

3^η εργασία του μαθήματος «Νευρωνικά Δίκτυα»

Χρυσολόγου Γεώργιος (ΑΕΜ: 10782)

Για την εκπόνηση της παρούσας εργασίας, επέλεξα να υλοποιήσω ένα Radial Basis Function Neural Network το οποίο επιλύει το πρόβλημα της κατηγοριοποίησης πολλαπλών κλάσεων. Το πρόγραμμα δημιουργήθηκε σε python, χωρίς την χρήση των σχετικών frameworks (pytorch, keras κλπ), και η εκπαίδευση πραγματοποιήθηκε στο data set Cifar-10.

Περιγραφή αλγορίθμου: Ένα απλό RBF νευρωνικό δίκτυο απαρτίζεται από ένα hidden layer και ένα output layer. Το πρώτο αποτελείται από έναν αριθμό νευρώνων, καθένας από τους οποίους έχει μία συγκεκριμένη θέση στον χώρο που ονομάζεται κέντρο c . Για ένα συγκεκριμένο δείγμα, η έξοδος ενός νευρώνα του hidden layer υπολογίζεται μέσω μίας συνάρτησης ακτινικής βάσης. Για την εκπόνηση της παρούσας εργασίας, επιλέχθηκε η Gaussian συνάρτηση $\varphi(x) = e^{\frac{-||x-c||}{2\sigma^2}}$. Η έξοδος αυτή δείχνει την συσχέτιση του δείγματος με το κέντρο στο οποίο αντιστοιχεί ο νευρώνας αυτός. Συγκεκριμένα, όσο μεγαλύτερη η τιμή της εξόδου, τόσο πιο κοντά στον χώρο βρίσκεται το δείγμα με το κέντρο. Όσον αφορά το output layer, αποτελείται από 10 νευρώνες. Για ένα συγκεκριμένο δείγμα, η έξοδος του output layer πρόκειται για ένα διάνυσμα 10 στοιχείων, καθένα από τα οποία δείχνει την πιθανότητα το δείγμα να ανήκει στην κλάση στην οποία αντιστοιχεί αυτό το στοιχείο.

Τα δύο στρώματα εκπαιδεύονται ξεχωριστά. Κατά την εκπαίδευση του hidden layer, καθορίζονται οι θέσεις των κέντρων καθώς και η τιμή (ή οι τιμές) της παραμέτρου σ , η οποία ονομάζεται παράμετρος εύρους. Η τελευταία ορίζει το εύρος της καμπύλης (δηλαδή πόσο στενή ή πλατιά θα είναι) της συνάρτησης ακτινικής βάσης γύρω από κάθε κέντρο. Όσον αφορά την επιλογή των κέντρων, αυτή μπορεί να πραγματοποιηθεί με διαφορετικούς τρόπους, όπως τυχαία ή με χρήση του αλγορίθμου κ-μέσων, μεταξύ άλλων. Αντίστοιχα, η τιμή της παραμέτρου σ μπορεί να οριστεί ως μία επιλεγμένη τιμή (μέσω δοκιμών), αλλά συχνά ορίζεται μέσω της σχέσης $\sigma = \frac{d}{\sqrt{2 * n}}$, όπου n το πλήθος των κέντρων. Αν η τιμή σ είναι ίδια για όλα τα κέντρα, ως d ορίζεται η μέση απόσταση μεταξύ όλων των αποστάσεων όλων των κέντρων. Μόλις οριστούν οι θέσεις των κέντρων και η τιμή της παραμέτρου σ , η εκπαίδευση του hidden layer ολοκληρώνεται.

Όσον αφορά την εκπαίδευση του outer layer, μπορεί να πραγματοποιηθεί με δύο τρόπους. Ο πρώτος είναι η χρήση του κανόνα delta rule. Σύμφωνα με αυτόν, τα βάρη των συνάψεων που συνδέουν το output layer με το hidden layer ανανεώνονται επαναληπτικά μέσω της σχέσης $w_{k+1} = w_k + \beta * (d - y_k) * X$ όπου w_{k+1} , w_k τα βάρη των εποχών $k + 1$ και k αντίστοιχα, β το βήμα εκπαίδευσης (learning rate), d ο one-hot encoded στόχος του δικτύου, y_k η έξοδος του outer layer για την εποχή k και X η έξοδος του hidden layer. Συγκεκριμένα, η έξοδος του hidden layer X πρόκειται για έναν πίνακα διαστάσεων (αριθμός δειγμάτων)X(αριθμός νευρώνων), ο στόχος d είναι ένας πίνακας διαστάσεων (αριθμός δειγμάτων)X10, όπου κάθε γραμμή του αποτελείται από εννιά 0 και ένα 1 το οποίο βρίσκεται στην στήλη που αντιστοιχεί στην κλάση που ανήκει το δείγμα, και η έξοδος του outer layer y_k ορίζεται ως ένας πίνακας ίδιων διαστάσεων με τον στόχο d , κάθε γραμμή του οποία αποτελείται από δέκα τιμές οι οποίες υποδεικνύουν την πιθανότητα το δείγμα να ανήκει σε καθεμία από τις 10 κλάσεις, όπως περιγράφηκε και παραπάνω. Με ανάλογο τρόπο ανανεώνονται και τα biases b των νευρώνων του outer layer, μέσω της σχέσης $b_{k+1} = b_k - \beta * (y_k - d)$.

Ο δεύτερος τρόπος εκπαίδευσης είναι η χρήση του αλγορίθμου Least Squares. Ο αλγόριθμος αυτός υπολογίζει τις τιμές των συναπτικών βαρών μεταξύ του hidden layer και του output layer, καθώς και των biases του τελευταίου, με χρήση γραμμικής άλγεβρας. Στόχο της εκπαίδευσης αποτελεί η ελαχιστοποίηση των σφαλμάτων του δικτύου. Ο αλγόριθμος ελαχιστοποιεί την συνάρτηση σφάλματος $L = ||d - y||$, όπου

d ο στόχος του δικτύου και y η έξοδος του output layer (όπως παραπάνω). Η συνάρτηση αυτή μπορεί να γραφεί ως εξής: $L = ||d - X' * w'||$, όπου X' ο επαυξημένος πίνακας εξόδων του hidden layer (με στήλη 1 για τα biases) και w' ο επαυξημένος πίνακας των συναπτικών βαρών και των biases. Για την εύρεση του ολικού ελαχίστου της συνάρτησης $L(w')$, μηδενίζεται η παράγωγος $\frac{dL}{dw'}$. Αναλυτικά: $\frac{dL}{dw'} = -2X'^T(d - X' * w) = 0 \Rightarrow w' = (X'^T * X')^{-1} * X'^T * d$.

Με τον τρόπο αυτό, υπολογίζονται οι τιμές των συναπτικών βαρών και των biases που ελαχιστοποιούν τα σφάλματα προβλέψεων του δικτύου.

Στόχο της εκπαίδευσης των δύο στρωμάτων αποτελεί η εύρεση των καταλληλότερων παραμέτρων για την επίλυση του προβλήματος κατηγοριοποίησης.

Περιγραφή κώδικα: Σύμφωνα με τον κώδικα, αρχικά, αποθηκεύονται τα 50000 δεδομένα εκπαίδευσης σε έναν πίνακα 50000 x 3072 με όνομα `data_tr`, όπου κάθε γραμμή αποτελεί ένα δείγμα εκπαίδευσης. Αντίστοιχα, αποθηκεύονται τα 10000 δείγματα ελέγχου σε έναν πίνακα 10000 x 3072 με όνομα `data_test`, όπου κάθε γραμμή αποτελεί ένα δείγμα ελέγχου.

Τα δεδομένα αυτά κανονικοποιούνται μέσω διαίρεσης όλων των στοιχείων των δύο παραπάνω πινάκων δια 255. Επιπλέον, εφαρμόζεται η μέθοδος PCA, όπου εξασφαλίζεται η συγκράτηση του 90% της πληροφορίας, για την μείωση των διαστάσεων των πινάκων, με σκοπό την απλοποίηση του προβλήματος, την επιτάχυνση των υπολογισμών και τον περιορισμό του συνολικού χρόνου εκπαίδευσης.

Κατόπιν, δημιουργούνται οι στόχοι d τόσο για τα training δείγματα όσο και για τα test δείγματα. Συγκεκριμένα, η μεταβλητή `d_inputs` είναι ένας πίνακας διαστάσεων 50000X10 (50000 είναι συνολικά τα training δείγματα) κάθε γραμμή της οποίας αποτελείται από μηδενικά στοιχεία, εκτός από το στοιχείο της στήλης η οποία αντιστοιχεί στην κλάση στην οποία ανήκει το δείγμα (όπως αναφέρθηκε και παραπάνω). Με ακριβώς ανάλογο τρόπο ορίζεται και η μεταβλητή `d_test`, η οποία είναι πίνακας διαστάσεων 10000X10 (10000 είναι συνολικά τα test δείγματα).

Ακολουθεί ο ορισμός της αρχιτεκτονικής του νευρωνικού δικτύου. Ο κώδικας σχεδιάστηκε ώστε να επιτρέπει δημιουργία πολυπλοκότερων δικτύων από το απλό RBF δίκτυο με ένα hidden layer και ένα outer layer. Υπάρχει η δυνατότητα προσθήκης περισσότερων hidden layers, τα οποία τοποθετούνται μετά το layer των RBF νευρώνων και πριν το outer layer, με σκοπό την κατασκευή βαθύτερων δικτύων που μπορούν να πετύχουν καλύτερη απόδοση.

Αρχικά ορίζεται το layer των RBF νευρώνων, το οποίο λειτουργεί ως input layer. Η κλάση `RbfLayer` αποτελείται μία συνάρτηση αρχικοποίησης, την συνάρτηση `train` και την συνάρτηση `forward`. Με την χρήση της συνάρτησης `train`, ορίζονται τυχαία δείγματα, ο αριθμός των οποίων καθορίζεται κατά την δημιουργία ενός αντικειμένου της κλάσης, ως τα κέντρα που αντιστοιχούν στους νευρώνες του layer. Στην περίπτωση της εκπαίδευσης με τυχαία επιλογή κέντρων, ακολουθεί ο υπολογισμός της παραμέτρου σ , η οποία γίνεται με κάποιον από τους δύο τρόπους που περιγράφηκαν παραπάνω. Αναλυτικά:

1) Υπολογισμός μέσω της μέσης απόστασης όλων των κέντρων μεταξύ τους

```
# Υπολογισμός του σ μέσω του μέσου όρου των αποστάσεων μεταξύ των κέντρων
pairwise_distances = np.linalg.norm(self.c[:, None, :] - self.c[None, :, :], axis=2)
mean_distance = np.mean(pairwise_distances[pairwise_distances > 0]) # Αποφεύγουμε τις μηδενικές αποστάσεις
self.sigma = mean_distance / np.sqrt(2 * self.n_centroids) * scaling_factor
```

Όπου `self.c` είναι ο πίνακας που περιέχει τα κέντρα του layer.

2) Απλή ανάθεση τιμής

Στην περίπτωση 1), προστέθηκε και μία μεταβλητή `scaling_factor`, οι καταλληλότερες τιμές της οποίας καθορίστηκαν μέσω δοκιμών. Στα πειράματα παρακάτω αναφέρεται τόσο ο τρόπος υπολογισμού της παραμέτρου σ όσο και η τιμή της `scaling_factor`.

Στην περίπτωση εκπαίδευσης με χρήση του αλγορίθμου k -μέσων, μετά την αρχικό τυχαίο ορισμό δειγμάτων ως κέντρα του layer, ακολουθεί μία `while` στην οποία πραγματοποιείται επαναληπτικά ο αλγόριθμος. Κατόπιν, υπολογίζεται η τιμή της παραμέτρου σ μέσω μίας από τις παραπάνω τρεις μεθόδους και η εκπαίδευση του RBF layer ολοκληρώνεται.

Στην συνέχεια, ορίζεται το `outer layer` των δέκα νευρώνων. Η κλάση `OuterLayer` περιλαμβάνει μία συνάρτηση αρχικοποίησης, την συνάρτηση `forward`, την συνάρτηση `activation`, την συνάρτηση `backward` και την συνάρτηση `train_least_squares`.

Αν επιλεγθεί η εκπαίδευση με χρήση του επαναληπτικού αλγορίθμου `backpropagation`, χρησιμοποιούνται οι πρώτες τέσσερις συνάρτησης. Συγκεκριμένα, η συνάρτηση `forward` πολλαπλασιάζει τις εισόδους στο στρώμα με τα αντίστοιχα βάρη που συνδέουν αυτό με το προηγούμενο στρώμα και προσθέτει το `bias`, σχηματίζοντας την αρχική έξοδο του στρώματος. Κατόπιν, η έξοδος αυτή αποτελεί είσοδο για την συνάρτηση `activation`, η οποία ενεργοποιεί την αρχική έξοδο του στρώματος με την συνάρτηση `softmax`, παράγοντας την τελική “ενεργοποιημένη” έξοδο του δικτύου. Η συνάρτηση `backward` είναι αυτή που υλοποιεί τον αλγόριθμο `backpropagation`. Κατά την κλήση της, υπολογίζονται νέες τιμές των βαρών και των `biases` με βάση το σφάλμα που ορίζεται ως $\delta = y - d$, όπου y η “ενεργοποιημένη” έξοδος του `outer layer` και d ο `one-hot encoded` στόχος. Οι ανανεώσεις των βαρών και των `biases` γίνονται με στόχο την ελαχιστοποίηση του σφάλματος αυτού. Αναλυτικά:

```
def backward(self, b, inputs, d): 1usage
    self.delta = self.activated_outputs - d
    weights_update = b * (self.delta.T @ inputs) / len(inputs)
    biases_update = b * np.mean(self.delta, axis=0)
    self.weights = self.weights - weights_update
    self.biases = self.biases - biases_update
```

Όπου b το βήμα εκπαίδευσης, `inputs` η είσοδος στο στρώμα (δηλαδή η έξοδος του προηγούμενου στρώματος).

Στην δεύτερη περίπτωση στην οποία επιλέγεται ο αλγόριθμος `Least Squares` για την εκπαίδευση του `outer layer`, η συνάρτηση `backward` αντικαθίσταται από την συνάρτηση `train_least_squares`. Κατά την κλήση της τελευταίας, υλοποιείται ο αλγόριθμος ελαχίστων τετραγώνων, όπως περιγράφηκε παραπάνω, και υπολογίζονται οι τιμές των βαρών και των `biases`.

```
def train_least_squares(self, inputs, d):
    # Προσθήκη bias στον πίνακα εισόδων
    inputs_with_bias = np.hstack([inputs, np.ones((inputs.shape[0], 1))])

    # Υπολογισμός ψευδοαντίστροφου του πίνακα εισόδων
    pseudo_inverse = np.linalg.pinv(inputs_with_bias)

    # Επίλυση Least Squares για βάρη και bias
    weights_bias = pseudo_inverse @ d # (M+1 x K)

    # Ενημέρωση των βαρών και bias
    self.weights = weights_bias[:-1, :].T # Τα βάρη (χωρίς την τελευταία στήλη)
    self.biases = weights_bias[-1, :] # Τα bias (τελευταία γραμμή)
```

Σε περίπτωση που επιλεχθεί να κατασκευαστεί ένα πολυπλοκότερο δίκτυο, με παραπάνω από ένα hidden layers, πριν την δημιουργία του outer layer πραγματοποιείται ορισμός των υπολοίπων hidden layers εκτός του RBF layer. Η κλάση Layer περιλαμβάνει ακριβώς τις ίδιες συναρτήσεις με την κλάση OuterLayer. Ωστόσο, οι συναρτήσεις activation και backward παρουσιάζουν κάποιες διαφοροποιήσεις. Στην πρώτη, η συνάρτηση ενεργοποίησης είναι η ReLu, και όχι η softmax που χρησιμοποιείται στο OuterLayer. Επιπλέον, στην συνάρτηση backward, το σφάλμα δέλτα ορίζεται ως $\delta = w^T * \delta' * \sigma'(z)$, όπου w ο πίνακας των βαρών που συνδέουν το εκάστοτε hidden layer με το επόμενο layer, δ' το σφάλμα δέλτα του επόμενου layer και $\sigma'(z)$ η παράγωγος της συνάρτησης ενεργοποίησης (ReLu) με όρισμα την έξοδο του hidden layer $z = inputs * W^T + b$. Κατόπιν, οι εξισώσεις ανανεώσεις των βαρών και των biases είναι ίδιες με τις αντίστοιχες του outer layer.

Αφού οριστεί η αρχιτεκτονική του δικτύου, εκπαιδεύεται το RBF layer. Κατόπιν, πραγματοποιείται η εκπαίδευση του outer layer. Αν δεν υπάρχουν άλλα hidden layers, το outer layer μπορεί να εκπαιδευτεί είτε με τον αλγόριθμο Least Squares ή με τον αλγόριθμο backpropagation. Στην δεύτερη περίπτωση, ορίζεται, πρώτα, η τιμή του βήματος β και ο αριθμός των εποχών. Εάν πρόκειται για ένα πολύπλοκο RBF δίκτυο με περισσότερα hidden layers, η εκπαίδευση των layers, εκτός του RBF layer, πραγματοποιείται αποκλειστικά με τον αλγόριθμο backpropagation, δεδομένου ότι η μέθοδος Least Squares δεν μπορεί να εφαρμοστεί εύκολα σε τέτοιου είδους δίκτυα.

Στην περίπτωση εκπαίδευσης με backpropagation, σε κάθε εποχή υπολογίζεται το training accuracy και το test accuracy, συγκρίνοντας τις προβλέψεις του δικτύου με τις πραγματικές κλάσεις των δειγμάτων. Ως πρόβλεψη του δικτύου για ένα δείγμα, ορίζεται η κλάση στην οποία αντιστοιχεί η μεγαλύτερη από τις 10 τιμές που παράγει ως έξοδο το δίκτυο και αποτελούν εκφράσεις των πιθανοτήτων το δείγμα να ανήκει σε καθεμία από τις 10 κλάσεις. Αντίστοιχα, όταν το δίκτυο αποτελείται από μόνο ένα hidden layer, ένα outer layer και εκπαιδεύεται με τον αλγόριθμο Least Squares, υπολογίζεται μόνο μία φορά το training και το test accuracy, καθώς σε αυτή την περίπτωση δεν υπάρχει επαναληπτικότητα.

Στο τέλος εκτυπώνεται ο συνολικός χρόνος εκτέλεσης του κώδικα.

Παραδείγματα ορθής και εσφαλμένης κατηγοριοποίησης

Για δίκτυο με ένα RBF layer και ένα outer layer. Το τελευταίο εκπαιδεύεται με τον αλγόριθμο backpropagation:

Ορθή κατηγοριοποίηση

1)

```
Εξοδος του outer layer για το δείγμα:
[0.22481252 0.0904622 0.0869797 0.05663854 0.03535628 0.0443232
 0.02964895 0.0731184 0.23088665 0.12777355]
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 8
Πραγματική κλάση του δείγματος: Κλάση 8
```

2)

```
Εξοδος του outer layer για το δείγμα:
[0.09135621 0.05551972 0.12642109 0.14172034 0.1224704 0.20997905
 0.09540553 0.06776952 0.07347185 0.01588629]
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 5
Πραγματική κλάση του δείγματος: Κλάση 5
```

Παρατηρείται ότι στο 1^ο παράδειγμα, οι πιθανότητες που αντιστοιχούν στις κλάσεις 0 και 8 είναι αρκετά μεγαλύτερες από τις υπόλοιπες ενώ στο 2^ο παράδειγμα αυτό συμβαίνει μόνο για την πιθανότητα που αντιστοιχεί στην κλάση 5. Ως συμπέρασμα, στο 1^ο παράδειγμα το δίκτυο “δυσκολεύεται να αποφασίσει” αν το δείγμα ανήκει στην κλάση 0 ή στην κλάση 8, ενώ στο 2^ο η πρόβλεψη πραγματοποιείται με μεγαλύτερη “βεβαιότητα”. Οι προβλέψεις και στις δύο περιπτώσεις είναι ορθές.

Εσφαλμένη κατηγοριοποίηση

1)

```
Εξοδος του outer layer για το δείγμα:  
[0.15423466 0.10809781 0.10612369 0.08218362 0.07659957 0.07290299  
 0.06514264 0.09085862 0.16431298 0.07954344]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 8  
Πραγματική κλάση του δείγματος: Κλάση 4
```

2)

```
Εξοδος του outer layer για το δείγμα:  
[0.04490612 0.08983825 0.10807334 0.12809196 0.14402699 0.13576895  
 0.14764564 0.10616972 0.05038608 0.04509295]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 6  
Πραγματική κλάση του δείγματος: Κλάση 3
```

Στο 1^ο παράδειγμα παρατηρείται ότι η πιθανότητα που αντιστοιχεί στην σωστή κλάση 4 είναι πολύ μικρή σε σχέση με τις πιθανότητες που αντιστοιχούν σε άλλες κλάσεις. Αυτό υποδεικνύει ότι το δίκτυο απέτυχε σε μεγάλο βαθμό να κατηγοριοποιήσει το δείγμα του παραδείγματος. Αντίθετα, στο 2^ο παράδειγμα, οι πιθανότητες που αντιστοιχούν σε διαφορετικές κλάσεις, μεταξύ των οποίων και της σωστής κλάσης 3, ήταν αρκετά υψηλότερες από τις υπόλοιπες και πολύ κοντά μεταξύ τους. Συμπερασματικά, παρά την λάθος πρόβλεψη, το δίκτυο πλησίασε στην σωστή κατηγοριοποίηση. Ενδεχομένως, αν η εκπαίδευση συνεχιζόταν για περισσότερες εποχές, το δίκτυο θα επιτύγχανε στην εύρεση της σωστής κλάσης.

Σε όλα τα παραδείγματα παρατηρείται ότι για την συγκεκριμένη αρχιτεκτονική και των συγκεκριμένο τρόπο εκπαίδευσης, δεν υπάρχουν τιμές πιθανοτήτων που να προσεγγίζουν την μονάδα, γεγονός που θα σήμαινε ότι το δίκτυο πραγματοποιεί προβλέψεις με μεγάλη “βεβαιότητα”.

Για δίκτυο με ένα RBF hidden layer, δύο ακόμα hidden layers 512 και 256 νευρώνων αντίστοιχα και ένα outer layer. Τα τρία τελευταία layers εκπαιδεύονται με τον αλγόριθμο backpropagation:

Ορθή κατηγοριοποίηση:

1)

```
Εξοδος του outer layer για το δείγμα:  
[0.30230466 0.05573385 0.05179503 0.03126366 0.01231278 0.02281993  
 0.00642092 0.01799378 0.43884358 0.06051181]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 8  
Πραγματική κλάση του δείγματος: Κλάση 8
```

2)

```
Εξοδος του outer layer για το δείγμα:  
[0.10618163 0.16111021 0.03070109 0.02035136 0.01421783 0.01138546  
0.01136156 0.04018639 0.2569788 0.34752568]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 9  
Πραγματική κλάση του δείγματος: Κλάση 9
```

Στα δύο αυτά παραδείγματα, παρατηρείται ότι η διακύμανση μεταξύ των τιμών των πιθανοτήτων είναι μεγαλύτερη σε σχέση με την αντίστοιχη των παραδειγμάτων με την προηγούμενη αρχιτεκτονική. Συγκεκριμένα, οι περισσότερες πιθανότητες πλησιάζουν την μηδενική τιμή, ενώ ορισμένες είναι πολύ υψηλότερες. Συμπερασματικά, η πολυπλοκότερη αρχιτεκτονική επιτρέπει σε αυτό το δίκτυο να "μάθει να ξεχωρίζει καλύτερα" τις κλάσεις ανάλογα με τα χαρακτηριστικά τους, συγκριτικά με το δίκτυο των προηγούμενων παραδειγμάτων.

Εσφαλμένη κατηγοριοποίηση

1)

```
Εξοδος του outer layer για το δείγμα:  
[0.58857641 0.01918532 0.05194666 0.02151172 0.01292152 0.0197187  
0.00381316 0.01372145 0.25133918 0.01726589]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 0  
Πραγματική κλάση του δείγματος: Κλάση 2
```

2)

```
Εξοδος του outer layer για το δείγμα:  
[0.17055163 0.09335647 0.14339011 0.09141264 0.0622013 0.0643762  
0.04278094 0.08541246 0.14387799 0.10264026]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 0  
Πραγματική κλάση του δείγματος: Κλάση 8
```

Όπως παρατηρήθηκε και στα παραδείγματα της προηγούμενης αρχιτεκτονικής, υπάρχουν περιπτώσεις στις οποίες η πρόβλεψη του δικτύου παρουσιάζει μεγάλο σφάλμα, ενώ σε άλλες η πρόβλεψη του δικτύου προσεγγίζει την σωστή κλάση. Η παρατήρηση που αναλύθηκε παραπάνω αναφορικά με την διακύμανση των τιμών των πιθανοτήτων είναι εμφανής και στα παραδείγματα αυτά (κυρίως στο 1^ο παράδειγμα).

Για δίκτυο με ένα RBF layer και ένα outer layer. Το τελευταίο εκπαιδεύεται με τον αλγόριθμο Least Squares:

Ορθή κατηγοριοποίηση:

1)

```
Εξοδος του outer layer για το δείγμα:  
[0.12016672 0.0908503 0.10584541 0.09673169 0.0988191 0.09707441  
0.09442133 0.09583025 0.11401773 0.08624306]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 0  
Πραγματική κλάση του δείγματος: Κλάση 0
```


2)

```
Εξοδος του outer layer για το δείγμα:  
[0.08975489 0.08899705 0.10576986 0.10145405 0.11689025 0.09919796  
0.11981719 0.09885692 0.08937113 0.08989071]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 6  
Πραγματική κλάση του δείγματος: Κλάση 6
```

Εσφαλμένη κατηγοριοποίηση:

1)

```
Εξοδος του outer layer για το δείγμα:  
[0.08498181 0.08342049 0.10406186 0.10902805 0.10250618 0.1095507  
0.10539044 0.09548775 0.1017449 0.10382782]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 5  
Πραγματική κλάση του δείγματος: Κλάση 6
```

2)

```
Εξοδος του outer layer για το δείγμα:  
[0.10906732 0.10036162 0.10282848 0.11826998 0.09742534 0.09438689  
0.09389481 0.08639542 0.10200656 0.09536359]  
Πρόβλεψη του δικτύου για το δείγμα: Κλάση 3  
Πραγματική κλάση του δείγματος: Κλάση 2
```

Παρατηρείται ότι αυτός ο τρόπος εκπαίδευσης του outer layer οδηγεί στο ίδιο φαινόμενο που παρατηρήθηκε και στο πρώτο σύνολο παραδειγμάτων (αρχιτεκτονική μόνο δύο layers και backpropagation). Στην περίπτωση αυτή, μάλιστα, η διακύμανση των τιμών των πιθανοτήτων είναι ακόμα μικρότερη. Είναι εμφανές ότι όλες οι τιμές βρίσκονται πολύ κοντά στην τιμή 0.1. Ωστόσο, το δίκτυο επιτυγχάνει σωστές προβλέψεις ακόμα και χωρίς να “αποφασίζει με μεγάλη βεβαιότητα”, με μεγάλη συχνότητα, γεγονός που θα αναλυθεί περαιτέρω παρακάτω.

Πειράματα:

Πραγματοποιήθηκαν πειράματα με διαφορετική αρχιτεκτονική, διαφορετικές τιμές παραμέτρων και διαφορετικό τρόπο εκπαίδευσης των στρωμάτων. Οι αρχικές δοκιμές σημειώθηκαν για μικρότερα τμήματα του συνολικού data set. Αυτά που θεωρήθηκαν αξιοσημείωτα πραγματοποιήθηκαν, στην συνέχεια, σε όλο το data set, τα αποτελέσματα των οποίων αναφέρονται παρακάτω.

Εκπαίδευση των layers με delta rule / backpropagation (εκτός του RBF layer)

Στα πειράματα που περιγράφονται παρακάτω η εκπαίδευση του outer layer, καθώς και των hidden layers (όπου υπάρχουν) εκτός του RBF layer, πραγματοποιήθηκαν με χρήση του αλγορίθμου backpropagation. Σε ορισμένα πειράματα τα κέντρα του RBF layer ορίστηκαν ως τυχαία δείγματα εκπαίδευσης, ενώ σε άλλα το RBF layer εκπαιδεύτηκε σύμφωνα με τον αλγόριθμο K-μέσων.

Η τιμή της παραμέτρου σ υπολογίστηκε σύμφωνα με τον τύπο $\sigma = \frac{d}{\sqrt{2 \cdot n}} * a$, όπου d ο μέσος όρος των αποστάσεων όλων των κέντρων μεταξύ τους, n ο αριθμός των κέντρων και a ένας επιλεγμένος σταθερός όρος. Για τον ορισμό του όρου a , πραγματοποιήθηκαν πειράματα με διαφορετικές τιμές του σε υποσύνολα του data set. Κατόπιν, για τα πειράματα στο σύνολο του data set επιλέχθηκαν οι τιμές του a οι οποίες παρουσίασαν το μεγαλύτερο ενδιαφέρον.

Τα δίκτυα των πειραμάτων που ακολουθούν φέρουν στο RBF layer τους 500, 1000 ή 2000 νευρώνες (κέντρα). Η επιλογή αυτή πραγματοποιήθηκε καθώς δίκτυα με άλλους αριθμούς κέντρων δεν επιτύγχαναν καθόλου ικανοποιητική απόδοση ή παρουσίαζαν παρόμοια συμπεριφορά με κάποιο δίκτυο με ένα από τους παραπάνω τρεις αριθμούς κέντρων. Για παράδειγμα, τα δίκτυα με 100 και 200 κέντρα δεν αποδείχθηκαν κατάλληλα όσον αφορά την κατηγοριοποίηση του Cifar 10, το οποίο πρόκειται για ένα πολύπλοκο data set. Ακόμη, τα δίκτυα με 600 και 700 κέντρα δεν παρουσίαζαν σημαντική διαφοροποίηση στην επίδοσή τους σε σχέση με το δίκτυο των 500 κέντρων. Πειράματα με περισσότερα από 2000 κέντρα δεν πραγματοποιήθηκαν καθώς αποδείχθηκε ιδιαίτερα υπολογιστικά απαιτητικό και χρονοβόρο.

Στην στήλη της αρχιτεκτονικής:

A -> Το δίκτυο αποτελείται από το RBF layer και το outer layer.

B -> Το δίκτυο αποτελείται από το RBF layer, ένα hidden layer 512 νευρώνων, ένα hidden layer 256 νευρώνων και το outer layer.

1)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
500	22.97 ($\alpha = 40$)	3000	0.01	A	Τυχαία επιλογή κέντρων	27.15%	27.85%	0.0026	68.97

2)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	15.78 ($\alpha = 40$)	3000	0.01	A	Τυχαία επιλογή κέντρων	29.07%	30.00%	0.0038	57.20

Μεταξύ των πειραμάτων 1) και 2), υπάρχει διαφοροποίηση μεταξύ του αριθμού των κέντρων των δύο δικτύων. Επιπλέον, η τιμή της παραμέτρου σ είναι διαφορετική, υπολογίζεται, όμως, με τον ίδιο τρόπο. Παρατηρούμε ότι η αύξηση του πλήθους των κέντρων επιδρά θετικά στο ποσοστό επιτυχίας του δικτύου τόσο στο training όσο και στο test. Δεδομένου ότι το δίκτυο με 1000 κέντρα καθίσταται πολυπλοκότερο, η αύξηση του χρόνου εκτέλεσης θα αποτελούσε μία λογική εκτίμηση. Ωστόσο, αυτό δεν παρατηρείται καθώς ο συνολικός χρόνος εκτέλεσης είναι μεγαλύτερος για το πείραμα 1) με 500 κέντρα. Αυτό, ενδεχομένως, οφείλεται σε παράγοντες που σχετίζονται με την λειτουργία του υπολογιστή, και όχι με την αρχιτεκτονική του δικτύου.

3)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	24.08 ($\alpha = 60$)	3000	0.02	A	Τυχαία επιλογή κέντρων	28.34%	28.97%	0.0042	88.22

Στο πείραμα 3), σε σύγκριση με το πείραμα 2), έχει αυξηθεί η τιμή του όρου α , γεγονός που οδηγεί στην ανάλογη αύξηση της παραμέτρου σ . Από πειράματα σε υποσύνολα του data set, διαπιστώθηκε ότι ο διπλασιασμός του βήματος συχνά είχε ως αποτέλεσμα την επίτευξη καλύτερης απόδοσης. Για τον λόγο αυτό, πραγματοποιήθηκε, ακόμη, η αλλαγή αυτή. Παρ' όλα αυτά, ο συνδυασμός των δύο τροποποιήσεων επίδρασε αρνητικά στα ποσοστά training και test accuracy.

4)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
2000	11.30 ($\alpha = 40$)	3000	0.01	A	Τυχαία επιλογή κέντρων	32.06%	32.81%	1.56	56.22

Στο πείραμα 4), έχουν επιλεγεί οι ίδιες τιμές για τις παραμέτρους (α , αριθμός εποχών, βήμα, αρχιτεκτονική, τρόπος εκπαίδευσης RBF layer) με τις αντίστοιχες του πειράματος 2), με μόνη διαφορά την αύξηση του πλήθους κέντρων. Παρατηρείται σημαντική αύξηση στο ποσοστό επιτυχίας προβλέψεων τόσο στο training όσο και στο test. Ο χρόνος εκπαίδευσης του RBF layer αυξάνεται, επίσης, αισθητά.

5)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	15.92 ($\alpha = 40$)	3000	0.02	B	Τυχαία επιλογή κέντρων	34.154%	34.81%	0.015	174.86

Για το πείραμα 5), επιλέχθηκαν οι τιμές παραμέτρων που κρίθηκαν καταλληλότερες για δίκτυο με 1000 κέντρα, με βάση τα προηγούμενα πειράματα. Επιπλέον, το δίκτυο υλοποιήθηκε με την αρχιτεκτονική B. Η αλλαγή αυτή στην αρχιτεκτονική οδήγησε σε ραγδαία αύξηση της απόδοσης του δικτύου. Ανάλογη, όμως, ήταν και η αύξηση του συνολικού χρόνου εκτέλεσης, καθώς η εκπαίδευση των layers, εκτός του RBF, αποτελεί πιο χρονοβόρα διαδικασία, σε σχέση με την αντίστοιχη μόνο του outer layer.

6)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
500	16.13 ($\alpha = 30$)	3000	0.01	A	Χρήση αλγορίθμου k-μέσων	26.94%	27.68%	260.01	304.07

Το πείραμα 6) είναι παρόμοιο με το πείραμα 1), με διαφορά στον τρόπο εκπαίδευσης του RBF layer. Παρόλο που τα ποσοστά επιτυχίας στο training και στο test εμφανίστηκαν παρόμοια για τα δύο πειράματα (η εκπαίδευση με τυχαία επιλογή κέντρων αποδείχθηκε, μάλιστα, καλύτερη), ο χρόνος εκπαίδευσης του RBF layer, και κατ' επέκταση ο συνολικός χρόνος εκτέλεσης, σημειώθηκε πολύ μεγαλύτερος για το πείραμα 6.

7)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	11.77 ($\alpha = 30$)	3000	0.01	A	Χρήση αλγορίθμου k-μέσων	30.23%	30.95%	244.63	293.92

Η σύγκριση των πειραμάτων 7) και 2) οδηγεί στα ίδια συμπεράσματα με την αντίστοιχη των πειραμάτων 6) και 1). Τα ποσοστά σωστών προβλέψεων είναι ,αυτή τη φορά, ελαφρώς μεγαλύτερα για την εκπαίδευση με k-μέσων, ενώ η εκπαίδευση του RBF layer με τον αλγόριθμο k-μέσων κοστίζει πολύ περισσότερο σε χρόνο.

8)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	14.59 ($\alpha = 40$)	3000	0.02	A	Χρήση αλγορίθμου k-μέσων	31.36%	31.72%	264.22	305.09

Στο πείραμα 8), έχουν πραγματοποιηθεί αλλαγές στην τιμή της παραμέτρου σ και του βήματος, σε σχέση με το πείραμα 7). Τα training και test accuracies εμφανίζονται αυξημένα. Η ελαφριά αύξηση των χρόνων κρίνεται συμπτωματική.

9)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
2000	10.35 ($\alpha = 40$)	3000	0.02	A	Χρήση αλγορίθμου k-μέσων	34.73%	35.08%	350.30	404.16

Για το πείραμα 9), επιλέχθηκαν οι ίδιες τιμές για τις διάφορες παραμέτρους με τις αντίστοιχες του πειράματος 8), με μόνη διαφορά το πλήθος κέντρων. Η αλλαγή αυτή οδήγησε σε σημαντική αύξηση του training και του test accuracy. Ο χρόνος εκπαίδευσης του RBF layer, ωστόσο, αυξήθηκε αρκετά.

Για τα πειράματα 10) και 11), η παράμετρος σ ορίστηκε στην τιμή 10, ώστε να συγκριθεί η απόδοση δύο δικτύων με ίδιες τιμές όλων των παραμέτρων, εκτός από το πλήθος των κέντρων του RBF layer. Στα δύο δίκτυα αυτά έχουν προστεθεί δύο ακόμα hidden layers, αποκτώντας την αρχιτεκτονική B.

10)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
500	10	3000	0.02	B	Χρήση αλγορίθμου k-μέσων	35.08%	35.43%	229.84	282.55

11)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Αριθμός εποχών	Βήμα (learning rate)	Αρχιτεκτονική	Τρόπος εκπαίδευσης RBF layer	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	10	3000	0.02	B	Χρήση αλγορίθμου k-μέσων	35.38%	35.99%	270.05	380.44

Αρχικά, όπως και στην περίπτωση της τυχαίας επιλογής κέντρων, η αύξηση των hidden layers του δικτύου οδήγησε σε αρκετά υψηλότερα ποσοστά επιτυχίας στις προβλέψεις. Όσον αφορά την σύγκριση των δικτύων στα πειράματα 10) και 11), παρατηρείται υπεροχή του δικτύου με τα 1000 κέντρα (πείραμα 11) όσον αφορά το training και το test accuracy. Ωστόσο, ο χρόνος εκπαίδευσης του RBF layer και ο συνολικός χρόνος εκτέλεσης είναι αρκετά μεγαλύτερα σε σχέση με το δίκτυο των 500 κέντρων.

Γενικές παρατηρήσεις

- Στα πειράματα που πραγματοποιήθηκε τυχαία επιλογή των κέντρων, ο χρόνος εκπαίδευσης του RBF layer ήταν πολύ μικρός, σε αντίθεση με τα πειράματα όπου χρησιμοποιήθηκε ο αλγόριθμος k-μέσων για τον προσδιορισμό των κέντρων. Παρά την μεγάλη αυτή διαφορά των δύο μεθόδων αναφορικά με την ταχύτητα τους, τα ποσοστά επιτυχίας στο training και στο test δεν εμφάνισαν ανάλογες διαφορές, με τις τιμές τους να είναι παρόμοιες. Ως συμπέρασμα, η μέθοδος επιλογής κέντρων τυχαία κρίνεται αποδοτικότερη.
- Παρατηρείται ότι σε όλα τα πειράματα το test accuracy προέκυψε μεγαλύτερο από το training accuracy. Σε ένα RBF νευρωνικό δίκτυο, αυτό πιθανόν οφείλεται στο γεγονός ότι η εκπαίδευση του RBF layer πραγματοποιείται χωρίς επίβλεψη, δηλαδή βασίζεται αποκλειστικά στη γεωμετρική διάταξη των δεδομένων εισόδου και όχι στους στόχους που θέλουμε να επιτύχουμε. Ως αποτέλεσμα, το δίκτυο μπορεί να μην προσαρμοστεί βέλτιστα στα δεδομένα εκπαίδευσης, οδηγώντας σε χαμηλότερη ακρίβεια σε σχέση με τα δεδομένα ελέγχου. Ακόμη, η χρήση της ίδιας τιμής για την παράμετρο σ σε όλα τα κέντρα ενδέχεται να μην επιτρέπει στο δίκτυο να προσαρμοστεί επαρκώς στις τοπικές ιδιαιτερότητες των δεδομένων εκπαίδευσης.
- Αυτή η μέθοδος εκπαίδευσης των layers, εκτός του RBF layer, δεν αποδείχθηκε ιδιαίτερα αποδοτική. Ο χρόνος εκτέλεσης ήταν μεγάλος, χωρίς να επιτυγχάνονται ιδιαίτερα σημαντικά ποσοστά training και test accuracy.

Εκπαίδευση του outer layer με την μέθοδο Least Squares

Στα πειράματα που ακολουθούν, το δίκτυο αποτελείται από το RBF layer και το outer layer, καθώς η μέθοδος Least Squares δεν μπορεί να εφαρμοστεί εύκολα σε πολυπλοκότερα layers, όπως αναφέρθηκε παραπάνω. Δεδομένου ότι ο αλγόριθμος αυτός δεν είναι επαναληπτικός, δεν ορίζονται αριθμός εποχών και τιμή βήματος.

Όπως και στα παραπάνω πειράματα, η τιμή της παραμέτρου σ υπολογίστηκε σύμφωνα με τον τύπο $\sigma = \frac{d}{\sqrt{2 \cdot n}} * a$, όπου d ο μέσος όρος των αποστάσεων όλων των κέντρων μεταξύ τους, n ο αριθμός των κέντρων και a ένας επιλεγμένος σταθερός όρος. Για τον ορισμό του όρου a , πραγματοποιήθηκαν πειράματα με διαφορετικές τιμές του σε υποσύνολα του data set. Κατόπιν, για τα πειράματα στο σύνολο του data set επιλέχθηκαν οι τιμές του a οι οποίες παρουσίασαν το μεγαλύτερο ενδιαφέρον. Σε ορισμένα πειράματα, η τιμή του σ επιλέχθηκε χωρίς την χρήση του παραπάνω τύπου.

Στα τρία παρακάτω πειράματα, η επιλογή των κέντρων πραγματοποιήθηκε τυχαία μεταξύ των δειγμάτων εκπαίδευσης. Επιπλέον, η τιμή της παραμέτρου σ υπολογίστηκε με βάση τον παραπάνω τύπο, όπου ο όρος a ορίστηκε 30.

1)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
500	16.86	48.86%	48.11%	0.002	0.52

2)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1000	12.07	52.47%	50.26%	0.01	0.77

3)

Αριθμός κέντρων	Τιμή παραμέτρου σ	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
2000	8.51	56.95%	51.42%	1.29	3.41

Τα συμπεράσματα που προκύπτουν από τα τρία αυτά πειράματα συμβαδίζουν με τις αντίστοιχες των προηγούμενων πειραμάτων, όπου χρησιμοποιήθηκε ο αλγόριθμος delta rule. Παρατηρείται ότι η αύξηση των κέντρων οδηγεί σε ανάλογη μεταβολή στα ποσοστά επιτυχίας στο training και στο test. Συνοδεύεται, επιπλέον, από ραγδαία αύξηση του χρόνου εκπαίδευσης του RBF layer (δηλαδή του χρόνου που

απαιτείται για να υπολογιστεί η τιμή της παραμέτρου σ σύμφωνα με τον τύπο), καθώς και ο συνολικός χρόνος εκτέλεσης.

Στα πέντε πειράματα που ακολουθούν, μόνη διαφοροποίηση αποτελεί το πλήθος των κεντρών του RBF layer. Η επιλογή τους έχει πραγματοποιηθεί τυχαία μεταξύ των δειγμάτων εκπαίδευσης. Η τιμή της παραμέτρου σ έχει οριστεί 10 για όλα τα πειράματα.

Πείραμα	Αριθμός κέντρων	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1	500	48.56%	47.13%	$5.1 \cdot 10^{-5}$	0.48
2	1000	52.15%	49.84%	$3.85 \cdot 10^{-5}$	0.67
3	2000	57.49%	52.11%	$4.35 \cdot 10^{-5}$	2.18
4	3000	60.85%	52.83%	$4.67 \cdot 10^{-5}$	3.55
5	4000	63.98%	53.83%	$5.55 \cdot 10^{-5}$	5.83

Σε αυτό το σύνολο πειραμάτων, επιβεβαιώνεται ότι, για σταθερό σ , η αύξηση του πλήθους των κέντρων οδηγεί σε αύξηση των επιδόσεων του δικτύου αναφορικά με τα ποσοστά σωστών κατηγοριοποιήσεων. Αξίζει να σημειωθεί ότι καθώς μεγαλώνει ο αριθμός των κέντρων, η αύξηση του ποσοστού επιτυχίας στο test δεν είναι ανάλογη. Αντίθετα, καθώς τα κέντρα πληθαίνουν όλο και περισσότερο, ο ρυθμός αύξησης του ποσοστού αυτού μειώνεται. Το ίδιο δεν παρατηρείται για το ποσοστό training accuracy, το οποίο μεγαλώνει με ταχύ ρυθμό. Επομένως, ενδέχεται, με περαιτέρω αύξηση του αριθμού των κέντρων, το test accuracy να αρχίσει να μειώνεται, γεγονός που υποδεικνύει την εμφάνιση του φαινομένου της υπερεκπαίδευσης (overfitting).

Στα επτά πειράματα που ακολουθούν, μόνη διαφοροποίηση αποτελεί η τιμή της παραμέτρου σ . Το πλήθος των κέντρων έχει οριστεί 1000 και η επιλογή τους έχει πραγματοποιηθεί τυχαία μεταξύ των δειγμάτων εκπαίδευσης.

Πείραμα	Τιμή παραμέτρου σ	Training accuracy	Test accuracy	Χρόνος εκπαίδευσης RBF layer (σε λεπτά)	Συνολικός χρόνος εκτέλεσης κώδικα (σε λεπτά)
1	5	48.06%	45.69%	$3.75 \cdot 10^{-5}$	0.73
2	10	52.15%	49.84%	$3.85 \cdot 10^{-5}$	0.67
3	15	52.92%	50.24%	$5.04 \cdot 10^{-5}$	0.73
4	40	53.29%	49.73%	$4.39 \cdot 10^{-5}$	0.74
5	60	53.13%	50.49%	$3.9 \cdot 10^{-5}$	0.72
6	100	53.42%	49.94%	$3.6 \cdot 10^{-5}$	0.72
7	200	53.03%	49.89%	$3.66 \cdot 10^{-5}$	0.69

Τα αποτελέσματα αυτού του συνόλου πειραμάτων οδηγούν στο συμπέρασμα ότι η αύξηση της τιμής της παραμέτρου σ επιδρά θετικά στα ποσοστά training και test accuracy. Για το τελευταίο, αυτό παρατηρείται μέχρι μία συγκεκριμένη τιμή, από την οποία και έπειτα το ποσοστό σωστών προβλέψεων στο test αρχίζει να μειώνεται. Για το αντίστοιχο ποσοστό στο training, τα αποτελέσματα δεν οδήγησαν σε ασφαλή ανάλογη διαπίστωση. Ο χρόνος εκπαίδευσης του RBF layer και ο συνολικός χρόνος εκτέλεσης δεν δείχνουν να επηρεάζονται από τις μεταβολές στην τιμή της παραμέτρου σ .

Σύγκριση απόδοσης με άλλους αλγόριθμους κατηγοριοποίησης:

Τα παρακάτω αποτελέσματα αντλήθηκαν από την ενδιάμεση εργασία.

Είδος κατηγοριοποιητή	Ποσοστό επιτυχίας προβλέψεων στο test	Χρόνος εκτέλεσης
1-NN	35.39%	85 λεπτά
Nearest Centroid	27.74%	2.04 δευτερόλεπτα

Το RBF νευρωνικό δίκτυο αποδείχθηκε, με μεγάλη διαφορά, αποδοτικότερο με τυχαία επιλογή των κέντρων και εκπαίδευση του outer layer σύμφωνα με τον αλγόριθμο Least Squares. Ένα τέτοιο δίκτυο, όπως αποδείχθηκε από τα παραπάνω πειράματα, επιτυγχάνει ποσοστό test accuracy μεγαλύτερο από 50%, με κατάλληλη επιλογή παραμέτρων, σε χρόνο λίγων λεπτών.

Συγκριτικά με τις επιδόσεις των κατηγοριοποιητών πλησιέστερου γείτονα και πλησιέστερου κέντρου, το RBF νευρωνικό δίκτυο υπερτερεί σημαντικά. Σε σχέση με τον πρώτο κατηγοριοποιητή, το δίκτυο επιτυγχάνει πολύ μεγαλύτερο ποσοστό επιτυχίας προβλέψεων σε πολύ μικρότερο χρόνο. Ακόμη, συγκρίνοντας το με τον δεύτερο κατηγοριοποιητή, το ποσοστό test accuracy του δικτύου είναι σχεδόν διπλάσιο. Ωστόσο, ο χρόνος που απαιτεί το δίκτυο είναι αισθητά μεγαλύτερος από τα δύο δευτερόλεπτα που χρειάζονται για την εκπαίδευση του κατηγοριοποιητή πλησιέστερου κέντρου, χωρίς, όμως, να καθίσταται ιδιαίτερα αργό.