

# Νευρωνικά Δίκτυα - Βαθιά Μάθηση

Τμήμα Πληροφορικής ΑΠΘ 7ο εξάμηνο

## Εργασία 3 - RBF Neural Network

Αλεξία Νταντουρή AEM: 3871

### 3η Εργασία

Να γραφεί πρόγραμμα σε οποιαδήποτε γλώσσα προγραμματισμού το οποίο να υλοποιεί ένα

1. **Radial Basis Function Neural Network** ή
2. ένα δίκτυο που εκπαιδεύεται με **Hebbian Learning** ή
3. ένα **δίκτυο αυτοκωδικοποίησης (autoencoder)**

που θα εκπαιδευτεί για να επιλύει

- ένα από τα προβλήματα που επιλύσατε στις προηγούμενες εργασίες σας ή
- αντίστοιχο πρόβλημα προσέγγισης συνάρτησης ή
- ανακατασκευή δεδομένων (Σε αυτήν την περίπτωση μπορείτε να χρησιμοποιήσετε την MNIST και το δίκτυο θα κατασκευάζει το επόμενο ψηφίο, δηλαδή αν δεχτεί στην είσοδο ένα 3 να κατασκευάσει ένα 4).

### Σημείωση

Προτείνεται να διαβαστεί το παρόν report παράλληλα με python notebook.

### Data Preprocessing

Εφάρμοσα Principal Component Analysis (PCA) στα δεδομένα και κράτησα 90% της πληροφορίας, ρίχνοντας των αριθμό των features από 3071 σε 99 (-96.7773% μείωση).

### Custom RBF Layer

Χρησιμοποίησα ένα custom RBF layer ([https://github.com/PetraVidnerova/rbf\\_keras](https://github.com/PetraVidnerova/rbf_keras)) για τη δημιουργία του RBF νευρωνικού δικτύου.

### Hyperparameter tuning

Χρησιμοποίησα τη συνάρτηση **RandomSearchCV** της sklearn για να βρω υπερπαραμέτρους για τη βελτίωση της ακρίβειας του μοντέλου, η οποία έδωσε το παρακάτω αποτέλεσμα.

```
Best model: 0.384300 using {'optimizer': 'adam', 'nodes': 100, 'epochs': 50, 'betas': 0.1, 'batch_size': 40}
```

Πειραματίστηκα, στη συνέχεια, με μοντέλα με διαφορετικό πλήθος νευρώνων και δοκίμασα διαφορετικές τιμές στις υπερπαραμέτρους για να δω πως επηρεαζόταν η ακρίβεια του μοντέλου.

## Different Number of Neurons

For K-Means initializer:

	Train Accuracy	Test Accuracy
Neurons = 120	0.4121	0.4067
Neurons = 150	0.4300	0.4269
Neurons = 200	0.4273	0.4261

Χρησιμοποιώντας k-means για αρχικοποίηση των κέντρων, το μοντέλο που δίνει την υψηλότερη ακρίβεια είναι αυτό με 150 νευρώνες στο κρυφό επίπεδο.

## K-Means vs Random Centers Initializer

	Train Accuracy	Test Accuracy
K-Means	0.4321	0.4242
Random Centers	0.4436	0.4397

Ανάμεσα σε k-means και τυχαία επιλογή σημείων για αρχικοποίηση των κέντρων, φαίνεται πως η επιλογή τυχαίων σημείων δίνει τα καλύτερα αποτελέσματα.

## Comparing Results

Ο κατηγοριοποιητής πλησιέστερου γείτονα (KNN) δεν έχει φάση εκπαίδευσης, γι' αυτό στις στήλες των KNN έβαλα τους χρόνους πρόβλεψης των train και test sets.

	NCC	KNN 1	KNN 3	FC MLP	CNN	SVM	SVM VOTING	CNN passed to SVM	RBF NN
Training time	0.7	574 (train set) 129 (test set)	556 (train set) 109 (test set)	53	745	337	5053	915	1583
Train accuracy	26.968%	100%	57.904%	77.094%	84.02%	86.304%	90.184%	88.128%	44.36%
Test accuracy	27.74%	35.39%	33.03%	56.03%	71.9%	56.36%	56.95%	72.4%	43.97%

Κατάταξη των μοντέλων σε φθίνουσα σειρά ως προς το test accuracy:

	Train Accuracy	Test Accuracy	Training time	
1. CNN + SVM	88.128%	72.4%	915	
2. CNN	84.02%	71.9%	745	
3. SVM Voting	90.184%	56.95%	5053	
4. SVM	86.304%	56.36%	337	
5. FC MLP	77.094%	56.03%	53	
6. RBF NN	44.36%	43.97%	1583	
7. KNN 1	100%	35.39%	574 (train set)	129 (test set)
8. KNN 3	57.904%	33.03%	556 (train set)	109 (test set)
9. NCC	26.968%	27.74%	0.7	

Παρατηρούμε πως ο συνδυασμός του συνελικτικού νευρωνικού δικτύου (CNN) με το SVM είναι το καλύτερο μοντέλο όσον αφορά το test accuracy, με δεύτερο καλύτερο το CNN μόνο του και τρίτο καλύτερο το SVM Voting Ensemble.

Το απλό SVM μπορεί να έχει λίγο μικρότερη ακρίβεια από το SVM Voting Ensemble, αλλά εκπαιδεύεται πολύ πιο γρήγορα από το SVM Voting Ensemble.

Το πλήρως συνδεδεμένο νευρωνικό δίκτυο πολυστρωματικού perceptron (FC MLP) δίνει μικρότερη ακρίβεια σε σχέση με τα παραπάνω μοντέλα.

Όσον αφορά, το RBF Neural Network, φαίνεται πως εκπαιδεύεται αργά κι έχει κακά αποτελέσματα σε σχέση με τα παραπάνω μοντέλα, αλλά δίνει καλύτερα αποτελέσματα σε σχέση με τον K Nearest Neighbors (KNN) και τον Nearest Class Centroid (NCC).