



Αριστοτέλειο Πανεπιστήμιο Θεσσαλονίκης  
Νευρωνικά Δίκτυα - Βαθιά Μάθηση

## Εργασία 1

Κούτση Χριστίνα

AEM: 9872

[cvkoutsi@ece.auth.gr](mailto:cvkoutsi@ece.auth.gr)

Δεκέμβριος, 2023

# Περιεχόμενα

<b>1 Φόρτωση και κανονικοποίηση δεδομένων</b>	<b>2</b>
<b>2 Δημιουργία κλάσης RBF Νευρωνικού δικτύου</b>	<b>3</b>
2.1 Αρχικοποίηση κέντρων . . . . .	3
2.2 Ορισμός συναρτήσεων υπολογισμού απόστασης και βοηθητικών συναρτήσεων . . . . .	4
2.3 Εκπαίδευση . . . . .	4
2.4 Predict . . . . .	5
<b>3 Απόδοση του νευρωνικού δικτύου σε δεδομένα εισόδου χωρίς μείωση διάστασης</b>	<b>6</b>
3.1 Αρχικοποίηση κέντρων με τυχαία επιλογή εικόνων . . . . .	6
3.1.1 Χρήση Gauss συνάρτησης ακτινικού τύπου . . . . .	6
3.1.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου . . . . .	7
3.1.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου . . . . .	7
3.2 Αρχικοποίηση κέντρων με χρήση μέσης τιμής εικόνων για κάθε ετικέτα . . . . .	7
3.2.1 Χρήση Gauss συνάρτησης ακτινικού τύπου . . . . .	8
3.2.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου . . . . .	8
3.2.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου . . . . .	8
3.3 Αρχικοποίηση κέντρων με χρήση του αλγορίθμου k-means . . . . .	9
3.3.1 Χρήση Gauss συνάρτησης ακτινικού τύπου . . . . .	9
3.3.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου . . . . .	9
3.3.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου . . . . .	10
<b>4 Απόδοση του νευρωνικού δικτύου σε δεδομένα εισόδου με μείωση διάστασης</b>	<b>11</b>
4.1 Μείωση χαρακτηριστικών με χρήση μέσης φωτεινότητας σειράς . . . . .	11
4.2 Μείωση χαρακτηριστικών με τον αλγόριθμο PCA . . . . .	11

# Κεφάλαιο 1

## Φόρτωση και κανονικοποίηση δεδομένων

Το dataset περιέχει 60.000 παραδείγματα εκπαίδευσης και 10.000 παραδείγματα ελέγχου. Τα πρώτα 5 παραδείγματα εκπαίδευσης φαίνονται παρακάτω:

	label	1x1	1x2	1x3	1x4	1x5	1x6	1x7	1x8	1x9	...	28x19	28x20	\
0	5	0	0	0	0	0	0	0	0	0	...	0	0	
1	0	0	0	0	0	0	0	0	0	0	...	0	0	
2	4	0	0	0	0	0	0	0	0	0	...	0	0	
3	1	0	0	0	0	0	0	0	0	0	...	0	0	
4	9	0	0	0	0	0	0	0	0	0	...	0	0	

	28x21	28x22	28x23	28x24	28x25	28x26	28x27	28x28
0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0

[5 rows x 785 columns]

Βλέπουμε ότι η πρώτη στήλη περιέχει τα labels των εικόνων και οι στήλες 1 έως 784 περιέχουν τα pixels. Παρατηρώ ότι οι εικόνες έχουν ανασχηματιστεί από έναν πίνακα 28 x 28 pixels σε μία στήλη με 784 pixels.

Επομένως, ορίζω ως ετικέτες εκπαίδευσης και ελέγχου τις πρώτες στήλες των δεδομένων εκπαίδευσης και ελέγχου αντίστοιχα και ως δεδομένα εκπαίδευσης και ελέγχου τις υπόλοιπες στήλες. Στη συνέχεια κανονικοποιώ τα δεδομένα διαιρώντας με 255, καθώς αυτή είναι η υψηλότερη τιμή που μπορεί να πάρει η φωτεινότητα ενός pixel.

Εφόσον τα δεδομένα εισόδου έχουν διάσταση 784xM, το στρώμα εισόδου του νευρωνικού δικτύου θα έχει 784 νευρώνες, ενώ εφόσον οι ετικέτες έχουν διάσταση 10xM το

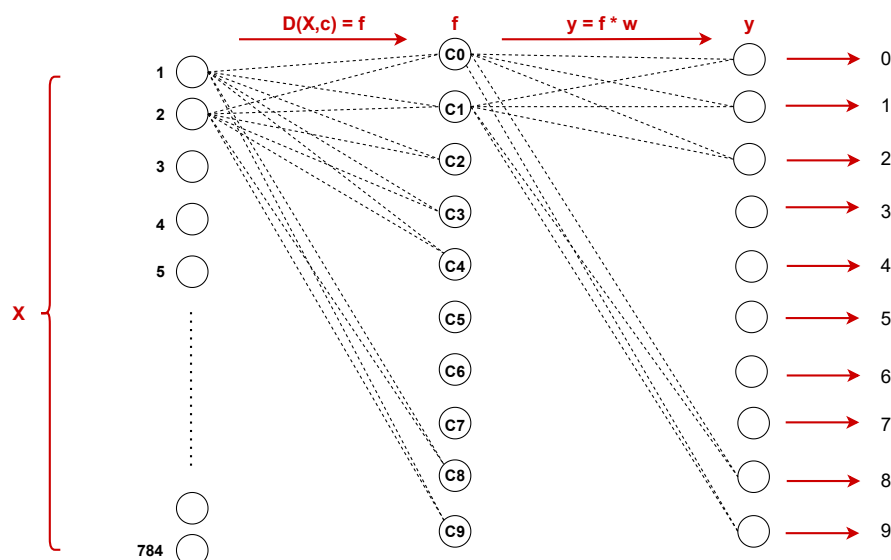
στρώμα εξόδου θα έχει 10 νευρώνες.

## Κεφάλαιο 2

# Δημιουργία κλάσης RBF Νευρωνικού δικτύου

Η παρούσα εργασία έχει αντικείμενο την δημιουργία ενός RBF νευρωνικού δικτύου, με σκοπό την ανίχνευση δεκαδικών ψηφίων από εικόνες χειρόγραφων αριθμών. Το dataset που θα χρησιμοποιηθεί θα είναι το mnist dataset σε csv μορφή

Το νευρωνικό δίκτυο θα δέχεται ως είσοδο τόσο την πλήρη εικόνα διάστασης 28x28 πίξελ όσο και μειωμένη είσοδο μετά από εφαρμογή αλγορίθμου μείωσης διάστασης στην εικόνα. Τα κέντρα  $c$ , δηλαδή οι νευρώνες του κρυφού στρώματος θα μπορούν να αρχικοποιούνται με τρεις διαφορετικούς τρόπους και θα συνδέονται με το στρώμα εισόδου μέσω μιας συνάρτησης  $G$ , η οποία θα υπολογίζει την απόσταση της εικόνας εισόδου από κάθε κέντρο. Στην έξοδο θα έχουμε 10 νευρώνες, έναν για κάθε κλάση.



## 2.1 Αρχικοποίηση κέντρων

Όπως προαναφέρθηκε, μπορούμε να αρχικοποιήσουμε τα κέντρα με τρεις διαφορετικούς τρόπους.

- Τυχαία επιλογή εικόνων κάθε ετικέτας και ορισμός της ως κέντρο: Με αυτόν τον τρόπο, και σύμφωνα με τον αριθμό των κέντρων που επιθυμούμε να έχουμε στο κρυφό στρώμα, επιλέγουμε τυχαία κάποιες εικόνες από το training dataset και τις θέτουμε ως κέντρα. Ο αλγόριθμος φροντίζει να θεωρείται τουλάχιστον μία εικόνα από κάθε ετικέτα ως κέντρο.
- Χρήση μέσης φωτεινότητας εικόνων ίδιας ετικέτας: Η συνάρτηση αρχικοποίησης κέντρων εντοπίζει τις φωτογραφίες με κοινά labels και υπολογίζει τον μέσο όρο της φωτεινότητάς τους. Έτσι για κάθε label υπάρχει μία εικόνα.
- Επιλογή των κέντρων μέσω του αλγορίθμου k-means. Χρησιμοποιείται η βιβλιοθήκη kmeans της sklearn.

## 2.2 Ορισμός συναρτήσεων υπολογισμού απόστασης και βοηθητικών συναρτήσεων

Ορίζουμε τρεις διαφορετικές συναρτήσεις για τον υπολογισμό της απόστασης των κέντρων από την εικόνα εισόδου:

- Gaussian συνάρτηση: Ορίζουμε την συνάρτηση απόστασης από τον τύπο:

$$f(x) = e^{-\frac{\|x - c\|^2}{\sigma^2}}$$

- Αντίστροφη πολυτετραγωνική συνάρτηση Ορίζουμε την συνάρτηση απόστασης από τον τύπο:

$$f(x) = \frac{1}{(\|x - c\|^2 + \sigma^2)^{1/2}}$$

- Συνάρτηση Cauchy

$$f(x) = \frac{(\|x - c\|^2 + \sigma^2)^{-1}}{\sigma}$$

Τέλος, ορίζουμε συνάρτηση για τον υπολογισμό του  $\sigma$ :

- Υπολογίζουμε το  $\sigma$  από τον τύπο:

$$\sigma = \frac{d}{\sqrt{2p}}$$

όπου  $d$  η μέγιστη απόσταση μεταξύ των κέντρων και  $p$  ο αριθμός των κέντρων.

## 2.3 Εκπαίδευση

Αφού υλοποιήσαμε τα παραπάνω, εκπαιδεύουμε το νευρωνικό δίκτυο με τον εξής τρόπο:

1. Αρχικοποιούμε τα κέντρα με έναν από τους παραπάνω τρόπους

2. Υπολογίζουμε το  $\sigma$  για τα συγκεκριμένα δεδομένα εκπαίδευσης και τα συγκεκριμένα κέντρα.
3. Υπολογίζουμε την απόσταση  $D$  των εικόνων εισόδου από το κάθε κέντρο
4. Υπολογίζουμε τον πίνακα επιθυμητών εξόδων με one-hot κωδικοποίηση
5. Υπολογίζουμε τα βάρη λύνοντας το εξής σύστημα:

$$\mathbf{Y} = \mathbf{D}\mathbf{\omega} \Leftrightarrow$$

$$\begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{60.000} \end{bmatrix}_{(60.000 \times 10)} = \begin{bmatrix} D(X_1, C_1) & D(X_1, C_2) & \dots & D(X_1, C_n) \\ D(X_2, C_1) & D(X_2, C_2) & \dots & D(X_2, C_n) \\ \dots & \dots & \dots & \dots \\ D(X_{60000}, C_1) & D(X_{60000}, C_2) & \dots & D(X_{60000}, C_n) \end{bmatrix}_{(60000 \times n)} \begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,10} \\ w_{2,1} & w_{2,2} & \dots & w_{2,10} \\ \dots & \dots & \dots & \dots \\ w_{n,1} & w_{n,2} & \dots & w_{n,10} \end{bmatrix}_{(n \times 10)} \Leftrightarrow$$

$$\begin{bmatrix} w_{1,1} & w_{1,2} & \dots & w_{1,10} \\ w_{2,1} & w_{2,2} & \dots & w_{2,10} \\ \dots & \dots & \dots & \dots \\ w_{n,1} & w_{n,2} & \dots & w_{n,10} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \\ \dots \\ y_{60.000} \end{bmatrix} \begin{bmatrix} D(X_1, C_1) & D(X_1, C_2) & \dots & D(X_1, C_n) \\ D(X_2, C_1) & D(X_2, C_2) & \dots & D(X_2, C_n) \\ \dots & \dots & \dots & \dots \\ D(X_{60000}, C_1) & D(X_{60000}, C_2) & \dots & D(X_{60000}, C_n) \end{bmatrix}^{-1}$$

όπου  $n$  ο αριθμός των κέντρων

6. Υπολογίζουμε τις εκτιμήσεις από τον τύπο

$$\bar{\mathbf{Y}} = \mathbf{D}\mathbf{\omega}$$

## 2.4 Predict

Αφού εκτιμήσουμε τα βάρη εκπαιδεύοντας το μοντέλο, στη συνέχεια προβλέπουμε τα labels των εικόνων κάνοντας τα παρακάτω:

1. Υπολογίζουμε την απόσταση  $D$  των εικόνων εισόδου από το κάθε κέντρο
2. Υπολογίζουμε τις εκτιμήσεις από τον τύπο

$$\bar{\mathbf{Y}} = \mathbf{D}\mathbf{\omega}$$

## Κεφάλαιο 3

# Απόδοση του νευρωνικού δικτύου σε δεδομένα εισόδου χωρίς μείωση διάστασης

### 3.1 Αρχικοποίηση κέντρων με τυχαία επιλογή εικόνων

Θα δοκιμάσουμε την αρχικοποίηση των κέντρων από τυχαία επιλογή εικόνων από το training dataset.

#### 3.1.1 Χρήση Gauss συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 100, 150, 200]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.377	13.1571	0.3803	13.2473	3.14
25	0.5427	7.7466	0.5497	7.5118	7.9257
50	0.6045	6.6037	0.6212	6.1681	16.2120
100	0.7231	4.8557	0.7367	4.5386	33.3884
150	0.7429	4.4246	0.7555	4.1462	49.2155
200	0.7800	3.9517	0.7944	3.6006	65.7530

Παρατήρηση: Παρατηρώ ότι η αύξηση του αριθμού των κέντρων αυξάνει την ακρίβεια του δικτύου, η οποία υφίσταται κορεσμό αφού ο αριθμός των κέντρων ξεπεράσει τα 100. Επιπλέον παρατηρώ αύξηση του χρόνου εκπαίδευσης όσο αυξάνουμε τα κέντρα, το οποίο είναι αναμενόμενο.

### 3.1.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.5621	7.1502	0.5647	6.9902	3.1656
25	0.7679	4.0565	0.7754	4.0294	7.8295
50	0.8120	3.2664	0.8200	3.0969	15.8396
75	0.8556	2.6166	0.8632	2.3606	23.9664
100	0.8611	2.3899	0.8650	2.3647	32.4080

Παρατήρηση: Παρατηρώ ότι η αύξηση του αριθμού των κέντρων αυξάνει την ακρίβεια του δικτύου, η οποία υφίσταται κορεσμό κοντά στα 50 κέντρα με ακρίβεια περίπου 0.8. Επιπλέον παρατηρώ ότι για ίδιο αριθμό κέντρων ο χρόνος εκπαίδευσης είναι παρόμοιος αν χρησιμοποιήσουμε την Gaussian ακτινική συνάρτηση.

### 3.1.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.5693	8.0545	0.5734	7.8713	2.8702
25	0.7245	4.8759	0.7345	4.7640	7.2021
50	0.7759	4.2339	0.7809	4.1021	14.7748
75	0.7960	3.7134	0.7992	3.7262	22.0696
100	0.7910	3.7508	0.8021	3.5074	29.6184

Παρατήρηση: Παρατηρώ ότι η αύξηση του αριθμού των κέντρων αυξάνει την ακρίβεια του δικτύου, η οποία υφίσταται κορεσμό κοντά στα 30 κέντρα με ακρίβεια περίπου 0.75. Σε σύγκριση με τις συναρτήσεις ακτινικού τύπου Gauss και αντίστροφη πολυτετραγωνική πετυχαίνουμε καλύτερη ακρίβεια με λιγότερα κέντρα.

## 3.2 Αρχικοποίηση κέντρων με χρήση μέσης τιμής εικόνων για κάθε ετικέτα

Θα δοκιμάσουμε την αρχικοποίηση των κέντρων με χρήση μέσης τιμής εικόνων για κάθε ετικέτα από το training dataset.



### 3.2.1 Χρήση Gauss συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.7764	3.3739	0.7886	3.3147	5.2749
25	0.7759	3.2830	0.7879	3.2623	12.5314
50	0.7753	3.2853	0.7870	3.2537	24.8196
75	0.5291	5.6386	0.5296	5.5987	37.7454
100	0.2109	14.9028	0.2090	14.7210	50.5365

Παρατήρηση: Παρατηρώ ότι η μέγιστη ακρίβεια επιτυγχάνεται με χρήση 10 κέντρων. Η αύξηση του αριθμού των κέντρων μειώνει την ακρίβεια, ενώ για μεγάλο αριθμό κέντρων η ακρίβεια μειώνεται δραματικά.

### 3.2.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.8132	3.4304	0.8216	3.3116	5.3063
25	0.8127	3.4349	0.8213	3.3248	12.6241
50	0.8125	3.4280	0.8211	3.3346	25.3277
75	0.8124	3.4280	0.8208	3.3504	38.1695
100	0.8124	3.4273	0.8208	3.3504	51.7233

Παρατήρηση: Παρατηρώ και πάλι ότι η μέγιστη ακρίβεια επιτυγχάνεται για 10 κέντρα. Επιπλέον, η μέγιστη ακρίβεια είναι μεγαλύτερη σε σύγκριση με την μέγιστη ακρίβεια που πετυχαίνουμε χρησιμοποιώντας Gaussian συνάρτηση ακτινικού τύπου. Ακόμη, παρατηρούμε ότι η αύξηση του αριθμού των κέντρων δεν έχει τόσο μεγάλη επίδραση στην ακρίβεια και το μέσο τετραγωνικό σφάλμα, καθώς μειώνονται πολύ λίγο.

### 3.2.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.8092	3.5088	0.8169	3.4535	4.9259
25	0.8082	3.5324	0.8154	3.4854	12.3581
50	0.8077	3.5437	0.8149	3.4980	23.9154
75	0.8075	3.5515	0.8147	3.5045	35.9213
100	0.8073	3.5544	0.8144	3.5060	47.2382

Παρατήρηση: Παρατηρώ ότι και πάλι, η μέγιστη ακρίβεια επιτυγχάνεται για 10 κέντρα και η περεταίρω αύξηση των κέντρων δεν έχει μεγάλη επίδραση στην ακρίβεια και το μέσο τετραγωνικό σφάλμα.

### 3.3 Αρχικοποίηση κέντρων με χρήση του αλγορίθμου k-means

Θα δοκιμάσουμε την αρχικοποίηση των κέντρων με χρήση μέσης τιμής εικόνων για κάθε ετικέτα από το training dataset.

#### 3.3.1 Χρήση Gauss συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.6038	5.9469	0.6088	5.8147	34.1989
25	0.7370	4.2536	0.7452	4.1847	57.7669
50	0.8027	3.6636	0.8087	3.6829	94.1579
75	0.8359	2.9682	0.8426	2.9543	131.2287
100	0.8547	2.3896	0.8619	2.3437	150.3260

Παρατήρηση: Παρατηρώ ότι πετυχαίνω αρκετά ικανοποιητική ακρίβεια με σχετικά μικρό αριθμό κέντρων. Ο χρόνος εκτέλεσης είναι μεγαλύτερος σε σύγκριση με άλλες μεθόδους για ίδιο αριθμό κέντρων, λόγω του χρόνου εκτέλεσης του αλγορίθμου k-means.

#### 3.3.2 Χρήση αντίστροφης πολυτετραγωνικής συνάρτησης ακτινικού τύπου

Δοκιμάζουμε  $n = [10, 25, 50, 75, 100]$  αριθμούς κέντρων. Τα αποτελέσματα που παίρνουμε είναι:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.6264	6.1969	0.6345	5.9534	32.3766
25	0.8054	3.6033	0.8096	3.6431	53.3204
50	0.8769	2.2912	0.8829	2.2579	86.6175
75	0.9019	1.7939	0.9091	1.7419	131.9787
100	0.9125	1.5471	0.9175	1.5367	165.1600

Παρατήρηση: Η μείωση χαρακτηριστικών με αυτή την μέθοδο δεν επιτυγχάνει ικανοποιητικά αποτελέσματα στην εκπαίδευση και απόδοση του δικτύου. Και πάλι πετυχαίνουμε ικανοποιητική ακρίβεια με σχετικά μικρό αριθμό κέντρων. Σε σύγκριση με την χρήση Gaussian συνάρτησης ακτινικού τύπου, η απόδοση για ίδιο αριθμό κέντρων είναι καλύτερη.

### 3.3.3 Χρήση Cauchy συνάρτησης ακτινικού τύπου

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.6211	6.2975	0.6274	6.0762	36.5752
25	0.7992	3.5917	0.8070	3.6355	55.5698
50	0.8536	2.7576	0.8597	2.8095	93.5927
75	0.8915	1.9855	0.8969	2.0183	122.1944
100	0.9123	1.5544	0.9150	1.6052	158.8771

Παρατήρηση: Έχουμε παρόμοια απόδοση με την χρήση αντίστροφης

πολυτετραγωνικής συνάρτησης, τόσο στο training όσο και στο testing dataset.

## Κεφάλαιο 4

# Απόδοση του νευρωνικού δικτύου σε δεδομένα εισόδου με μείωση διάστασης

### 4.1 Μείωση χαρακτηριστικών με χρήση μέσης φωτεινότητας σειράς

Τροποποιούμε τα δεδομένα μας έτσι, ώστε τα δεδομένα εισόδου να αποτελούν την μέση φωτεινότητα σειράς κάθε εικόνας. Έτσι, τα δεδομένα εισόδου έχουν διάσταση  $28 \times M$ . Επιπλέον, θα χρησιμοποιήσω 10 κέντρα και τα αρχικοποιώ με χρήση μέσης τιμής ει-κόνων για κάθε ετικέτα και χρησιμοποιώ την αντίστροφη πολυτετραγωνική συνάρτηση για τον υπολογισμό των αποστάσεων. Οι παραπάνω επιλογές έγιναν, καθώς είχαμε ικανοποιητικά αποτελέσματα στο πλήρες σύνολο δεδομένων.

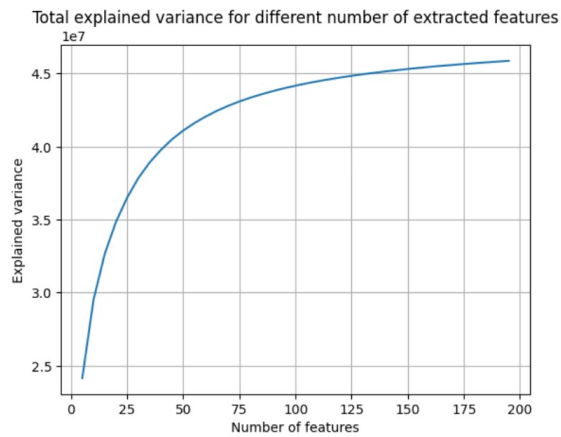
Εκπαιδεύω το δίκτυο με τον παραπάνω τρόπο και παίρνω τα εξής αποτελέσματα :

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.1150	18.2482	0.1085	18.1248	4.1858

Παρατήρηση : Η μείωση χαρακτηριστικών με αυτή την μέθοδο δεν επιτυγχάνει ικανοποιητικά αποτελέσματα στην εκπαίδευση και απόδοση του δικτύου.

### 4.2 Μείωση χαρακτηριστικών με τον αλγόριθμο PCA

Υλοποιώ τον αλγόριθμο PCA και στη συνέχεια τρέχω τον αλγόριθμο για διάφορα  $k$ , όπου  $k$  είναι ο αριθμός των χαρακτηριστικών μετά την μείωση διάστασης. Παρακάτω φαίνεται η συνολική διακύμανση των δεδομένων συναρτήσει του  $k$ :



Επιλέγω  $\kappa = 125$ .

Τα δεδομένα εισόδου έχουν πλέον διάσταση  $125 \times M$ . Και πάλι θα χρησιμοποιήσω μέση τιμή εικόνων για την αρχικοποίηση των κέντρων, αντίστροφη πολυτετραγωνική ακτινική συνάρτηση και 10 κέντρα. Εκπαιδεύω το δίκτυο και παίρνω τα εξής αποτελέσματα:

n	Train Accuracy	Train MSE	Test Accuracy	Test MSE	Total training time (s)
10	0.8127	3.4393	0.1005	16.9560	7.3255

Παρατήρηση: Παρατηρώ ότι το δίκτυο υπερπροσαρμόζεται στα δεδομένα εκπαίδευσης, καθώς πετυχαίνει μεγάλη ακρίβεια στα δεδομένα εκπαίδευσης αλλά πολύ μικρή ακρίβεια στα δεδομένα ελέγχου.