Εθνικο Μετσοβιο Πολυτεχνειο
Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Διατμηματικο Προγραμμα Μεταπτυχιακων Σπουδων
Επιστημη Δεδομενων και Μηχανικη Μαθηση

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων
Εργαστηριο Υπολογιστικων Συστηματων

Τεχνικές αξιοποίησης της κατανομής των δεδομένων στην εκπαίδευση νευρωνικών δικτύων

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόδημου Κ. Προβατά

Επιβλέπων: Νεκτάριος Κοζύρης
Καθηγητής Ε.Μ.Π.

Αθήνα, Νοέμβριος 2020

Εθνικο Μετσοβιο Πολυτεχνειο
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΤΕΧΝΟΛΟΓΙΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ
ΚΑΙ ΥΠΟΛΟΓΙΣΤΩΝ
ΕΡΓΑΣΤΗΡΙΟ ΥΠΟΛΟΓΙΣΤΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

Τεχνικές αξιοποίησης της κατανομής των δεδομένων στην εκπαίδευση νευρωνικών δικτύων

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

Νικόδημου Κ. Προβατά

Επιβλέπων: Νεκτάριος Κοζύρης
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 13η Νοεμβρίου 2020:

.....	.....	.....
Νεκτάριος Κοζύρης Καθηγητής Ε.Μ.Π.	Γεώργιος Γκούμας Επικ. Καθηγητής Ε.Μ.Π.	Ιωάννης Κωνσταντίνου Επικ. Καθηγητής Παν. Θεσσαλίας

Αθήνα, Νοέμβριος 2020.

.....
Νικόδημος Κ. Προβατάς
Κάτοχος Μεταπτυχιακού Διπλώματος στον τομέα της
Επιστήμης Δεδομένων και Μηχανικής Μάθησης

Περίληψη

Ο τομέας της βαθιάς μηχανικής μάθησης παρουσιάζει ιδιαίτερη άνθηση στο πλαίσιο μιας πληθώρας διαφορετικών εφαρμογών, όπως είναι η κατηγοριοποίηση εικόνων. Για την κατασκευή καλύτερων προβλεπτικών μοντέλων έχει μια αναπτυχθεί μια πληθώρα πολύπλοκων αρχιτεκτονικών νευρωνικών δικτύων. Ωστόσο, η πολυπλοκότητα ενός δικτύου δεν επαρκεί από μόνη της για την δημιουργία μοντέλων υψηλής ακρίβειας, η οποία συνδέεται άμεσα με την ποιότητα και τον σειρά πρόσβασης στα δεδομένα κατά τη διάρκεια της εκπαίδευσης. Συνήθως, για την αύξηση της ακρίβειας ενός μοντέλου, κάθε βήμα της εκπαίδευσης πραγματοποιείται με τη χρήση τυχαίου υποσυνόλου των δεδομένων αντί σειριακά. Στη συγκεκριμένη εργασία εξετάζονται τεχνικές για την αξιοποίηση της πληροφορίας που παρέχει η κατανομή των δεδομένων, ώστε να ωφεληθεί η διαδικασία της εκπαίδευσης. Αρχικά, γίνεται μελέτη για τον καθορισμό της σειράς πρόσβασης στα δεδομένα σύμφωνα με την στατιστική κατανομή που τα παράγει βελτιώνοντας έτσι τις μετρικές αξιολόγησης έως και 4.58%, χωρίς ιδιαίτερη χρονική επιβάρυνση. Επιπλέον, εξετάζεται πως μπορεί να αξιοποιηθεί η κατανομή των δεδομένων, ώστε η εκπαίδευση να πραγματοποιηθεί σε συγκεκριμένο υποσύνολο του συνόλου εκπαίδευσης. Επιτυγχάνεται έτσι η δημιουργία προβλεπτικών μοντέλων αντίστοιχης εκπαίδευσης σε περίπου $1.2X$ λιγότερο χρόνο.

Λέξεις - Κλειδιά: βαθιά μηχανική μάθηση, κατανομές δεδομένων, κατηγοριοποίηση εικόνας, ομαδοποίηση, δειγματοληψία

Abstract

Deep learning is particularly flourishing in a context of different applications, such as image classification. In order to train better predictive models a plethora of complex neural networks architectures has been developed. However, network's complexity alone is not sufficient for the creation of high-precision models, which is directly linked to the quality and the data access pattern during training. A common practise to increase the accuracy of a model is to use a random batch of the data during the training instead of accessing them in a serial manner. In this diploma thesis, we examine the use of data distribution related techniques which will benefit the training process. Initially, we experiment with defining data access pattern according to their statistical distribution and produce validation metrics by up to 4.58%, without a significant time overhead. In addition, how data distribution can be utilized to perform the training process on a specific subset of the training set. Thus, we achieve the creation of equivalent predictive models in $1.2X$ times less time.

Keywords: deep learning, data distribution, image classification, clustering, sampling

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον καθήγητη μου, κύριο Νεκτάριο Κοζύρη, και το Εργαστήριο Υπολογιστικών Συστημάτων όπου εργάζομαι και στα πλαίσια του όποιου εκπονήθηκε η παρούσα εργασία και παράλληλα εξελλίσσεται η διδακτορική μου διατριβή. Επίσης, θα ήθελα να ευχαρίστησω και τον Γιάννη Κωνσταντίνου, με τον οποίο συνεργάζομαι στα πλαίσια των υποχρεώσεων μου στο εργαστήριο και πολλές φορές μου έδωσε το περιθώριο του χρόνου όταν ήταν απαραίτητο για να ασχολήθω με τις υποχρεώσεις του μεταπτυχιακού προγράμματος. Παράλληλα, οφείλω ένα μεγάλο ευχαριστώ στον πατέρα μου για όλη την βοήθεια και την υποστήριξη που μου έχει δώσει όλα αυτά τα χρόνια. Ολοκληρώνοντας θέλω να ευχαριστήσω και τους φίλους μου και τη σχέση μου για την στήριξη και την κατανόηση τους σε αυτήν την δύσκολη περίοδο εκπόνησης της διπλωματικής εργασίας.

Στον πατέρα μου Κώστα
και στη μνήμη
της μητέρας μου Γιώτας

Περιεχόμενα

Κατάλογος Εικόνων

Κατάλογος Πινάκων

Κατάλογος Αλγορίθμων

Κεφάλαιο 1

Εισαγωγή

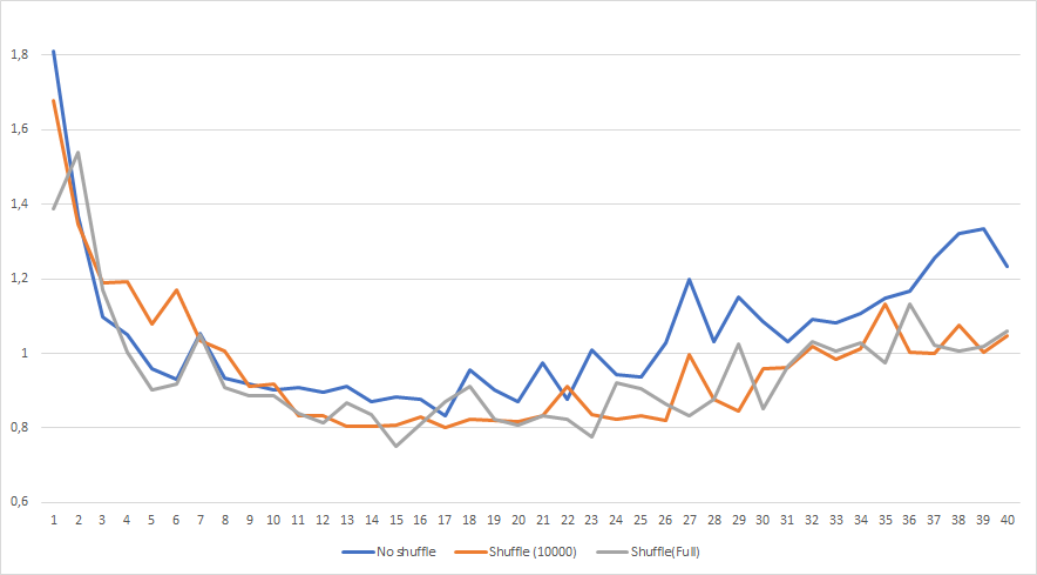
1.1 Κίνητρο της εργασίας

Τα τελευταία χρόνια ο τομέας της βαθιάς μηχανικής μάθησης έχει αποτελέσει ένα πολύ σημαντικό χαρακτηριστικό μίας πληθώρας εφαρμογών, όπως είναι η κατηγοριοποίηση εικόνων [1] και η αναγνώριση φωνής [2]. Τέτοιες εφαρμογές απαιτούν διαχείριση μία πληθώρας δεδομένων για να κατασκευαστούν μοντέλα νευρωνικών δικτύων τα οποία θα μπορούν να υποστηρίξουν επαρκώς την πληροφορία που κρύβεται πίσω από τα δεδομένα. Για παράδειγμα, το 2015, η Microsoft πρότεινε την αρχιτεκτονική ResNet που αποτελείται από 152 επίπεδα και εκατομμύρια παραμέτρων για να ελαχιστοποίησει το σφάλμα πρόβλεψης στο σύνολο ImageNet [3].

Ωστόσο, η ανάπτυξη πολύπλοκων αρχιτεκτονικών νευρωνικών δικτύων δεν είναι αρκετή για την δημιουργία ενός αποτελεσματικού και ακριβούς προβλεπτικού μοντέλου. Ισχυρό ρόλο στην ποιότητα των προβλέψεων ενός μοντέλου παίζουν τα δεδομένα εκπαίδευσης. Η ποιότητα των δεδομένων εκπαίδευσης, είτε απο άποψη ποικιλίας είτε από άποψη αντιπροσωπευτικότητας του χώρου όπου περιγράφουν, έχει σημαντικό αντίκτυπο στο μοντέλο το οποίο θα προκύψει από την διαδικασία εκμάθησης [4]. Ένα ακόμα σύνηθες πρόβλημα είναι το σύνολο δεδομένων εκπαίδευσης να μην έχει ομοιόμορφη κατανομή ως προς τις ετικέτες κατηγοριοποίησης, με αποτέλεσμα να προκύπτουν μοντέλα τα οποία είναι προκατειλημμένα ως προς την πρόβλεψη συγκεκριμένης κατηγορίας [5].

Για να αντιμετωπιστούν πιθανά προβλήματα που υπεισέρχονται στην εκπαίδευση από τα δεδομένα, μία συνήθης τακτική είναι η χρήση μηχανισμού που ανακατεύει τα δεδομένα, πριν την εξαγωγή κάθε ενός υποσυνόλου που χρησιμοποιηθεί στο επόμενο βήμα της εκπαίδευσης. Η διαδικασία αυτή προσομοιώνει τη διαδικασία μίας τυχαίας δειγματοληψίας πάνω στα δεδομένα και είναι πιθανότερο έτσι να προκύψει ένα αντιπροσωπευτικό υποσύνολο των δεδομένων. Κατά αυτό τον τρόπο το μοντέλο σε κάθε βήμα εκπαίδευεται λαμβάνοντας υπόψιν μεγαλύτερο ποσοστό της ποικιλίας των δεδομένων. Γενικα, έχει παρατηρηθεί ότι η χρήση τυχαίου δείγματος σε κάθε βήμα της εκπαίδευσης, μπορεί να επιταγχύνει τη διαδικασία σύγκλισης της εκπαίδευσης του

νευρωνικού δικτύου [6].



Εικόνα 1.1: Παράδειγμα κόστους αξιολόγησης κατά την εκπαίδευση του δικτύου ResNet-10 με σειριακή και τυχαία πρόσβαση στα δεδομένα.

Στην Εικόνα 1.1 δίνεται το αποτέλεσμα της συνάρτησης κόστους για το σύνολο αξιολόγησης κατά την εκπαίδευση ενός δικτύου της αρχιτεκτονικής ResNet-10 πάνω στο σύνολο δεδομένων CIFAR-10,είτε πραγματοποιώντας τυχαία είτε σειριακή πρόσβαση στα δεδομένα κατά τη διάρκεια της διαδικασίας. Όπως είναι φανερό, η εκπαίδευση με την τυχαία δειγματοληψία μπορεί να βοηθήσει στην καλύτερη γενίκευση του μοντέλου, αφού παρατηρείται περαιτέρω μείωση της τιμής της συνάρτησης κόστους.

Με αφορμή το γεγονός ότι η τυχαία πρόσβαση στα δεδομένα επηρέαζει την ποιότητα του τελικού μοντέλου, γεννάται το ερώτημα εάν μία συστηματική μελέτη της κατανομής των δεδομένων πριν την εκπαίδευση θα ήταν αξιοποίησιμη για την εκπαίδευση του δικτύου. Στόχος της συγκεκριμένης διπλωματικής εργασίας είναι η μελέτη και αξιοποίηση της κατανομής των δεδομένων εκπαίδευσης για τον καθορίσμο της σειράς πρόσβασης στα δεδομένα αυτά κατά την εκπαίδευση, επιτυγχάνοντας την κατασκευή ενός ακόμα πιο αποδοτικού μοντέλου. Παράλληλα, γίνεται μελέτη του τρόπου αξιοποίησης της πληροφορίας σχετικά με την κατανομή, ώστε να μπορεί να μειώθει το μέγεθος του συνόλου εκπαίδευσης, εάν είναι εφικτό. Επιδιώκεται κατά αυτό τον τρόπο η κατασκευή ισοδύναμου μοντέλου, αλλά σε λιγότερο χρόνο.

1.2 Δομή της εργασίας

Η συγκεκριμένη διπλωματική εργασία διαρθρώνεται ως εξής:

- Στο **Κεφάλαιο 2** γίνεται μία βιβλιογραφική ανασκόπηση σχετικά με έννοιες βελτιστοποίησης στην μηχανική μάθηση.
- Στο **Κεφάλαιο 3** παρουσιάζονται βασικές έννοιες που αφορούν τα νευρωνικά δίκτυα.
- Στο **Κεφάλαιο 4** δίνονται απαραίτητες έννοιες σχετικά με κατανομές δεδομένων, που θα χρησιμοποιηθούν εκτενώς στα πλαίσια της εργασίας.
- Στο **Κεφάλαιο 5** παρουσιάζονται οι τεχνικές που θα μελετηθούν στα πλαίσια της παρούσας διπλωματικής εργασίας.
- Στο **Κεφάλαιο 6** γίνεται μία εκτενής πειραματική ανάλυση των τεχνικών που προτείνονται και παρατίθεται εκτενής σχολιασμός των αποτελεσμάτων.
- Στο **Κεφάλαιο 7** συνοψίζονται οι τεχνικές και τα αποτελέσματα τους, καθώς γίνεται και αναφορά σε πιθανές μελλοντικές επεκτάσεις της παρούσας εργασίας.

Κεφάλαιο 2

Βελτιστοποίηση στη Μηχανική Μάθηση

Στον παρόν κεφάλαιο παρουσιάζονται ορισμένες βασικές βιβλιογραφικές έννοιες σχετικά με τα προβλήματα ελαχιστοποίησης και τους τρόπους επίλυσης τους. Οι έννοιες αυτές είναι απαραίτητες για την πλήρη κατανόηση επόμενων εννοιών που θα παρατεθούν, όπως στην εκπαίδευση νευρωνικών δικτύων.

2.1 Γενικά περί Προβλημάτων Ελαχιστοποίησης

Η έννοια της βελτιστοποίησης αφορά στην εύρεση της βέλτιστης τιμής, σύμφωνα με κάποιο κριτήριο, για την επίλυση ενός μαθηματικού προβλήματος. Τέτοιου είδους προβλήματα εμφανίζονται σε μία πληθώρα πεδίων [7, 8]. Ένα οποιοδήποτε πρόβλημα μηχανικής ή βαθιάς μηχανικής μάθησης μπορεί να οριστεί ως ένα πρόβλημα ελαχιστοποίησης και σε αυτό το πλαίσιο αφορά η ανάλυση των προβλήματων βελτιστοποίησης στην παρούσα εργασία. Η βελτιστοποίηση συνήθως αφορά στην ελαχιστοποίηση μίας συνάρτησης, που ονομάζεται *συνάρτηση κόστους*, κάτω από κάποιες συνθήκες. Ένας μαθηματικός ορισμός ενός προβλήματος βελτιστοποίησης δίνεται στον Ορισμό 1 [9].

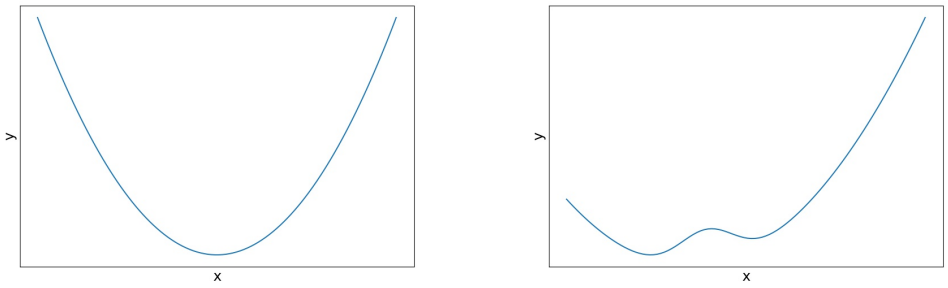
Ορισμός 1 (Πρόβλημα Βελτιστοποίησης) *Έστω $f : \mathbb{R}^n \longrightarrow \mathbb{R}$ μια συνάρτηση ορισμένη στην περιοχή $F \subseteq S$, όπου S κάποιος χώρος αναζήτησης που είναι υποσύνολο του $\mathbb{R}^n$. Ως πρόβλημα βελτιστοποίησης P ορίζουμε την εύρεση ενός $\vec{x_0} \in F$ τέτοιο ώστε*

$$f(\vec{x_0}) = \min_{x \in F} f(\vec{x})$$

Αν το σύνολο F δεν ταυτίζεται με το $\mathbb{R}^n$ τότε το πρόβλημα λύνεται κάτω από τους περιορισμούς που εισάγει η καμπύλη F. Τα προβλήματα βελτιστοποίησης διακρίνονται

σε δύο κατηγορίες, τα *κυρτά* και τα *μη κυρτά*. Ως κυρτά ορίζουμε τα προβλήματα βελτιστοποίησης σύμφωνα με τον Ορισμό 2. Οποιοδήποτε πρόβλημα βελτιστοποίησης δεν υπάγεται στον ορισμό αυτόν θεωρείται μη κυρτό πρόβλημα βελτιστοποίησης.

Ορισμός 2 (Κυρτό Πρόβλημα Βελτιστοποίησης) *Έστω P ένα πρόβλημα βελτιστοποίησης για την ελαχιστοποίηση μιας συνάρτησης $\mathbb{R}^n \longrightarrow \mathbb{R}$ στην περιοχή $F \in \mathbb{R}^n$. Το πρόβλημα βελτιστοποίησης είναι κυρτό αν και μόνο αν το σύνολο F είναι κυρτό και η συνάρτηση F είναι κυρτή.*



(α′) Κυρτή συνάρτηση (β′) Μη κυρτή συνάρτηση

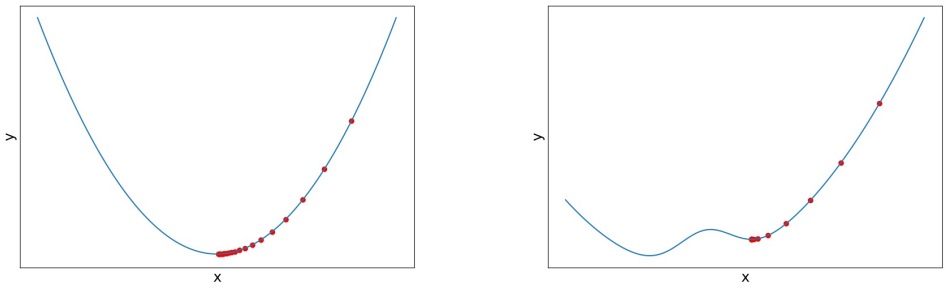
Εικόνα 2.1: Παραδείγματα κυρτής και μη κυρτής συνάρτησης.

Στην Εικόνα 2.1 δίνεται ένα παράδειγμα από μία κυρτή (2.1α′) και μία μη κυρτή συνάρτηση (2.1β′) αντιστοίχως. Όπως φανερώνεται και από την εικόνα, οι κυρτές συναρτήσεις έχουν ένα μοναδικό ολικό σημείο ελαχιστοποίησης, ενώ οι μη κυρτές έχουν τόσο τοπικά όσο και ολικά ελάχιστα. Στην συνέχεια, θα ασχοληθούμε με προβλήματα ελαχιστοποίησης τα οποία αφορούν την εκπαίδευση νευρωνικών δικτύων. Τα προβλήματα που αφορούν ελαχιστοποίηση της συνάρτησης κόστους στην περίπτωση της βαθιάς μηχανικής μάθησης είναι μη κυρτά προβλήματα βελτιστοποίησης [10], σε αντίθεση με τα απλά προβλήματα μηχανικής μάθησης που είναι κυρτά προβλήματα βελτιστοποίησης.

2.2 Ο αλγόριθμος Καθόδου Κλίσεων (Gradient Descent)

Ο αλγόριθμος καθόδου κλίσεων (Gradient Descent / GD) [11] είναι ένας από τους δημοφιλέστερους και πιο σημαντικούς αλγορίθμους στην περίπτωση της επίλυσης προβλημάτων βελτιστοποίησης. Πρόκειται για έναν τρόπο ελαχιστοποίησης της αντικειμενικής συνάρτησης ενός προβλήματος βελτιστοποίησης ενημερώνοντας την μεταβλητή ελαχιστοποίησης χρησιμοποιώντας το αντίθετο της κλίσης της αντικειμενικής συνάρτησης ως προς την πολυδιάστατη μεταβλητή ελαχιστοποίησης. Επιπλέον, πρόκειται για έναν πρώτης τάξης επαναληπτικό αλγόριθμο. Αξίζει να αναφερθεί ότι ο αλγόριθμος αυτός συγκλίνει σε κάποιο τοπικό ελάχιστο. Επομένως, σε κυρτές συνάρτησης εξασφαλίζεται η ελαχιστοποίηση τους, ενώ στις μη κυρτές συναρτήσεις εξασφαλίζεται η σύγκλιση σε κάποιο τοπικό ελάχιστο. Το μέγεθος του βήματος με το οποίο κατευθυνόμαστε προς το σημείο ελαχιστοποίησης της συνάρτησης καθορίζεται από τον ρυθμό μάθησης (learning rate - η). Ο αλγόριθμος εξελίσσεται σύμφωνα με την εξίσωση 2.1.

$$\vec{x_{n+1}} = \vec{x_n} + \eta \cdot \nabla_{\vec{x_n}} f(x_n) \tag{2.1}$$



(α′) Κυρτή συνάρτηση (β′) Μη κυρτή συνάρτηση

Εικόνα 2.2: Εφαρμογή Gradient Descent στις συναρτήσεις της Εικόνας 2.1

Ένα παράδειγμα εφαρμογής του αλγορίθμου Gradient Descent δίνεται στην εικόνα 2.2, στις συναρτήσεις της Εικόνας 2.1. Όπως αναφέρθηκε παραπάνω, στην περίπτωση της κυρτής συνάρτησης ο αλγόριθμος συγκλίνει επαναληπτικά στο ολικό ελάχιστο της συνάρτησης, ενώ στην περίπτωση της μη κυρτής συνάρτησης συγκλίνει στο τοπικό ελάχιστο αυτής.

(α′) Μεγάλος Ρυθμός Μάθησης (β′) Μικρός Ρυθμός Μάθησης

Εικόνα 2.3: Επίδραση του μεγέθους του ρυθμού μάθησης

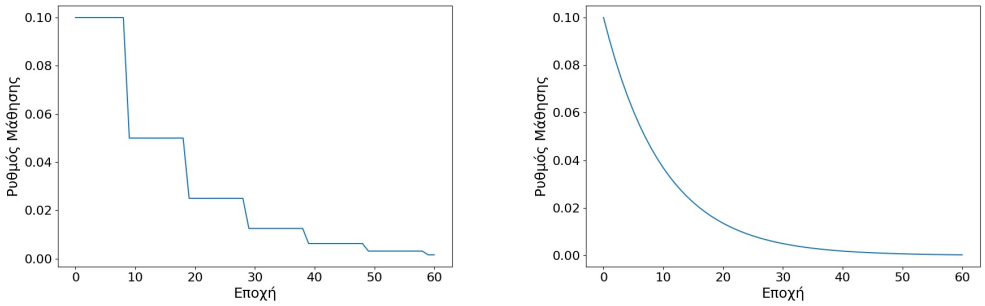
Είναι πολύ σημαντικό να αναφερθεί ότι ο ρυθμός μάθησης παίζει σημαντικό ρόλο στην εξέλιξη του επαναληπτικού αλγορίθμου. Στην Εικόνα 2.3 [12] βλέπουμε την εφαρμογή του αλγορίθμου καθόδου κλίσεων για μεγάλο 2.3α′ και μικρό 2.3β′ ρυθμό μάθησης. Γίνεται φανερό ότι η χρήση ενός μεγάλου ρυθμού μάθησης οδηγεί στο μετακινούμαστε εκατέρωθεν από το ελάχιστο. Παρότι θεωρητικά γίνεται μετακίνηση πιο γρήγορα προς το σημείο βελτιστοποίησης, το σημαντικά μεγάλο μέγεθος των βημάτων, οδηγεί σε μετακίνηση πιο μακριά από αυτό με αποτέλεσμα τελικά την καθυστέρηση της σύγκλισης στο επιθυμητό σημείο. Επιπλέον, η χρήση υπερβολικά μικρού ρυθμού μάθησης έχει τα ίδια αποτελέσματα, καθώς το βήμα είναι τόσο μικρό που αργεί η προσέγγιση του σημείου βελτιστοποίησης. Για τους παραπάνω λόγους είναι σημαντικός ο σωστός προσδιορισμός του ρυθμού μάθησης ώστε να μπορεί να προσεγγιστεί το σημείο βελτιστοποίησης με ικανοποιητική ταχύτητα.

2.3 Εφαρμογή Gradient Descent στην επίλυση προβλημάτων κατηγοριοποίησης με χρήση τεχνικών Μηχανικής Μάθησης.

Έστω ένα σύνολο k δεδομένων $\vec{x_i} \in \mathbb{R}^n$ και το αντίστοιχο σύνολο ετικετών κατηγοριοποίησης $y_i \in \mathbb{N}$, όπου $i \in \{1, 2, ..., k\}$. Εάν $\tilde{y} = m(\vec{x}, \vec{w})$ είναι ένα μοντέλο μηχανικής μάθησης, όπου $\vec{x}$ το δεδομένο εισόδου, $\vec{w}$ το σύνολο των παραμέτρων του και $\tilde{y}$ η έξοδος - πρόβλεψη του μοντέλου και ορίσουμε μια συνάρτηση κόστους ως προς την πρόβλεψη του μοντέλου έναντι στις πραγματικές ετικέτες του μοντέλου, το πρόβλημα εύρεσης του βέλτιστου μοντέλου, ανάγεται σε ένα πρόβλημα ελαχιστοποίησης της συνάρτησης κόστους ως προς την εύρεση των κατάλληλων παραμέτρων $\vec{w}$. Συμβολίζοντας την συνάρτηση κόστους ως $L(y; \tilde{y})$ (ή $L(\vec{w_i}; \vec{x_j}, y_j)$), ο αλγόριθμος 1 δείχνει την εφαρμογή της μεθόδου Gradient Descent στην εκπαίδευση ενός μοντέλου μηχανικής μάθησης. Κάθε μία από τις επαναλήψεις πάνω στο σύνολο δεδομένων που θα πραγματοποίησει ο αλγόριθμος μέχρι να συγκλίνει

Αλγόριθμος 1 Εφαρμογή Gradient Descent στην εκπαίδευση μοντέλων Μηχανικής Μάθησης σε προβλήματα κατηγοριοποίησης.

procedure GRADIENTDESCENT($S, \eta, m, L, epochs$)
    ▷ Εφαρμογή GD στο σύνολο δεδομένων $S = \{(\vec{x_1}, y_1), ..., (\vec{x_k}, y_k)\}$
    ▷ L : συνάρτηση κόστους
    Αρχικοποίηση $\vec{w_0}$ γύρω από το 0
    $i \leftarrow 0$
    while $i < epochs$ **do**
        for $\vec{x_j}, y_j \in S$ **do**
            $\tilde{y_j} = m(\vec{w_i}, \vec{x_j})$
        end for
        $\vec{w_{i+1}} = \vec{w_i} - \frac{\eta}{k} \cdot \sum_{j=1}^{k} \nabla_{\vec{w_i}} L(y_j; \tilde{y_j})$
        $i \leftarrow i + 1$
    end while
    return m
end procedure



(α′) Βηματική Μείωση ανά 5 εποχές (β′) Εκθετική Μείωση ανά εποχή

Εικόνα 2.4: Συνήθη Προγράμματα Μείωσης Ρυθμού Μάθησης

Η εφαρμογή του αλγορίθμου 1 οδηγεί στην εξερεύνηση της επιφάνειας του χώρου και στην μετακίνηση των παραμέτρων του μοντέλου προς τέτοια κατεύθυνση ώστε να ελαχιστοποιείται η συνάρτηση κόστους των πραγματικών κατηγοριών και των προβλέψεων του μοντέλου. Αξίζει να αναφερθεί ότι συνηθίζεται ο ρυθμός μάθησης να μεταβάλλεται κατά την εξέλιξη της εκπαίδευσης ανάλογα με την εποχή εκπαίδευσης στην οποία βρισκόμαστε [13–15]. Ενδεικτικά, οι πιο συνηθισμένες μεταβολές [16] του ρυθμού μάθησης κατά την εκπαίδευση παρουσιάζονται στην Εικόνα 2.4. Καθώς σε κάθε επανάληψη ο αλγόριθμος χρησιμοποιεί ολόκληρο το σύνολο δεδομένων, δεν είναι αποδοτικός, ωστόσο πάνω σε αυτόν στηρίζονται όλοι οι σύγχρονοι αλγόριθμοι βελτιστοποίησης που χρησιμοποιούνται στον τομέα της μηχανικής μάθησης.

2.4 Παραλλαγές του αλγορίθμου καθόδου κλίσεων

Είτε για λόγους ταχύτητας είτε για λόγους ποιότητας του αλγορίθμου έχουν αναπτυχθεί μια πληθώρα παραλλαγών του αλγορίθμου καθόδου κλίσεων. Στην παρούσα ενότητα γίνεται μια βιβλιογραφική ανασκόπηση μεθόδων βελτιστοποίησης που αποτελούν παραλλαγές του αλγορίθμου καθόδου κλίσεων και χρησιμοποιούνται στο πειραματικό μέρος της εργασίας. Η ανασκόπηση των αλγόριθμων θα γίνει ως προς την εφαρμογή τους σε προβλήματα μάθησης σύμφωνα με όσα αναφέρθηκαν στην ενότητα 2.3.

2.4.1 Ο Στοχαστικός Αλγορίθμος Καθόδου Κλίσεων (SGD)

Όπως αναφέρθηκε στην ενότητα 2.3, ο βασικός αλγόριθμος κλίσεων χρησιμοποιεί σε κάθε μία επανάληψη όλο το σύνολο δεδομένων. Η πιο βασική παραλλαγή του αλγορίθμου είναι ο Στοχαστικός Αλγόριθμος Καθόδου Κλίσεων (Stochastic Gradient Descent / SGD) [11]. Η παραλλαγή αυτή αντί σε κάθε βήμα της επαναληπτικής διαδικασίας χρησιμοποιεί μόνο ένα τυχαίο επιλεγμένο δεδομένο εκπαίδευσης από το σύνολο δεδομένων σε κάθε βήμα. Στην περίπτωση αυτή το βήμα 10 του αλγορίθμου 1 γράφεται σύμφωνα με την εξίσωση 2.2 για κάποιο τυχαία επιλέγμενο σημείο $(\vec{x_i}, y_i)$ στην i-οστή επαναλήψη της στοχαστικής μεθόδου.

$$\vec{w_{i+1}} = \vec{w_i} - \eta \cdot \nabla_{\vec{w_i}} L(\vec{w_i}; \vec{x_j}, y_j) \tag{2.2}$$

Καθώς ο SGD χρησιμοποιεί μόνο ένα σημείο σε κάθε βήμα του είναι σημαντικό να αναφερθεί ότι κάθε βήμα εκπαίδευσης πραγματοποιείται πολύ πιο γρήγορα σε σχέση με τον κλασσικό αλγόριθμο καθόδου κλίσεων και μπορεί να εφαρμοστεί για να εκπαιδευτούν μοντέλα από ροές δεδομένων. Ωστόσο, η συγκεκριμένη παραλλαγή, λόγω των συχνών ενημερώσεων των βαρών του μοντέλου, εισάγει μεγάλη διασπορά στις τιμές της αντικειμενικής συνάρτησης κόστους L. Γενικά, χωρίς την κατάλληλη τιμή του ρυθμού μάθησης υπάρχει πιθανότητα σύγκλισης της στοχαστικής έκδοσης σε διαφορετικό τοπικό ελάχιστο από τον κλασσικό αλγόριθμο. Ωστόσο, μειώνοντας σταδιακά τον ρυθμό μάθησης ο SGD θα συγκλίνει με παρόμοιο τρόπο με τον κλασσικό.

(α′) Εφαρμογή GD (β′) Εφαρμογή SGD

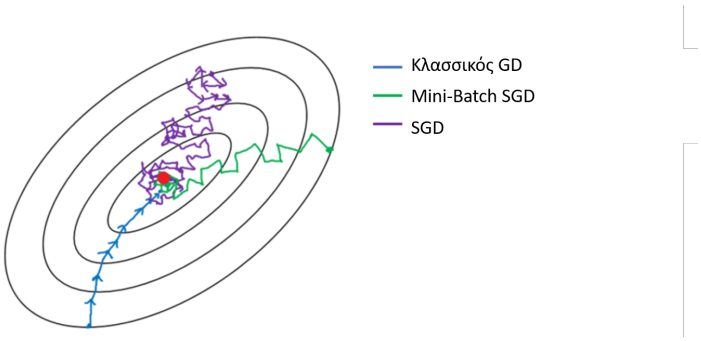
Εικόνα 2.5: Παραδείγμα Ισοϋψών Καμπυλών Συνάρτησης Κόστους με εφαρμογή του Gradient Descent και του Stochastic Gradient Descent

Για την καλύτερη κατανόηση της διασποράς που εισάγει ο SGD στη κατεύθυνση προς το ελάχιστο, στην Εικόνα 2.5 [17] δίνονται ένα παράδειγμα από ισοϋψείς καμπύλες μίας συνάρτησης κόστους της οποίας το ελάχιστο αναζητείται μέσω του κλασσικού αλγορίθμου κλίσεων (2.5α′) και του SGD (2.5β′). Η Εικόνα 2.5α′ εκφράζει ότι η συνάρτηση κόστους κατευθύνεται απευθείας προς το σημείο ελαχιστοποίησης της μέσω του κλασσικού αλγορίθμου κλίσεων, ενώ στην Εικόνα 2.5β′ γίνεται φανερό ότι η χρήση του SGD κατευθύνει την συνάρτηση κόστους με μία σειρά από παλινδρομικά βήματα γύρω από την ευθεία που ορίζει το αρχικό σημείο των παραμέτρων με τις παραμέτρους που ελαχιστοποιούν την συνάρτηση κόστους.

2.4.2 Ο Στοχαστικός Αλγορίθμος Καθόδου Κλίσεων με χρήση μικρο-ομάδων (Mini-Batch SGD)

Ο στοχαστικός αλγόριθμος καθόδου κλίσεων με χρήση μικρό-ομάδων (Mini-Batch Stochastic Gradient Descent / Mini-batch SGD) [18] αποτελεί μία μέθοδο που συνδυάζει τα στοιχεία της βασικής μεθόδου και της στοχαστικής μεθόδου που περιγράφηκε στην ενότητα 2.4.1. Η μέθοδος δεν χρησιμοποιεί ούτε μία μεμονωμένη εγγραφή, αλλά ούτε και ολόκληρο το σύνολο δεδομένων. Χρησιμοποιεί ένα υποσύνολο τυχαία επιλεγμένων εγγραφών, το πλήθος στοιχείων του οποίου ονομάζεται μέγεθος της μικρο-ομάδας (Mini-Batch Size B). Με αυτή την τροποποίηση, η γραμμή 10 του αλγορίθμου 1 γράφεται σύμφωνα με την εξίσωση 2.3

$$\vec{w}_{i+1} = \vec{w}_i - \frac{\eta}{B} \cdot \sum_{j=1}^{B} \nabla_{\vec{w}_i} L(\vec{w}_i; \vec{x_j}, y_j) \tag{2.3}$$



Εικόνα 2.6: Ισοϋψείς Καμπύλη Συνάρτησης Κόστους και συγκριση του Mini-Batch SGD, με SGD και GD

Η συγκεκριμένη μέθοδος είναι πιο γρήγορη από τον κλασσικό αλγόριθμο κλίσεων, ενώ παράλληλα επιτυγχάνει να μειώνει τη διασπορά που εισάγει η ταχύτερη στοχαστική μέθοδος της ενότητας 2.4.1. Για καλύτερη κατανόηση, στην Εικόνα 2.6 [19], δίνεται ένα παράδειγμα ισοϋψούς καμπύλης μιας συνάρτησης κόστους στην οποία εφαρμόζεται

ο αλγόριθμος κλίσεων και οι δύο παραλλαγές του που έχουν αναφερθεί μέχρι στιγμής. Η εικόνα επιβεβαιώνει την μειωμένη διασπορά αυτής της μεθόδου καθώς υπάρχουν λιγότερες αλλαγές κατεύθυνσης μέχρι το σημείο ελαχιστοποίησης σε σχέση με την απλή στοχαστική μέθοδο.

Τυπικά μεγέθη μικρο-ομάδας κυμαίνονται από 32 έως 1024 ανάλογα με τον αλγόριθμο μηχανικής μάθησης που χρησιμοποιείται. Η συγκεκριμένη μέθοδος έχει οδηγήσει στην κατασκευή πολλών άλλων παραλλαγών, οι οποίες είναι ιδιαίτερα χρήσιμες στην εκπαίδευση νευρωνικών δικτύων. Μία από αυτές παρουσιάζεται στην επόμενη ενότητα.

2.4.3 Ο Ορμητικός Βελτιστοποιητής

Ο ορμητικός βελτιστοποιητής (Momentum Optimizer) [20, 21] είναι μία επιταχυνόμενη παραλλαγή του Mini-Batch SGD . Για την επιτάχυνση της τεχνικής συμπεριλαμβάνεται στην ενημέρωση το διάνυσμα της ταχύτητας της κλήσης. Θεωρώντας των συντελεστή επιτάχυνσης μ, που εκφράζει το πόσο επιταχύνεται η διαδικασία, και $\vec{v}_i$ το διάνυσμα της ταχύτητας στην i-οστή επανάληψη του αλγορίθμου, η εξίσωση 2.3 αντικαθίσταται με το σύνολο των εξισώσεων 2.4 και 2.5.

$$\vec{v}_{i+1} = \mu \cdot \vec{v}_i - \frac{\eta}{B} \cdot \sum_{j=1}^{B} \nabla_{\vec{w}_i} L(\vec{w}_i; \vec{x_j}, y_j) \tag{2.4}$$

$$\vec{w}_{i+1} = \vec{w}_i + \vec{v}_{i+1} \tag{2.5}$$

Μία συνηθισμένη τιμή για τον συντελεστή επιτάχυνσης είναι το 0.9. Η επιτάχυνση που προσφέρει αυτή η παραλλαγή οφείλεται στη χρήση της ταχύτητας. Εάν η κλίση στο τρέχον βήμα του αλγορίθμου δεν έχει αλλάξει κατεύθυνση, τότε το διάνυσμα της ταχύτητας επιταχύνει το συνολικό βήμα. Αντίθετα, όταν προκύπτει αλλαγή κατεύθυνσης, τότε το διάνυσμα της ταχύτητας μειώνει το βήμα μετακίνησης των παραμέτρων. Έτσι, είναι δυνατό να αποφεύγονται φαινόμενα ταλάντωσης που εισάγονται από τις στοχαστικές μεθόδους και να εξασφαλιστεί ταχύτερη σύγκλιση προς τις παραμέτρους βελτιστοποίησης.

Εκτός από τον ορμητικό βελτιστοποιητή, ένα ευρύ σύνολο άλλων παραλλαγών του Mini-Batch SGD έχουν αναπτυχθεί για την εκπαίδευση νευρωνικών δικτύων, οι οποίες όμως δεν θα επεξηγηθούν στα πλαίσια της παρούσας διπλωματικής εργασίας. Ενδεικτικά, αναφέρονται η τεχνική επιταχυνόμενου διανύσματος κλίσης του Nesterov [22], η τεχνική προσαρμοσμένου διανύσματος κλίσης (Adagrad) [23] και η τεχνική εκτίμησης προσαρμοσμένης στιγμής (Adam) [24].

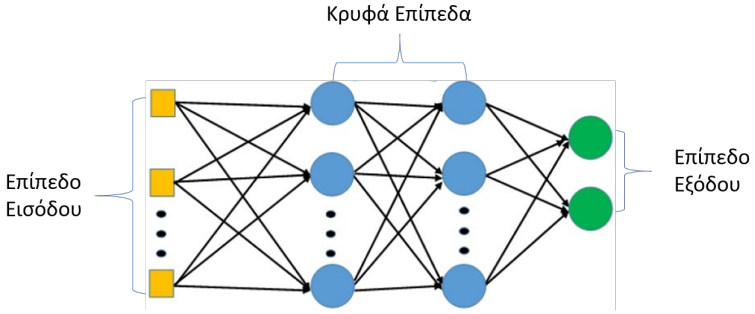
Κεφάλαιο 3

Νευρωνικά Δίκτυα

Στην συγκεκριμένη ενότητα γίνεται μια βιβλιογραφική ανασκόπηση σχετικά με τα νευρωνικά δίκτυα. Αρχικά θα σχολιαστούν επαρκώς τα τροφοδοτικά νευρωνικά δίκτυα, τα οποία αποτελούν την βάση για συγκεκριμένα είδη δικτύων και θα παρουσιαστεί τόσο ο τρόπος λειτουργίας όσο και ο τρόπος εκπαίδευσης τους, συνοδευόμενοι με τις απαραίτητες μαθηματικές έννοιες. Στη συνέχεια, θα γίνει μια μια συνοπτική ανασκόπηση της κατηγορίας των συνελικτικών νευρωνικών δικτύων, στα οποία προσανατολίζεται το πειραματικό μέρος της παρούσας διπλωματικής εργασίας. Σημειώνεται ότι καθώς θα χρησιμοποιηθούν εφαρμογές από τον τομέα της κατηγοριοποίησης, στην συνέχεια του παρόντος κεφαλαίου θα εξηγηθούν οι έννοιες προσαρμοσμένες σε τέτοιου τύπου προβλήματα.

3.1 Γενικά περί Τροφοδοτικών Νευρωνικών Δικτύων

Οι μέθοδοι βαθιάς μηχανικής μάθησης αποσκοπούν σε μια διαδικάσια εκμάθησης ιεραρχικών χαρακτηριστικών [25]. Πιο συγκεκριμένα, τα χαρακτηριστικά των υψηλότερων επιπέδων σχηματίζονται από την σύνθεση χαρακτηριστικών στα χαμηλότερα επίπεδα. Έτσι, η αλληλουχία πολλών επιπέδων που αποτελούνται από κρυφές μεταβλητές αξιοποιείται από βαθιές αρχιτεκτονικές εκμάθησης, ιδιαίτερα σε προβλήματα επιβλεπόμενης μάθησης. Τα πολυεπίπεδα τροφοδοτικά νευρωνικά δίκτυα αποτελούν το πιο δημοφιλές γενικό μοντέλο στον τομέα της βαθιάς μηχανικής μάθησης. Τα δίκτυα αυτά προσομοιώνουν την λειτουργία του ανθρώπινου εγκεφάλου, που είναι οργανωμένος σε νευρώνες, για να επιλύσουν το ζητούμενο πρόβλημα επιβλεπόμενης μάθησης [26]. Για να ακολουθήσουν αυτή την προσομοίωση τα δίκτυα οργανώνονται σε μια σειρά από επίπεδα, καθένα από τα οποία διαρθρώνεται σε νευρώνες. Οι πληροφορία μεταφέρεται ανάμεσα στα επίπεδα και όταν φτάσει στο τέλος δικτύου δίνει την επιθυμητή απάντηση, δηλαδή τη ζητούμενη κατηγορία ενός αντικείμενου στο θέμα της κατηγοριοποίησης.



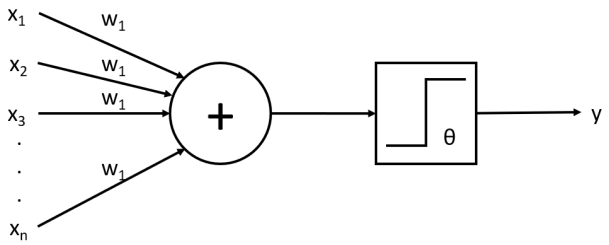
Εικόνα 3.1: Παραδείγμα ενός πολυεπίπεδου τροφοδοτικού νευρωνικού δικτύου.

Ένα τυπικό παράδειγμα πολυεπίπεδου τροφοδοτικού νευρωνικού δικτύου δίνεται στην Εικόνα 3.1 [27]. Το δίκτυο αποτελείται από μια σειρά από επίπεδα, τα οποία μεταφέρουν πληροφορία διαδοχικά το κάθε ένα στο επόμενο, μέσω των συνδέσεων που υπάρχουν μεταξύ τους. Έτσι η έξοδος του πρώτου επιπέδου αποτελεί είσοδο για το δεύτερο με τη διαδικασία να συνεχίζει μέχρι το τελευταίο επίπεδο. Κάθε νευρώνας του δικτύου είναι Τα επίπεδα ενός δικτύου διακρίνονται στις τρεις επόμενες βασικές κατηγορίες:

- **Επίπεδο Εισόδου:** Το συγκεκριμένο επίπεδο χρησιμοποιείται για την είσοδο ενός αντικειμένου προς κατηγοριοποίηση στο νευρωνικό δίκτυο. Κάθε ένας νευρώνας του επιπέδου αντιστοιχεί σε ένα συγκεκριμένο χαρακτηριστικό του αντικειμένου εισόδου.

- **Κρυφά Επίπεδα:** Ένα τροφοδοτικό νευρωνικό δίκτυο μπορεί να αποτελείται από ένα ή περισσότερα κρυφά επίπεδα. Στο παράδειγμα της Εικόνας 3.1 το δίκτυο αποτελείται από δύο κρυφά επίπεδα. Κάθε κρυφό επίπεδο εφαρμόζει ένα μη γραμμικό μετασχηματισμό στην έξοδο του αμέσως προηγούμενου επιπέδου και υπολογίζει μία συγκεκριμένη πιο περίπλοκη πληροφορία που υποδηλώνεται από τον συνδυασμό των χαρακτηριστικών που παίρνει ως είσοδο.

- **Επίπεδο Εξόδου:** Το τελικό επίπεδο του δικτύου μετασχηματίζει τα δεδομένα σε μία κατάλληλη μορφή ώστε να ταιριάζει η έξοδος του δικτύου με την απάντηση του προβλήματος το οποίο προσπαθεί να επιλύσει το δίκτυο. Για παράδειγμα, στα προβλήματα κατηγοριοποίησης, η έξοδος αποτελεί μία πιθανότητα για την ένταξη του αντικειμένου εισόδου σε κάθε κλάση κατηγοριοποίησης.



Εικόνα 3.2: Το δομικό στοιχείο Perceptron

3.1.1 Δομικό στοιχείο: Αισθητήρας

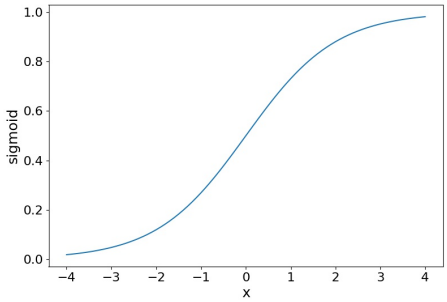
Κάθε ένας νευρώνας ενός απλού τροφοδοτικού δικτύου μπορεί να μοντελοποιηθεί ως ένα απλός Αισθητήρας (Perceptron) [28]. Η δομή ενός Perceptron παρουσιάζεται στην Εικόνα 3.2. Συγκεκριμένα, αυτό το μοντέλο παίρνει ως είσοδο ένα σύνολο από χαρακτηριστικά $x_1, x_2, ..., x_n$ και τους εφαρμόσει ένα γραμμικό μετασχηματισμό. Στη συνέεια, η έξοδος του γραμμικού μετασχηματισμού περνάει από μια βηματική συνάρτηση κατωφλίου, έστω θ, η οποία έχει έξοδο 0 ή 1. Η μαθηματική μοντελοποίηση της δομής του Perceptron δίνεται στην εξίσωση 3.1

$$y = \begin{cases} 1, & \text{if} \quad w^T \cdot x > \theta \\ 0, & \text{if} \quad w^T \cdot x \leq \theta \end{cases} \tag{3.1}$$

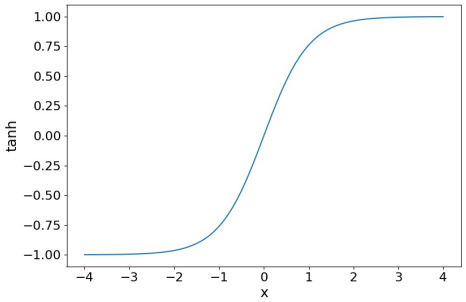
Όπως αναφέρθηκε ήδη, κάθε νευρώνας ενός απλού δικτύου ακολουθεί την παραπάνω μοντελοποίηση. Η διαφορά έχει να κάνει με την αντικατάσταση της βηματικής συνάρτησης με άλλες καταλληλότερες ανάλογα με την επιθυμητή λειτουργία του κάθε επίπεδου, οι οποίες ονομάζονται συναρτήσεις ενεργοποίησεις και θα συζητηθούν στην Ενότητα 3.1.2.

3.1.2 Συναρτήσεις Ενεργοποίησης

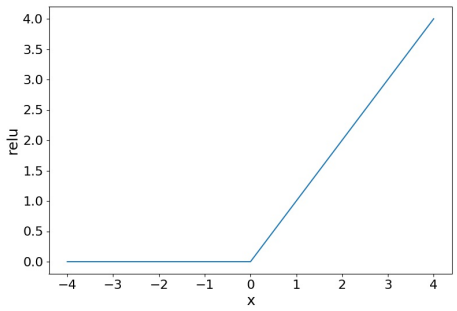
Τα απλά τροφοδοτικά νευρωνικά δίκτυα, όπως εξηγήθηκε στην Ενότητα 3.1.1, είναι ένα σύνολο από δομές Perceptron , με χρήση διαφορετικών **συναρτήσεων ενεργοποίησης** αντί της βηματικής που χρησιμοποιεί το απλό Perceptron. Για το επίπεδο εισόδου και τα κρυφά επίπεδα, υπάρχει ένα σύνολο διαφορετικών συναρτήσεων ενεργοποίησης, κάθε μία από τις οποίες παρουσιάζει διαφορετικές ιδιότητες. Οι συνηθέστερα χρησιμοποιούμενες συναρτήσεις ενεργοποίησης είναι η Σιγμοειδής (Sigmoid) [29], η υπερβολική εφαπτομένη (tanh) [30], και η μονάδα γραμμικού ανορθωτή (Rectified Linear Unit - ReLU) [31].



(α′) Σιγμοειδής



(β′) Υπερβολική Εφαπτομένη



(γ′) ReLU

Εικόνα 3.3: Βασικές Συναρτήσεις Ενεργοποίησης Κρυφών Επιπέδων

Στην Εικόνα 3.3 παρουσιάζονται οι γραφικές παραστάσεις των βασικών συναρτήσεων ενεργοποίησης για τα κρυφά επίπεδα και το επίπεδο εισόδου που αναφέρθηκαν παραπάνω. Η Σιγμοειδής συνάρτηση (3.3α′) έχει μία μορφή σαν τελικό σίγμα, από όπου έχει πάρει και το όνομα της. Πρόκειται για μία αύξουσα και παραγωγίσιμη παντού (σε αντίθεση με τη βηματική) συνάρτηση, με θετικές παραγώγους και ορισμένη σε όλο το σύνολο των πραγματικών αριθμών. Επιπλέον, εμφανίζει ιδιότητες ομαλότητας. Αποτελεί μία προσέγγιση της βηματικής συνάρτησης ενεργοποίησης καθώς μηδενίζεται για πολύ αρνητικές τιμές, ενώ παίρνει την τιμή 1 για πολύ θετικές τιμές. Μία βελτίωση της συγκεκριμένης συνάρτησης αποτελεί η υπερβολική εφαπτομένη (3.3β′). Πρόκειται για μία ικανοποιητικότερη προσέγγιση της βηματικής σε σχέση με τη σιγμοειδή συνάρτηση, ιδίως στην περιοχή γύρω από το 0, η οποία όμως αντιμετωπίζει το πρόβλημα των νεκρών νευρώνων σε ορισμένες περιπτώσεις. Η πιο συνηθισμένη συνάρτησης ενεργοποίησης είναι η ReLU (3.3γ′), η οποία δεν νεκρώνει τις ακραίες θετικές τιμές εισόδου της συνάρτησης. Η συνάρτηση αυτή ταυτίζεται με την γραμμική για θετικές τιμές εισόδου και επιπλέον διατηρεί μεγάλες τιμές παραγώγων όταν είναι ενεργή, οι οποίες είναι επιπλέον και συνεπείς. Η συγκεκριμένη συνάρτηση εμφανίζει πρόβλημα στην εκπαίδευσης μέσω αλγορίθμων της οικογένειας των καθόδου κλίσεων όταν η ενεργοποίηση της είναι μηδενική δηλαδή στις αρνητικές τιμές εισόδου. Για λόγους πληρότητας, παρουσιάζονται σύμφωνα με τους αναλυτικούς μαθηματικούς τύπους του

οι προαναφερθέντες συναρτήσεις ενεργοποίησης στον Πίνακα 3.1

Σιγμοειδής	$f(x) = \frac{1}{1+e^{-x}}$
Υπερβολική Εφαπτομένη	$f(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}}$
ReLU	$f(x) = max(0, x)$

Πίνακας 3.1: Μαθηματικοί Τύποι των βασικών συναρτήσεων ενεργοποίησης των κρυφών επιπέδων.

Σχετικά με το επίπεδο εξόδου, η συνάρτηση ενεργοποίησης, η οποία χρησιμοποιείται κατά κόρον στις περιπτώσεις προβλημάτων κατηγοριοποίησης είναι η συνάρτηση Softmax. Ο μαθηματικός τύπος αυτής της συνάρτησης δίνεται στην εξίσωση 3.2, θεωρώντας ως $x = [x_1, x_2, ..., x_k]$ το διάνυσμα που σχηματίζεται από το αποτέλεσμα του επιπέδου εξόδου.

$$\sigma(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{k} e^{x_j}} \tag{3.2}$$

Η συνάρτηση αυτή, όπως φανερώνει ο μαθηματικός τύπος 3.2, κανονικοποιεί τα αποτελέσματα του επιπέδου εξόδου, ώστε να προσομοιώνουν κατανομή πιθανότητας. Έτσι, κάθε στοιχείο του διανύσματος εξόδου εκφράζει την πιθανότητα το αντικείμενο εισόδου να ανήκει στην αντίστοιχη κλάση, δίνοντας την δυνατότητα να κατατάσσουμε το αντικείμενο εισόδου στην κατηγορία με την μεγαλύτερη πιθανότητα.

3.1.3 Μαθηματική Μοντελοποίηση

Στα προβλήματα κατηγοριοποίησης, ο στόχος ενός νευρωνικού δικτύου είναι να προβλέψει την κατηγορία $y \in \mathbb{N}$ στην οποία εντάσσεται ένα αντικείμενο $\vec{x} \in \mathbb{R}^n$ που χρησιμοποιείται ως είσοδος στο δίκτυο [26]. Επομένως, είναι απαραίτητος ο προσδιορισμός μιας προσέγγισης κάποιας συνάρτησης $f^* : \mathbb{R}^n \longrightarrow \mathbb{N}$, η οποία θα έχει με όσο μεγαλύτερη ακρίβεια θα ικανοποιεί τη σχέση $y = f^*(x)$ για διάφορα ζεύγη x, y. Το νευρωνικό δίκτυο ορίζει μία γενική συνάρτηση $f(\vec{x}; \vec{\theta})$, όπου $\vec{\theta}$ το σύνολο των παραμέτρων των νευρώνων του δικτύου (*βάρη του δικτύου*). Η εύρεση της συνάρτησης f^* μπορεί να προκύψει με επίλυση του προβλήματος ελαχιστοποίησης της συνάρτησης κόστους του προβλήματος κατηγοριοποίησης ως προς το σύνολο των παραμέτρων $\vec{\theta}$.

Κάθε επίπεδο αποτελείται από πολλούς νευρώνες, κάθε ένας από τους οποίους είναι ένα Perceptron με την επιλεγμένη συνάρτηση ενεργοποίησης, έστω $g^{(i)}$ για το i-οστό επίπεδο. Επομένως, αν $o^{(i)}$ η έξοδος του i-οστού επιπέδου και $W^{(i)}$ ο πίνακας των βαρών από όλους τους νευρώνες, τότε η συνάρτηση που μοντελοποιεί αυτό το επίπεδο $f^{(i)}$ δίνεται στην σχέση 3.3, θεωρώντας ως $\vec{x}$ την είσοδο του επιπέδου.

$$o^{(i)} = f^{(i)}(\vec{x}) = g^{(i)}(W^{(i)T} \cdot \vec{x}) \tag{3.3}$$

Επομένως, η συνάρτηση f του νευρωνικού δικτύου μπορεί να αναπαρασταθεί από την σύνθεση όλων των επιμέρους συναρτήσεων των διαδοχικών m επιπέδων του νευρωνικού δικτύου, όπως δίνεται στην εξίσωση , και το διάνυσμα $\vec{\theta}$ να περιλαμβάνει την ένωση όλων επιμέρους $W^{(i)}$.

$$f(\vec{x};\vec{\theta}) = f^{(m)}(f^{(m-1)}(...(f^{(2)}(f^{(1)}(\vec{x})))))) \tag{3.4}$$

3.1.4 Εκπαίδευση Νευρωνικών Δικτύων

Έχοντας ολοκληρώσει την περιγραφή της μοντελοποίησης και όλων των απαραίτητων εννοιών σχετικά με τα νευρωνικά δίκτυα, στην συγκεκριμένη ενότητα περιγράφεται η διαδικασία της εκπαίδευσης ενός δικτύου [26, 32]. Όπως αναφέρθηκε, κάθε πρόβλημα εκπαίδευσης είναι ένα πρόβλημα βελτιστοποίησης μιας συνάρτησης κόστους. Για τα προβλήματα κατηγοριοποίησης συγκεκριμένα, η συνηθέστερη συνάρτηση κόστους είναι η **κατηγορική εγκάρσια εντροπία (categorical cross-entropy) [33].** Γενικά, η μετρική της εγκάρσιας εντροπίας χρησιμοποιείται ανάμεσα σε δύο κατανομές πιθανότητας για να μετρήσει αν το σύνολο της προσεγγιστικής κατανομής είναι βελτιστοποιημένο σε σχέση με την πραγματική κατανομή. Στα προβλήματα κατηγοριοποίησης συγκεκριμένα χρησιμοποιείται για να μετρήσει την απόκλιση της κατανομής πιθανότητας που προκύπτει από τη συνάρτηση Softmax του επιπέδου εξόδου, ως προς την πραγματική κατηγορία στην οποία ανήκει ένα αντικείμενο. Η μετρική εκφράζεται μέσω της συνάρτησης 3.5, όπου f η συνάρτηση του δικτύου και m το πλήθος των κλάσεων του προβλήματος.

$$L(\vec{\theta_i};\vec{x}_j, y_j) = -\sum_{i=1}^{m} y_k \cdot \log\left(f(\vec{x}_i;\vec{\theta_i})\right)_k \tag{3.5}$$

Έχοντας ορίσει τη παραπάνω συνάρτηση κόστους είναι εύκολο να συνοψιστούν τα βήματα του αλγορίθμου εκπαίδευσης του νευρωνικού δικτύου. Η διαδικασία συνοψίζεται ως εξής για κάθε επανάληψη:

- Ευθύ πέρασμα του δικτύου για την μικρο-ομάδα εισόδου και εύρεση κατανομής εξόδου για κάθε στοιχείο της μικρο-ομάδας.
- Υπολογισμός συνάρτησης κόστους για το σύνολο των κατανομών εξόδου ως προς τις πραγματικές κατηγορίες.
- Εφαρμογή του αλγόριθμου οπισθοδρόμησης για υπολογισμό και εφαρμογή του στοχαστικού αλγορίθμου κλίσεων για μικρο-ομάδες (ή κάποια παραλλαγή του). Οι κλίσεις εφαρμόζεται από το τελευταίο επίπεδο προς το πρώτο μέσω του κανόνα της αλυσίδας.

Η παραπάνω διαδικασία επαναλαμβάνεται για συγκεκριμένο πλήθος εποχών πάνω στο σύνολο δεδομένων που καθορίζει ο χρήστης. Εναλλακτικά, χρησιμοποιούνται τεχνικές πρόωρου τερματισμού (earlu stopping) [34].

3.2 Συνελικτικά Τροφοδοτικά Νευρωνικά Δίκτυα

Στην παρούσα ενότητα γίνεται μία συνοπτική βιβλιογραφική ανασκόπηση σχετικά με την τεχνολογία των συνελικτικών νευρωνικών δικτύων [1, 35–40]. Η συγκεκριμένη κατηγορία τροφοδοτικών δικτύων χρησιμοποιείται ευρύτατα σε περιπτώσεις που χρησιμοποιείται οπτικό υλικό. Σε αντιδιαστολή με τα απλά τροφοδοτικά δίκτυα που παρουσιάστηκαν στην Ενότητα 3.1, τα δίκτυα αυτά έχουν κρυφά επίπεδα που η έξοδος τους περιγράφεται από πιο περίπλοκες συναρτήσεις και ως συνάρτηση ενεργοποίησης χρησιμοποιούν κατά βάση την ReLU. Οι βασικές κατηγορίες κρυφών επιπέδων που χρησιμοποιούνται στα συνελικτικά δίκτυα είναι:

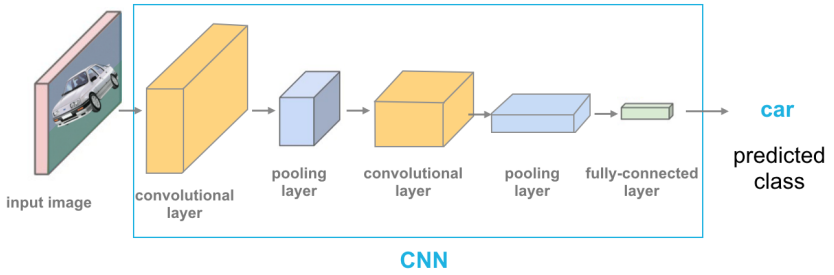
- **Συνελικτικό (Convolutional):** Σε αυτό το επίπεδο εφαρμόζεται ένας συνελικτικός τελεστής στην εικόνα εισόδου για την παραγωγή της εικόνας εξόδου. Επομένως, πέρασμα από ένα τέτοιο επίπεδο αντιστοιχεί σε εφαρμογή φίλτρου στην εικόνα. Υπερπαράμετροι του επιπέδου αφορούν το μέγεθος του συνελικτικού πυρήνα, το πλήθος των καναλιών εισόδου και εξόδου. Η αντικατάσταση των απλών πλήρως συνδεδεμένων δικτύων με αυτά συνεπάγεται μεγαλύτερο βάθος επεξεργασίας στις εικόνες, με σημαντικά μικρότερο μέγεθος παραμέτρων, αφού στα επίπεδα αυτά οι μόνες παράμετροι προς εκπαίδευση αντιστοιχούν στον συνελικτικό πυρήνα που εφαρμόζεται στην εικόνα εισόδου.

- **Χωρικής Υπο-δειγματοληψίας (Pooling):** Τα συνελικτικά δίκτυα ενδέχεται να περιέχουν τοπικά η ολικά επίπεδα που πραγματοποιούν δειγματοληψία για να μετατρέψουν σε ροή τους υπολογισμούς. Μειώνουν τις διαστάσεις των δεδομένων συνδυάζοντας τις εξόδους από ομάδες νευρώνων σε έναν μοναδικό νευρώνα στο επόμενο επίπεδο. Στην τοπική χωρική υπο-δειγματοληψία χρησιμοποιούνται ομάδες μικρού μεγέθους, με σύνηθες το 2×2. Για την ολική δειγματοληψία χρησιμοποιούνται όλοι οι νευρώνες ενός επιπέδου. Από τους διαθέσιμους νευρώνες χρησιμοποιείται συνήθως είτε ο μέσος όρος είτε το μέγιστο για το αποτέλεσμα της δειγματοληπτούμενης τιμής.

- **Κανονικοποίησης (Normalization):** Χρησιμοποιείται για να κανονικοποιήσει το σύνολο των οπτικών πληροφοριών που παίρνει ως είσοδο χρησιμοποιώντας τη μέση τιμή και τη διασπορά του διαθέσιμου υλικού. Έχει την ικανότητα να σταθεροποιεί τη διαδικασία εκπαίδευσης και να μειώνει δραματικά τον αριθμό των εποχών που απαιτούνται για την εκπαίδευση του νευρωνικού δικτύου.

- **Πλήρως Συνδεδεμένο (Fully Connected):** Είναι αντίστοιχα με τα επίπεδα ενός απλού τροφοδοτικού νευρωνικού δικτύου. Τέτοιας κατηγορίας είναι και το επίπεδο εξόδου του συνελικτικού νευρωνικού δικτύου έχοντας ως συνάρτηση εξόδου την Softmax.

- **Απόσυρσης (Dropout):** Τα συνελικτικά δίκτυα είναι ευαίσθητα στο φαινόμενο της υπερπροσαρμογής στα δεδομένα εκπαίδευσης (overfitting). Για την αποφυγή του φαινόμενου αυτού είναι σύνηθες να χρησιμοποιούνται επίπεδα απόσυρσης, όπου συγκεκριμένοι νευρώνες αφήνονται εκτός από την αλληλουχία των υπολογισμών που εισάγει το δίκτυο. Έτσι μόνο μέρος της πληροφορίας χρησιμοποιείται μειώνοντας τα φαινόμενα υπερπροσαρμογής. Η διαδικασίας επίλογης ή όχι του κάθε νευρώνα θεωρείται ένα πείραμα τύχης.



Εικόνα 3.4: Τυπικό παράδειγμα συνελικτικού τροφοδοτικού νευρωνικού δικτύου.

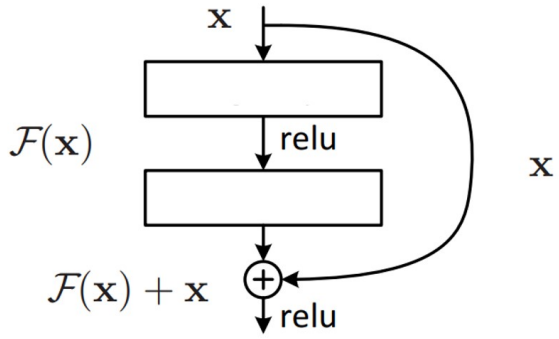
Ένα τυπικό παράδειγμα συνελικτικού δικτύου δίνεται στην Εικόνα 3.4 [41]. Το δίκτυο που δίνεται ως παράδειγμα αποτελείται από μια αλληλουχία συνελικτικών επιπέδων και επιπέδων χωρικής υπο-δειγματοληψίας, ενώ στο τέλος με τη βοήθεια πλήρως συνδεδεμένων επιπέδων ταξινομεί την εικόνα εισόδου. Τα πρώτα επίπεδα φιλτράρουν και μείωνουν τις διαστάσεις του οπτικού υλικού, ενώ τα τελευταία αναλύουν την τελική πληροφορία για να κατηγοριοποιήσουν τελικά την εικόνα εισόδου.

Στις ενότητες 3.2.1 και 3.2.2 παρουσιάζονται δύο από τις σημαντικότερες αρχιτεκτονικές συνελικτικών δικτύων οι οποίες θα χρησιμοποιηθούν στο πειραματικό μέρος της εργασίας.

3.2.1 Η αρχιτεκτονική ResNet

Τα υπολειπόμενα νευρωνικά δίκτυα (ResNet) [3] στηρίζονται στη λογική προσέγγισης περίπλοκων συναρτήσεων μέσω της τεχνικής των υπολοίπων. Έστω $\mathcal{H}(x)$ η περίπλοκη συνάρτηση που θεωρητικά προσομοιώνει κάποιο επίπεδο και $\mathcal{F}(x)$ η προσέγγιση η οποία δίνει το επίπεδο. Τότε σύμφωνα με την θεωρία των υπολοίπων μπορεί να γίνει η παραδοχή της εξίσωσης 3.6.

$$\mathcal{H}(x) = \mathcal{F}(x) + x \tag{3.6}$$



Εικόνα 3.5: Το δομικό στοιχείο ενός δικτύου ResNet.

Η εξίσωση 3.6 προσομοιώνεται σε ένα τμήμα νευρωνικού δικτύου σύμφωνα με την Εικόνα 3.5 [3]. Τα λευκά τετραγωνικά κουτιά αποτελούν συνήθως συνελικτικά δίκτυα. Ανάλογα το πλήθος των δομικών στοιχείων που εμφανίζονται υπάρχουν διαφορετικά δίκτυα της αρχιτεκτονικής. Τα δίκτυα αυτά διαφέρουν ως προς το μέγεθος των χαρακτηριστικών που εξάγουν τα συνελικτικά δίκτυα και της αλληλουχίας των επίπεδων που ορίζουν την συνάρτηση $\mathcal{F}$ της εξίσωσης 3.6. Αξίζει επιπλέον να αναφερθεί ότι τα δίκτυα δεν εισάγουν πολυπλοκότητα στην εκπαίδευση με τη χρήση των υπολοίπων, σε σχέση με τα αντίστοιχα που δεν αξιοποιούν τα υπόλοιπα, καθώς αυτά δεν εισάγουν επιπλέον παραμέτρους εκπαίδευσης στο δίκτυο.

3.2.2 Η αρχιτεκτονική Inception-v3

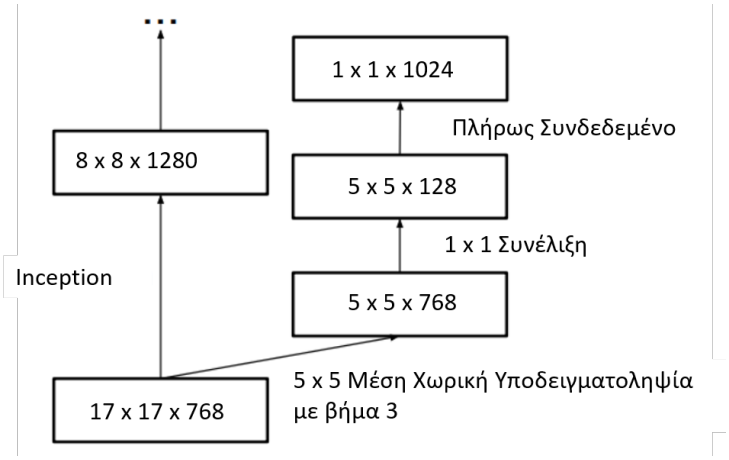
Η αρχιτεκτονική Inception-v3 [42] ανήκει στην οικογένεια των συνελικτικών νευρωνικών δικτύων. Στην αρχιτεκτονική αυτή χρησιμοποιούνται διάφορες βελτιώσεις σε σχέση με ένα απλό συνελικτικό δίκτυο, όπως η εξομάλυνση ετικετών, παραγοντοποιημένες 7×7 συνελίξεις και η χρήση βοηθητικού εσωτερικού ταξινομητή για τη διάδοση πληροφορίας σχετικά με τις ετικέτες σε χαμηλότερα επίπεδα του δικτύου. Στην συνέχεια, γίνεται μια ανασκόπηση των τεχνικών που υλοποιεί την παρούσα αρχιτεκτονική.

- **Παραγοντοποιημένες Συνελίξεις:** Η τεχνική μειώνει την υπολογιστική πολυπλοκότητα του δικτύου, καθώς μειώνει το πλήθος των παραμέτρων του δικτύου που χρησιμοποιούνται.

- **Μικρότερες Συνελίξεις:** Η παραπάνω αντικατάσταση οδηγεί σε γρηγορότερη εκπαίδευση καθώς το πλήθος των παραμέτρων μειώνεται.

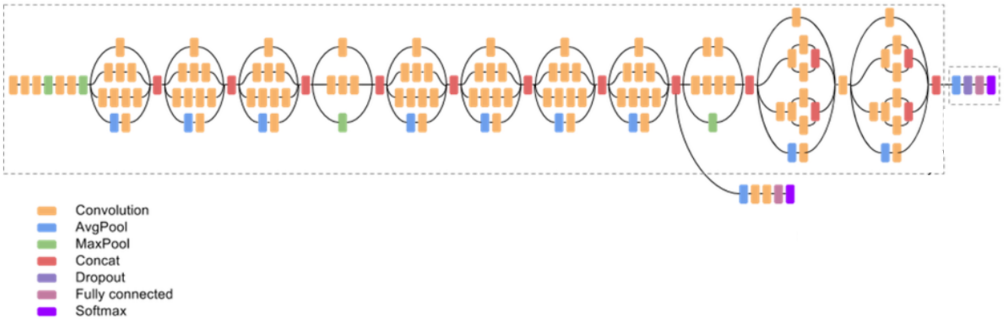
- **Ασσύμετρες Συνελίξεις:** Μία $n \times n$ συνέλιξη είναι ισοδύναμη με μία $1 \times n$ συνέλιξη ακολουθούμενη με μία $n \times 1$ συνέλιξη. Όσο μεγαλύτερη είναι η τιμή του n τόσο μεγαλύτερη είναι η μείωση του κόστους που συνεπάγεται αυτή η αντικατάσταση.

- **Βοηθητικός Ταξινομητής:** Ο βοηθητικός ταξινομητής είναι ένα μικρό συνελικτικό δίκτυο, όπως απεικονίζεται στην Εικόνα 3.6, το οποίο τοποθετείται ανάμεσα σε επίπεδα του Inception κατά το στάδιο της εκπαίδευσης. Το σφάλμα από αυτό το μικρό δίκτυο συμπεριλαμβάνεται στο συνολικό σφάλμα που προκύπτει από το δίκτυο. Πρακτικά στη συγκεκριμένη αρχιτεκτονική ο βοηθητικός ταξινομητής έχει το ρόλο του κανονικοποιητή.



Εικόνα 3.6: Βοηθητικός ταξινομητής στο δίκτυο Inception.

Συνοψίζοντας όλα τα παραπάνω, η τελική αρχιτεκτονική του δικτύου δίνεται στην Εικόνα 3.7.



Εικόνα 3.7: Η αρχιτεκτονικη Inception-v3.

Κεφάλαιο 4

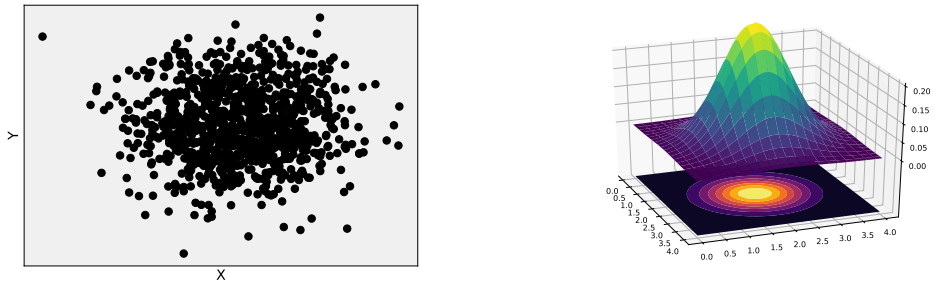
Δεδομένα: Κατανομές και σχετικοί Αλγόριθμοι

Για την εκπαίδευση ενός νευρωνικού δικτύου χρησιμοποιείται ένα σύνολο δεδομένων, το οποίο ονομάζεται *σύνολο εκπαίδευσης*. Κάθε σύνολο δεδομένων ορίζει μια κατανομή δεδομένων στο χώρο. Η κατανομή παρέχει μία παραμετροποιημένη μοντελοποίηση με χρήση συναρτήσεων για την περιγραφή της πυκνότητας των δεδομένων σε κάθε σημείο του χώρου στον οποίο ορίζονται [43]. Στο συγκεκριμένο κεφάλαιο, παρατίθεται μία βιβλιογραφική ανασκόπηση σχετικά με τις κατανομές των δεδομένων (ενότητα 4.1). Επιπλέον παρουσιάζονται δύο βασικοί αλγόριθμοι για τη μείωση των διαστάσεων του χώρου (ενότητα 4.2) που ορίζονται τα δεδομένα, καθώς και της ομαδοποίησης (ενότητα 4.3) αυτών σε ομάδες που εντοπίζονται εντός της συνολικής κατανομής.

4.1 Μοντελοποίηση με χρήση της κανονικής κατανομής

Για σύνολα δεδομένων που ορίζονται με χρήση χαρακτηριστικών που έχουν συνεχή τιμές σε κάποιο διάστημα, μία από τις πιο συνηθισμένες κατανομές που εντοπίζονται είναι η *κανονική κατανομή* [44]. Η συγκεκριμένη κατανομή είναι συμμετρική γύρω από τη μέση τιμή, γεγονός που σημαίνει ότι είναι πιο συχνό να εντοπίσουμε δεδομένα στο χώρο γύρω από τη μέση τιμή, ενώ η πιθανότητα εμφάνισης μειώνονται καθώς απομακρυνόμαστε από αυτή. Σε ένα n-διάστατο χώρο, η συνάρτηση πυκνότητας πιθανότητας, δίνεται από την εξίσωση 4.1, $\vec{\mu}$ είναι το διάνυσμα της μέσης τιμής και Σ ο πίνακας συνδιακύμανσης των δεδομένων. Εάν ένα διάνυσμα $\vec{x}$ ανήκει σε αυτήν την κατανομή, συμβολίζεται ως $\vec{x} \sim \mathcal{N}_n(\vec{\mu}, \Sigma)$.

$$f(\vec{x}; \vec{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^n \cdot \|\Sigma\|}} \exp \frac{1}{2}(\vec{x} - \vec{\mu})^T \Sigma^{-1} (\vec{x} - \vec{\mu}) \tag{4.1}$$



(α′) Δεδομένα κανονικής κατανομής (β′) Απεικόνιση σ.π.π.

Εικόνα 4.1: Παράδειγμα δεδομένων κανονικής κατανομής στο 2-Δ χώρο

Στην Εικόνα 4.1 δίνεται ένα παράδειγμα δεδομένων που προέρχονται από μία δισδιάστατη κανονική κατανομή με μέση τιμή το διάνυσμα $\vec{\mu} = \begin{bmatrix} 2 & 2 \end{bmatrix}$ και πίνακα συνδιακύμανσης $\Sigma = \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix}$. Γίνεται φανερό ότι η πλειοψηφία των δεδομένων εκτείνεται γύρω από το διάνυσμα της μέσης τιμής (4.1α′), γεγονός που επιβεβαιώνεται και από την γραφική παράσταση της συνάρτησης πυκνότητας πιθανότητας (4.1β′). Το ύψος της τρισδιάστασης αυτής καμπύλης δίνει την πιθανότητα εμφάνισης του κάθε σημείου, ενώ η προβολή του στο επίπεδο 2-Δ ορίζει ότι όσο πιο θερμό είναι το χρώμα μιας περιοχής τόσο πιο πιθανό είναι να εντοπιστούν εκεί σημεία της δεδομένης κατανομής.

Σύμφωνα με όσα περιγράφηκαν παραπάνω, η κανονική κατανομή μπορεί να χρησιμοποιηθεί για να περιγράψει σύνολα δεδομένων τα οποία εκτείνονται γύρω από μία μέση τιμή, με μεγαλύτερη πυκνότητα κοντά σε αυτήν. Ωστόσο, στη γενική περίπτωση πραγματικά σύνολα δεδομένων δεν ακολουθούν συνήθως αυτήν την δομή στο πολυδιάστατο χώρο. Ωστόσο, ένας τρόπος να περιγράψουμε ένα οποιοδήποτε σύνολο δεδομένων είναι με χρήση μίας σύνθεσης από κανονικές κατανομές. Συγκεκριμένα, μπορούμε να θεωρήσουμε ότι τα δεδομένα του χώρου χωρίζονται σε υποσύνολα κάθε ένα από τα οποία προέρχεται από κάποια κανονική κατανομή. Ένα τέτοιο μοντέλο ικανό να περιγράψει οποιαδήποτε κατανομή ονομάζεται *μοντέλο κανονικής σύνθεσης* [45, 46] και ο μαθηματικός ορισμός του παρατίθεται στον Ορισμό 3.

Ορισμός 3 (Μοντέλο Κανονικής Σύνθεσης - Gaussian Mixture Model) *Το Μοντέλο Κανονικής Σύνθεσης ορίζει μια κατανομή με συνάρτηση πυκνότητας πιθανότητας $P(x)$ ως σύνθεση n κανονικών κατανομών με διανύσματα μέσης τιμής $\vec{\mu}_i$ και πίνακες συνδιακύμανσης Σ_i , $i \in \{1, 2, ..., n\}$ σύμφωνα με τη σχέση*

$$P(x) = \sum_{i=1}^{n} \pi_i \cdot \mathcal{N}_i(\vec{\mu}_i, \Sigma_i)$$

όπου $\pi_i \geq 0$ και $\sum_{i=1}^{n} \pi_i = 1$.

Η αύξηση του πλήθους των κανονικών κατανομών που χρησιμοποιείται για να οριστεί το μοντέλο σύνθεσης μπορεί να οδηγήσει στην αναπαράσταση οποιασδήποτε κατανομής. καθώς υποδιαιρείται ο χώρος σε όλο και μικρότερα τμήματα καθένα από τα οποία μπορεί να προσεγγιστεί επαρκώς από μία κανονική κατανομή. Οι παράμετροι π_i εκφράζουν την πιθανότητα κάποιο δεδομένο του χώρου να προέρχεται από την i-οστή κατανομή της σύνθεσης. Για την εύρεση των κατάλληλων παραμέτρων για το μοντέλο κανονικής σύνθεσης χρησιμοποιείται συνήθως ο αλγόριθμος Μεγιστοποίησης Αναμονής (Expectation Maximization - EM) [47]. Ωστόσο, ο αλγόριθμος αυτός είναι αρκετά αργός και μια απλουστευμένη ταχύτερη και προσεγγιστική εκδοχή του αποτελεί ο αλγόριθμος ομαδοποίησης Κ συστάδων, ο οποίος αναλύεται στην ενότητα 4.3.

4.2 Ο αλγόριθμος Ανάλυσης Κυρίαρχων Συνιστωσών

Για την καλύτερη κατανόηση του αλγορίθμου είναι χρήσιμο να παρουσιαστεί η έννοια των κυρίαρχων συνιστωσών, η οποία δίνεται στον Ορισμό 4.

Ορισμός 4 (Κυρίαρχες Συνιστώσες) *Ως κυρίαρχες συνιστώσες ενός συνόλου σημείων σε έναν πραγματικό χώρο n διαστάσεων ορίζεται ένα σύνολο n διανυσμάτων κατεύθυνσης, όπου το i-οστό διάνυσμα πληρεί τις ακόλουθες δύο προϋποθέσεις:*

1. *έχει την κατεύθυνση της γραμμής που προσαρμόζει καλύτερα πάνω στα δεδομένα (η γραμμή που ελαχιστοποιεί τη μέση τετραγωνική απόσταση των σημείων από αυτή)*

2. *είναι ορθογωνικό ως προς τα προηγούμενα $i-1$ σημεία*

Τα διανύσματα αυτά σχηματίζουν μία ορθοκανονική βάση στην οποία οι διάφορες διαστάσεις των δεδομένων είναι γραμμικά ασυσχέτιστες μεταξύ τους.

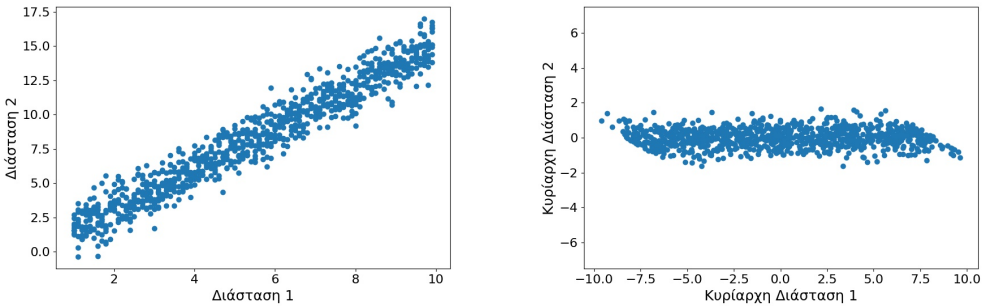
Ο αλγόριθμος Ανάλυσης Κυρίαρχων Συνιστωσών (Principal Component Analysis - PCA) [48, 49] υπολογίζει τις κυρίαρχες συνιστώσες και τις χρησιμοποιεί για να αλλάξει την βάση ως προς την οποία εκφράζονται τα δεδομένα. Σε πολλές περιπτώσεις συνηθίζεται να χρησιμοποιεί μόνο κάποιες από τις πρώτες κυρίαρχες συνιστώσες αγνοώντας τις υπόλοιπες. Διατηρώντας μόνο ένα μέρος με τις σημαντικότερες κυρίαρχες συνιστώσες, ο αλγόριθμος χρησιμοποιείται για να μειώσει τις διαστάσεις ενός χώρου προβάλλοντας κάθε διανυσματικό σημείο τους πάνω στις σημαντικές κυρίαρχες συνιστώσες που έχουν επιλεχθεί. Έτσι προκύπτει ένας χώρος λιγότερων διαστάσεων, διατηρώντας παράλληλα τη περισσότερη δυνατή ποικιλομορφία που διακρίνει το σύνολο των σημείων. Συγκεκριμένα, οι πρώτες κυρίαρχες συνιστώσες μπορούν ισοδύναμα να οριστούν ως η κατεύθυνση εκείνη που μεγιστοποιεί τη διασπορά των προβαλλόμενων δεδομένων. Έτσι η i-οστή κυρίαρχη συνιστώσα μπορεί να επιλεχθεί ώστε να είναι ορθογώνια στις προηγούμενες $i-1$ και ταυτόχρονα να μεγιστοποιείται η διασπορά των

προβαλλόμενων δεδομένων. Για την επίτευξη όλων των παραπάνω μπορεί να αποδειχθεί ότι οι κυρίαρχες συνιστώσες μπορούν να υπολογιστούν ως τα ιδιοδιανύσματα του πίνακα συνδιακύμανσης των αρχικών δεδομένων. Ο αλγόριθμος PCAπαρουσιάζεται συνοπτικά στον Αλγόριθμο 2.

Αλγόριθμος 2 Ο αλγόριθμος PCA

procedure PCA(X, θ)
  ▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων
  ▷ θ παραμέτρος καθορισμού επιλογής σημαντικών συνιστωσών
  $\vec{\mu} \leftarrow \frac{1}{n} \sum_{i=1}^{n} \vec{x}_i$ ▷ $\vec{\mu}$ η μέση τιμή των δεδομένων
  ▷ C ο πίνακας συνδιακύμανσης των δεδομένων
  $C \leftarrow \frac{1}{n}\{\vec{x}_i - \vec{\mu}\} \cdot \{\vec{x}_i - \vec{\mu}\}^T$
  ▷ q το πλήθος των χαρακτηριστικών στα δεδομένα
  Εύρεση ιδιοτιμών λ_i και ιδιοδιανυσμάτων v_i, $i \in \{1, 2, ..., q\}$
  Ταξινόμηση των ιδιοτιμών λ_i σε φθίνουσα σειρά
  Επιλογή των s πρώτων ιδιοτιμών, ώστε: $(\sum_{i=1}^{s} \lambda_i) \cdot (\sum_{i=1}^{q} \lambda_i)^{-1} \geq \theta$
  Διατήρηση των s ιδιοδιανυσμάτων που αντιστοιχούν στις s ιδιοτιμές
  Σχηματισμός πίνακα V από τα s ιδιοδιανύσματα
  $P \leftarrow V^T \cdot X$ ▷ P ένας χώρος με $s < q$ διαστάσεις
  Επέστρεψε P
end procedure

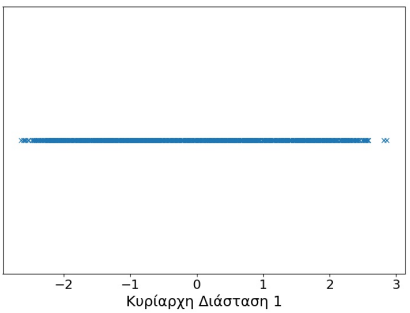
Σύμφωνα με τον αλγόριθμο 2 ο τρόπος υπολογισμού του μειωμένου χώρου διαστάσεων είναι αρκετά εύκολος αλγοριθμικά. Έχοντας υπολογίσει τον πίνακα συνδιακύμανσης, γίνεται εύρεση των ιδιοτιμών και ιδιοδιανσμάτων του. Ταξινομώντας τις ιδιοτιμές σε φθίνουσα σειρά προκύπτει το σύνολο των ιδιοδιανυσμάτων ταξινομημένο με σειρά σημαντικότητας. Έτσι πρέπει να διατηρηθούν οι πρώτες s συνιστώσες, που αντιστοιχούν στα ιδιοδιανύσματα των πρώτων s μεγαλύτερων ιδιοτιμών, ώστε η συνολική ενέργεια των επιλεγόμενων ιδιοτιμών να είναι τουλάχιστον το ζητούμενο ποσοστό θ της συνολικής ενέργειας όλων των ιδιοτιμών. Όσο μεγαλύτερο είναι το κατώφλι θ που χρησιμοποιείται, τότε περισσότερες συνιστώσες θα επιλεγούν. Στην συνέχεια σχηματίζεται ο πίνακας V των s επιλεγμένων διανυσμάτων που δίνουν την νέα ορθοκανονική βάση, ώστε να προκύψει το σύνολο P που περιέχει την προβολή των αρχικών σημείων στον διανυσματικό χώρο της νέας βάσης. Η υπολογιστική πολυπλοκότητα του αλγορίθμου είναι $\mathcal{O}(q^2 \cdot n + q^3)$.



(α′) Αρχικό 2-Δ Σύνολο (β′) Προβαλλόμενο 2-Δ σύνολο

Εικόνα 4.2: 2-Δ Σύνολο δεδομένων πριν και μετά την εφαρμογή του αλγορίθμου PCA.

Στην Εικόνα 4.2 παρουσιάζεται ένα σύνολο 2-Δ σύνολο δεδομένων στην αρχική του μορφή (Εικόνα 4.2α′) και αφού έχει μετασχηματιστεί μέσω του αλγορίθμου PCA(Εικόνα 4.2β′). Το συγκεκριμένο παράδειγμα δείχνει ότι το αποτέλεσμα από την εφαρμογή του αλγορίθμου είναι μια περιστροφή των δεδομένων κατά τέτοιο τρόπο ώστε τα δεδομένα να παρουσιάζουν την μέγιστη διασπορά σε αυτήν. Πρακτικά, η Κυρίαρχη Συνιστώσα 1 είναι η διαγώνια ευθεία που θα παρέμβαλε τα δεδομένα της Εικόνας 4.2α′. Επομένως, σε περίπτωση που είναι επιθυμητή η μείωση της διάστασης του χώρου, η αφαίρεση της κυρίαρχης συνιστώσας 2 θα οδηγούσε σε μικρότερη απώλεια πληροφορίας σε σχέση με την 1. Το αποτέλεσμα μιας τέτοιας διαδικασίας φαίνεται στην Εικόνα 4.3, όπου τα μονοδιάστατα πλέον δεδομένα εκτείνονται κατά μήκους τους άξονα της Κυριάρχης Συνιστώσας 1.



Εικόνα 4.3: Το σύνολο δεδομένων της Εικόνας 4.2α′ προβαλλόμενο στην πρώτη κυρίαρχη συνιστώσα.

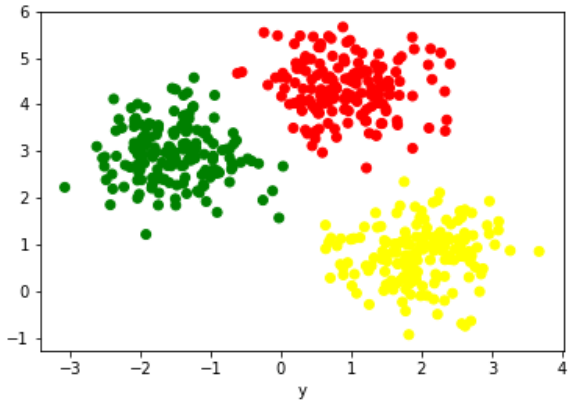
4.3 Ο αλγόριθμος ομαδοποίησης Κ - συστάδων

Ο αλγόριθμος ομαδοποίησης Κ - συστάδων (KMeans) [50–52]είναι μία μέθοδος που σκοπεύει να διαμοιράσει ένα σύνολο n διανυσματικών σημείων σε k συστάδες. Κάθε παρατήρηση ανήκει στη συστάδα με το κοντινότερο μέσο (κεντρικό σημείο της κλάσης) το οποίο θεωρείται ο αντιπρόσωπος της δεδομένης συστάδας. Η συγκεκριμένη μέθοδος χρησιμοποιείται ευρέως σε ανάλυση συστάδων στην περιοχή της εξόρυξης γνώσης από δεδομένα. Η μέθοδος έχει τη δυνατότητα να ελαχιστοποιεί την διασπορά εντός συστάδας, θεωρώντας ως διασπορά τις τετραγωνικές ευκλείδιες αποστάσεις. Ο αλγόριθμος KMeans δίνεται στον Αλγόριθμο 3.

Αλγόριθμος 3 Ο αλγόριθμος KMeans

1: **procedure** KMeans(X, k)
2: ▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων
3: ▷ k το πλήθος των ομάδων διαχωρισμού
4: $(\vec{c}_1, \vec{c}_2, ..., \vec{c}_k) \leftarrow$ αρχικοποίηση()
5: **while** κριτήριο **do**
6: **for** $i \in \{1, 2, ..., k\}$ **do**
7: $\text{group}_i \leftarrow \emptyset$
8: **end for**
9: **for** $i \in \{1, 2, ..., n\}$ **do**
10: $j \leftarrow \arg\min_{m \in \{1,2,...,k\}} \|\vec{x}_i - \vec{c}_m\|$
11: $\text{group}_j \leftarrow \text{group}_j \cup \{\vec{x}_i\}$
12: **end for**
13: **for** $i \in \{1, 2, ..., k\}$ **do**
14: $\vec{c}_i \leftarrow \frac{1}{\| \text{group}_i \|} \sum_{\vec{x}_m \in \text{group}_i} \vec{x}_m$
15: **end for**
16: **end while**
17: **end procedure**

Στην Γραμμή 4 του Αλγορίθμου 3 αναφέρεται ότι γίνεται μια αρχικοποίηση των κεντρικών διανυσματικών σημείων που αντιπροσωπεύουν κάθε μία από τις k συστάδες. Στη συνέχεια, μέχρι την σύγκλιση του αλγορίθμου, βρίσκουμε για κάθε στοιχείο του συνόλου δεδομένων το κοντινότερο από τα κεντρικά σημεία και αναθέτουμε το στοιχείο στην αντίστοιχη συστάδα (Γραμμές 9-12). Το κεντρικό σημείο μίας συστάδας ενημερώνονται ως το μέσο διάνυσμα από τα στοιχεία που τοποθετήθηκαν στην αντίστοιχη συστάδα. Ένα παράδειγμα εκτέλεσης του αλγορίθμου KMeans δίνεται στην Εικόνα 4.4 [53].



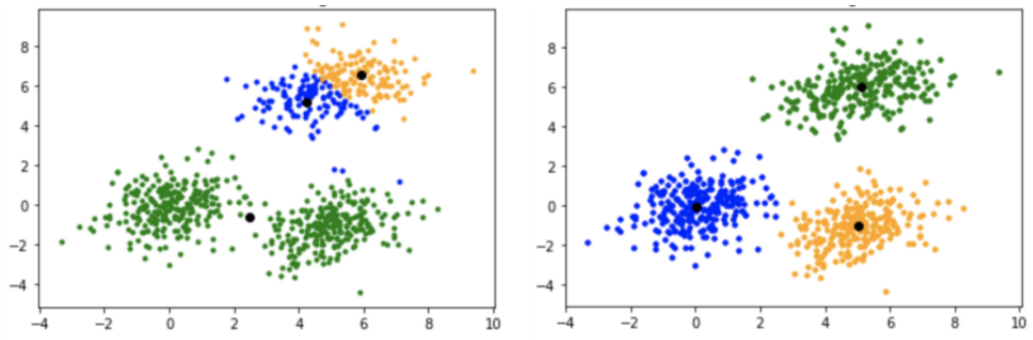
Εικόνα 4.4: Ένα παράδειγμα δημιουργίας τριών συστάδων μέσω του KMeans.

Η πιο συνηθισμένη τεχνική αρχικοποίησης αφορά την τυχαία επιλογή k σημείων από το σύνολο των n δεδομένων τα οποία χρησιμοποιούνται ως αρχικά κεντρικά σημεία των συστάδων. Μία άλλη τεχνική αρχικοποίησης που χρησιμοποιείται ευρέως είναι η διαδικασία Kmeans++ [54].Ο αλγόριθμος αρχικοποίησης δίνεται στον Αλγόριθμο 4.

Αλγόριθμος 4 Ο αλγόριθμος KMeans++

procedure KMeans++(X, k)
▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων
▷ k το πλήθος των ομάδων διαχωρισμού
    $centroids \leftarrow \emptyset$
    $\vec{c} \leftarrow$ επιλογή_τυχαίου_σημείου_από(X)
    $centroids \leftarrow centroids \cup \{\vec{c}\}$
    repeat
        for $\vec{x} \in X$ **do**
            $j \leftarrow \arg\min_{\vec{c} \in centroids} \|\vec{x} - \vec{c}\|^2$
            $d(\vec{x}) \leftarrow \|\vec{x} - \vec{c_j}\|^2$
        end for
        $choose\_probs \leftarrow \emptyset$
        for $\vec{x} \in X - centroids$ **do**
            $choose\_prob \leftarrow \frac{d(\vec{x})}{\sum_{\vec{x} \in X} d(\vec{x})}$
            $choose\_probs \leftarrow probs \cup \{choose\_prob\}$
        end for
        $\vec{c} \leftarrow$ τυχαία_επιλογή$(X - centroids, choose_probs)$
        $centroids \leftarrow centroids \cup \{\vec{c}\}$
    until έχουν επιλεγεί k σημεία
end procedure

Ο KMeans++ (Αλγόριθμος 4) εγγυάται μία εξυπνότερη επιλογή αρχικών σημείων ως κεντρικά των συστάδων. Ακολουθώντας αυτή τη διαδικασία γίνεται επιλογή αρχικών κεντρικών σημείων τα οποία είναι μακριά το ένα από το άλλο. Κατά αυτόν τον τρόπο αυξάνεται η πιθανότητα να επιλεγούν σημεία τα οποία ανήκουν όντως σε διαφορετικές συστάδες. Αυτό επιτυγχάνεται θεωρώντας ως πιθανότητα επιλογής ενός σημείου ανάλογη με την απόσταση του από το κοντινότερο από τα ήδη προηγουμένως επιλεγμένα κεντρικά σημεία. Έτσι πιο μακρινό σημείο έχει μεγαλύτερη πιθανότητα επιλογής σε σχέση με τα πιο κοντινά.



(α′) Κακή επιλογή αρχικών (β′) Καλή επιλογή αρχικών

Εικόνα 4.5: Εφαρμογή KMeans με καλή και κακή επιλογή αρχικών κεντρικών διανυσματικών σημείων

Για να γίνει κατανοητή η σημασία επιλογής κατάλληλων αρχικών κέντρων μέσω του, στην Εικόνα 4.5 [53] δίνεται ένα παράδειγμα εκτέλεσης του KMeans με κακή τυχαία επιλογή αρχικών σημείων (4.5α′) και με επιλογή αρχικών σημείων μέσω του KMeans++ (4.5β′). Ενώ τα δεδομένα σχηματιζούν τρεις διακρίσιμες ομάδες, στην κακή επιλογή των αρχικών κεντρικών των συστάδων ο αλγόριθμος συγκλίνει χωρίζοντας τα με λάθος τρόπο αφού οι 2 ομάδες έχουν ενοποιηθεί και η μία έχει μοιραστεί στα 2. Αντίθετα, η συστηματική επιλογή των αρχικών σημείων του αλγορίθμου οδηγεί στο σωστό διαχωρισμό των δεδομένων στις τρεις αναμενόμενες συστάδες.

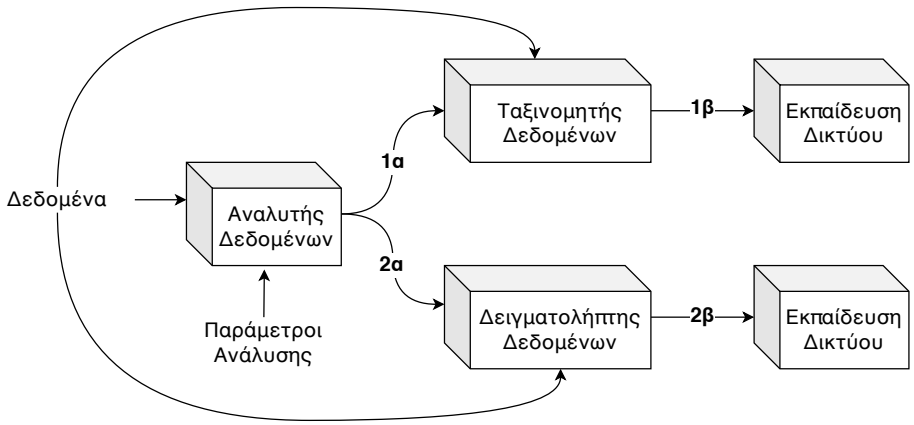
Κεφάλαιο 5

Τεχνικές Αξιοποίησης Κατανομής Δεδομένων

Στο συγκεκριμένο κεφάλαιο γίνεται περιγραφή του συστήματος που υλοποιήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας. Στην ενότητα 5.1 παρατίθεται μία συνολική περιγραφή του συστήματος, ενώ στις υπόλοιπες ενότητες (5.2, 5.3, 5.4) περιγράφονται τα επιμέρους στοιχεία του.

5.1 Περιγραφή Συστήματος

Το σύστημα που υλοποιήθηκε στα πλαίσια της παρούσας διπλωματικής εργασίας αποσκοπεί στην αξιοποίηση της πληροφορίας που υπάρχει στην κατανομή των δεδομένων με στόχο τη βελτίωση της εκπαίδευσης μοντέλων νευρωνικών δικτύων είτε από πλευράς ταχύτητας είτε από πλευράς ποιότητας. Η γενική δομή του συστήματος παρουσιάζεται στην Εικόνα 5.1.



Εικόνα 5.1: Διάγραμμα περιγραφής του συστήματος

Όπως φαίνεται στην Εικόνα 5.1, το σύστημα που υλοποιείται αποτελείται από επι μέρους στοιχεία τα οποία επιτελούν συγκεκριμένες λειτουργίες πριν ξεκινήσει η εκπαίδευση του δικτύου. Το σύστημα παρέχει τα στοιχεία *αναλυτής δεδομένων*, *ταξινομητής δεδομένων* και *δειγματολήπτης δεδομένων*. Το σύστημα ολοκληρώνει τη λειτουργίας του με την εκπαίδευση του επιθυμητού νευρωνικού δικτύου ρυθμίζοντας κατάλληλα τα δεδομένα προς εκπαίδευση με χρήση των προηγούμενων στοιχείων του. Στην συνέχεια, παρατίθεται μια συνοπτική περιγραφή των επιμέρους στοιχείων του συστήματος.

1. *Αναλυτής Δεδομένων:* Το συγκεκριμένο στοιχείο χρησιμοποιείται για να αναλύσει τα δεδομένα. Η ανάλυση στοχεύει στην οργάνωση των δεδομένων σε ομάδες που περιγράφουν κάποιο υποσύνολο του χώρου. Η γνώση αυτή αξιοποιείται από τα ακόλουθα δύο στοιχεία του συστήματος με στόχο την βελτίωση της εκπαίδευσης.

2. *Ταξινομητής Δεδομένων:* Ο ταξινομητής δεδομένων χρησιμοποιείται για να ορίσει τη σειρά με την οποία θα χρησιμοποιούνται τα δεδομένα για την εκπαίδευση του μοντέλου. Χρησιμοποιεί την ανάλυση που πραγματοποίησε ο αναλυτής και ταξινομεί με συστηματικό τρόπο τα δεδομένα μέσω αυτής.

3. *Δειγματολήπτης Δεδομένων:* Το στοιχείο αυτό αξιοποιεί την πληροφορία που διατίθεται από τον αναλυτή δεδομένων με στόχο να πραγματοποιήσει μία αποδοτική δειγματοληψία στα δεδομένα. Έτσι, αποσκοπεί μεν σε ταχύτερη εκπαίδευση με χρήση μόνο ενός δείγματος από το σύνολο δεδομένων, το οποίο όμως θα είναι αντιπροσωπευτικό της κατανομής των δεδομένων, ώστε να υπάρχει η μικρότερη δυνατή απώλεια στην ακρίβεια του μοντέλου.

Επομένως, ανάλογα την προτίμηση του χρήστη το σύστημα δίνει είτε τη δυνατότητα κατασκευής ενός καλύτερου μοντέλου είτε την δυνατότητα ταχύτερης εκπαίδευσης χωρίς να μειωθεί δραματικά η ακρίβεια του μοντέλου. Και στις δύο περιπτώσεις χρήσης αξιοποιείται ο αναλυτής δεδομένων, ενώ τα στοιχεία του ταξινομητή και του δειγματολήπτη δεδομένων αξιοποιούνται αποκλειστικά στην πρώτη και στην δεύτερη περίπτωση αντίστοιχα. Στις επόμενες ενότητες θα αναλυθεί κάθε ένα από τα τρία στοιχεία ξεχωριστά για την πλήρη κατανόηση του τρόπου λειτουργίας τους. Σημειώνεται ότι το βήμα της εκπαίδευσης του νευρωνικού δικτύου, που παρατίθεται στην Εικόνα 5.1, δεν παρουσιάζεται αναλυτικά σε επόμενες ενότητες, καθώς δεν είναι κάτι που τροποποιείται στον παρόν σύστημα. Αντίθετα, θα χρησιμοποιηθεί το μοντέλο που θα χρησιμοποιούσε ο χρήστης εξ αρχής με τις ίδιες υπερπαραμέτρους που θα έθετε.

5.2 Αναλυτής Δεδομένων

Στην συγκεκριμένη ενότητα γίνεται περιγραφή του αναλυτή δεδομένων. Ο στόχος του αναλυτή δεδομένων είναι να επεξεργαστεί τα δεδομένα και να τα περιγράψει όσο το δυνατόν πιο αποδοτικά με την μορφή της μοντέλου κανονικής σύνθεσης, το οποίο περιγράφηκε στην Ενότητα 4.1, προσεγγίζοντας έτσι όσο το δυνατόν περισσότερο την κατανομή των δεδομένων στον χώρο. Ο αλγόριθμος που ακολουθεί ο αναλυτής δεδομένων δίνεται στον Αλγόριθμο 5.

Αλγόριθμος 5 Ο αλγόριθμος ανάλυσης των δεδομένων

1: **procedure** ΑΝΑΛΥΣΗ($X, \theta, \mathcal{C}$)
2: ▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων εκπαίδευσης
3: ▷ θ η παράμετρος επιθυμητής διασποράς του PCA
4: ▷ $\mathcal{C}$το πλήθος των ομαδών διαχωρισμού των δεδομένων
5: **if** $\vec{x} \in X$ είναι τανυστής $m \times k$ **then**
6: $X \leftarrow$επιπεδοποίηση(X)
7: **end if**
8: $\vec{\mu} \leftarrow \frac{1}{n} \sum_{i=1}^{n} \vec{x}_i$
9: $\vec{s} \leftarrow$ διασπορά_ανά_χαρακτηριστικό(X)
10: $X \leftarrow$ κλιμακωση_χαρακτηριστικών($X, \vec{\mu}, \vec{s}$)
11: $X_{reduced} \leftarrow PCA(X, \theta)$
12: $clusters \leftarrow KMeans(X_{reduced}, \mathcal{C})$
13: **return** $clusters$
14: **end procedure**

Ο Αλγόριθμος 5 αξιοποιεί τις τεχνικές που αναφέρθηκαν στο Κεφαλαίο 4. Ωστόσο, είναι χρήσιμο για λόγους κατανόησης να εξηγηθούν τα επιμέρους βήματα του αλγορίθμου:

- **Παράμετροι Εισόδου:** Ως είσοδος στον αλγόριθμο χρησιμοποιείται το σύνολο δεδομένων που πρόκειται να αναλυθεί. Επιπλέον, δίνονται ως είσοδος οι παράμετροι $\theta, \mathcal{C}$, οι οποίοι αξιοποιούνται για την μείωση διαστάσεων των δεδομένων και την ομαδοποίηση τους σε γειτονιές από κανονικές κατανομές.

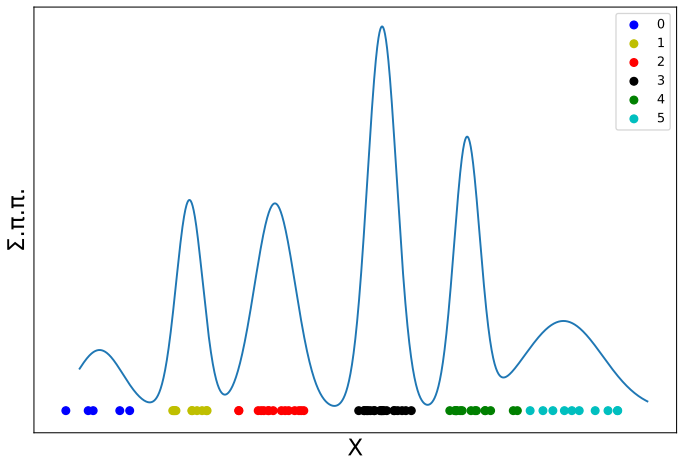
- **Βήμα 5 - 7:** Αρχικά, ο αλγόριθμος ελέγχει εάν τα επιμέρους δεδομένα είναι πολυδιάστατοι τανυστές σε μορφή πίνακα, όπως για παράδειγμα στην περίπτωση δεδομένων εικόνων. Στην περίπτωση αυτή τα επιπεδοποιεί σε μονοδιάστατα διανύσματα για την σωστή λειτουργία των επόμενων βημάτων.

- **Βήμα 8-9:** Στο συγκεκριμένο στάδιο, αξιοποιούνται τα δεδομένα για να υπολογιστεί η μέση τιμή και η διασπορά ανά χαρακτηριστικό. Έτσι προκύπτουν δύο διανύσματα όπου το κάθε ένα έχει στη θέση i την πληροφορία για το i-οστό χαρακτηριστικό.

- **Βήμα 10:** Στο συγκεκριμένο βήμα, τα δεδομένα κανονικοποιούνται ως προς κάθε χαρακτηριστικό σύμφωνα με τη μέση τιμή και τη διασπορά των χαρακτηριστικών. Το συγκεκριμένο βήμα είναι απαραίτητο ώστε κάθε χαρακτηριστικό να ακολουθεί προσεγγιστικά την τυποποιημένη κανονική κατανομή. Έτσι, τα επόμενα βήματα θα μπορούν να παράγουν τα σωστά αποτελέσματα. Η κανονικοποιήση για κάθε χαρακτηριστικό x_i ενός διανύσματος x γίνεται σύμφωνα με τη σχέση
$$\bar{x_i} = \frac{x_i - \mu_i}{\sigma_i}$$
όπου μ_i, σ_i η μέση τιμή και η τυπική απόκλιση του χαρακτηριστικού.

- **Βήμα 11:** Στο συγκεκριμένο βήμα εκτελείται ο αλγόριθμος PCA για τη μείωση των διαστάσεων του σύνολου δεδομένων με χρήση της παραμέτρου θ, η οποία καθορίζει το ποσοστό της σημαντικής πληροφορίας που θα υπάρχει στο μειωμένο ως προς τις διαστάσεις σύνολο, το οποίο θα χρησιμοποιεί το δεδομένο στοιχείο του συστήματος για να πραγματοποιήσει την απαραίτητη ανάλυση.

- **Βήμα 12:** Το συγκεκριμένο βήμα χρησιμοποιεί τον αλγόριθμο KMeans με στόχο την ανάλυση των δεδομένων σε μικρές συνιστώσες κανονικής κατανομής οι οποίες σχηματίζουν το μοντέλο κανονικής σύνθεσης. Η ομαδοποίηση αυτή των δεδομένων θα χρησιμοποιηθεί από τον ταξινομητή και τον δειγματολήπτη δεδομένων, όπως θα εξηγηθεί στις επόμενες ενότητες.



Εικόνα 5.2: Παράδειγμα δεδομένων που αναπαρίστανται από μοντέλο κανονικής σύνθεσης.

Για να γίνει κατανοητός ο τρόπος λειτουργίας του αναλυτή δεδομένων, είναι σκόπιμο να εξηγηθεί η επιλογή του αλγορίθμου KMeans για την ανάλυση των δεδομένων σε

συστάδες που να αντιπροσωπεύουν κάποια περιοχή του χώρου. Για την τεκμηριώση της επιλογής εξετάζεται ένα παράδειγμα μονοδιάστατων δεδομένων τα οποία περιγράφονται από την κατανομή της Εικόνας 5.2. Η μορφή της συνάρτησης πυκνότητας πιθανότητας δείχνει ότι τα δεδομένα αυτά μπορούν να παραστάθουν από ένα μοντέλο κανονικής σύνθεσης το οποίο αποτελείται από 6 κανονικές κατανομές, οι μέσες τιμές των οποίων δίνονται στον Πίνακα 5.1. Αξίζει να αναφερθεί ότι κάθε επιμέρους κανονική κατανομή της σύνθεσης προκύπτει από διαφορετικό πλήθος σημείων, όπως φαίνεται και στην εικόνα. Το ύψος της καμπύλης της αντίστοιχης κατανομής είναι ανάλογο με το πλήθος των σημείων από τα οποία προσεγγίζεται, δίνοντας μια εκτίμηση το πόσο πιθανό είναι ένα σημείο να προκύψει από αυτήν την κατανομή. Σύμφωνα με τη βιβλιογραφική ανασκόπηση που παρατέθηκε στο κεφάλαιο 4, ο KMeans είναι ένας ασθενής και ταχύτερος τρόπος να προσεγγιστούν οι ομάδες των δεδομένων, κάθε μία από τις οποίες προέρχεται από την ίδια κανονική κατανομή. Εφαρμόζοντας τον αλγόριθμο βρίσκονται τα κεντρικά σημεία γύρω από τα οποία εκτείνονται τα δεδομένα της κάθε ομάδες και τα οποία παρατίθενται επίσης στον Πίνακα 5.1.

Ομάδα	**Κεντρικό Σημείο Ομάδας**	**Μέση Τιμή Κανονικής Συνιστώσας**
0	0.624	0.5
1	3.479	3.5
2	6.193	6.5
3	9.589	9.5
4	12.502	12.5
5	15.799	15.5

Πίνακας 5.1: Κεντρικά Σημεία από την εκτέλεση του KMeανς και οι αντίστοιχες Μέσες Τιμές των Κανονικών Συνιστωσών που προσεγγίζουν την κατανομή της Εικόνας 5.2.

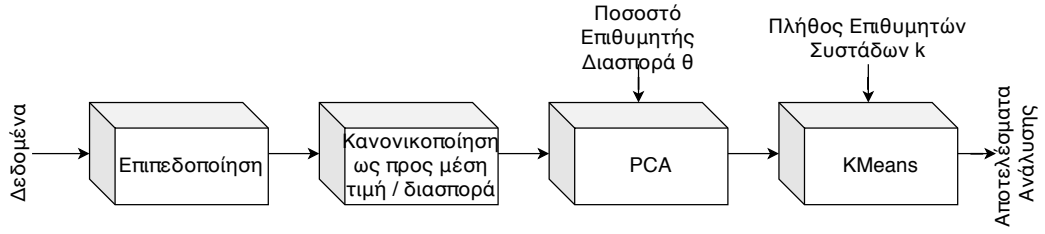
Σύμφωνα με τον Πίνακα 5.1 γίνεται φανερό ότι τα κεντρικά σημεία που προκύπτουν από την εφαρμογή του αλγορίθμου KMeans προσεγγίζουν σημαντικά την μέση τιμή των επιμέρους κανονικών κατανομών που περιγράφουν τα μονοδιάστατα αυτά δεδομένα. Επομένως, η χρήση του KMeans με κατάλληλη ρύθμιση του αριθμού των συστάδων που πρόκειται να εντοπίσει μπορεί να περιγράψει ικανοποιητικά το χώρο σε μορφή ομάδων δίνοντας τελικά στα δεδομένα μία ετικέτα που περιγράφει την προέλευση τους.

Όπως αναφέρθηκε παραπάνω, η κάθε κανονική κατανομή της σύνθεσης προκύπτει από διαφορετικό πλήθος σημείων και επομένως η κάθε αντίστοιχη συστάδα που προκύπτει από τον KMeans θα απαρτίζεται από διαφορετικό πλήθος σημείων. Επομένως διακρίνουμε δύο κατηγορίες συστάδων, τις πυκνές και τις αραιές, οι οποίες αντιμετωπίζονται διαφορετικά από τον ταξινομητή και τον δειγματολήπτη, όπως εξηγείται στις επόμενες ενότητες. Σημειώνεται ότι ως αραιές γειτονιές θεωρούνται αυτές με τη μικρότερη τάξη μεγέθους πλήθους στοιχείων εάν υπάρχει αντίστοιχο φαινόμενο.

Ωστόσο, καθώς στα πλαίσια της διπλωματικής γίνεται αναφορά σε προβλήματα επιβλεπόμενης κατηγοριοποίησης, προκύπτει η απορία γιατί δεν χρησιμοποιούνται οι ίδιες οι ετικέτες των δεδομένων για τον καθορισμό της προέλευσης τους και η χρήση του αναλυτή είναι απαραίτητη. Οι ετικέτες περιγράφουν μία φυσική ιδιότητα των δεδομένων, η οποία είναι επιθυμητό να αναγνωρίζεται, χωρίς όμως απαραίτητα να μπορεί να προσδιορίσει επαρκώς τη θέση των αντικειμένων στον χώρο. Στο παράδειγμα της Εικόνας 5.2, θα μπορούσε δεδομένα των συστάδων 1 και 5 να έχουν την ίδια φυσική ιδιότητα και επομένως την ίδια ετικέτα. Επομένως, δεν θα υπήρχε πλήρη γνώση του χώρου με χρήση της διαθέσιμης ετικέτας.

Σχετικά με τις υπόλοιπες λειτουργίες του αλγορίθμου, η επιλογή του PCA αφορά στη μείωση των διαστάσεων του χώρου του προβλήματος με στόχο την ταχύτερη εκτέλεση του αλγορίθμου KMeans. Καθώς χρειάζεται να ομαδοποιήσουμε τα δεδομένα σύμφωνα με τη θέση που βρίσκονται στο χώρο, είναι αρκετό να λάβουμε υπόψιν τις συνιστώσες υψηλής σημαντικότητας. Η προσέγγιση αυτή, καθώς δεν πρόκειται για επίλυση προβλήματων ομαδοποίησης αλλά για ένα βήμα σχετικό με τη βελτίωση εκπαίδευσης μοντέλου κατηγοριοποίησης, είναι αποδεκτή γιατί αποτυπώνει την απαραίτητη πληροφορία. Τα βήματα της κανονικοποίησης αφορούν καθαρά στην πιο αποδοτική εκτέλεση των αλγορίθμων και ιδίως του PCA. Η χρησιμότητα της πληροφορίας της ανάλυσης των δεδομένων θα φανεί εκτενώς στους αλγορίθμους των ενοτήτων 5.3 και 5.4.

Στην Εικόνα 5.3 δίνεται μια συνοπτική περιληπτική απεικόνιση του αναλυτή δεδομένων.



Εικόνα 5.3: Σχηματικό διάγραμμα του αναλυτή δεδομένων.

5.3 Ταξινομητής Δεδομένων

Στην συγκεκριμένη ενότητα σχολιάζεται η λειτουργία του ταξινομητή δεδομένων. Ο ταξινομητής αποσκοπεί στον καθορισμό της σειράς με την οποία θα χρησιμοποιούνται τα δεδομένα του συνόλου εκπαίδευσης, αντί για τη χρήση τυχαίας σειράς, για την εκπαίδευση του μοντέλου αξιοποιώντας τα αποτελέσματα του αναλυτή που περιγράφηκε στην ενότητα 5.2. Ο Αλγόριθμος 6 περιγράφει τον τρόπο λειτουργίας του ταξινομητή δεδομένων.

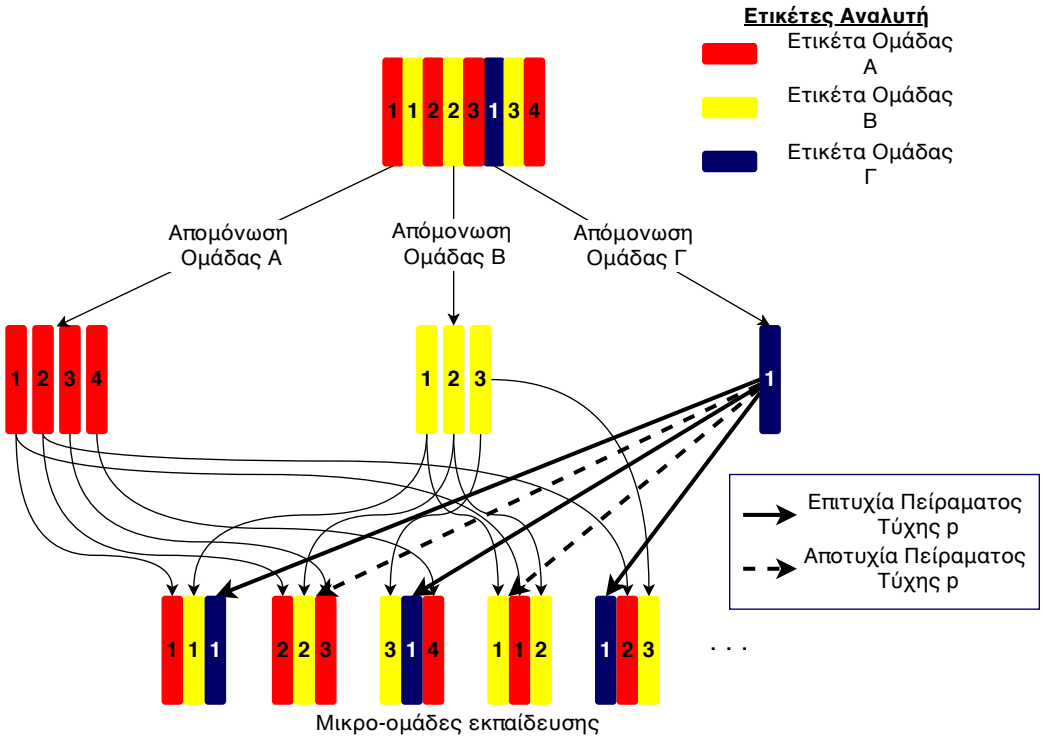
Αλγόριθμος 6 Ο αλγόριθμος του ταξινομητή των δεδομένων

procedure ΕΠΑΝΑΔΙΑΤΑΞΗ($X, y, labels, p$)
    ▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων εκπαίδευσης
    ▷ $y = \{y_1, y_2, ..., y_n\}$ οι ετικέτες των δεδομένων εκπαίδευσης
    ▷ $labels = l_1, l_2, ..., l_n$ετικέτα ομάδας σύμφωνα με το βήμα ανάλυσης
    $ul \leftarrow$ μοναδικά_στοιχεία($labels$) ▷ εύρεση διαφορετικών μοναδικών ετικετών ομάδας
    $data\_groups \leftarrow \emptyset$
    for $group \in ul$ **do**
        $x\_group, y\_group \leftarrow X(labels == group), y(labels == group)$
        $data\_groups \leftarrow data\_groups \cup \{(x\_group, y\_group)\}$
    end for
    $g\_iter \leftarrow$ επαναλήπτης_ομάδων($data\_groups$)
    $data\_iters \leftarrow \{\}$
    for $g \in data\_groups$ **do**
        $data\_iters\{g\} \leftarrow$ επαναλήπτης_δεδομένων(g)
    end for
    while training **do**
        $g \leftarrow data\_groups.next()$
        $\vec{x}, y \leftarrow data\_iters\{g\}.next()$
        if g είναι αραιή **then**
            Αποδέξου το σημείο εκπαίδευσης $\vec{x}, y$ ως επόμενο στη σειρά με πιθανότητα p
        else
            Αποδέξου το σημείο εκπαίδευσης $\vec{x}, y$ ως επόμενο
        end if
    end while
end procedure

Για καλύτερη κατανόηση, τα βήματα του αλγορίθμου 6 περιγράφονται στη συνέχεια:

- **Βήμα 5:** Εντοπισμός των διαφορετικών συστάδων που προέκυψαν από τον αναλυτή. Γίνεται έτσι γνωστό πως διαρθρώνονται τα δεδομένα στο χώρο.

- **Βήματα 6 - 10:** Για κάθε μία συστάδα εντοπίζεται το σύνολο των δεδομένων που ανήκουν σε αυτήν, ώστε να είναι δυνατή η συστηματική σειρά χρήσης τους.
- **Βήματα 11 - 15:** Κατασκευή επαναληπτών για κυκλική χρήση τόσο των διαθέσιμων συστάδων όσο και των δεδομένων της καθεμίας με επανατοποθέτηση της συστάδας / των δεδομένων στο τέλος ώστε να υπάρχει επαναληψιμότητα.
- **Βήματα 16 - 18:** Με χρήση των επαναληπτών προκύπτει η επόμενη ομάδα και από αυτήν το επόμενο προς χρήση δεδομένο εκπαίδευσης σε Round- Robin σειρά.
- **Βήμα 20:** Εάν το επόμενο προς χρήση δεδομένο προέρχεται από αραιή συστάδα, τότε αυτό θα τοποθετηθεί ως επόμενο μετά από ένα πείραμα τύχης Bernoulli με πιθανότητα p, υπό την προϋπόθεση ότι αυτό είναι επιτυχές. Η πιθανότητα επιτυχίας του πειράματος θεωρείται παράμετρος εισόδου του αλγορίθμου.
- **Βήμα 22:** Εάν το επόμενο προς χρήση δεδομένο προέρχεται από πυκνή συστάδα, τότε ακολουθεί στη σειρά χωρίς την εκτέλεση ενός πειράματος τυχής.



Εικόνα 5.4: Παράδειγμα λειτουργίας του ταξινομητή δεδομένων

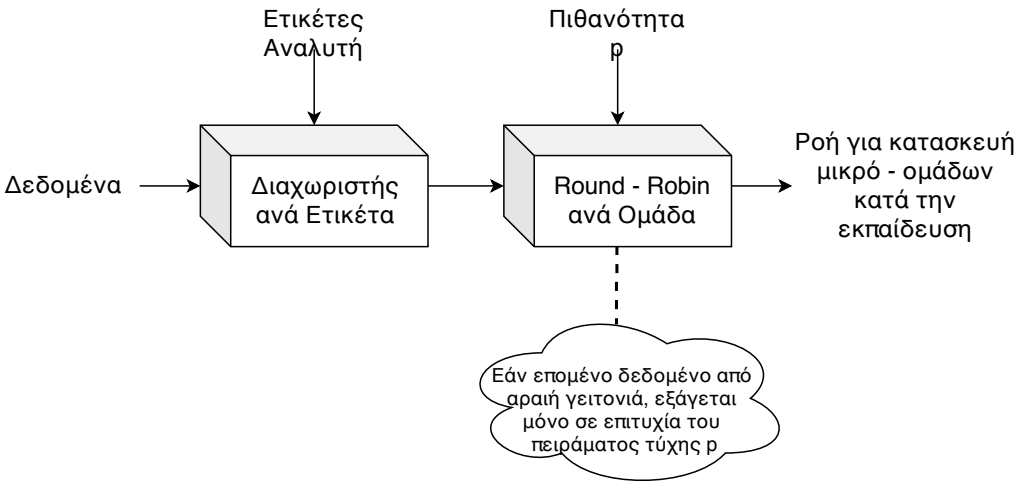
Ένα παράδειγμα εκτέλεσης του Αλγορίθμου 6 δίνεται στην Εικόνα 5.4. Στο παράδειγμα παρουσιάζεται ένα σύνολο 8 δεδομένων κάθε τα οποία ο αναλυτής χωρίζει

σε 3 ομάδες, την κόκκινη, την κίτρινη και την μπλε, σύμφωνα με τον τρόπο τον οποίο απεικονίζονται. Στην μπλε ομάδα έχει αντιστοιχηθεί μόνο ένα σημείο και αυτή η ομάδα για το παράδειγμα αποτελεί μία αραιή συστάδα, ενώ οι άλλες δύο θεωρούνται πυκνές. Έστω ότι το ζητούμενο είναι η κατασκευή μικρο-ομάδων για εκπαίδευση ενός μοντέλου μεγέθους 3. Διατρέχοντας κυκλικά τις συστάδες με τη σειρά κόκκινη, κίτρινη και μετά μπλε, τοποθετείται το επόμενο δεδομένο της συστάδας στην μικρο-ομάδα. Αρχικά, για τις πυκνές συστάδες, όταν έρχεται η σειρά προσπέλασης τους παρατηρούμε ότι χρησιμοποιείται πάντα το επόμενο κατά σειρά δεδομένο εκπαίδευσης. Για την περίπτωση της αραίης συστάδας (μπλε) γίνεται φανερό ότι δεν ισχύει το ίδιο, αλλά το μοναδικό δεδομένο που την απαρτίζει επιλέγεται μέσω ενός πειράματος τύχης. Στο παράδειγμα υποτίθεται ότι η πιθανότητα επιτυχίας τους πειράματος είναι 0.5. Για αυτό, στην Εικόνα παρουσιάζεται εναλλάξ η επιτυχία ή η αποτυχία του πειράματος τύχης και επομένως η επιλογή ή μη του σημείου.

Έχοντας εξηγήσει την λειτουργία του αλγορίθμου είναι απαραίτητο να σχολιαστεί γιατί επιχειρείται ο καθορισμός της σειράς πρόσβασης στα δεδομένα εκπαίδευσης. Στην ενότητα 2.3 αναφέρεται ότι ο βασικός αλγόριθμος καθόδου κλίσεων χρησιμοποιεί όλο το σύνολο δεδομένων σε κάθε επανάληψη εκπαίδευσης ενός μοντέλου. Επομένως, σε κάθε επανάληψη το μοντέλο εκπαίδευεται πάνω σε μια συνολική εικόνα του χώρου που περιγράφουν τα διαθέσιμα δεδομένα. Ωστόσο, στις παραλλαγές που χρησιμοποιούν μικρο-ομάδες, σε κάθε επανάληψη τα βάρη του νευρωνικού δικτύου προσαρμόζονται μόνο σύμφωνα με τη γνώση που εισάγουν οι εικόνες που χρησιμοποιήθηκαν στην κάθε μικρο-ομάδα. Η τυχαία επιλογή στοιχείων εγγυάται σε ένα βαθμό την διαφορετική πληροφορία μίας μικρο-ομάδας. Ωστόσο, στην παρούσα διπλωματική εξετάζεται αν μία συστηματική επιλογή είναι ικανή να επιφέρει καλύτερα αποτελέσματα εκπαίδευσης.

Σημαντικό είναι επίσης να εξηγηθεί ο τρόπος χειρισμού των δεδομένων που προέρχονται από αραιές συστάδες στην παραπάνω διαδικασία. Η ποικιλομορφία των δεδομένων στις πυκνές συστάδες δίνει την δυνατότητα χρήσης διαφορετικών σημείων εκπαίδευσης σε κάθε μικρο-ομάδα, δίνοντας έτσι στο μοντέλο σε κάθε επανάληψη μία αντιπροσωπευτική αλλά και ταυτόχρονα διαφορετική όψη. Ωστόσο, για να παίρνει το μοντέλο την εικόνα των αραιών συστάδων, η σημαντικά συχνότερη επανάληψη χρήσης των ίδιων δεδομένων που τις απαρτίζουν, σε σχέση πάντα με αυτήν των δεδομένων πυκνών συστάδων, επιφέρει κίνδυνο υπερπροσαρμογής (overfitting) του μοντέλου. Για την αντιμετώπιση του φαινομένου αυτού, ο αλγόριθμος τροποποιείται στην περίπτωση των αραιών συστάδων, όπως έχει αναφερθεί, ώστε να επιλέγει το επόμενο δεδομένο από αυτές στην περίπτωση επιτυχίας ενός πειράματος τύχης. Σε αντίθετη περίπτωση ο αλγόριθμος προσπερνάει την αντίστοιχη συστάδα και προχωράει στην επόμενη συστάδα όπου την εξέταζει με τον ίδιο τρόπο. Καθώς ακόμα και κατα αυτόν τον τρόπο είναι δεδομένη η συχνότερη επανάληψη χρήσης δεδομένων από αραιές γειτονιές, τεχνικές επαύξησης δεδομένων μπορούν να χρησιμοποιηθούν παράλληλα ώστε να αποφευχθούν φαινόμενα υπερπροσαρμογής ολοκληρωτικά.

Στην Εικόνα 5.5 δίνεται μία σχηματική απεικόνιση της λειτουργίας του ταξινομητή δεδομένων που εξηγήθηκε παραπάνω.



Εικόνα 5.5: Σχηματικό διάγραμμα του ταξινομητή δεδομένων

5.4 Δειγματολήπτης Δεδομένων

Ο δειγματολήπτης δεδομένων χρησιμοποιείται ώστε να απομονώσει ένα υποσύνολο του αρχικού συνόλου εκπαίδευσης, με στόχο την γρηγορότερη εκπαίδευση ενός νευρωνικού δικτύου, χωρίς όμως το μοντέλο που προκύπτει να στερείται της προβλεπτικής του ικανότητας, όσο αυτό είναι εφικτό. Ο αλγόριθμος του δειγματολήπτη αξιοποιεί την πληροφορία που παράγεται από τον αναλυτή δεδομένων και δίνεται στον Αλγόριθμο 7.

Αλγόριθμος 7 Ο αλγόριθμος του δειγματολήπτη δεδομένων

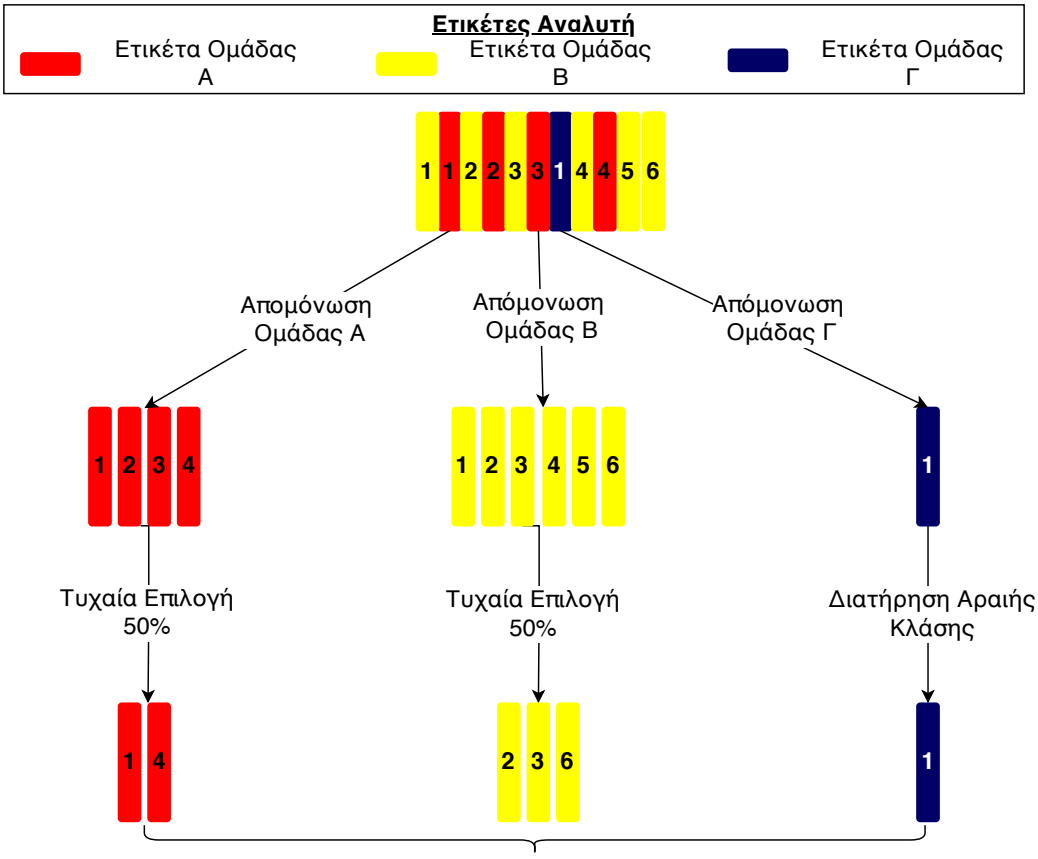
1: **procedure** ΔΕΙΓΜΑΤΟΛΗΨΙΑ($X, y, labels, p$)
2: ▷ $X = \vec{x}_1, \vec{x}_2, ..., \vec{x}_n$ το σύνολο δεδομένων εκπαίδευσης
3: ▷ $y = \{y_1, y_2, ..., y_n\}$ οι ετικέτες των δεδομένων εκπαίδευσης
4: ▷ $labels = l_1, l_2, ..., l_n$ετικέτα ομάδας σύμφωνα με το βήμα ανάλυσης
5: ▷ pτο επιθυμητό ποσοστό εγγραφών στο δείγμα
6: $ul \leftarrow$ μοναδικά_στοιχεία($labels$) ▷ εύρεση διαφορετικών μο-
7: $data\_groups \leftarrow \emptyset$ ναδικών ετικετών ομάδας
8: **for** $group \in ul$ **do**
9: $x\_group, y\_group \leftarrow X(labels == group), y(labels == group)$
10: $data\_groups \leftarrow data\_groups \cup \{(x\_group, y\_group)\}$
11: **end for**
12: $g\_iter \leftarrow$ επαναλήπτης_ομάδων($data\_groups$)
13: $sample \leftarrow \emptyset$
14: **for** $g \in g\_iter.next()$ **do**
15: **if** g είναι αραιή **then**
16: $sample \leftarrow sample \cup g$
17: **else**
18: $\tilde{g} \leftarrow$ τυχαίο_δείγμα(g, p)
19: $sample \leftarrow sample \cup \tilde{g}$
20: **end if**
21: **end for**
22: **end procedure**

Για καλύτερη κατανόηση, τα βήματα του αλγορίθμου 6 περιγράφονται στη συνέχεια:

- **Βήματα 6 - 12:** Τα βήματα αυτά είναι αντίστοιχη με αυτά του ταξινομητή, για αυτό και δεν περιγράφονται και εδώ.
- **Βήματα 13 - 21:** Γίνεται η επιλογή των δεδομένων από τις επιμέρους συστάδες που έχουν προκύψει από τα προηγούμενα βήματα

- **Βήμα 16:** Εάν η γειτονιά που εξετάζεται είναι αραιή, αυτή συμπεριλαμβάνεται ολόκληρη στο δείγμα.

- **Βήματα 18 - 19:** Από την ομάδα που εξετάζεται, γίνεται τυχαία επιλογή τόσων σημείων, ώστε το πλήθος τους να αποτελεί το ποσοστό p του δείγματος. Στην συνέχεια αυτό το υποσύνολο συμπεριλαμβάνεται στο δείγμα που θα χρησιμοποιηθεί στην εκπαίδευση.



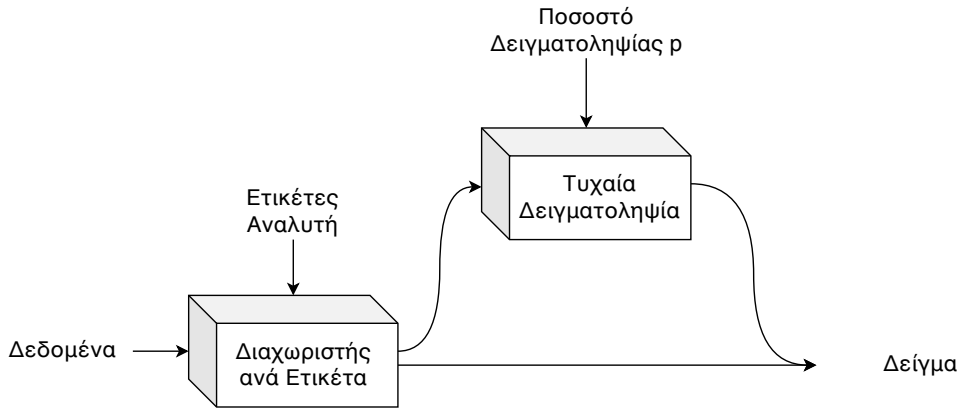
Εικόνα 5.6: Παράδειγμα λειτουργίας του δειγματολήπτη δεδομένων

Στην Εικόνα 5.6 δίνεται ένα παραδείγμα εκτέλεσης του δειγματολήπτη πάνω σε ένα σύνολο δεδομένων, με στόχο την απομόνωση ενός δείγματος μεγέθους ίσου με το μισό του πλήθους των συνολικών δεδομένων. Έχοντας απομονώσει τα δεδομένα κάθε ομάδας, το σύστημα είναι ελεύθερο να προχωρήσει σε δειγματοληψία του απαραίτητου πλήθους δεδομένων από την κάθε μία. Έτσι, από την κίτρινη και την κόκκινη ομάδα που είναι πυκνές, το σύστημα επιλέγει τυχαία τις μισές εγγραφές, δηλαδή 3 και 2

αντίστοιχα. Το δεδομένο της μπλε ομάδας θα εισαχθεί ανεξάρτητα στο δείγμα, καθώς προέρχεται από αραιή ομάδα.

Έχοντας κατασκευάσει το δείγμα, η διαδικασία εκπαίδευσης συνεχίζεται με ίδιες τιμές υπερπαραμέτρων σαν να μην είχε προηγηθεί η διαδικασία δειγματοληψίας. Η μόνη διαφορά εντοπίζεται στο πλήθος των μικρο-ομάδων που πρέπει να επεξεργαστούν, ώστε να έχουμε ολοκληρώσει μία εποχή εκπαίδευσης δεδομένων. Επομένως, είναι αναμενόμενο η διαδικασία της εκπαίδευσης να διαρκεί λιγότερο καθώς η κάθε εποχή ολοκληρώνεται με λιγότερες εφαρμογές του αλγορίθμου καθόδου κλίσεων.

Ένα συνοπτικό διάγραμμα του συστήματος δειγματολήψιας δίνεται στην Εικόνα 5.7.



Εικόνα 5.7: Σχηματικό διάγραμμα του δειγματολήπτη δεδομένων

Κεφάλαιο 6

Πειραματική Αξιολόγηση

Στο συγκεκριμένο κεφάλαιο της διπλωματικής εργασίας παρουσιάζεται η πειραματική αξιολόγηση των τεχνικών που αναλύθηκαν στο Κεφάλαιο 5 καθώς και τα αποτελέσματα αυτής. Η πειραματική αξιολόγηση γίνεται με χρήση δημοφιλών συνόλων δεδομένων που εξετάζονται για την μελέτη της απόδοσης νευρωνικών δικτύων και τα οποία παρουσιάζονται στην Ενότητα 6.1. Στην Ενότητα 6.2 παρουσιάζεται η διάρθρωση της πειραματικής αξιολόγησης, με τα σχετικά αποτελέσματα να δίνονται στις Ενότητες 6.3 και 6.4 για την περίπτωση χρήσης του ταξινομητή και του δειγματολήπτη δεδομένων αντίστοιχα.

6.1 Σύνολα Δεδομένων

Για τη μέλετη των τεχνικών αξιοποίησης της κατανομής που κρύβουν τα δεδομένα, χρησιμοποιήθηκαν σύνολα εικόνων, τα οποία αποτελούν ορόσημο σε πειραματισμούς με νευρωνικά δίκτυα στον τομέα της κατηγοριοποίησης εικόνας. Τα σύνολα που αξιοποιήθηκαν είναι:

- CIFAR-10
- CIFAR-100
- Tiny-Imagenet

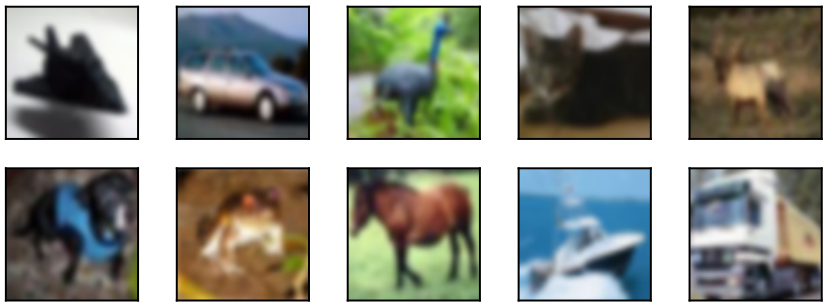
Στις επόμενες υποενότητες παρουσιάζονται συνοπτικά τα χαρακτηριστικά του κάθε συνόλου δεδομένων και δίνονται παραδείγματα εικόνων από κάθε κατηγορία που παρέχεται στο κάθε σύνολο.

6.1.1 CIFAR-10

Το σύνολο εικόνων CIFAR-10 [55] είναι ένα σύνολο έγχρωμων εικόνων με πλήρως διακρίσιμες κατηγορίες. Συνοπτικά, τα χαρακτηριστικά του συγκεκριμένου συνόλου εικόνων δίνονται στον Πίνακα 6.1, ενώ παραδείγματα εικόνων από την κάθε κλάση δίνονται στην Εικόνα 6.1.

Πίνακας 6.1: Περιγραφή του CIFAR-10.

Χαρακτηριστικό	**Τιμή**
Μέγεθος Εικόνας	$32 \times 32 \times 3$
Μέγεθος Συνόλου Εκπαίδευσης	50000
Εικόνες Εκπαίδευσης Ανά Κατηγορία	5000
Πλήθος Κλάσεων Ταξινόμησης	10
Μέγεθος Συνόλου Αξιολόγησης	10000
Εικόνες Αξιολόγησης Ανά Κατηγορία	1000



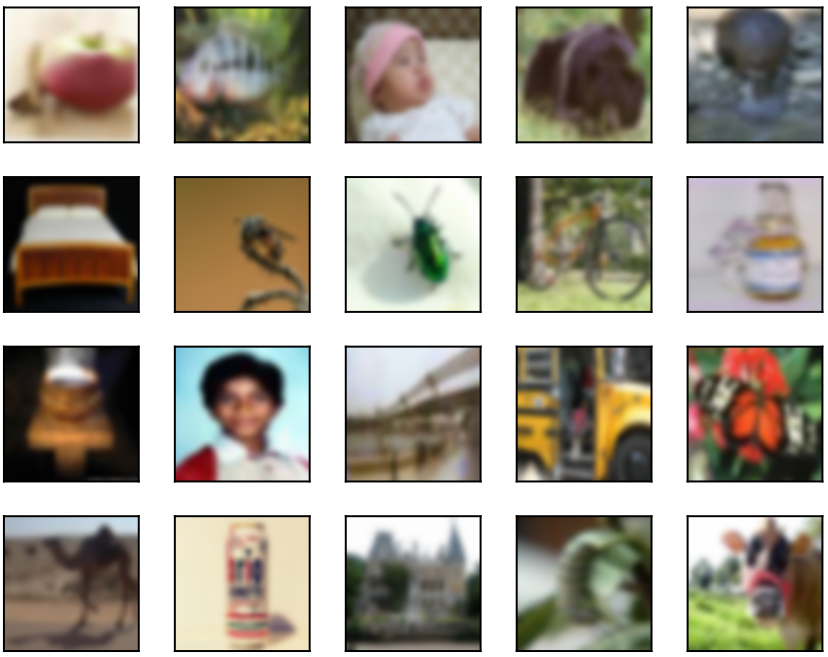
Εικόνα 6.1: CIFAR-10: Παραδείγματα Εικόνων

6.1.2 CIFAR - 100

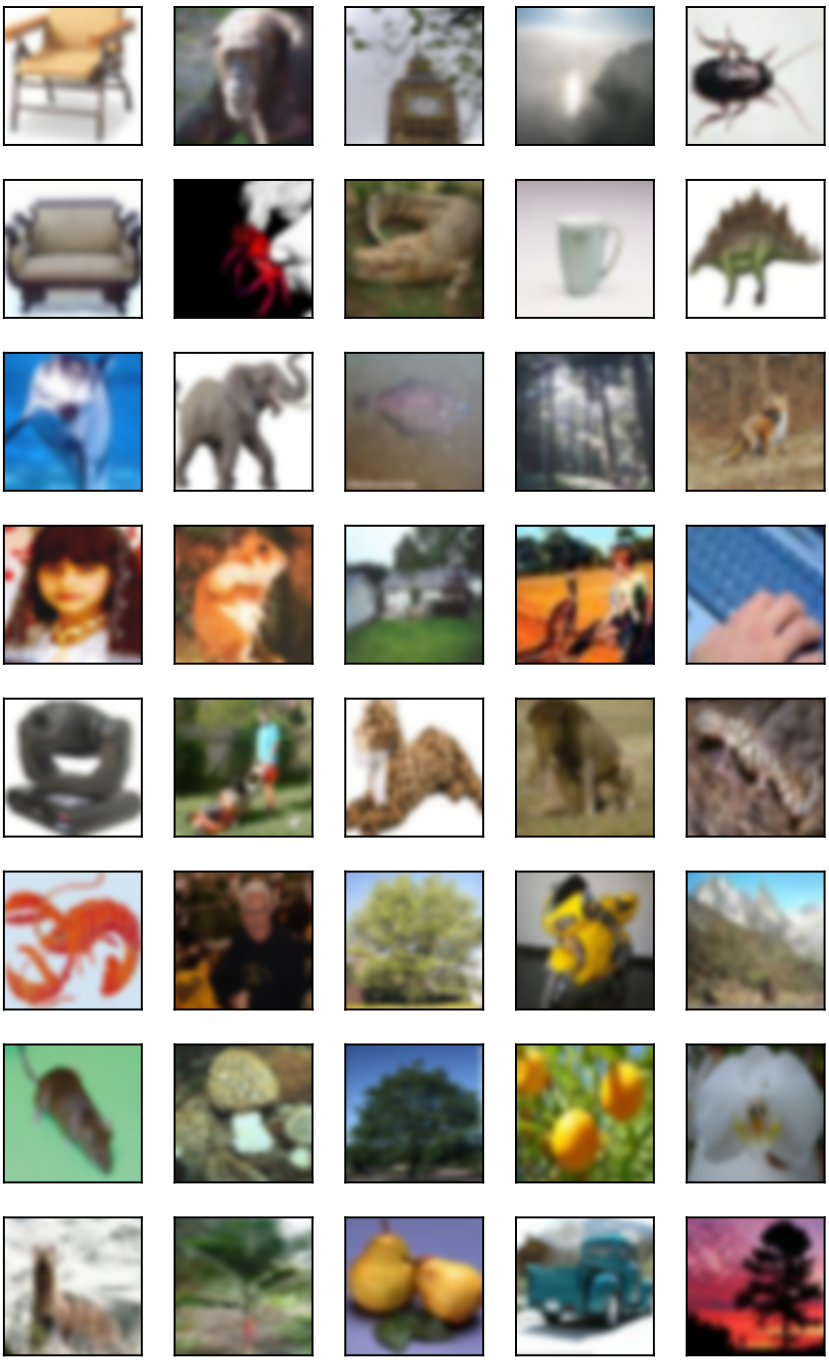
Το σύνολο εικόνων CIFAR-100 [55] είναι ένα σύνολο έγχρωμων εικόνων με επίσης πλήρως διακρίσιμες κατηγορίες. Οι κατηγορίες του συνόλου αυτού μπορούν να ομαδοποιηθούν σε άλλες γενικότερες, αλλά οι γενικές αυτές κατηγορίες δεν θα αξιοποιηθούν στο πειραματικό μέρος της εργασίας. Συνοπτικά, τα χαρακτηριστικά του συγκεκριμένου συνόλου εικόνων δίνονται στον Πίνακα 6.2, ενώ παραδείγματα εικόνων από την κάθε κλάση δίνονται στις Εικόνες 6.2 - 6.4.

Πίνακας 6.2: Περιγραφή του CIFAR-100.

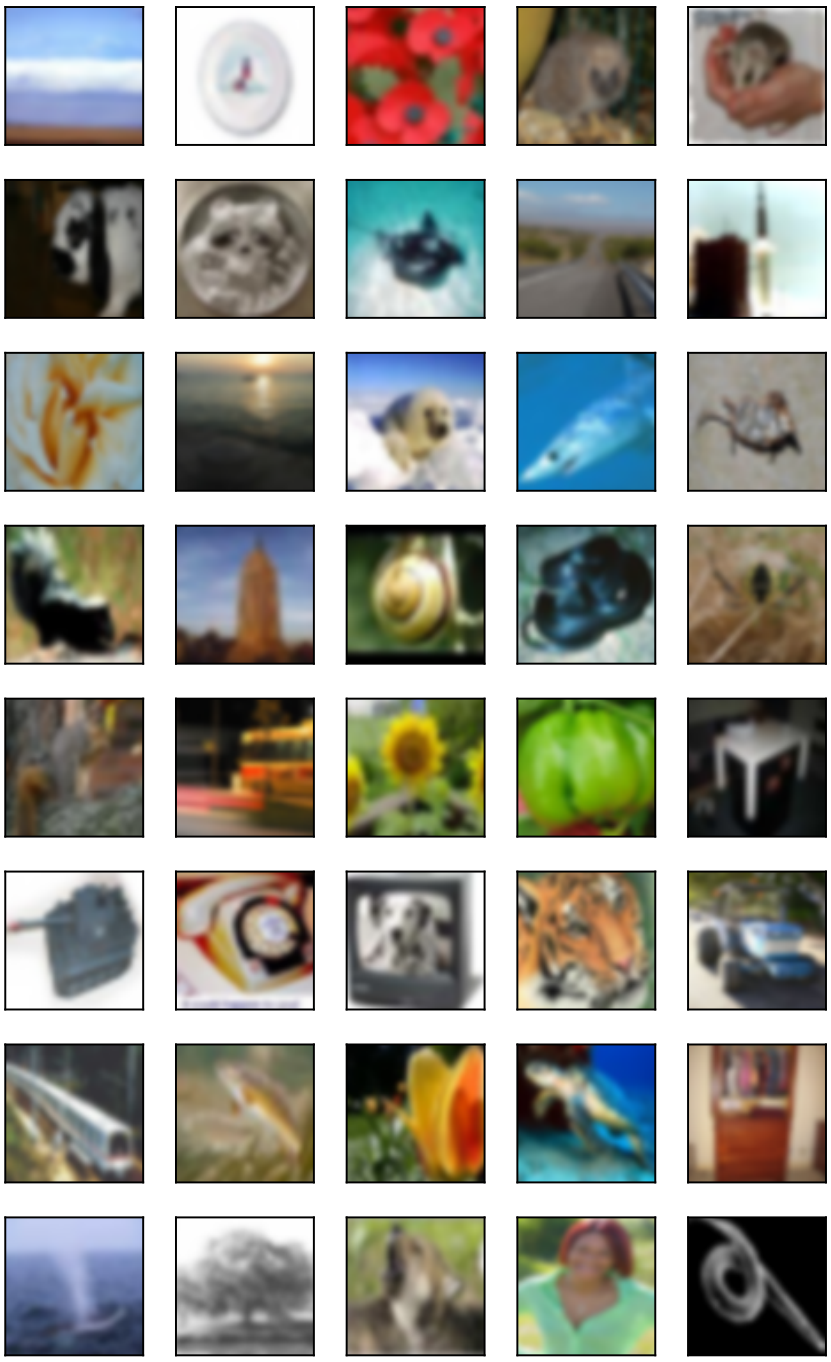
Χαρακτηριστικό	**Τιμή**
Μέγεθος Εικόνας	$32 \times 32 \times 3$
Μέγεθος Συνόλου Εκπαίδευσης	50000
Εικόνες Εκπαίδευσης Ανά Κατηγορία	500
Πλήθος Κλάσεων Ταξινόμησης	100
Μέγεθος Συνόλου Αξιολόγησης	10000
Εικόνες Αξιολόγησης Ανά Κατηγορία	100



Εικόνα 6.2: CIFAR-100 (A): Παραδείγματα Εικόνων



Εικόνα 6.3: CIFAR-100 (B): Παραδείγματα Εικόνων



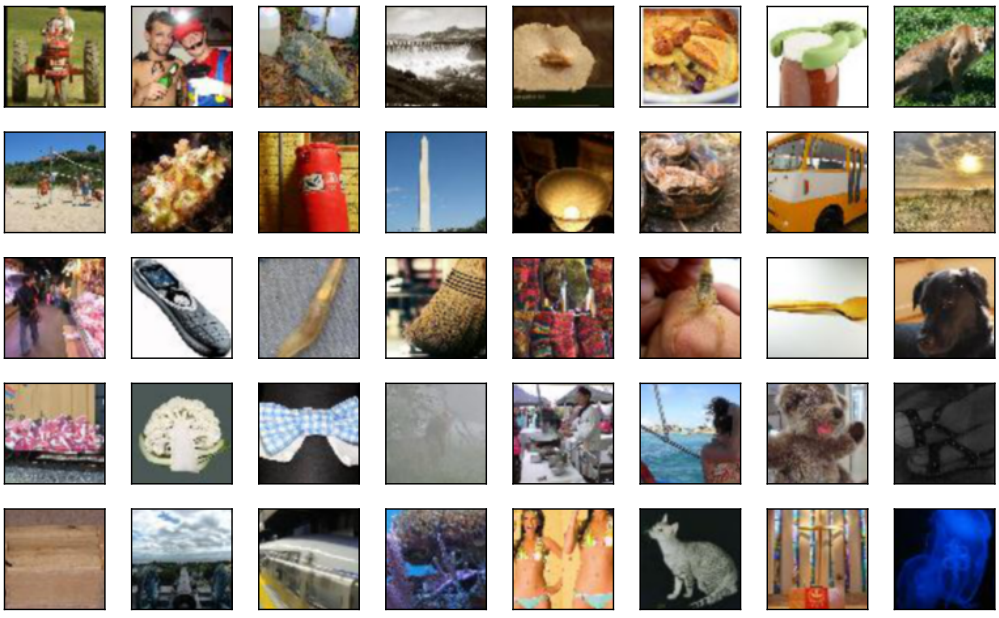
Εικόνα 6.4: CIFAR-100 (C): Παραδείγματα Εικόνων

6.1.3 Tiny - Imagenet

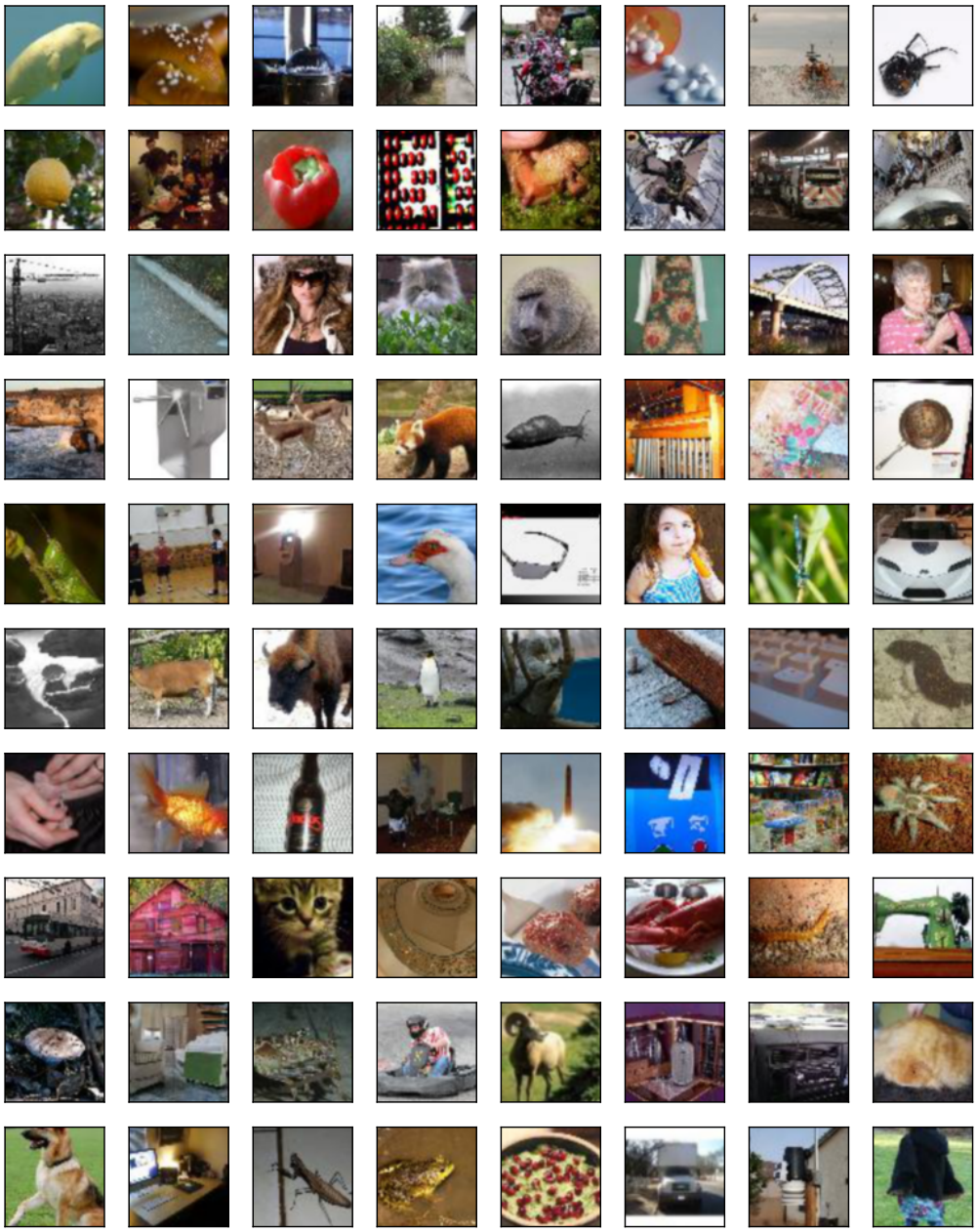
Το σύνολο δεδομένων Imagenet [56] είναι μία τεράστια συλλογή οπτικών δεδομένων για χρήση στον τομέα της αναγνώρισης αντικειμένων. Η συλλογή αυτή αποτελείται από τουλάχιστον 14 εκατομμύρια εικόνες, οι οποίες διαχωρίζονται σε πάνω από 20.000 κατηγορίες. Επείδη το μέγεθος του συνόλου αυτού είναι πολύ μεγάλο για τη διαθέσιμη υποδομή, χρησιμοποιήθηκε ένα υποσύνολο του στην πειραματική διαδικασία, το Tiny - Imagenet [57], το οποίο χρησιμοποιείται και προτείνεται από το Stanford. Τα χαρακτηριστικά αυτού του υποσυνόλου δίνονται στον Πίνακα 6.3, ενώ παραδείγματα εικόνων από τις επιμέρους κλάσεις του συνόλου δίνονται στις Εικόνες 6.5 - 6.7.

Πίνακας 6.3: Περιγραφή του Tiny - Imagenet.

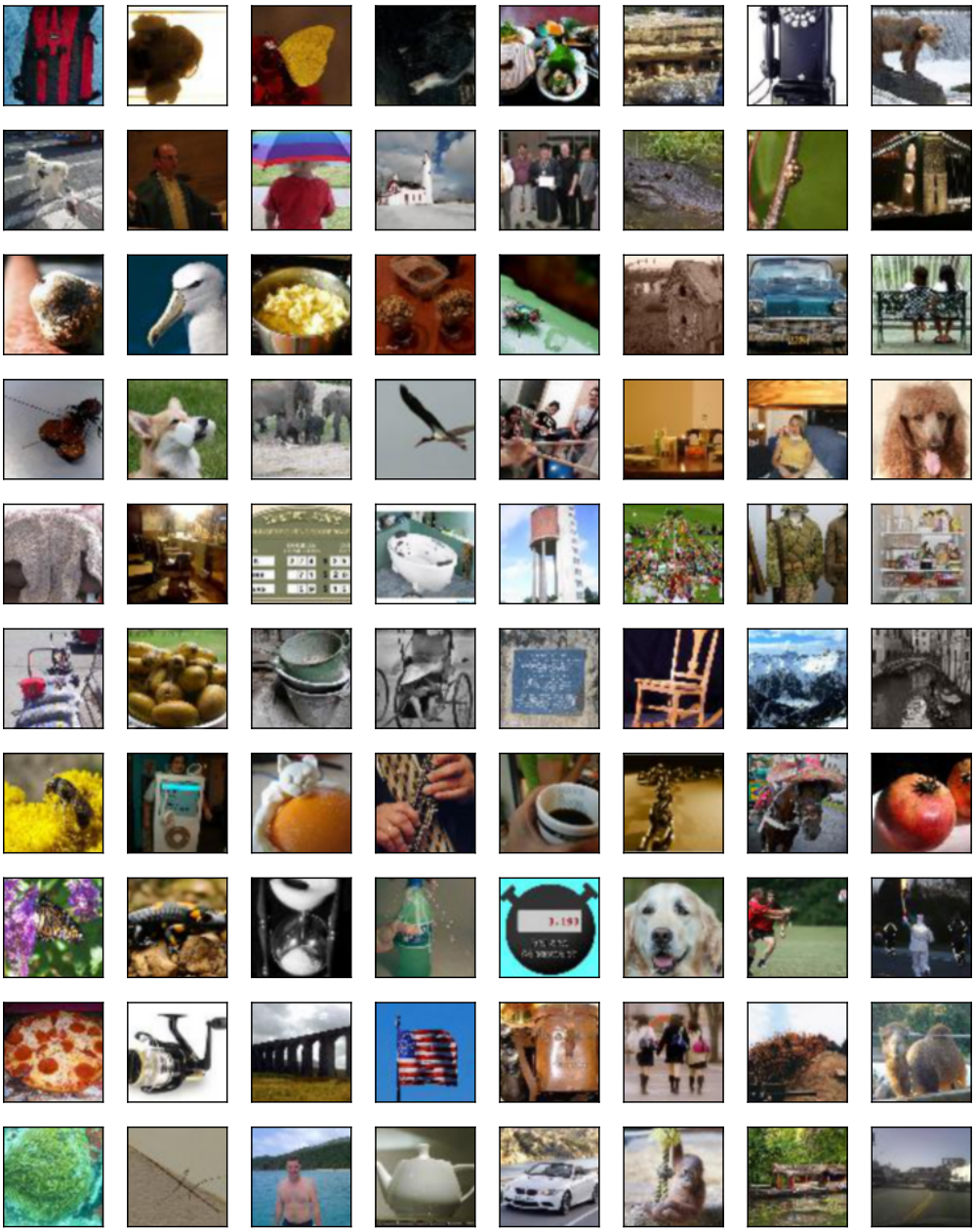
Χαρακτηριστικό	**Τιμή**
Μέγεθος Εικόνας	$64 \times 64 \times 3$
Μέγεθος Συνόλου Εκπαίδευσης	100000
Πλήθος Κλάσεων Ταξινόμησης	200
Εικόνες Εκπαίδευσης Ανά Κατηγορία	500
Μέγεθος Συνόλου Αξιολόγησης	10000
Εικόνες Αξιολόγησης Ανά Κατηγορία	50



Εικόνα 6.5: Tiny - Imagenet (A): Παραδείγματα Εικόνων



Εικόνα 6.6: Tiny - Imagenet (B): Παραδείγματα Εικόνων



Εικόνα 6.7: Tiny - Imagenet (C): Παραδείγματα Εικόνων

6.2 Περιγραφή και Διάρθρωση Πειραματικής Αξιολόγησης

6.2.1 Περιγραφή Υποδομής

Η πειραματική αξιολόγηση των τεχνικών που παρουσιάστηκαν στο Κεφάλαιο 5 πραγματοποιήθηκε απομακρυσμένα στο μηχάνημα Gold1 του Εργαστηρίου Υπολογιστικών Συστημάτων του Ε.Μ.Π. Τα χαρακτηριστικά της συγκεκριμένης υποδομής παρουσιάζονται στον Πίνακα 6.4.

Πίνακας 6.4: Χαρακτηριστικά Υποδομής που χρησιμοποιήθηκε για την Εκτέλεση των Πειραμάτων.

Χαρακτηριστικό	**Τιμή**
Μοντέλο Επεξεργαστή	Intel(R) Xeon(R) Gold 5120 CPU @ 2.20GHz
Πλήθος Πυρήνων	56
Νήματα / Πυρήνα	2
Μνήμη RAM	256 GB
Μοντέλο GPU	NVidia GeForce GTX 1060 6GB

Η υλοποίηση των τεχνικών του Κεφαλαίου 5 έγινε σε γλώσσα Python, και συγκεκριμένα με χρήση των βιβλιοθηκών NumPy [58] και scikit-learn [59]. Η δεύτερη βιβλιοθήκη χρησιμοποιήθηκε για την αξιοποίηση των αλγορίθμων KMeans και PCA, που όπως αναφέρθηκε στην Ενότητα 5.2 χρησιμοποιεί ο αναλυτής δεδομένων. Για την εκπαίδευση των νευρωνικών δικτύων στις διάφορες πειραματικές δοκιμές χρησιμοποιήθηκε το σύστημα TensorFlow [60], ενώ για την υλοποίηση των νευρωνικών δικτύων χρησιμοποιήθηκε η βιβλιοθήκη Keras [61] που παρέχεται στο TensorFlow. Στον Πίνακα 6.5 παρουσιάζονται οι εκδόσεις των βιβλιοθηκών και των συστημάτων που αναφέρθηκαν.

Πίνακας 6.5: Εκδόσεις Συστημάτων και Βιβλιοθηκών που χρησιμοποιήθηκαν.

Σύστημα	**Έκδοση**
NumPy	1.18.5
Scikit-Learn	0.23.2
TensorFlow	2.3.0

Τα πειράματα εκτελέστηκαν σε ένα Docker container [62] το οποίο περιέχει όλες τις παραπάνω βιβλιοθήκες.

6.2.2 Περιγράφη Πειραματικής Διαδικασίας

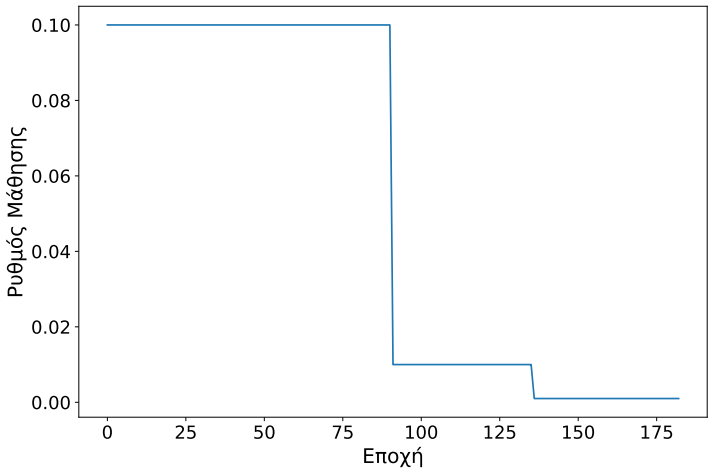
Τόσο για την αξιολόγηση του ταξινομητή όσο και για την αξιολόγηση του δειγματολήπτη χρησιμοποιούνται όλα τα σύνολα δεδομένων που παρουσιάστηκαν στην Ενότητα 6.1. Συγκεκρίμενα, για τα διάφορα πειράματα χρησιμοποιούνται δίκτυα της οικογένειας αρχιτεκτονικής ResNet (Ενότητα 3.2.1) για τα σύνολα δεδομένων CIFAR-10 και CIFAR-100, ενώ το δίκτυο InceptionV3 (Ενότητα 3.2.2) χρησιμοποιείται για πειράματα με το σύνολο δεδομένων Tiny - Imagenet.

Στους συνδυάσμους δικτύων και δεδομένων που αναφέρθηκαν, εξετάζονται τα εξής:

- **Μελέτη του ταξινομητή:** Σχετικά με τον ταξινομητή σχολιάζεται κατά πόσο βελτιώνει την τελική απόδοση του προβλεπτικού νευρωνικού δικτύου και σχολιάζεται κατά πόσο προσδίδει σημαντική χρονική καθυστέρηση στη συνολική διαδικασία εκπαίδευσης. Επίσης, μελετάται η επίδραση της παραμέτρου πιθανότητας επιτυχίας του πειράματος τύχης στην τελική απόδοση.

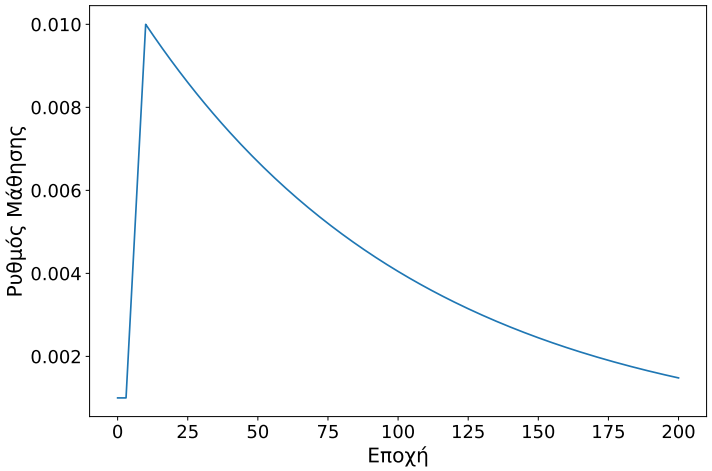
- **Μελέτη του δειγματολήπτη:** Στην περίπτωση του δειγματολήπτη εξετάζεται κατά πόσο είναι εφικτό να προκύψει παρόμοιο μοντέλο χρησιμοποιώντας ένα κατάλληλα επιλεγμένο υποσύνολο του συνόλου εκπαίδευσης κάνοντας γρηγορότερη τη διαδικασία εκπαίδευσης.

Ανεξαρτήτως του τι μελετά κάθε στάδιο της πειραματικής διαδικασίας και του συνόλου δεδομένων που χρησιμοποιείται, τα δύο δίκτυα χρησιμοποιούν τον αλγορίθμο mini-batch SGD (Ενότητα 2.4.2) για την εκπαίδευση του με χρήση Momentum τιμής 0.9 (Ενότητα 2.4.3). Ωστόσο, ανάλογα αν χρησιμοποιείται σύνολο δεδομένων με το δίκτυο ResNet ή με το InceptionV3 διαφοροποιείται ο μεταβλητός ρυθμός μάθησης που χρησιμοποιείται στη διαδικασία εκπαίδευσης.



Εικόνα 6.8: Ρυθμός Μάθησης για το δίκτυο ResNet.

Στην Εικόνα 6.8 δίνεται ο ρυθμός μάθησης που χρησιμοποιείται κατά την εκπαίδευση ενός δικτύου ResNet στην εξέλιξη του ανά εποχή. Στην Εικόνα φαίνεται πρακτικά ότι ο ρυθμός μάθησης ακολουθεί μια βηματική μείωση, η οποία έχει εξηγηθεί στο Κεφάλαιο 2.



Εικόνα 6.9: Ρυθμός Μάθησης για το δίκτυο InceptionV3.

Για την περίπτωση ενός δικτύου InceptionV3 χρησιμοποιείται ο ρυθμός μάθησης που παρουσιάζεται στην Εικόνα 6.9. Αρχικά, χρησιμοποιείται μία σημαντικά χαμηλότερη τιμή ως ρυθμός μάθησης για τις τρεις πρώτες εποχές, με στόχο να μην αποκλίνει το μοντέλο λόγω αυξημένης πολυπλοκότητας. Στη συνέχεια, ο ρυθμός μάθησης αυξάνεται σταδιακά για τις επτά επόμενες εποχές, ώστε να υπάρχει μια διαδικασία προθέρμανσης στην εκπαίδευσης σχετικά με την αύξηση της τιμής του ρυθμού μάθησης. Έπειτα ο ρυθμός μάθησης ακολουθεί εκθετικά φθίνουσα πορεία ως προς τις εποχές.

Πολλές φορές για αντιμετώπιση της υπερπροσαρμογής των δικτύων στα δεδομένα εκπαίδευσης χρησιμοποιούνται τεχνικές επαύξησης δεδομένων, για να αυξήθεί δυναμικά η πολυπλοκότητα του συνόλου εκπαίδευσης. Ένας απλός μηχανισμός επαύξησης δεδομένων χρησιμοποιείται σε όλα τα πειράματα που πραγματοποιήθηκαν. Συγκεκριμένα, τα δεδομένα διαιρούνται με την μέγιστη τιμή τους και στη συνέχεια κανονικοποιούνται περαιτέρω αφαιρώντας τη μέση τιμή τους. Έπειτα μεγαλώνουν οι διαστάσεις των εικόνων με μηδενικές τιμές πίξελ περιμετρικά. Από αυτή τη μεγαλύτερη σε διαστάσεις εικόνα αποκόπτεται τυχαία ένα κομμάτι που θεωρείται το νέο δεδομένο εκπαίδευσης, έχοντας περάσει πρώτα από πιθανή οριζόντια περιστροφή.

Στον Πίνακα 6.6 παρουσιάζονται συνοπτικά όλες οι λεπτομέρειες που αφορούν στην διαδικασία εκπαίδευσης των νευρωνικών δικτύων.

Πίνακας 6.6: Παράμετροι Εκπαίδευσης Νευρωνικών Δικτύων

	ResNet	**InceptionV3**
Βελτιστοποίητης	Mini-Batch SGD	Mini-Batch SGD
Momentum	0.9	0.9
Εποχές Εκπαίδευσης	182	200
Μέγεθος Mini-Batch	128	1024
Ρυθμός Μάθησης	Βηματική Μείωση	Ομαλή Έναρξη + Προθέρμανση + Εκθετική Μείωση
Επαύξηση Δεδομένων	Ναι	Ναι

Σχετικά με τον αναλυτή δεδομένων, χρησιμοποιείται στα πειράματα ως εργαλείο και δεν εξετάζεται στα πλαίσια της εργασίας η επίδραση των διαφόρων παραμέτρων του στην διαδικασία. Συγκεκριμένα, για κάθε σύνολο δεδομένων έχει γίνει παραδοχή ότι θα αναλύεται σε διπλάσιο πλήθος συστάδων σε σχέση με τις διαφορετικές κατηγορίες από τις οποίες συνοδεύεται στα πλαίσια του προβλήματος κατηγοριοποίησης. Αντίστοιχα σταθερό είναι και το ποσοστό των σημαντικών συνιστωσών που εξετάζει ο αλγόριθμος και δεν μελετάται ως παράμετρος.

Τα πειραματικά αποτελέσματα που παρουσιάζονται έχουν προκύψει ύστερα από 3 επαναλήψεις του κάθε πειράματος για καλύτερη ποιότητα των αποτελέσματων. Επομένως, ο σχολιασμός που θα ακολουθήσει στις επόμενες ενότητες, αφορά των μέσο όρο των τιμών που προέκυψαν από τα επιμέρους πειράματα που πραγματοποίηθηκαν.

6.3 Αξιολόγηση Χρήσης Ταξινομητή Δεδομένων

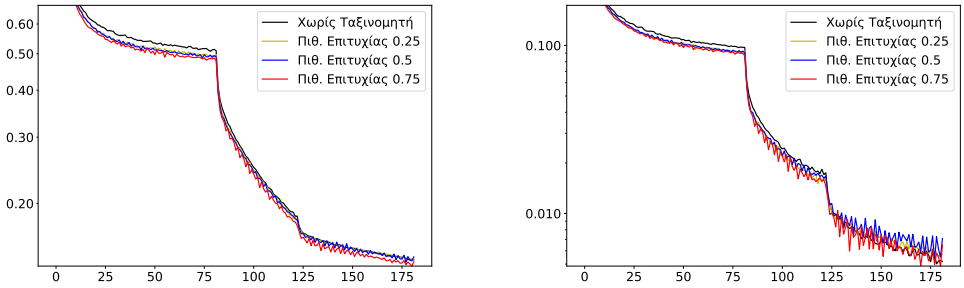
Στην παρούσα ενότητα γίνεται παρουσίασή και σχολιασμός των πειραματικών αποτελεσμάτων που προέκυψαν από την εφαρμογή του ταξινομητή πάνω στα τρία σύνολα δεδομένων που παρουσιάστηκαν στην Ενότητα 6.1. Το πρώτο μέρος αφορά στην μελέτη της επίδρασής του ταξινομητή στην προβλεπτική ικανότητα του μοντέλου, ενώ στο δεύτερο μέρος γίνεται αναφορά στο επιπλέον χρόνο που εισάγεται στην συνολική διαδικασία από τη χρήση του ταξινομητή. Για τα σύνολα δεδομένων της οικογένειας CIFAR γίνονται πειράματα και με ένα απλό και με ένα πιο πολύπλοκο νευρωνικό δίκτυο της οικογένειας αρχιτεκτονικής ResNet. Συγκεκριμένα χρησιμοποιούνται τα νευρωνικά δίκτυα ResNet-20v1 και ResNet-56v1.

6.3.1 CIFAR-10

Αρχικά, θα παρουσιαστούν τα πειραματικά αποτελέσματα για το πιο απλό σύνολο δεδομένων ανάμεσα σε αυτά που εξετάζεται που είναι το CIFAR-10. Για καλύτερη

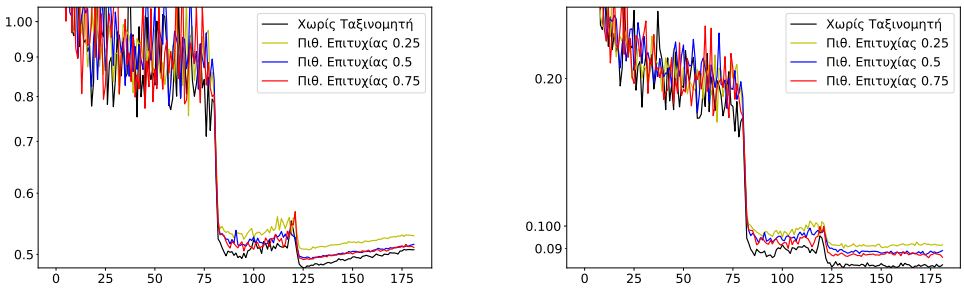
επεξέγηση των αποτελεσμάτων παρουσιάζονται πρώτα τα αποτελέσματα με εκπαίδευση του δικτύου ResNet-20v1 και στην συνέχεια αυτά που προκύπτουν από την εκπαίδευση του ResNet-56v1.

ResNet-20v1



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.10: CIFAR-10 με ResNet-20v1 : Εξέλιξη Μετρικών Εκπαίδευσης



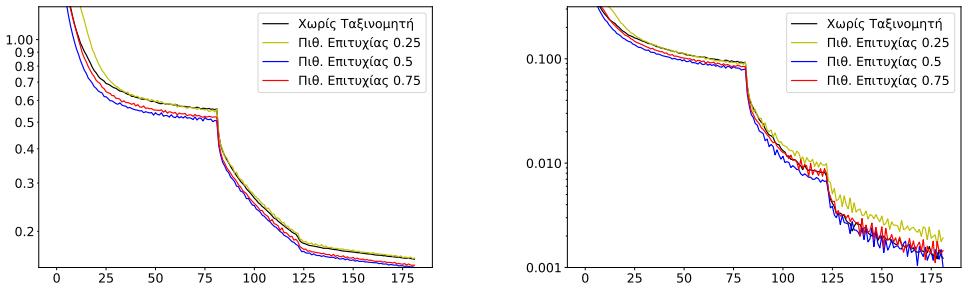
(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.11: CIFAR-10 με ResNet-20v1 : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.10 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.10α′) και του σφάλματος πρόβλεψης (Εικόνα 6.10β′) κατά την εκπαίδευση του δικτύου ResNet-20v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.11. Κάθε διάγραμμα περιλαμβάνει την εξέλιξη της αντίστοιχης μετρικής όταν δεν έχει χρησιμοποιηθεί ταξινομητής (τυχαία προσπέλαση στα δεδομένα) και όταν έχει χρησιμοποιηθεί δοκιμάζοντας διάφορες τιμές της πιθανότητας επιλογής δεδομένων στην περίπτωση αραιής κλάσης. Παρατηρώντας τα διαγράμματα γίνεται άμεσα φανερό, ότι όσο μειώνεται η πιθανότητα επιλογής ενός στοιχείου από μια αραιή γειτονιά, τόσο η συνάρτηση κόστους όσο και το σφάλμα πρόβλεψης προσεγγίζουν λιγότερο τις αντίστοιχες τιμές που ανακύπτουν χωρίς τη χρήση ταξινομητή. Πρακτικά πολύ μικρή πιθανότητα επιτυχίας ως είσοδο στον αλγόριθμο, θα οδηγήσει στην χρήση σημείων

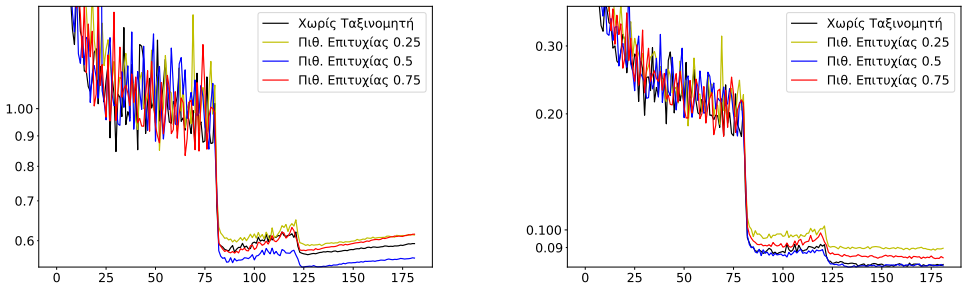
από αραιές συστάδες περιστασιακά με αποτέλεσμα το μοντέλο να μην μπορεί να μάθει την πληροφορία που περιλαμβάνεται σε αυτές. Γενικά, στο συγκεκριμένο παράδειγμα, δεν φαίνεται να υπάρχει κάποιο εμφανές όφελος από τη χρήση του ταξινομητή σε αντίθεση με ότι θα δούμε σε επομένες σειρές πειραματικών αξιολογήσεων.

ResNet-56v1



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.12: CIFAR-10 με ResNet-56v1 : Εξέλιξη Μετρικών Εκπαίδευσης



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.13: CIFAR-10 με ResNet-56v1 : Εξέλιξη Μετρικών Αξιολόγησης

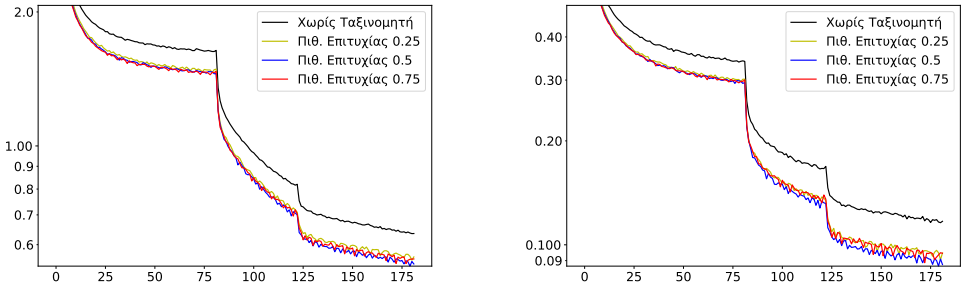
Στην Εικόνα 6.12 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.12α′) και του σφάλματος πρόβλεψης (Εικόνα 6.12β′) κατά την εκπαίδευση του δικτύου ResNet-56v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.13. Η αυξημένη πολυπλοκότητα του δικτύου ResNet-56v1 σε σχέση με αυτήν του δικτύου ResNet-20v1 φαίνεται να ωφελεί τη χρήση του ταξινομήτη σε σχέση με μία τυχαία προσπέλαση στα δεδομένα. Με εξαίρεση τη χρήση ταξινομητή με μικρή τιμή της πιθανότητας επιτυχίας, γίνεται φανερό ότι ο ταξινομητής μπορεί να οδηγήσει σε μικρότερη τιμή της συνάρτησης κόστους στο σύνολο εκπαίδευσης σε σχέση με τον κλασσικό τρόπο εκπαίδευσης, στο τέλος της εκμάθησης του μοντέλου. Ωστόσο, το πιο σημαντικό είναι ότι χρησιμοποιώντας μία ενδιάμεση πιθανότητα επιτυχίας (0.5),

παρατηρείται περαιτέρω μείωση της τάξης του 5.5% στην συνάρτηση κόστους στο σύνολο αξιολόγησης και μια μικρή βελτίωση στο σφάλμα πρόβλεψης, υποδεικνύοντας έτσι ένα πιο σταθερό και γενικό μοντέλο σε σχέση με μοντέλο που προέκυψε από τυχαία προσπέλαση των δεδομένων. Αντίθετα, στην περίπτωση που χρησιμοποιείται μεγάλη πιθανότητα επιτυχίας (0.75) τα δεδομένα από τις αραιές συστάδες χρησιμοποιούνται συχνότερα από το επιθυμητό, δίνοντας μεν βελτίωση στις μετρικές που αφορούν το σύνολο εκπαίδευσης, άλλα επιδείνωση σε αυτές που αφορούν το σύνολο αξιολόγησης, υποδεικνύοντας έτσι πιθανή ύπαρξη φαινομένων υπερπροσαρμογής.

6.3.2 CIFAR-100

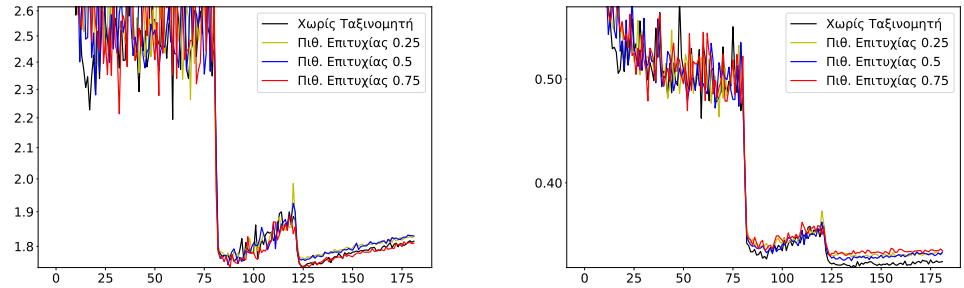
Στη συνέχεια, παρουσιάζονται τα πειραματικά αποτελέσματα για το πιο σύνολο δεδομένων CIFAR-100, το οποίο αποτελεί μια πολύπλοκη περίπτωση από το CIFAR-10. Ξανα, για καλύτερη επεξέγηση των αποτελεσμάτων παρουσιάζονται πρώτα τα αποτελέσματα με εκπαίδευση του δικτύου ResNet-20v1 και στην συνέχεια αυτά που προκύπτουν από την εκπαίδευση του ResNet-56v1.

ResNet-20v1



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.14: CIFAR-100 με ResNet-20v1 : Εξέλιξη Μετρικών Εκπαίδευσης

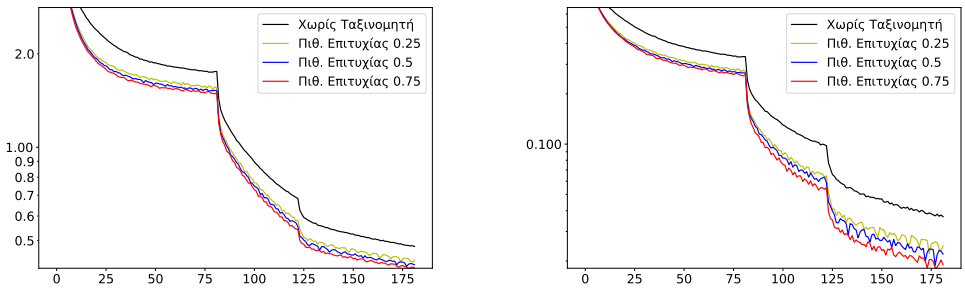


(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.15: CIFAR-100 με ResNet-20v1 : Εξέλιξη Μετρικών Αξιολόγησης

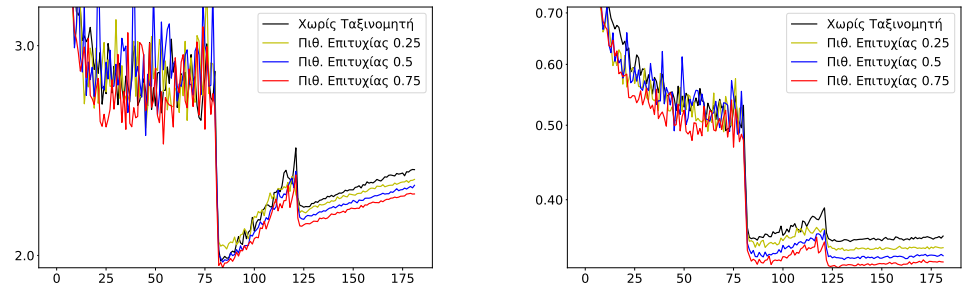
Στην Εικόνα 6.14 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.14α′) και του σφάλματος πρόβλεψης (Εικόνα 6.14β′) κατά την εκπαίδευση του δικτύου ResNet-20v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.15. Παρότι το μοντέλο φαίνεται να οδηγείται σε βελτιστοποίηση των τελικών τιμών των μετρικών εκπαίδευσης με τη χρήση ταξινομήτη, γίνεται φανερό ότι ο ταξινομητής δεν βελτιώνει περαιτέρω τις μετρικές του συνόλου αξιολόγησης. Αυτή η ταυτόχρονη βελτιστοποίηση στις μετρικές εκπαίδευσης συνοδευόμενη από αδράνεια στις μετρικές αξιολόγησεις, υποδεικνύει ότι η σχετική απλή μορφή του νευρωνικού που χρησιμοποιείται δεν μπορεί να γενικεύσει περαίτερω το τελικό μοντέλο, αλλά ξεκινάει να υπερ-προσαρμόζεται πάνω στα δεδομένα εκπαίδευσης.

ResNet-56v1



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.16: CIFAR-100 με ResNet-56v1 : Εξέλιξη Μετρικών Εκπαίδευσης



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.17: CIFAR-100 με ResNet-56v1 : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.16 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.16α′) και του σφάλματος πρόβλεψης (Εικόνα 6.16β′) κατά την εκπαίδευση του δικτύου

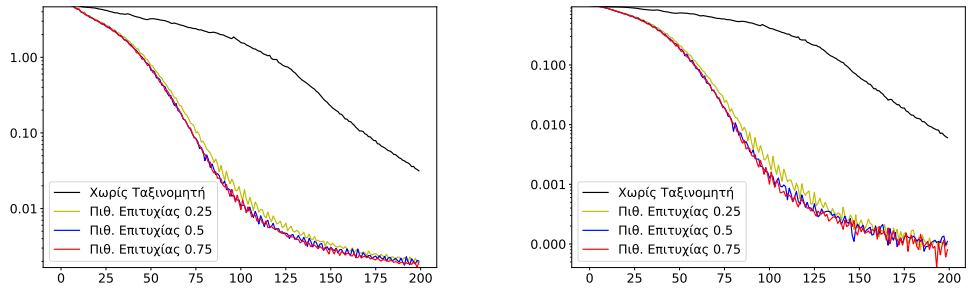
ResNet-56v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.17. Τα συγκεκριμένα πειράματα παρουσιάζουν αρκετό ενδιαφέρον, καθώς η χρήση του ταξινομητή φαίνεται να ευνοεί την εκπαίδευση ανεξαρτήτως της τιμής που θα χρησιμοποιηθεί για την πιθανότητα επιτυχίας. Ωστόσο, ότι αύξηση της πιθανότητας επιτυχίας βελτιώνει περαιτέρω το μοντέλο. Η αύξηση της πολυπλοκότητας του νευρωνικού φαίνεται να μπορεί να γενικεύσει καλύτερα χρησιμοποιώντας την επαυξημένη πληροφορία από τα επαναλαμβανόμενα δεδομένα. Στον Πίνακα 6.7 παρουσιάζεται το ποσοστό βελτίωσης της κάθε μετρικής με χρήση του ταξινομητή, για κάθε μία από τις τιμές της πιθανότητας επιτυχίας που δοκιμάστηκαν.

Πίνακας 6.7: Ποσοστά βελτίωσης της εκπαίδευσης με χρήση ταξινομητή.

Πιθανότητα Επιτυχίας Ταξινομητή	Μετρικές Εκπαίδευσης		Μετρικές Αξιολόγησης	
	Κόστος	Σφάλμα	Κόστος	Σφάλμα
0.25	9.72%	1.26%	1.86%	1.91%
0.5	12.71%	1.53%	2.97%	3.19%
0.75	14.84%	1.84%	4.58%	4.17%

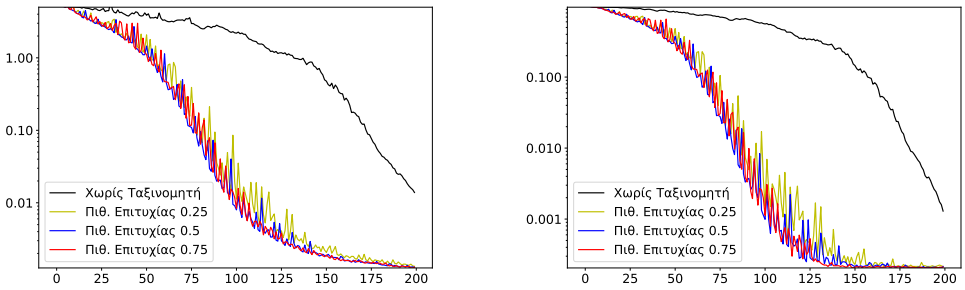
6.3.3 Tiny - Imagenet

Το τελευταίο κομμάτι της ποιοτικής αξιολόγησης των μοντέλων που προκύπτουν με χρήση του βελτιστοποιητή αποτελεί το μοντέλο Inception-V3 εκπαιδευμένο πάνω στο σύνολο δεδομένων Tiny - Imagenet.



(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.18: Tiny - Imagenet : Εξέλιξη Μετρικών Εκπαίδευσης



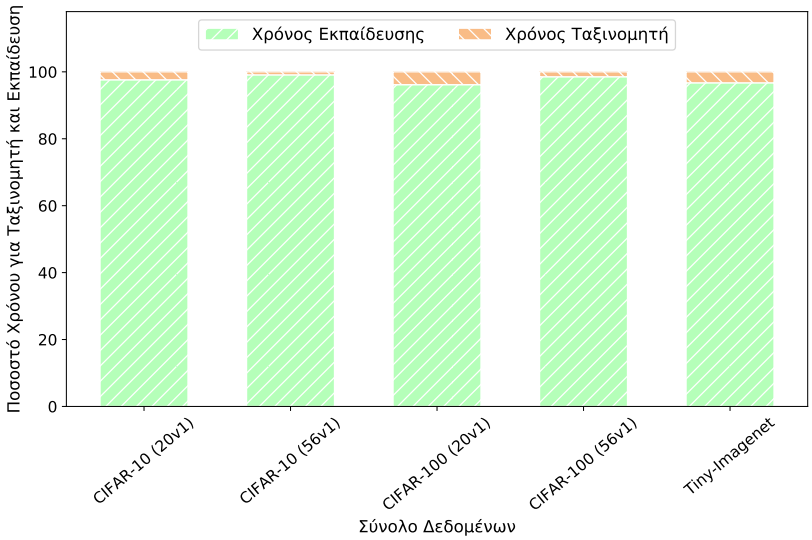
(α′) Συνάρτηση Κόστους (β′) Σφάλμα Πρόβλεψης

Εικόνα 6.19: Tiny - Imagenet : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.18 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.18α′) και του σφάλματος πρόβλεψης (Εικόνα 6.18β′) κατά την εκπαίδευση του δικτύου Inception-V3. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.19. Στο συγκεκριμένο παράδειγμα, φαίνεται καθαρά η βελτίωση που εισάγεται με τη χρήση του ταξινομητή. Αρχικά, είναι φανερό πως τόσο οι μετρικές εκπαίδευσης όσο και οι μετρικές αξιολόγησης προσεγγίζουν το 0 κατά μία τάξη μεγέθους σε σχέση με την εκπαίδευσης με τυχαία προσπέλαση των δεδομένων. Επομένως, η χρήση του ταξινομητή οδηγεί σε ένα καλύτερο μοντέλο. Ενδεικτικά, το σφάλμα πρόβλεψης για το σύνολο αξιολόγησης από την τιμή του 99.87% μπορεί να φτάσει την τιμή 99.98%. Επομένως, προκύπτει ένα μοντέλο με σχεδόν απόλυτη ακρίβεια πρόβλεψης.

Παράλληλα, αξίζει να αναφερθεί ότι, ακόμα και εάν δεν μας ενδιέφερε η περαιτέρω βελτιστοποίηση του μοντέλου, μπορούμε να πετύχουμε το ίδιο κόστος και σφάλμα στο σύνολο αξιολόγησης με την βασική μέθοδο στις μισές περίπου εποχές εκπαίδευσης και επομένως και στο μισό περίπου χρόνο, ανεξαρτήτως της παραμέτρου που αφορά τον χειρισμό των αραιών συστάδων που προκύπτουν από τον αναλυτή. Επομένως, σε πιο πολύπλοκα σύνολα δεδομένων και νευρωνικά δίκτυα, όπως είναι αυτά που εξετάζονται εδώ, είναι φανερό πως η χρήση του ταξινομητή μπορεί να έχει πολύπλευρα οφέλη για την διαδικασία της εκπαίδευσης.

6.3.4 Επιβάρυνση Χρόνου από τη Χρήση του Ταξινομητή



Εικόνα 6.20: Ποσοστό χρόνου εκπαίδευσης και ταξινόμησης για κάθε συνδυασμό συνόλου δεδομένων και νευρωνικού δικτύου που δοκιμάστηκε.

Στην Εικόνα 6.20 δίνεται το ποσοστό του χρόνου που καταλαμβάνει η διαδικασία της ταξινόμησης και την εκπαίδευσης σε περίπτωση που χρησιμοποιείται ο ταξινομητής για κάθε έναν συνδυασμό συνόλου δεδομένων και νευρωνικού δικτύου που χρησιμοποιήθηκαν στην παραπάνω πειραματική αξιολόγηση. Όπως είναι φανερό, σε κάθε περίπτωση ο χρόνος ταξινόμησης λαμβάνει λιγότερο από το 5% του συνολικού χρόνου. Αυτό πρακτικά σημαίνει ότι ο χρόνος είναι αμελητέος σε σχέση με τον χρόνο που απαιτείται για να πραγματοποιηθεί η εκπαίδευση του μοντέλου. Επομένως, η χρήση του ταξινομητή δεν είναι απαγορευτική. Αντιθέτως, ο ταξινομητής μπορεί να χρησιμοποιείται στη γενική περίπτωση πριν την εκπαίδευση καθώς όπως διαπιστώθηκε στον σχολιασμό των πειραμάτων των προηγούμενων υποενοτήτων είτε θα αφήσει στάσιμες τις μετρικές του μοντέλου είτε θα τις βελτιστοποίησει περισσότερο.

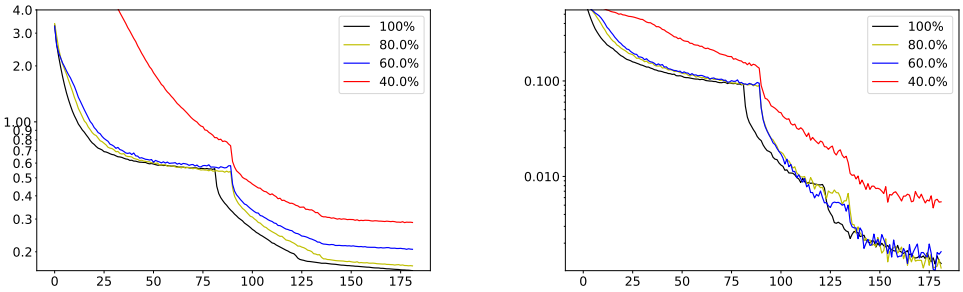
6.4 Αξιολόγηση Χρήσης Δειγματολήπτη Δεδομένων

Στην παρούσα ενότητα γίνεται παρουσίασή και σχολιασμός των πειραματικών αποτελεσμάτων που προέκυψαν από την εφαρμογή του δειγματολήπτη πάνω στα τρία σύνολα δεδομένων που παρουσιάστηκαν στην Ενότητα 6.1. Για κάθε ένα από τα σύνολα δεδομένων που εξετάζονται, σχολιάζεται τόσο η ποιότητα των μετρικών που

αξιολογούν την προβλεπτική ικανότητα του μοντέλου ενώ παράλληλα παρουσιάζεται και η σχετική επιτάχυνση που εισάγει η χρήση ενός μικρότερου δείγματος. Σε αντίθεση με τα πειράματα της ενότητας 6.3, τα σύνολα δεδομένων CIFAR εξετάζονται μόνο με το νευρωνικό δίκτυο ResNet-56v1. Σε κάθε σύνολο εξετάζονται οι περιπτώσεις χρήσης ολόκληρου του συνόλου δεδομένων και του 80%, 60% και 40% των δεδομένων του.

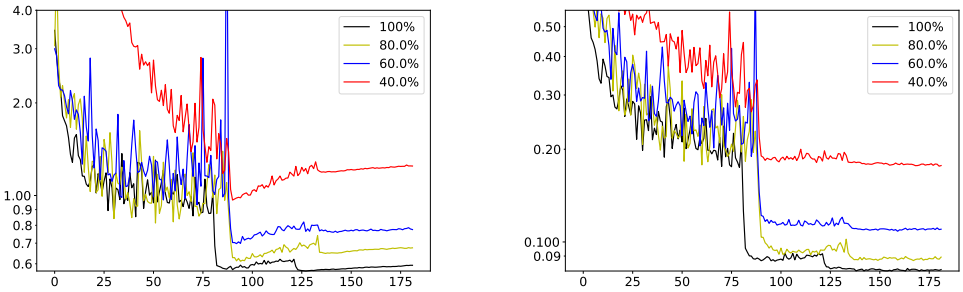
6.4.1 CIFAR-10

Όπως και στην προηγούμενη σειρά πειραμάτων, το πειραματικό μέρος που αφορά τον δειγματολήπτη ξεκινάει με το σύνολο δεδομένων CIFAR-10, το οποίο χρησιμοποιείται για την εκπαίδευση ενός δικτύου ResNet-56v1.



(α′) Συνάρτηση Κόστους (β′) Σφάλμα

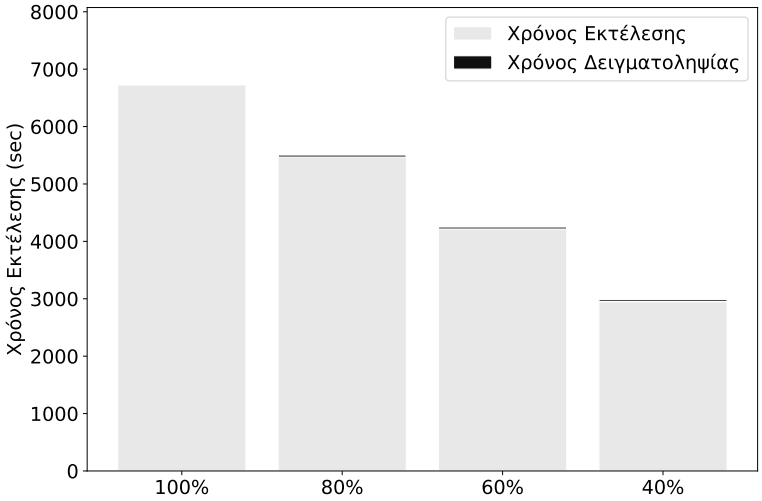
Εικόνα 6.21: Διάφορα Ποσοστά Χρήσης CIFAR-10 : Εξέλιξη Μετρικών Εκπαίδευσης



(α′) Συνάρτηση Κόστους (β′) Σφάλμα

Εικόνα 6.22: Διάφορα Ποσοστά Χρήσης CIFAR-10 : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.21 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.21α′) και του σφάλματος πρόβλεψης (Εικόνα 6.22α′) κατά την εκπαίδευση του δικτύου ResNet-56v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.22. Χρησιμοποιώντας το 40% του δείγματος γίνεται φανερό ότι δεν είναι εφικτό καν να προσεγγίσει το 0 η συνάρτηση κόστους πάνω στο σύνολο εκπαίδευσης, γεγονός που υποδεικνύει ότι δείγμα τόσο μικρού μεγέθους δεν είναι αρκετό ώστε να εκπαιδεύσει επαρκώς το μοντέλο. Ωστόσο, στα δείγματα που αποτελούν το 60% και το 80% το μοντέλο φαίνεται να εκπαιδεύεται επαρκώς όσον αφορά τις μετρικές εκπαίδευσης, καθώς το σφάλμα εκπαίδευσης φαίνεται να ακολουθεί την πορεία του αντίστοιχου που προκύπτει από εκπαίδευση με όλο το σύνολο δεδομένων. Συγκεκριμένα στην περίπτωση που έχει επιλεγεί το 80% του συνόλου δεδομένων βλέπουμε ότι το κόστος εκπαίδευσης είναι μόλις $1.01x$ φορές μικρότερο. Το πιο ενδιαφέρον είναι ότι παρόμοια συμπεριφορά επιδεικνύει το μοντέλο και στις μετρικές αξιολόγησης, όπου το κόστος αξιολόγησης υστερεί μόλις κατά $1.1x$ ενώ το σφάλμα είναι μεγαλύτερο μόλις κατά 0.7%.

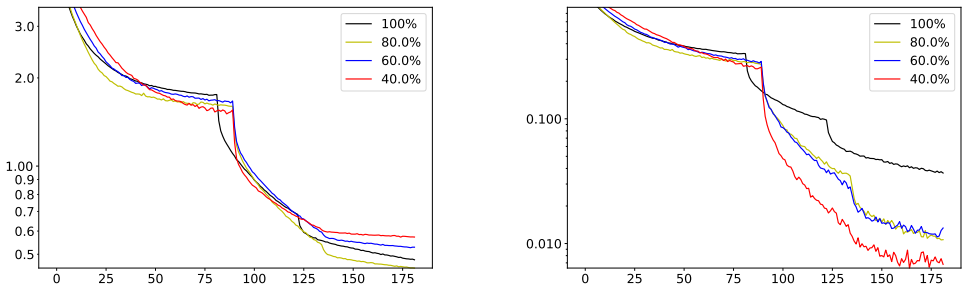


Εικόνα 6.23: Χρόνος Εκτέλεσης διαδικασίας εκπαίδευσης και δειγματοληψίας για διάφορα ποσοστά χρήσης του συνόλου δεδομένων CIFAR-10.

Στην Εικόνα 6.23 δίνονται οι χρόνοι εκτέλεσης, συμπεριλαμβανομένου του χρόνου δειγματοληψίας, για την εκπαίδευση του δικτύου με κάθε ένα υποσύνολο του CIFAR-10. Όπως ήταν αναμενόμενο, όσο μειώνεται το μέγεθος του δείγματος τόσο μειώνεται και ο συνολικός χρόνος που σχετίζεται με την εκπαίδευση. Επιπλέον αξίζει να σημειωθεί ότι ο χρόνος δειγματοληψίας είναι πρακτικά αμελητέος σε σχέση με το χρόνο εκπαίδευσης του δικτύου. Εστιάζοντας στην περίπτωση του δείγματος 80%, το μοντέλο έχει εκπαιδευτεί κατά $1.22X$ γρηγορότερα, με μόλις ελάχιστες αποκλίσεις σε ποιοτικά χαρακτηριστικά, όπως διαπιστώθηκε παραπάνω.

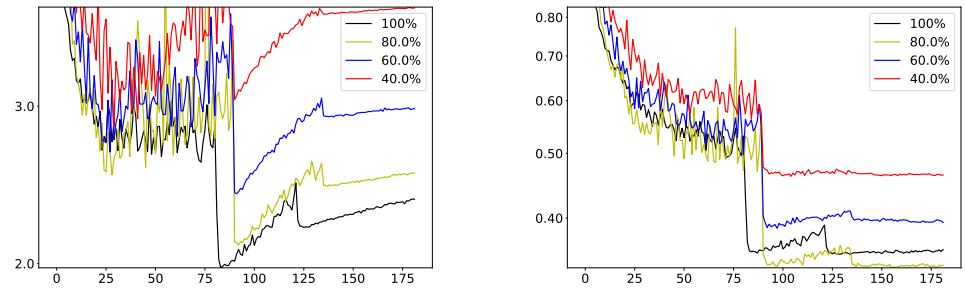
6.4.2 CIFAR-100

Η μελέτη της εκπαίδευσης μέσω ενός δείγματος συνεχίζεται με το σύνολο δεδομένων CIFAR-100 και το ResNet-56v1 ως νευρωνικό δίκτυο.



(α′) Συνάρτηση Κόστους (β′) Σφάλμα

Εικόνα 6.24: Διάφορα Ποσοστά Χρήσης CIFAR-100 : Εξέλιξη Μετρικών Εκπαίδευσης

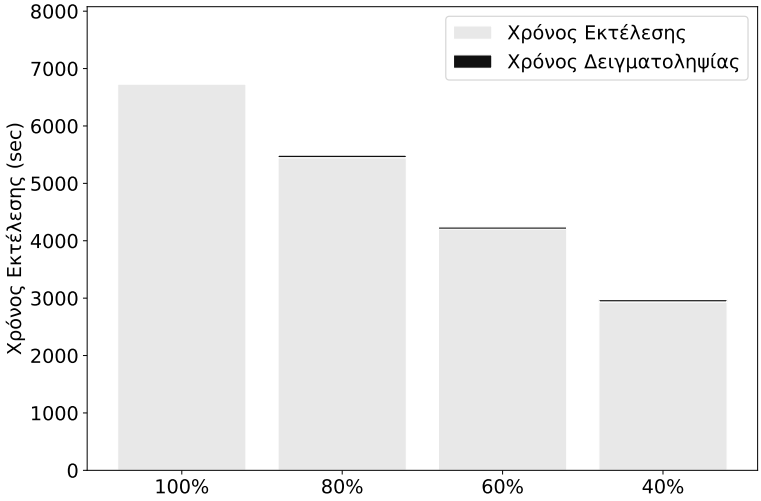


(α′) Συνάρτηση Κόστους (β′) Σφάλμα

Εικόνα 6.25: Διάφορα Ποσοστά Χρήσης CIFAR-100 : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.24 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.24α′) και του σφάλματος πρόβλεψης (Εικόνα 6.25α′) κατά την εκπαίδευση του δικτύου ResNet-56v1. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.25. Σχετικά με το σφάλμα εκπαίδευσης, η Εικόνα 6.21β′ υποδεικνύει ότι μείωση του μεγέθους του δείγματος που λαμβάνεται υπόψιν το μοντέλο στην διαδικασία εκπαίδευσης οδηγεί σε μείωση την τιμή του σφάλματος εκπαίδευσης. Ωστόσο, αυτό το γεγονός είναι ένδειξη υπερ-προσαρμογής στα δεδομένα, καθώς ακόμα και στην περίπτωση του κόστους εκπαίδευσης, αυτό βελτιώνεται σε σχέση με την περίπτωση χρήσης όλου του συνόλου δεδομένων μόνο όταν χρησιμοποιείται το 80% των

δεδομένων. Πράγματι μελετώντας και τις μετρικές αξιολόγησεις γίνεται φανερό ότι χρησιμοποιώντας για παράδειγμα το 40% των δεδομένων η τιμή του σφάλματος αξιολόγησης αυξάνεται από περίπου 35% σε 50%, ενώ η τιμή της συνάρτησης κόστους στο σύνολο επικύρωσης σχεδόν διπλασίαζεται. Ωστόσο, στην περίπτωση που χρησιμοποιηθεί ως δείγμα το 80% των δεδομένων, η τιμή της συνάρτησης κόστους αυξάνεται μόλις κατά $1.06X$ ενώ η τιμή του σφάλματος μειώνεται κατά 1.88%

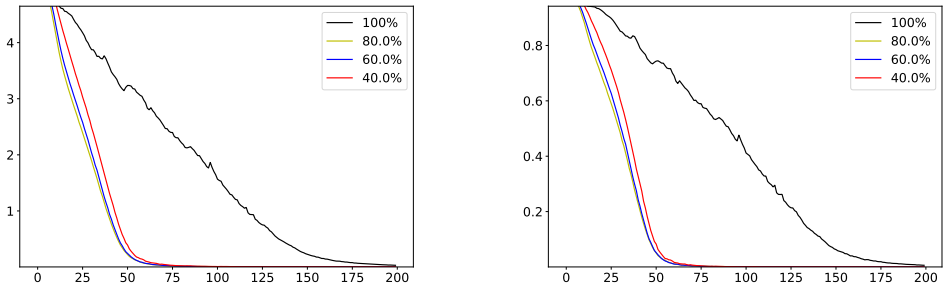


Εικόνα 6.26: Χρόνος Εκτέλεσης διαδικασίας εκπαίδευσης και δειγματοληψίας για διάφορα ποσοστά χρήσης του συνόλου δεδομένων CIFAR-100.

Στην Εικόνα 6.26 δίνονται οι χρόνοι εκτέλεσης, συμπεριλαμβανομένου του χρόνου δειγματοληψίας, για την εκπαίδευση του δικτύου με κάθε ένα υποσύνολο του CIFAR-100. Όπως σχολιάστηκε και παραπάνω, μείωση του δείγματος οδηγεί δεδομένα σε μείωση του συνολικού χρόνου εκτέλεσης. Ωστόσο, η μεγαλύτερη ταχύτητα δεν συνεπάγεται και ένα επιθυμητό μοντέλο σύμφωνα με την ανάλυση των ποιοτικών χαρακτηριστικών του μοντέλου αναφορικά με το CIFAR-100. Το σημαντικό είναι ότι η αρχική διαδικασία ανάλυσης και δειγματοληψίας έχει αμελητέο χρόνο σε σχέση με το σύνολο του χρόνου εκπαίδευσης. Έτσι σύμφωνα με όλο το σχολιασμό που έχει προηγηθεί και πάλι καλή προσέγγιση θα ήταν η επιλογή του 80% των δεδομένων για την εκπαίδευση του δικτύου, που θα οδηγούσε και σε βελτίωση του σφάλματος αξιολόγησης με ελάχιστη επιδείνωση της τιμής της συνάρτησης κόστους για το σύνολο επικύρωσης.

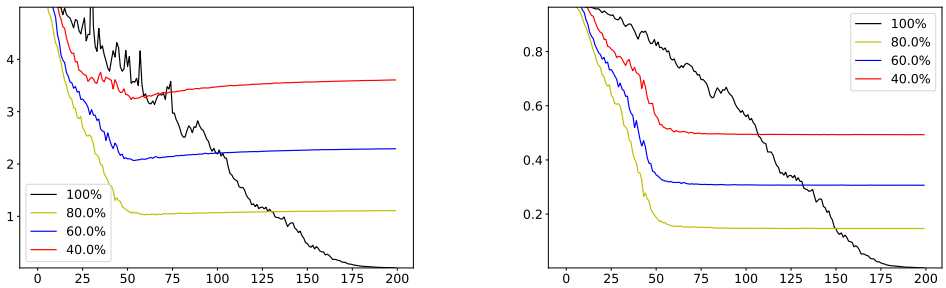
6.4.3 Tiny - Imagenet

Τελευταίο μέρος της πειραματικής αξιολόγησης αποτελεί η μελέτη της εκπαίδευσης του δικτύου Inception-V3 με υποσύνολα που έχουν προκύψει από τον δειγματολήπτη για το σύνολο δεδομένων Tiny - Imagenet.



(α′) Συνάρτηση Κόστους (β′) Σφάλμα

Εικόνα 6.27: Διάφορα Ποσοστά Χρήσης Tiny - Imagenet : Εξέλιξη Μετρικών Εκπαίδευσης

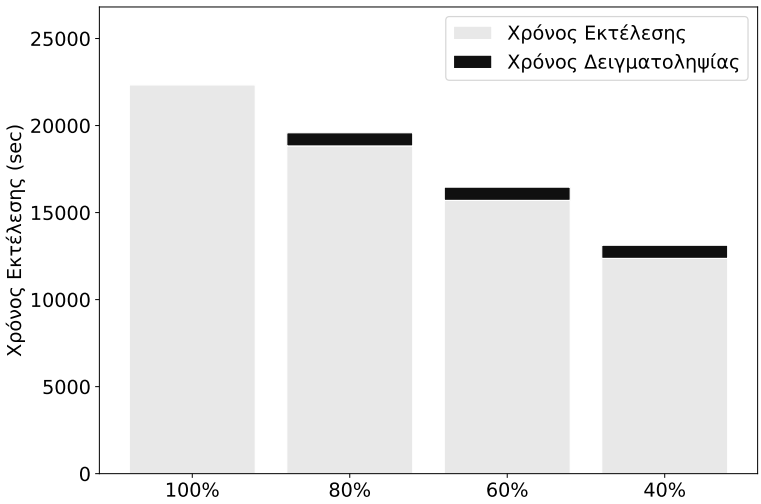


(α′) Συνάρτηση Κόστους (β′) Σφάλμα

Εικόνα 6.28: Διάφορα Ποσοστά Χρήσης Tiny - Imagenet : Εξέλιξη Μετρικών Αξιολόγησης

Στην Εικόνα 6.27 παρουσιάζεται η εξέλιξη της συνάρτησης κόστους (Εικόνα 6.27α′) και του σφάλματος πρόβλεψης (Εικόνα 6.28α′) κατά την εκπαίδευση του δικτύου Inception-V3. Η εξέλιξη των αντίστοιχων μετρικών αξιολόγησης δίνεται στην Εικόνα 6.28. Παρατηρώντας τις μετρικές εκπαίδευσης γίνεται φανερό ότι αυτές μειώνονται πολύ γρήγορα σε σχέση με την περίπτωση χρήσης ολόκληρου του συνόλου δεδομένων. Η ίδια συμπεριφορά παρατηρείται και στις μετρικές αξιολόγησης, οι οποίες μειώνονται και αυτές γρήγορα και μετά σταθεροποιούνται αφού το δίκτυο δεν μπορεί

να εκπαιδευτεί περαιτέρω. Συγκερκιμένα, επειδή το δίκτυο Inception-V3 είναι αρκετά περίπλοκο, θα μάθει γρήγορα να προσαρμοστεί πάνω σε απλά και λίγα δεδομένα, χωρίς όμως να μπορεί να γενικεύσει απαραίτητα εξίσου καλά με την ύπαρξη περισσότερων δεδομένων. Ενώ από την Εικόνα 6.28α΄ φαίνεται ότι χρησιμοποιώντας ολόκληρο το σύνολο δεδομένων η συνάρτηση κόστους σχεδόν μηδενίζεται στο σύνολο αξιολόγησης, δεν παρατηρείται η ίδια συμπεριφορά όταν έχουμε υποσύνολο δεδομένων για εκπαίδευση, ανεξαρτήτως το ποσοστό του συνόλου που θα διατηρηθεί. Στην καλύτερη περίπτωση γίνεται φανερό ότι με χρήση του 80% υποσυνόλου η συνάρτηση κόστους για το σύνολο αξιολόγησης μπορεί να ελαχιστοποιηθεί ως την τιμή 1.1 πετυχαίνοντας ένα ασθενέστερο σφάλμα αξιολόγησης της τάξης του 15%. Επίσης, δεδομένου ότι περίπου από την εποχή 50 και μετά δεν μειώνεται περαιτέρω το κόστος στο σύνολο αξιολόγησης, τεχνικές early - stopping θα σταματούσαν την εκπαίδευση πολύ νωρίτερα και θα δημιουργούσαν ένα μοντέλο σε σημαντικά μικρότερο χρονικό διάστημα.



Εικόνα 6.29: Χρόνος Εκτέλεσης διαδικασίας εκπαίδευσης και δειγματοληψίας για διάφορα ποσοστά χρήσης του συνόλου δεδομένων Tiny - Imagenet.

Συνέχιζοντας τον σχολιασμό γύρω από τον χρόνο εκτέλεσης, στην Εικόνα 6.29 δίνεται ο συνολικός χρόνος εκτέλεσης του κάθε πειράματος (χρόνος δειγματοληψίας και χρόνος εκπαίδευσης). Φυσικά η μείωση του μεγέθους του δείγματος, μειώνει το συνολικό χρόνο όπως έχει ήδη σχολιαστεί. Αναφορικά με τη χρήση του 80% του συνόλου εκπαίδευσης που παρατηρήθηκε ότι έχει τη μικρότερη δυνατή απώλεια στην απόδοση του μοντέλου, φαίνεται ότι η διαδικασία μπορεί να ολοκληρωθεί κατά $1.14X$ πιο γρήγορα. Δεδομένου ότι το μοντέλο αν εκπαιδευτεί με το δείγμα θα έχει αυτή την λιγότερο καλή ικανότητα γενίκευσης, εάν χρησιμοποιούταν επιπλέον τεχνική early - stopping, τότε θα μπορούσε προσεγγιστικά να σταμάτησει η εκπαίδευση μετά από περίπου 60 εποχές. Στην περίπτωση αυτή, αν ο χρήστης επιθυμούσε δηλαδή ένα ασθενέστερο μοντέλο, το οποίο όμως θα είχε εκπαιδευτεί πολύ γρηγορότερα, θα ολοκληρωνόταν η διαδικασία εκπαίδευσης κατά περίπου $3.5X$ πιο γρήγορα!

Κεφάλαιο 7

Συμπεράσματα και Επεκτάσεις

Στην συγκεκριμένη εργασία μελετήθηκαν τεχνικές εκμετάλλευσης της κατανομής που ορίζει ένα σύνολο δεδομένων εκπαίδευσης προς όφελος της εκπαίδευσης νευρωνικών δικτύων. Η εργασία κινήθηκε σε δύο άξονες: στον καθορισμό της σειράς πρόσβασης στα δεδομένα κατά την εκπαίδευση μέσω ενός ταξινομητή και στην επιτάχυνση της εκπαίδευσης ενός νευρωνικού δικτύου χρησιμοποιώντας συστηματικά κατασκευασμένα υποσύνολα του συνόλου εκπαίδευσης. Και οι δύο αλγόριθμοι αξιοποιούν την πληροφορία ενός αναλυτή δεδομένων που προηγείται, ο οποίος χωρίζει τα δεδομένα σε ομάδες που θεωρείται ότι προσομοιώνουν δεδομένα κάποιας κανονικής κατανομής.

Αναφορικά με τη χρήση του ταξινομητή, παρατηρείται ότι μπορεί να βελτιώσει της ικανότητα γενίκευσης του μοντέλου, μειώνοντας το σφάλμα αξιολόγησης του. Για παράδειγμα, στην περίπτωση του συνόλου δεδομένων CIFAR-100 οι μετρικές αξιολόγησης μειώνονται έως και 5%, ενώ στην περίπτωση του Tiny - Imagenet επιτυγχάνεται ο μηδενισμός. Ωστόσο, σε καμία περίπτωση, έχοντας ρυθμίσει κατάλληλα τις παραμέτρους εισόδου του ταξινομητή, το εξαγόμενο μοντέλο θα έχει στην χειρότερη περίπτωση ίδια ικανότητα γενίκευσης με το μοντέλο που προκύπτει αν δεν γίνει χρήση του ταξινομητή.

Σχετικά με τον δειγματολήπτη των δεδομένων, παρατηρείται ότι χρησιμοποιώντας τα αποτελέσματα του αναλυτή μπορεί να εκπαιδεύσει ταχύτερα ένα μοντέλο χωρίς ιδιαίτερα μεγάλες απώλειες πληροφορίες. Παρατηρήθηκε ότι η χρήση ενός δείγματος με το 80% των δεδομένων επιφέρει τα καλύτερα αποτελέσματα. Συγκεκριμένα, στην περίπτωση του CIFAR-10 πετυχαίνει μοντέλο αντίστοιχης προβλεπτικής ικανότητας σε $1.22X$ φορές λιγότερο χρόνο από το να χρησιμοποιούσε ολόκληρο το σύνολο δεδομένων. Στην περίπτωση του CIFAR-100, το μοντέλο που δίνει το υποσύνολο των δεδομένων διατηρεί σταθερό το κόστος αξιολόγησης, βελτιώνει κατά 1.88% την τιμή του σφάλματος αξιολόγησης, ενώ ολοκληρώνεται η εκπαίδευση του και πάλι περίπου $1.2X$ φορές ταχύτερα. Τέλος, στην περίπτωση του Ιμαγενετ, η εκπαίδευση μέσω δείγματος αντί για όλου του συνόλου δεδομένων δεν μπορεί να φτάσει την απόδοση του αρχικού μοντέλου, αλλά σε $3.5X$ φορές λιγότερο χρόνο, μπορεί να παράγει ένα μοντέλο με σφάλμα αξιολόγησης μικρότερο κατά 15%.

Συνοπτικά, τα συμπεράσματα που ανακύπτουν είναι τα ακόλουθα:

- Σε νευρωνικά δίκτυα μεγαλύτερης πολυπλοκότητας, η χρήση του ταξινομητή στα δεδομένα φαίνεται να επιφέρει καλύτερα αποτελέσματα ως προς τις μετρικές αξιολόγησης του μοντέλου που προκύπτει.
- Αυξάνοντας την πολυπλοκότητα του συνόλου δεδομένων, η χρήση του ταξινομητή και πάλι φαίνεται να ωφελεί τη διαδικασία της εκπαίδευσης.
- Στις περισσότερες περιπτώσεις, χρησιμοποιώντας το 80% του συνόλου εκπαίδευσης μπορεί να κατασκευαστεί ένα αντίστοιχο μοντέλο σε λιγότερο χρόνο.

Η παρουσα διπλωματική εργασία θα μπορούσε να επεκταθεί στους ακόλουθους άξονες:

- Να πραγματοποιηθεί μελέτη για το πως οι παράμετροι του αναλυτή μπορούν να επηρεάσουν περαιτέρω την λειτουργία του ταξινομητή και του δειγματολήπτη.
- Να μελετηθούν άλλες τεχνικές δειγματοληψίας και διάταξης των δεδομένων.
- Να γίνει πιο εκτεταμένη πειραματική ανάλυση χρησιμοποιώντας περισσότερα, μεγαλύτερα και πιο πολύπλοκα σύνολα δεδομένων εκπαίδευσης.

Βιβλιογραφία

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. «ImageNet Classification with Deep Convolutional Neural Networks». In: *Commun. ACM* 60.6 (2017), 84–90. ISSN: 0001-0782. DOI: 10.1145/3065386. URL: https://doi.org/10.1145/3065386.

[2] Alex Graves, Abdel-rahman Mohamed, and Geoffrey Hinton. «Speech recognition with deep recurrent neural networks». In: *2013 IEEE international conference on acoustics, speech and signal processing*. IEEE. 2013, pp. 6645–6649.

[3] Kaiming He et al. «Deep residual learning for image recognition». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[4] Valerie Sessions and Marco Valtorta. «The Effects of Data Quality on Machine Learning Algorithms.» In: *ICIQ* 6 (2006), pp. 485–498.

[5] Jinseok Kim and Jenna Kim. «The impact of imbalanced training data on machine learning for author name disambiguation». In: *Scientometrics* 117.1 (2018), pp. 511–526.

[6] Yoshua Bengio. *Practical recommendations for gradient-based training of deep architectures*. 2012. arXiv: 1206.5533 [cs.LG].

[7] Ding-Zhu Du, Panos M Pardalos, and Weili Wu. *Mathematical theory of optimization*. Vol. 56. Springer Science & Business Media, 2013.

[8] Giorgio Ausiello, Alessandro D'Atri, and Marco Protasi. «Structure preserving reductions among convex optimization problems». In: *Journal of Computer and System Sciences* 21.1 (1980), pp. 136–153.

[9] M. F. Tasgetiren and P. N. Suganthan. «A Multi-Populated Differential Evolution Algorithm for Solving Constrained Optimization Problem». In: *2006 IEEE International Conference on Evolutionary Computation*. 2006, pp. 33–40.

[10] Kenji Kawaguchi. «Deep learning without poor local minima». In: *Advances in neural information processing systems*. 2016, pp. 586–594.

[11] Sebastian Ruder. «An overview of gradient descent optimization algorithms». In: *arXiv preprint arXiv:1609.04747* (2016).

[12] *small learning rate vs big learning rate*. URL: https://stackoverflow.com/questions/62690725/small-learning-rate-vs-big-learning-rate.

[13] Christian Darken, Joseph Chang, John Moody, et al. «Learning rate schedules for faster stochastic gradient search». In: *Neural networks for signal processing*. Vol. 2. Citeseer. 1992.

[14] Christian Darken and John E Moody. «Note on learning rate schedules for stochastic optimization». In: *Advances in neural information processing systems*. 1991, pp. 832–838.

[15] George D. Magoulas, Michael N. Vrahatis, and George S Androulakis. «Improving the convergence of the backpropagation algorithm using learning rate adaptation methods». In: *Neural Computation* 11.7 (1999), pp. 1769–1796.

[16] Rong Ge et al. «Rethinking learning rate schedules for stochastic optimization». In: (2018).

[17] Muhammad Rizwan. *Mini-batch Gradient Descent for Deep Learning*. 2018. URL: https://engmrk.com/mini-batch-gd/.

[18] Mu Li et al. «Efficient Mini-Batch Training for Stochastic Optimization». In: KDD '14. Association for Computing Machinery, 2014, 661–670. ISBN: 9781450329569. DOI: 10.1145/2623330.2623612. URL: https://doi.org/10.1145/2623330.2623612.

[19] Z^2 Little. *Gradient Descent: Stochastic vs. Mini-batch vs. Batch vs. AdaGrad vs. RMSProp vs. Adam*. 2020. URL: https://medium.com/@xzz201920/gradient-descent-stochastic-vs-mini-batch-vs-batch-vs-adagrad-vs-rmsprop-vs-adam-3aa652318b0d.

[20] Ning Qian. «On the Momentum Term in Gradient Descent Learning Algorithms». In: *Neural Netw.* 12.1 (1999), 145–151. ISSN: 0893-6080. DOI: 10.1016/S0893-6080(98)00116-6. URL: https://doi.org/10.1016/S0893-6080(98)00116-6.

[21] Ilya Sutskever et al. «On the Importance of Initialization and Momentum in Deep Learning». In: *Proceedings of the 30th International Conference on International Conference on Machine Learning - Volume 28*. ICML'13. JMLR.org, 2013, III–1139–III–1147.

[22] Yurii E Nesterov. «A method for solving the convex programming problem with convergence rate O (1/kˆ 2)». In: *Dokl. akad. nauk Sssr*. Vol. 269. 1983, pp. 543–547.

[23] John Duchi, Elad Hazan, and Yoram Singer. «Adaptive Subgradient Methods for Online Learning and Stochastic Optimization». In: *Journal of Machine Learning Research* 12.61 (2011), pp. 2121–2159. URL: http://jmlr.org/papers/v12/duchi11a.html.

[24] Diederik P. Kingma and Jimmy Ba. «Adam: A Method for Stochastic Optimization». In: *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*. Ed. by Yoshua Bengio and Yann LeCun. 2015. URL: http://arxiv.org/abs/1412.6980.

[25] Xavier Glorot and Yoshua Bengio. «Understanding the difficulty of training deep feedforward neural networks». In: *Proceedings of the thirteenth international conference on artificial intelligence and statistics*. 2010, pp. 249–256.

[26] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. http://www.deeplearningbook.org. MIT Press, 2016.

[27] *TensorFlow - Multi-Layer Perceptron Learning*. URL: https://www.tutorialspoint.com/tensorflow/tensorflow_multi_layer_perceptron_learning.htm.

[28] Stephen Marsland. *Machine Learning: An Algorithmic Perspective, Second Edition*. 2nd. Chapman & Hall/CRC, 2014. ISBN: 1466583282.

[29] Jun Han and Claudio Moraga. «The influence of the sigmoid function parameters on the speed of backpropagation learning». In: *International Workshop on Artificial Neural Networks*. Springer. 1995, pp. 195–201.

[30] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. «Deep learning». In: *nature* 521.7553 (2015), pp. 436–444.

[31] Vinod Nair and Geoffrey E Hinton. «Rectified linear units improve restricted boltzmann machines». In: *ICML*. 2010.

[32] Daniel Svozil, Vladimír Kvasnicka, and Jirˆí Pospichal. «Introduction to multi-layer feed-forward neural networks». In: *Chemometrics and Intelligent Laboratory Systems* 39.1 (1997), pp. 43 –62. ISSN: 0169-7439. DOI: https://doi.org/10.1016/S0169-7439(97)00061-0. URL: http://www.sciencedirect.com/science/article/pii/S0169743997000610.

[33] Shie Mannor, Dori Peleg, and Reuven Rubinstein. «The Cross Entropy Method for Classification». In: *Proceedings of the 22nd International Conference on Machine Learning*. ICML '05. Association for Computing Machinery, 2005, 561–568. ISBN: 1595931805. DOI: 10.1145/1102351.1102422. URL: https://doi.org/10.1145/1102351.1102422.

[34] Rich Caruana, Steve Lawrence, and Lee Giles. «Overfitting in Neural Nets: Backpropagation, Conjugate Gradient, and Early Stopping». In: *Proceedings of the 13th International Conference on Neural Information Processing Systems*. NIPS'00. MIT Press, 2000, 381–387.

[35] Hamed Habibi Aghdam and Elnaz Jahani Heravi. *Guide to Convolutional Neural Networks: A Practical Application to Traffic-Sign Detection and Classification*. 1st. Springer Publishing Company, Incorporated, 2017. ISBN: 331957549X.

[36] M.V. Valueva et al. «Application of the residue number system to reduce hardware costs of the convolutional neural network implementation». In: *Mathematics and Computers in Simulation* 177 (2020), pp. 232 –243. ISSN: 0378-4754. DOI: https://doi.org/10.1016/j.matcom.2020.04.031. URL: http://www.sciencedirect.com/science/article/pii/S0378475420301580.

[37] Kunihiko Fukushima and Sei Miyake. «Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition». In: *Competition and cooperation in neural nets*. Springer, 1982, pp. 267–285.

[38] Dan Ciregan, Ueli Meier, and Jürgen Schmidhuber. «Multi-column deep neural networks for image classification». In: *2012 IEEE conference on computer vision and pattern recognition*. IEEE. 2012, pp. 3642–3649.

[39] Yann Lecun, Yoshua Bengio, and Geoffrey Hinton. «Deep learning». In: *Nature* 521.7553 (2015), pp. 436–444. DOI: 10.1038/nature14539.

[40] Kouichi Yamaguchi et al. «A neural network for speaker-independent isolated word recognition». In: *First International Conference on Spoken Language Processing*. 1990.

[41] Cezanne Camacho. *Convolutional Neural Networks*. URL: https://cezannec.github.io/Convolutional_Neural_Networks/.

[42] Christian Szegedy et al. «Rethinking the inception architecture for computer vision». In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 2818–2826.

[43] Jason Brownlee. *A Gentle Introduction to Statistical Data Distributions*. 2019. URL: https://machinelearningmastery.com/statistical-data-distributions/.

[44] Friedrich-Wilhelm Wellmer. «The Normal Distribution». In: *Statistical Evaluations in Exploration for Mineral Deposits*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, pp. 27–30. ISBN: 978-3-642-60262-7. DOI: 10.1007/978-3-642-60262-7_4. URL: https://doi.org/10.1007/978-3-642-60262-7_4.

[45] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Berlin, Heidelberg: Springer-Verlag, 2006. ISBN: 0387310738.

[46] Kevin P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012. ISBN: 0262018020.

[47] T. K. Moon. «The expectation-maximization algorithm». In: *IEEE Signal Processing Magazine* 13.6 (1996), pp. 47–60. DOI: 10.1109/79.543975.

[48] Karl Pearson F.R.S. «LIII. On lines and planes of closest fit to systems of points in space». In: *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2.11 (1901), pp. 559–572. DOI: 10.1080/14786440109462720. eprint: https://doi.org/10.1080/14786440109462720. URL: https://doi.org/10.1080/14786440109462720.

[49] Harold Hotelling. «Relations Between Two Sets of Variates». In: *Biometrika* 28.3/4 (1936), pp. 321–377. ISSN: 00063444. URL: http://www.jstor.org/stable/2333955.

[50] J. MacQueen. «Some methods for classification and analysis of multivariate observations». In: *Proceedings of the Fifth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Statistics*. Berkeley, Calif.: University of California Press, 1967, pp. 281–297. URL: https://projecteuclid.org/euclid.bsmsp/1200512992.

[51] S. Lloyd. «Least squares quantization in PCM». In: *IEEE Transactions on Information Theory* 28.2 (1982), pp. 129–137. DOI: 10.1109/TIT.1982.1056489.

[52] «Abstracts». In: *Biometrics* 21.3 (1965), pp. 761–777. ISSN: 0006341X, 15410420. URL: http://www.jstor.org/stable/2528559.

[53] *Implementing K-means clustering in Python from Scratch*. 2020. URL: https://cmdlinetips.com/2019/05/k-means-clustering-in-python/.

[54] David Arthur and Sergei Vassilvitskii. «K-Means++: The Advantages of Careful Seeding». In: *Proceedings of the Eighteenth Annual ACM-SIAM Symposium on Discrete Algorithms*. SODA '07. Society for Industrial and Applied Mathematics, 2007, 1027–1035. ISBN: 9780898716245.

[55] Alex Krizhevsky. *Learning multiple layers of features from tiny images*. Tech. rep. 2009.

[56] J. Deng et al. «ImageNet: A Large-Scale Hierarchical Image Database». In: *CVPR09*. 2009.

[57] Y. Le and X. Yang. «Tiny ImageNet Visual Recognition Challenge». In: 2015.

[58] Charles R. Harris et al. «Array programming with NumPy». In: *Nature* 585.7825 (2020), pp. 357–362. DOI: 10.1038/s41586-020-2649-2. URL: https://doi.org/10.1038/s41586-020-2649-2.

[59] F. Pedregosa et al. «Scikit-learn: Machine Learning in Python». In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[60] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[61] Luis Capelo. *Beginning Application Development with TensorFlow and Keras: Learn to Design, Develop, Train, and Deploy TensorFlow and Keras Models as Real-World Applications*. Packt Publishing, 2018. ISBN: 1789537290.

[62] Dirk Merkel. «Docker: Lightweight Linux Containers for Consistent Development and Deployment». In: *Linux J.* 2014.239 (2014). ISSN: 1075-3583.