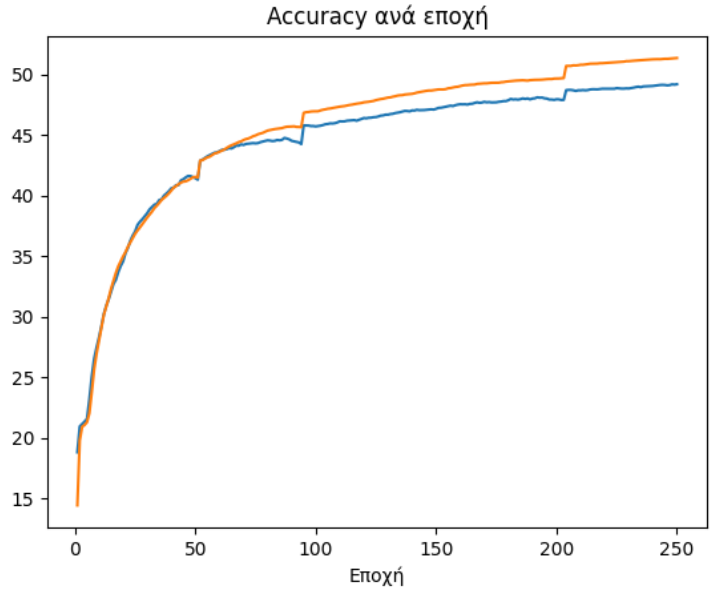
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.24	56.72	56.72	52.13	52.09	91.86

Παρατηρούμε ότι η αύξηση του αριθμού των νευρώνων σε κάθε hidden layer, σε σχέση με το προηγούμενο πείραμα, οδηγεί σε καλύτερο ποσοστό test accuracy αλλά ο χρόνος εκπαίδευσης αυξάνεται ραγδαία. Συγκριτικά με το αντίστοιχο πείραμα με χρήση της sigmoid, το test accuracy είναι μικρότερο για την ReLu.

Πείραμα 10:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
4	512,256,128,64	0.003 με πολ/σμο επί 0.5 όταν loss_of_epoch >= loss_of_previous_epoch	256	ReLu	250



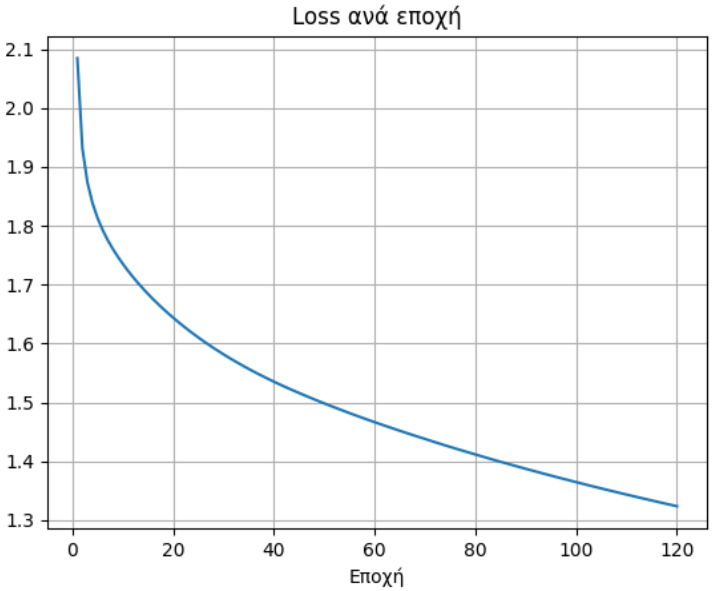


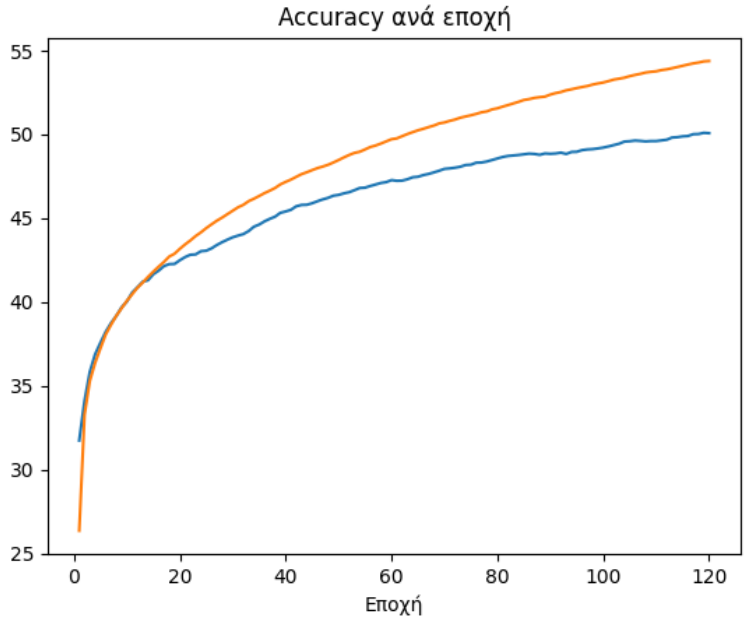
Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.36	51.33	51.33	49.17	49.17	51.66

Παρατηρούμε ότι η προσθήκη 4ου hidden layer δεν οδήγησε σε αύξηση του test accuracy. Αυτό πιθανόν να οφείλεται στο γεγονός ότι το δίκτυο, με αυτήν την αρχιτεκτονική, καθίσταται αρκετά πολύπλοκο για τις ανάγκες του data set CIFAR-10.

Πείραμα 11:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
1	128	Σταθερό 0.003	128	Leaky ReLu	120

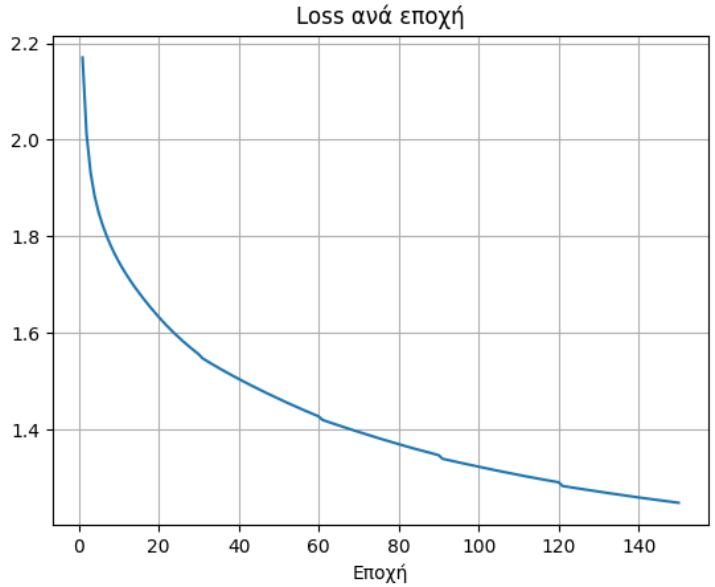


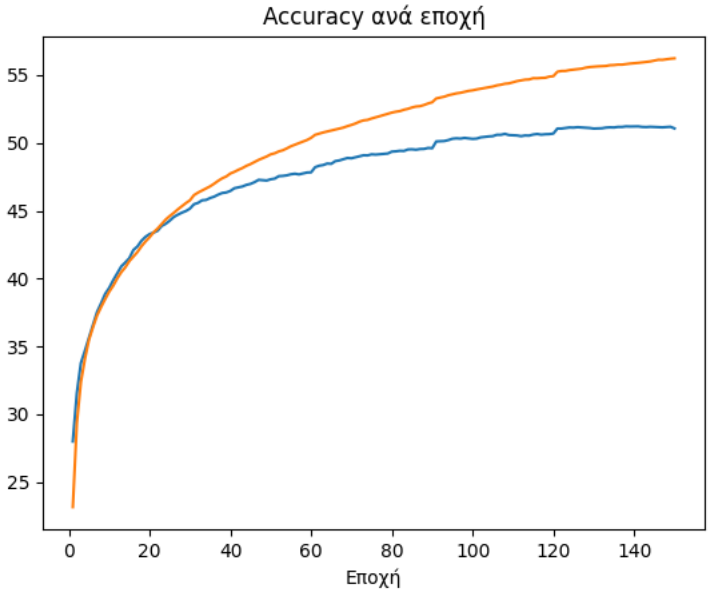


Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.32	54.38	54.38	50.1	50.08	7.46

Πείραμα 12:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
2	256,128	0.05 με πολ/σμο επί 0.9 κάθε 30 εποχές	128	Leaky ReLu	150

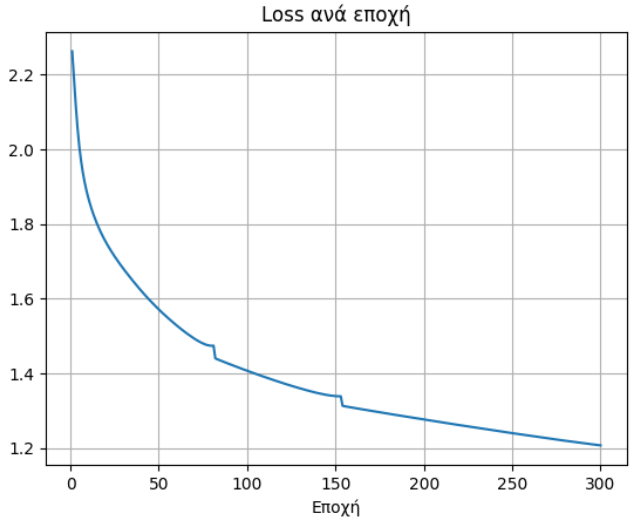


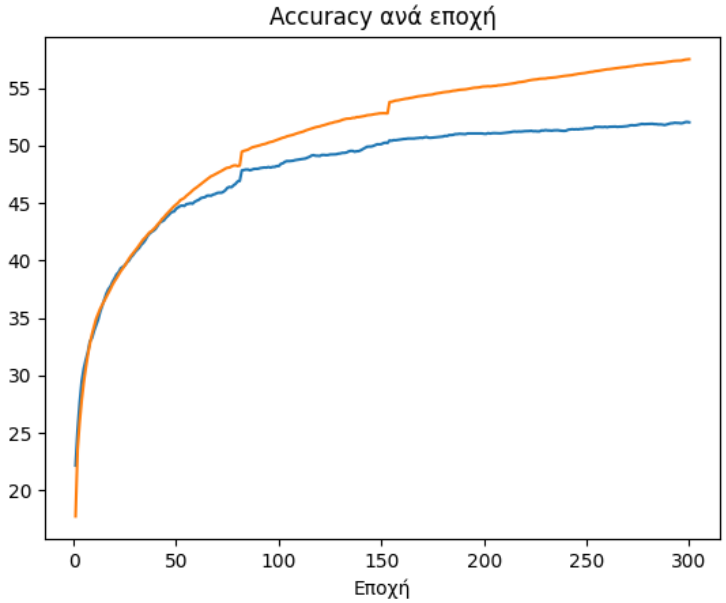


Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.24	56.23	56.23	51.22	51.07	16.97

Πείραμα 13:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	512,256,128	0.003 με πολ/σμο επί 0.5 όταν loss_of_epoch >= loss_of_previous_epoch	256	Leaky ReLu	300

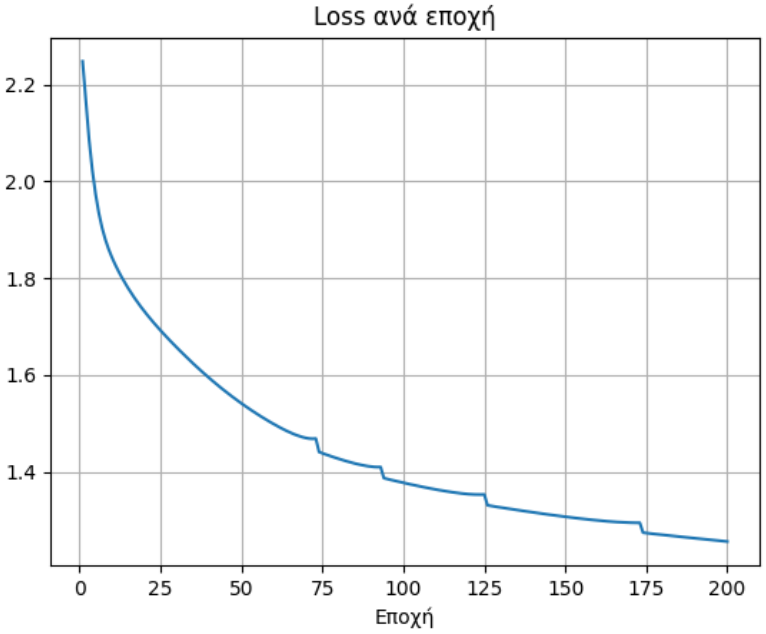


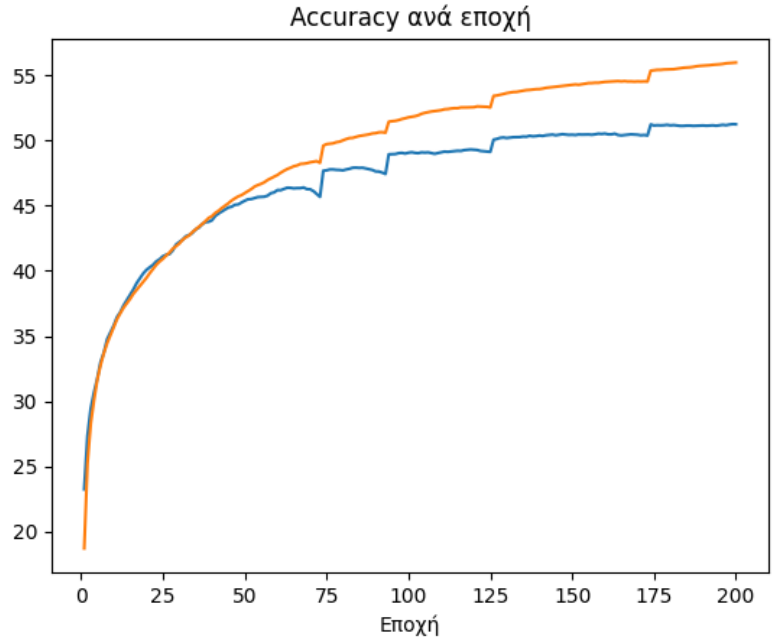


Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.2	57.51	57.51	52.06	52.03	51.45

Πείραμα 14:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
3	1024,512,256	0.003 με πολ/σμο επί 0.7 όταν loss_of_epoch >= loss_of_previous_epoch	256	Leaky ReLu	200

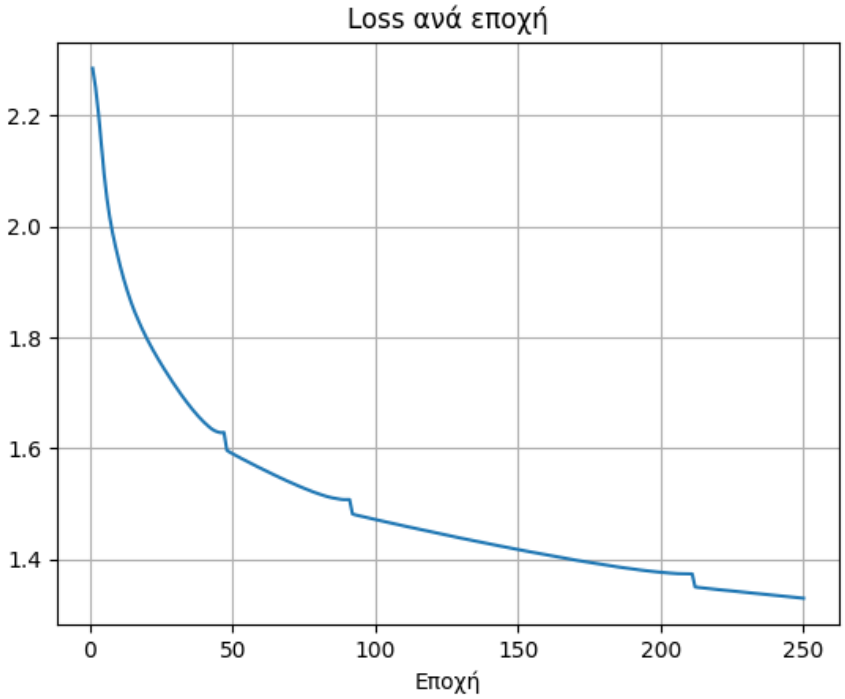


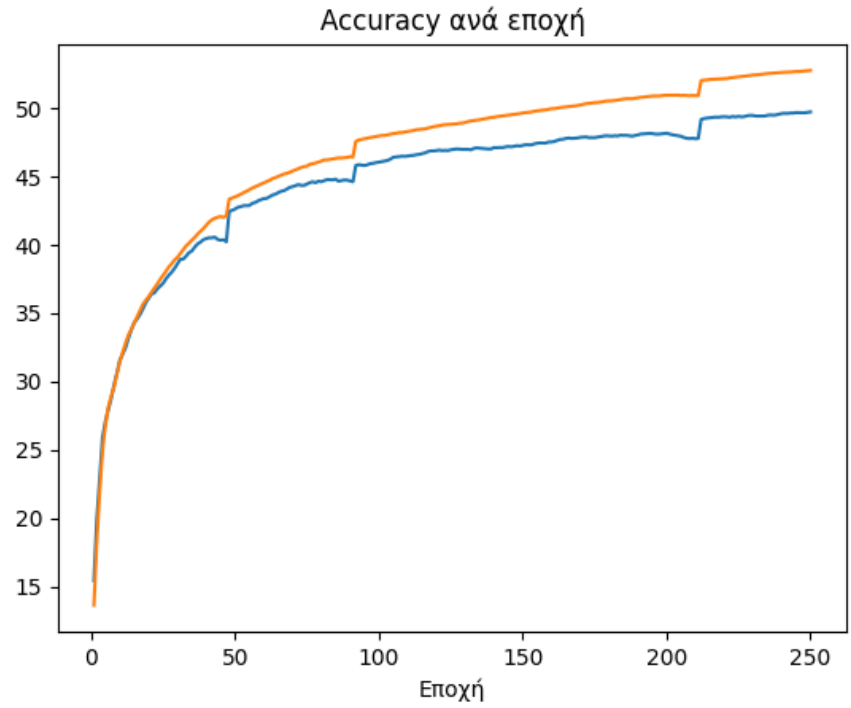


Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.25	55.96	55.96	51.24	51.23	72.99

Πείραμα 15:

Hidden layers	Νευρώνες ανά hidden layer	Learning rate	Batch size	Activation function στα hidden layers	Αριθμός εποχών
4	512,256,128,64	0.003 με πολ/σμο επί 0.5 όταν loss_of_epoch >= loss_of_previous_epoch	256	Leaky ReLu	250





Τελική τιμή loss	Μέγιστο ποσοστό training accuracy	Τελικό ποσοστό training accuracy	Μέγιστο ποσοστό test accuracy	Τελικό ποσοστό test accuracy	Συνολικός χρόνος εκπαίδευσης (σε λεπτά)
1.33	52.76	52.76	49.73	49.73	39.46

Στα πειράματα 11-15 χρησιμοποιούνται παράμετροι ίδιων τιμών με αυτές των πειραμάτων 6-10, με μόνη διαφορά την χρήση της Leaky Relu. Δεν παρατηρούνται μεγάλες διαφορές σε αυτά τα πειράματα συγκριτικά με τα αντίστοιχα που χρησιμοποιήθηκε η ReLu. Το γεγονός αυτό μας οδηγεί στο συμπέρασμα ότι δεν εμφανίζεται το φαινόμενο της ‘’νέκρωσης’’ σε αρκετούς νευρώνες, ώστε η ιδιότητα της Leaky ReLu να επιτρέπει και εξόδους αρνητικών τιμών από τους νευρώνες να είναι ιδιαίτερα εμφανής.

Σύγκριση με κατηγοριοποιητές ενδιάμεσης εργασίας:

Τα ποσοστά επιτυχίας όσον αφορά την πρόβλεψη των κλάσεων των test δειγμάτων, τα οποία αντιστοιχούν στο ποσοστό test accuracy του νευρωνικού δικτύο, καθώς και ο χρόνος εκτέλεσης τους είναι:

Είδος κατηγοριοποιητή	Ποσοστό επιτυχίας προβλέψεων	Χρόνος εκτέλεσης
1-NN	35.39	85 λεπτά
3-NN	64.61	88 λεπτά
Nearest Centroid	27.74	2.04 δευτερόλεπτα

Το νευρωνικό δίκτυο σημείωσε την καλύτερη του απόδοση 53.78% ποσοστό επιτυχίας κατά την πρόβλεψη των κλάσεων των test δειγμάτων του CIFAR-10, με αρχιτεκτονική 3 hidden layers 512,256,128 νευρώνων και χρήση της sigmoid συνάρτησης ενεργοποίησης στα layers αυτά. Σε σχέση με τα ποσοστά επιτυχίας των κατηγοριοποιητών παραπάνω, το δίκτυο επιτυγχάνει πολύ υψηλότερη επίδοση. Παράλληλα, ο χρόνος εκπαίδευσης του δικτύου με αυτήν την αρχιτεκτονική, 68.82 λεπτά, είναι μικρότερος από τους απαιτούμενος χρόνους εκτέλεσης των κατηγοριοποιητών 1-NN και 3-NN που είχαν τις υψηλότερες επιδόσεις μεταξύ των 3 κατηγοριοποιητών. Μια σημαντική παρατήρηση είναι ότι το ποσοστό test accuracy του νευρωνικού είναι υψηλότερο από αυτά των κατηγοριοποιητών ακόμη και μετά τις πρώτες λίγες εποχές. Ωστόσο, η εκπαίδευση σε μεγαλύτερο αριθμό συνολικών εποχών με σκοπό την περαιτέρω αύξηση της τελικής ακρίβειας του δικτύου στο test, εάν υπήρχε η υπολογιστική δυνατότητα, θα οδηγούσε σε αρκετά μεγαλύτερο χρόνο εκπαίδευσης.