



ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

**Ανίχνευση Εγκεφαλικών Όγκων με χρήση τεχνικών Βαθιάς Μάθησης**

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΜΑΡΙΑ-ΙΩΑΝΝΑ Λ. ΤΖΩΡΤΖΗ**

**Επιβλέπων :** Ανδρέας - Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

**Συνεπιβλέπων :** Θάνος Τάγαρης  
Υποψήφιος Διδάκτορας Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2018





## ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ

Σχολή Ηλεκτρολόγων Μηχανικών  
και Μηχανικών Υπολογιστών  
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

### Ανίχνευση Εγκεφαλικών Όγκων με χρήση τεχνικών Βαθιάς Μάθησης

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

**ΜΑΡΙΑ-ΙΩΑΝΝΑ Λ. ΤΖΩΡΤΖΗ**

**Επιβλέπων :** Ανδρέας - Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

**Συνεπιβλέπων :** Θάνος Τάγαρης  
Υποψήφιος Διδάκτορας Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 12<sup>η</sup> Σεπτεμβρίου 2018.

.....  
Ανδρέας-Γεώργιος Σταφυλοπάτης  
Καθηγητής Ε.Μ.Π.

.....  
Γεώργιος Στάμου  
Αναπληρωτής Καθηγητής Ε.Μ.Π.

.....  
Κωνσταντίνα Νικήτα  
Καθηγήτρια Ε.Μ.Π.

Αθήνα, Σεπτέμβριος 2018

.....  
**ΜΑΡΙΑ-ΙΩΑΝΝΑ Λ. ΤΖΩΡΤΖΗ**

Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Copyright © Μαρία-Ιωάννα Λ. Τζώρτζη, 2018.  
Με επιφύλαξη παντός δικαιώματος. All rights reserved.

Απαγορεύεται η αντιγραφή, αποθήκευση και διανομή της παρούσας εργασίας, εξ ολοκλήρου ή τμήματος αυτής, για εμπορικό σκοπό. Επιτρέπεται η ανατύπωση, αποθήκευση και διανομή για σκοπό μη κερδοσκοπικό, εκπαιδευτικής ή ερευνητικής φύσης, υπό την προϋπόθεση να αναφέρεται η πηγή προέλευσης και να διατηρείται το παρόν μήνυμα. Ερωτήματα που αφορούν τη χρήση της εργασίας για κερδοσκοπικό σκοπό πρέπει να απευθύνονται προς τον συγγραφέα.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευθεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Εθνικού Μετσόβιου Πολυτεχνείου.

# Περίληψη

Τα τελευταία χρόνια έχει διαπιστωθεί μία ραγδαία ανάπτυξη στις τεχνικές της Όρασης Υπολογιστών (Computer Vision) και ιδιαίτερα σε αυτές που σχετίζονται με τη Βαθιά Μάθηση (Deep Learning). Ένα πολύ ενδιαφέρον πεδίο εφαρμογής των τεχνικών Βαθιάς Μάθησης είναι η κατάτμηση Εικόνας (Image Segmentation). Κάποιες από τις πρακτικές εφαρμογές της κατάτμησης εικόνας είναι η Ιατρική Απεικόνιση, συμπεριλαμβανομένων των εικόνων που έχουν παραχθεί από μαγνητική τομογραφία.

Ο σκοπός αυτής της διπλωματικής εργασίας είναι η μελέτη εφαρμογών κατάτμησης εγκεφαλικών όγκων από μαγνητικό τομογράφο. Μία από τις κύριες προκλήσεις που αντιμετωπίστηκαν ήταν η εξεύρεση αξιόπιστης βάσης δεδομένων που να ήταν πλήρως ετικετοποιημένη. Οι ιατρικές εικόνες είναι δύσκολο να βρεθούν λόγω προβλημάτων ιδιωτικότητας. Οι εικόνες που χρησιμοποιήθηκαν προέρχονται από τη βάση δεδομένων του BRATS2017 - Multimodal Brain Tumor Segmentation Challenge 2017.

Για τη μελέτη αυτής της βάσης δεδομένων χρησιμοποιήθηκαν νευρωνικά δίκτυα. Συγκεκριμένα χρησιμοποιήθηκε η δημοφιλής αρχιτεκτονική UNet που χρησιμοποιείται ευρέως και επιτυγχάνει εξαιρετική απόδοση για την κατάτμηση ιατρικών εικόνων. Για τους σκοπούς της διπλωματικής έγινε μια τροποποίηση της αρχιτεκτονικής ώστε να μην χρειάζεται ένα συγκεκριμένο μέγεθος εισόδου για να ταιριάζει στο μοντέλο, αλλά να μπορεί να παίρνει σαν είσοδο ένα αυθαίρετο μέγεθος. Επιπλέον χρησιμοποιήθηκε η αρχιτεκτονική Fully Convolutional DenseNet, η οποία είναι ένας συνδυασμός της αρχιτεκτονικής FCN και της DenseNet. Με αυτό τον τρόπο υπάρχει η δυνατότητα επεξεργασίας εικόνων αυθαίρετου μεγέθους, μεγαλύτερη αποτελεσματικότητα των παραμέτρων καθώς επίσης όλα τα επίπεδα μπορούν να έχουν εύκολη πρόσβαση στα προηγούμενα επίπεδα, καθιστώντας εύκολη την επαναχρησιμοποίηση της πληροφορίας.

Για την αξιολόγηση των μοντέλων αυτών χρησιμοποιήθηκαν ορισμένες μετρικές. Συγκρίνοντας όλα τα μοντέλα για κάθε μετρική που χρησιμοποιήθηκε παρατηρείται ότι τα καλύτερα μοντέλα ήταν αυτά που είχαν εκπαιδευθεί αγνοώντας την τελευταία ετικέτα. Για τα καλύτερα αυτά μοντέλα πιο αξιόπιστες μετρικές για το σκοπό της διπλωματικής αποδείχθηκαν οι μετρικές `accuracy_ignoring_last_label` και `jaccard`.

## **ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ**

Τεχνητή Νοημοσύνη, Βαθιά Μάθηση, Κατάτμηση Εγκεφαλικού Όγκου, UNet, FCN-DenseNet, Μετρικές



# Abstract

Over the past years, there has been a rapid development in the field of Computer Vision, especially through techniques involving Deep Learning. A very interesting field of application of the Techniques of Deep Learning is the image segmentation. Some of the practical applications of image segmentation are Medical Imaging, including images produced by magnetic resonance imaging.

The aim of this diploma thesis is the study of magnetic tumor tomography segmentation applications. One of the main challenges faced was finding a reliable, fully-labeled database. Medical images are difficult to find because of privacy problems. The images used are derived from the database of BRATS2017-Multimodal Brain Tumor Segmentation Challenge 2017.

Neural Networks were used to study this database. Specifically, the popular UNet architecture, which is widely used and achieves excellent performance for the segmentation of medical images. For the purpose of this diploma thesis, a modification of the architecture was made, so that it was necessary for a particular input size to fit the model, but to be able to take an arbitrary size. In addition, the Fully Convolutiona DenseNet architecture was used, which is a combination of the FCN and DenseNet architectures. In this way, it is possible to process images of arbitrary size, greater efficiency of the parameters and also having the ability of all levels easily access the previous levels, making it easy to reuse the information.

Some metrics were used to evaluate these models. By comparing all models for each metric used, it is observed that the best models were those that were trained ignoring the last label. For these best models more reliable metrics for the purpose of the diploma thesis were the `accuracy_ignoring_last_label` and `jaccard`.

## **KEY WORDS**

Artificial Intelligence, Deep Learning, Brain Tumor Segmentation, UNet, FCN-DenseNet, Metrics





# Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε στο πλαίσιο του Προπτυχιακού Προγράμματος Σπουδών της Σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου και σηματοδοτεί την ολοκλήρωση των σπουδών μου. Πριν όμως από οποιαδήποτε αναφορά στη διαδικασία που ακολουθήθηκε και στα αποτελέσματα που προέκυψαν, θα ήθελα να ευχαριστήσω θερμά τους ανθρώπους με τους οποίους συνεργάστηκα και συνέβαλαν στην ολοκλήρωση της εργασίας αυτής.

Κατ'αρχάς απευθύνω τις ευχαριστίες μου στον επιβλέποντα κ. Ανδρέα - Γεώργιο Σταφυλοπάτη, Καθηγητή Ε.Μ.Π., για την δυνατότητα που μου προσέφερε να εργαστώ σε ένα αντικείμενο ιδιαίτερα ελκυστικό για μένα και να διευρύνω τις γνώσεις μου. Παράλληλα, θα ήθελα να ευχαριστήσω τον κ. Γεώργιο Στάμου, Αναπληρωτή Καθηγητή Ε.Μ.Π. για τη βοήθεια του και τη συνεργασία που είχαμε όλο αυτό το διάστημα. Ακόμη θα ήθελα να ευχαριστήσω την κ. Κωνσταντίνα Νικήτα, Καθηγήτρια Ε.Μ.Π. που με τίμησε με την παρουσία της στην τριμελή επιτροπή εξέτασης, αλλά και για την πολύτιμη συνεργασία μας στα πρώτα μου φοιτητικά χρόνια.

Επίσης οφείλω ιδιαίτερες ευχαριστίες στο Θάνο Τάγαρη, Υποψήφιο Διδάκτορα Ε.Μ.Π., για το χρόνο που αφιέρωσε και τη θεμελιώδη συνεισφορά του στην εκπόνηση της παρούσας εργασίας. Τόσο η επιστημονική όσο και η πνευματική του στήριξη, ιδιαίτερα στα τελευταία στάδια της εργασίας, ήταν ιδιαίτερα σημαντικές για εμένα. Η εμπειρία και οι γνώσεις του στάθηκαν καθοριστικές και η συνεργασία μας θεωρώ πως ήταν άκρως επιτυχημένη και εποικοδομητική. Οφείλω να ευχαριστήσω και όλα τα παιδιά του εργαστηρίου που με βοήθησαν όλους αυτούς τους μήνες.

Θέλω να ευχαριστήσω τους γονείς μου, την αδελφή μου Πένυ, το αγόρι μου Γιάννη και τους φίλους μου οι οποίοι βρίσκονται κοντά μου και με στηρίζουν όλα αυτά τα χρόνια.

Τέλος, αφιερώνω αυτή την εργασία στον παππού μου Γιάννη, ο οποίος πάλεψε με αυτή την ασθένεια και ηττήθηκε αλλά που θα ήταν πολύ περήφανος για εμένα αν ήταν εδώ σήμερα.

Μαρία - Ιωάννα Τζώρτζη Αθήνα, 12/09/2018



# Contents

<b>1</b>	<b>Εισαγωγή</b>	<b>13</b>
1.1	Τεχνητή Νοημοσύνη . . . . .	13
1.2	Μηχανική Μάθηση . . . . .	15
1.3	Βαθιά Μάθηση . . . . .	16
1.4	Κατάτμηση Εικόνας . . . . .	16
1.5	Εγκεφαλικός Όγκος . . . . .	19
<b>2</b>	<b>Θεωρία</b>	<b>23</b>
2.1	Τεχνητά Νευρωνικά Δίκτυα . . . . .	23
2.1.1	Perceptron . . . . .	24
2.1.2	Multilayer Perceptron (MLP) . . . . .	25
2.1.3	Overfitting-Imbalance . . . . .	28
2.1.4	Ανισορροπία Κλάσεων - Imbalance classes . . . . .	31
2.2	Συνελικτικά Νευρωνικά Δίκτυα . . . . .	32
2.2.1	Συνελικτικό Επίπεδο . . . . .	33
2.2.2	Συναρτήσεις Ενεργοποίησης . . . . .	37
2.2.3	Επίπεδο Συγκέντρωσης . . . . .	40
2.2.4	Λοιπά Επίπεδα . . . . .	41
2.3	Απόδοση Νευρωνικών Δικτύων . . . . .	42
2.3.1	Σφάλματα . . . . .	42
2.3.2	Μετρικές . . . . .	44
2.4	Βελτιστοποίηση . . . . .	49
2.4.1	Gradient Descent . . . . .	49
2.4.2	Batch Gradient Descent . . . . .	50
2.4.3	Stochastic Gradient Descent-SGD . . . . .	50
2.4.4	Adam . . . . .	53
<b>3</b>	<b>Μοντέλα</b>	<b>55</b>
3.1	UNet . . . . .	55
3.2	Fully Convolutional DenseNet . . . . .	56
<b>4</b>	<b>Πειραματική Διαδικασία</b>	<b>61</b>
4.1	Σύνολο Δεδομένων . . . . .	61
4.1.1	Προεπεξεργασία Δεδομένων . . . . .	62
4.2	Εργαλεία . . . . .	63
4.3	Εκπαίδευση . . . . .	63
4.3.1	Απλά Μοντέλα . . . . .	63
4.3.2	Εξισορρόπηση δεδομένων με βάρη κλάσεων . . . . .	65
4.3.3	Εκπαίδευση Αγνοώντας την Τελευταία Ετικέτα . . . . .	66
4.3.4	Μοντέλα cw με αρχικοποίηση από ignoring . . . . .	67

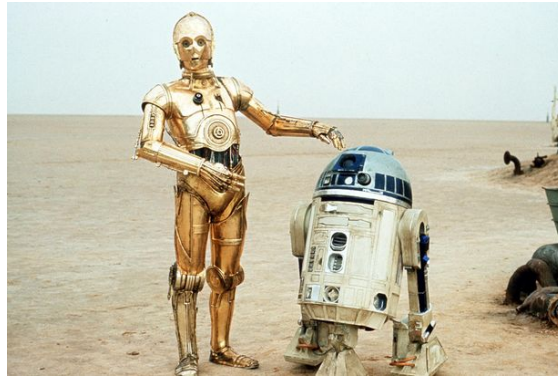
4.4	Αποτελέσματα . . . . .	67
4.4.1	Απλά μοντέλα . . . . .	70
4.4.2	Μοντέλα με χρήση cw . . . . .	72
4.4.3	Μοντέλα αγνοώντας την τελευταία κλάση . . . . .	74
4.4.4	Μοντέλα cw με αρχικοποίηση από ignoring . . . . .	76
4.4.5	Σύγκριση του συνόλου Εκπαίδευσης και του συνόλου Δοκιμής για τις μετρικές accuracy_ignoring_last_label και jaccard . . . .	78
4.5	Συμπεράσματα . . . . .	78
4.6	Μελλοντική Εργασία . . . . .	80
	<b>Βιβλιογραφία</b>	<b>81</b>

# Κεφάλαιο 1

## Εισαγωγή

### 1.1 Τεχνητή Νοημοσύνη

Από την αρχαιότητα ακόμα είχαν κάνει την εμφάνιση τους μέσα από παραμύθια και ιστορίες τεχνητά όντα με δυνατότητα σκέψης. Ωστόσο η σύγχρονη *Τεχνητή Νοημοσύνη* (“Artificial Intelligence”) ξεκινά από την δεκαετία του 1940. Η πρόσβαση των ερευνητών στους ηλεκτρονικούς υπολογιστές καθώς επίσης και η εισαγωγή νέων ιδεών όπως η θεωρία παιγνίων του John von Neumann το 1944 και τα τεχνητά νευρωνικά δίκτυα (McCulloch-Pitts, 1943) οδήγησαν ερευνητές από διάφορους τομείς στο να εξετάσουν την πιθανότητα δημιουργίας ενός τεχνητού εγκεφάλου ή μιας σκεπτόμενης μηχανής. Μετά το Β΄ Παγκόσμιο Πόλεμο ο Βρετανός μαθηματικός Alan Turing με το άρθρο του *Computing Machinery and Intelligence* συζήτησε τις συνθήκες για την εξέταση μιας μηχανής ως έξυπνης. Ισχυρίστηκε ότι εάν η μηχανή μπορούσε να προσποιηθεί με επιτυχία ότι είναι άνθρωπος σε έναν έμπειρο παρατηρητή τότε σίγουρα θα πρέπει να θεωρηθεί έξυπνο - το περίφημο Turing test (Harasim, 2012). Το 1956, το πεδίο της έρευνας της Τεχνητής Νοημοσύνης γεννήθηκε σε ένα συνέδριο στο Dartmouth College στις Ηνωμένες Πολιτείες της Αμερικής. Κατά τη διάρκεια του συνεδρίου ο John McCarthy όρισε την Τεχνητή Νοημοσύνη ως η επιστήμη και η μηχανική της δημιουργίας ευφών μηχανών και ειδικότερα ευφών προγραμμάτων (McCarthy et al., 2006). Οι περισσότεροι από αυτούς που παρακολούθησαν το συνέδριο έγιναν οι πρώτοι ερευνητές της νέας επιστήμης, και ίδρυσαν εργαστήρια στα μεγαλύτερα πανεπιστήμια των Ηνωμένων Πολιτειών. Από τη στιγμή αυτή και μέχρι τα μέσα της δεκαετίας του 1970, η έρευνα στον τομέα της Τεχνητής Νοημοσύνης άνθησε. Τα αποτελέσματα ήταν εντυπωσιακά: οι ηλεκτρονικοί υπολογιστές μάθαιναν να λύνουν προβλήματα άλγεβρας, να καταλαβαίνουν και να εκτελούν απλές εντολές σε φυσική γλώσσα. Η χρηματοδότηση ήταν γενναιοδωρη και χωρίς περιορισμούς και οι προσδοκίες των ερευνητών ήταν υψηλές: ο Marvin Minsky το 1967 δήλωσε ότι μέσα σε μια γενιά, το πρόβλημα της δημιουργίας τεχνητής νοημοσύνης θα είχε λυθεί (Minsky, 1967).



**Σχήμα 1.1:** C-3PO και R2-D2 από την ταινία ο Πόλεμος των Άστρων ([imdb](#)).

Ωστόσο η έρευνα στην πορεία σκόνταψε σε κάποια προβλήματα τα οποία δεν λύνονταν εύκολα, και τα πρώτα χρόνια της δεκαετίας του 70 έφτασαν χωρίς τα αναμενόμενα αποτελέσματα. Η έλλειψη αποτελεσμάτων οδήγησε σταδιακά στην διακοπή της χρηματοδότησης και άρχισε ο πρώτος "χειμώνας" της Τεχνητής Νοημοσύνης.

Στις αρχές της δεκαετίας του 1980, το ενδιαφέρον για την Τεχνητή Νοημοσύνη άρχισε να ανακάμπτει με την δημιουργία ενός νέου είδους προγραμμάτων τεχνητής νοημοσύνης, των "έμπειρων συστημάτων" (McCorduck, 2004). Τα έμπειρα συστήματα είναι προγράμματα που απαντάνε σε ερωτήσεις ή βγάζουν χρήσιμα συμπεράσματα για έναν πολύ συγκεκριμένο τομέα, χρησιμοποιώντας ένα σύνολο από κανόνες και μια βάση δεδομένων που περιέχει γνώσεις σχετικές με τον τομέα. Οι κανόνες και οι γνώσεις εισάγονται στο σύστημα από ειδικούς με εμπειρία στον συγκεκριμένο τομέα. Τα έμπειρα συστήματα άρχισαν να χρησιμοποιούνται με επιτυχία από επιχειρήσεις παγκοσμίως, με αποτέλεσμα να δημιουργηθεί μια αγορά δισεκατομμυρίων από την πώληση των εξειδικευμένων μηχανημάτων που έτρεχαν τα συστήματα. Οι ερευνητές την δεκαετία αυτή επικεντρώθηκαν σε συστήματα βασισμένα στην γνώση και την αναπαράστασή της.

Το ενδιαφέρον όμως δεν κράτησε για πολύ μιας και οι ηλεκτρονικοί υπολογιστές αποκτούσαν όλο και αυξανόμενη ταχύτητα, ξεπερνώντας τα μηχανήματα έμπειρων συστημάτων σε υπολογιστική ισχύ.

Από τις αρχές της δεκαετίας του 1990 μέχρι και σήμερα, ο τομέας της Τεχνητής Νοημοσύνης έχει σταδιακά ανακάμψει και έχει πετύχει αρκετούς από τους αρχικούς στόχους που είχαν θέσει οι πρώτοι ερευνητές. Τα αποτελέσματα της έρευνας αυτών των χρόνων όλο και περισσότερο χρησιμοποιούνται σε πρακτικές εφαρμογές. Η Τεχνητή Νοημοσύνη χρησιμοποιείται στην εξόρυξη δεδομένων, την ιατρική διάγνωση και άλλους τομείς. Η επιτυχία αυτή οφείλεται στην αύξηση της υπολογιστικής ισχύος, στην επίλυση συγκεκριμένων προβλημάτων, στους νέους δεσμούς μεταξύ της Τεχνητής Νοημοσύνης και άλλων τομέων (όπως στατιστική, οικονομικά και μαθηματικά) και στην αφοσίωση των ερευνητών σε μαθηματικές μεθόδους και επιστημονικά πρότυπα.

Σύμφωνα με τον Jack Clark του Bloomberg, το 2015 ήταν ένα έτος ορόσημο για την Τεχνητή Νοημοσύνη, με τον αριθμό των προγραμμάτων λογισμικού που χρησιμοποιούν Τεχνητή Νοημοσύνη στη Google να αυξάνεται ραγδαία. Πλέον έχουμε ένα τεράστιο όγκο δεδομένων, διότι η ανθρωπότητα λόγω της δραστηριότητάς της στο Ίντερνετ έχει αφήσει ένα τεράστιο ψηφιακό αποτύπωμα, το οποίο μπορούμε να εκμεταλλευτούμε για να προπονήσουμε τους αλγόριθμους τεχνητής νοημοσύνης. Πολλοί ερευνητές και επιστήμονες εκφράζουν βέβαια μια πιο δυστοπική εκδοχή του

μέλλοντος, σύμφωνα με την οποία η Τεχνητή Νοημοσύνη είναι δυνατόν να προκαλέσει σοβαρές κοινωνικές αναταραχές και να αποτελέσει άμεση απειλή για τη δημοκρατία και την παγκόσμια ειρήνη. Ωστόσο όποιος προβληματισμός κι αν υπάρχει δεν μπορεί να σταματήσει το κύμα της προόδου της τεχνολογίας: “Είναι σαν να γυρίζουμε την πλάτη σε ένα κύμα που έρχεται να παρασύρει τα πάντα”. Το πιο σημαντικό είναι να σκεφτόμαστε πώς μπορούμε να κατευθύνουμε αυτή την αναμορφωτική δύναμη με τρόπο ώστε να αλλάξει θετικά τη ζωή του ανθρώπου ([Δασκαλάκης, 2018](#)).

## 1.2 Μηχανική Μάθηση

Η *Μηχανική Μάθηση* (Machine Learning) είναι ένα υποπεδίο της επιστήμης των υπολογιστών, το οποίο προέκυψε από την τομή της Επιστήμης των Υπολογιστών με την Στατιστική ([Mitchell, 2006](#)). Μία πολύ σημαντική ανακάλυψη από τον Arthur Samuel το 1959, είναι η υλοποίηση ότι αντί να διδάσκεται στους υπολογιστές ό,τι πρέπει να ξέρουν για τον κόσμο και τον τρόπο εκτέλεσης καθηκόντων, ίσως είναι δυνατόν να διδαχθούν πως να μάθουν για τον εαυτό τους. Ο Samuel ορίζει τη μηχανική μάθηση ως “Πεδίο μελέτης που δίνει στους υπολογιστές την ικανότητα να μαθαίνουν, χωρίς να έχουν ρητά προγραμματιστεί”. Την περασμένη δεκαετία, η μηχανική μάθηση μας έδωσε αυτόματη-οδήγηση αυτοκινήτων, πρακτική αναγνώριση ομιλίας, αποτελεσματική αναζήτηση ιστού καθώς και πολύ βελτιωμένη κατανόηση του ανθρώπινου γονιδιώματος. Η μηχανική μάθηση είναι τόσο διαδεδομένη σήμερα που πιθανώς να χρησιμοποιείται από τον κάθε άνθρωπο δεκάδες φορές την ημέρα χωρίς να γίνεται άμεσα αντιληπτό. Στην ουσία η μηχανική μάθηση είναι μια κατηγορία αλγορίθμων που είναι σε θέση να κάνουν προβλέψεις και να προσαρμοστούν όταν παρουσιάζονται νέα δεδομένα. Ένας αλγόριθμος μηχανικής μάθησης στην αρχή είναι τυπικά ανακριβής, αλλά βελτιώνεται με την πρόβλεψη πάνω σε νέα δεδομένα και προσαρμόζεται βάσει μιας διαδικασίας δοκιμής και σφάλματος. Η μέθοδος αυτή είναι σε θέση να λύσει προβλήματα που ο κλασικός προγραμματισμός αδυνατεί, με μεγαλύτερο όμως κόστος σε υπολογιστικούς πόρους και χρόνο. Επίσης με την αύξηση των δεδομένων που έχει στην διάθεση του ένας τέτοιος αλγόριθμος, είναι σε θέση να επιλύει δυσκολότερα προβλήματα, όπως αναγνώριση εικόνας, ήχου, φυσικής γλώσσας με μεγαλύτερη ακρίβεια.

Υπάρχουν τρεις μεγάλες κατηγορίες αλγορίθμων μηχανικής μάθησης, ανάλογα με τον τρόπο που επεξεργάζονται τα δεδομένα εισόδου και την ανατροφοδότηση που υπάρχει στο σύστημα εκμάθησης ([J Russell and Norvig, 1995](#)):

1. **Επιβλεπόμενη μάθηση (Supervised learning):** Οι αλγόριθμοι δέχονται ως είσοδο τα δεδομένα μαζί με τις ετικέτες τους και καλούνται να μάθουν ένα γενικό κανόνα προκειμένου να βρίσκουν την αντιστοίχιση (συσχέτιση) μεταξύ δεδομένων και ετικετών.
2. **Μη-επιβλεπόμενη μάθηση (Unsupervised learning):** Οι αλγόριθμοι δέχονται ως είσοδο δεδομένα χωρίς ετικέτες (χωρίς να παρέχεται δηλαδή κάποια εμπειρία στον αλγόριθμο μάθησης) και καλούνται να βρουν την δομή τους. Η μη-επιβλεπόμενη μάθηση μπορεί να ανακαλύπτει κρυμμένα μοτίβα σε δεδομένα ή μπορεί να αποτελέσει μέσο για την εύρεση χαρακτηριστικών, που στη συνέχεια χρησιμοποιούνται στη διαδικασία της μάθησης.
3. **Ενισχυτική μάθηση (Reinforcement learning):** Αλγόριθμοι που αλληλεπιδρούν με ένα δυναμικό περιβάλλον, για την επίτευξη ενός συγκεκριμένου στόχου.

Στην περίπτωση αυτή δεν υπάρχει πληροφορία που να υποδεικνύει πόσο κοντά ή μακριά έχει φτάσει η εκπαίδευση από τον στόχο. Παράδειγμα τέτοιας μάθησης θα μπορούσε να είναι η αυτόνομη οδήγηση ενός οχήματος.

Στο πλαίσιο αυτής της διπλωματικής εργασίας θα ασχοληθούμε με προβλήματα επιβλεπόμενης μάθησης.

### 1.3 Βαθιά Μάθηση

Ο όρος *Βαθιά Μάθηση* (Deep Learning) εισήχθη στην κοινότητα της Μηχανικής Μάθησης από τη Rina Dechter το 1986. Η Βαθιά Μάθηση-Deep Learning είναι μια υποπεριοχή της μηχανικής μάθησης. Για τη Βαθιά Μάθηση σε αντίθεση με τα απλά τεχνητά νευρωνικά δίκτυα (Artificial Neural Network - ANN) ισχύει ότι υπάρχουν περισσότερα επίπεδα στοιβαγμένα το ένα μετά το άλλο με αποτέλεσμα να δίνεται η δυνατότητα να εξάγονται περισσότερα υψηλού επιπέδου χαρακτηριστικά.

Αρχιτεκτονικές βαθιάς μάθησης όπως τα δίκτυα βαθιάς πίστης, (deep belief networks), τα CNN, τα RNN έχουν εφαρμοστεί σε πεδία που περιλαμβάνουν την όραση υπολογιστών, (computer vision), την αναγνώριση ομιλίας, (speech recognition), την επεξεργασία φυσικής γλώσσας, (natural language processing), το φιλτράρισμα κοινωνικών δικτύων, τη βιοπληροφορική και τον σχεδιασμό φαρμάκων και έχουν παράγει συγκρίσιμα αποτελέσματα και σε ορισμένες περιπτώσεις καλύτερα από αυτά των ανθρώπων. Οι τεχνικές Βαθιάς Μάθησης λοιπόν έχουν βελτιώσει την ικανότητα ταξινόμησης, αναγνώρισης, ανίχνευσης και περιγραφής - με μια λέξη, την κατανόηση. Παραδείγματα χρήσης της Βαθιάς Μάθησης είναι τα συστήματα Siri και Cortana της Apple και της Microsoft αντίστοιχα.

Σήμερα πολλές είναι οι εξελίξεις που προωθούν τη βαθιά μάθηση, όπως για παράδειγμα οι βελτιώσεις στις αρχιτεκτονικές και στους αλγορίθμους μάθησης. Επίσης οι νέες προσεγγίσεις μηχανικής μάθησης έχουν βελτιώσει την ακρίβεια των μοντέλων, έχουν αναπτυχθεί νέες τάξεις νευρωνικών δικτύων που ταιριάζουν καλά σε εφαρμογές όπως η μετάφραση κειμένου και η ταξινόμηση των εικόνων. Διαθέτουμε πολύ περισσότερα δεδομένα για την κατασκευή νευρωνικών δικτύων, με πολλά βαθιά επίπεδα. Πολύ βασικό να αναφερθεί είναι ότι τώρα πια υπάρχει στη διάθεση των ανθρώπων άπιστευτη υπολογιστική ισχύ, η οποία είναι απαραίτητη για την κατάρτιση βαθιών αλγορίθμων. Με την χρήση GPU επιταχύνονται οι αλγόριθμοι εκπαίδευσης κατά τάξεις μεγέθους, μειώνοντας τους χρόνους λειτουργίας από εβδομάδες σε ημέρες.

Η Βαθιά Μάθηση λοιπόν αλλάζει τον τρόπο σκέψης της αναπαράστασης των προβλημάτων από την υπόδειξη προς τον υπολογιστή στο πως να λύσει το πρόβλημα, στην εκπαίδευση του υπολογιστή για να λύσει το ίδιο το μηχανήμα το πρόβλημα. Η υπόσχεση της Βαθιάς Μάθησης είναι ότι μπορεί να οδηγήσει σε συστήματα πρόγνωσης που γενικεύουν καλά, προσαρμόζονται καλά, βελτιώνεται η απόδοσή τους με περισσότερα δεδομένα και είναι πιο δυναμικά από τα συστήματα πρόβλεψης που βασίζονται σε αυστηρούς κανόνες. Δεν χρειάζεται πια να εγκατασταθεί ένα μοντέλο, αντί αυτού εκπαιδεύεται η εργασία.

### 1.4 Κατάτμηση Εικόνας

Στην όραση του υπολογιστή (Computer Vision) η *Κατάτμηση Εικόνας* (Image Segmentation) είναι η διαδικασία διαίρεσης μιας ψηφιακής εικόνας σε πολλαπλά τμήματα (σύνολα εικονοστοιχείων (sets of pixels), επίσης γνωστά ως



υπερ-εικονοστοιχεία). Ο στόχος της τμηματοποίησης είναι να απλοποιήσει και/ή να αλλάξει την αναπαράσταση μιας εικόνας σε κάτι που είναι πιο ουσιαστικό και ευκολότερο να αναλυθεί (Srinivasan and Shobha, 2007). Η κατάτμηση εικόνων χρησιμοποιείται συνήθως για τον εντοπισμό αντικειμένων και ορίων (γραμμών, καμπύλων κ.λπ.) στις εικόνες. Πιο συγκεκριμένα, η κατάτμηση της εικόνας είναι η διαδικασία εκχώρησης μιας ετικέτας σε κάθε εικονοστοιχείο σε μια εικόνα, έτσι ώστε εικονοστοιχεία με την ίδια ετικέτα να μοιράζονται ορισμένα χαρακτηριστικά. Το αποτέλεσμα της κατάτμησης της εικόνας είναι ένα σύνολο από τμήματα που καλύπτουν συλλογικά ολόκληρη την εικόνα ή ένα σύνολο από περιγράμματα που εξάγονται από την εικόνα. Κάθε ένα από τα εικονοστοιχεία μιας περιοχής είναι παρόμοια σε σχέση με κάποια χαρακτηριστική ή υπολογιζόμενη ιδιότητα, όπως χρώμα, ένταση ή υφή. Οι γειτονικές περιοχές διαφέρουν σημαντικά σε σχέση με τα ίδια χαρακτηριστικά (Srinivasan and Shobha, 2007). Κάποιες από τις πρακτικές εφαρμογές της κατάτμησης εικόνας είναι:

- Μηχανική Όραση
- Ιατρική Απεικόνιση (Pham et al., 2000), συμπεριλαμβανομένων των εικόνων που έχουν παραχθεί σε όγκο από υπολογιστική τομογραφία και μαγνητική τομογραφία (εντοπισμός όγκων και άλλων παθολογιών, μέτρηση όγκων των ιστών, διάγνωση, μελέτη της ανατομικής δομής, χειρουργικός σχεδιασμός, εικονική χειρουργική προσομοίωση, ενδοχειρουργική πλοήγηση)
- Ανίχνευση αντικειμένων
- Ανίχνευση πεζών
- Ανίχνευση προσώπου
- Ανίχνευση φώτων φρένων
- Αναγνώριση προσώπου
- Αναγνώριση δακτυλικών αποτυπωμάτων
- Συστήματα ελέγχου της κυκλοφορίας
- Παρακολούθηση βίντεο

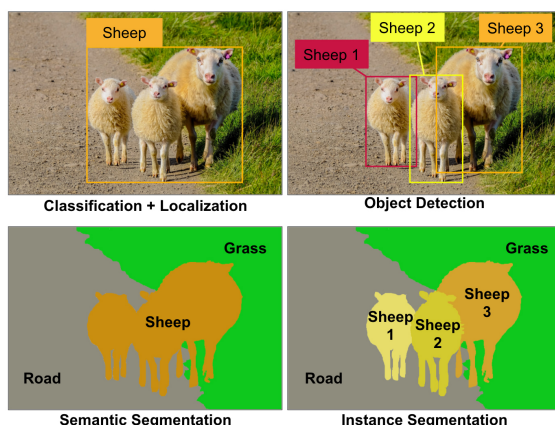
Η κατάτμηση της εικόνας θεωρείται ως μια βασική λειτουργία για την ουσιαστική ανάλυση και ερμηνεία της εικόνας που αποκτήθηκε. Είναι ένα κρίσιμο και ουσιαστικό στοιχείο της ανάλυσης εικόνας ή αναγνώρισης προτύπου και είναι ένα από τα πιο δύσκολα καθήκοντα στην επεξεργασία εικόνας. Οι ερευνητές έχουν εργαστεί εκτενώς πάνω σε αυτό το βασικό πρόβλημα και πρότειναν διάφορες μεθόδους για τη κατάτμηση εικόνας. Οι μέθοδοι αυτές μπορούν να ταξινομηθούν ευρέως σε επτά ομάδες (Sathya and Manavalan, 2011):

1. Ιστόγραμμα κατωφλίου (Histogram thresholding)
2. Όμαδοποίηση (Clustering)
3. Ανάπτυξη, διάσπαση περιοχής και συγχώνευση (Region growing, region splitting and merging)

4. Βασισμένη στις ακμές (Edge-Based)
5. Βασισμένη σε φυσικό μοντέλο (Physical model-based)
6. Προσεγγίσεις ασαφούς λογικής (Fuzzy approaches)
7. με βάση τα Νευρωνικά Δίκτυα (Neural Network and GA-Genetic algorithm)

Για να είναι χρήσιμες, αυτές οι μέθοδοι πρέπει συνήθως να συνδυάζονται με τις ειδικές γνώσεις ενός τομέα, προκειμένου να επιλυθούν αποτελεσματικά τα προβλήματα τμηματοποίησης του τομέα.

Οι περισσότερες μέθοδοι κατακερματισμού βασίζονται μόνο στις πληροφορίες χρώματος των εικονοστοιχείων στην εικόνα. Οι άνθρωποι χρησιμοποιούν πολύ περισσότερη γνώση από αυτό όταν κάνουν κατάτμηση της εικόνας, αλλά η εφαρμογή της γνώσης αυτής θα κοστίζει σημαντικό χρόνο υπολογισμού και θα απαιτούσε μια τεράστια βάση δεδομένων γνώσης κάποιου τομέα, η οποία αυτή την στιγμή δεν είναι διαθέσιμη. Εκτός από τις παραδοσιακές μεθόδους κατακερματισμού, υπάρχουν μέθοδοι τμηματοποίησης που μπορούν να εκπαιδευτούν και οι οποίες μπορούν να διαμορφώσουν μερικές από αυτές τις γνώσεις. Η τμηματοποίηση με Νευρωνικά δίκτυα βασίζεται στην επεξεργασία μικρών περιοχών μια εικόνας. Μετά από αυτήν την επεξεργασία ο μηχανισμός λήψης αποφάσεων επισημαίνει τις περιοχές μιας εικόνας ανάλογα με την κατηγορία που αναγνωρίζεται από το νευρωνικό δίκτυο (Pathegama and Göl, 2004).



**Σχήμα 1.2:** Μερικά από τις βασικές λειτουργίες της υπολογιστικής όρασης. Σήμερα κάθε μία από αυτές τις εργασίες απαιτεί μια πολύ διαφορετική αρχιτεκτονική CNN, για παράδειγμα ResNet για ταξινόμηση, YOLO για ανίχνευση αντικειμένων, μάσκα R-CNN για στιγμιαία κατάτμηση κ.ο.κ. (Géron, 2018).

Η αυτόματη κατάτμηση εικόνας, όπως ήδη αναφέρθηκε, χρησιμοποιείται κατά κόρον στην Ιατρική. Η εφαρμογή της σε μία μεγάλη ποικιλία ιατρικών εικόνων, βοηθάει ουσιαστικά τους γιατρούς διαφόρων ειδικοτήτων, στη συντόμηση χρονοβόρων διαδικασιών, στην έγκαιρη και ακριβέστερη πρόγνωση και διάγνωση παθήσεων και στην λήψη αποφάσεων για τον βέλτιστο τρόπο θεραπείας των διαφόρων παθήσεων. Υπάρχουν αρκετά είδη ιατρικών εικόνων που μπορεί να εφαρμοστεί η κατάτμηση, όπως τα υπερηχογραφήματα (ultrasound), η τομογραφία εκπομπής ποζιτρονίων (Positron Emission Tomography-PET), η μαγνητική τομογραφία (Magnetic Resonance Imaging-MRI) και άλλες. Η συγκεκριμένη διπλωματική εργασία αναφέρεται στην αυτοματοποιημένη κατάτμηση εγκεφαλικού όγκου από εικόνες που προέρχονται από μαγνητική τομογραφία.

## 1.5 Εγκεφαλικός Όγκος

Ένας *Εγκεφαλικός Όγκος* είναι οποιοσδήποτε ενδοκρανιακός όγκος που χαρακτηρίζεται από ανεξέλεγκτη διαίρεση κυττάρων, κανονικά είτε μέσα στον εγκέφαλο, στα κρανιακά νεύρα, στις μεμβράνες που καλύπτουν τον εγκέφαλο (μήνιγγες), στο κρανίο, στο βλεννογόνο και κωνοειδή αδένα ή μπορεί να εξαπλωθεί από κάποιον καρκίνο που εντοπίζεται σε άλλο μέρος του σώματος (μεταστατικοί όγκοι). Οι πρωτογενείς εγκεφαλικοί όγκοι στα παιδιά εντοπίζονται συνήθως στο οπίσθιο κρανιακό βόθριο και στους ενήλικες συνήθως εντοπίζονται στα εμπρόσθια εγκεφαλικά ημισφαίρια, παρόλο που μπορούν να αναπτυχθούν σε οποιοδήποτε μέρος του εγκεφάλου.

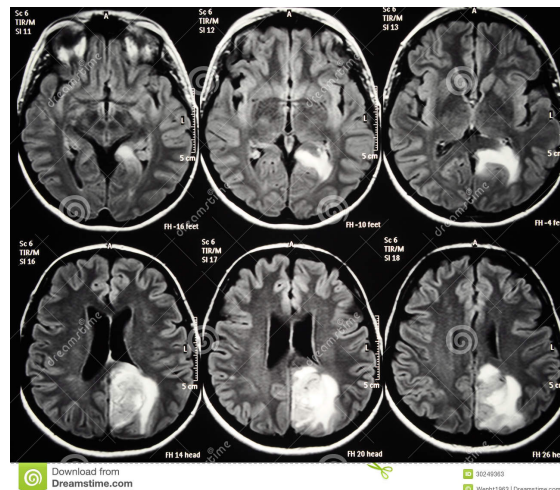
Ο καρκίνος είναι μια από τις κύριες αιτίες θανάτου στον κόσμο. Στις ΗΠΑ, ο καρκίνος είναι η 2η κύρια αιτία που ξεπερνιέται μόνο από καρδιακές παθήσεις. Για να το θέσουμε σε προοπτική, ένας στους τέσσερις θανάτους στις ΗΠΑ προκαλείται από καρκίνο. (National Cancer Institute, 2014). Οι εγκεφαλικοί όγκοι χωρίζονται σε δύο κατηγορίες, σε πρωτογενείς και σε δευτερογενείς. Οι πρωτογενείς όγκοι εμφανίζονται από μόνοι τους, χωρίς να έχει γίνει μετάσταση από κάποιο άλλο μέρος του σώματος. Οι δευτερογενείς όγκοι είναι μεταστάσεις καρκίνων από διαφορετικά όργανα του σώματος. Οι δευτερογενείς αυτοί όγκοι είναι συχνότεροι από ότι οι πρωτογενείς όγκοι του εγκεφάλου. Στα παιδιά οι περισσότεροι όγκοι είναι πρωτογενείς. Στους ενήλικες συμβαίνει το αντίθετο, δηλαδή οι περισσότεροι είναι μεταστάσεις καρκίνων που εκδηλώθηκαν αλλού στο σώμα.

Η ιατρική ονομασία ενός όγκου στον εγκέφαλο είναι *γλοίωμα*. Ένα γλοίωμα είναι ένας τύπος όγκου που αρχίζει στον εγκέφαλο ή στη σπονδυλική στήλη. Ονομάζεται γλοίωμα επειδή προέρχεται από τα γλοιακά κύτταρα. Τα γλοιώματα αποτελούν περίπου το 30% όλων των όγκων του εγκεφάλου και του κεντρικού νευρικού συστήματος και το 80% όλων των κακοήθων όγκων στον εγκέφαλο. Από τα πολυάριθμα συστήματα ταξινόμησης που χρησιμοποιούνται, ο πιο κοινός οργανισμός είναι ο Παγκόσμιος Οργανισμός Υγείας (World Health Organization (WHO)). Σύμφωνα με αυτόν οι όγκοι βαθμολογούνται από τον I (λιγότερο προηγμένη πρόγνωση για την καλύτερη ασθένεια), έως την IV (χειρότερη πρόγνωση της νόσου). Γενικά ο βαθμός I και ο βαθμός II είναι καλοήθης όγκος στον εγκέφαλο. Οι βαθμοί III και IV είναι κακοήθεις όγκοι στον εγκέφαλο. Χρησιμοποιώντας αυτό το σύστημα ταξινόμησης, το οποίο καθορίζεται από την παθολογική εξέλιξη του όγκου, τα γλοιώματα μπορούν να ταξινομηθούν στα εξής (Moreno Lopez, 2017):

- Τα χαμηλού βαθμού γλοιώματα (Low-Grade gliomas) (βαθμός II της WHO) είναι καλά διαφοροποιημένα, τείνουν να παρουσιάζουν καλοήθεις τάσεις και προμηνύουν μια καλύτερη πρόγνωση για τον ασθενή. Ωστόσο, έχουν ομοιόμορφο ρυθμό επανάληψης και αύξηση κατά βαθμό έτσι ώστε να καταλήξουν να κατατάσσονται ως κακοήθεις.
- Τα υψηλού βαθμού γλοιώματα (High-Grade gliomas) (βαθμός III-IV της WHO) είναι αδιαφοροποίητα, είναι κακοήθη και φέρουν τη χειρότερη πρόγνωση.

Ένας από τους πιο κοινούς τρόπους διάγνωσης όγκων στον εγκέφαλο είναι ο *Μαγνητικός Τομογράφος* (Magnetic Resonance Imaging-MRI). Αυτό το μηχάνημα έχει σχεδιαστεί για να εξαναγκάσει όλα τα ιόντα υδρογόνου στο σώμα μας σε μια συγκεκριμένη κατεύθυνση και αυτό που κάνει είναι ότι παίρνει την εικόνα σε διαφορετικές γωνίες, διαφορετικό χρόνο, διαφορετική ένταση και έτσι

παράγει διαφορετικούς τύπους ταινιών, όπως τα *T1*, *T1-contrast*, *T2*, *flair* (Fluid Attenuation Inversion Recovery), etc. Οι διαφορετικές μορφές μπορούν να παρέχουν συμπληρωματικές πληροφορίες για την ανάλυση διαφορετικών υποπεριφερειών των γλοιωμάτων, όπως οι πυρήνες των όγκων και οι περιοχές οιδήματος. Η τμηματοποίηση του όγκου του εγκεφάλου από τη μαγνητική τομογραφία βοηθάει στη βελτιωμένη διάγνωση, πρόβλεψη του ρυθμού ανάπτυξης και στον προγραμματισμό της θεραπείας του εγκεφαλικού όγκου, όπως επίσης είναι ζωτικής σημασίας για την παρακολούθηση της ανάπτυξης του όγκου ή της συρρίκνωσης σε ασθενείς κατά τη διάρκεια της θεραπείας. Ακόμη είναι πολύ σημαντικό για το χειρουργικό σχεδιασμό ή τον σχεδιασμό της ακτινοθεραπείας. Η ταχεία και ακριβής κατάτμηση ενός όγκου στον εγκέφαλο είναι μια μη-τετριμμένη εργασία. Οι δυσκολίες για την κατάτμηση του όγκου στον εγκέφαλο οφείλονται κυρίως σε υψηλή μεταβολή όγκων εγκεφάλου σε μέγεθος, σχήμα, κανονικότητα, τοποθεσία και την ετερογενή εμφάνισή τους (Wang, 2018). Αυτή η εγγενής ανομοιογένεια των γλοιωμάτων απεικονίζεται επίσης στον φαινότυπο απεικόνισης (εμφάνιση και σχήμα), καθώς οι υπο-περιοχές τους περιγράφονται με μεταβαλλόμενα προφίλ έντασης που διαδίδονται σε πολλαπλές σαρώσεις MRI, αντανακλώντας ποικίλες βιολογικές ιδιότητες του όγκου.



**Σχήμα 1.3:** Αξονική τομή από MRI εγκεφάλου που εμφανίζει αριστερό όγκο. Αυτός ο όγκος είναι κακοήθης και ονομάζεται γλιοβλάστωμα. Απαιτεί χειρουργική επέμβαση, ακτινοθεραπεία και χημειοθεραπεία για να θεραπευτεί..

Η Μαγνητική Απεικόνιση Συντονισμού (Magnetic Resonance Imaging-MRI) δεν δείχνει μόνο τις λεπτομερείς και πλήρεις πτυχές των εγκεφαλικών όγκων, αλλά βοηθάει τους κλινικούς γιατρούς να μελετήσουν τον μηχανισμό των όγκων του εγκεφάλου για τη βελτίωση της θεραπείας. Στην ημι-αυτόματη κατάτμηση όγκου στον εγκέφαλο, ο χρήστης πρέπει να εισάγει μερικές παραμέτρους και είναι υπεύθυνος για την ανάλυση οπτικών πληροφοριών και παροχής ανάδρασης για τον υπολογισμό του λογισμικού. Ο υπολογισμός του λογισμικού στοχεύει στην πραγματοποίηση του αλγορίθμου κατάτμησης του εγκεφαλικού όγκου. Η αλληλεπίδραση ελέγχει την προσαρμογή των πληροφοριών κατακερματισμού μεταξύ του χρήστη και του λογισμικού. Αν και η ημι-αυτόματη κατάτμηση εγκεφαλικού όγκου έχει καλύτερα αποτελέσματα από τη μη-αυτόματη κατάτμηση, μπορούν ωστόσο να προκύψουν διαφορετικά αποτελέσματα από διαφορετικούς εμπειρογνώμονες ή από τον ίδιο χρήστη σε διαφορετικές ώρες. Ως εκ τούτου, είναι αναγκαίες οι μέθοδοι πλήρως αυτόματης κατάτμησης όγκου στον εγκέφαλο. Για αυτή την αυτόματη τμηματοποίηση

ο υπολογιστής είναι αυτός που καθορίζει την κατάτμηση, χωρίς ανθρώπινη αλληλεπίδραση. Σε γενικές γραμμές, ένας πλήρως αυτόματος αλγόριθμος κατάτμησης συνδυάζει την τεχνητή νοημοσύνη μαζί με την προηγούμενη γνώση. Η ανάπτυξη αλγορίθμων μηχανικής μάθησης, όπως ήδη έχει αναφερθεί, δίνει τη δυνατότητα προσομοίωσης της ανθρώπινης νοημοσύνης για την αποτελεσματικότερη γνώση (Moreno Lopez, 2017). Η μηχανική μάθηση λοιπόν παρέχει έναν αποτελεσματικό τρόπο αυτοματοποίησης της ανάλυσης και τη διάγνωσης ιατρικών εικόνων. Μπορεί ενδεχομένως να μειώσει το βάρος σε ακτινολόγους στην πρακτική της ακτινολογίας, μπορεί να μάθει σύνθετες σχέσεις ή μοντέλα από εμπειρικά δεδομένα και να λαμβάνει ακριβείς αποφάσεις.



# Κεφάλαιο 2

## Θεωρία

### 2.1 Τεχνητά Νευρωνικά Δίκτυα

Τα τεχνητά νευρωνικά δίκτυα (Artificial Neural Networks - ANN) όπως έχει ήδη αναφερθεί στην ενότητα 1.3 είναι αλγόριθμοι που μιμούνται τη βιολογική δομή του εγκεφάλου. Αυτά τα συστήματα “μαθαίνουν” να εκτελούν εργασίες εξετάζοντας παραδείγματα γενικά χωρίς να προγραμματίζονται με ειδικούς κανόνες. Για παράδειγμα, στην αναγνώριση εικόνων μπορεί να μάθουν να εντοπίζουν εικόνες που περιέχουν γάτες αναλύοντας παραδείγματα εικόνων που έχουν επισημανθεί με το χέρι ως “γάτα” ή “όχι γάτα” και χρησιμοποιώντας τα αποτελέσματα για τον εντοπισμό των γάτων σε άλλες εικόνες. Το κάνουν αυτό χωρίς προηγούμενη γνώση για τις γάτες, π.χ. ότι έχουν γούνα, ουρές, μουστάκια και πρόσωπα που μοιάζουν με γάτες. Αντ’αυτού δημιουργούν αυτόματα χαρακτηριστικά ταυτοποίησης από το μαθησιακό υλικό πάνω στο οποίο εκπαιδεύονται.

Ένα Τεχνητό Νευρωνικό Δίκτυο βασίζεται σε μια συλλογή από συνδεδεμένες μονάδες ή κόμβους που ονομάζονται τεχνητοί νευρώνες και μοντελοποιούν τους νευρώνες ενός βιολογικού εγκεφάλου. Κάθε σύνδεση, όπως οι συνάψεις σε έναν βιολογικό εγκέφαλο, μπορεί να μεταδώσει ένα σήμα από έναν τεχνητό νευρώνα στον άλλο. Ένας τεχνητός νευρώνας που λαμβάνει ένα σήμα μπορεί να το επεξεργαστεί και στη συνέχεια να σηματοδοτήσει πρόσθετους τεχνητούς νευρώνες που συνδέονται με αυτό.

Στις κοινές εφαρμογές, το σήμα σε μια σύνδεση μεταξύ των τεχνητών νευρώνων είναι ένας πραγματικός αριθμός και η έξοδος κάθε τεχνητού νευρώνα υπολογίζεται από κάποια μη-γραμμική συνάρτηση από το άθροισμα των εισόδων του. Οι συνδέσεις μεταξύ των τεχνητών νευρώνων ονομάζονται “άκρες”. Οι τεχνητοί νευρώνες και οι άκρες έχουν συνήθως ένα βάρος που προσαρμόζεται όπως προχωρά η εκπαίδευση. Το βάρος αυξάνει ή μειώνει την ισχύ του σήματος σε μια σύνδεση. Οι τεχνητοί νευρώνες μπορεί να έχουν ένα κατώφλι τέτοιο ώστε το σήμα να αποστέλλεται μόνο αν το συνολικό σχήμα διασχίσει αυτό το όριο-κατώφλι. Τυπικά, οι τεχνητοί νευρώνες συσσωματώνονται σε επίπεδα (layers). Τα διαφορετικά επίπεδα μπορούν να εκτελούν διαφορετικά είδη μετασχηματισμών στις εισόδους τους. Τα σήματα ταξιδεύουν από το πρώτο επίπεδο (το επίπεδο εισόδου), μέχρι το τελευταίο επίπεδο (το επίπεδο εξόδου), πιθανώς μετά από πολλαπλές διαδρομές.

Ο αρχικός στόχος της προσέγγισης των Τεχνητών Νευρωνικών Δικτύων ήταν να λυθούν τα προβλήματα με τον ίδιο τρόπο με τον ανθρώπινο εγκέφαλο. Ωστόσο με την πάροδο του χρόνου, η προσοχή μετακινήθηκε στην εκτέλεση συγκεκριμένων καθηκόντων, οδηγώντας σε αποκλίσεις από τη βιολογία. Τα Τεχνητά Νευρωνικά

Δίκτυα έχουν χρησιμοποιηθεί σε ποικίλες εργασίες, όπως η όραση υπολογιστών, αναγνώριση ομιλίας, φιλτράρισμα κοινωνικών δικτύων, τα βιντεοπαιχνίδια και η ιατρική διάγνωση (van Gerven and Bohte, 2018).

### 2.1.1 Perceptron

Στη μηχανική μάθηση, το **perceptron** είναι ένας αλγόριθμος για την επιβλεπόμενη μάθηση των δυαδικών ταξινομητών (λειτουργίες που μπορούν να αποφασίσουν εάν μια είσοδος, που αντιπροσωπεύεται από έναν φορέα αριθμών, ανήκει σε κάποια κατηγορία ή όχι) (van Gerven and Bohte, 2018).

Συγκεκριμένα, το 1958, ο ερευνητής ψυχολόγος από το Cornell Frank Rosenblatt πρότεινε ένα πρότυπο νευρωνικό δίκτυο, το Perceptron, για το οποίο χρησιμοποίησε έναν υπολογιστή με τρυπητή κάρτα που έπιανε ένα ολόκληρο δωμάτιο. Μετά από 50 δοκιμές έμαθε να διακρίνει ανάμεσα σε κάρτες που σημειώνονται στα αριστερά τους και κάρτες που σημειώνονται στα δεξιά τους.

Το μοντέλο Perceptron δημιουργήθηκε μόνο από έναν νευρώνα. Η είσοδος του νευρώνα είναι ένα διάνυσμα  $x \in \mathbb{R}$  και παράγει μια έξοδο  $a \in \mathbb{R}$ , χρησιμοποιώντας τα ακόλουθα δύο βήματα:

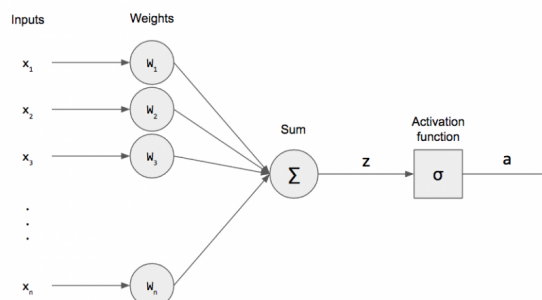
1. Κάθε στοιχείο  $x_i$  του διανύσματος εισόδου  $x$  πολλαπλασιάζεται με ένα βάρος σύνδεσης  $w_i$  (connection weight) και τα αποτελέσματα προστίθενται. Υπάρχει ένας όρος bias-πόλωση  $b \in \mathbb{R}$  που προστίθεται στο άθροισμα. Αυτός είναι προφανώς ένας γραμμικός μετασχηματισμός.

$$z(x) = x^T w = \sum_{i=1}^n w_i x_i + b \quad (2.1)$$

2. Η τιμή του  $z(x)$  δίνεται σαν μια παράμετρο σε μια μη-γραμμική συνάρτηση ενεργοποίησης  $\sigma(\cdot)$ , με αποτέλεσμα την τελική έξοδο  $a$  του συστήματος. Αυτό είναι το μη-γραμμικό μέρος της διαδικασίας.

$$a = \sigma\left(\sum_{i=1}^n w_i x_i + b\right) \quad (2.2)$$

Το μοντέλο Perceptron συνοψίζεται με το ακόλουθο σχήμα.



**Σχήμα 2.1:** Το μοντέλο Perceptron.

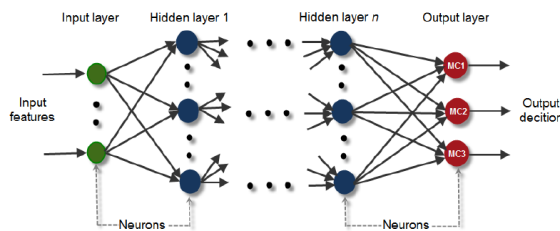


## 2.1.2 Multilayer Perceptron (MLP)

Ένα πολυεπίπεδο (Multilayer) Perceptron (MLP) είναι μια κατηγορία του τεχνητού νευρωνικού δικτύου. Ένα MLP αποτελείται από τουλάχιστον τρία επίπεδα κόμβων. Η τυπική τους αρχιτεκτονική είναι ένα επίπεδο εισόδου, ακολουθούμενο από ένα ή περισσότερα κρυφά επίπεδα και στη συνέχεια ένα επίπεδο εξόδου. Ένα MLP λέγεται ότι έχει βάθος  $D$  και μπορεί να ονομαστεί νευρωνικό δίκτυο επιπέδου  $D$ , όταν έχει  $D-1$  κρυφά επίπεδα και ένα επίπεδο εξόδου. Το επίπεδο εξόδου δεν συμπεριλαμβάνεται όταν υπολογίζεται το βάθος ενός νευρωνικού δικτύου. Το διάνυσμα εξόδου όλων των νευρώνων σε ένα κρυφό επίπεδο ή σε ένα επίπεδο εξόδου μπορεί να υπολογιστεί χρησιμοποιώντας την παρακάτω εξίσωση:

$$y_\ell = \phi(W_\ell y_{\ell-1} + b_\ell) \quad (2.3)$$

όπου  $W_\ell$  είναι ένας πίνακας  $N \times M$  που περιέχει ένα  $M$ -διαστάσεως διανυσματικό βάρος (υπάρχουν  $M$  νευρώνες στο επίπεδο  $\ell - 1$ ) για καθένα από τους  $N$  νευρώνες στο επίπεδο  $\ell$ ,  $y_{\ell-1}$  είναι το  $M$ -διαστάσεως διάνυσμα εξόδου του επιπέδου  $\ell - 1$ ,  $b_\ell$  είναι το  $N$ -διαστάσεως διάνυσμα της πόλωσης (bias) και το  $\phi(\cdot)$  είναι μια μη-γραμμική συνάρτηση ενεργοποίησης.



**Σχήμα 2.2:** Στην εικόνα αυτή υπάρχει ένα MLP με  $(n+1)$  επίπεδα. Αυτό το δίκτυο είναι Πλήρως Συνδεδεμένο Δίκτυο (Fully Connected Network (FCN)), μιας και κάθε νευρώνας είναι συνδεδεμένος με όλους τους νευρώνες από το προηγούμενο επίπεδο.

Τα πολλαπλά επίπεδα και η μη γραμμική ενεργοποίησή του διακρίνουν το MLP από ένα γραμμικό perceptron. Τα MLP χρησιμοποιούν μια επιβλεπόμενη τεχνική μάθησης που ονομάζεται **ανάστροφη διάδοση (backpropagation)** για εκπαίδευση (Rosenblatt, 1961).

- **Ανάστροφη Διάδοση-Backpropagation**

Αν και ο αλγόριθμος Ανάστροφης Διάδοσης του σφάλματος είχε περιγραφεί πολύ νωρίς, σε κάποιες δημοσιεύσεις, όπως στο (Linnainmaa, 1976), δεν ήταν παρά μέχρι το 1982 που περιγράφεται στο πλαίσιο των νευρωνικών δικτύων από τον Werbos και μετά μέχρι το 1986 που η σημασία της εκπαίδευσης των νευρωνικών δικτύων επιβεβαιώθηκε από πειράματα που διεξήχθησαν από τους (Rumelhart et al., 1986). Σε αυτή την ενότητα θα περιγραφεί ο αλγόριθμος της ανάστροφης διάδοσης και πως χρησιμοποιείται κατά τη διάρκεια της διαδικασίας εκπαίδευσης ενός νευρωνικού δικτύου. Ωστόσο, πρέπει πρώτα να εισαχθούν δύο μαθηματικά εργαλεία, ο κανόνας της αλυσίδας και οι υπολογιστικές γραφικές παραστάσεις, που είναι θεμελιώδεις για τον αλγόριθμο ανάστροφης διάδοσης. Με τη βοήθεια αυτών των δύο εργαλείων,

ο αλγόριθμος του Backpropagation μπορεί να συνοψιστεί χρησιμοποιώντας μόνο τέσσερις εξισώσεις.

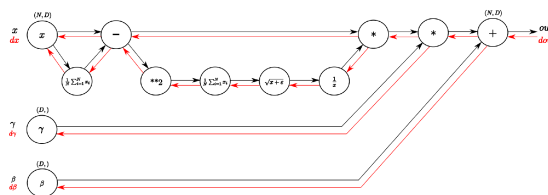
Η υπολογιστική διαδικασία κατά την οποία ένα νευρωνικό δίκτυο χρησιμοποιεί ένα διάνυσμα εισόδου  $x$  για να κάνει μια πρόβλεψη  $y$  ονομάζεται Forward Propagation - εμπρόσθια διάδοση. Ο αλγόριθμος Backpropagation υπολογίζει την απώλεια ή το κόστος που προκύπτει από τη διαφορά μεταξύ της πρόβλεψης  $y$  και της ετικέτας  $\hat{y}$ , και μεταδίδει αυτό το σφάλμα σε προηγούμενα επίπεδα της αρχιτεκτονικής έτσι ώστε να υπολογίσει την παράγωγο του σφάλματος σε σχέση με όλες τις παραμέτρους του μοντέλου. Ο κανόνας της αλυσίδας, που φαίνεται να έχει χρησιμοποιηθεί αρχικά από τον Leibniz, διαδραματίζει πολύ σημαντικό ρόλο κατά τη διάρκεια αυτής της διαδικασίας. Ειδικότερα, εάν  $x \in R^n$ ,  $z \in R$ ,  $y = g(x)$  και  $z = f(y) = f(g(x))$ , τότε ο κανόνας της αλυσίδας υποδηλώνει ότι:

$$\frac{\partial z}{\partial x_i} = \sum_{j=1}^n \frac{\partial z}{\partial y_j} \frac{\partial y_j}{\partial x_i} \quad (2.4)$$

Στη ρύθμιση του νευρωνικού δικτύου θα μπορούσε κάποιος να ερμηνεύσει το  $x$  ως το διάνυσμα χαρακτηριστικών,  $y$  ως την έξοδο ή το διάνυσμα πρόβλεψης του δικτύου,  $g(\cdot)$  ως τη συνάρτηση που υλοποιείται από το δίκτυο η οποία αντιστοιχεί τα χαρακτηριστικά  $x$  στις προβλέψεις  $y$  και  $z$  το κλιμακώμενο κόστος υπολογισμένο χρησιμοποιώντας τη συνάρτηση σφάλματος  $f(\cdot)$ . Γενικά, ο κανόνας της αλυσίδας μπορεί να εφαρμοστεί ακόμα κι αν τα  $\mathbf{X}$  και  $\mathbf{Y}$  είναι τανυστές (tensors), το  $g(\cdot)$  είναι μια συνάρτηση αντιστοίχισης ενός τανυστή  $\mathbf{X}$  σε ένα τανυστή  $\mathbf{Y}$  και  $f(\cdot)$  είναι μια συνάρτηση αντιστοίχισης ενός τανυστή  $\mathbf{Y}$  σε μια κλιμακώμενη τιμή  $z$ . Αυτή η ρύθμιση είναι χαρακτηριστική σε ένα CNN, το οποίο εκτελεί μετασχηματισμούς με τανυστές. Ο κανόνας της αλυσίδας θα έχει τότε την ακόλουθη μορφή:

$$\nabla x = \sum_j \nabla x_j \frac{\partial z}{\partial Y_j} \quad (2.5)$$

Ένα υπολογιστικό γράφημα είναι ένας κατευθυνόμενος γράφος όπου οι κόμβοι αντιστοιχούν σε λειτουργίες ή μεταβλητές. Οι μεταβλητές μπορούν να τροφοδοτήσουν την αξία τους στις λειτουργίες, και οι λειτουργίες μπορούν να τροφοδοτήσουν την έξοδο τους μέσα σε άλλες λειτουργίες. Με αυτόν τον τρόπο, κάθε κόμβος στο γράφημα ορίζει μια συνάρτηση των μεταβλητών. Επομένως, όλα τα νευρωνικά δίκτυα μπορούν να αναπαρασταθούν χρησιμοποιώντας ένα υπολογιστικό γράφημα. Επίσης, στην περίπτωση εμπρόσθιας διάδοσης, το γράφημα δεν περιέχει κανέναν κύκλο.



**Σχήμα 2.3:** Ένας υπολογιστικός γράφος αντιπροσωπεύει ένα νευρωνικό δίκτυο και μαζί ένα εμπρόσθιο και ένα ανάστροφο πέρασμα κατά τη διάρκεια της εκπαίδευσης. Οι μαύρες γραμμές υποδηλώνουν την εμπρόσθια διάδοση και οι κόκκινες γραμμές υποδηλώνουν τη διαδικασία της ανάστροφης διάδοσης.

Ο αλγόριθμος της ανάστροφης διάδοσης είναι αυτός που βοηθάει στο συντονισμό των παραμέτρων του νευρωνικού δικτύου (βάρη και πολώσεις), προκειμένου να ελαχιστοποιηθεί η αξία της αντικειμενικής συνάρτησης - συνάρτησης κόστους (objective function - loss function). Τελικά, αυτό οδηγεί στον υπολογισμό των μερικών παραγώγων  $\partial J/\partial W^\ell$  και  $\partial J/\partial b^\ell$  για κάθε επίπεδο  $\ell$  στο δίκτυο. Για την εφαρμογή του αλγορίθμου ανάστροφης διάδοσης είναι αναγκαίο να κατασκευαστεί ο υπολογιστικός γράφος του δικτύου και να εκτελέσει μια εμπρόσθια διάδοση στο γράφημα. Κατά τη διάρκεια της προώθησης, το σήμα εισόδου μεταδίδεται από τα ρηχότερα στα πιο βαθύτερα επίπεδα της αρχιτεκτονικής, οδηγώντας σε ένα σήμα εξόδου στο τελικό επίπεδο του δικτύου. Οι ενδιάμεσες τιμές του σήματος αποθηκεύονται στα σχετικά επίπεδα του δικτύου. Ειδικότερα, ο υπολογιστικός γράφος του δικτύου, έχει ένα τελικό επίπεδο  $u_n$ , που παριστάνει τη τελική απώλεια  $J$ . Η αξία του  $u_n$  εξαρτάται από τα επίπεδα  $u_1, \dots, u_n$ , οπότε πρέπει να υπολογιστούν όλες οι παράγωγοι  $\partial u_n/\partial u_i$ , όπου  $i \in \{1, \dots, n\}$ . Ωστόσο, κάθε μία από αυτές τις τιμές μπορεί επίσης να έχει προηγούμενες εξαρτήσεις στο γράφο, οπότε θα πρέπει να υπολογιστούν οι παράγωγοι που συλλαμβάνουν αυτές τις εξαρτήσεις. Αυτό γίνεται υπολογίζοντας όλα τα παράγωγα με τον αλγόριθμο της ανάστροφης διάδοσης. Πριν από την έναρξη του αλγορίθμου, όλα τα επίπεδα στον υπολογιστικό γράφο έχουν τιμές, που λαμβάνονται από την εμπρόσθια διάδοση του σήματος εισόδου.

Έστω  $g[i]$  ένας πίνακας, όπου είναι αποθηκευμένες οι κλίσεις  $\partial u_n/\partial u_i$ , τότε ο αλγόριθμος ανάστροφης διάδοσης μπορεί να συνοψιστεί χρησιμοποιώντας τρία βήματα:

1. Υπολογισμός της παραγώγου του επιπέδου  $u_n$  σε συνάρτηση με τον ευατό της, η οποία είναι 1 και αποθήκευση της τιμής αυτή σε ένα πίνακα  $g$ .
2. Για κάθε επίπεδο  $u_i$ , όπου  $i$  ξεκινά από το  $n-1$  και πάει μέχρι το 1, υπολογίζεται  $\partial u_n/\partial u_i = \sum_{j:i \in \text{ancestors}(u_j)} \frac{\partial u_n}{\partial u_j} \frac{\partial u_j}{\partial u_i}$  και το αποθηκεύει σε ένα πίνακα  $g$ .
3. Επιστρέφονται οι τιμές  $g[u_i]$ , όπου  $i \in \{1, \dots, n\}$ .

Η εκμάθηση στο perceptron γίνεται μεταβάλλοντας τα βάρη σύνδεσης μετά από κάθε επεξεργασία δεδομένων, με βάση το ποσό σφάλματος στην έξοδο σε σύγκριση με το αναμενόμενο αποτέλεσμα. Αυτό είναι ένα παράδειγμα επιβλεπόμενης μάθησης και πραγματοποιείται μέσω της ανάστροφης διάδοσης, όπως ήδη περιγράφηκε.

### • Θεώρημα Καθολικής Προσέγγισης-Universal Approximation Theorem

Στη μαθηματική θεωρία των τεχνητών νευρωνικών δικτύων, το θεώρημα της καθολικής προσέγγισης δηλώνει (Csáji, 2001) ότι ένα δίκτυο εμπρόσθιας διάδοσης με ένα απλό κρυφό επίπεδο που περιέχει ένα πεπερασμένο αριθμό νευρώνων μπορεί να προσεγγίσει τις συνεχείς συναρτήσεις σε συμπαγή υποσύνολα του  $R^n$ , κάτω από ήπιες υποθέσεις για τη συνάρτηση ενεργοποίησης.

Το θεώρημα δηλώνει έτσι ότι τα απλά νευρωνικά δίκτυα μπορούν να αντιπροσωπεύουν μια ευρεία ποικιλία ενδιαφέροντων λειτουργιών όταν δίδονται οι κατάλληλες παράμετροι. Ωστόσο, δεν αγγίζει την αλγοριθμική μάθηση των παραμέτρων αυτών.

Μια από τις πρώτες εκδοχές του θεωρήματος αποδείχθηκε από τον George Cybenko το 1989 για τη σιγμοειδή συνάρτηση ενεργοποίησης (Cybenko, 1989).

Ο Kurt Hornik έδειξε το 1991 (Hornik, 1991) ότι δεν είναι η συγκεκριμένη επιλογή της συνάρτησης ενεργοποίησης, αλλά η ίδια αρχιτεκτονική πολλαπλών

επιπέδων, η οποία δίνει στα νευρωνικά δίκτυα τη δυνατότητα να μπορούν να προσεγγίσουν καθολικά μια λειτουργία.

Το θεώρημα με μαθηματικούς όρους:

Έστω  $\phi(\cdot)$  είναι μια μη-σταθερή, οριοθετημένη και μονοτονικά-αυξανόμενη συνεχής συνάρτηση. Έστω  $I_m$  υποδηλώνει τη  $m$ -διαστάσεων μονάδα hypercube<sup>1</sup>  $[0, 1]^m$ . Ο χώρος των συνεχών συναρτήσεων  $I_m$  συμβολίζεται με  $C(I_m)$ . Τότε, με δεδομένο κάθε  $\varepsilon > 0$  και κάθε συνάρτηση  $f \in C(I_m)$ , υπάρχει ένας ακέραιος  $N$ , πραγματικές σταθερές  $v_i, b_i \in \mathbb{R}$  και πραγματικά διανύσματα  $w_i \in \mathbb{R}^m$ , όπου  $i = 1, \dots, N$ , τέτοιο ώστε να ορίζεται (Cybenko, 1989):

$$F(x) = \sum_{i=1}^N v_i \phi(w_i^T x + b_i) \quad (2.6)$$

ως μια κατά προσέγγιση υλοποίηση της συνάρτησης  $f$ , όπου  $f$  είναι ανεξάρτητη από τη  $\phi$ , δηλαδή,

$$|F(x) - f(x)| < \varepsilon \quad (2.7)$$

για κάθε  $x \in I_m$ . Με άλλα λόγια, οι συναρτήσεις της μορφής  $F(x)$  είναι πυκνές dense<sup>2</sup> στο  $C(I_m)$ .

### 2.1.3 Overfitting-Imbalance

Η επιβλεπόμενη μηχανική μάθηση κατανοείται καλύτερα προσεγγίζοντας μια συνάρτηση στόχου (target function- $f$ ) που αντιστοιχεί τις μεταβλητές εισόδου ( $X$ ) σε μια μεταβλητή εξόδου  $Y$ .

$$Y = f(X) \quad (2.8)$$

Ένα σημαντικό στοιχείο στην εκπαίδευση της συνάρτησης στόχου από τα δεδομένα εκπαίδευσης (training test) είναι το πόσο καλά γενικεύεται το μοντέλο σε νέα δεδομένα. Η γενίκευση είναι σημαντική επειδή τα δεδομένα που συλλέγουμε είναι μόνο ένα δείγμα, είναι ατελή και θορυβώδη.

Η γενίκευση αναφέρεται στο πόσο καλά εφαρμόστηκαν οι έννοιες που αποκτήθηκαν από ένα μοντέλο μηχανικής μάθησης σε συγκεκριμένα παραδείγματα που δεν έχει ξαναδεί το μοντέλο όταν εκπαιδευόταν.

Πολλές φορές παρουσιάζεται το φαινόμενο, ενώ το μοντέλο αποδίδει καλά να μη γενικεύει. Προκειμένου να έχουμε μια αντικειμενική εικόνα της απόδοσης του αλγορίθμου στην πράξη χωρίζουμε το σύνολο των δεδομένων που έχουμε στη διάθεση μας σε τρία υποσύνολα.

- **Σύνολο Εκπαίδευσης (Train Set)**

<sup>1</sup>Ένας κύβος μονάδας, τυπικά ένας κύβος πλευράς 1, είναι ένας κύβος των οποίων οι πλευρές έχουν μήκος 1 μονάδα. Ο όγκος ενός τρισδιάστατου κύβου μονάδας είναι 1 κυβική μονάδα και η συνολική επιφάνεια του είναι 6 τετραγωνικές μονάδες. Ο όρος κύβος μονάδας ή μονάδα hypercube-υπερκύβου χρησιμοποιείται επίσης και για υπερκύβους ή "κύβους"  $n$ -διαστάσεων, για  $n$  διαφορετικές από 3 και μήκος ακμής 1.

<sup>2</sup>Στην τοπολογία και σε συναφείς τομείς των μαθηματικών, ένα υποσύνολο  $A$  ενός τοπολογικού χώρου  $X$  ονομάζεται πυκνό (σε  $X$ ), αν κάθε σημείο  $x$  στο  $X$  είτε ανήκει στο  $A$  είτε είναι οριακό σημείο του  $A$ , δηλαδή το κλείσιμο του  $A$  αποτελεί το ολόκληρο σύνολο  $X$  (Steen et al., 1978).

Ένα σύνολο δεδομένων εκπαίδευσης είναι ένα σύνολο δεδομένων από παραδείγματα που χρησιμοποιούνται για τη μάθηση, δηλαδή ένα σύνολο παραδειγμάτων που χρησιμοποιούνται για την προσαρμογή των παραμέτρων (π.χ. τα βάρη-weights των συνδέσεων μεταξύ νευρώνων στα τεχνητά νευρωνικά δίκτυα) του μοντέλου (Ripley, 2007). Το μοντέλο (π.χ. ένα νευρωνικό δίκτυο) εκπαιδεύεται στο σύνολο δεδομένων εκπαίδευσης χρησιμοποιώντας μια εποπτευόμενη μέθοδο εκμάθησης. Στην πράξη, το σύνολο δεδομένων εκπαίδευσης αποτελείται συχνά από ζεύγη διάνυσματος εισόδου και το αντίστοιχο διάνυσμα απάντησης, το οποίο συνήθως υποδηλώνεται ως ο στόχος. Το τρέχων μοντέλο τρέχει με το σύνολο δεδομένων εκπαίδευσης και παράγει ένα αποτέλεσμα, το οποίο στη συνέχεια συγκρίνεται με τον στόχο, για κάθε διάνυσμα εισόδου στο σύνολο δεδομένων εκπαίδευσης. Με βάση το αποτέλεσμα της σύγκρισης και τον συγκεκριμένο αλγόριθμο μάθησης που χρησιμοποιείται, οι παράμετροι του μοντέλου ρυθμίζονται. Η προσαρμογή του μοντέλου μπορεί να περιλαμβάνει τόσο την επιλογή μεταβλητών όσο και την εκτίμηση των παραμέτρων.

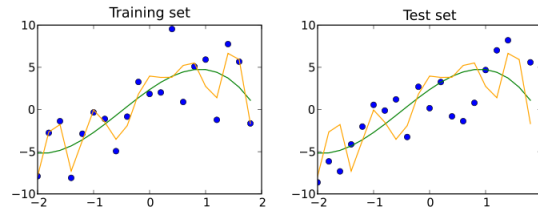
- **Σύνολο Επαλήθευσης (Validation Set)**

Στη συνέχεια, το προσαρμοσμένο μοντέλο χρησιμοποιείται για την πρόβλεψη των απαντήσεων για τις παρατηρήσεις σε ένα δεύτερο σύνολο δεδομένων που ονομάζεται σύνολο δεδομένων επαλήθευσης (James et al., 2013). Το σύνολο δεδομένων επαλήθευσης παρέχει μια αμερόληπτη αξιολόγηση (unbiased evaluation) ενός μοντέλου που προσαρμόζει το σύνολο δεδομένων εκπαίδευσης, ενώ συντονίζει τις υπερπαραμέτρους του μοντέλου (π.χ. τον αριθμό των κρυφών μονάδων σε ένα νευρωνικό δίκτυο (Ripley, 2007)). Τα σύνολα δεδομένων επαλήθευσης μπορούν να χρησιμοποιηθούν για την τακτοποίηση με την πρόωρη διακοπή: διακοπή εκπαίδευσης όταν αυξάνεται το σφάλμα στο σύνολο δεδομένων επαλήθευσης, καθώς αυτό αποτελεί ένδειξη υπερφόρτωσης στο σύνολο δεδομένων εκπαίδευσης (Prechelt, 1998). Αυτή η απλή διαδικασία περιπλέκεται στην πράξη από το γεγονός ότι το σφάλμα του συνόλου δεδομένων επαλήθευσης μπορεί να κυμαίνεται κατά τη διάρκεια της εκπαίδευσης, παράγοντας πολλαπλά τοπικά ελάχιστα. Αυτή η επιπλοκή οδήγησε στη δημιουργία πολλών κανόνων για να αποφασιστεί πότε ξεκίνησε πραγματικά η υπερφόρτωση (Prechelt, 1998).

- **Σύνολο Δοκιμής (Test Set)**

Ένα σύνολο δεδομένων δοκιμής είναι ένα σύνολο δεδομένων που είναι ανεξάρτητο από το σύνολο δεδομένων εκπαίδευσης, αλλά ακολουθεί την ίδια κατανομή πιθανότητας με το σύνολο δεδομένων εκπαίδευσης. Αν ένα μοντέλο που προσαρμόζει σωστά στο σύνολο δεδομένων εκπαίδευσης επίσης προσαρμόζει σωστά στο σύνολο δεδομένων δοκιμής, τότε έχει σημειωθεί ελάχιστη υπερπροσαρμογή. Επομένως, ένα σύνολο δοκιμών είναι ένα σύνολο παραδειγμάτων που χρησιμοποιούνται μόνο για την αξιολόγηση της απόδοσης (δηλαδή της γενίκευσης) ενός πλήρως καθορισμένου ταξινομητή.

Στην ουσία αυτό το σύνολο δεδομένων χρησιμοποιείται μόνο για τη δοκιμή της τελικής λύσης προκειμένου να επιβεβαιωθεί η πραγματική ισχύς πρόβλεψης του δικτύου.



**Σχήμα 2.4:** Ένα σύνολο εκπαίδευσης (αριστερά) και ένα σύνολο δοκιμών (δεξιά) από τον ίδιο στατιστικό πληθυσμό εμφανίζονται ως μπλε σημεία. Δύο προγνωστικά μοντέλα είναι κατάλληλα για τα δεδομένα εκπαίδευσης. Και τα δύο προσαρμοσμένα μοντέλα σχεδιάζονται με τα σύνολα δεδομένων εκπαίδευσης και δοκιμής. Στο σύνολο εκπαίδευσης, το Μέσο Τετραγωνικό Σφάλμα  $MSE$ - $Mean Squared Error$  της προσαρμογής που εμφανίζεται με πορτοκαλί χρώμα είναι 4, ενώ το  $MSE$  για την προσαρμογή εμφανίζεται με πράσινο και είναι 9. Στο σύνολο δοκιμών το  $MSE$  για την προσαρμογή με πορτοκαλί είναι 15 και για την προσαρμογή με πράσινο είναι 13. Η πορτοκαλί καμπύλη υπερβαίνει τα δεδομένα εκπαίδευσης, καθώς το  $MSE$  αυξάνεται σχεδόν κατά τέσσερις φορές όταν συγκρίνεται το σύνολο δοκιμής με το σύνολο εκπαίδευσης. Η πράσινη καμπύλη υπερβαίνει τα στοιχεία της εκπαίδευσης πολύ λιγότερο, καθώς το  $MSE$  της αυξάνεται κατά λιγότερο από ένα συντελεστή 2.

Ο στόχος ενός καλού μοντέλου μηχανικής μάθησης είναι να γενικεύσει από τα δεδομένα εκπαίδευσης σε οποιαδήποτε δεδομένα από τον τομέα προβλημάτων. Αυτό δίνει τη δυνατότητα να γίνονται προβλέψεις στο μέλλον για δεδομένα που το μοντέλο δεν έχει δει ποτέ.

Υπάρχει μια ορολογία που χρησιμοποιείται στη μηχανική μάθηση σχετικά με το κατά πόσο μαθαίνει ένα μοντέλο και κατά πόσο γενικεύει σε νέα δεδομένα, μία από αυτές είναι η υπερπροσαρμογή (overfitting).

- **Υπερπροσαρμογή (Overfitting)**

- **Στατιστική Προσαρμογή (Statistical Fit)**

Στη στατιστική, μια προσαρμογή αναφέρεται στο πόσο καλά προσεγγίζεται μια λειτουργία στόχου. Αυτή είναι μια καλή ορολογία που χρησιμοποιείται στη μηχανική μάθηση, επειδή οι εποπτευόμενοι αλγόριθμοι μάθησης επιδιώκουν να προσεγγίσουν την άγνωστη συνάρτηση αντιστοίχισης για τις μεταβλητές εξόδου δεδομένου τις μεταβλητές εισόδου. Οι στατιστικές περιγράφουν συχνά την ακρίβεια της προσαρμογής η οποία αναφέρεται στα μέτρα που χρησιμοποιούνται για να εκτιμηθεί κατά πόσο καλά η προσέγγιση της συνάρτησης ταιριάζει με τη συνάρτηση στόχου.

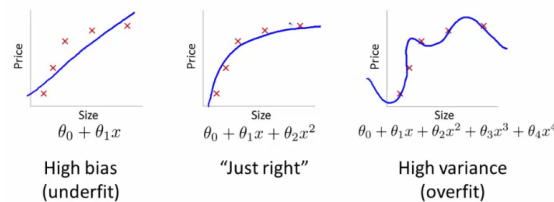
Μερικές από αυτές τις μεθόδους είναι χρήσιμες στη μηχανική μάθηση (π.χ. υπολογισμός των υπολειπόμενων σφαλμάτων), αλλά μερικές από αυτές τις τεχνικές υποθέτουν ότι είναι γνωστή η μορφή της συνάρτησης στόχου που προσεγγίζεται, κάτι που δεν συμβαίνει στη μηχανική μάθηση.

Αν είναι γνωστή η μορφή της συνάρτησης στόχου, θα χρησιμοποιηθεί άμεσα για τις προβλέψεις, αντί να γίνονται προσπάθειες να γίνει γνωστή μια προσέγγιση από δείγματα θορυβώδους δεδομένων εκπαίδευσης.

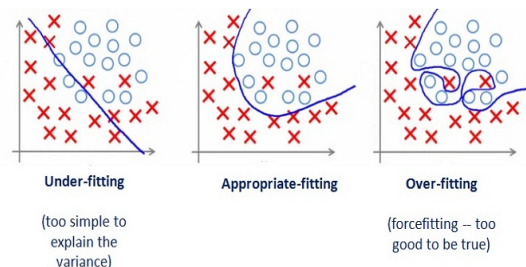
Η **υπερπροσαρμογή (overfitting)** αναφέρεται σε ένα μοντέλο που αποδίδει πολύ καλά τα δεδομένα εκπαίδευσης αλλά δε γενικεύει καλά. Η υπερπροσαρμογή συμβαίνει όταν ένα μοντέλο μαθαίνει να μοντελοποιεί την λεπτομέρεια και το θόρυβο

στα δεδομένα εκπαίδευσης, σε βαθμό που επηρεάζει αρνητικά την απόδοση του μοντέλου σε νέα δεδομένα. Αυτό σημαίνει ότι ο θόρυβος ή οι τυχαίες διακυμάνσεις στα δεδομένα εκπαίδευσης (training data) συλλέγονται και εκπαιδεύονται ως έννοιες από το μοντέλο. Το πρόβλημα είναι ότι αυτές οι έννοιες δεν ισχύουν για νέα δεδομένα και επηρεάζουν αρνητικά την ικανότητα των μοντέλων να γενικεύουν. Η υπερπροσαρμογή είναι πιθανότερη με μη-παραμετρικά και μη-γραμμικά μοντέλα που έχουν μεγαλύτερη ευελιξία και ελευθερία στην οικοδόμηση του μοντέλου με βάση το σύνολο δεδομένων και συνεπώς μπορούν πραγματικά να χτίσουν μη ρεαλιστικά μοντέλα. Ως εκ τούτου, πολλοί μη-παραμετρικοί αλγόριθμοι μάθησης μηχανών περιλαμβάνουν επίσης παραμέτρους ή τεχνικές για τον περιορισμό του πόση λεπτομέρεια μαθαίνει το μοντέλο. Μια λύση για να αποφευχθεί η υπερπροσαρμογή - overfitting είναι χρησιμοποιούνται λιγότερο πολύπλοκοι αλγόριθμοι, πχ γραμμικοί αλγόριθμοι αν τα δεδομένα είναι γραμμικά ή περιορίζοντας την πολυπλοκότητα ορισμένων αλγορίθμων πχ ορισμός του μέγιστου βάθους σε δέντρα αποφάσεων.

Στην **υποπροσαρμογή (underfitting)** σε αντιδιαστολή με την υπερπροσαρμογή το μοντέλο δεν είναι αρκετά πολύπλοκο για να εξάγει όλες τις ουσιώδεις σχέσεις από τα δεδομένα.



Σχήμα 2.5: Παράδειγμα υπερπροσαρμογής.



Σχήμα 2.6: Παράδειγμα υπερπροσαρμογής.

## 2.1.4 Ανισορροπία Κλάσεων - Imbalance classes

Οι περισσότερες ταξινομήσεις στον πραγματικό κόσμο εμφανίζουν κάποιο επίπεδο **ανισορροπίας κλάσης (class imbalance)**, το οποίο σημαίνει ότι δεν αντιπροσωπεύεται κάθε κλάση με μια ισότιμη μερίδα του συνόλου δεδομένων. Είναι σημαντικό όμως να προσαρμοστούν σωστά οι μετρήσεις και οι μέθοδοι στους στόχους της κάθε έρευνας. Για παράδειγμα, έστω ότι υπάρχουν 2 κλάσεις - A και B. Η κλάση A είναι το 90% του συνόλου δεδομένων και η κλάση B είναι το άλλο 10%, αλλά το μεγαλύτερο ενδιαφέρον είναι για τον εντοπισμό περιπτώσεων κλάσης B. Είναι εύκολο να επιτευχθεί ακρίβεια 90%, απλά προβλέποντας την κλάση A κάθε φορά, αλλά αυτό δεν προσφέρει κάτι στον αρχικό στόχο της έρευνας. Αντ'αυτού, μια κατάλληλη βαθμονομημένη μέθοδος μπορεί να επιτύχει μια χαμηλότερη ακρίβεια, αλλά θα έχει πολύ υψηλότερο πραγματικό θετικό ρυθμό (true positive rate) (ή ανάκληση-recall).

Στην συγκεκριμένη εργασία αντιμετωπίζεται και εδώ το πρόβλημα της ανισοροπίας κλάσης, στην ενότητα 2.3.2 γίνεται αναφορά σε μετρικές που είναι πιο αντικειμενικές όταν υπάρχει ανισοροπία κλάσεων.

## 2.2 Συνελικτικά Νευρωνικά Δίκτυα

Τα συνελικτικά νευρωνικά δίκτυα (Convolutional Neural Networks CNNs ή ConvNets) εισήχθησαν το 1988 από τον Yann LeCun. Το πρωτοποριακό του έργο ονομάστηκε LeNet (LeCun et al., 1998) και χρησιμοποιήθηκε κυρίως για εργασίες αναγνώρισης χαρακτήρων όπως διάβασμα ταχυδρομικών κωδικών και ψηφίων. Τα CNNs είναι παρόμοια με τα Νευρωνικά Δίκτυα που περιγράφηκαν στην προηγούμενη ενότητα, καθώς αποτελούνται από νευρώνες και η βασική ιδέα εξακολουθεί να χρησιμοποιεί παραμέτρους μάθησης (βάρος, πολώσεις κλπ.) για την προσέγγιση ορισμένων συναρτήσεων. Κάθε νευρώνας λαμβάνει μερικές εισόδους, εκτελεί ένα εσωτερικό γινόμενο και το ακολουθεί με μια μη-γραμμική συνάρτηση ενεργοποίησης. Επίσης το όλο δίκτυο εξακολουθεί να έχει μια συνάρτηση σφάλματος στο τελευταίο (πλήρως συνδεδεμένο) επίπεδο.

Οπότε τι αλλάζει; Οι αρχιτεκτονικές ConvNets κάνουν τη ρητή παραδοχή ότι οι εισροές είναι εικόνες ή βίντεο, που επιτρέπουν να κωδικοποιούνται ορισμένες ιδιότητες στην αρχιτεκτονική. Αυτές οι ιδιότητες καθιστούν την εμπρόσθια συνάρτηση πιο αποτελεσματική και μειώνεται σημαντικά το πλήθος των παραμέτρων στο δίκτυο. Συγκεκριμένα, σε αντίθεση με ένα κανονικό νευρωνικό δίκτυο, η είσοδος ενός CNN έχουν νευρώνες διατεταγμένους σε 3 διαστάσεις: πλάτος, ύψος, βάθος, όπου η λέξη βάθος εδώ δεν αναφέρεται στον συνολικό αριθμό επιπέδων στο νευρωνικό δίκτυο, αλλά στον αριθμό των καναλιών. Το γεγονός ότι οι νευρώνες οργανώνονται σε τρισδιάστατους όγκους σημαίνει ότι οι νευρώνες σε ένα επίπεδο θα συνδεθούν μόνο σε μια μικρή περιοχή του επιπέδου πριν από αυτήν, αντί για όλους τους νευρώνες με ένα πλήρως συνδεδεμένο τρόπο.

### • Επισκόπηση Αρχιτεκτονικής

#### Fully Connected Neural Networks - Πλήρως Συνδεδεμένα Νευρωνικά Δίκτυα.

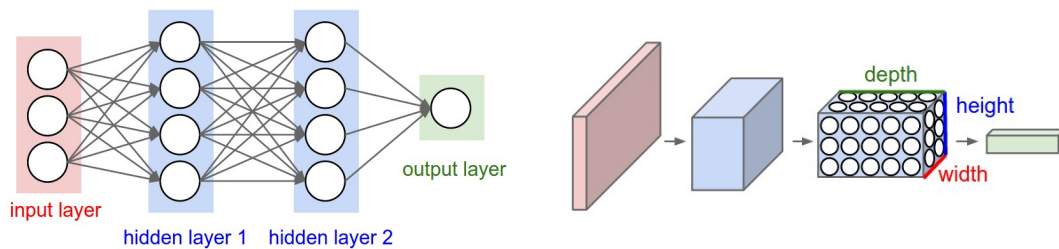
Τα νευρωνικά δίκτυα όπως ήδη έχει αναφερθεί λαμβάνουν μια είσοδο (ένα μόνο διάνυσμα) και τη μετατρέπουν μέσα από μια σειρά κρυφών επιπέδων. Κάθε κρυμμένο επίπεδο αποτελείται από ένα σύνολο νευρώνων, όπου κάθε νευρώνας είναι συνδεδεμένος με όλους τους νευρώνες στο προηγούμενο επίπεδο και όπου οι νευρώνες σε κάθε επίπεδο λειτουργούν εντελώς ανεξάρτητα και δε μοιράζονται καμία σύνδεση.

Τα κανονικά νευρωνικά δίκτυα δεν κλιμακώνονται καλά σε πλήρεις εικόνες. Για παράδειγμα για εικόνες που έχουν μέγεθος μόνο  $32 \times 32 \times 3$  (32 πλάτος, 32 ύψος, 3 κανάλια χρώματος), έτσι ένας μόνο πλήρως συνδεδεμένος νευρώνας στο πρώτο κρυφό επίπεδο ενός κανονικού νευρωνικού δικτύου θα έχει  $32 * 32 * 3 = 3072$  παραμέτρους. Ο αριθμός αυτός εξακολουθεί να φαίνεται εύχρηστος, αλλά σαφώς αυτή η πλήρως συνδεδεμένη δομή δεν κλιμακώνεται σε μεγαλύτερες εικόνες. Για παράδειγμα, μια εικόνα πιο αξιολογιστικού μεγέθους, π.χ.  $200 \times 200 \times 3$  θα οδηγούσε σε νευρώνες που έχουν  $200 * 200 * 3 = 120000$  παραμέτρους. Επιπλέον, σχεδόν σίγουρα θα ήταν αναγκαία η ύπαρξη αρκετών νευρώνων, έτσι ώστε οι παράμετροι να προστίθενται γρήγορα! Είναι σαφές ότι αυτή η πλήρης συνδεσιμότητα είναι χωρίς χρησιμότητα και ο τεράστιος αριθμός παραμέτρων θα οδηγούσε γρήγορα σε υπερπροσαρμογή. Ακόμη



με την αναπαράσταση των συνελικτικών νευρωνικών δικτύων διατηρείται η χωρική πληροφορία των εικόνων.

3D volumes of neurons - Τρισδιάστατοι όγκοι νευρώνων. Τα συνελικτικά νευρωνικά δίκτυα επωφελούνται από το γεγονός ότι η είσοδος αποτελείται από εικόνες και περιορίζουν την αρχιτεκτονική με ένα πιο λογικό τρόπο. Ειδικότερα, σε αντίθεση με ένα κανονικό νευρωνικό δίκτυο, τα επίπεδα ενός ConvNet έχουν νευρώνες διατεταγμένους σε 3 διαστάσεις όπως αναφέρθηκε και προηγουμένως: πλάτος, ύψος και βάθος.



**Σχήμα 2.7:** Αριστερά: Ένα κανονικό Νευρωνικό Δίκτυο 3-επιπέδων. Δεξιά: Το ConvNet οργανώνει τους νευρώνες σε τρεις διαστάσεις (πλάτος, ύψος, βάθος), όπως απεικονίζεται σε ένα από τα επίπεδα. Κάθε επίπεδο ενός ConvNet μετατρέπει τον όγκο εισόδου 3D σε έναν όγκο εξόδου 3D των ενεργοποιήσεων των νευρώνων. Σε αυτό το παράδειγμα, το κόκκινο επίπεδο εισόδου αναπαριστά την εικόνα, έτσι το πλάτος και το ύψος είναι οι διαστάσεις της εικόνας και το βάθος θα είναι 3 (κόκκινο, πράσινο, μπλε κανάλι)

Όπως υπονοεί το όνομά τους, το CNN βασίζεται ουσιαστικά στη μαθηματική λειτουργία της συνέλιξης. Σε γενικές γραμμές, υπάρχουν τέσσερις πράξεις που αποτελούν το βασικό δομικό στοιχείο του κάθε συνελικτικού νευρωνικού δικτύου:

- Convolution - Συνέλιξη
- Non Linearity (ReLU) - Μη γραμμικότητα (Γραμμικός Ανορθωτής)
- Pooling or Sub-Sampling - Συγκέντρωση (Υπο-δειγματοληψία)
- Decoding - Αποκωδικοποίηση

### 2.2.1 Συνελικτικό Επίπεδο

Τα CNN παίρνουν το όνομά τους από τη λειτουργία της συνέλιξης. Ο πρωταρχικός σκοπός του συνελικτικού επιπέδου (convolutional layer) στην περίπτωση ενός συνελικτικού νευρωνικού δικτύου (CNN) είναι η εξαγωγή χαρακτηριστικών από την εικόνα εισόδου. Η συνέλιξη διατηρεί τη χωρική σχέση μεταξύ των εικονοστοιχείων (pixels) με την εκμάθηση χαρακτηριστικών εικόνας χρησιμοποιώντας μικρά τετράγωνα δεδομένων εισόδου.

Οι παράμετροι του συνελικτικού επιπέδου αποτελούνται από ένα σύνολο φίλτρων που μπορούν να εκπαιδευθούν, τα οποία ονομάζονται επίσης πυρήνες ή ανιχνευτές χαρακτηριστικών. Κάθε φίλτρο είναι μια πολύ μικρή περιοχή σε σχέση με το πλάτος και το ύψος του όγκου εισόδου, αλλά εκτείνεται μέσω του πλήρους βάρους. Σχεδόν όλα τα φίλτρα που χρησιμοποιούνται στην πράξη είναι

τετράγωνα πλέγματα, αλλά έχουν χρησιμοποιηθεί και μη τετραγωνικοί πυρήνες. Κατά τη διάρκεια του προς τα εμπρός περάσματος, κάθε φίλτρο μετακινείται κατά πλάτος και ύψος του όγκου εισόδου και το γινόμενο στοιχείων με στοιχείων ανάμεσα στις εισόδους του φίλτρου και στην είσοδο σε κάθε θέση υπολογίζεται. Καθώς ο πυρήνας στρέφεται πάνω από το πλάτος και το ύψος του όγκου εισόδου, ένας 2-διαστάσεων χάρτης ενεργοποίησης ή χάρτης χαρακτηριστικών παράγεται. Ο χάρτης χαρακτηριστικών περιέχει τις απαντήσεις αυτού του πυρήνα σε κάθε χωρική θέση. Διαισθητικά, το δίκτυο θα μάθει φίλτρα που θα ενεργοποιηθούν όταν δουν κάποιο είδος οπτικού χαρακτηριστικού, όπως μια ακμή με κάποιο προσανατολισμό ή μια κηλίδα κάποιου χρώματος στο πρώτο στρώμα ή τελικά πιο αφηρημένα μοτίβα σε υψηλότερα επίπεδα. Στην πράξη, κάθε συνελικτικό επίπεδο (Convolutional layer) έχει ένα ολόκληρο σετ φίλτρων και κάθε ένα από αυτά παράγουν ένα ξεχωριστό 2-διαστάσεων χάρτη ενεργοποίησης. Αυτοί οι Χάρτες Ενεργοποίησης στη συνέχεια στοιβάζονται κατά μήκος της διάστασης του βάθους και παράγουν τον όγκο εξόδου. Είναι σημαντικό να σημειωθεί ότι η λειτουργία της συνέλιξης καταγράφει τις τοπικές εξαρτήσεις στην αρχική εικόνα. Όσο περισσότερα φίλτρα διαθέτουμε, τόσες περισσότερες εικόνες εξάγονται και τόσο καλύτερα το δίκτυο μας εκπαιδεύεται στην αναγνώριση μοτίβων στις εικόνες. Αυτή η διαδικασία μπορεί να περιγραφεί χρησιμοποιώντας την ακόλουθη εξίσωση, όπου  $a_l(i, j)$  είναι ένα εικονοστοιχείο του χάρτη ενεργοποίησης,  $W_l$  είναι ο πυρήνας (kernel) με μέγεθος  $H \times W \times C$  και  $b_l$  ένας  $C$ -διαστάσεων διάνυσμα πόλωσης (Goodfellow et al., 2016).

$$a_l(i, j) = \sum_{h=1}^H \sum_{w=1}^W \sum_{c=1}^C W_l(h, w, c) I(i - h, j - w, c) + b_l(c) \quad (2.9)$$

**Τοπική Συνδεσιμότητα - Local Connectivity.** Όταν η είσοδος είναι πολύ υψηλής διάστασης (π.χ. εικόνες), είναι πρακτικά αδύνατον να συνδεθούν οι νευρώνες ενός επιπέδου με όλους τους νευρώνες στο προηγούμενο επίπεδο. Αντ'αυτού, θα συνδεθεί κάθε νευρώνας μόνο με μια τοπική περιοχή της εισόδου. Η χωρική έκταση αυτής της συνδεσιμότητας είναι μια υπερπαράμετρος που ονομάζεται **πεδίο υποδοχής (receptive field) ή F** του νευρώνα (ισοδύναμα αυτό είναι το μέγεθος του φίλτρου). Η έκταση της συνδεσιμότητας κατά μήκος του άξονα βάθους είναι πάντα ίση με το βάθος του όγκου εισόδου.

Παράδειγμα: Έστω ότι ο όγκος εισόδου έχει μέγεθος  $[16 \times 16 \times 20]$ . Στη συνέχεια, χρησιμοποιώντας ένα παράδειγμα πεδίου υποδοχής μεγέθους  $3 \times 3$ , κάθε νευρώνας στο επίπεδο Conv θα έχει τώρα συνολικά  $3 * 3 * 20 = 180$  συνδέσεις στον όγκο εισόδου, ενώ στην περίπτωση του πλήρους συνδεδεμένου επιπέδου ο αντίστοιχος νευρώνας θα είχε  $16 * 16 * 20 = 5120$  συνδέσεις. Σημαντικό είναι να παρατηρήσει κάποιος ότι πάλι η συνδεσιμότητα είναι τοπική στο διάστημα (π.χ.  $3 \times 3$ ), αλλά πλήρης κατά μήκος του βάθους εισόδου (20).

Το μέγεθος του όγκου εξόδου (output volume) ελέγχεται από 3 υπερπαραμέτρους, το βάθος (depth), το βήμα (stride) και το γέμισμα των ακριανών τιμών (padding), όπως πχ zero-padding, same-padding, no-padding.

- Το **βάθος - depth** του όγκου εξόδου είναι μία υπερπαράμετρος. Είναι ο αριθμός των φίλτρων που είναι επιθυμητό να χρησιμοποιείται, με κάθε μάθηση για να αναζητηθεί κάτι διαφορετικό στην είσοδο. Για παράδειγμα, αν το πρώτο συνελικτικό επίπεδο (Convolutional Layer) παίρνει ως είσοδο την ακατέργαστη εικόνα, τότε διαφορετικοί νευρώνες κατά μήκος της διάστασης

βάθους μπορούν να ενεργοποιηθούν παρουσία διαφορετικά προσανατολισμένων ακμών, ή σφαιρών χρώματος.

- Δεύτερον, πρέπει να καθοριστεί το **βήμα - stride** με το οποίο θα κυλήσει το φίλτρο. Όταν το βήμα είναι 1 τότε μετακινούνται τα εικονοστοιχεία ένα εικονοστοιχείο τη φορά. Όταν το βήμα είναι 2 (ή λιγότερο συχνά 3 ή παραπάνω, αν και αυτό είναι σπάνιο στην πράξη) τότε τα φίλτρα πηδούν 2 εικονοστοιχεία τη φορά καθώς κυλάνε. Αυτό θα παράγει χωρικά μικρότερους όγκους εξόδου.
- Προκειμένου οι χάρτες στην έξοδο του επιπέδου να έχουν τις ίδιες διαστάσεις με την είσοδο, μερικές φορές είναι βολικό να τοποθετηθούν στον όγκο εισόδου μηδενικά γύρω από το περίγραμμα. Το μέγεθος αυτού του **γεμίσματος με μηδενικά - zero padding** είναι μια υπερπαράμετρος. Το ωραίο χαρακτηριστικό του zero padding είναι ότι επιτρέπει να ελεγχθεί το χωρικό μέγεθος των όγκων εξόδου.

Εάν ένα στρώμα Convolutional έχει έναν όγκο εισόδου μεγέθους  $H_1 \times W_1 \times C_1$ ,  $K$  φίλτρα με μέγεθος  $N \times M \times C_1$ , βήμα  $S$  και  $P$  μέγεθος του zero padding, τότε το μέγεθος του όγκου εξόδου μετά από τη συνέλιξη του όγκου εισόδου με το φίλτρο περιγραφής θα είναι  $H_2 \times W_2 \times C_2$ , όπου

$$H_2 = (H_1 - N + 2P)/S + 1, W_2 = (W_1 - M + 2P)/S + 1, C_2 = K \quad (2.10)$$

Χρησιμοποιώντας την παραπάνω εξίσωση υπολογίζεται ο όγκος εξόδου του πρώτου συνελκτικού επιπέδου του Krizhevsky (Krizhevsky et al., 2012) για τον υπολογισμό του αριθμού των παραμέτρων που χρησιμοποιήθηκαν και για να γίνει γνωστό πώς η τοπική δυνατότητα σύνδεσης μπορεί να μειώσει τις χρησιμοποιούμενες παραμέτρους. Η αρχιτεκτονική που κέρδισε το στο ImageNet το 2012 (παράδειγμα πραγματικού κόσμου) δέχτηκε εικόνες μεγέθους  $227 \times 227 \times 3$ . Χρησιμοποίησε φίλτρα  $K=96$ , με μέγεθος  $F=N=M=11$ , βήμα  $S=4$  και δεν υπάρχει μηδενική επένδυση  $P=0$  στο πρώτο επίπεδο συνέλιξης. Ο όγκος εξόδου είχε μέγεθος  $55 \times 55 \times 96$  επειδή  $(227 - 11)/4 + 1 = 55$ . Κάθε ένας από τους  $55 * 55 * 96 = 290400$  νευρώνες σε αυτό τον όγκο συνδέονταν σε μια περιοχή μεγέθους  $11 \times 11 \times 3$  στην εικόνα εισόδου, έτσι οι νευρώνες είχαν  $11 * 11 * 3 = 363$  βάρη και 1 πόλωση (bias), προσθέτοντας πάνω από  $290400 * 364 = 105705600$  παραμέτρους του πρώτου επιπέδου του CNN. Παρ'όλα αυτά, εάν το δίκτυο ήταν πλήρως συνδεδεμένο ο συνολικός αριθμός παραμέτρων θα ήταν  $290400 * 290400 = 84332160000$  το οποίο είναι απαγορευτικά υψηλό. Ακόμα και ο αριθμός  $105705600$  εξακολουθεί να είναι πολύ υψηλός και στην πραγματικότητα υπάρχει ένας μικρότερος αριθμός παραμέτρων που χρησιμοποιούνται, με βάση μια θεμελιώδη ιδέα των CNNs, την κοινή χρήση παραμέτρων (Parameter Sharing).

Επομένως, η τοπική συνδεσιμότητα μειώνει σημαντικά την ποσότητα των παραμέτρων και του χρόνου υπολογισμού των συνέλιξεων σε ένα CNN. Επιπλέον οι χάρτες ενεργοποίησης των πρώτων συνελκτικών επιπέδων συλλαμβάνουν τα τοπικά χαρακτηριστικά. Ωστόσο, καθώς κάποιος μετακινείται βαθύτερα στην αρχιτεκτονική, η είσοδος των συνελκτικών επιπέδων εξαρτάται από ένα μεγαλύτερο μέρος του αρχικού όγκου εισόδου. Ως συνάρτηση των τοπικών χαμηλότερου-επιπέδου χαρακτηριστικών, τα χαρακτηριστικά που συλλαμβάνονται σε βαθύτερα επίπεδα, τείνουν να συνδυάζουν αυτά τα χαμηλού-επιπέδου χαρακτηριστικά προκειμένου να παράγουν υψηλότερου επιπέδου αναπαραστάσεις.

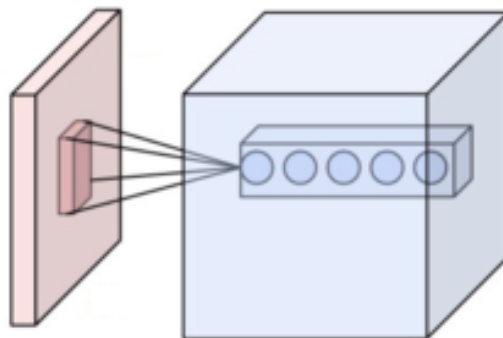
**Κοινή Χρήση Παραμέτρων - Parameter Sharing.** Ο αριθμός των

παραμέτρων ενός συνελικτικού επιπέδου μπορεί να μειωθεί δραματικά κάνοντας μια λογική υπόθεση:

“Εάν ένα χαρακτηριστικό είναι χρήσιμο να υπολογιστεί σε κάποια χωρική θέση  $(x,y)$ , τότε επίσης θα να είναι χρήσιμο να υπολογιστεί σε διαφορετική θέση  $(x',y')$ ” (Epperson, 2018).

Συγκεκριμένα, όλοι οι νευρώνες σε αντίστοιχες θέσεις κατά μήκος του βάθους θα πρέπει να χρησιμοποιούν τα ίδια βάρη και πόλωση. Με αυτό το σχήμα ανταλλαγής παραμέτρων, το CNN θα έχει τη δυνατότητα να εντοπίσει ακριβώς τα ίδια χαρακτηριστικά σε διαφορετικές περιοχές της εισόδου, μια ιδιότητα γνωστή ως *ισοδυναμία με τη μετάφραση*. Ένα πλήρως συνδεδεμένο μοντέλο δεν διαθέτει αυτή την ιδιότητα, καθώς χρησιμοποιεί διαφορετικές παραμέτρους σε διαφορετικές περιοχές του όγκου εισόδου. Με την χρήση του προαναφερθέντος περιορισμού, το πρώτο συνελικτικό επίπεδο στο παραπάνω παράδειγμα τώρα θα έχει μόνο 96 μοναδικά σύνολα βαρών (ένα για κάθε φέτα βάθους). Ο συνολικός αριθμός παραμέτρων θα είναι τώρα  $96 * 11 * 11 * 3 + 96 = 34944$ . Δηλαδή και οι  $55 * 55$  νευρώνες σε κάθε φέτα βάθους τώρα θα χρησιμοποιούν τις ίδιες παραμέτρους. Δεδομένου ότι όλοι οι νευρώνες σε ένα ενιαίο κομμάτι βάθους χρησιμοποιούν το ίδιο διάνυμα βάρους, το πέρασμα προς τα εμπρός του συνελικτικού επιπέδου μπορεί σε κάθε κομμάτι βάθους να υπολογιστεί ως μια συνέλιξη των βαρών του νευρώνα με τον όγκο εισόδου. Αυτός είναι ο λόγος για τον οποίο είναι σύνηθες να αναφέρονται τα σύνολα των βαρών ως φίλτρο ή πυρήνας (kernel), που συμπλέκεται με την είσοδο. Στην διάρκεια της στρατηγικής ανάστροφης διάδοσης (backpropagation), κάθε νευρώνας στον όγκο θα υπολογίσει την κλίση (gradient) για τα βάρη του, αλλά αυτές οι κλίσεις θα προστεθούν σε κάθε φέτα βάθους και θα ενημερώσουν μόνο ένα ενιαίο σύνολο βαρών ανά φέτα.

Συμπερασματικά, τα σχέδια τοπικής συνδεσιμότητας και η κοινή χρήση παραμέτρων αυξάνουν δραματικά την αποτελεσματικότητα, ενώ παράλληλα μειώνεται ο αριθμός των παραμέτρων που χρησιμοποιούνται από το μοντέλο σε ένα διαχειρίσιμο αριθμό (από 84332160000 σε 34944 στο παράδειγμα που περιγράφηκε παραπάνω).



**Σχήμα 2.8:** Οι νευρώνες ενός συνελικτικού επιπέδου (μπλε), συνδεδεμένοι στο δεκτικό τους πεδίο (κόκκινο).

Στην παρούσα διπλωματική γίνεται χρήση των παρακάτω συνελικτικών επιπέδων και επιπέδων μετασχηματισμών:

- 2D επίπεδα συνέλιξης (2D convolution layer). Αυτό το επίπεδο δημιουργεί έναν πυρήνα συνέλιξης ο οποίος συμπλέκεται με την είσοδο του επιπέδου για να παράγει έναν τανυστή εξόδων.

- Μετασχηματισμένο επίπεδο συνέλιξης (Transposed convolution layer - Conv2DTranspose). Η ανάγκη για μετασχηματισμένες συνελίξεις γενικά προκύπτει από την επιθυμία να χρησιμοποιηθεί ένας μετασχηματισμός που πηγαίνει προς την αντίθετη κατεύθυνση μιας κανονικής συνέλιξης, δηλαδή από κάτι που έχει τη μορφή της εξόδου κάποιας συνέλιξης σε κάτι που έχει τη μορφή της εισόδου του ενώ διατηρεί μοτίβο συνδεσιμότητας που είναι συμβατό με την εν λόγω συνέλιξη.
- Επίπεδο περικοπής για είσοδο 2D (Cropping2D) (π.χ. εικόνα). Περικόπτει κατά μήκος των χωρικών διαστάσεων, δηλαδή το ύψος και το πλάτος.
- Επίπεδο ανατροπής για εισόδους 2D (UpSampling2D). Επαναλαμβάνει τις σειρές και τις στήλες των δεδομένων κατά μέγεθος [0] και μέγεθος [1] αντίστοιχα.
- Επίπεδο μηδενικής παρεμβολής για είσοδο 2D (ZeroPadding2D) (π.χ. εικόνα). Αυτό το επίπεδο μπορεί να προσθέσει σειρές και στήλες με μηδενικά στην επάνω, κάτω, αριστερή και δεξιά πλευρά ενός ταυστή εικόνας.
- **Non-linear Layer - Μη γραμμικό Επίπεδο**

Το συνελικτικό επίπεδο μετατρέπει τον όγκο εισόδου σε έναν όγκο εξόδου με γραμμικό τρόπο, αφού ο χειριστής συνέλιξης είναι γραμμικός. Επομένως, θα πρέπει να χρησιμοποιούνται μη-γραμμικότητες σε κάθε τιμή του χάρτη χαρακτηριστικών, με αποτέλεσμα ένα διορθωμένο χάρτη χαρακτηριστικών.

### 2.2.2 Συναρτήσεις Ενεργοποίησης

Οι **Συναρτήσεις Ενεργοποίησης (Activation Functions)** είναι πραγματικά σημαντικές για ένα Τεχνητό Νευρωνικό Δίκτυο (Artificial Neural Network), ώστε να μάθει και να κατανοήσει κάτι πολύ περίπλοκο και με μη-γραμμική σύνθετη λειτουργική αντιστοίχιση (mapping) μεταξύ των εισόδων και της μεταβλητής απόκρισης. Εισάγουν μη γραμμικές ιδιότητες στο Δίκτυο. Ο κύριος σκοπός τους είναι να μετατρέψουν ένα σήμα εισόδου ενός κόμβου από ένα Τεχνητό Νευρωνικό Δίκτυο, σε ένα σήμα εξόδου. Αυτό το σήμα εξόδου χρησιμοποιείται τώρα ως είσοδος στο επόμενο επίπεδο της στοίβας.

Συγκεκριμένα σε ένα Τεχνητό Νευρωνικό Δίκτυο γίνεται το άθροισμα του γινομένου των εισόδων ( $X$ ) με τα αντίστοιχα Βάρη ( $W$ ) και της πόλωσης (bias) και εφαρμόζεται μια συνάρτηση Ενεργοποίησης  $f(x)$  ώστε να προκύψει η έξοδος αυτού του επιπέδου και να τροφοδοτηθεί στο επόμενο επίπεδο ως είσοδος.

Αν δεν εφαρμοστεί μια συνάρτηση Ενεργοποίησης, τότε το σήμα εξόδου θα ήταν απλά μια απλή γραμμική συνάρτηση, η οποία είναι απλώς ένα πολυώνυμο ενός βαθμού. Μια γραμμική εξίσωση είναι εύκολο να λυθεί, αλλά είναι περιορισμένη στην πολυπλοκότητά της και έχει λιγότερη ισχύ για να μάθει σύνθετες λειτουργικές αντιστοιχίσεις από τα δεδομένα. Ένα επίπεδο νευρωνικού δικτύου χωρίς συνάρτηση ενεργοποίησης θα ήταν απλώς ένα μοντέλο γραμμικής παλινδρόμησης (Linear regression Model), το οποίο έχει περιορισμένη ισχύ και δεν αποδίδει καλά τις περισσότερες φορές. Επιπλέον αυτό που χρειάζεται είναι το Νευρωνικό Δίκτυο όχι μόνο να μαθαίνει και να υπολογίζει μια γραμμική λειτουργία, αλλά κάτι πιο περίπλοκο από αυτό.

Επίσης, χωρίς τη λειτουργία ενεργοποίησης, το Νευρωνικό μας δίκτυο δε θα είναι σε θέση να μάθει και να μοντελοποιήσει άλλα περίπλοκα είδη δεδομένων, όπως εικόνες, βίντεο, ήχο, ομιλία κλπ. Αυτός είναι και ο λόγος για τον οποίο χρησιμοποιούνται τεχνικές Τεχνητού Νευρωνικού Δικτύου, όπως η Βαθιά Μάθηση, για να κατανοηθούν πιο πολύπλοκα, μεγάλης διαστάσεως, μη γραμμικά μεγάλα σύνολα δεδομένων, όπου το μοντέλο έχει πολλά κρυφά επίπεδα μεταξύ τους και έχει μια πολύ περίπλοκη αρχιτεκτονική που βοηθά στο να κατανοηθούν και να εξαχθούν γνώσεις από τέτοια πολύπλοκα μεγάλα σύνολα δεδομένων.

**Οπότε γιατί είναι αναγκαίες οι μη-γραμμικότητες;** Οι μη γραμμικές λειτουργίες είναι αυτές που έχουν βαθμό περισσότερο από ένα και έχουν καμπυλότητα όταν σχεδιάζεται μια μη γραμμική συνάρτηση. Τώρα χρειάζεται ένα μοντέλο νευρωνικού δικτύου για να γίνει γνωστό και να αντιπροσωπεύει σχεδόν οτιδήποτε και οποιαδήποτε αυθαίρετη σύνθετη λειτουργία που αντιστοιχεί τις εισόδους με τις εξόδους. Τα νευρωνικά δίκτυα όπως ήδη έχει αναφερθεί παραπάνω, θεωρούνται **Προσεγγιστές Καθολικής Συνάρτησης (Universal Function Approximators)**. Αυτό σημαίνει, όπως ήδη έχει γίνει γνωστό και παραπάνω, ότι μπορούν να υπολογίσουν και να μάθουν οποιαδήποτε λειτουργία. Σχεδόν κάθε διαδικασία που μπορεί κάποιος να σκεφτεί μπορεί να εκπροσωπείται ως λειτουργικός υπολογισμός στα Νευρωνικά Δίκτυα. Για το λόγο αυτό, πρέπει να εφαρμοστεί μια συνάρτηση ενεργοποίησης  $f(x)$  ώστε να καταστεί το δίκτυο πιο ισχυρό και να προστεθεί η ικανότητα να μάθει κάτι περίπλοκο από τα δεδομένα και να αντιπροσωπεύει μη-γραμμικές σύνθετες αυθαίρετες λειτουργικές αντιστοιχίσεις μεταξύ εισόδων και εξόδων. Ως εκ τούτου, χρησιμοποιώντας μη-γραμμική ενεργοποίηση, είναι δυνατό να παράγονται μη-γραμμικές απεικονίσεις από τις εισόδους στις εξόδους.

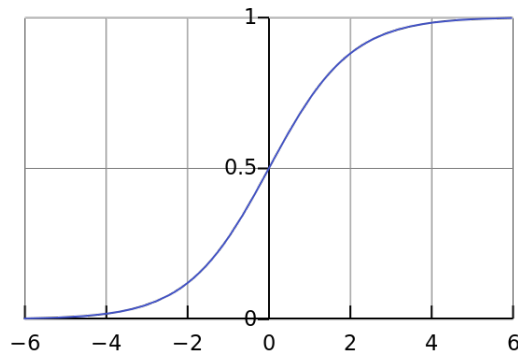
Ακόμη ένα άλλο σημαντικό χαρακτηριστικό μιας συνάρτησης ενεργοποίησης είναι ότι πρέπει να είναι διαφοροποιήσιμο. Χρειάζεται να είναι έτσι ώστε να εκτελείται στρατηγική ανάστροφης διάδοσης (backpropagation) ενώ παράλληλα προωθείται προς τα πίσω στο δίκτυο ώστε να υπολογίζονται οι διαβαθμίσεις του Σφάλματος σε σχέση με τα Βάρη και στη συνέχεια να γίνεται βελτιστοποίηση των βαρών με την χρήση της Gradient descend ή οποιαδήποτε άλλης τεχνική βελτιστοποίησης για να μετρηθεί το σφάλμα (Walia, 2017a).

Οι πιο διαδεδομένες συναρτήσεις ενεργοποίησης είναι οι παρακάτω.

- **Σιγμοειδής Συνάρτηση**

Η Σιγμοειδής συνάρτηση ενεργοποίησης έχει την ακόλουθη μαθηματική εξίσωση και φαίνεται παρακάτω.

$$A = \frac{1}{1 + e^{-x}}$$



Σχήμα 2.9: Σιγμοειδής Συνάρτηση (Sharma, 2017).

Παίρνει έναν πραγματικό αριθμό και τον περιορίζει στην περιοχή  $[0,1]$ . Πιο συγκεκριμένα, οι μεγάλοι αρνητικοί αριθμοί γίνονται 0 και οι μεγάλοι θετικοί γίνονται 1. Η λειτουργία σιγμοειδούς έχει μια πολύ ωραία ερμηνεία ως ποσοστό πυροδότησης ενός νευρώνα, δεδομένου ότι η τιμή του κυμαίνεται μεταξύ 0 (δεν πυροτεύεται καθόλου) και 1 (πλήρης κορεσμένη πυροδότηση σε υποτιθέμενη μέγιστη συχνότητα) και έτσι έχει χρησιμοποιηθεί εκτενώς.

- **Ανορθωμένη Γραμμική Μονάδα**

Η συνάρτηση ενεργοποίησης Ανορθωμένη Γραμμική Μονάδα (Rectified Linear Unit-ReLU) η οποία μπορεί να οριστεί χρησιμοποιώντας τη παρακάτω μαθηματική μορφή, έχει γίνει πολύ δημοφιλής τα τελευταία χρόνια.

$$y = \varphi(x) = \max(0, x) \quad (2.11)$$

Η μη-γραμμικότητα του ReLU έχει βρεθεί ότι επιταχύνει τη σύγκλιση της Stochastic Gradient Descent σε σύγκριση με τις συναρτήσεις σιγμοειδής/εφαπτομένη (sigmoid/tanh). Επίσης και η γραμμική της μορφή είναι υπεύθυνη για την ταχύτερη σύγκλιση. Παρ'όλα αυτά, οι μονάδες ReLU έχουν επίσης ένα ενοχλητικό μειονέκτημα, το οποίο καλείται ανεπίσημα το πρόβλημα του “πεθαμένου ReLU”: μπορεί να είναι εύθραστη κατά τη διάρκεια της εκπαίδευσης και μπορεί να “πεθάνει”. Για παράδειγμα μια μεγάλη κλίση (gradient) που ρέει μέσω ενός νευρώνα ReLU θα μπορούσε να αναγκάσει τα βάρη να ενημερωθούν με τέτοιο τρόπο ώστε ο νευρώνας να μην ενεργοποιηθεί ποτέ ξανά σε κανένα σημείο δεδομένων. Εάν συμβεί αυτό, μετά η κλίση (gradient) που ρέει μέσα από τη μονάδα θα είναι για πάντα μηδέν από εκείνο το σημείο και μετά. Δηλαδή, οι ReLU μονάδες μπορούν να πεθάνουν μη αναστρέψιμα κατά τη διάρκεια της εκπαίδευσης.

Πιο συγκεκριμένα, ένα “νεκρό” ReLU παράγει πάντα την ίδια τιμή (μηδέν) για οποιαδήποτε είσοδο. Πιθανώς αυτό έρχεται με την εκμάθηση ενός μεγάλου αρνητικού όρου μεροληψίας για τα βάρη του. Με τη σειρά του, αυτό σημαίνει ότι δεν παίζει κανένα ρόλο στη διάκριση μεταξύ εισόδων. Για ταξινόμηση, θα μπορούσατε να το απεικονίσετε ως επίπεδο απόφασης εκτός από όλα τα πιθανά δεδομένα εισόδου.

Μόλις ένα ReLU καταλήξει σε αυτή την κατάσταση, είναι απίθανο να ανακάμψει, επειδή η κλίση της συνάρτησης στο 0 είναι επίσης 0, και δεν μπορεί να μεταβάλλει τα βάρη.

Όπως αναφέρθηκε και παραπάνω ο ορισμός της (ReLU), δίνεται από την εξίσωση 2.11, με  $x = Wa + b$ . Ένα σημαντικό πλεονέκτημα της είναι η μειωμένη πιθανότητα

να εξαφανιστεί η κλίση. Αυτό προκύπτει όταν  $x > 0$ . Σε αυτή την κατάσταση η κλίση έχει σταθερή τιμή. Αντίθετα, η κλίση των σιγμοειδών καθίσταται όλο και μικρότερη καθώς αυξάνεται η απόλυτη τιμή του  $x$ . Η σταθερή κλίση των (ReLU) έχει ως αποτέλεσμα ταχύτερη εκμάθηση. Το άλλο πλεονέκτημα είναι η αραιότητα (sparsity), η οποία προκύπτει όταν  $x \leq 0$ . Όσο περισσότερες μονάδες υπάρχουν σε ένα επίπεδο τόσο πιο αραιή είναι η προκύπτουσα αναπαράσταση. Οι σιγμοειδείς από την άλλη πλευρά είναι πάντα πιθανό να παράγουν κάποια μη μηδενική τιμή με αποτέλεσμα πιο πυκνές αναπαραστάσεις. Οι αραιές αναπαραστάσεις φαίνεται να είναι πιο ευεργετικές από τις πυκνές αναπαραστάσεις.

- **Softmax Function**

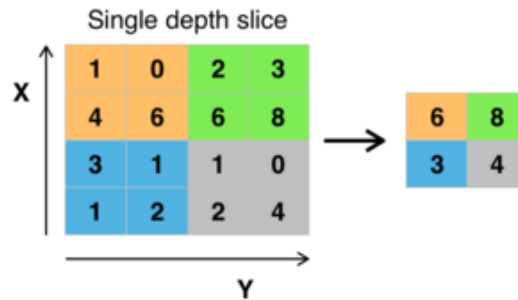
Στα μαθηματικά, η συνάρτηση Softmax ή κανονικοποιημένη εκθετική συνάρτηση, είναι μια γενίκευση της λογικής λειτουργίας και παίρνει την έξοδο του νευρωνικού και επιστρέφει πιθανότητες. Η λογική λειτουργία ή λογική καμπύλη είναι ένα κοινό σχήμα “S” (Σιγμοειδής καμπύλη). Η συνάρτηση Softmax χρησιμοποιείται σε διάφορες μεθόδους multi-class ταξινόμησης. Για παράδειγμα έστω η είσοδος [1,2,3,4,1,2,3], η softmax είναι [0.024, 0.064, 0.175, 0.475, 0.024, 0.064, 0.175]. Η έξοδος έχει το μεγαλύτερο μέρος του βάρους, όπου το “4” ήταν στην αρχική είσοδο. Γι’αυτό συνήθως χρησιμοποιείται αυτή η συνάρτηση: για να επισημανθούν οι μεγαλύτερες τιμές και να κατασταλούν τιμές που είναι σημαντικά χαμηλότερες από τη μέγιστη τιμή. Είναι πολύ σημαντικό να σημειωθεί ότι η softmax δεν είναι αμετάβλητη κλίμακα, οπότε αν η είσοδος ήταν [0.1, 0.2, 0.3, 0.4, 0.1, 0.2, 0.3] (που αθροίζεται στο 1.6), η softmax θα είναι [0.125, 0.138, 0.153, 0.169, 0.138, 0.153]. Αυτό δείχνει ότι για τιμές μεταξύ 0 και 1, η softmax στην πραγματικότητα, “υποτιμά” τη μέγιστη τιμή (σημειώστε ότι το 0.169 δεν είναι μόνο μικρότερη από 0.475, είναι επίσης μικρότερη από την αρχική τιμή 0.4). Όπως αναφέρθηκε και προηγουμένως, η συνάρτηση softmax συχνά χρησιμοποιείται στο τελικό επίπεδο ενός ταξινομητή που βασίζεται σε νευρωνικό δίκτυο (Nasrabadi, 2007).

### 2.2.3 Επίπεδο Συγκέντρωσης

Είναι σύνηθες για τις αρχιτεκτονικές του CNN να περιέχουν περιοδικά ένα Επίπεδο Συγκέντρωσης (Pooling Layer) μεταξύ των διαδοχικών επιπέδων Συνέλιξης. Η χωρική συγκέντρωση ή υποδειγματοληψία μειώνει τη διάσταση του κάθε χάρτη χαρακτηριστικών αλλά διατηρεί τις πιο σημαντικές πληροφορίες. Με τον τρόπο αυτό το επίπεδο συγκέντρωσης (pooling layer) χρησιμεύει για την σταδιακή μείωση του χωροταξικού μεγέθους της αναπαράστασης, τη μείωση του αριθμού των παραμέτρων της και του αριθμού των υπολογισμών στο δίκτυο και συνεπώς τον έλεγχο της υπερφόρτωσης (Scherer et al., 2010). Το επίπεδο συγκέντρωσης λειτουργεί ανεξάρτητα σε κάθε κομμάτι βάθους της εισόδου και το αλλάζει ξανά χωρικά, χρησιμοποιώντας τη λειτουργία του Μέγιστου, του Μέσου ή του Αθροίσματος. Η συγκέντρωση (Pooling) λοιπόν είναι μια μορφή μη γραμμικής δειγματοληψίας. Υπάρχουν πολλές μη γραμμικές λειτουργίες για την υλοποίηση της συγκέντρωσης μεταξύ των οποίων η μέγιστη συγκέντρωση (max pooling) η οποία είναι η συνηθέστερη. Διαχωρίζει την εικόνα εισόδου σε ένα σύνολο μη επικαλυπτόμενων τμημάτων και για κάθε τέτοια υποπεριοχή εξάγει το μέγιστο. Η διαίσθηση είναι ότι η ακριβής τοποθεσία ενός χαρακτηριστικού είναι λιγότερο σημαντική από την ακατέργαστη θέση του σε σχέση με άλλα χαρακτηριστικά. Στην πράξη, προτιμάται η Μέγιστη συγκέντρωση, καθώς αποφεύγεται η ακύρωση αρνητικών στοιχείων και



εμποδίζει τη θόλωση των ενεργοποιήσεων και των κλίσεων σε όλο το δίκτυο μιας και η κλίση τοποθετείται σε μια θέση κατά τη διάρκεια του backpropagation. Το επίπεδο χωρικής συγκέντρωσης ορίζεται από τη συνάρτηση συσσωμάτωσής του, τις διαστάσεις ύψους και πλάτους της περιοχής όπου αυτή εφαρμόζεται και τις ιδιότητες της συνέλιξης (π.χ. γέμισμα-padding, βήμα-stride). Στην παρούσα διπλωματική γίνεται χρήση του MaxPooling2D, για μέγιστη συγκέντρωση για χωρικά δεδομένα.



Σχήμα 2.10: Μέγιστη Συγκέντρωση με φίλτρο  $2 \times 2$  και βήμα = 2.

## 2.2.4 Λοιπά Επίπεδα

Στην παρούσα διπλωματική γίνεται χρήση των παρακάτω επιπέδων πυρήνα (KERAS):

- Πυκνά (Dense) Πλήρως Συνδεδεμένα (FC)

Το κανονικό πυκνά συνδεδεμένο επίπεδο νευρωνικού δικτύου.

- Επίπεδο Ενεργοποίησης (Activation layer)

Εφαρμόζει μια μη-γραμμική συνάρτηση ενεργοποίησης σε μια έξοδο.

- Επίπεδο Εξισορρόπησης (Flatten layer)

Εξισορροπεί την είσοδο. Δεν επηρεάζει το μέγεθος της παρτίδας (batch size).

- Επίπεδο Εισόδου (Input layer)

Η είσοδος χρησιμοποιείται για την παράσταση ενός διανύσματος. Για παράδειγμα μια εικόνα  $32 \times 32 \times 3$  θα κρατήσει τις τιμές των ακατέργαστων εικονοστοιχείων της εικόνας, στην περίπτωση αυτή είναι μια εικόνα με πλάτος 32, ύψος 32 και με τρία κανάλια R, G, B. Το επίπεδο εισόδου (που περιέχει την εικόνα) θα πρέπει να διαιρείται με το 2. Οι συνήθεις αριθμοί είναι 32, 64, 96, 224, 256, 384 και 512.

- Επίπεδο Μετατροπής (Permute layer)

Μετατρέπει τις διαστάσεις τις εισόδου σύμφωνα με ένα δεδομένο μοτίβο.

- Επίπεδο Συγχώνευσης

Άλλο ένα επίπεδο που έχει χρησιμοποιηθεί στη διπλωματική εργασία είναι το Επίπεδο Συγχώνευσης (Concatenate layer). Πρόκειται για ένα επίπεδο που συγχωνεύει μια λίστα εισόδων. Παίρνει ως είσοδο ένα σύνολο ταχυστών, όλων ίδιου σχήματος και επιστρέφει έναν μοναδικό ταχυστή που προκύπτει από τη συνένωση όλων των εισόδων.

- **Επίπεδο Κανονικοποίησης**

Επίσης έχει γίνει χρήση επιπέδου κανονικοποίησης (normalization layer) και συγκεκριμένα επιπέδου Κανονικοποίησης Παρτίδων (BatchNormalization).

Η Κανονικοποίηση Παρτίδων (BatchNormalization) είναι μια προσαρμοστική μέθοδος επαναμέτρησης των κύριων στατιστικών τιμών της παρτίδας προκειμένου να την “κανονικοποιήσει” σε μηδενική μέση τιμή ( $\mu = 0$ ) και μοναδιαία τυπική απόκλιση ( $\sigma = 1$ ). Με τον τρόπο αυτό, διευκολύνει την εκπαίδευση πολύ βαθιών μοντέλων. Η εκπαίδευση αυτών των βαθιών δικτύων είναι περίπλοκη, επειδή οι εισροές σε κάθε επίπεδο επηρεάζονται από τις παραμέτρους όλων των προηγούμενων επιπέδων και οι παράμετροι όλων των επιπέδων ενημερώνονται ταυτόχρονα. Όταν εμείς ενημερώνουμε τις παραμέτρους, μπορεί να συμβούν απροσδόκητα αποτελέσματα επειδή πολλές λειτουργίες ή επίπεδα ενημερώνονται ταυτόχρονα. Σε αυτές τις περιπτώσεις, θα είναι πολύ δύσκολο να επιλέξετε ένα κατάλληλο ρυθμό εκμάθησης, επειδή τα αποτελέσματα της ενημέρωσης των παραμέτρων ενός επιπέδου εξαρτώνται έντονα σχετικά με τις παραμέτρους όλων των άλλων επιπέδων. Η εξομάλυνση των παρτίδων (BatchNormalization) παρέχει έναν απλό και αποτελεσματικό τρόπο επαναμέτρησης σχεδόν οποιουδήποτε τέτοιου δικτύου. Η επαναμέτρηση ενός βαθιού νευρωνικού δικτύου μπορεί να απλοποιήσει σημαντικά το πρόβλημα συντονισμού των ενημερώσεων σε πολλά επίπεδα. Η κανονικοποίηση παρτίδων μπορεί να χρησιμοποιηθεί σε οποιοδήποτε επίπεδο εισόδου ή κρυφό επίπεδο σε ένα νευρωνικό δίκτυο (Wang, 2018).

- **Dropout-Απόσυρση**

Η απόσυρση είναι μια μέθοδος που εισάγεται στο (Srivastava et al., 2014), σύμφωνα με την οποία οι μονάδες τυχαία αποσύρονται (μαζί με τις συνδέσεις τους) από το νευρωνικό δίκτυο κατά τη διάρκεια της εκπαίδευσης. Αυτό αποτρέπει τις μονάδες από τη πάρα πολύ συν-προσαρμογή. Κατά τη διάρκεια της εκπαίδευσης, η απόσυρση δοκιμάζεται από έναν εκθετικό αριθμό διαφορετικών “αραιωμένων” δικτύων. Ως εκ τούτου, η απόσυρση μπορεί να θεωρηθεί ως μια μορφή του συνόλου εκπαίδευσης. Σε κάθε επανάληψη, τα ξεχωριστά υποσύνολα παραμέτρων εκπαιδεύονται σε απομόνωση με άλλες παραμέτρους των οποίων η τιμή είναι 0. Αυτό μπορεί να βοηθήσει σημαντικά τη γενίκευση. Την ώρα της δοκιμής (test), είναι εύκολο να προσεγγιστεί η επίδραση του μέσου όρου των προβλέψεων όλων αυτών των αραιωμένων δικτύων με την χρήση απλά ενός απλού δικτύου που έχει λιγότερα βάρη. Αυτό μειώνει σημαντικά την υπερπροσαρμογή (overfitting) και επομένως η απόσυρση μπορεί να θεωρηθεί ως μορφή συστηματοποίησης.

## 2.3 Απόδοση Νευρωνικών Δικτύων

### 2.3.1 Σφάλματα

Στα περισσότερα δίκτυα μάθησης, το σφάλμα υπολογίζεται ως η διαφορά μεταξύ της πραγματικής παραγωγής και της προβλεπόμενης απόδοσης.

$$J(w) = p - \bar{p} \quad (2.12)$$

Η συνάρτηση που χρησιμοποιείται για τον υπολογισμό αυτού του σφάλματος είναι γνωστή ως Συνάρτηση Σφάλματος (Loss Function),  $J$ . Οι διαφορετικές συναρτήσεις σφάλματος θα δώσουν διαφορετικά σφάλματα για την ίδια πρόβλεψη και έτσι θα έχουν

σημαντική επίδραση στην απόδοση του μοντέλου. Η συνάρτηση σφάλματος είναι μία από τις δύο παραμέτρους που απαιτούνται για την κατάρτιση ενός μοντέλου (KERAS). Στην παρούσα διπλωματική εργασία γίνεται χρήση της `binary_crossentropy` και της `softmax_sparse_crossentropy_ignoring_last_label`.

- **Εντροπία**

Έστω ότι μια κατανομή είναι ένα εργαλείο που χρησιμοποιείται για την κωδικοποίηση συμβόλων, τότε η Εντροπία μετρά τον αριθμό των bits που θα είναι αναγκαίος αν γίνεται χρήση του σωστού εργαλείου  $y$ . Αυτό είναι βέλτιστο, δεδομένου ότι δεν είναι δυνατό να κωδικοποιηθούν τα σύμβολα χρησιμοποιώντας λιγότερα bits κατά μέσο όρο.

Αντίθετα η Σταυροειδής Εντροπία (Cross Entropy) είναι ο αριθμός των δυαδικών ψηφίων που θα είναι αναγκαίος αν κωδικοποιηθούν σύμβολα από το  $y$  χρησιμοποιώντας το λάθος εργαλείο  $\hat{y}$ . Φυσικά το αποτέλεσμα είναι η αναμενόμενη τιμή στην πραγματική κατανομή  $y$ , εφόσον η κατανομή αυτή είναι που πραγματικά δημιουργεί τα σύμβολα. Η σταυροειδής εντροπία είναι πάντα μεγαλύτερη από την εντροπία καθώς τα σύμβολα κωδικοποίησης σύμφωνα με τη λανθασμένη κατανομή  $\hat{y}$  οδηγούν στο να χρησιμοποιούνται περισσότερα κομμάτια. Η μόνη εξαίρεση είναι η ασήμαντη περίπτωση όπου τα  $y$  και  $\hat{y}$  είναι ίσα, στην περίπτωση αυτή η εντροπία και η σταυροειδής εντροπία είναι ίσες (DiPietro, 2016).

- **Binary Cross-Entropy**

Η συνάρτηση σφάλματος αυτή χρησιμοποιείται συνήθως σε προβλήματα δυαδικής ταξινόμησης (binary classification). Παρ'όλα αυτά μπορεί να χρησιμοποιηθεί και σε προβλήματα με περισσότερες από δύο κλάσεις, (multi-class), άμα είναι κωδικοποιημένα σε one-hot διανύσματα. Στην κωδικοποίηση αυτή κάθε κλάση αναπαριστάται από μία ξεχωριστή διάσταση. Κάθε ετικέτα (label) λοιπόν αναπαριστάται από ένα διάνυσμα με όσες διαστάσεις όσες είναι και οι κλάσεις του προβλήματος. Για να πραγματοποιήσουμε το μετασχηματισμό βάζουμε τη τιμή 1 στη διάσταση που αντιστοιχεί στην παλιά ετικέτα και 0 σε όλες τις υπόλοιπες.

Για παράδειγμα, σε ένα πρόβλημα με 4 τέσσερις κλάσεις έχουμε τις εξής ετικέτες:

$$y = \begin{pmatrix} 1 \\ 0 \\ 3 \\ 1 \\ 2 \\ 0 \end{pmatrix} \rightarrow y = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}$$

**Σχήμα 2.11:** Αριστερά μια “αραιή” (sparse) αναπαράσταση των κλάσεων. Δεξιά μια one-hot αναπαράσταση.

Σ'αυτή τη διπλωματική εργασία, η οποία πραγματεύεται ένα multi-class πρόβλημα γίνεται χρήση της Binary Cross-Entropy ως συνάρτηση σφάλματος (loss function). Ο λόγος λοιπόν που χρησιμοποιείται ο one-hot κωδικοποιητής είναι για να γίνει η δυαδικοποίηση της κατηγορίας και να συμπεριληφθεί ως χαρακτηριστικό για την εκπαίδευση του μοντέλου. Στο τελευταίο επίπεδο του νευρωνικού δικτύου χρησιμοποιείται η σιγμοειδής συνάρτηση (sigmoid function). Η σιγμοειδής συνάρτηση συνηθίζεται να χρησιμοποιείται με Binary Cross-Entropy.

- **Softmax-Sparse-Cross-Entropy-Ignoring-last-label**

Όπως αναφέρθηκε και στις παραπάνω παραγράφους στη διπλωματική χρησιμοποιήθηκε η `binary_cross_entropy` και η `softmax_sparse_crossentropy_ignoring_last_label`. Η Σταυροειδής Εντροπία (Cross Entropy) περιγράφηκε και παραπάνω.

Η συνάρτηση σφάλματος αυτή, απαιτεί οι ετικέτες να έχουν “αραιή” (sparse) αναπαράσταση, δηλαδή να μην έχουν one-hot κωδικοποίηση.

Σε πολλές εφαρμογές κατάτμησης εικόνων παρουσιάζεται το πρόβλημα της ανισορροπίας των κλάσεων όσον αφορά το background με τις υπόλοιπες κλάσεις. Αυτό το φαινόμενο οδηγεί πολλά μοντέλα που χρησιμοποιούν συμβατικές συναρτήσεις σφάλματος στο να προβλέπουν μόνο το background.

Η `softmax_sparse_crossentropy_ignoring_last_label` προσπαθεί να το επιλύσει το πρόβλημα αυτό, αγνοώντας εντελώς την τελευταία σε σειρά κλάση από τον υπολογισμό του σφάλματος. Για να χρησιμοποιήσουμε τη συνάρτηση αυτή, βέβαια, έπρεπε πρώτα να αλλάξουμε τη σειρά των κλάσεων έτσι ώστε το background να αντιστοιχεί στην τελευταία.

Στο τελευταίο επίπεδο του νευρωνικού δικτύου χρησιμοποιείται η `softmax` συνάρτηση ενεργοποίησης.

### 2.3.2 Μετρικές

Στη διπλωματική αυτή εργασία χρησιμοποιούνται διάφορες Μετρικές (Metrics). Μια μετρική είναι μια συνάρτηση που χρησιμοποιείται για να κρίνεται η απόδοση του μοντέλου. Με δεδομένες τις προβλεπόμενες τιμές και τις πραγματικές (ground truth)<sup>3</sup> τιμές η συνάρτηση αυτή δίνει ένα βαθμωτό μέτρο της καταλληλότητας του μοντέλου, με τα υπάρχοντα δεδομένα. Μια μετρική συνάρτηση είναι παρόμοια με τη συνάρτηση σφάλματος, εκτός από το ότι τα αποτελέσματα από την αξιολόγηση μιας μέτρησης δεν χρησιμοποιούνται κατά την εκπαίδευση του μοντέλου.

- **Πίνακας Σύγχυσης - Confusion Matrix**

Στο πεδίο της μηχανικής μάθησης και ειδικότερα του προβλήματος της στατιστικής ταξινόμησης, ένας πίνακας σύγχυσης (Confusion Matrix), επίσης γνωστός ως πίνακας σφάλματος, (Stehman, 1997) είναι μια συγκεκριμένη διάταξη πίνακα που επιτρέπει την απεικόνιση της απόδοσης ενός αλγορίθμου, συνήθως ενός επιβλεπόμενου εκπαιδευτικού (σε μη-επιβλεπόμενη εκπαίδευση, συνήθως ονομάζεται πίνακας αντιστοιχίας (Matching Matrix)). Κάθε γραμμή του πίνακα αντιπροσωπεύει τα στιγμιότυπα σε μια προβλεπόμενη κλάση ενώ κάθε στήλη αντιπροσωπεύει τα στιγμιότυπα σε μια πραγματική κλάση (ή αντίστροφα) (Powers, 2011). Το όνομα πηγάζει από το γεγονός ότι καθιστά εύκολο το να διαπιστωθεί εάν το σύστημα προκαλεί σύγχυση σε δύο κλάσεις (π.χ. συχνά δίνοντας λάθος ετικέτες η μία με την άλλη).

Πρόκειται για ένα ειδικός είδος πίνακα ενδεχομένων, με δύο διαστάσεις (“πραγματικό”, και “προβλεπόμενο”) και πανομοιότυπα σύνολα “κλάσεων” και στις δύο διαστάσεις (κάθε συνδυασμός διαστάσεων και κλάσης είναι μια μεταβλητή στον πίνακα ενδεχομένων).

---

<sup>3</sup>Η ground truth (βασική αλήθεια) είναι αυτό που μετριέται για τη μεταβλητή-στόχο για τα παραδείγματα εκπαίδευσης και δοκιμών. Συνήθως οι τιμές βασικής αλήθειας αντιμετωπίζονται όπως και ετικέτες. Υπάρχουν βέβαια και περιπτώσεις που δεν είναι ακριβώς η ίδια με την ετικέτα

		Actual Value (as confirmed by experiment)	
		positives	negatives
Predicted Value (predicted by the test)	positives	<b>TP</b> True Positive	<b>FP</b> False Positive
	negatives	<b>FN</b> False Negative	<b>TN</b> True Negative

Σχήμα 2.12: Πίνακας Σύγχυσης .

Παράδειγμα. Εάν ένα σύστημα ταξινόμησης έχει εκπαιδευτεί για να γίνει διάκριση μεταξύ γάτας, σκύλου και κουνελιών, ένας πίνακας σύγχυσεων θα συνοψίσει τα αποτελέσματα της δοκιμής του αλγορίθμου για περαιτέρω επιθεώρηση. Υποθέτοντας ένα δείγμα από 27 ζώα - 8 γάτες, 6 σκύλοι και 13 κουνέλια, ο προκύπτων πίνακας σύγχυσης μπορεί να μοιάζει με τον παρακάτω πίνακα:

		Actual class		
		Cat	Dog	Rabbit
Predicted class	Cat	5	2	0
	Dog	3	3	2
	Rabbit	0	1	11

Σχήμα 2.13: Πίνακας Σύγχυσης .

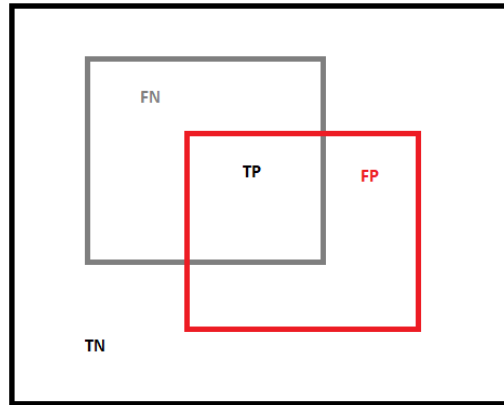
Σε αυτόν τον πίνακα σύγχυσης, από τις 8 πραγματικές γάτες, το σύστημα πρόβλεπε ότι 3 ήταν σκυλιά και από τα 6 σκυλιά, πρόβλεπε ότι 1 ήταν κουνέλι και 2 ήταν γάτες. Είναι εύκολο να δει κανείς από τον πίνακα ότι το εν λόγω σύστημα δυσκολεύει να διακρίνει τις γάτες από τα σκυλιά, αλλά μπορεί να κάνει τη διάκριση αναμέσα σε κουνέλια και άλλους τύπους ζώων αρκετά καλά. Όλες οι σωστές προβλέψεις εντοπίζονται στη διαγώνιο του πίνακα (επισημαίνονται με έντονους χαρακτήρες), επομένως είναι εύκολο να ελεγχθεί οπτικά ο πίνακας για σφάλματα πρόβλεψης, καθώς θα αντιπροσωπεύονται από τιμές εκτός της διαγωνίου.

**True Positive-TP:** Αληθώς θετική ονομάζεται μια πρόβλεψη που εκτιμήθηκε ότι ανήκει σε μια συγκεκριμένη κλάση και αυτό ισχύει. Για παράδειγμα στο Σχήμα 2.13 τα αληθώς θετικά για τις γάτες είναι 5, για τα σκυλιά είναι 3 και για κουνέλια είναι 11.

**False Positive-FP:** Ψευδώς θετική ονομάζεται μια πρόβλεψη που εκτιμήθηκε ότι ανήκει σε μια συγκεκριμένη κλάση και αυτό δεν ισχύει. Για παράδειγμα στο Σχήμα 2.13 τα FP για τις γάτες είναι 2, για τους σκύλους είναι 5 και για τα κουνέλια είναι 1.

**False Negative-FN:** Ψευδώς αρνητική ονομάζεται μια πρόβλεψη που εκτιμήθηκε ότι δεν ανήκει σε μια συγκεκριμένη κλάση και αυτό δεν ισχύει. Για παράδειγμα στο Σχήμα 2.13 τα FN για τις γάτες είναι 3, για τους σκύλους 3 και για τα κουνέλια 2.

**True Negative-TN:** Αληθώς αρνητική ονομάζεται μια πρόβλεψη που εκτιμήθηκε ότι δεν ανήκει σε μια κλάση και αυτό ισχύει. Για παράδειγμα στο Σχήμα 2.13 τα TN για τις γάτες είναι 17 (4 από την κλάση των σκύλων, 13 από την κλάση των κουνελιών). Για τους σκύλους είναι 16 (5 από την κλάση των γάτων και 11 από την κλάση των κουνελιών), ενώ για τα κουνέλια είναι 13 (8 από την κλάση των γάτων και 5 από την κλάση των σκύλων).



**Σχήμα 2.14:** *FP-FN-TP-TN*, Η κλάση είναι το γκρι κουτάκι και το TP είναι στην τομή από το γκρι και το κόκκινο κουτάκι, οι προβλέψεις που εκτιμήθηκαν ότι ανήκουν στην κλάση και ισχύει. Στο γκρι κουτάκι το FN είναι οι προβλέψεις που εκτιμήθηκαν ότι δεν ανήκουν στην κλάση και αυτό δεν ισχύει. Στο κόκκινο κουτάκι οι προβλέψεις που εκτιμήθηκαν ότι ανήκουν στην κλάση αλλά δεν ισχύει. Τέλος έξω από όλα τα εσωτερικά κουτάκια και μέσα στο μαύρο κουτάκι είναι το TN που είναι οι προβλέψεις που εκτιμήθηκαν ότι δεν ανήκουν στην κλάση και ισχύει..

Οι μετρικές που χρησιμοποιούνται στην εργασία αυτή είναι: η “accuracy”, η “accuracy\_ignoring\_last\_label”, η “jaccard\_coeff”, το precision, το recall και η f1.

Οι παρακάτω μετρικές υπολογίζονται ανά κλάση.

- **accuracy (ακρίβεια)**

Η ακρίβεια είναι μια μετρική η οποία αξιολογεί όπως δηλώνει και το όνομα της, πόσο ακριβής είναι η πρόβλεψη του μοντέλου σε σχέση με τα πραγματικά δεδομένα. Δηλαδή, η ακρίβεια είναι ο λόγος των πραγματικών αποτελεσμάτων (τόσο αληθινών όσο και αρνητικών) μεταξύ του συνολικού αριθμού των περιπτώσεων που εξετάστηκαν.

$$\frac{TP + TN}{TP + FP + FN + TN} \quad (2.13)$$

- **accuracy\_ignoring\_last\_label**

Η μετρική αυτή υπολογίζει την ακρίβεια αγνοώντας τη τελευταία ετικέτα.

$$\frac{1}{M-1} \cdot \sum_{i=1}^{M-1} \left( \frac{TP_i}{TP_i + FP_i + FN_i} \right) \quad (2.14)$$

Στην παραπάνω εξίσωση τα  $TP_i, FP_i, FN_i$  είναι τα  $TP, FP, FN$  της κλάσης  $i$ , όπου  $i \in \{1, 2, \dots, M\}$ .

- **precision**

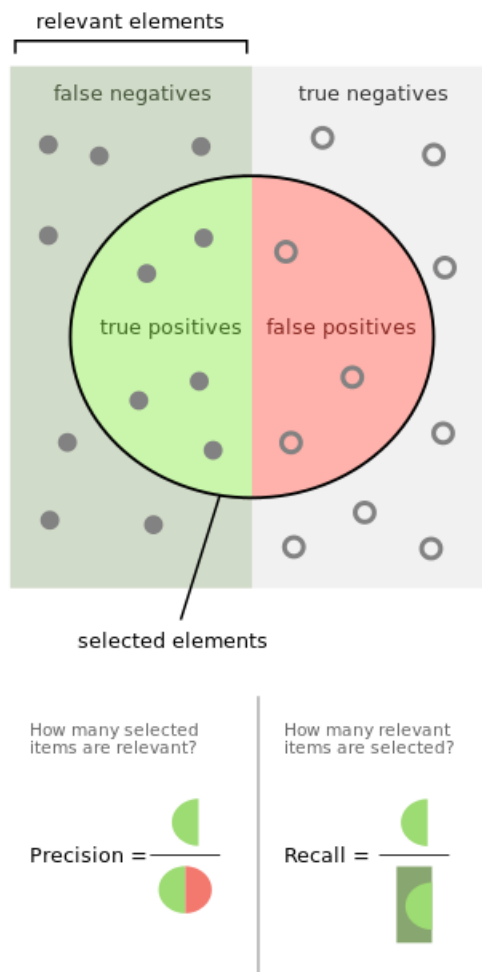
Δείχνει το ποσοστό των επιτυχών προς των συνολικών προβλέψεων μιας κλάσης από ένα μοντέλο. Το precision μπορεί να θεωρηθεί ως μέτρο ποιότητας.

$$\frac{TP}{TP + FP} \quad (2.15)$$

- **recall**

Η ανάκληση (γνωστή και ως ευαισθησία-sensitivity) είναι μια παραδοσιακή μέτρηση αξιολόγησης, η οποία όπως όλα τα μέτρα, υποθέτουν μια έννοια βασικής αλήθειας (ground truth). Είναι το κλάσμα των επιτυχών προβλέψεων μιας κλάσης προς τα συνολικά παραδείγματα αυτής. Η ανάκληση μπορεί να θεωρηθεί ως μέτρο πληρότητας ή ποσότητας. Χρησιμοποιείται και αυτή στο κομμάτι της αξιολόγησης.

$$\frac{TP}{TP + FN} \quad (2.16)$$

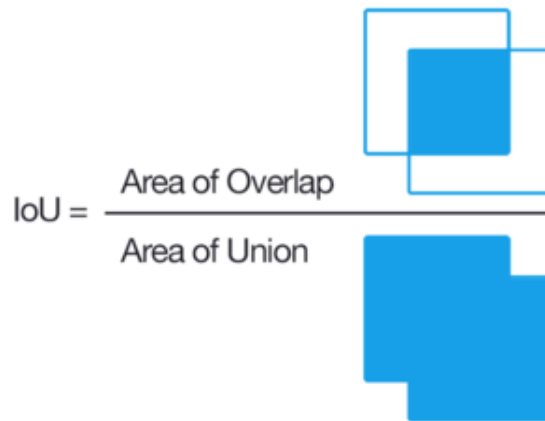


Σχήμα 2.15: Precision-Ακρίβεια και Recall-Ανάκληση .

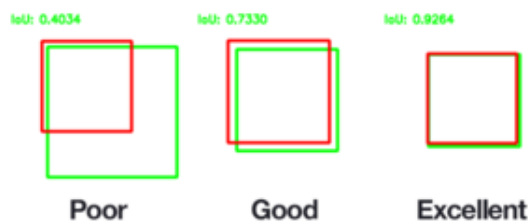
- **Jaccard coefficient ή Intersection Over Union**

Η μετρική αυτή είναι ένα στατιστικό στοιχείο που χρησιμοποιείται για τη σύγκριση της ομοιότητας και της ποικιλομορφίας των συνόλων δειγμάτων. Ο συντελεστής Jaccard μετρά την ομοιότητα μεταξύ συνόλων πεπερασμένων δειγμάτων και ορίζεται ως το μέγεθος της τομής διαιρούμενο με το μέγεθος της ένωσης των συνόλων δειγμάτων. Το μέγεθος της τομής που αντιστοιχεί στο True Positive ονομάζεται και intersection.

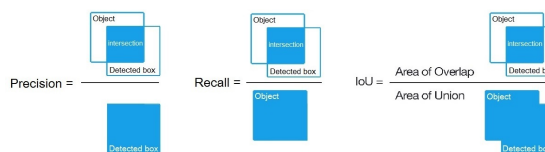
$$\frac{TP}{TP + FP + FN} \quad (2.17)$$



Σχήμα 2.16: Στο σχήμα φαίνεται η εξίσωση της μετρικής jaccard .



Σχήμα 2.17: Στο σχήμα φαίνεται η επίδοση της μετρικής jaccard .



Σχήμα 2.18: Στο σχήμα φαίνονται οι εξισώσεις της ακρίβειας, της ανάκλησης και του IOU.

- **f1 measure**

Η μετρική αυτή χρησιμοποιείται στο κομμάτι της αξιολόγησης και συνδυάζει την ακρίβεια και την ανάκληση. Συγκεκριμένα είναι ο αρμονικός μέσος όρος της ακρίβειας και της ανάκλησης. Η f1 φτάνει την καλύτερη τιμή της στο 1 (τέλεια ακρίβεια και ανάκληση) και χειρότερη στο 0.

$$\frac{2 \cdot Precision \cdot Recall}{Precision + Recall} \quad (2.18)$$



## 2.4 Βελτιστοποίηση

Οι αλγόριθμοι βελτιστοποίησης ελαχιστοποιούν μια συνάρτηση σφάλματος  $J(x)$ , η οποία είναι απλώς μια μαθηματική συνάρτηση που εξαρτάται από τις εσωτερικές παραμέτρους του μοντέλου, που χρησιμοποιούνται για τον υπολογισμό των τιμών στόχων ( $Y$ ) από το σύνολο των προγνωστικών τιμών ( $X$ ) που χρησιμοποιούνται στο μοντέλο. Οι εσωτερικές παράμετροι ενός μοντέλου διαδραματίζουν πολύ σημαντικό ρόλο στην αποτελεσματική και αποδοτική εκπαίδευση του μοντέλου και παράγουν ακριβή αποτελέσματα. Αυτός είναι ο λόγος που χρησιμοποιούμε διάφορες στρατηγικές και αλγορίθμους βελτιστοποίησης για να ενημερώσουμε και να υπολογίσουμε τις κατάλληλες και βέλτιστες τιμές των παραμέτρων αυτού του μοντέλου που επηρεάζουν τη διαδικασία μάθησης του μοντέλου μας καθώς και την έξοδο του. Στην διπλωματική εργασία οι αλγόριθμοι βελτιστοποίησης που χρησιμοποιούνται είναι η SGD και ο Adam.

### 2.4.1 Gradient Descent

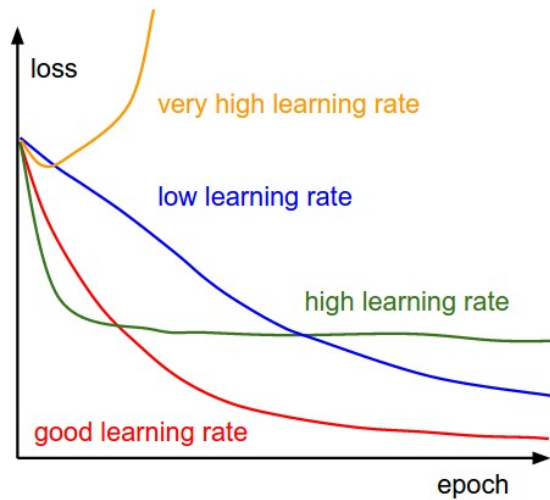
Η Gradient Descent είναι η πιο σημαντική τεχνική και το θεμέλιο του τρόπου με τον οποίο εκπαιδεύουμε και βελτιστοποιούμε τα Ευφυή Συστήματα. Είναι ο πιο δημοφιλής αλγόριθμος βελτιστοποίησης ενός νευρικού δικτύου. Χρησιμοποιείται σε μεγάλο βαθμό για την πραγματοποίηση ενημερώσεων των βαρών σε ένα μοντέλο νευρωνικών δικτύων, δηλαδή την ενημέρωση και τον συντονισμό των παραμέτρων του μοντέλου σε μια κατεύθυνση ώστε να ελαχιστοποιήσουμε τη συνάρτηση σφάλματος.

Ταυτόχρονα, κάθε σύγχρονη βιβλιοθήκη Deep Learning περιλαμβάνει υλοποιήσεις διαφόρων αλγορίθμων για τη βελτιστοποίηση της Gradient Descent (π.χ. caffe και keras). Αυτοί οι αλγόριθμοι, ωστόσο, χρησιμοποιούνται συχνά ως βελτιστοποιητές μαύρου κουτιού, καθώς οι πρακτικές εξηγήσεις σχετικά με τις δυνατότητες και αδυναμίες είναι δύσκολο να γίνουν.

Η Gradient Descent είναι ένας τρόπος ελαχιστοποίησης μιας αντικειμενικής συνάρτησης  $J(\theta)$  παραμετροποιημένης από τις παραμέτρους του μοντέλου  $\theta \in \mathbb{R}^d$  με την ενημέρωση των παραμέτρων στην αντίθετη κατεύθυνση της κλίσης της αντικειμενικής συνάρτησης  $\nabla_{\theta} J(\theta)$  στις παραμέτρους.

Ο **ρυθμός εκμάθησης - learning rate**  $\eta$  είναι μια **υπερ-παράμετρος** που ελέγχει πόσο προσαρμόζονται τα βάρη του δικτύου σε σχέση με την κλίση της απώλειας loss gradient. Καθορίζει το μέγεθος των βημάτων που παίρνονται για να προσεγγιστεί ένα (τοπικό) ελάχιστο. Με άλλα λόγια, ακολουθείται η κατεύθυνση της κλίσης της επιφάνειας που δημιουργείται από την αντικειμενική συνάρτηση προς τα κάτω μέχρι να προσεγγιστεί η κοιλάδα. Η επιλογή κατάλληλου ρυθμού μάθησης μπορεί να είναι δύσκολη. Ένας πολύ χαμηλός ρυθμός μάθησης οδηγεί σε οδυνηρά αργή σύγκλιση, ενώ ένας ρυθμός εκμάθησης που είναι πολύ μεγάλος μπορεί να εμποδίσει τη σύγκλιση και να προκαλέσει τη διακύμανση της συνάρτησης απώλειας στο ελάχιστο ή ακόμα και την απόκλιση. Προγράμματα ρυθμού εκμάθησης ([Robbins and Monro, 1985](#)) προσπαθούν να προσαρμόσουν τον ρυθμό εκμάθησης κατά τη διάρκεια της εκπαίδευσης, π.χ. με ανόπτηση, δηλαδή μείωση του ρυθμού εκμάθησης σύμφωνα με ένα προκαθορισμένο χρονοδιάγραμμα ή όταν η αλλαγή στόχου μεταξύ των εποχών πέσει κάτω από ένα κατώφλι. Ωστόσο τα προγράμματα και τα κατώφλια πρέπει να καθοριστούν εκ των προτέρων και συνεπώς δεν είναι σε θέση να προσαρμοστούν στα χαρακτηριστικά ενός συνόλου δεδομένων ([Darken et al., 1992](#)). Επιπλέον, ο ίδιος ρυθμός εκμάθησης ισχύει για όλες τις ενημερώσεις παραμέτρων. Εάν τα δεδομένα είναι αραιά και τα χαρακτηριστικά έχουν πολύ διαφορετικές συχνότητες, ίσως να μην

είναι επιθυμητό να ενημερώνονται όλα τους στον ίδιο βαθμό, αλλά να εκτελείται μια μεγαλύτερη ενημέρωση για σπάνια χαρακτηριστικά.



**Σχήμα 2.19:** Η επίδραση διαφόρων μαθησιακών ρυθμών στη σύγκλιση.

Υπάρχουν τρεις παραλλαγές της Gradient Descent, οι οποίες διαφέρουν ως προς το πόσα δεδομένα χρησιμοποιούνται για να υπολογιστεί η κλίση της αντικειμενικής συνάρτησης. Ανάλογα με την ποσότητα των δεδομένων, γίνεται μια ανταλλαγή μεταξύ της ακρίβειας της ενημέρωσης των παραμέτρων και του χρόνου που απαιτείται για την εκτέλεση μιας ενημέρωσης.

## 2.4.2 Batch Gradient Descent

Η Vanilla gradient descent, γνωστή ως batch gradient descent, υπολογίζει την κλίση της συνάρτησης κόστους στις παραμέτρους  $\theta$  για ολόκληρο το σύνολο δεδομένων εκπαίδευσης:

$$\theta \leftarrow \theta - \eta \cdot \nabla_{\theta} J(\theta) \quad (2.19)$$

Καθώς πρέπει να υπολογιστούν οι κλίσεις-gradients για ολόκληρο το σύνολο δεδομένων για να εκτελεστεί μόνο μία ενημέρωση, η batch gradient descent μπορεί να είναι πολύ αργή και είναι αδύνατο για τα σύνολα δεδομένων που δεν ταιριάζουν στη μνήμη. Η batch gradient descent δεν επιτρέπει επίσης στο να ενημερώνεται online το μοντέλο, δηλαδή με νέα παραδείγματα επί τόπου.

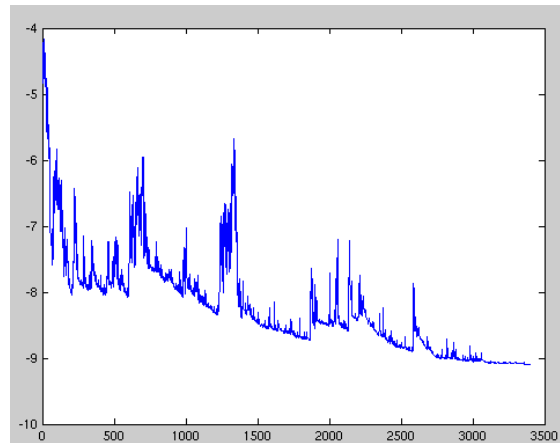
## 2.4.3 Stochastic Gradient Descent-SGD

Η Stochastic Gradient Descent - SGD εκτελεί μια ενημέρωση παραμέτρων για κάθε παράδειγμα εκπαίδευσης. Είναι συνήθως πολύ πιο γρήγορη τεχνική. Εκτελεί μία ενημέρωση κάθε φορά για κάθε παράδειγμα εκπαίδευσης  $x^{(i)}$  και ετικέτα  $y^{(i)}$ :

$$\theta \leftarrow \theta - \eta \cdot \nabla_{\theta} J(\theta; x^{(i)}; y^{(i)}) \quad (2.20)$$

Η batch gradient descent εκτελεί περιττούς υπολογισμούς για μεγάλα σύνολα δεδομένων, διότι επαναπροσδιορίζει τις κλίσεις για παρόμοια παραδείγματα πριν από

κάθε ενημέρωση παραμέτρων. Η SGD απομακρύνει αυτή την πλεοναστικότητα εκτελώντας μία ενημέρωση τη φορά. Είναι επομένως συνήθως πολύ ταχύτερη και μπορεί επίσης να χρησιμοποιηθεί για να μάθει online. Η SGD εκτελεί συχνές ενημερώσεις με μεγάλη διακύμανση με συνέπεια οι ενημερώσεις παραμέτρων να έχουν μεγάλη διακύμανση η οποία προκαλεί μεγάλη αντικειμενική μεταβολή της αντικειμενικής λειτουργίας όπως στην εικόνα 2.20:



Σχήμα 2.20: Διακυμάνσεις SGD. (Ruder, 2016).

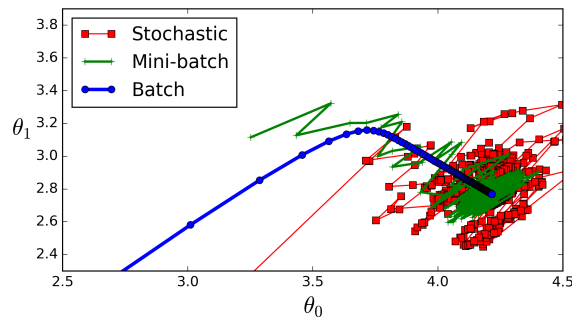
Οι διακυμάνσεις της SGD προκαλούν τη διακύμανση της συνάρτησης σφάλματος σε διαφορετικές εντάσεις. Αυτό είναι πράγματι καλό γιατί βοηθά να ανακαλυφθούν νέα και ίσως καλύτερα τοπικά ελάχιστα (Walia, 2017b). Από την άλλη πλευρά, αυτό τελικά περιπλέκει τη σύγκλιση στο ακριβές ελάχιστο καθώς η SGD θα συνεχίσει να υπερβαίνει. Ωστόσο, έχει αποδειχθεί ότι όταν μειώνεται αργά ο ρυθμός εκμάθησης, το SGD σχεδόν σίγουρα συγκλίνει σε ένα τοπικό ή παγκόσμιο ελάχιστο για μη κυρτή και κυρτή βελτιστοποίηση αντίστοιχα (Ruder, 2016).

### Μέγεθος της Παρτίδας (Batch Size)

Ορίζει τον αριθμό των δειγμάτων που βλέπει το μοντέλο πριν ενημερώσει τα βάρη. Για παράδειγμα, έστω ότι έχουμε 1050 δείγματα εκπαίδευσης και μέγεθος παρτίδας ίσο με 100. Ο αλγόριθμος παίρνει τα πρώτα 100 δείγματα (από το 1ο έως το 100ο) από το σύνολο δεδομένων εκπαίδευσης και εκπαιδεύει το δίκτυο. Στη συνέχεια παίρνει τα δεύτερα 100 δείγματα (από το 101ο έως το 200ο) και πάλι εκπαιδεύει το δίκτυο. Αυτή η διαδικασία μπορεί να συνεχιστεί μέχρι να διαδωθεί μέσω των δικτύων όλα τα δείγματα. Το πρόβλημα συνήθως συμβαίνει με το τελευταίο σετ δειγμάτων. Στο συγκεκριμένο παράδειγμα που αναφέρθηκε υπάρχουν 1050 δείγματα, που όμως το 1050 δεν διαρείται με το 100 χωρίς υπόλοιπο. Η απλούστερη λύση είναι απλά να περαστούν τα τελικά 50 δείγματα στον αλγόριθμο και να εκπαιδευτεί το δίκτυο. Σε αυτό το σημείο ο αλγόριθμος έχει εκτελέσει μια εποχή.

Με αυτό τον τρόπο απαιτείται λιγότερη μνήμη, δεδομένου ότι εκπαιδεύεται δίκτυο χρησιμοποιώντας μικρότερο αριθμό δειγμάτων. Είναι ιδιαίτερα σημαντικό σε περίπτωση που δεν υπάρχει η δυνατότητα να προσαρμοστεί το σύνολο δεδομένων στη μνήμη. Συνήθως τα δίκτυα εκπαιδεύονται ταχύτερα με μίνι-παρτίδες (mini-batches). Αυτό επειδή ενημερώνονται τα βάρη μετά από κάθε διάδοση. Στο παράδειγμα αυτό έχουν διαδοθεί 11 παρτίδες (10 από αυτά είχαν 100 δείγματα και 1 είχε 50 δείγματα) και μετά από κάθε μία από αυτές ενημερώθηκαν οι παράμετροι του δικτύου. Εάν χρησιμοποιούνταν όλα τα δείγματα κατά τη διάδοση, θα γινόταν μόνο 1 ενημέρωση για την παράμετρο του δικτύου.

Βέβαια όσο μικρότερη είναι η παρτίδα τόσο λιγότερο ακριβής είναι η εκτίμηση της κλίσης.



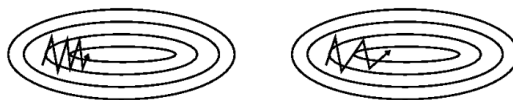
**Σχήμα 2.21:** Στο σχήμα είναι εύκολο να παρατηρήσει κάποιος ότι η κατεύθυνση κλίσης της μίνι-παρτίδας (πράσινο χρώμα) κυμαίνεται σε σύγκριση με την πλήρη παρτίδα (μπλε χρώμα).

Η **Stochastic Gradient Descent - SGD** είναι μόνο μία μίνι-παρτίδα (mini-batch) με μέγεθος (batch-size) ίσο με 1. Η κλίση αλλάζει την κατεύθυνση της ακόμη πιο συχνά από μια μίνι-παρτίδα.

Όταν γίνεται η εκπαίδευση (training) του νευρωνικού δικτύου, ή η αξιολόγηση (evaluation) η διάδοση των δεδομένων γίνεται με τις παρτίδες (batches) όπως αναφέρθηκε και παραπάνω. Επίσης από τα παραπάνω είναι προφανές ότι το νευρωνικό δίκτυο θα τροφοδοτηθεί με όλες τις παρτίδες. Η χρήση όλων των παρτίδων μία φορά είναι 1 εποχή. Εάν λοιπόν υπάρχουν 10 εποχές σημαίνει ότι θα χρησιμοποιηθούν όλα τα δεδομένα (χωριστά σε παρτίδες) 10 φορές.

### Momentum-Ορμή

Η SGD έχει πρόβλημα πλοήγησης σε χαράδρες, δηλαδή περιοχές όπου η επιφάνεια καμπυλώνεται πολύ πιο απότομα σε μια διάσταση από ότι σε μια άλλη (Sutton, 1986), οι οποίες είναι κοινές γύρω από το τοπικό βέλτιστο. Σε αυτά τα σενάρια, η SGD ταλαντώνεται στις πλαγιές της χαράδρας ενώ παράλληλα κάνει διστακτική πρόοδο κατά μήκος του βυθού προς το τοπικό βέλτιστο.



**Σχήμα 2.22:** Η *sgd* χωρίς ορμή-momentum αριστερά και η *sgd* με ορμή-momentum δεξιά .

Η ορμή-momentum (Qian, 1999) είναι μια μέθοδος που βοηθά στην επιτάχυνση της *sgd* στη σχετική κατεύθυνση και μειώνει τις ταλαντώσεις όπως φαίνεται και στη δεξιά εικόνα 2.22. Αυτό επιτυγχάνεται προσθετόντας ένα κλάσμα  $\gamma$  του διανύσματος ενημέρωσης του προηγούμενου βήματος χρόνου στο τρέχον διάνυσμα ενημέρωσης:

$$\begin{aligned} v_t &= \gamma v_{t-1} + \eta \nabla_{\theta} J(\theta) \\ \theta &= \theta - v_t \end{aligned} \quad (2.21)$$

Σημείωση: Ορισμένες εφαρμογές ανταλλάσσουν τα σημεία στις εξισώσεις. Ο όρος ορμής  $\gamma$  ορίζεται συνήθως σε 0.9 ή σε κάποια παρόμοια τιμή.

Ουσιαστικά όταν χρησιμοποιείται η ορμή, είναι σα να πιέζεται μια σφαίρα κάτω σε ένα λόφο. Η σφαίρα συσσωρεύει την ορμή, καθώς κυλάει προς τα κάτω, γρηγορότερα και ταχύτερα στο δρόμο (μέχρι να φτάσει στην τελική της ταχύτητας αν υπάρχει αντίσταση αέρα, δηλαδή  $\gamma < 1$ ). Το ίδιο συμβαίνει με τις ενημερώσεις των παραμέτρων μας: Ο όρος ορμή αυξάνεται για τις διαστάσεις των οποίων οι κλίσεις δείχνουν προς τις ίδιες κατευθύνσεις και μειώνουν τις ενημερώσεις για τις διαστάσεις των οποίων οι κλίσεις αλλάζουν κατευθύνσεις. Ως αποτέλεσμα, επιτυγχάνεται ταχύτερη σύγκλιση και μειωμένη ταλάντωση.

## 2.4.4 Adam

Η επιλογή αλγορίθμου βελτιστοποίησης για ένα μοντέλο βαθιάς μάθησης μπορεί να σημαίνει τη διαφορά ανάμεσα σε χρόνους σύγκλισης λεπτών, ωρών και ημερών.

Ο αλγόριθμος βελτιστοποίησης Adam είναι μια επέκταση της sgd (2.4.3) και χρησιμοποιείται για την ενημέρωση των βαρών του δικτύου με βάση τα δεδομένα εκπαίδευσης. Ο Adam παρουσιάστηκε από τους Diederik Kingma και Jimmy Ba (Kingma and Ba, 2014).

Ο Adam είναι διαφορετικός από την κλασική sgd. Από τη μία η sgd διατηρεί ένα μοναδικό ρυθμό εκμάθησης (ονομάζεται άλφα) για όλες τις ενημερώσεις βάρους και ο ρυθμός εκμάθησης δεν αλλάζει κατά τη διάρκεια της εκπαίδευσης. Το παραπάνω δημοσίευμα περιγράφει ότι ο Adam λειτουργεί ως συνδυασμός των πλεονεκτημάτων των δύο άλλων επεκτάσεων της sgd.

- Adaptive Gradient Algorithm - AdaGrad που διατηρεί ρυθμό εκμάθησης ανά παράμετρο που βελτιώνει την απόδοση σε προβλήματα με αραιές κλίσεις (π.χ. προβλήματα φυσικής γλώσσας και προβλήματα υπολογιστικής όρασης).
- Root Mean Square Propagation - RMSProp που διατηρεί επίσης ποσοστά εκμάθησης ανά παράμετρο που προσαρμόζονται με βάση το μέσο όρο των πρόσφατων μεγεθών των διαβαθμίσεων για το βάρος (π.χ. πόσο γρήγορα αλλάζει). Αυτό σημαίνει ότι ο αλγόριθμος λειτουργεί καλά σε επιγραμμικά και μη στάσιμα προβλήματα (π.χ. θορυβώδη).

Ο Adam εκμεταλλεύεται τα οφέλη τόσο της AdaGrad όσο και της RMSProp. Είναι πολύ δημοφιλής αλγόριθμος στον τομέα της βαθιάς μάθησης επειδή επιτυγχάνει γρήγορα και καλά αποτελέσματα. Υπολογίζει τα ποσοστά προσαρμοστικής εκμάθησης για κάθε παράμετρο. Εκτός από την αποθήκευση ενός εκθετικά φθίνοντος μέσου όρου, ο Adam διατηρεί επίσης έναν εκθετικά φθίνοντα μέσο όρο προηγούμενων βαθμίδων  $m_t$ , παρόμοιο με το momentum. Ενώ το momentum μπορεί να θεωρηθεί σαν μια μπάλα που τρέχει σε μια πλαγιά, ο Adam συμπεριφέρεται σαν μια βαριά μπάλα με τριβή, η οποία προτιμά τα επίδεδα ελάχιστα στην επιφάνεια σφάλματος. Υπολογίζουμε τους μέσους βαθμούς απόσβεσης των προηγούμενων και παρελθόντων τετραγωνικών διαβαθμίσεων  $m_t$  και  $v_t$  αντιστοίχως ως εξής:

$$\begin{aligned} m_t &= \beta_1 m_{t-1} + (1 - \beta_1) g_t \\ v_t &= \beta_2 v_{t-1} + (1 - \beta_2) g_t^2 \end{aligned} \tag{2.22}$$

$m_t$  και  $u_t$  είναι οι εκτιμήσεις της πρώτης στιγμής (μέσος όρος) και της δεύτερης στιγμής (της μη συγκεντρωμένης απόκλισης) των κλίσεων αντίστοιχα, εξ ου και το όνομα της μεθόδου. Δεδομένου ότι τα  $m_t$  και  $u_t$  αρχικοποιούνται ως φορείς των 0, οι συγγραφείς του Adam παρατηρούν ότι είναι προκατειλημμένοι προς το μηδέν, ειδικά κατά τη διάρκεια των αρχικών χρονικών βημάτων και ειδικά όταν οι ρυθμοί αποσύνθεσης είναι μικροί (δηλαδή  $\beta_1$  και  $\beta_2$  είναι κοντά στο 1).

Αντισταθμίζουν αυτές τις προκαταλήψεις με υπολογισμό των διορθωμένων προκαταρκτικών εκτιμήσεων πρώτης και δεύτερης στιγμής:

$$\begin{aligned}\hat{m}_t &= \frac{m_t}{1 - \beta_1^t} \\ \hat{v}_t &= \frac{v_t}{1 - \beta_2^t}\end{aligned}\tag{2.23}$$

Στη συνέχεια τα χρησιμοποιούν για να ενημερώσουν τις παραμέτρους όπως έχουμε δει στα Adadelta και RMSprop, τα οποία αποδίδουν τον κανόνα ενημέρωσης του Adam:

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t\tag{2.24}$$

Οι συγγραφείς του Adam προτείνουν προκαθορισμένες τιμές 0.9 για  $\beta_1$  και 0.999 για  $\beta_2$  και  $10^{-8}$  για  $\epsilon$ . Δείχνουν εμπειρικά ότι ο Adam λειτουργεί καλά στην πράξη και συγκρίνεται ευνοϊκά με άλλους αλγόριθμους προσαρμοστικής μεθόδου μάθησης ([Ruder, 2016](#)).

# Κεφάλαιο 3

## Μοντέλα

Σε αυτή την ενότητα περιγράφονται οι αρχιτεκτονικές βαθιών συνελικτικών δικτύων που χρησιμοποιήθηκαν στην εργασία, όπως το UNet και το DenseNet.

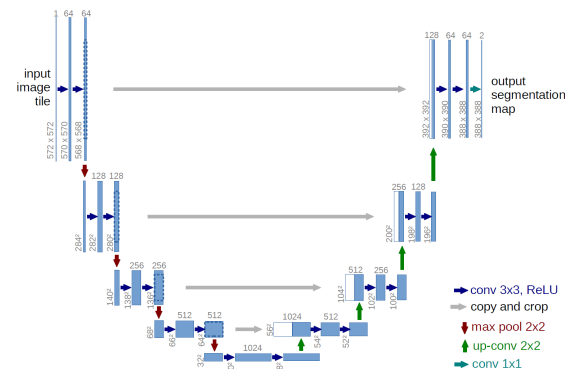
### 3.1 UNet

Το UNet (Ronneberger et al., 2015) είναι μια δημοφιλής αρχιτεκτονική που χρησιμοποιείται ευρέως και επιτυγχάνει εξαιρετική απόδοση για την κατάτμηση ιατρικών εικόνων. Η αρχιτεκτονική αυτή είναι μια πιο κομψή αρχιτεκτονική, το λεγόμενο “πλήρως συνελικτικό δίκτυο” (fully convolutional network).

Η κύρια ιδέα είναι να συμπληρώσει ένα συνηθισμένο συνελικτικό δίκτυο με διαδοχικά επίπεδα, όπου οι φορείς συγκέντρωσης (pooling operators) αντικαθίστανται από φορείς υπερδειγματοληψίας (upsampling operators). Ως εκ τούτου, αυτά τα επίπεδα αυξάνουν την ανάλυση εξόδου. Προκειμένου να συγχωνευθούν διαφορετικά επίπεδα πληροφοριών, χαρακτηριστικά υψηλής ανάλυσης από τη συμβιβαστική πορεία συνδυάζονται με την έξοδο της υπερδειγματοληψίας (upsampling). Αυτά τα συνδυασμένα επίπεδα μπορούν έπειτα να μάθουν πληροφορίες περιβάλλοντος. Το τμήμα υπερδειγματοληψίας επιτρέπει στο δίκτυο τη διάδοση των πληροφοριών πλαισίου σε επίπεδα υψηλότερης ανάλυσης. Ως συνέπεια η επεκτατική διαδρομή και η πορεία συμβολής αποδίδουν μια αρχιτεκτονική σχήματος U. Το δίκτυο δεν έχει πλήρως συνδεδεμένα στρώματα και χρησιμοποιεί μόνο το έγκυρο τμήμα κάθε συνέλιξης, δηλαδή ο χάρτης αντιστοιχίσεων περιέχει μόνο τα εικονοστοιχεία για τα οποία το πλήρες κείμενο είναι διαθέσιμο στην εικόνα εισόδου. Αυτή η στρατηγική επιτρέπει την ομαλή κατάτμηση αυθαίρετα μεγάλων εικόνων.

Η αρχιτεκτονική δικτύου αποτελείται από μια συμβιβαστική πορεία (αριστερή πλευρά) και μια επεκτατική διαδρομή (δεξιά πλευρά). Η συμβιβαστική πορεία ακολουθεί τη τυπική αρχιτεκτονική ενός συνελικτικού δικτύου. Αποτελείται από την επαναλαμβανόμενη εφαρμογή δύο  $3 \times 3$  συνελίξεων (unpadded convolutions) που ακολουθούνται από διορθωμένη γραμμική μονάδα - rectified linear unit (ReLU) και μία  $2 \times 2$  μέγιστη λειτουργία συγκέντρωσης - max pooling operation με βήμα (stride) 2. Σε κάθε βήμα υποδειγματοληψίας διπλασιάζεται ο αριθμός των χαρακτηριστικών καναλιών. Κάθε βήμα στο επεκτατικό στάδιο αποτελείται από μια υπερδειγματοληψία του χάρτη χαρακτηριστικών ακολουθούμενη από μια συνέλιξη (convolution)  $2 \times 2$  (“up-convolution”) που μοιράζει τον αριθμό των καναλιών χαρακτηριστικών, μια συσχέτιση με τον αντίστοιχο περικομμένο χάρτη χαρακτηριστικών από τη συμβιβαστική πορεία και δύο  $3 \times 3$  συνελίξεις (convolutions), καθένα ακολουθούμενο από ένα ReLU. Η περικοπή είναι απαραίτητη λόγω της απώλειας των συνοριακών

εικονοστοιχείων σε κάθε συνέλιξη. Στο τελικό επίπεδο χρησιμοποιείται μια συνέλιξη  $1 \times 1$  για να αντιστοιχίσει κάθε 64-φορέα χαρακτηριστικών συνιστωσών στον επιθυμητό αριθμό κλάσεων. Συνολικά το δίκτυο έχει 23 επίπεδα συνέλιξης (convolutional layers) (Ronneberger et al., 2015).



**Σχήμα 3.1:** U-net αρχιτεκτονική (παράδειγμα για  $32 \times 32$  pixels στη χαμηλότερη ανάλυση). Κάθε μπλε πλαίσιο αντιστοιχεί σε ένα χάρτη πολλαπλών καναλιών. Ο αριθμός των καναλιών σημειώνεται στην κορυφή του πλαισίου. Το μέγεθος  $x$ - $y$  παρέχεται στο κάτω αριστερό άκρο του πλαισίου. Τα λευκά πλαίσια αντιπροσωπεύουν τους αντιγραμμένους χάρτες χαρακτηριστικών. Τα βέλη υποδηλώνουν τις διάφορες λειτουργίες. (Ronneberger et al., 2015).

Στην εργασία αυτή χρησιμοποιείται η αρχιτεκτονική U-net με μια τροποποίηση στα τμήματα δειγματοληψίας για την αυτόματη προσαρμογή του πλάτους και του ύψους του χάρτη χαρακτηριστικών όταν συγχωνεύεται με προηγούμενα επίπεδα. Με αυτό τον τρόπο, δεν χρειάζεται να υπολογιστεί ένα συγκεκριμένο μέγεθος εισόδου για να ταιριάζει στο μοντέλο, αλλά μπορεί να πάρει ένα αυθαίρετο μέγεθος (Zhang, 2017). Επίσης σε αντίθεση με το αρχικό μοντέλο (Ronneberger et al., 2015), το μοντέλο που χρησιμοποιείται υποστηρίζει την χρήση έγχρωμων εικόνων (Zhang, 2017).

## 3.2 Fully Convolutional DenseNet

Κορυφαίες προσεγγίσεις για τον κατακερματισμό εικόνας στηρίζονται στα Συνελικτικά Νευρωνικά Δίκτυα (Convolutional Neural Networks - CNNs). Στη βιβλιογραφία πλέον έχουν εισαχθεί ως φυσική επέκταση των CNNs τα Πλήρως Συνελικτικά Δίκτυα (Fully Convolutional Networks - FCNs) για την αντιμετώπιση προβλημάτων πρόβλεψης ανά pixel όπως στην κατάτμηση εικόνας.

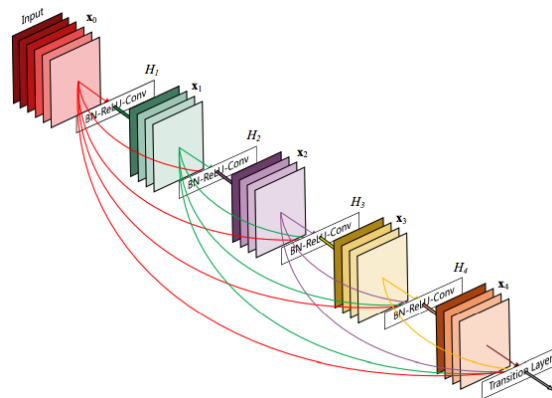
Τα FCNs προσθέτουν επίπεδα υπερδειγματοληψίας στο πρότυπο CNNs ώστε να ανακτήσουν τη χωρική ανάλυση της εισόδου στο επίπεδο εξόδου. Κατά συνέπεια, τα FCN μπορούν να επεξεργάζονται εικόνες αυθαίρετου μεγέθους. Προκειμένου να αντισταθμιστεί η απώλεια ανάλυσης που προκαλείται από τα επίπεδα συγκέντρωσης (pooling layers), τα FCNs παραλείπουν τις συνδέσεις μεταξύ της διαδρομής της υποδειγματοληψίας και της διαδρομής της υπερδειγματοληψίας. Η παράκαμψη αυτή των συνδέσεων βοηθά τη διαδρομή της υπερδειγματοληψίας στην ανάκτηση πληροφοριών από τα επίπεδα υποδειγματοληψίας.

Πρόσφατα, μια νέα αρχιτεκτονική CNN, από το Facebook AI Research (FAIR) η DenseNet Connected Convolutional Networks (DenseNets), έχει αποδείξει εξαιρετικά αποτελέσματα σε εργασίες ταξινόμησης εικόνων. Η ιδέα του DenseNets βασίζεται



στην παρατήρηση ότι εάν κάθε επίπεδο συνδέεται άμεσα σε κάθε άλλο επίπεδο με τρόπο προώθησης προς τα εμπρός, το δίκτυο τότε θα είναι πιο ακριβές και πιο εύκολο στην εκπαίδευση. Πιο συγκεκριμένα, τα DenseNets είναι χτισμένα από πυκνά μπλοκ (dense blocks) και λειτουργίες συγκέντρωσης (pooling layers), όπου κάθε πυκνό μπλοκ είναι επαναληπτική σύνδεση των προηγούμενων χαρτών χαρακτηριστικών. Αυτή η αρχιτεκτονική μπορεί να θεωρηθεί ως επέκταση του ResNets (He et al., 2016), το οποίο εκτελεί επαναληπτική άθροιση προηγούμενων χαρτών χαρακτηριστικών. Αυτή η μικρή τροποποίηση ωστόσο έχει πολύ θετικές επιπτώσεις στο DenseNet:

1. αποτελεσματικότητα των παραμέτρων, τα DenseNets είναι πιο αποτελεσματικά στην χρήση παραμέτρων.
2. έμμεση βαθιά επίβλεψη, το DenseNet πραγματοποιεί βαθιά επίβλεψη χάρη στα μικρά μονοπάτια προς όλους τους χάρτες χαρακτηριστικών στην αρχιτεκτονική.
3. επαναχρησιμοποίηση χαρακτηριστικών, όλα τα επίπεδα μπορούν να έχουν εύκολη πρόσβαση στα προηγούμενα επίπεδα τους, καθιστώντας εύκολη την επαναχρησιμοποίηση της πληροφορίας από προηγούμενους υπολογισμένους χάρτες χαρακτηριστικών.



**Σχήμα 3.2:** Ένα πυκνό μπλοκ 5 στρώσεων με ρυθμό ανάπτυξης  $k = 4$ . Κάθε στρώμα λαμβάνει όλους τους προηγούμενους χάρτες χαρακτηριστικών ως είσοδο. (Huang et al., 2017).

Ειδικότερα για το DenseNet έστω  $x_\ell$  η έξοδος του  $\ell - 1$  επιπέδου. Σε ένα τυπικό CNN, το  $x_\ell$  υπολογίζεται με εφαρμογή ενός μη γραμμικού μετασχηματισμού  $H_\ell$  στην έξοδο του προηγούμενου επιπέδου  $x_{\ell-1}$ .

$$x_\ell = H_\ell(x_{\ell-1}), \quad (3.1)$$

όπου το  $H$  ορίζεται συνήθως ως συνέλιξη που ακολουθείται από μη γραμμικό ανορθωτή (ReLU) και συχνά απόσυρση (Dropout).

Για να διευκολυνθεί η εκπαίδευση πολύ βαθιών δικτύων, το ResNet εισάγει ένα υπολειπόμενο μπλοκ που συνοψίζει την ταυτότητα χαρτογράφησης της εισόδου στην έξοδο ενός επιπέδου. Η έξοδος του αποτελέσματος  $x_\ell$  γίνεται:

$$x_\ell = H_\ell(x_{\ell-1}) + x_{\ell-1}, \quad (3.2)$$

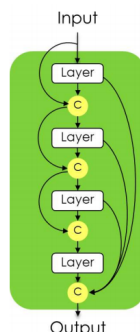
επιτρέποντας την επαναχρησιμοποίηση χαρακτηριστικών και επιτρέποντας την κλίση να ρέει απευθείας σε προηγούμενα επίπεδα. Σε αυτή την περίπτωση, το  $H$  ορίζεται ως η επανάληψη (2 ή 3 φορές) ενός μπλοκ που αποτελείται από ένα επίπεδο Κανονικοποίησης Παρτίδων (Batch Normalization - BN) ακολουθούμενο από την ReLU και μια συνέλιξη (Convolution).

Προωθώντας αυτή την ιδέα περαιτέρω, το DenseNet σχεδιάζει ένα ακόμα εξελιγμένο μοτίβο συνδεσιμότητας το οποίο αλληλοσυνδέεται επαναληπτικά με όλες τις εξόδους χαρακτηριστικών με ένα feedforward τρόπο. Έτσι η έξοδος του  $\ell - 1$  επιπέδου ορίζεται ως

$$x_\ell = H_\ell([x_{\ell-1}, x_{\ell-2}, \dots, x_0]), \quad (3.3)$$

όπου [...] αντιπροσωπεύει τη λειτουργία συγκόλλησης. Σε αυτή την περίπτωση, το  $H$  ορίζεται ως BN, ακολουθούμενο από το ReLU, μια συνέλιξη (convolution) και μια απόσυρση (dropout). Αυτό το μοτίβο σύδεσης ενθαρρύνει έντονα την επαναχρησιμοποίηση των χαρακτηριστικών και καθιστά όλα τα επίπεδα στην αρχιτεκτονική να λαμβάνουν σήμα άμεσης επίβλεψης. Η διάσταση εξόδου από κάθε επίπεδο  $\ell$  έχει  $k$  χάρτες χαρακτηριστικών όπου  $k$ , ο ρυθμός ανάπτυξης, συνήθως ρυθμίζεται σε μια μικρή τιμή (π.χ.,  $k = 2$ ). Έτσι, ο αριθμός των χαρτών χαρακτηριστικών στο DenseNet μεγαλώνει γραμμικά με το βάθος (π.χ. μετά από  $\ell$  επίπεδα, η είσοδος  $[x_{\ell-1}, x_{\ell-2}, \dots, x_0]$  θα έχει  $\ell \times k$  χάρτες χαρακτηριστικών).

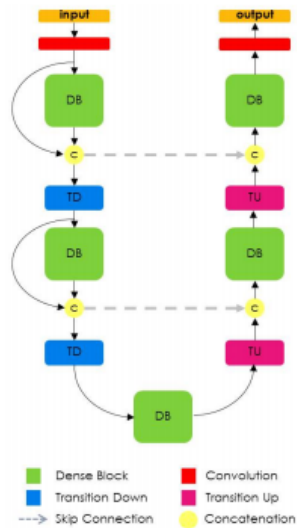
Στο υπόλοιπο κείμενο θα ονοματίζεται πυκνό μπλοκ (dense block) η συνένωση των νέων χαρτών χαρακτηριστικών που δημιουργήθηκαν σε μία δεδομένη ανάλυση.



**Σχήμα 3.3:** Διάγραμμα πυκνού μπλοκ από 4 στρώματα. Χρησιμοποιείται ένα πρώτο στρώμα στην είσοδο για να δημιουργήσετε χάρτες χαρακτηριστικών  $k$ , οι οποίοι είναι συναρμολογημένοι στην είσοδο. Στη συνέχεια εφαρμόζεται ένα δεύτερο στρώμα για να δημιουργηθούν κι άλλοι  $k$  χάρτες χαρακτηριστικών, οι οποίοι συνδέονται εκ νέου με τους προηγούμενους χάρτες χαρακτηριστικών. Η λειτουργία επαναλαμβάνεται 4 φορές. Η έξοδος του μπλοκ είναι η συγκόλληση των εξόδων των 4 επιπέδων και έτσι περιέχει  $4 * k$  χάρτες χαρακτηριστικών. (Jégou et al., 2017).

Όπως έχει περιγραφεί και παραπάνω η αρχιτεκτονική DenseNet αποτελεί τη διαδρομή υποδειγματοληψίας του Πλήρως Συνελικτικού DenseNet (FC-DenseNet). Αξίζει να σημειωθεί ότι στο μονοπάτι της υποδειγματοληψίας η γραμμική αύξηση του αριθμού των χαρακτηριστικών αντισταθμίζεται από τη μείωση της χωρικής ανάλυσης του κάθε χάρτη χαρακτηριστικών μετά τη λειτουργία συγκέντρωσης. Το τελευταίο επίπεδο της διαδρομής υποδειγματοληψίας αναφέρεται ως συμφόρηση (bottleneck). Για να ανακτηθεί η χωρική ανάλυση εισόδου, τα

FCN εισάγουν μια διαδρομή υπερδειγματοληψίας, που αποτελείται από συνέλιξη (convolution), υπερδειγματοληψία (upsampling) (μετασχηματισμένες συνέλιξεις (transposed convolutions) ή λειτουργίες μη συγκέντρωσης (unpooling operations) και συνδέσεις παράκαμψης (skip connections). Στα FC-DenseNets, η λειτουργία της συνέλιξης αντικαθίσταται με ένα πυκνό μπλοκ και μια διαδικασία υπερδειγματοληψίας που αναφέρεται ως μετάβαση προς τα πάνω (transition up). Η μετάβαση προς τα πάνω (transition up) αποτελείται από μια μετασχηματισμένη συνέλιξη (transposed convolution) η οποία υπερδειγματοληπτεί τους προηγούμενους χάρτες χαρακτηριστικών. Οι χάρτες χαρακτηριστικών που έχουν υπερδειγματοληφτηθεί συνδέονται στη συνέχεια με εκείνους που προέρχονται από τη σύνδεση παράκαμψης για να σχηματίσει την είσοδο ενός νέου πυκνού μπλοκ. Δεδομένου ότι η διαδρομή υπερδειγματοληψίας αυξάνει τη χωρική ανάλυση των χαρτών χαρακτηριστικών, η γραμμική αύξηση του αριθμού των χαρακτηριστικών θα απαιτούσε πολύ μνήμη, ειδικά για τα χαρακτηριστικά πλήρους ανάλυσης στο προ-softmax επίπεδο. Προκειμένου να ξεπεραστεί αυτός ο περιορισμός, η είσοδος ενός πυκνού μπλοκ δεν συνδέεται με την έξοδο του. Έτσι, η μετασχηματισμένη συνέλιξη (transposed convolution) εφαρμόζεται μόνο στους χάρτες χαρακτηριστικών που λαμβάνονται από το τελευταίο πυκνό μπλοκ και όχι απ'όλους τους μέχρι τώρα συνδεδεμένους χάρτες χαρακτηριστικών. Το τελευταίο πυκνό μπλοκ συνοψίζει τις πληροφορίες που περιέχονται σε όλα τα προηγούμενα πυκνά μπλοκ στην ίδια ανάλυση. Σημαντικό να σημειωθεί ότι ορισμένη πληροφορία από τα προηγούμενα πυκνά μπλοκ χάνονται κατά τη μετάβαση προς τα κάτω (transition down) εξαιτίας της λειτουργίας συγκέντρωσης (pooling operation). Παρ'όλα αυτά, αυτή η πληροφορία είναι διαθέσιμη στη διαδρομή υποδειγματοληψίας του δικτύου και μπορεί να περάσει μέσω των συνδέσεων παράκαμψης (skip connections). Ως εκ τούτου, τα πυκνά μπλοκ της διαδρομής υπερδειγματοληψίας υπολογίζονται χρησιμοποιώντας το σύνολο όλων των διαθέσιμων χαρτών χαρακτηριστικών σε μια δεδομένη ανάλυση (Jégou et al., 2017).



**Σχήμα 3.4:** Διάγραμμα της αρχιτεκτονικής μας για σημασιολογική κατάτμηση. Η αρχιτεκτονική μας είναι χτισμένη από πυκνά τετράγωνα. Το διάγραμμα αποτελείται μιας διαδρομής υποδειγματοληψίας με 2 μεταβάσεις προς τα κάτω (*Transitions Down* - *TD*) και μια διαδρομή υπερδειγματοληψίας με 2 μεταβάσεις προς τα πάνω (*Transitions Up* - *TU*). Ο κύκλος αντιπροσωπεύει τη σύζευξη και τα βέλη αντιπροσωπεύουν τα πρότυπα συνδεσιμότητας στο δίκτυο. Τα γκρι οριζόντια βέλη αντιπροσωπεύουν τις συνδέσεις παράκαμψης, οι χάρτες χαρακτηριστικών από το μονοπάτι υποδειγματοληψίας συνενώνονται με τους αντίστοιχους χάρτες χαρακτηριστικών στο μονοπάτι υπερδειγματοληψίας. Σημείωση ότι το μοτίβο συνδεσιμότητας στη διαδρομή αναδειγματοληψίας και στην υποδειγματοληψίας είναι διαφορετικά. Στη διαδρομή υποδειγματοληψίας, η είσοδος στο πυκνό μπλοκ συνενώνεται με την έξοδο του, οδηγώντας σε μια γραμμική αύξηση του αριθμού των χαρτών χαρακτηριστικών, ενώ στη διαδρομή υπερδειγματοληψίας δεν ισχύει αυτό. (Jégou et al., 2017).

# Κεφάλαιο 4

## Πειραματική Διαδικασία

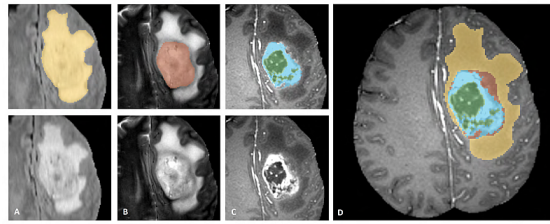
### 4.1 Σύνολο Δεδομένων

Στην εργασία αυτή μελετούνται εφαρμογές κατάτμησης καρκινικών όγκων από μαγνητικό τομογράφο. Μία από τις κύριες προκλήσεις που αντιμετωπίστηκαν ήταν η εξεύρεση αξιόπιστης βάσης δεδομένων που να ήταν πλήρως ετικετοποιημένη. Οι ιατρικές εικόνες είναι δύσκολο να βρεθούν λόγω προβλημάτων ιδιωτικότητας. Οι εικόνες που χρησιμοποιήθηκαν για την εκπαίδευση και την επικύρωση των μοντέλων της διπλωματικής εργασίας προέρχονται από τη βάση δεδομένων του BRATS2017 - Multimodal Brain Tumor Segmentation Challenge 2017. Το σύνολο δεδομένων αποτελείται από πολυτροπικές εικόνες MRI 285 ασθενών, με 210 ασθενείς από την κατηγορία υψηλού βαθμού γλοιομάτων (HGG) και 75 ασθενείς από την κατηγορία χαμηλού βαθμού γλοιομάτων (LGG). Οι ακόλουθοι τρόποι απεικόνισης MRI παρέχονται για κάθε ασθενή: T2-σταθμισμένο εξασθετισμένο υγρό (FLAIR), T1-σταθμισμένο (T1), T1-σταθμισμένο αντεστραμμένο (T1ce) και T2-σταθμισμένο(T2).

Οι παρεχόμενες μη αυτόματες κατατμήσεις περιλαμβάνουν τέσσερις ετικέτες: 1 για το νεκρωτικό και το μη-ενισχυτικό όγκο NCR και NET, 2 για το οίδημα ED, 4 για ενίσχυση όγκου ET και 0 για οτιδήποτε άλλο, δηλαδή φυσιολογικό ιστό και υπόβαθρο (background - black padding). Η επίσημη αξιολόγηση υπολογίζεται με τη συγχώνευση των προβλεπόμενων ετικετών σε τρεις ομάδες: τον ολικό όγκο (1,2,4), τον πυρήνα του όγκου (1,4) και την ενίσχυση του όγκου (4).

Οι περιοχές λοιπόν που εξετάζονται στην παρούσα διπλωματική εργασία είναι(BRATS, 2017):

1. Ο *Ενισχυτικός Όγκος (Enhancing Tumor-ET)*, ο οποίος περιγράφεται από περιοχές που παρουσιάζουν υπερ-ένταση στο T1Gd σε σύγκριση με την T1, αλλά και σε σύγκριση με την "υγιή" λευκή ύλη στο T1Gd.
2. Ο *Πυρήνας του Όγκου (Tumor Core-TC)*, ο οποίος περιγράφει το κύριο μέρος του όγκου, το οποίο συνήθως εκτοπίζεται. Ο πυρήνας του όγκου - TC συνεπάγεται το ET, καθώς και τα νεκρωτικά (γεμάτα με υγρό) και τα μη ενισχυτικά (στερεά) μέρη του όγκου. Η εμφάνιση του νεκρωτικού NCR και του μη ενισχυτικού πυρήνα του όγκου NET είναι τυπικά υπο-έντονη στην T1Gd όταν συγκρίνεται με την T1.
3. Ο *Ολόκληρος Όγκος (Whole Tumor-WT)* περιγράφει την πλήρη έκταση της νόσου, καθώς συνεπάγεται τον πυρήνα του όγκου - TC και το περιτομοειδές οίδημα ED, το οποίο απεικονίζεται με υπερ-έντονο σήμα στο FLAIR.



**Σχήμα 4.1:** Υποπεριοχές γλοίωμα. Εμφανίζονται ετικέτες με τις υποπεριοχές των όγκων που σχολιάζονται στις διαφορετικές λεπτομέρειες (πάνω αριστερά) και τις τελικές ετικέτες για ολόκληρο το σύνολο δεδομένων (δεξιά). Τα επιθέματα εικόνας εμφανίζονται από αριστερά προς τα δεξιά: ο ολικός όγκος (κίτρινος) ορατός στο T2-FLAIR (Σχήμα A), ο πυρήνας του όγκου (κόκκινος) ορατός στο T2 (Σχήμα B), οι ενισχυτικές δομές όγκου στο T1Gd, που περιβάλλει τα κυστικά / νεκρωτικά συστατικά του πυρήνα (πράσινο) (Σχήμα C). Οι κατατμήσεις συνδυάζονται για να δημιουργήσουν τις τελικές ετικέτες των υπο-περιοχών όγκου (Σχήμα D): οίδημα (κίτρινο), μη ενισχυτικός στερεός πυρήνας (κόκκινος), νεκρωτικός / κυστικός πυρήνας (πράσινος), ενισχυτικός πυρήνας (μπλε). (Menze et al., 2015).

#### 4.1.1 Προεπεξεργασία Δεδομένων

Το σύνολο των δεδομένων είναι οργανωμένο σε φακέλους για το κάθε ασθενή της κάθε κατηγορίας HGG ή LGG όπως αναφέρθηκε και παραπάνω. Συγκεκριμένα για ένα τυχαίο ασθενή της κατηγορίας υψηλού βαθμού γλοιομάτων (HGG) το όνομα του φακέλου είναι Brats17\_2013\_2\_1 και περιέχει μέσα 5 αρχεία .nii.gz τα οποία αντιπροσωπεύουν τα T1, T1ce, T2 και flair καθώς επίσης και το label, τη μη αυτόματα ετικέτα δηλαδή που έχει δοθεί σ'αυτό το MRI, έτσι για παράδειγμα για το flair είναι Brats17\_2013\_2\_1\_flair.nii.gz. Συνολικά υπάρχουν 1425 αρχεία με την κατάληξη .nii.gz.

Κατά την προεπεξεργασία των δεδομένων που έχουν ληφθεί χρησιμοποιείται ένας αλγόριθμος ο οποίος μετανομάζει το σύνολο των δεδομένων σε μια έξοδο, κάθε μία εικόνα των ασθενών περικόπτεται και ονομάζεται με βάση αυτή τη φόρμα: "ΤύποςΔείκτης\_τρόποςαπεικόνισης.nii.gz". Κάθε αρχείο .nii.gz περιέχει ολόκληρη την ακολουθία.

Για παράδειγμα για το φάκελο τον ασθενή 100 στην κατηγορία υψηλού βαθμού γλοιομάτων (HGG), ισχύει: HGG100\_Flair.nii.gz, HGG100\_Label.nii.gz, HGG100\_T1ce.nii.gz, HGG100\_T1.nii.gz, HGG100\_T2.nii.gz.

Κάθε τέτοιο αρχείο, που έχει την κατάληξη .nii.gz στην έξοδο περικόπτεται με ένα σφιχτό πλαίσιο ώστε να μην υπάρχει πολύ πληροφορία από το υπόβαθρο (background).

Στη συνέχεια μετατρέπονται τα αρχεία με την κατάληξη .nii.gz σε .png εικόνες. Οι εικόνες που αντιπροσωπεύουν τα label αποθηκεύονται σε διαφορετικό φάκελο (39386 εικόνες) απ'ότι οι υπόλοιπες png εικόνες (157544 εικόνες). Οπότε για παράδειγμα μία εικόνα LGG9\_Label.nii.gz που αντιπροσωπεύει ετικέτα είναι αποθηκευμένη τώρα ως .png με αυτή τη μορφή: LGG9\_Label\_0.png, LGG9\_Label\_1.png, LGG9\_Label\_2.png ...

Μετά από αυτή την μετατροπή οι εικόνες-ετικέτες μετανομάζονται ώστε να έχουν τα ίδια ονόματα με τα αρχεία των υπόλοιπων εικόνων. Οπότε για παράδειγμα η εικόνα HGG81\_Label\_106.png αντιγράφεται και μετανομάζεται 4 φορές, όσες δηλαδή είναι και οι τρόποι απεικόνισης MRI: ['Flair', 'T1', 'T1ce', 'T2'], δηλαδή γίνεται HGG81\_Flair\_106.png, HGG81\_T1\_106.png, HGG81\_T1ce\_106.png

και HGG81\_T2\_106.png.

Έπειτα τα δεδομένα από το φάκελο των εικόνων χρησιμοποιούνται ώστε να δημιουργηθεί ένα αρχείο txt που θα έχει τα ονόματα από τα δεδομένα αυτά, το αρχείο αυτό ονομάζεται images.txt και έχει όπως είναι λογικό 157544 ονόματα-γραμμές. Από το images.txt αρχείο δημιουργούνται και τα άλλα δύο txt αρχεία που είναι απαραίτητα για την εργασία, τα train.txt και test.txt με 100000 και 57544 γραμμές αντίστοιχα.

Το επόμενο βήμα που ακολουθείται στη διαδικασία της προεπεξεργασίας των δεδομένων είναι ότι για κάθε μετανομασμένη εικόνα-ετικέτα υπολογίζονται οι μοναδικές τιμές της και εν τέλει υπολογίζονται συνολικά οι μοναδικές τιμές απ'όλες τις μετανομασμένες εικόνες-ετικέτες. Αυτό έχει ως σκοπό να επανατικετοποιηθούν όλες οι εικόνες έτσι ώστε το υπόβαθρο (background) να αντιστοιχεί στην τελευταία ετικέτα και στη συνέχεια να μπορεί να αγνοηθεί.

## 4.2 Εργαλεία

Για την υλοποίηση της παρούσας διπλωματικής εργασίας έχει χρησιμοποιηθεί το *KERAS*, μια διεπαφή προγραμματισμού εφαρμογών (Application Programming Interface-API) υψηλού επιπέδου νευρωνικών δικτύων, το οποίο είναι γραμμένο σε Python και είναι ικανό να τρέχει πάνω από το TensorFlow και άλλες βιβλιοθήκες. Το KERAS περιέχει πολλές υλοποιήσεις των δομικών στοιχείων που χρησιμοποιούνται συνήθως σε νευρωνικά δίκτυα, όπως επίπεδα (layers), σφάλματα (losses), λειτουργίες ενεργοποίησης (activation functions), βελτιστοποιητές (optimizers), μετρικές (metrics). (KERAS) Εκτός από το KERAS χρησιμοποιήθηκαν και συναρτήσεις από το tensorflow. Οι γραφικές παραστάσεις έγιναν με το matplotlib. Όλος ο κώδικας της διπλωματικής γράφτηκε σε Python3.

## 4.3 Εκπαίδευση

Στην συγκεκριμένη εργασία αντιμετωπίζεται το πρόβλημα της ανισορροπίας κλάσης όπως έχει ήδη αναφερθεί στο 2.1.4. Συνεπώς σ'αυτή την ενότητα θα περιγραφεί η εκπαίδευση του κάθε μοντέλου και ο στόχος που είχε η κάθε μία αναφορικά με το πρόβλημα της ανισορροπίας των κλάσεων.

Σ'όλες τις υλοποιήσεις των μοντέλων η εκπαίδευση γίνεται με μέγεθος παρτίδας (batch\_size) 4. Για την sgd ρυθμό μάθησης (learning rate) 0.0001, ορμή (momentum) 0.9. Για τον Adam και σύμφωνα με το Keras Documentation έχουν χρησιμοποιηθεί οι προεπιλεγμένες παράμετροι οι οποίες είναι:  $lr = 0.001$ ,  $beta\_1 = 0.9$ ,  $beta\_2 = 0.999$ ,  $epsilon = None$ ,  $decay = 0.0$ ,  $amsgrad = False$ .

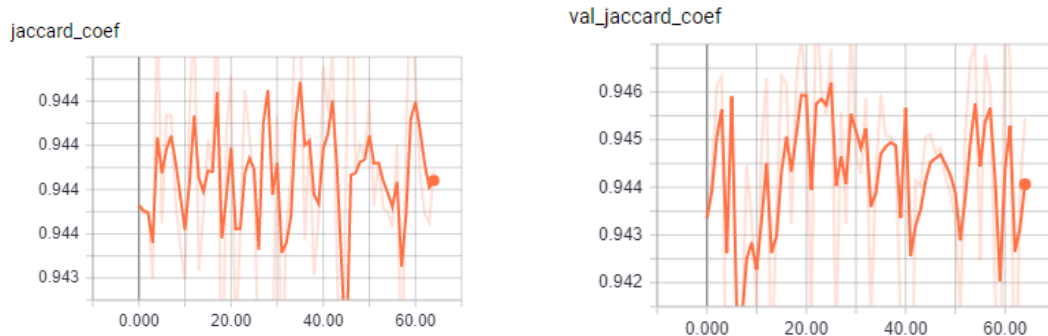
Κατά την εκπαίδευση υπήρχε επίβλεψη στο tensorboard για το πως το κάθε μοντέλο τα πηγαίνει στο validation set, οπότε όταν έπεφτε η τιμή του η εκπαίδευση σταματούσε.

### 4.3.1 Απλά Μοντέλα

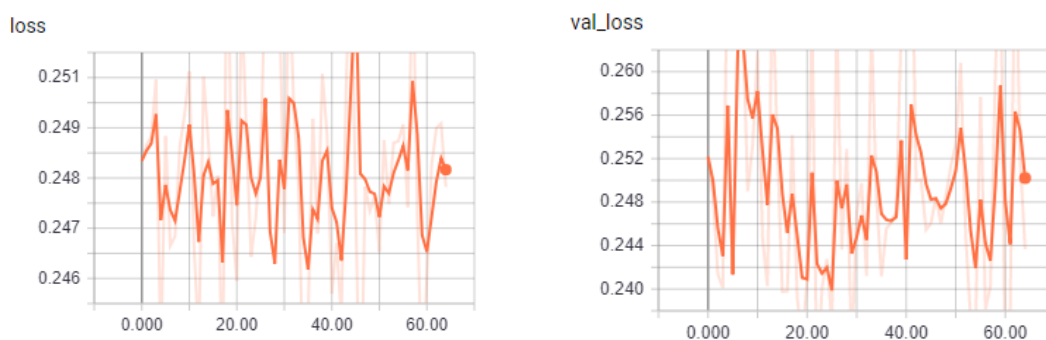
Αρχικά χρησιμοποιήθηκαν ως σημείο αναφοράς τα απλά μοντέλα που έχουν ήδη αναφερθεί στο κεφάλαιο 3 με τη μετρική "jaccard\_coeff" για το σύνολο εκπαίδευσης και το σύνολο επαλήθευσης. Στα διαγράμματα το loss είναι το binary\_crossentropy. Παρατηρείται από τα διαγράμματα ότι ενώ το densenet πιάνει μικρότερο loss (σχήμα

4.5, 0.0380 και val\_loss 0.0400 για densenet και σχήμα 4.3 0.248 για UNet) από το UNet, το jaccard του densenet τα πάει χειρότερα απ'ότι του το jaccard του UNet.

1. **UNet:** Η εκπαίδευση διήρκησε 6 ώρες 16 λεπτά και 33 δευτερόλεπτα.

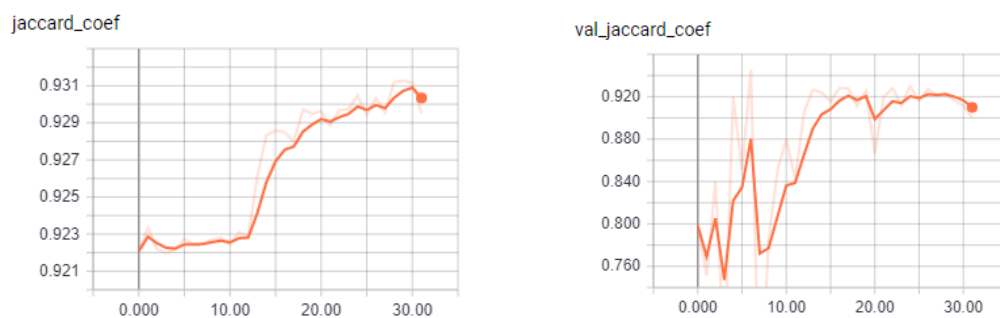


Σχήμα 4.2: Τα διαγράμματα της *jaccard\_coef* και της *validation jaccard\_coef* του μοντέλου UNet για 64 εποχές.



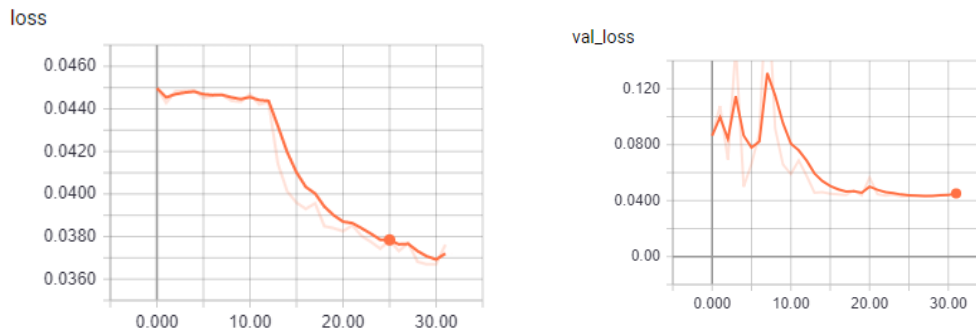
Σχήμα 4.3: Τα διαγράμματα του *loss* για το μοντέλο UNet για 64 εποχές αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.

2. **FC-DenseNet** Η εκπαίδευση διήρκησε 1 ημέρα, 1 ώρα, 55 λεπτά και 47 δευτερόλεπτα.



Σχήμα 4.4: Τα διαγράμματα της *jaccard\_coef* για το μοντέλο FC-DenseNet για 31 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.



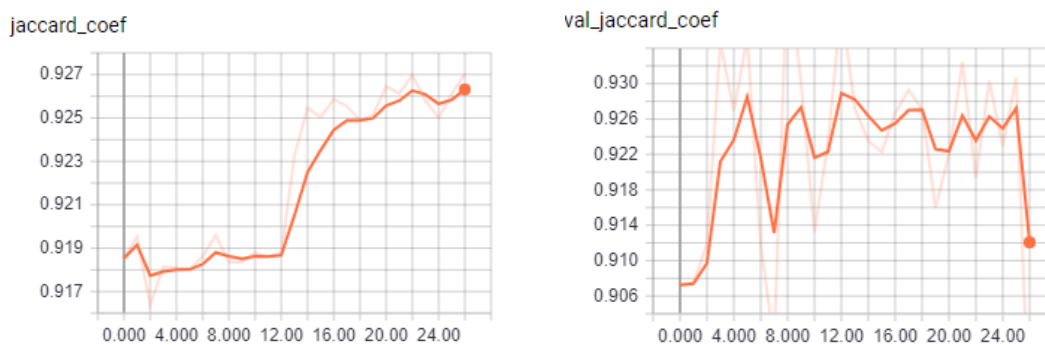


**Σχήμα 4.5:** Τα διαγράμματα του *loss* που αντιστοιχεί στο μοντέλο *FC-DenseNet* για 31 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.

### 4.3.2 Εξισορρόπηση δεδομένων με βάρη κλάσεων

Ωστόσο επειδή στις εικόνες προβλέπεται μόνο background ήταν αναγκαία μια ακόμα καλύτερη μετρική. Με τη μέθοδο αυτή εισάγονται διαφορετικά βάρη σε διαφορετικές κλάσεις, χρησιμοποιώντας την εντολή `sklearn.utils.class_weight.compute_class_weight(class_weight, classes, y)` (Pedregosa et al., 2011).

1. **UNet:** Η εκπαίδευση διήρκησε 1 ημέρα, 40 λεπτά και 43 δευτερόλεπτα και το μοντέλο έτρεξε για 245 εποχές.
2. **FC-DenseNet:** Η εκπαίδευση διήρκησε 22 ώρες 7 λεπτά και 29 δευτερόλεπτα.



**Σχήμα 4.6:** Τα διαγράμματα της *jaccard\_coef* και της *val\_jaccard\_coef* του μοντέλου *FC-DenseNet* για 26 εποχές.

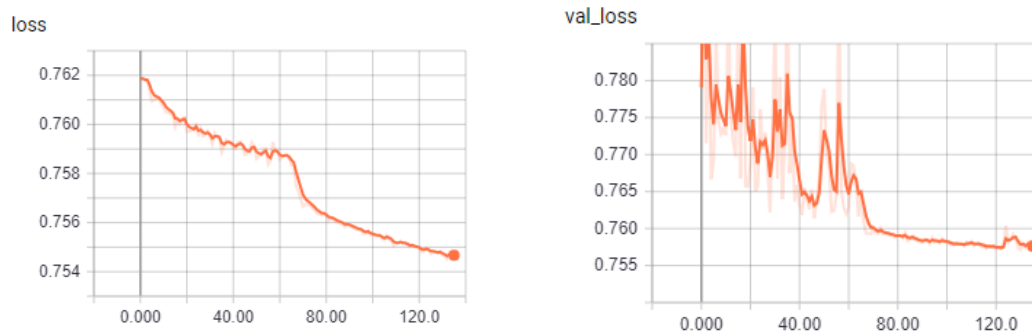


**Σχήμα 4.7:** Τα διαγράμματα του *loss* που αντιστοιχούν στο μοντέλο *FC-DenseNet* για 26 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.

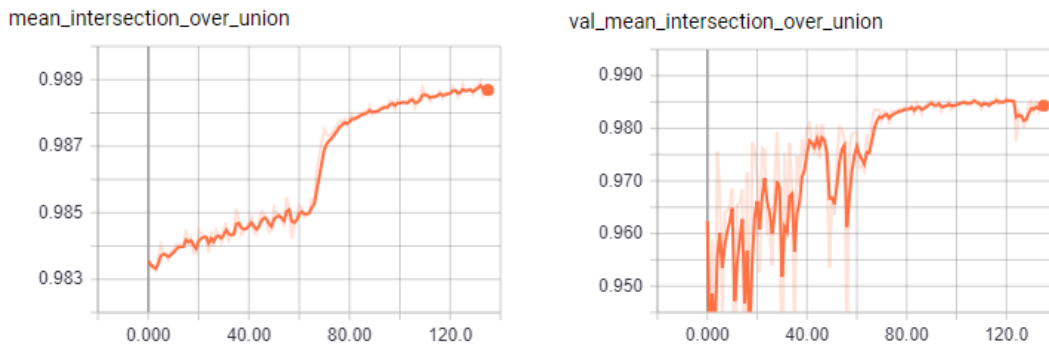
### 4.3.3 Εκπαίδευση Αγνοώντας την Τελευταία Ετικέτα

Παρά τη χρήση βαρών κλάσεων, τα μοντέλα εξακολουθούσαν να προβλέπουν μόνο background, γι'αυτό το λόγο χρησιμοποιήθηκαν μοντέλα πιο σύνθετα. Στα μοντέλα αυτά η εκπαίδευση γίνεται αγνοώντας τη τελευταία ετικέτα (`ignoring_last_label`) και χρησιμοποιώντας τη μετρική `"sparse_accuracy_ignoring_last_label"`.

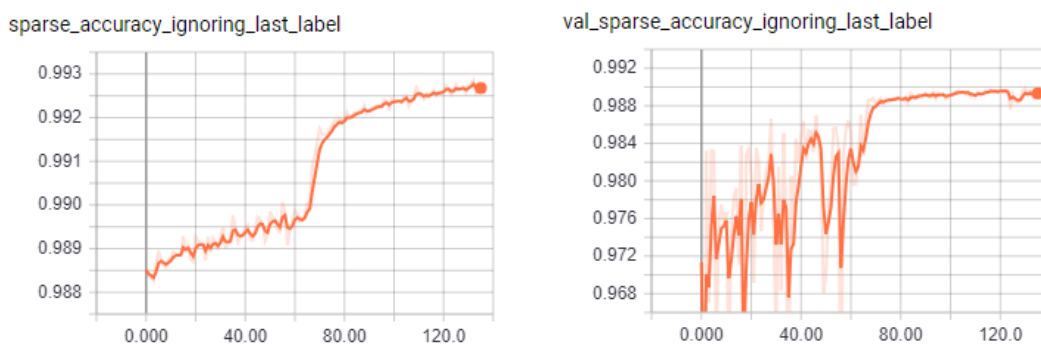
1. **UNet:** Η εκπαίδευση διήρκησε 20 ώρες, 22 λεπτά και 29 δευτερόλεπτα. Το μοντέλο εκπαιδεύθηκε για 999 εποχές.
2. **FC-DenseNet:** Η εκπαίδευση διήρκησε 4 ημέρες, 17 ώρες, 35 λεπτά και 45 δευτερόλεπτα.



**Σχήμα 4.8:** Τα διαγράμματα του *loss* που αντιστοιχούν στο μοντέλο *FC-DenseNet* για 135 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.



**Σχήμα 4.9:** Τα διαγράμματα της *mean\_intersection\_over\_union* του μοντέλου *FC-DenseNet* για 135 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.



**Σχήμα 4.10:** Τα διαγράμματα της *sparse\_accuracy\_ignoring\_last\_label* του μοντέλου *FC-DenseNet* για 135 εποχές, αριστερά για το σύνολο εκπαίδευσης και δεξιά για το σύνολο επαλήθευσης.

#### 4.3.4 Μοντέλα cw με αρχικοποίηση από ignoring

Ακόμη μία δοκιμή ήταν να χρησιμοποιηθεί το μοντέλο που δίνει διαφορετικά βάρη σε διαφορετικές κλάσεις έχοντας όμως “αρχικοποιηθεί” με τα βάρη από την εκπαίδευση αγνοώντας τη τελευταία ετικέτα του υπόβαθρου.

1. **FC-DenseNet** Η εκπαίδευση διήρκησε 26 ημέρες, 20 ώρες, 21 λεπτά και 54 δευτερόλεπτα και έτρεξε για 663 εποχές.

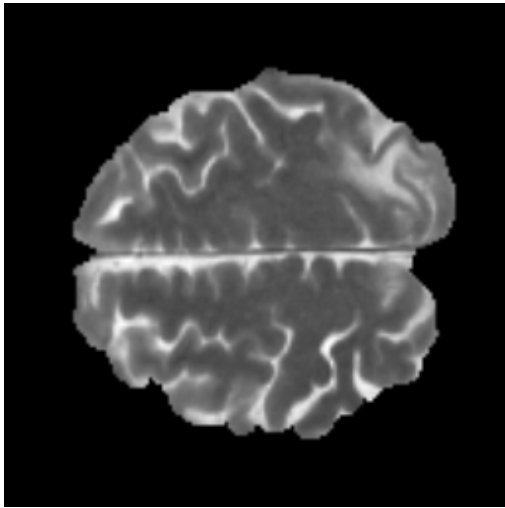
## 4.4 Αποτελέσματα

Για την αξιολόγηση των διάφορων τεχνικών χρησιμοποιήθηκαν, όπως αναφέρθηκε και στην ενότητα 4.1, 285 ασθενείς, με 210 ασθενείς από την κατηγορία υψηλού βαθμού γλοιωμάτων και 75 ασθενείς από την κατηγορία χαμηλού βαθμού γλοιωμάτων. Επιπλέον για όλους τους ασθενείς υπάρχουν συνολικά 39386 εικόνες που αντιπροσωπεύουν τις ετικέτες, (label), και 157544 εικόνες που είναι προς επεξεργασία.

Η εργασία αυτή είχε ως σκοπό την κατάτμηση εγκεφαλικών όγκων από εικόνες που προέρχονται από μαγνητικό τομογράφο. Τα αποτελέσματα της είναι αρκετά ικανοποιητικά, καθώς όπως παρατηρείται και στις εικόνες 4.11, 4.12 και 4.13 η πρόβλεψη του όγκου είναι αρκετά επιτυχημένη και συμβατή με την αντίστοιχη ετικέτα.

Επιπλέον, πρέπει να σημειωθεί ότι η πρόβλεψη των νευρωνικών γίνεται με 2 διαφορετικούς τρόπους επεξεργασίας. Ο πρώτος τρόπος είναι να επιλέγεται η πρόβλεψη στην οποία το νευρωνικό έδινε το μέγιστο βάρος ( $\max$ ). Ο δεύτερος τρόπος είναι να επιλέγεται η πρόβλεψη την οποία το νευρωνικό δίνει με ασφάλεια ώστε να μη ξεπερνάει ένα συγκεκριμένο κατώφλι, στην συγκεκριμένη εργασία είναι το  $0.5/1$ , ( $\text{threshold}$ ).

Για την εικόνα της ετικέτας 4.11 παρατηρείται ότι εμφανίζεται μόνο η περιοχή του ολόκληρου όγκου και όχι ο ενισχυτικός όγκος ή ο πυρήνας του όγκου. Όπως είναι φανερό και από τις παραπάνω εικόνες, για το μοντέλο DenseNet 4.12 η πρόβλεψη είναι πιο ακριβής για τη θέση του όγκου στον εγκέφαλο. Παρατηρούνται κάποιες διαφορές προφανώς ανάμεσα στους δύο τρόπους πρόβλεψης του μοντέλου αυτού, όμως χωρίς να υπάρχει πολύ μεγάλη απόκλιση. Όσον αφορά το μοντέλο UNet, 4.13 ο όγκος εντοπίζεται, αλλά στην πρόβλεψη εμφανίζεται και μια δεύτερη περιοχή που δεν αντιστοιχεί στο γλοίωμα.



**Σχήμα 4.11:** Αριστερά πραγματική μαγνητική τομογραφία ενός εγκεφάλου και δεξιά η ετικέτα η οποία υποδεικνύει τη θέση του όγκου.

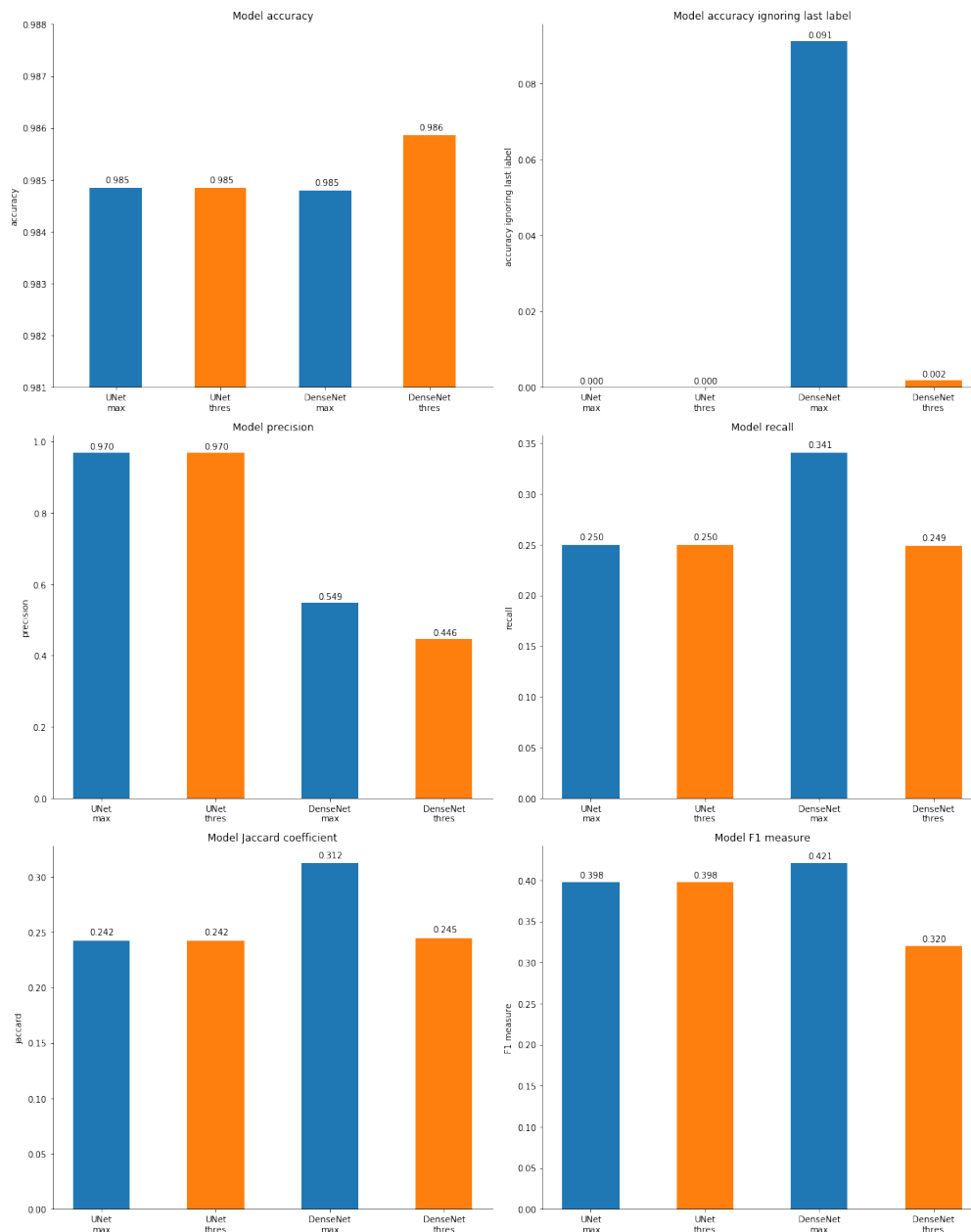


**Σχήμα 4.12:** Αριστερά η πρόβλεψη της θέσης του όγκου για το μοντέλο DenseNet με τον τρόπο πρόβλεψης  $\max$  και δεξιά με τον τρόπο πρόβλεψης  $\text{thres}$ .



**Σχήμα 4.13:** *Η πρόβλεψη της θέσης του όγκου για το μοντέλο UNet.*

## 4.4.1 Απλά μοντέλα



Σχήμα 4.14: Τα διαγράμματα που αναπαριστούν τα απλά μοντέλα UNet και DenseNet για όλες τις μετρικές που έχουν χρησιμοποιηθεί.

Παρατηρείται ότι η μετρική **accuracy** είναι λιγότερο αξιόπιστη, καθώς τα απλά μοντέλα και του UNet και του DenseNet πετυχαίνουν μόνο background, συνεπώς η μετρική είναι υψηλή γιατί προβλέπεται σωστά η τελευταία ετικέτα, χωρίς αυτό να σημαίνει όμως ότι εντοπίζει τον όγκο. Αυτό φαίνεται και από το σχήμα 4.14, όπου τα μοντέλα UNet\_max, UNet\_thres και DenseNet\_max έχουν ακριβώς την ίδια τιμή, 0.985 και το DenseNet\_thres έχει 0.986.

Για τη μετρική **recall** γίνεται πιο φανερό τι συμβαίνει με τα απλά μοντέλα. Από τα διαγράμματα φαίνεται ότι για τα UNet οι προβλέψεις είναι οι ίδιες μεταξύ τους, 0.250, γιατί προβλέπουν μόνο μια από τις 4 κλάσεις. Από την άλλη για τα DenseNet υπάρχει διαφορά ανάμεσα στα επιμέρους μοντέλα, για το DenseNet\_max η μετρική έχει τιμή

0.341, ενώ για το DenseNet\_thres έχει 0.249. Η μετρική αυτή δείχνει από το πόσα pixel ήταν όντως αυτά που έπρεπε να προβλεφθούν πόσα εν τέλει προβλέφθηκαν.

Για τη μετρική **precision** παρατηρούνται υψηλά ποσοστά, γιατί μ'αυτή τη μετρική αν αυτά που προβλέπονται πετυχαίνονται, τότε αυξάνεται. Με βάση το διάγραμμα 4.14 παρατηρείται ότι για τα απλά μοντέλα UNet και με τον τρόπο πρόβλεψης max και με τον τρόπο πρόβλεψης thres, έχουμε ίδια υψηλά ποσοστά, 0.970 γιατί προβλέπεται μόνο το background το οποίο και πετυχαίνεται, καθώς είναι πολύ έντονη η παρουσία του στις εικόνες. Από την άλλη για τα μοντέλα DenseNet\_max και DenseNet\_thres δεν προβλέπεται μόνο το background, επειδή κάνουν προβλέψεις και για τις άλλες κλάσεις και μερικές είναι αποτυχημένες, πετυχαίνουν μικρότερα ποσοστά 0.549 και 0.446.

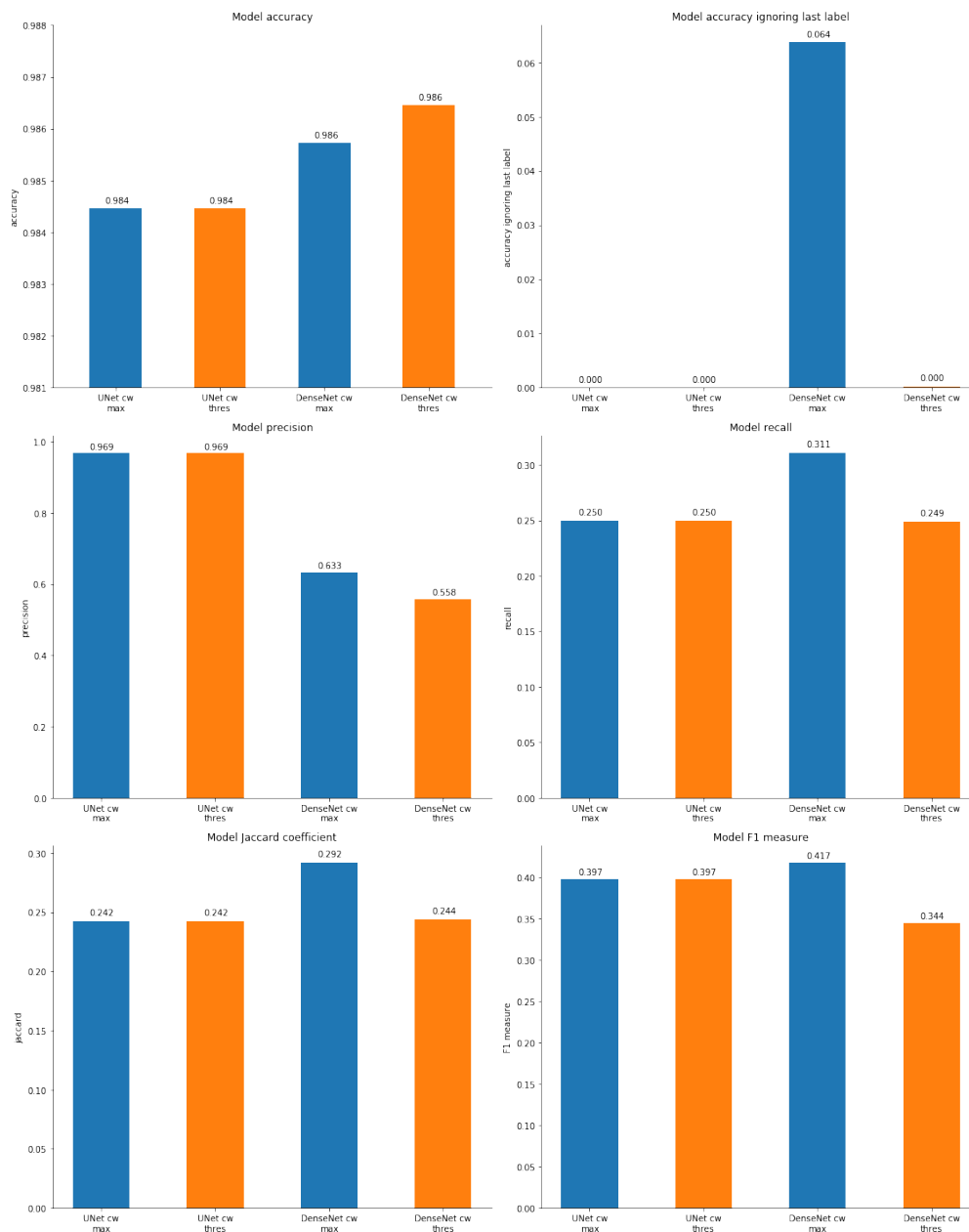
Η **jaccard** εκφράζει πληρότητα που βρίσκεται και στο precision και στο recall. Φαίνεται από τα διαγράμματα ότι τα απλά μοντέλα του UNet έχουν και τα δύο τιμή 0.242 καθώς υπολογίζεται μόνο το background. Για τα μοντέλα του densenet είναι λίγο καλύτερα τα πράγματα αφού το μοντέλο είναι πιο σύνθετο, οπότε δεν προβλέπει μόνο την τελευταία ετικέτα. Από το διάγραμμα 4.14, το μοντέλο DenseNet\_max έχει τιμή 0.312 και το μοντέλο DenseNet\_thres έχει τιμή 0.245.

Η μετρική **accuracy ignoring last label** είναι ιδιαίτερα χρήσιμη στην παρούσα διπλωματική, γιατί δείχνει την επίδοση του αλγορίθμου στις κλάσεις που αντιστοιχούν στον όγκο και δεν προβλέπει μόνο background. Αυτό φαίνεται από τα απλά μοντέλα του UNet τα οποία δε φαίνονται στους άξονες γιατί είναι πολύ χαμηλά τα ποσοστά τους, καθώς αυτό που προβλέπουν είναι το background. Τα απλά μοντέλα του densenet γι'αυτή τη μετρική είναι λίγο πιο πάνω από το 0, (DenseNet\_max: 0.091 και DenseNet\_thres: 0.002), καθώς το μοντέλο είναι πιο σύνθετο απ'ότι το UNet.

Για τη μετρική **F1** ισχύει όπως έχει ήδη αναφερθεί στην ενότητα 2.3.2, δηλαδή ότι είναι ο αρμονικός μέσος της ακρίβειας και της ανάκλησης. Τα απλά μοντέλα του UNet έχουν την ίδια τιμή καθώς προβλέπουν το background, 0.398, το οποίο δεν είναι επιθυμητό να προβλέπεται, γι'αυτό και έχουν τόσο χαμηλά ποσοστά. Τα απλά μοντέλα του densenet όπως έχει αναφερθεί και στις παραπάνω μετρικές καθότι είναι πιο σύνθετα από τα UNet το ποσοστό τους παρότι είναι πιο χαμηλό, είναι πιο αντικειμενικό και εδώ μεγαλύτερο, πιο συγκεκριμένα για το DenseNet\_max είναι 0.421 και για το DenseNet\_thres είναι 0.320.

Σύμφωνα με τις 6 αυτές μετρικές συμπεραίνουμε ότι το καλύτερο μοντέλο ήταν το **DenseNet max**.

## 4.4.2 Μοντέλα με χρήση cw



Σχήμα 4.15: Τα διαγράμματα που αναπαριστούν τα μοντέλα UNet και Densenet με χρήση cw για όλες τις μετρικές που έχουν χρησιμοποιηθεί .

Στα μοντέλα με χρήση βάρη κλάσεων ισχύει:

Για τη μετρική **accuracy** τα μοντέλα του UNet είναι αρκετά υψηλά, αλλά αυτό δεν είναι αξιόπιστο γιατί είναι μοντέλα τα οποία υπολογίζουν το background, όπως αναφέρθηκε και παραπάνω, οπότε η accuracy είναι υψηλή καθώς εντοπίζεται έντονο το background στις περισσότερες εικόνες. Για τα μοντέλα του densenet ισχύει το ίδιο. Σύμφωνα με το σχήμα 4.15, τα μοντέλα UNet\_cw\_max και UNet\_cw\_thres έχουν ίδια πρόβλεψη 0.984 ενώ τα μοντέλα densenet\_cw\_max και densenet\_cw\_thres έχουν πρόβλεψη 0.986.

Για τη μετρική **recall** ισχύει το ίδιο με τα απλά μοντέλα τόσο του UNet, όσο και του densenet αντίστοιχα για την ίδια μετρική. Για τα μοντέλα UNet\_cw\_max



και UNet\_cw\_thres η πρόβλεψη είναι 0.250 και για τα μοντέλα densenet\_cw\_max και densenet\_cw\_thres πρόβλεψη είναι 0.311 και 0.249.

Για τη μετρική **precision** παρατηρούνται υψηλά ποσοστά, γιατί μ'αυτή τη μετρική μετριέται από τα πόσα προβλέφθηκαν, πόσα είναι τα σωστά. Με βάση το διάγραμμα 4.15 παρατηρείται ότι για τα απλά μοντέλα UNet και με τον τρόπο πρόβλεψης max και με τον τρόπο πρόβλεψης thres, έχουμε ίδια υψηλά ποσοστά, 0.969 γιατί προβλέπεται μόνο το background το οποίο και πετυχαίνεται, καθώς είναι πολύ έντονη η παρουσία του στις εικόνες. Από την άλλη για τα μοντέλα DenseNet\_cw\_max και DenseNet\_cw\_thres πάλι προβλέπεται μόνο το background, με τη διαφορά όμως ότι επειδή είναι πιο ισχυρά μοντέλα απ'ότι τα UNet τα ποσοστά επιτυχίας του είναι μικρότερα, 0.633 και 0.558.

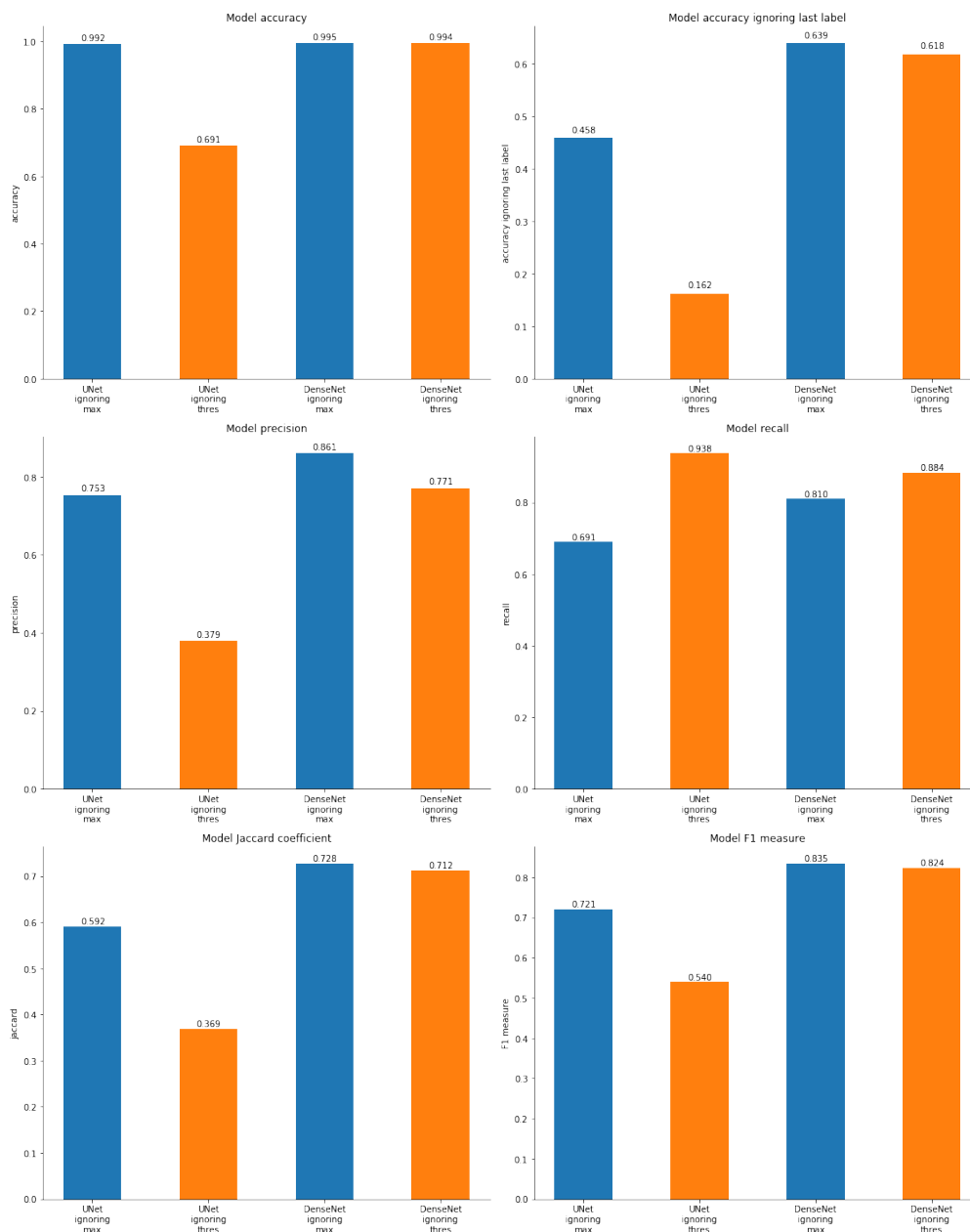
Για τη μετρική **jaccard** ισχύει ότι είναι η τομή / ένωση -i της πρόβλεψης με την ετικέτα. Σύμφωνα και με το σχήμα 2.17 είναι εύκολο κανείς να καταλάβει ότι τα μοντέλα τα οποία δεν προβλέπουν σωστά αυτή τη μετρική με βάση τον τύπο της 2.16 θα έχουν μικρότερη τιμή. Οπότε στα UNet\_cw\_max και UNet\_cw\_thres που προβλέπουν μόνο την κλάση του background η τιμή της μετρικής αυτής είναι χαμηλή και ίδια και στα δύο μοντέλα, 0.242, καθώς επίσης για τα μοντέλα DenseNet\_cw\_max και DenseNet\_cw\_thres που και πάλι προβλέπεται μόνο το background, η τιμή της μετρικής είναι 0.292 και 0.244 αντίστοιχα. Είναι πιο υψηλή η τιμή σ'αυτά τα μοντέλα γιατί είναι πιο ισχυρά απ'ότι τα UNet.

Η μετρική **accuracy\_ignoring\_last\_label** υπολογίζει την ακρίβεια αγνοώντας την τελευταία ετικέτα. Αυτό συνεπάγεται ότι και τα τέσσερα μοντέλα θα έχουν πολύ χαμηλές τιμές γιατί προβλέπουν μόνο το background καθώς αυτό κυριαρχεί στις περισσότερες εικόνες. Συγκεκριμένα, από το σχήμα 4.15 για τα μοντέλα UNet\_cw\_max και UNet\_cw\_thres η μετρική είναι 0, ενώ και για τα μοντέλα DenseNet\_cw\_max και DenseNet\_cw\_thres ισχύει ότι είναι 0.064 και 0 αντίστοιχα.

Η μετρική **F1** είναι όπως έχει αναφερθεί και παραπάνω ο αρμονικός μέσος της ακρίβειας (από τα πόσα προβλέφθηκαν πόσα είναι σωστά) και της ανάκλησης (από τα πόσα όντως ισχύουν πόσα εν τέλει προβλέφθηκαν). Οπότε με βάση και το σχήμα 4.15 οι μετρικές για τα δύο μοντέλα UNet είναι 0.397, γιατί προβλέπουν background, ενώ για τα DenseNet\_cw\_max και DenseNet\_cw\_thres είναι 0.417 και 0.344 αντίστοιχα.

Παρόλο που στο πρόβλημά μας υπάρχει έντονη ανισορροπία κλάσεων, η χρήση βαρών κλάσεων κατά την εκπαίδευση δεν έδειξε να βοηθάει ιδιαίτερα. Πάλι το καλύτερο μοντέλο είναι το **DenseNet max**.

### 4.4.3 Μοντέλα αγνοώντας την τελευταία κλάση



Σχήμα 4.16: Τα διαγράμματα που αναπαριστούν τα μοντέλα UNet και Densenet αγνοώντας την τελευταία ετικέτα για όλες τις μετρικές που έχουν χρησιμοποιηθεί.

Στα μοντέλα αυτά ισχύει για κάθε μετρική:

Η μετρική **accuracy** για τα μοντέλα UNet\_ignoring\_max και UNet\_ignoring\_thres είναι υψηλή γιατί τώρα τα μοντέλα αυτά αγνοούν την τελευταία ετικέτα, οπότε η ανίχνευση του όγκου είναι πιο ακριβής και η μετρική έχει μεγαλύτερες τιμές καθώς προβλέπονται σωστά οι άλλες κλάσεις. Τα μοντέλα αυτά έχουν accuracy 0.992 και 0.691 αντίστοιχα. Τα μοντέλα DenseNet\_ignoring\_max και DenseNet\_ignoring\_max έχουν μεγαλύτερες τιμές για τη μετρική αυτή καθώς είναι ισχυρότερα μοντέλα. Σύμφωνα με το σχήμα 4.16 οι τιμές τους είναι 0.995 και 0.994. Μ'αυτά τα μοντέλα που αποτρέπουν την πρόβλεψη του background η μετρική accuracy γίνεται πιο αξιόπιστη για τον σκοπό της διπλωματικής αυτής εργασίας.

Για τη μετρική **recall** παρατηρείται μια τεράστια βελτίωση στη μετρική αυτή σε σχέση με τις προηγούμενες. Αυτό συμβαίνει γιατί προβλέπονται όλες οι κλάσεις αξιόπιστα. Σύμφωνα με το διάγραμμα 4.16 παρατηρείται ότι για το μοντέλο UNet\_ignoring\_max η μετρική είναι 0.691 και για το UNet\_ignoring\_thres η μετρική είναι 0.938. Για τα μοντέλα densenet\_ignoring\_max και densenet\_ignoring\_thres πρόβλεψη είναι 0.810 και 0.884.

Για τη μετρική **precision** παρατηρούνται σχετικά υψηλά ποσοστά, γιατί μ'αυτή τη μετρική μετριέται από τα πόσα προβλέφθηκαν, πόσα είναι τα σωστά. Με βάση το διάγραμμα 4.16 παρατηρείται ότι το UNet\_ignoring\_max έχει τιμή της μετρικής στα 0.753 και το UNet\_ignoring\_thres έχει τιμή μετρικής στα 0.379. Από την άλλη για τα μοντέλα DenseNet\_ignoring\_max και DenseNet\_ignoring\_thres οι τιμές της μετρικής είναι 0.861 και 0.771 αντίστοιχα. Μπορεί να φαίνεται πεσμένο σε σχέση με κάποια από τα προηγούμενα μοντέλα, αυτό όμως συμβαίνει γιατί τα μοντέλα πια "τολμούν" να προβλέπουν κι άλλες κλάσεις πέρα από την τελευταία.

Σημειώνεται η εξής παρατήρηση: Το μοντέλο UNet\_ignoring\_thres έχει για τη μετρική **recall** τιμή 0.938 και για τη μετρική **precision** τιμή 0.379, αυτό συμβαίνει γιατί πιθανότατα το μοντέλο ignoring\_last\_label κατά την πρόβλεψη προβλέπει λάθος την τελευταία ετικέτα, παρόλο που την αγνοεί, αυτό συνεπάγεται την αύξηση της ανάκλησης - recall και της μείωσης της ακρίβειας - precision.

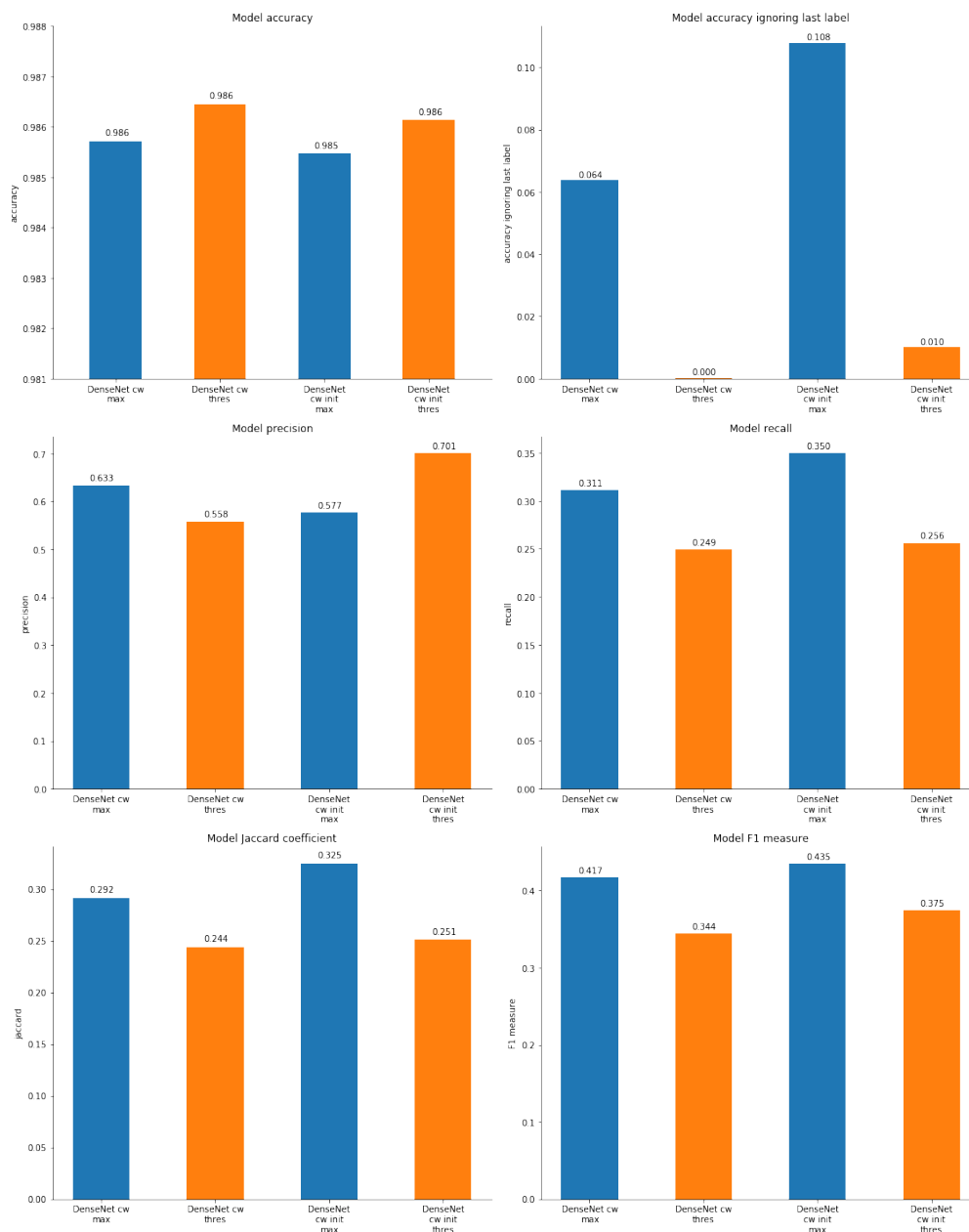
Για τη μετρική **jaccard** ισχύει ότι είναι το μέγεθος της τομής διαιρούμενο με το μέγεθος της ένωσης των συνόλων δειγμάτων. Σύμφωνα και με το σχήμα 2.17 είναι εύκολο κανείς να καταλάβει ότι τα μοντέλα τα οποία δεν προβλέπουν σωστά αυτή τη μετρική με βάση τον τύπο της 2.16 θα έχουν μικρότερη τιμή και αντίστοιχα τα μοντέλα που προβλέπουν σωστά θα έχουν μεγαλύτερη τιμή της jaccard. Οπότε παρατηρώντας το σχήμα 4.16 τα μοντέλα UNet\_ignoring\_max και UNet\_ignoring\_thres έχουν τιμές 0.592 και 0.369 αντίστοιχα, ενώ τα μοντέλα DenseNet\_ignoring\_max και DenseNet\_ignoring\_thres έχουν τιμές 0.728 και 0.712 αντίστοιχα.

Η μετρική **accuracy\_ignoring\_last\_label** υπολογίζει την ακρίβεια αγνοώντας την τελευταία ετικέτα. Οπότε για τα μοντέλα αυτά που έχουν εκπαιδευθεί αγνοώντας την τελευταία ετικέτα αυτή η μετρική είναι αρκετά αξιόπιστη. Από το σχήμα 4.16 γίνεται φανερό ότι το μοντέλο UNet\_ignoring\_thres που έχει τιμή αρκετά χαμηλή, 0.162 δεν προβλέπει με μεγάλη επιτυχία, κάτι το οποίο αναφέρθηκε και λίγο παραπάνω. Αντίθετα τα υπόλοιπα μοντέλα έχουν αρκετά ικανοποιητικές τιμές, το UNet\_ignoring\_max έχει 0.458 και τα αρκετά πιο ισχυρά μοντέλα DenseNet\_ignoring\_max με 0.639 και DenseNet\_ignoring\_thres με 0.618.

Η μετρική **F1** με δεδομένο ότι είναι ο αρμονικός μέσος της ακρίβειας και της ανάκλησης αναμένεται να βγάλει αρκετά υψηλά ποσοστά. Πράγματι από το σχήμα 4.16 για το μοντέλο UNet\_ignoring\_max η τιμή είναι 0.721, για το μοντέλο UNet\_ignoring\_thres η τιμή είναι 0.540 και για τα πιο ισχυρά μοντέλα DenseNet\_ignoring\_max και DenseNet\_ignoring\_thres οι τιμές είναι 0.835 και 0.824 αντίστοιχα.

Αγνοώντας την τελευταία ετικέτα κατά την εκπαίδευση, επέτρεψε στα μοντέλα να επικεντρωθούν στις κλάσεις που μας ενδιαφέρει περισσότερο και να σταματήσουν να προβλέπουν μόνο background. Πάλι το καλύτερο μοντέλο είναι το **DenseNet max**

#### 4.4.4 Μοντέλα cw με αρχικοποίηση από ignoring



Σχήμα 4.17: Τα διαγράμματα που αναπαριστούν τα μοντέλα UNet και Densenet με χρήση cw και τα οποία έχουν αρχικοποιηθεί αγνοώντας την τελευταία ετικέτα για όλες τις μετρικές που έχουν χρησιμοποιηθεί .

Συγκρίνονται τα μοντέλα που έχουν εκπαιδευθεί με βάρη κλάσεων μαζί με τα μοντέλα που έχουν εκπαιδευθεί με βάρη κλάσεων και έχουν αρχικοποιηθεί αγνοώντας την τελευταία ετικέτα, για κάθε μετρική:

Η μετρική **accuracy** όπως ήδη έχει αναφερθεί είναι υψηλή όταν προβλέπει σωστά αυτά που εντοπίζονται. Τα μοντέλα DenseNet\_cw\_max και DenseNet\_cw\_thres προβλέπουν το background καθώς αυτό κυριαρχεί στις εικόνες, οπότε γι' αυτό το λόγο και η μετρική σ' αυτά τα μοντέλα είναι μεγάλη. Οι τιμές της είναι και για τα δύο όπως έχει ήδη αναφερθεί 0.986. Για τα DenseNet\_cw\_init\_max και DenseNet\_cw\_init\_thres η μετρική είναι και εδώ υψηλή, με τη βασική όμως διαφορά ότι τώρα τα μοντέλα έχουν αρχικοποιηθεί να αγνοούν την τελευταία ετικέτα, οπότε και να είναι πιο σωστά για τον

σκοπό της διπλωματικής καθώς προβλέπονται σωστά οι άλλες κλάσεις. Τα μοντέλα αυτά έχουν με βάση το σχήμα 4.17 τιμές 0.985 και 0.986.

**Μετρική recall.** Για το πρώτο μοντέλο DenseNet\_cw\_max και DenseNet\_cw\_thres όπως έχει ήδη αναφερθεί το ποσοστό είναι χαμηλό καθώς προβλέπεται μόνο η τελευταία ετικέτα κάτι που δεν είναι επιθυμητό, άρα οι τιμές εδώ είναι 0.311 και 0.249. Για τα μοντέλα DenseNet\_cw\_init\_max και DenseNet\_cw\_init\_thres η τιμή της μετρικής είναι πιο υψηλή, γιατί τώρα τα μοντέλα είναι λίγο καλύτερα καθώς έχουν αρχικοποιηθεί να αγνοούν την τελευταία ετικέτα, άρα έχουν τιμές 0.350 και 0.256.

**Μετρική precision.** Σύμφωνα με το σχήμα 4.17 παρατηρείται ότι για τα μοντέλα DenseNet\_cw\_max και DenseNet\_cw\_thres που προβλέπουν μόνο το background οι τιμές της μετρικής είναι 0.633 και 0.558 αντίστοιχα. Για τα DenseNet\_cw\_init\_max και DenseNet\_cw\_init\_thres η τιμή της μετρικής είναι πιο υψηλή, γιατί τώρα τα μοντέλα είναι λίγο καλύτερα καθώς έχουν αρχικοποιηθεί να αγνοούν την τελευταία ετικέτα, άρα έχουν τιμές 0.577 και 0.701.

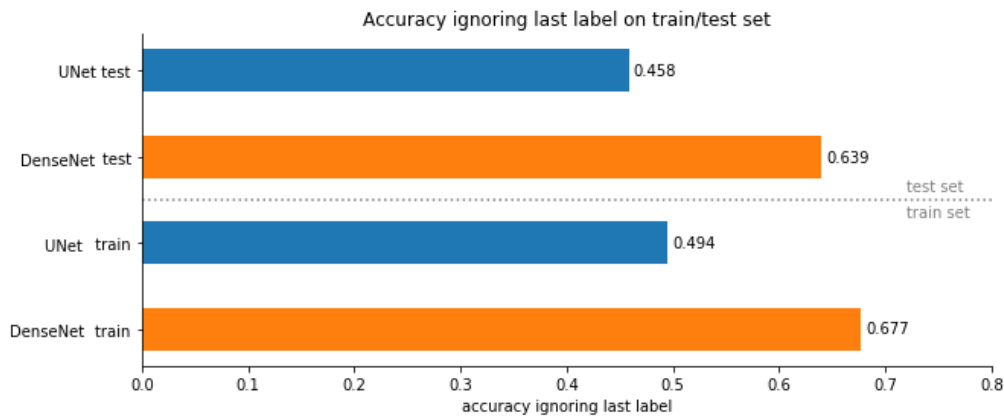
Για τη μετρική **jaccard**. Σύμφωνα και με το σχήμα 2.17 είναι εύκολο κανείς να καταλάβει ότι τα μοντέλα τα οποία δεν προβλέπουν σωστά αυτή τη μετρική με βάση τον τύπο της 2.16 θα έχουν μικρότερη τιμή και αντίστοιχα τα μοντέλα που προβλέπουν σωστά θα έχουν μεγαλύτερη τιμή της jaccard. Άρα το σχήμα 4.17 δείχνει ότι τα DenseNet\_cw\_max και DenseNet\_cw\_thres έχουν τιμές 0.292 και 0.244, ενώ τα μοντέλα DenseNet\_cw\_init\_max και DenseNet\_cw\_init\_thres έχουν τιμές 0.325 και 0.251 μεγαλύτερες προφανώς από τα προηγούμενα μοντέλα, καθώς αυτά δεν προβλέπουν μόνο το background.

Η μετρική **accuracy\_ignoring\_last\_label** υπολογίζει την ακρίβεια αγνοώντας την τελευταία ετικέτα. Οπότε για τα μοντέλα αυτά που έχουν αρχικοποιηθεί αγνοώντας την τελευταία ετικέτα αυτή η μετρική είναι αρκετά αξιόπιστη, ενώ για τα μοντέλα αυτά που προβλέπουν μόνο το background η τιμή αυτής της μετρικής είναι 0. Οπότε από το σχήμα 4.17 δείχνει ότι τα DenseNet\_cw\_max και DenseNet\_cw\_thres έχουν τιμές 0.064 και 0 αντίστοιχα. Ενώ τα μοντέλα DenseNet\_cw\_init\_max και DenseNet\_cw\_init\_thres έχουν τιμές 0.108 και 0.010 αντίστοιχα.

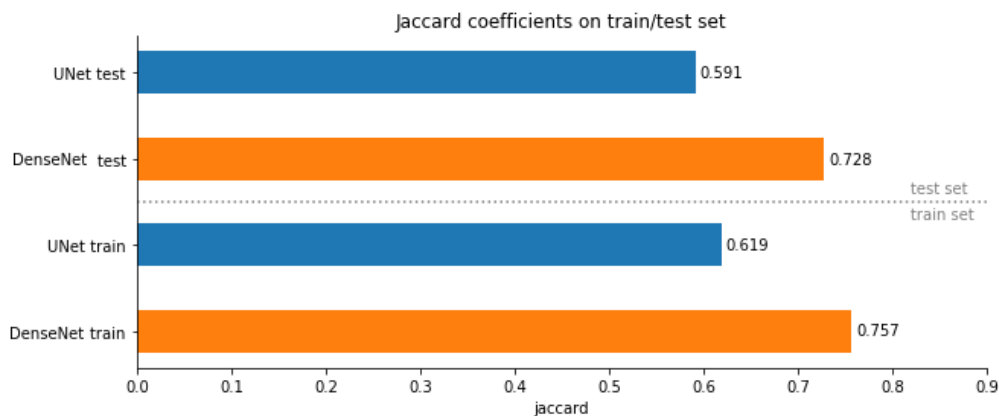
Η μετρική **F1** με δεδομένο ότι είναι ο αρμονικός μέσος της ακρίβειας και της ανάληψης αναμένεται να βγάλει αρκετά υψηλά ποσοστά. Από το σχήμα 4.17 δείχνει ότι τα DenseNet\_cw\_max και DenseNet\_cw\_thres έχουν τιμές 0.417 και 0.344 αντίστοιχα, αρκετά χαμηλά γιατί προβλέπουν την τελευταία ετικέτα, ενώ για τα DenseNet\_cw\_max και DenseNet\_cw\_thres οι τιμές είναι 0.435 και 0.375 λίγο καλύτερα λόγω της αρχικοποίησης που τους έχει γίνει.

Παρά την αρχικοποίησή τους από την εκπαίδευση χωρίς την τελευταία κλάση, δεν τα κατάφεραν καλύτερα από πριν. Πάλι το καλύτερο μοντέλο είναι **DenseNet max**.

#### 4.4.5 Σύγκριση του συνόλου Εκπαίδευσης και του συνόλου Δοκιμής για τις μετρικές `accuracy_ignoring_last_label` και `jaccard`



Σχήμα 4.18: Το διάγραμμα που αναπαριστά τα μοντέλα `UNet_ignoring_max` και `DenseNet_ignoring_max` για τη μετρική `accuracy_ignoring_last_label` τόσο στο `test set` (πάνω) όσο και στο `train set` (κάτω).



Σχήμα 4.19: Το διάγραμμα που αναπαριστά τα μοντέλα `UNet_ignoring_max` και `DenseNet_ignoring_max` για τη μετρική `jaccard` τόσο στο `test set` (πάνω) όσο και στο `train set` (κάτω).

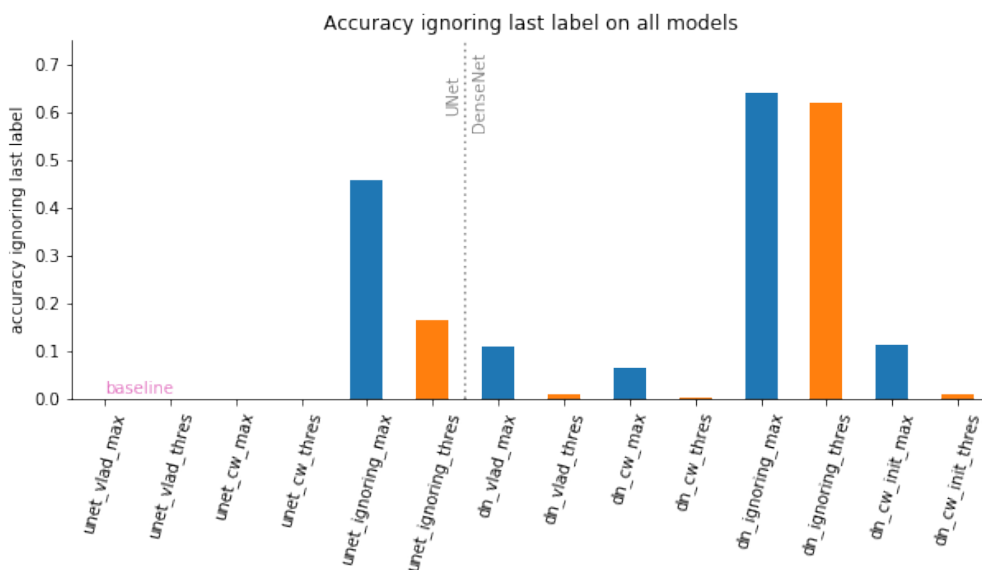
Από τις παραπάνω εικόνες παρατηρείται ότι οι τιμές της μετρικής `accuracy_ignoring_last_label`, αλλά και της μετρικής `jaccard` τόσο για το σύνολο εκπαίδευσης όσο και για το σύνολο δοκιμής είναι αρκετά κοντά και για τα δύο μοντέλα αντίστοιχα. Αυτό σημαίνει ότι δεν υπάρχει σημαντική υπερπροσαρμογή - `overfitting` ανάμεσα στο `test` και στο `train set`, αφού οι τιμές των μετρικών στο `train set` είναι πολύ λίγο μεγαλύτερες απ'ότι στο `test set`.

## 4.5 Συμπεράσματα

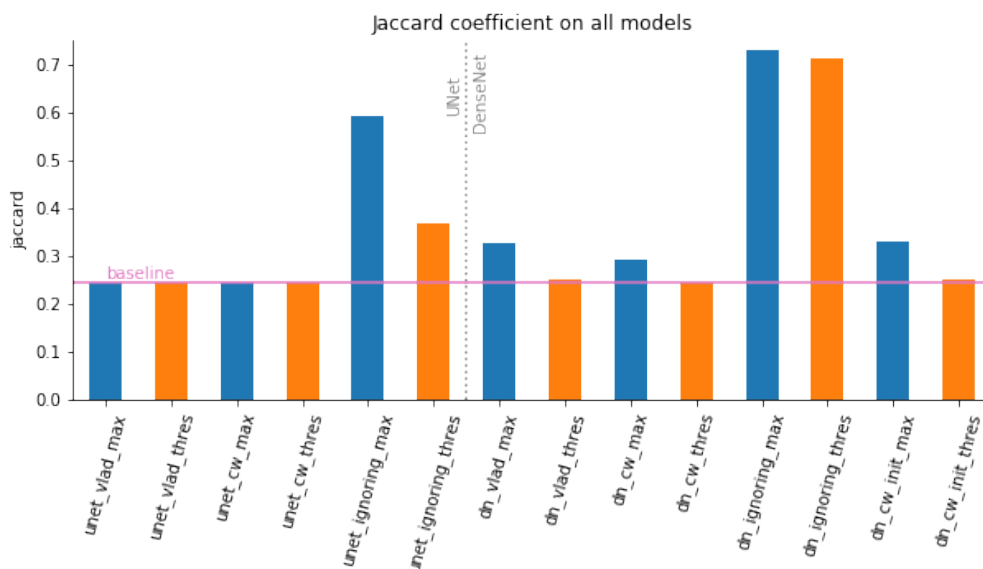
Η συγκεκριμένη διπλωματική εργασία πραγματεύεται εφαρμογές κατάτμησης καρκινικών όγκων από μαγνητικό τομογράφο. Ένας καρκινικός όγκος χαρακτηρίζεται από ανεξέλεγκτη διαίρεση κυττάρων. Οι εικόνες MRI της βάσης δεδομένων

προέρχονται από το BRATS. Αυτή η εργασία κατάτμησης βέβαια έχει αρκετές προκλήσεις καθώς το μέγεθος, το σχήμα και η θέση του εγκεφαλικού όγκου έχει πολύ σημαντικές διαφοροποιήσεις μεταξύ των ασθενών.

Μετά από την προεπεξεργασία των εικόνων και την χρήση των μοντέλων UNet και DenseNet προκύπτει η κατάτμηση των εικόνων εγκεφάλου και η ανίχνευση του υπάρχοντος ή μη όγκου. Από την διαδικασία που ακολουθήθηκε τα μοντέλα που προέβλεπαν καλύτερα ήταν αυτά που αγνοούσαν την τελευταία ετικέτα που έχει οριστεί να αντιστοιχεί στο background. Αντίθετα τα μοντέλα που εκπαιδεύονται χωρίς να αγνοούν την τελευταία ετικέτα στις περισσότερες περιπτώσεις προβλέπουν μόνο background. Το πρόβλημα με την πρόβλεψη από πολλά μοντέλα μόνο της τελευταίας ετικέτας έγκειται στο γεγονός ότι παρόλο που εξαρχής έγινε μια προεπεξεργασία των αρχικών εικόνων ώστε να φύγει το μεγαλύτερο μέρος του μαύρου υποβάθρου, αυτό παραμένει σε μεγάλο βαθμό πάνω στην εικόνα, καθώς τέτοια περιοχή θεωρείται ακόμα και το μέρος του εγκεφάλου που δεν έχει κάποιον όγκο, πόσο μάλλον οι εικόνες του εγκεφάλου που δεν έχουν καθόλου όγκο. Το καλύτερο μοντέλο καθ'όλη τη διάρκεια των πειραμάτων ήταν το DenseNet max. Για τον σκοπό της διπλωματικής αυτής, οι πιο αντικειμενικές μετρικές είναι η `accuracy_ignoring_last_label` και η `jaccard` και η `F1`. Η πρώτη μετρική εκφράζει την ακρίβεια των μοντέλων αγνοώντας την τελευταία ετικέτα, της οποίας η παρουσία είναι τόσο έντονη που επηρεάζει και πολλές μετρικές (πχ. `accuracy`). Γι'αυτό το λόγο, τα μοντέλα που προβλέπουν παντού μηδέν δεν εμφανίζονται στα διαγράμματα, όπως φαίνεται και στο σχήμα 4.20, ενώ τα καλύτερα μοντέλα έχουν υψηλά ποσοστά. Η δεύτερη μετρική ορίζεται ως το μέγεθος της τομής διαιρούμενο με το μέγεθος της ένωσης των συνόλων δειγμάτων (τομή και ένωση των προβλέψεων με τις ετικέτες). Όπως φαίνεται και στο σχήμα 4.21 τα μοντέλα που έχουν παντού μηδενικά είναι σχεδόν στην ίδια ευθεία και σε χαμηλότερα ποσοστά, ενώ τα καλά μοντέλα έχουν πιο υψηλά ποσοστά.



**Σχήμα 4.20:** Το διάγραμμα αναπαριστά όλα τα μοντέλα για τη μετρική `accuracy_ignoring_last_label`. Αριστερά από την γκριζα διακεκομμένη γραμμή οι αρχιτεκτονικές UNet, ενώ δεξιά οι DenseNet. Baseline ορίζεται η απόδοση ενός δικτύου που προβλέπει μόνο background, εδώ το 0.



**Σχήμα 4.21:** Το διάγραμμα αναπαριστά όλα τα μοντέλα για τη μετρική *jaccard*. Αριστερά από την γκριζα διακεκομμένη γραμμή οι αρχιτεκτονικές UNet, ενώ δεξιά οι DenseNet. Baseline ορίζεται η απόδοση ενός δικτύου που προβλέπει μόνο *background*, εδώ το  $\theta$ .

Αυτή η εργασία έχει πολύ ενδιαφέρον για μελλοντικές μελέτες καθώς αποτελεί ένα σημαντικό εργαλείο της Ιατρικής Απεικόνισης. Με την αυτόματη κατάτμηση εικόνας, η πρόβλεψη και διάγνωση του καρκίνου στον εγκέφαλο από τους γιατρούς στο μέλλον θα μετατρέπεται σε πιο εύκολη και κυρίως πιο ακριβής διαδικασία. Αυτό μπορεί να οδηγήσει στην καλύτερη λήψη αποφάσεων για το βέλτιστο τρόπο θεραπείας καθώς και χειρουργικής αντιμετώπισης.

## 4.6 Μελλοντική Εργασία

Υπάρχουν πολλά ακόμα μοντέλα και βελτιώσεις πάνω σε αυτά που μπορούν να γίνουν, ώστε η ανίχνευση του εγκεφαλικού όγκου με χρήση βαθιάς μάθησης να δίνει ακόμα καλύτερα αποτελέσματα.

Μία πολύ σημαντική βελτίωση θα ήταν μελετηθεί κι άλλο το μοντέλο που αγνοεί την τελευταία ετικέτα κατά την εκπαίδευση, ώστε να βγάζει ακόμα καλύτερα αποτελέσματα. Θα μπορούσαν να γίνουν επιπλέον πειράματα με διαφορετικές τιμές του κατωφλίου που χρησιμοποιείται για τον τρόπο πρόβλεψης. Αυτό θα είχε ως στόχο να αυξηθεί η μετρική *precision* ώστε να κάνει πιο στοχευμένες προβλέψεις.

Μια ακόμα ιδέα που έχει εφαρμοστεί ήδη από συμμετέχοντες στο challenge του brats είναι από την εικόνα εισόδου να γίνει κατάτμηση του συνολικού όγκου και μετά κλιμακωτά να γίνει κατάτμηση από την εικόνα του συνολικού όγκου, για τον πυρήνα του όγκου και στη συνέχεια αυτής της αλυσίδας να γίνει κατάτμηση του ενισχυτικού όγκου, από την εικόνα του πυρήνα του.

Κλείνοντας, είναι φανερό ότι αυτή η εργασία έχει αρκετές πιθανές μελλοντικές εξελίξεις. Τα αποτελέσματα προσγνωστικής διάγνωσης παρακινούν να επεκταθεί αυτό το πλαίσιο και για την ανίχνευση και τον εντοπισμό όγκων και στα άλλα όργανα. Αυτό μπορεί να βοηθήσει τους γιατρούς για τη διάγνωση σημαντικών ασθενειών έγκαιρα.



# Βιβλιογραφία

- . Δασκαλάκης. Video: Διάλεξη του Δρ. Κωνσταντίνου Δασκαλάκη — Τεχνητή Νοημοσύνη: Το μέλλον τώρα;. <https://www.auth.gr/video/24941>, 2018.
- BRATS. Multimodal Brain Tumor Segmentation Challenge 2017. <https://www.med.upenn.edu/sbia/brats2017/tasks.html>, 2017.
- B. C. Csáji. Approximation with artificial neural networks. *Faculty of Sciences, Eötvös Loránd University, Hungary*, 24:48, 2001.
- G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of control, signals and systems*, 2(4):303–314, 1989.
- C. Darken, J. Chang, and J. Moody. Learning rate schedules for faster stochastic gradient search. In *Neural Networks for Signal Processing [1992] II., Proceedings of the 1992 IEEE-SP Workshop*, pages 3–12. IEEE, 1992.
- R. DiPietro. A Friendly Introduction to Cross-Entropy Loss. <https://rdipietro.github.io/friendly-intro-to-cross-entropy-loss/>, 2016.
- M. Epperson. Empowering conservation through deep convolutional neural networks and unmanned. 2018.
- I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio. *Deep learning*, volume 1. MIT press Cambridge, 2016.
- A. Géron. Introducing capsule networks. <https://www.oreilly.com/ideas/introducing-capsule-networks>, 2018.
- L. Harasim. *Learning theory and online technologies*. Routledge, 2012.
- K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.
- K. Hornik. Approximation capabilities of multilayer feedforward networks. *Neural networks*, 4(2):251–257, 1991.
- G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten. Densely connected convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, volume 1, page 3, 2017.
- imdb. Star Wars. [https://www.imdb.com/title/tt0076759/?ref\\_=ttmi\\_tt](https://www.imdb.com/title/tt0076759/?ref_=ttmi_tt).
- S. J Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*, chapter Intelligent agents, pages 31–52. Prentice Hall, 01 1995. ISBN 0137903952.

- G. James, D. Witten, T. Hastie, and R. Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.
- S. Jégou, M. Drozdal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 1175–1183. IEEE, 2017.
- KERAS. Keras Documentation. <https://keras.io/>.
- D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.
- Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.
- S. Linnainmaa. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, 16(2):146–160, 1976.
- J. McCarthy, M. L. Minsky, N. Rochester, and C. E. Shannon. A proposal for the dartmouth summer research project on artificial intelligence, august 31, 1955. *AI magazine*, 27(4):12, 2006.
- P. McCorduck. *Machines Who Think (2nd Ed.)*. A. K. Peters, 2004.
- B. H. Menze, A. Jakab, S. Bauer, J. Kalpathy-Cramer, K. Farahani, J. Kirby, Y. Burren, N. Porz, J. Slotboom, R. Wiest, et al. The multimodal brain tumor image segmentation benchmark (brats). *IEEE transactions on medical imaging*, 34(10):1993–2024, 2015.
- M. L. Minsky. *Computation: finite and infinite machines*. Prentice-Hall, Inc., 1967.
- T. M. Mitchell. *The discipline of machine learning*, volume 9. Carnegie Mellon University, School of Computer Science, Machine Learning Department, 2006.
- M. Moreno Lopez. *Deep Learning For Brain Tumor Segmentation*. PhD thesis, University of Colorado Colorado Springs. Kraemer Family Library, 2017.
- N. M. Nasrabadi. Pattern recognition and machine learning. *Journal of electronic imaging*, 16(4):049901, 2007.
- N. National Cancer Institute. General Information About Adult Brain Tumors. <https://web.archive.org/web/20140705225928/http://www.cancer.gov/cancertopics/pdq/treatment/adultbrain/Patient/page1/AllPages>, 2014.
- M. Pathegama and Ö. Göl. Edge-end pixel extraction for edge-based image segmentation. ICSP, 2004.
- F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

- D. L. Pham, C. Xu, and J. L. Prince. Current methods in medical image segmentation. *Annual Review of Biomedical Engineering*, 2(1):315–337, 2000. doi: 10.1146/annurev.bioeng.2.1.315. URL <https://doi.org/10.1146/annurev.bioeng.2.1.315>. PMID: 11701515.
- D. M. Powers. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. 2011.
- L. Prechelt. Early stopping-but when? In *Neural Networks: Tricks of the trade*, pages 55–69. Springer, 1998.
- N. Qian. On the momentum term in gradient descent learning algorithms. *Neural networks*, 12(1):145–151, 1999.
- B. D. Ripley. *Pattern recognition and neural networks*. Cambridge university press, 2007.
- H. Robbins and S. Monro. A stochastic approximation method. In *Herbert Robbins Selected Papers*, pages 102–109. Springer, 1985.
- O. Ronneberger, P. Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. *CoRR*, abs/1505.04597, 2015. URL <http://arxiv.org/abs/1505.04597>.
- F. Rosenblatt. Principles of neurodynamics. perceptrons and the theory of brain mechanisms. Technical report, CORNELL AERONAUTICAL LAB INC BUFFALO NY, 1961.
- S. Ruder. An overview of gradient descent optimization algorithms. <http://ruder.io/optimizing-gradient-descent/index.html#stochasticgradientdescent>, 2016.
- D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning representations by back-propagating errors. *nature*, 323(6088):533, 1986.
- B. Sathya and R. Manavalan. Image segmentation by clustering methods: performance analysis. 2011.
- D. Scherer, A. Müller, and S. Behnke. Evaluation of pooling operations in convolutional architectures for object recognition. In *Artificial Neural Networks–ICANN 2010*, pages 92–101. Springer, 2010.
- A. Sharma. Understanding Activation Functions in Neural Networks. <https://medium.com/the-theory-of-everything/understanding-activation-functions-in-neural-networks-9491262884e0>, 2017.
- G. Srinivasan and G. Shobha. Segmentation techniques for target recognition. *International Journal of Computers and Communications*, 1(3):313–333, 2007.
- N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

- L. A. Steen, J. A. Seebach, and L. A. Steen. *Counterexamples in topology*, volume 18. Springer, 1978.
- S. V. Stehman. Selecting and interpreting measures of thematic classification accuracy. *Remote sensing of Environment*, 62(1):77–89, 1997.
- R. Sutton. Two problems with back propagation and other steepest descent learning procedures for networks. In *Proceedings of the Eighth Annual Conference of the Cognitive Science Society, 1986*, pages 823–832, 1986.
- M. van Gerven and S. Bohte. *Artificial neural networks as models of neural information processing*. Frontiers Media SA, 2018.
- A. S. Walia. What are Activation functions and what are it uses in a Neural Network Model? <https://towardsdatascience.com/activation-functions-and-its-types-which-is-better-a9a5310cc8f>, 2017a.
- A. S. Walia. Types of Optimization Algorithms used in Neural Networks and Ways to Optimize Gradient Descent. <https://towardsdatascience.com/types-of-optimization-algorithms-used-in-neural-networks-and-ways-to-optimize-gr> 2017b.
- Z. Wang. Automatic brain tumor segmentation by deep convolutional networks and graph cuts. 2018.
- Z. Zhang. Unet-tensorflow-keras. <https://github.com/zizhaozhang/unet-tensorflow-keras>, 2017.