

# Πρώτη εργασία

Ματάκος Αλέξανδρος

## Εισαγωγή

Για την εργασία επέλεξα να ασχοληθώ με το πρόβλημα της αναγνώρισης χειρόγραφων ψηφίων. Για την επίλυσή του προβλήματος αυτού γράφηκε πρόγραμμα σε **Python (v3.5)** το οποίο υλοποιεί ένα νευρωνικό δίκτυο πολυστρωματικού perceptron. Χρησιμοποιήθηκε η βάση δεδομένων **MNIST**. Ο κώδικας που χρησιμοποιήθηκε ως βάση για το πρόγραμμα, όπως και η βάση δεδομένων μπορούν να βρεθούν εδώ:

<https://github.com/MichalDanielDobrzanski/DeepLearningPython35>

Η βάση δεδομένων που χρησιμοποιήθηκε, όπως εξηγείται και στον κώδικα, είναι μία ελαφρώς παραλλαγμένη μορφή της **MNIST**, η οποία αποτελείται από πενήντα χιλιάδες training samples, δέκα χιλιάδες validation samples και δέκα χιλιάδες test samples. Τα δείγματα είναι ήδη σε μετασχηματισμένη μορφή (784,1) arrays, με τιμές οι οποίες από 0-255 είναι scaled down σε 0-1, κάτι που διαπίστωσα όταν επιχείρησα ο ίδιος να κάνω scale down.

Όσο αφορά τον κώδικα, όπως αναφέρεται και στο αρχείο **test.py**, μεγάλο μέρος του κώδικα παρέμεινε αυτούσιο, και ανήκει στον original author. Δυστυχώς όντας φοιτητής του Μαθηματικού δεν κατάφερα να το υλοποιήσω μόνος μου from scratch. Παρόλα αυτά, έκανα προσπάθεια να προσθέσω δικά μου features και αλλαγές, οι οποίες είναι:

- Δυνατότητα εφαρμογής **PCA** για μείωση της διάστασης των δεδομένων, καθώς και δυνατότητα αποθήκευσης / φόρτωσης των δεδομένων αυτών. Αυτό έγινε για εξοικονόμηση χρόνου στην εκπαίδευση.
- Δυνατότητα χρήσης της **ReLU** ως συνάρτηση ενεργοποίησης των νευρώνων.
- Δυνατότητα χρήσης **Lasso Regularization** για τον υπολογισμό του κόστους των δεδομένων ανά εποχή. Εδώ ως κόστος ερμηνεύεται η διαφορά του output του νευρωνικού δικτύου με το desired output με τη μέθοδο των ελαχίστων τετραγώνων.

- Προσθήκη λειτουργικών γραμμών κώδικα καθώς και σχολίων σε διάφορα σημεία για την διευκόλυνση του χρήστη και για να φαίνεται πιο ξεκάθαρα τί κάνει το νευρωνικό.

## Πρόγραμμα

Το πρόγραμμα αποτελείται από τρία αρχεία: ***mnist\_loader.py***, ***network2.py*** και ***test.py***.

Παρακάτω περιγράφεται η λειτουργία του καθενός:

### mnist\_loader.py:

Το module αυτό είναι υπεύθυνο για τη φόρτωση, την αποθήκευση και την τροποποίηση των δεδομένων που χρησιμοποιεί το πρόγραμμα.

Πιο συγκεκριμένα, η συνάρτηση **load\_data** επιστρέφει τα δεδομένα της **MNIST** ως tuple που περιέχει τα training\_data, validation\_data και test\_data.

Το training\_data είναι ένα tuple που αποτελείται από δύο εισόδους. Η πρώτη είσοδος είναι οι 50000 φωτογραφίες των χειρόγραφων ψηφίων σε μορφή (784, 1) arrays, ενώ η δεύτερη είσοδος είναι το ψηφίο που αντιστοιχεί στην κάθε εικόνα.

Τα validation\_data και test\_data είναι της ίδιας μορφής, με το καθένα να περιέχει 10000 (784,1) arrays για το test του νευρωνικού δικτύου.

Η συνάρτηση **load\_data\_wrapper** τροποποιεί τη μορφή των δεδομένων αυτών κάνοντας τα πιο κατάλληλα για χρήση στο πρόγραμμα. Στην συνάρτηση αυτή παρέχεται η δυνατότητα γραφικής αναπαράστασης των εικόνων. Επιπροσθέτως, εδώ ο χρήστης μπορεί να πραγματοποιήσει μείωση της διάστασης των δεδομένων με χρήση PCA (Principal Component Analysis), θέτοντας **use\_PCA = True** και εισάγοντας την επιθυμητή διάσταση στη μεταβλητή **n\_components**.

Η συνάρτηση **vectorised\_result** μετατρέπει τα ψηφία που αντιστοιχούν στις εικόνες σε (10,1) arrays με μηδενικά παντού εκτός από την θέση που αντιστοιχεί στον αριθμό του ψηφίου που μετατρέπεται, όπου και μπαίνει άσσος. Με τον τρόπο αυτό μετατρέπονται τα ψηφία σε desired output vectors, για να μπορούν να χρησιμοποιηθούν αργότερα στην εκπαίδευση. Η συνάρτηση **save** αποθηκεύει τα δεδομένα που έχουν υποστεί μείωση διάστασης σε ένα αρχείο με όνομα **PCA\_data**, έτσι ώστε να μπορούν να χρησιμοποιηθούν για χρήση του προγράμματος αργότερα, καλώντας την συνάρτηση **load**.

## network2.py:

Το module αυτό είναι αυτό που περιέχει τον core κώδικα του νευρωνικού. Αρχικά ορίζονται δύο κλάσεις, οι **QuadraticCost** και **CrossEntropyCost**, οι οποίες δίνουν δύο διαφορετικές συναρτήσεις κόστους. Στα πειράματα που θα δείξω στη συνέχεια χρησιμοποιήθηκε κυρίως η **CrossEntropyCost**.

Έπειτα ορίζεται η κύρια κλάση του δικτύου **Network**, η οποία παίρνει ως είσοδο το μέγεθος του νευρωνικού - πλήθος στρωμάτων καθώς και πλήθος νευρώνων ανά στρώμα - και την κλάση που θα χρησιμοποιηθεί για τον υπολογισμό των κόστων.

Με τη συνάρτηση **default\_weight\_initializer** γίνεται αρχικοποίηση των βαρών και των biases για όλο το δίκτυο. Για τα biases χρησιμοποιείται η συνήθης κανονική κατανομή  $N(\mu=0, \sigma=1)$ . Για τα βάρη χρησιμοποιείται η συνήθης κανονική κατανομή  $N(\mu=0, \sigma=1)$  διαιρεμένη με τον όρο  $\sqrt{x}$ , όπου  $x$  είναι το πλήθος των βαρών που συνδέουν έναν νευρώνα με το στρώμα που βρίσκεται στα αριστερά του.

Η συνάρτηση **feedforward** επιστρέφει την έξοδο του νευρωνικού για είσοδο **a**.

Η εκπαίδευση του νευρωνικού δικτύου υλοποιείται με τις συναρτήσεις **SGD**, **update\_mini\_batch**, και **backprop**.

Η συνάρτηση **SGD** (Stochastic Gradient Descent) έχει για παραμέτρους τα δεδομένα εισόδου **training\_data**, το πλήθος των εποχών εκπαίδευσης **epochs**, τον ρυθμό εκπαίδευσης **eta** και regularization parameter **lmbda**. Παρέχει την δυνατότητα εκπαίδευσης με **batches** για την επιτάχυνση της διαδικασίας εκπαίδευσης, με παράμετρο **mini\_batch\_size** η οποία καθορίζει το μέγεθος των **batches**.

Επιπροσθέτως, μέσω της **SGD** έχουμε δυνατότητα να κάνουμε monitor τους εξής δείκτες:

**evaluation\_cost**, **evaluation\_accuracy**, **training\_cost**, **training\_accuracy**.

Η ακρίβεια στο test και στο train τυπώνονται στον χρήστη με απόλυτες τιμές καθώς και με ποσοστά, και υπολογίζονται από τη συνάρτηση **accuracy**.

Οι άλλες δύο συναρτήσεις είναι αυτές που κάνουν τις αλλαγές στα συναντικα βάρη και τα biases και υλοποιούν το μαθηματικό μοντέλο του αλγόριθμου **back-propagation**. Δίνεται η δυνατότητα χρήσης σιγμοειδούς και συνάρτησης **ReLU** ως συναρτήσεις ενεργοποίησης.

### test.py:

Αυτό είναι το module με το οποίο ο χρήστης τρέχει το πρόγραμμα. Εδώ καλούνται τα άλλα δύο modules και εδώ ορίζονται όλες οι παράμετροι που προαναφέρθηκαν για την εκπαίδευση του νευρωνικού δικτύου.

### Πειράματα

Στα πρώτα πειράματα, και μέχρι να αναφέρω την αλλαγή παρακάτω, χρησιμοποιείται συνάρτηση κόστους QuadraticCost. Η διάσταση του **Network** θα συμβολίζει το πλήθος των στρωμάτων του δικτύου και η τιμή της κάθε συντεταγμένης το πλήθος των νευρώνων στο αντίστοιχο στρώμα. Αρχικά επιχείρησα να αλλάξω τη συνάρτηση ενεργοποίησης σε **ReLU**. Τα πρώτα αποτελέσματα δεν ήταν καλά, όπως φαίνεται παρακάτω. Οι υπόλοιπες παράμετροι ήταν ως εξής: **Network** = [784, 30, 10], **training\_data** = 50000, **epochs** = 30, **batch size** = 100, **eta** = 0.1, **lmbda** = 5.0

(base)

Έπειτα απο δική σας παρότρυνση, χρησιμοποίησα σιγμοειδή συνάρτηση για επιστροφή των gradients μόνο από το output layer, και τα αποτελέσματα ήταν σαφώς καλύτερα, με ποσοστά κοντά στο 86% και στο test και στο train:

```

$ python test.py
Epoch 0 training complete
Accuracy on training data: 27843 / 50000
Accuracy on evaluation data: 5714 / 10000
Epoch 1 training complete
Accuracy on training data: 30935 / 50000
Accuracy on evaluation data: 6419 / 10000
Epoch 2 training complete
Accuracy on training data: 35945 / 50000
Accuracy on evaluation data: 7419 / 10000
Epoch 3 training complete
Accuracy on training data: 38142 / 50000
Accuracy on evaluation data: 7947 / 10000
Epoch 4 training complete
Accuracy on training data: 38967 / 50000
Accuracy on evaluation data: 8029 / 10000
Epoch 5 training complete
Accuracy on training data: 38613 / 50000
Accuracy on evaluation data: 7976 / 10000
Epoch 6 training complete
Accuracy on training data: 39365 / 50000
Accuracy on evaluation data: 8129 / 10000
Epoch 7 training complete
Accuracy on training data: 39982 / 50000
Accuracy on evaluation data: 8241 / 10000
Epoch 8 training complete
Accuracy on training data: 40127 / 50000
Accuracy on evaluation data: 8299 / 10000
Epoch 9 training complete
Accuracy on training data: 40575 / 50000
Accuracy on evaluation data: 8354 / 10000
Epoch 10 training complete
Accuracy on training data: 41035 / 50000
Accuracy on evaluation data: 8441 / 10000
Epoch 11 training complete
Accuracy on training data: 40738 / 50000
Accuracy on evaluation data: 8395 / 10000
Epoch 12 training complete
Accuracy on training data: 40922 / 50000
Accuracy on evaluation data: 8434 / 10000
Epoch 13 training complete
Accuracy on training data: 41073 / 50000
Accuracy on evaluation data: 8405 / 10000
Epoch 14 training complete
Accuracy on training data: 40868 / 50000
Accuracy on evaluation data: 8423 / 10000
Epoch 15 training complete
Accuracy on training data: 41142 / 50000
Accuracy on evaluation data: 8452 / 10000
Epoch 16 training complete
Accuracy on training data: 41380 / 50000
Accuracy on evaluation data: 8485 / 10000
Epoch 17 training complete
Accuracy on training data: 40938 / 50000
Accuracy on evaluation data: 8408 / 10000
Epoch 18 training complete
Accuracy on training data: 41046 / 50000
Accuracy on evaluation data: 8395 / 10000
Epoch 19 training complete
Accuracy on training data: 41730 / 50000
Accuracy on evaluation data: 8554 / 10000
Epoch 20 training complete
Accuracy on training data: 41769 / 50000
Accuracy on evaluation data: 8552 / 10000
Epoch 21 training complete
Accuracy on training data: 42129 / 50000
Accuracy on evaluation data: 8603 / 10000
Epoch 22 training complete
Accuracy on training data: 41791 / 50000
Accuracy on evaluation data: 8561 / 10000
Epoch 23 training complete
Accuracy on training data: 41559 / 50000
Accuracy on evaluation data: 8539 / 10000
Epoch 24 training complete
Accuracy on training data: 41811 / 50000
Accuracy on evaluation data: 8562 / 10000
Epoch 25 training complete
Accuracy on training data: 41337 / 50000
Accuracy on evaluation data: 8477 / 10000
Epoch 26 training complete
Accuracy on training data: 41700 / 50000
Accuracy on evaluation data: 8549 / 10000
Epoch 27 training complete
Accuracy on training data: 41560 / 50000
Accuracy on evaluation data: 8525 / 10000
Epoch 28 training complete
Accuracy on training data: 41642 / 50000
Accuracy on evaluation data: 8537 / 10000
Epoch 29 training complete
Accuracy on training data: 41828 / 50000
Accuracy on evaluation data: 8589 / 10000

```



Προσθέτοντας ακόμη ένα κρυφό στρώμα με 30 νευρώνες πέφτουμε κοντά στο 81%:

```
$ python test.py
Epoch 0 training complete
Accuracy on training data: 20163 / 50000
Accuracy on evaluation data: 4269 / 10000
Epoch 1 training complete
Accuracy on training data: 27733 / 50000
Accuracy on evaluation data: 5770 / 10000
Epoch 2 training complete
Accuracy on training data: 31066 / 50000
Accuracy on evaluation data: 6386 / 10000
Epoch 3 training complete
Accuracy on training data: 31834 / 50000
Accuracy on evaluation data: 6560 / 10000
Epoch 4 training complete
Accuracy on training data: 34477 / 50000
Accuracy on evaluation data: 7124 / 10000
Epoch 5 training complete
Accuracy on training data: 34899 / 50000
Accuracy on evaluation data: 7210 / 10000
Epoch 6 training complete
Accuracy on training data: 34262 / 50000
Accuracy on evaluation data: 6984 / 10000
Epoch 7 training complete
Accuracy on training data: 36315 / 50000
Accuracy on evaluation data: 7479 / 10000
Epoch 8 training complete
Accuracy on training data: 36386 / 50000
Accuracy on evaluation data: 7552 / 10000
Epoch 9 training complete
Accuracy on training data: 37221 / 50000
Accuracy on evaluation data: 7659 / 10000
Epoch 10 training complete
Accuracy on training data: 36326 / 50000
Accuracy on evaluation data: 7511 / 10000
Epoch 11 training complete
Accuracy on training data: 36966 / 50000
Accuracy on evaluation data: 7621 / 10000
Epoch 12 training complete
Accuracy on training data: 36630 / 50000
Accuracy on evaluation data: 7564 / 10000
Epoch 13 training complete
Accuracy on training data: 37353 / 50000
Accuracy on evaluation data: 7706 / 10000
Epoch 14 training complete
Accuracy on training data: 37927 / 50000
Accuracy on evaluation data: 7852 / 10000
Epoch 15 training complete
Accuracy on training data: 37692 / 50000
Accuracy on evaluation data: 7764 / 10000
Epoch 16 training complete
Accuracy on training data: 37980 / 50000
Accuracy on evaluation data: 7831 / 10000
Epoch 17 training complete
Accuracy on training data: 38651 / 50000
Accuracy on evaluation data: 7979 / 10000
Epoch 18 training complete
Accuracy on training data: 38610 / 50000
Accuracy on evaluation data: 7994 / 10000
Epoch 19 training complete
Accuracy on training data: 39084 / 50000
Accuracy on evaluation data: 8066 / 10000
Epoch 20 training complete
Accuracy on training data: 38931 / 50000
Accuracy on evaluation data: 8014 / 10000
Epoch 21 training complete
Accuracy on training data: 39075 / 50000
Accuracy on evaluation data: 8069 / 10000
Epoch 22 training complete
Accuracy on training data: 38538 / 50000
Accuracy on evaluation data: 7972 / 10000
Epoch 23 training complete
Accuracy on training data: 39283 / 50000
Accuracy on evaluation data: 8093 / 10000
Epoch 24 training complete
Accuracy on training data: 39004 / 50000
Accuracy on evaluation data: 8055 / 10000
Epoch 25 training complete
Accuracy on training data: 39033 / 50000
Accuracy on evaluation data: 8094 / 10000
Epoch 26 training complete
Accuracy on training data: 39688 / 50000
Accuracy on evaluation data: 8171 / 10000
Epoch 27 training complete
Accuracy on training data: 39481 / 50000
Accuracy on evaluation data: 8129 / 10000
Epoch 28 training complete
Accuracy on training data: 39358 / 50000
Accuracy on evaluation data: 8121 / 10000
Epoch 29 training complete
Accuracy on training data: 39525 / 50000
Accuracy on evaluation data: 8133 / 10000
```



Αφαιρώντας και τα δύο κρυφά στρώματα συνεχίζουμε να έχουμε πολύ καλή απόδοση κοντά στο 86%:

```
$ python test.py
Epoch 0 training complete
Accuracy on training data: 33377 / 50000
Accuracy on evaluation data: 6973 / 10000
Epoch 1 training complete
Accuracy on training data: 39168 / 50000
Accuracy on evaluation data: 8063 / 10000
Epoch 2 training complete
Accuracy on training data: 41301 / 50000
Accuracy on evaluation data: 8453 / 10000
Epoch 3 training complete
Accuracy on training data: 41959 / 50000
Accuracy on evaluation data: 8556 / 10000
Epoch 4 training complete
Accuracy on training data: 42417 / 50000
Accuracy on evaluation data: 8645 / 10000
Epoch 5 training complete
Accuracy on training data: 42378 / 50000
Accuracy on evaluation data: 8630 / 10000
Epoch 6 training complete
Accuracy on training data: 42380 / 50000
Accuracy on evaluation data: 8644 / 10000
Epoch 7 training complete
Accuracy on training data: 42451 / 50000
Accuracy on evaluation data: 8650 / 10000
Epoch 8 training complete
Accuracy on training data: 42613 / 50000
Accuracy on evaluation data: 8669 / 10000
Epoch 9 training complete
Accuracy on training data: 42666 / 50000
Accuracy on evaluation data: 8716 / 10000
Epoch 10 training complete
Accuracy on training data: 42263 / 50000
Accuracy on evaluation data: 8642 / 10000
Epoch 11 training complete
Accuracy on training data: 42436 / 50000
Accuracy on evaluation data: 8637 / 10000
Epoch 12 training complete
Accuracy on training data: 42350 / 50000
Accuracy on evaluation data: 8617 / 10000
Epoch 13 training complete
Accuracy on training data: 42569 / 50000
Accuracy on evaluation data: 8644 / 10000
Epoch 14 training complete
Accuracy on training data: 42278 / 50000
Accuracy on evaluation data: 8582 / 10000
Epoch 15 training complete
Accuracy on training data: 42511 / 50000
Accuracy on evaluation data: 8634 / 10000
Epoch 16 training complete
Accuracy on training data: 42388 / 50000
Accuracy on evaluation data: 8612 / 10000
Epoch 17 training complete
Accuracy on training data: 42464 / 50000
Accuracy on evaluation data: 8618 / 10000
Epoch 18 training complete
Accuracy on training data: 42420 / 50000
Accuracy on evaluation data: 8633 / 10000
Epoch 19 training complete
Accuracy on training data: 42634 / 50000
Accuracy on evaluation data: 8668 / 10000
Epoch 20 training complete
Accuracy on training data: 42623 / 50000
Accuracy on evaluation data: 8672 / 10000
Epoch 21 training complete
Accuracy on training data: 42429 / 50000
Accuracy on evaluation data: 8615 / 10000
Epoch 22 training complete
Accuracy on training data: 42417 / 50000
Accuracy on evaluation data: 8624 / 10000
Epoch 23 training complete
Accuracy on training data: 42583 / 50000
Accuracy on evaluation data: 8643 / 10000
Epoch 24 training complete
Accuracy on training data: 42482 / 50000
Accuracy on evaluation data: 8622 / 10000
Epoch 25 training complete
Accuracy on training data: 42710 / 50000
Accuracy on evaluation data: 8677 / 10000
Epoch 26 training complete
Accuracy on training data: 42287 / 50000
Accuracy on evaluation data: 8614 / 10000
Epoch 27 training complete
Accuracy on training data: 42718 / 50000
Accuracy on evaluation data: 8663 / 10000
Epoch 28 training complete
Accuracy on training data: 42548 / 50000
Accuracy on evaluation data: 8644 / 10000
Epoch 29 training complete
Accuracy on training data: 42581 / 50000
Accuracy on evaluation data: 8639 / 10000
```

Στο δίκτυο χωρίς κρυφό στρώμα για τιμές της παραμέτρου **lmbda** = 0.0, 5.0, 10.0, 50.0 η εικόνα παρέμεινε ίδια, με ποσοστά γύρω στο 86%. Ενδεικτικά για 10 εποχές και **lmbda** = 50.0:

```
$ python test.py
Epoch 0 training complete
Accuracy on training data: 37416 / 50000
Accuracy on evaluation data: 7738 / 10000
Epoch 1 training complete
Accuracy on training data: 40975 / 50000
Accuracy on evaluation data: 8383 / 10000
Epoch 2 training complete
Accuracy on training data: 42011 / 50000
Accuracy on evaluation data: 8561 / 10000
Epoch 3 training complete
Accuracy on training data: 42057 / 50000
Accuracy on evaluation data: 8584 / 10000
Epoch 4 training complete
Accuracy on training data: 42698 / 50000
Accuracy on evaluation data: 8708 / 10000
Epoch 5 training complete
Accuracy on training data: 42642 / 50000
Accuracy on evaluation data: 8690 / 10000
Epoch 6 training complete
Accuracy on training data: 42572 / 50000
Accuracy on evaluation data: 8655 / 10000
Epoch 7 training complete
Accuracy on training data: 42629 / 50000
Accuracy on evaluation data: 8637 / 10000
Epoch 8 training complete
Accuracy on training data: 42500 / 50000
Accuracy on evaluation data: 8658 / 10000
Epoch 9 training complete
Accuracy on training data: 42510 / 50000
Accuracy on evaluation data: 8653 / 10000
Epoch 10 training complete
Accuracy on training data: 42545 / 50000
Accuracy on evaluation data: 8666 / 10000
Epoch 11 training complete
```

Έπειτα για Network[784, 30, 10], epochs = 30, eta = 0.2, lmbda = 0.0 έχουμε ελαφρώς χειρότερα ποσοστά, κοντά στο 75%:

```

$ python test.py
Network([784, 30, 10], cost_function=QuadraticCost)
training_data: 50000, epochs=30, batch_size=100
learning_rate=0.2, lmbda = 0.0 (Lasso regularization)
evaluation_data: 10000
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 0 training complete
Accuracy on training data: 143 / 1000 - 1.43 %
Cost on evaluation data: 0.7795540236802844
Accuracy on evaluation data: 1427 / 10000 - 14.27 %
Epoch 1 training complete
Accuracy on training data: 192 / 1000 - 1.92 %
Cost on evaluation data: 0.7683427066887858
Accuracy on evaluation data: 1804 / 10000 - 18.04 %
Epoch 2 training complete
Accuracy on training data: 210 / 1000 - 2.10 %
Cost on evaluation data: 0.7806878801664985
Accuracy on evaluation data: 1963 / 10000 - 19.63 %
Epoch 3 training complete
Accuracy on training data: 365 / 1000 - 3.65 %
Cost on evaluation data: 0.5514499497029887
Accuracy on evaluation data: 3259 / 10000 - 32.59 %
Epoch 4 training complete
Accuracy on training data: 464 / 1000 - 4.64 %
Cost on evaluation data: 0.5143735366155502
Accuracy on evaluation data: 4051 / 10000 - 40.51 %
Epoch 5 training complete
Accuracy on training data: 532 / 1000 - 5.32 %
Cost on evaluation data: 0.456369686201266
Accuracy on evaluation data: 4738 / 10000 - 47.38 %
Epoch 6 training complete
Accuracy on training data: 581 / 1000 - 5.81 %
Cost on evaluation data: 0.4287901443255604
Accuracy on evaluation data: 5350 / 10000 - 53.50 %
Epoch 7 training complete
Accuracy on training data: 619 / 1000 - 6.19 %
Cost on evaluation data: 0.4083857352512213
Accuracy on evaluation data: 5875 / 10000 - 58.75 %
Epoch 8 training complete
Accuracy on training data: 616 / 1000 - 6.16 %
Cost on evaluation data: 0.3965671772673078
Accuracy on evaluation data: 5843 / 10000 - 58.43 %
Epoch 9 training complete
Accuracy on training data: 642 / 1000 - 6.42 %
Cost on evaluation data: 0.3945032942130341
Accuracy on evaluation data: 6194 / 10000 - 61.94 %
Epoch 10 training complete
Accuracy on training data: 671 / 1000 - 6.71 %
Cost on evaluation data: 0.36621432893247036
Accuracy on evaluation data: 6400 / 10000 - 64.00 %
Epoch 11 training complete
Accuracy on training data: 694 / 1000 - 6.94 %
Cost on evaluation data: 0.3450448625203328
Accuracy on evaluation data: 6473 / 10000 - 64.73 %
Epoch 12 training complete
Accuracy on training data: 667 / 1000 - 6.67 %
Cost on evaluation data: 0.36599616689849396
Accuracy on evaluation data: 6173 / 10000 - 61.73 %
Epoch 13 training complete
Accuracy on training data: 715 / 1000 - 7.15 %
Cost on evaluation data: 0.3390624891414089
Accuracy on evaluation data: 6656 / 10000 - 66.56 %
Epoch 14 training complete
Accuracy on training data: 726 / 1000 - 7.26 %
Cost on evaluation data: 0.3409611802112521
Accuracy on evaluation data: 6861 / 10000 - 68.61 %
Epoch 15 training complete
Accuracy on training data: 716 / 1000 - 7.16 %
Cost on evaluation data: 0.327352835435548
Accuracy on evaluation data: 6843 / 10000 - 68.43 %
Epoch 16 training complete
Accuracy on training data: 762 / 1000 - 7.62 %
Cost on evaluation data: 0.3309501984250789
Accuracy on evaluation data: 7058 / 10000 - 70.58 %
Epoch 17 training complete
Accuracy on training data: 772 / 1000 - 7.72 %
Cost on evaluation data: 0.32483474147924446
Accuracy on evaluation data: 7116 / 10000 - 71.16 %
Epoch 18 training complete
Accuracy on training data: 785 / 1000 - 7.85 %
Cost on evaluation data: 0.3173651824943835
Accuracy on evaluation data: 7249 / 10000 - 72.49 %
Epoch 19 training complete
Accuracy on training data: 755 / 1000 - 7.55 %
Cost on evaluation data: 0.3179333333878829
Accuracy on evaluation data: 6924 / 10000 - 69.24 %
Epoch 20 training complete
Accuracy on training data: 811 / 1000 - 8.11 %
Cost on evaluation data: 0.2960408541668034
Accuracy on evaluation data: 7426 / 10000 - 74.26 %
Epoch 21 training complete
Accuracy on training data: 783 / 1000 - 7.83 %
Cost on evaluation data: 0.295706062200475
Accuracy on evaluation data: 7225 / 10000 - 72.25 %
Epoch 22 training complete
Accuracy on training data: 800 / 1000 - 8.00 %
Cost on evaluation data: 0.2921515731144407
Accuracy on evaluation data: 7487 / 10000 - 74.87 %
Epoch 23 training complete
Accuracy on training data: 779 / 1000 - 7.79 %
Cost on evaluation data: 0.2939436087418344
Accuracy on evaluation data: 7256 / 10000 - 72.56 %
Epoch 24 training complete
Accuracy on training data: 810 / 1000 - 8.10 %
Cost on evaluation data: 0.28661388876229676
Accuracy on evaluation data: 7505 / 10000 - 75.05 %
Epoch 25 training complete
Accuracy on training data: 802 / 1000 - 8.02 %
Cost on evaluation data: 0.28395677848296336
Accuracy on evaluation data: 7481 / 10000 - 74.81 %
Epoch 26 training complete
Accuracy on training data: 813 / 1000 - 8.13 %
Cost on evaluation data: 0.2849993333966323
Accuracy on evaluation data: 7463 / 10000 - 74.63 %
Epoch 27 training complete
Accuracy on training data: 824 / 1000 - 8.24 %
Cost on evaluation data: 0.28138777622277217
Accuracy on evaluation data: 7587 / 10000 - 75.87 %
Epoch 28 training complete
Accuracy on training data: 817 / 1000 - 8.17 %
Cost on evaluation data: 0.28479630819161943
Accuracy on evaluation data: 7515 / 10000 - 75.15 %
Epoch 29 training complete
Accuracy on training data: 814 / 1000 - 8.14 %
Cost on evaluation data: 0.28800526690532285
Accuracy on evaluation data: 7469 / 10000 - 74.69 %

```



Από εδώ και πέρα στα πειράματα χρησιμοποιήθηκε CrossEntropy cost function. Στη συνέχεια παραθέτω κάποια ακόμη πειράματα, με τις παραμέτρους που αλλάζουν να φαίνονται στις φωτογραφίες:

```

Network: [784, 16, 16, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.1, lambda = 5.0
epochs = 3, batch size = 10
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 160.05625560401978
Cost on evaluation data: 159.98874767104215
Accuracy on evaluation data: 5396 / 10000 = 53.96 %
Epoch 2 training complete
Cost on training data: 189.8411845713301
Cost on evaluation data: 189.84538141083038
Accuracy on evaluation data: 4279 / 10000 = 42.79 %
Epoch 3 training complete
Cost on training data: 209.29648408038076
Cost on evaluation data: 209.30869923990255
Accuracy on evaluation data: 5227 / 10000 = 52.27 %
(base)

Network: [784, 100, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.1, lambda = 50.0
epochs = 3, batch size = 10
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1842.1172538807273
Cost on evaluation data: 1842.0910201537229
Accuracy on evaluation data: 6339 / 10000 = 63.39 %
Epoch 2 training complete
Cost on training data: 1308.144911233069
Cost on evaluation data: 1308.1141807457982
Accuracy on evaluation data: 5457 / 10000 = 54.57 %
Epoch 3 training complete
Cost on training data: 1106.54038194241
Cost on evaluation data: 1106.4787088273529
Accuracy on evaluation data: 4456 / 10000 = 44.56 %
(base)

Network: [784, 10, 10, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.1, lambda = 50.0
epochs = 3, batch size = 10
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 689.2203578644632
Cost on evaluation data: 689.1954715477657
Accuracy on evaluation data: 6269 / 10000 = 62.69 %
Epoch 2 training complete
Cost on training data: 834.9985157360944
Cost on evaluation data: 834.9479859091115
Accuracy on evaluation data: 4719 / 10000 = 47.19 %
Epoch 3 training complete
Cost on training data: 937.2935934677369
Cost on evaluation data: 937.3257754329813
Accuracy on evaluation data: 3571 / 10000 = 35.71 %
(base)

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.05, lambda = 50.0
epochs = 3, batch size = 10
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1431.524869642881
Cost on evaluation data: 1431.4635689070833
Accuracy on evaluation data: 8022 / 10000 = 80.22 %
Epoch 2 training complete
Cost on training data: 1888.1635785017036
Cost on evaluation data: 1888.3081814927332
Accuracy on evaluation data: 4665 / 10000 = 46.65 %
Epoch 3 training complete
Cost on training data: 1870.8198949890966
Cost on evaluation data: 1870.7856768964302
Accuracy on evaluation data: 5095 / 10000 = 50.95 %
#

```

Βλέπουμε ότι κανένα απο αυτά τα νευρωνικά δεν συγκλίνει.

Έπειτα θέτουμε **batch\_size = 1**, δηλαδή εκπαιδεύουμε με το σύνολο της βάσης δεδομένων και κάνουμε αλλαγές στις παραμέτρους **eta**, **lmbda**.

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 50.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 526.1052345769451
Cost on evaluation data: 526.0626421014296
Accuracy on evaluation data: 4629 / 10000 - 46.29 %
Epoch 2 training complete
Cost on training data: 538.9316029129699
Cost on evaluation data: 538.86456276582
Accuracy on evaluation data: 3865 / 10000 - 38.65 %
Epoch 3 training complete
Cost on training data: 507.613810107047
Cost on evaluation data: 507.5734531969353
Accuracy on evaluation data: 2418 / 10000 - 24.18 %
Epoch 4 training complete
Cost on training data: 528.723860216563
Cost on evaluation data: 528.7258179191522
Accuracy on evaluation data: 4128 / 10000 - 41.28 %
Epoch 5 training complete
Cost on training data: 532.6502748476829
Cost on evaluation data: 532.6182813388853
Accuracy on evaluation data: 4712 / 10000 - 47.12 %
Epoch 6 training complete
Cost on training data: 521.3842200144529
Cost on evaluation data: 521.3364761684155
Accuracy on evaluation data: 4490 / 10000 - 44.90 %
Epoch 7 training complete
Cost on training data: 503.02286782833204
Cost on evaluation data: 502.99511899397316
Accuracy on evaluation data: 4708 / 10000 - 47.08 %
Epoch 8 training complete
Cost on training data: 491.33995989095274
Cost on evaluation data: 491.3104385110864
Accuracy on evaluation data: 4505 / 10000 - 45.05 %
Epoch 9 training complete
Cost on training data: 515.6648809257151
Cost on evaluation data: 515.6258513514307
Accuracy on evaluation data: 4300 / 10000 - 43.00 %
Epoch 10 training complete
Cost on training data: 481.06465028574524
Cost on evaluation data: 481.0562649330806
Accuracy on evaluation data: 1804 / 10000 - 18.04 %

```

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 5.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 125.45964125087728
Cost on evaluation data: 125.4147157586893
Accuracy on evaluation data: 1375 / 10000 - 13.75 %
Epoch 2 training complete
Cost on training data: 143.20768398294405
Cost on evaluation data: 143.14704054407005
Accuracy on evaluation data: 1898 / 10000 - 18.98 %
Epoch 3 training complete
Cost on training data: 178.87470864629236
Cost on evaluation data: 178.88312078229242
Accuracy on evaluation data: 1037 / 10000 - 10.37 %
Epoch 4 training complete
Cost on training data: 161.87298233422328
Cost on evaluation data: 161.8722964353552
Accuracy on evaluation data: 4131 / 10000 - 41.31 %
Epoch 5 training complete
Cost on training data: 143.35530392378877
Cost on evaluation data: 143.3614836403945
Accuracy on evaluation data: 1235 / 10000 - 12.35 %
Epoch 6 training complete
Cost on training data: 138.55272900674467
Cost on evaluation data: 138.5159234551091
Accuracy on evaluation data: 3950 / 10000 - 39.50 %
Epoch 7 training complete
Cost on training data: 160.59318410262804
Cost on evaluation data: 160.644599878538
Accuracy on evaluation data: 1128 / 10000 - 11.28 %
Epoch 8 training complete
Cost on training data: 137.24822463874295
Cost on evaluation data: 137.23293975871516
Accuracy on evaluation data: 1675 / 10000 - 16.75 %
Epoch 9 training complete
Cost on training data: 155.37370484037555
Cost on evaluation data: 155.3626923624651
Accuracy on evaluation data: 3531 / 10000 - 35.31 %
Epoch 10 training complete
Cost on training data: 126.14974473340864
Cost on evaluation data: 126.1603380971365
Accuracy on evaluation data: 2703 / 10000 - 27.03 %
(base)

```



```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.01, lambda = 200.0
epochs = 3, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1202.2527380598885
Cost on evaluation data: 1202.195831928694
Accuracy on evaluation data: 8279 / 10000 - 82.79 %
Epoch 2 training complete
Cost on training data: 1661.266511051923
Cost on evaluation data: 1661.214236935451
Accuracy on evaluation data: 8114 / 10000 - 81.14 %
Epoch 3 training complete
Cost on training data: 2633.1118753550672
Cost on evaluation data: 2633.0655863453258
Accuracy on evaluation data: 7469 / 10000 - 74.69 %
(base)

```

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lambda = 500.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1410.6616417158805
Cost on evaluation data: 1410.6308153339835
Accuracy on evaluation data: 7903 / 10000 - 79.03 %
Epoch 2 training complete
Cost on training data: 1298.381319343652
Cost on evaluation data: 1298.3497255895547
Accuracy on evaluation data: 7642 / 10000 - 76.42 %
Epoch 3 training complete
Cost on training data: 1302.6975987144617
Cost on evaluation data: 1302.6612950662613
Accuracy on evaluation data: 7223 / 10000 - 72.23 %
Epoch 4 training complete
Cost on training data: 1305.8280292310767
Cost on evaluation data: 1305.794719545183
Accuracy on evaluation data: 7719 / 10000 - 77.19 %
Epoch 5 training complete
Cost on training data: 1305.1665865333819
Cost on evaluation data: 1305.1339981315518
Accuracy on evaluation data: 7486 / 10000 - 74.86 %

```

```
Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.05, lambda = 500.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 932.5416989577024
Cost on evaluation data: 932.5154774262621
Accuracy on evaluation data: 3733 / 10000 - 37.33 %
Epoch 2 training complete
Cost on training data: 927.7213226968216
Cost on evaluation data: 927.6924474955703
Accuracy on evaluation data: 4282 / 10000 - 42.82 %
Epoch 3 training complete
Cost on training data: 946.0553114871167
Cost on evaluation data: 946.0322264708162
Accuracy on evaluation data: 5775 / 10000 - 57.75 %
Epoch 4 training complete
Cost on training data: 962.719477175885
Cost on evaluation data: 962.6978679830194
Accuracy on evaluation data: 5722 / 10000 - 57.22 %
Epoch 5 training complete
Cost on training data: 974.0351727252263
Cost on evaluation data: 974.0102011123657
Accuracy on evaluation data: 5267 / 10000 - 52.67 %
(base)
```

```
Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lambda = 50.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1289.3945746022318
Cost on evaluation data: 1289.350118486389
Accuracy on evaluation data: 7754 / 10000 - 77.54 %
Epoch 2 training complete
Cost on training data: 1961.4886872144261
Cost on evaluation data: 1961.4178184496684
Accuracy on evaluation data: 4312 / 10000 - 43.12 %
Epoch 3 training complete
Cost on training data: 2060.670022171471
Cost on evaluation data: 2060.6369301685713
Accuracy on evaluation data: 4073 / 10000 - 40.73 %
Epoch 4 training complete
Cost on training data: 2206.685208635652
Cost on evaluation data: 2206.6527595199755
Accuracy on evaluation data: 2664 / 10000 - 26.64 %
Epoch 5 training complete
Cost on training data: 2412.0105173183892
Cost on evaluation data: 2412.0084951734943
Accuracy on evaluation data: 5124 / 10000 - 51.24 %
```

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lambda = 1000.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1063.291554682884
Cost on evaluation data: 1063.2818333643706
Accuracy on evaluation data: 4619 / 10000 - 46.19 %
Epoch 2 training complete
Cost on training data: 986.3526437960329
Cost on evaluation data: 986.3434710043481
Accuracy on evaluation data: 4675 / 10000 - 46.75 %
Epoch 3 training complete
Cost on training data: 962.771190160282
Cost on evaluation data: 962.7615881582369
Accuracy on evaluation data: 4931 / 10000 - 49.31 %
Epoch 4 training complete
Cost on training data: 940.0093623241056
Cost on evaluation data: 939.9966261440505
Accuracy on evaluation data: 5337 / 10000 - 53.37 %
Epoch 5 training complete
Cost on training data: 933.2488229194224
Cost on evaluation data: 933.2362697372441
Accuracy on evaluation data: 4785 / 10000 - 47.85 %
(base)

```

```
Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lmbda = 700.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1457.5544069408927
Cost on evaluation data: 1457.5317310677617
Accuracy on evaluation data: 5907 / 10000 - 59.07 %
Epoch 2 training complete
Cost on training data: 1301.9509719138448
Cost on evaluation data: 1301.9274372524517
Accuracy on evaluation data: 6318 / 10000 - 63.18 %
Epoch 3 training complete
Cost on training data: 1254.6797950665139
Cost on evaluation data: 1254.657769095939
Accuracy on evaluation data: 6554 / 10000 - 65.54 %
Epoch 4 training complete
Cost on training data: 1250.610698566401
Cost on evaluation data: 1250.5880132123393
Accuracy on evaluation data: 6508 / 10000 - 65.08 %
Epoch 5 training complete
Cost on training data: 1233.2805103602711
Cost on evaluation data: 1233.25980164148
Accuracy on evaluation data: 7247 / 10000 - 72.47 %
(base)
```

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lambda = 700.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1355.2866697931722
Cost on evaluation data: 1355.2700848609506
Accuracy on evaluation data: 6662 / 10000 - 66.62 %
Epoch 2 training complete
Cost on training data: 1235.6205018952705
Cost on evaluation data: 1235.6027389075464
Accuracy on evaluation data: 7007 / 10000 - 70.07 %
Epoch 3 training complete
Cost on training data: 1224.1614230499079
Cost on evaluation data: 1224.1438344354547
Accuracy on evaluation data: 6602 / 10000 - 66.02 %
Epoch 4 training complete
Cost on training data: 1202.2237280709423
Cost on evaluation data: 1202.2052214523587
Accuracy on evaluation data: 6030 / 10000 - 60.30 %
Epoch 5 training complete
Cost on training data: 1202.5372325535106
Cost on evaluation data: 1202.5181795003539
Accuracy on evaluation data: 6362 / 10000 - 63.62 %
Epoch 6 training complete
Cost on training data: 1199.004409344362
Cost on evaluation data: 1198.9867602839884
Accuracy on evaluation data: 6241 / 10000 - 62.41 %
Epoch 7 training complete
Cost on training data: 1188.6348113387342
Cost on evaluation data: 1188.6163802906228
Accuracy on evaluation data: 6302 / 10000 - 63.02 %
Epoch 8 training complete
Cost on training data: 1194.929351302167
Cost on evaluation data: 1194.9069334488197
Accuracy on evaluation data: 5737 / 10000 - 57.37 %
Epoch 9 training complete
Cost on training data: 1193.6398768540566
Cost on evaluation data: 1193.6210863929414
Accuracy on evaluation data: 6994 / 10000 - 69.94 %
Epoch 10 training complete
Cost on training data: 1191.03555601771
Cost on evaluation data: 1191.015059664025
Accuracy on evaluation data: 6888 / 10000 - 68.88 %

```



```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.005, lmbda = 300.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1587.4588881493157
Cost on evaluation data: 1587.4138639351204
Accuracy on evaluation data: 7842 / 10000 - 78.42 %
Epoch 2 training complete
Cost on training data: 1230.9523071066637
Cost on evaluation data: 1230.9098635607425
Accuracy on evaluation data: 8358 / 10000 - 83.58 %
Epoch 3 training complete
Cost on training data: 1269.5634127027658
Cost on evaluation data: 1269.514498733904
Accuracy on evaluation data: 8038 / 10000 - 80.38 %
Epoch 4 training complete
Cost on training data: 1306.4337513222401
Cost on evaluation data: 1306.387841592198
Accuracy on evaluation data: 8297 / 10000 - 82.97 %
Epoch 5 training complete
Cost on training data: 1332.8608832704317
Cost on evaluation data: 1332.811733350246
Accuracy on evaluation data: 8578 / 10000 - 85.78 %
Epoch 6 training complete
Cost on training data: 1351.7212685645698
Cost on evaluation data: 1351.6746444594567
Accuracy on evaluation data: 8544 / 10000 - 85.44 %
Epoch 7 training complete
Cost on training data: 1428.6066101029996
Cost on evaluation data: 1428.5518756383597
Accuracy on evaluation data: 8296 / 10000 - 82.96 %
Epoch 8 training complete
Cost on training data: 1500.0675220103174
Cost on evaluation data: 1500.017801317904
Accuracy on evaluation data: 8356 / 10000 - 83.56 %
Epoch 9 training complete
Cost on training data: 1573.7228083393486
Cost on evaluation data: 1573.673794625503
Accuracy on evaluation data: 8410 / 10000 - 84.10 %
Epoch 10 training complete
Cost on training data: 1657.8708756982032
Cost on evaluation data: 1657.8245643413582
Accuracy on evaluation data: 8340 / 10000 - 83.40 %

```

Καταφέραμε να βρούμε τη χρυσή τομή για  $\eta = 0.005$ ,  $\lambda = 300.0$ !!!

Στη συνέχεια βλέπουμε ότι το ίδιο περίπου συμβαίνει αφαιρώντας ένα κρυφό στρώμα, με το νευρωνικό να πιάνει τελικά τις καλύτερες αποδόσεις για  $\eta = 0.001$ ,  $\lambda = 300.0$ .

```
Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lambda = 5.0
epochs = 3, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1847.64016734619
Cost on evaluation data: 1847.6389492436106
Accuracy on evaluation data: 1959 / 10000 - 19.59 %
Epoch 2 training complete
Cost on training data: 2201.7445409646016
Cost on evaluation data: 2201.7473668964885
Accuracy on evaluation data: 1437 / 10000 - 14.37 %
Epoch 3 training complete
Cost on training data: 1170.7936612199571
Cost on evaluation data: 1170.7936612206872
Accuracy on evaluation data: 1661 / 10000 - 16.61 %
```

```
Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lambda = 50.0
epochs = 3, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 376.39984995921446
Cost on evaluation data: 376.35197768037506
Accuracy on evaluation data: 4469 / 10000 - 44.69 %
Epoch 2 training complete
Cost on training data: 365.341716908209
Cost on evaluation data: 365.30263027963383
Accuracy on evaluation data: 4428 / 10000 - 44.28 %
Epoch 3 training complete
Cost on training data: 351.56943341538647
Cost on evaluation data: 351.55747595999435
Accuracy on evaluation data: 3169 / 10000 - 31.69 %
```

```

Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.1, lambda = 300.0
epochs = 3, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 685.3010739956188
Cost on evaluation data: 685.2640908279487
Accuracy on evaluation data: 6219 / 10000 - 62.19 %
Epoch 2 training complete
Cost on training data: 664.8518746378212
Cost on evaluation data: 664.8137905791467
Accuracy on evaluation data: 6796 / 10000 - 67.96 %
Epoch 3 training complete
Cost on training data: 651.151893993118
Cost on evaluation data: 651.116170935752
Accuracy on evaluation data: 6304 / 10000 - 63.04 %
(base)

```

```

Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 0.0001, lambda = 300.0
epochs = 3, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 5576.337169783114
Cost on evaluation data: 5576.298479001882
Accuracy on evaluation data: 4333 / 10000 - 43.33 %
Epoch 2 training complete
Cost on training data: 5207.106837516271
Cost on evaluation data: 5207.065364973813
Accuracy on evaluation data: 5984 / 10000 - 59.84 %
Epoch 3 training complete
Cost on training data: 4889.453117948391
Cost on evaluation data: 4889.412774073147
Accuracy on evaluation data: 6921 / 10000 - 69.21 %

```

```

Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.001, lmbda = 300.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 3280.434674686462
Cost on evaluation data: 3280.383828493165
Accuracy on evaluation data: 8340 / 10000 - 83.40 %
Epoch 2 training complete
Cost on training data: 2006.7928554924235
Cost on evaluation data: 2006.736370571751
Accuracy on evaluation data: 8497 / 10000 - 84.97 %
Epoch 3 training complete
Cost on training data: 1356.0027010152887
Cost on evaluation data: 1355.9474832779104
Accuracy on evaluation data: 8543 / 10000 - 85.43 %
Epoch 4 training complete
Cost on training data: 1023.7296158707059
Cost on evaluation data: 1023.6773979933441
Accuracy on evaluation data: 8627 / 10000 - 86.27 %
Epoch 5 training complete
Cost on training data: 852.3214789073423
Cost on evaluation data: 852.2657161756962
Accuracy on evaluation data: 8584 / 10000 - 85.84 %
Epoch 6 training complete
Cost on training data: 767.3044425364125
Cost on evaluation data: 767.2509743799654
Accuracy on evaluation data: 8641 / 10000 - 86.41 %
Epoch 7 training complete
Cost on training data: 732.1194876033544
Cost on evaluation data: 732.0631148050861
Accuracy on evaluation data: 8647 / 10000 - 86.47 %
Epoch 8 training complete
Cost on training data: 721.7302153238356
Cost on evaluation data: 721.6775888560011
Accuracy on evaluation data: 8618 / 10000 - 86.18 %
Epoch 9 training complete
Cost on training data: 723.1894540646408
Cost on evaluation data: 723.1371124151057
Accuracy on evaluation data: 8632 / 10000 - 86.32 %
Epoch 10 training complete
Cost on training data: 725.8969101648752
Cost on evaluation data: 725.8423546460958
Accuracy on evaluation data: 8659 / 10000 - 86.59 %
(base)

```

Για τα υπόλοιπα πειράματα η συνάρτηση ενεργοποίησης έχει αλλάξει σε σιγμοειδή. Εδώ παρατήρησα ότι το νευρωνικό τα πήγαινε σαφώς καλύτερα με ποσοστά που πολλές φορές έφταναν το 95% σε όλα σχεδόν τα configurations. Ενδεικτικά παραθέτω κάποια:

```

Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 5.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 8431.631095738345
Accuracy on training data: 46881 / 50000 - 93.76 %
Cost on evaluation data: 8431.618352964251
Accuracy on evaluation data: 9387 / 10000 - 93.87 %
Epoch 2 training complete
Cost on training data: 9954.374049438326
Accuracy on training data: 47235 / 50000 - 94.47 %
Cost on evaluation data: 9954.387351191368
Accuracy on evaluation data: 9429 / 10000 - 94.29 %
Epoch 3 training complete
Cost on training data: 10328.381829038888
Accuracy on training data: 46987 / 50000 - 93.97 %
Cost on evaluation data: 10328.382804594063
Accuracy on evaluation data: 9407 / 10000 - 94.07 %
Epoch 4 training complete
Cost on training data: 10385.274708150031
Accuracy on training data: 47492 / 50000 - 94.98 %
Cost on evaluation data: 10385.283774626221
Accuracy on evaluation data: 9483 / 10000 - 94.83 %
Epoch 5 training complete
Cost on training data: 10616.250111487363
Accuracy on training data: 46937 / 50000 - 93.87 %
Cost on evaluation data: 10616.251485735154
Accuracy on evaluation data: 9398 / 10000 - 93.98 %

```

```
Network: [784, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.01, lambda = 300.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 36488.41795287423
Accuracy on training data: 41903 / 50000 - 83.81 %
Cost on evaluation data: 36488.37808890653
Accuracy on evaluation data: 8529 / 10000 - 85.29 %
Epoch 2 training complete
Cost on training data: 40850.97494134909
Accuracy on training data: 42814 / 50000 - 85.63 %
Cost on evaluation data: 40850.93322120753
Accuracy on evaluation data: 8697 / 10000 - 86.97 %
Epoch 3 training complete
Cost on training data: 41671.75193929602
Accuracy on training data: 43572 / 50000 - 87.14 %
Cost on evaluation data: 41671.7095826283
Accuracy on evaluation data: 8837 / 10000 - 88.37 %
Epoch 4 training complete
Cost on training data: 41894.84043693517
Accuracy on training data: 43777 / 50000 - 87.55 %
Cost on evaluation data: 41894.796582734445
Accuracy on evaluation data: 8896 / 10000 - 88.96 %
Epoch 5 training complete
Cost on training data: 42048.25624491073
Accuracy on training data: 43446 / 50000 - 86.89 %
Cost on evaluation data: 42048.21298374091
Accuracy on evaluation data: 8806 / 10000 - 88.06 %
```



```
Network: [784, 100, 30, 16, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 5.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 10593.123712948223
Accuracy on training data: 45946 / 50000 - 91.89 %
Cost on evaluation data: 10593.118230121936
Accuracy on evaluation data: 9228 / 10000 - 92.28 %
Epoch 2 training complete
Cost on training data: 12795.627652999035
Accuracy on training data: 47532 / 50000 - 95.06 %
Cost on evaluation data: 12795.626974451623
Accuracy on evaluation data: 9521 / 10000 - 95.21 %
Epoch 3 training complete
Cost on training data: 13173.534095252404
Accuracy on training data: 47514 / 50000 - 95.03 %
Cost on evaluation data: 13173.54768797983
Accuracy on evaluation data: 9504 / 10000 - 95.04 %
Epoch 4 training complete
Cost on training data: 13482.880317926967
Accuracy on training data: 48098 / 50000 - 96.20 %
Cost on evaluation data: 13482.891740209303
Accuracy on evaluation data: 9605 / 10000 - 96.05 %
Epoch 5 training complete
Cost on training data: 13584.76563928962
Accuracy on training data: 48322 / 50000 - 96.64 %
Cost on evaluation data: 13584.787039342773
Accuracy on evaluation data: 9635 / 10000 - 96.35 %
(base)
```

```

$ python test.py
Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lambda = 5.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 6323.203273228205
Accuracy on training data: 46607 / 50000 - 93.21 %
Cost on evaluation data: 6323.182464846805
Accuracy on evaluation data: 9357 / 10000 - 93.57 %
Epoch 2 training complete
Cost on training data: 7306.348104455936
Accuracy on training data: 47348 / 50000 - 94.70 %
Cost on evaluation data: 7306.360745612709
Accuracy on evaluation data: 9466 / 10000 - 94.66 %
Epoch 3 training complete
Cost on training data: 7704.692316587288
Accuracy on training data: 47315 / 50000 - 94.63 %
Cost on evaluation data: 7704.695996816349
Accuracy on evaluation data: 9460 / 10000 - 94.60 %
Epoch 4 training complete
Cost on training data: 7839.882407473323
Accuracy on training data: 46631 / 50000 - 93.26 %
Cost on evaluation data: 7839.878668398039
Accuracy on evaluation data: 9338 / 10000 - 93.38 %
Epoch 5 training complete
Cost on training data: 8014.128666107113
Accuracy on training data: 47061 / 50000 - 94.12 %
Cost on evaluation data: 8014.123371233496
Accuracy on evaluation data: 9408 / 10000 - 94.08 %
(base)

```

Το νευρωνικό τα έχει πάει πολύ καλά και με τις ουδέτερες τιμές των παραμέτρων  $\eta$ ,  $\lambda$  με  $\eta = 1$ ,  $\lambda = 0.0$ , πιάνοντας ποσοστά σχεδόν 90%:

```

$ python test.py
Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning_rate = 1.0, lmbda = 0.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 1.7257936709401145
Accuracy on training data: 33983 / 50000 - 67.97 %
Cost on evaluation data: 1.5350511365409072
Accuracy on evaluation data: 7200 / 10000 - 72.00 %
Epoch 2 training complete
Cost on training data: 1.2037864157991864
Accuracy on training data: 42012 / 50000 - 84.02 %
Cost on evaluation data: 1.1242586424243273
Accuracy on evaluation data: 8546 / 10000 - 85.46 %
Epoch 3 training complete
Cost on training data: 0.9303747116444784
Accuracy on training data: 43192 / 50000 - 86.38 %
Cost on evaluation data: 0.8752208148206786
Accuracy on evaluation data: 8722 / 10000 - 87.22 %
Epoch 4 training complete
Cost on training data: 0.9912470014365637
Accuracy on training data: 42425 / 50000 - 84.85 %
Cost on evaluation data: 0.9332836240978031
Accuracy on evaluation data: 8597 / 10000 - 85.97 %
Epoch 5 training complete
Cost on training data: 0.9209244323856051
Accuracy on training data: 43649 / 50000 - 87.30 %
Cost on evaluation data: 0.8799927418654107
Accuracy on evaluation data: 8826 / 10000 - 88.26 %
Epoch 6 training complete
Cost on training data: 0.7922746366414621
Accuracy on training data: 44460 / 50000 - 88.92 %
Cost on evaluation data: 0.7611111949796995
Accuracy on evaluation data: 8946 / 10000 - 89.46 %
Epoch 7 training complete
Cost on training data: 0.7581857053706758
Accuracy on training data: 44433 / 50000 - 88.87 %
Cost on evaluation data: 0.7058980150293338
Accuracy on evaluation data: 9005 / 10000 - 90.05 %
Epoch 8 training complete
Cost on training data: 0.8483605022021997
Accuracy on training data: 42860 / 50000 - 85.72 %
Cost on evaluation data: 0.7897677805167173
Accuracy on evaluation data: 8697 / 10000 - 86.97 %
Epoch 9 training complete
Cost on training data: 0.9892670896777038
Accuracy on training data: 43016 / 50000 - 86.03 %
Cost on evaluation data: 0.953376714393446
Accuracy on evaluation data: 8673 / 10000 - 86.73 %
Epoch 10 training complete
Cost on training data: 0.7398402053086155
Accuracy on training data: 44392 / 50000 - 88.78 %
Cost on evaluation data: 0.7124677826845623
Accuracy on evaluation data: 8938 / 10000 - 89.38 %
(chase)

```

Παρακάτω η καλύτερη απόδοση που κατάφερα να πιάσω χρησιμοποιώντας ολόκληρο το dataset:

```

Network: [784, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 0.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Epoch 1 training complete
Cost on training data: 0.3887286124410595
Accuracy on training data: 47144 / 50000 - 94.29 %
Cost on evaluation data: 0.3835792245779938
Accuracy on evaluation data: 9435 / 10000 - 94.35 %
Epoch 2 training complete
Cost on training data: 0.35133006679671935
Accuracy on training data: 47464 / 50000 - 94.93 %
Cost on evaluation data: 0.3605345930801014
Accuracy on evaluation data: 9480 / 10000 - 94.80 %
Epoch 3 training complete
Cost on training data: 0.31390425069209843
Accuracy on training data: 47784 / 50000 - 95.57 %
Cost on evaluation data: 0.33742880592188007
Accuracy on evaluation data: 9542 / 10000 - 95.42 %
Epoch 4 training complete
Cost on training data: 0.3061955220419118
Accuracy on training data: 47842 / 50000 - 95.68 %
Cost on evaluation data: 0.33815205746806315
Accuracy on evaluation data: 9534 / 10000 - 95.34 %
Epoch 5 training complete
Cost on training data: 0.28105004198182393
Accuracy on training data: 48025 / 50000 - 96.05 %
Cost on evaluation data: 0.3290326643218983
Accuracy on evaluation data: 9532 / 10000 - 95.32 %
Epoch 6 training complete
Cost on training data: 0.2875245808763236
Accuracy on training data: 48020 / 50000 - 96.04 %
Cost on evaluation data: 0.33154834268650485
Accuracy on evaluation data: 9548 / 10000 - 95.48 %
Epoch 7 training complete
Cost on training data: 0.2607255622105985
Accuracy on training data: 48200 / 50000 - 96.40 %
Cost on evaluation data: 0.3304032334644026
Accuracy on evaluation data: 9552 / 10000 - 95.52 %
Epoch 8 training complete
Cost on training data: 0.2605089075347564
Accuracy on training data: 48185 / 50000 - 96.37 %
Cost on evaluation data: 0.32249018316673994
Accuracy on evaluation data: 9552 / 10000 - 95.52 %
Epoch 9 training complete
Cost on training data: 0.25257160500746084
Accuracy on training data: 48278 / 50000 - 96.56 %
Cost on evaluation data: 0.31980021389031393
Accuracy on evaluation data: 9574 / 10000 - 95.74 %
Epoch 10 training complete
Cost on training data: 0.21779602888964122
Accuracy on training data: 48550 / 50000 - 97.10 %
Cost on evaluation data: 0.30617652755362396
Accuracy on evaluation data: 9585 / 10000 - 95.85 %

```

Τέλος, έκανα κάποια πειράματα με δεδομένα μειωμένης διάστασης. Αρχικά δοκίμασα να μειώσω την διάσταση των φωτογραφιών απο 784 σε 10, κάτι που όπως ήταν αναμενόμενο δεν πήγε καλά:

```
Network: [10, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.01, lmbda = 300.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=False
Reducing data dimensionality using PSA to 10 dimensions
Epoch 1 training complete
Cost on training data: 457.57279373142705
Accuracy on evaluation data: 2265 / 10000 - 22.65 %
Epoch 2 training complete
Cost on training data: 454.674303694366
Accuracy on evaluation data: 2230 / 10000 - 22.30 %
Epoch 3 training complete
Cost on training data: 445.8338304114528
Accuracy on evaluation data: 2713 / 10000 - 27.13 %
Epoch 4 training complete
Cost on training data: 454.1863124517965
Accuracy on evaluation data: 2858 / 10000 - 28.58 %
Epoch 5 training complete
Cost on training data: 447.0598540926789
Accuracy on evaluation data: 2683 / 10000 - 26.83 %
(base)
```



```

Network: [10, 16, 16, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.01, lmbda = 300.0
epochs = 10, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Reducing data dimensionality using PSA to 10 dimensions
Epoch 1 training complete
Cost on training data: 1577.5833909229289
Cost on evaluation data: 1578.226050581201
Accuracy on evaluation data: 2468 / 10000 - 24.68 %
Epoch 2 training complete
Cost on training data: 720.5494291393676
Cost on evaluation data: 721.17652588341
Accuracy on evaluation data: 2364 / 10000 - 23.64 %
Epoch 3 training complete
Cost on training data: 425.8142287078324
Cost on evaluation data: 426.4515303402503
Accuracy on evaluation data: 2662 / 10000 - 26.62 %
Epoch 4 training complete
Cost on training data: 422.76031248689986
Cost on evaluation data: 423.4609373276995
Accuracy on evaluation data: 2160 / 10000 - 21.60 %
Epoch 5 training complete
Cost on training data: 436.6900817205091
Cost on evaluation data: 437.3307737482443
Accuracy on evaluation data: 2586 / 10000 - 25.86 %
Epoch 6 training complete
Cost on training data: 437.6526170890536
Cost on evaluation data: 438.3323963459986
Accuracy on evaluation data: 2434 / 10000 - 24.34 %
Epoch 7 training complete
Cost on training data: 453.2261331639438
Cost on evaluation data: 453.90263615708767
Accuracy on evaluation data: 2683 / 10000 - 26.83 %
Epoch 8 training complete
Cost on training data: 447.72049192905405
Cost on evaluation data: 448.3360106919107
Accuracy on evaluation data: 2476 / 10000 - 24.76 %
Epoch 9 training complete
Cost on training data: 445.40748301261027
Cost on evaluation data: 446.037193716027
Accuracy on evaluation data: 2951 / 10000 - 29.51 %
Epoch 10 training complete
Cost on training data: 453.60069140886145
Cost on evaluation data: 454.27450943521956
Accuracy on evaluation data: 2406 / 10000 - 24.06 %
(base)

```

Βλέπουμε όμως, ότι με 50 διαστάσεις πρακτικά δεν χάνουμε σχεδόν καθόλου πληροφορία, καθώς πιάνουμε παρόμοια ποσοστά με της πλήρους διάστασης, κοντά στο 95%:

```

$ python test.py
Network: [50, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 5.0
epochs = 15, batch size = 10
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Reducing data dimensionality using PCA to 50 dimensions
Epoch 1 training complete
Cost on training data: 1436.6660984359892
Accuracy on training data: 45456 / 50000 - 90.91 %
Cost on evaluation data: 1436.6209156521186
Accuracy on evaluation data: 9203 / 10000 - 92.03 %
Epoch 2 training complete
Cost on training data: 1946.229400877275
Accuracy on training data: 46266 / 50000 - 92.53 %
Cost on evaluation data: 1946.1984652970411
Accuracy on evaluation data: 9334 / 10000 - 93.34 %
Epoch 3 training complete
Cost on training data: 2328.716407360361
Accuracy on training data: 46771 / 50000 - 93.54 %
Cost on evaluation data: 2328.698776966982
Accuracy on evaluation data: 9411 / 10000 - 94.11 %
Epoch 4 training complete
Cost on training data: 2640.22930608866
Accuracy on training data: 47099 / 50000 - 94.20 %
Cost on evaluation data: 2640.21734208106
Accuracy on evaluation data: 9467 / 10000 - 94.67 %
Epoch 5 training complete
Cost on training data: 2883.4805372734418
Accuracy on training data: 47326 / 50000 - 94.65 %
Cost on evaluation data: 2883.4706447845074
Accuracy on evaluation data: 9485 / 10000 - 94.85 %
Epoch 6 training complete
Cost on training data: 3094.171285829348
Accuracy on training data: 47323 / 50000 - 94.65 %
Cost on evaluation data: 3094.1680200580035
Accuracy on evaluation data: 9492 / 10000 - 94.92 %
Epoch 7 training complete
Cost on training data: 3259.1417275363747
Accuracy on training data: 47448 / 50000 - 94.90 %
Cost on evaluation data: 3259.1421987053195
Accuracy on evaluation data: 9504 / 10000 - 95.04 %
Epoch 8 training complete
Cost on training data: 3394.964506302004
Accuracy on training data: 47587 / 50000 - 95.17 %
Cost on evaluation data: 3394.964370285824
Accuracy on evaluation data: 9544 / 10000 - 95.44 %
Epoch 9 training complete
Cost on training data: 3497.9403616428085
Accuracy on training data: 47687 / 50000 - 95.37 %
Cost on evaluation data: 3497.941711858645
Accuracy on evaluation data: 9543 / 10000 - 95.43 %
Epoch 10 training complete
Cost on training data: 3596.0429000611457
Accuracy on training data: 47737 / 50000 - 95.47 %
Cost on evaluation data: 3596.0463007692147
Accuracy on evaluation data: 9557 / 10000 - 95.57 %
Epoch 11 training complete
Cost on training data: 3675.4642298662766
Accuracy on training data: 47715 / 50000 - 95.43 %
Cost on evaluation data: 3675.4668664989085
Accuracy on evaluation data: 9566 / 10000 - 95.66 %

```

```

$ python test.py
Network: [50, 30, 16], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lmbda = 5.0
epochs = 15, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Reducing data dimensionality using PCA to 50 dimensions
Epoch 1 training complete
Cost on training data: 4579.062002610595
Accuracy on training data: 46854 / 50000 - 93.71 %
Cost on evaluation data: 4579.042931072967
Accuracy on evaluation data: 9434 / 10000 - 94.34 %
Epoch 2 training complete
Cost on training data: 5400.798807616725
Accuracy on training data: 47224 / 50000 - 94.45 %
Cost on evaluation data: 5400.79402735553
Accuracy on evaluation data: 9459 / 10000 - 94.59 %
Epoch 3 training complete
Cost on training data: 5580.22848054557
Accuracy on training data: 47117 / 50000 - 94.23 %
Cost on evaluation data: 5580.229485625231
Accuracy on evaluation data: 9432 / 10000 - 94.32 %
Epoch 4 training complete
Cost on training data: 5731.175049672695
Accuracy on training data: 47146 / 50000 - 94.29 %
Cost on evaluation data: 5731.17590999895
Accuracy on evaluation data: 9449 / 10000 - 94.49 %
Epoch 5 training complete
Cost on training data: 5767.1516754024615
Accuracy on training data: 47436 / 50000 - 94.87 %
Cost on evaluation data: 5767.156784661065
Accuracy on evaluation data: 9485 / 10000 - 94.85 %
Epoch 6 training complete
Cost on training data: 5963.202954943454
Accuracy on training data: 47303 / 50000 - 94.61 %
Cost on evaluation data: 5963.204673343584
Accuracy on evaluation data: 9480 / 10000 - 94.80 %
Epoch 7 training complete
Cost on training data: 5927.46400252686
Accuracy on training data: 47375 / 50000 - 94.75 %
Cost on evaluation data: 5927.454097515526
Accuracy on evaluation data: 9497 / 10000 - 94.97 %
Epoch 8 training complete
Cost on training data: 5929.425687001056
Accuracy on training data: 47366 / 50000 - 94.73 %
Cost on evaluation data: 5929.431795970656
Accuracy on evaluation data: 9471 / 10000 - 94.71 %
Epoch 9 training complete
Cost on training data: 5908.111913164279
Accuracy on training data: 47440 / 50000 - 94.88 %
Cost on evaluation data: 5908.113669776185
Accuracy on evaluation data: 9483 / 10000 - 94.83 %
Epoch 10 training complete
Cost on training data: 5957.829315145626
Accuracy on training data: 47377 / 50000 - 94.75 %
Cost on evaluation data: 5957.8327175866925
Accuracy on evaluation data: 9497 / 10000 - 94.97 %
Epoch 11 training complete
Cost on training data: 5972.073548874169
Accuracy on training data: 47590 / 50000 - 95.18 %
Cost on evaluation data: 5972.074536033908
Accuracy on evaluation data: 9527 / 10000 - 95.27 %

```

Το 16 στο τελευταίο layer της προηγούμενης φωτογραφίας είναι typo.

```
$ python test.py
Network: [50, 30, 30, 10], cost function = CrossEntropyCost
training_data: 50000, evaluation_data: 10000
learning rate = 0.1, lambda = 5.0
epochs = 5, batch size = 1
monitor_evaluation_accuracy=True
monitor_training_accuracy=True
monitor_evaluation_cost=True
Reducing data dimensionality using PCA to 50 dimensions
Epoch 1 training complete
Cost on training data: 6559.192948603865
Accuracy on training data: 46775 / 50000 - 93.55 %
Cost on evaluation data: 6559.171230235461
Accuracy on evaluation data: 9404 / 10000 - 94.04 %
Epoch 2 training complete
Cost on training data: 7575.209082275742
Accuracy on training data: 47104 / 50000 - 94.21 %
Cost on evaluation data: 7575.2017394074455
Accuracy on evaluation data: 9453 / 10000 - 94.53 %
Epoch 3 training complete
Cost on training data: 7754.232804066392
Accuracy on training data: 47411 / 50000 - 94.82 %
Cost on evaluation data: 7754.2251680282225
Accuracy on evaluation data: 9488 / 10000 - 94.88 %
Epoch 4 training complete
Cost on training data: 8023.641353070543
Accuracy on training data: 47152 / 50000 - 94.30 %
Cost on evaluation data: 8023.633920640828
Accuracy on evaluation data: 9424 / 10000 - 94.24 %
Epoch 5 training complete
Cost on training data: 8037.535210003179
Accuracy on training data: 47420 / 50000 - 94.84 %
Cost on evaluation data: 8037.5446224480565
Accuracy on evaluation data: 9462 / 10000 - 94.62 %
(100%)
```

## Πορίσματα

Βλέπουμε ότι υπό ορισμένες συνθήκες το νευρωνικό δίκτυο τα έχει πάει πολύ καλά, με ποσοστά κοντά στο 95%. Πολύ σημαντικές ήταν οι τιμές των παραμέτρων **eta**, **lmbda** καθώς πολλές φορές το δίκτυο δεν συνέκλινε παραμόνο με συγκεκριμένες τιμές τους. Ιδιαίτερα σημαντικές ήταν στα πειράματα που χρησιμοποιήθηκε συνάρτηση ενεργοποίησης **ReLU**. Η χρήση της συνάρτησης ενεργοποίησης **ReLU** ήταν σε γενικές γραμμές επιτυχής, με καλά ποσοστά κοντά στο 86%.

Η μείωση διάστασης στις 50 διαστάσεις βλέπουμε ότι ήταν και αυτή επιτυχής, χωρίς να ισχύει το ίδιο για τις 10 διαστάσεις.

Συγκρίνοντας με τα αποτελέσματα της Ενδιάμεσης Εργασίας, παρατηρούμε ότι η βέλτιστη απόδοση αυτού του νευρωνικού δικτύου στο 95% είναι ελάχιστα χαμηλότερη από τις αποδόσεις των **KNeighborClassifier** για αριθμό κοντινότερων γειτόνων **NN=1** και **NN=3**, οι οποίες κυμαίνονταν στο 97%. Δεν κατάφερε παρόλα αυτά να τους ξεπεράσει.

Όσο για τον **NearestCentroid** classifier, βλέπουμε ότι το νευρωνικό δίκτυο αυτό τα έχει πάει καλύτερα από τον **NearestCentroid** σχεδόν σε όλες τις περιπτώσεις που συνέκλινε, με ποσοστά μεγαλύτερα του 85%, σε αντίθεση με τον **NearestCentroid** που ήταν γύρω στο 82%.