National Technical University of Athens
School of Electrical and Computer Engineering
Τομέας Συστημάτων Μετάδοσης Πληροφορίας και Τεχνολογίας Υλικών

Texture Analysis of Histopathology Slides for the prediction of EGFR Gene Mutation

DIPLOMA THESIS

MARIA AMALIA TOURNI

Supervisor : Γεώργιος Ματσόπουλος
Καθηγητής Ε.Μ.Π.

Athens, September 2018

National Technical University of Athens
School of Electrical and Computer Engineering
Τομέας Συστημάτων Μετάδοσης Πληροφορίας και Τεχνολογίας Υλικών

Texture Analysis of Histopathology Slides for the prediction of EGFR Gene Mutation

DIPLOMA THESIS

ΜΑΡΊΑ ΑΜΑΛΊΑ ΤΟΥΡΝΉ

Supervisor : Γεώργιος Ματσόπουλος
Καθηγητής Ε.Μ.Π.

Approved by the examining committee on the September 27, 2018.

Γεώργιος Ματσόπουλος Αναπληρωτής Καθηγητής Ε.Μ.Π.	Λεωνίδας Αλεξόπουλος Επίκουρος Καθηγητής Ε.Μ.Π.	Νικόλαος Ουζούνογλου Καθηγητής Ε.Μ.Π.

Athens, September 2018

Maria Amalia Tourni
Electrical and Computer Engineer

Περίληψη

Το αντικείμενο της διπλωματικής αυτής εργασίας είναι η ανίχνευση της πιθανής επίδρασης μιας σωματικής μετάλλαξης στην οπτική υφή μιας ιστοπαθολογικής εικόνας όγκου του καρκίνου του πνεύμονα. Συγκεκριμένα, εξετάζουμε τη μετάλλαξη του γονιδίου EGFR στην περίπτωση του αδενοκαρκινώματος, με τη χρήση των σύγχρονων υπολογιστικών μεθόδων αναγνώρισης εικόνας. Το αδενοκαρκίνωμα αποτελεί περίπου το 40% των περιπτώσεων καρκίνου του πνεύμονα και κατά τα τελευταία έτη έχουν αναπτυχθεί αρκετές στοχευμένες θεραπείες που σχετίζονται με συγκεκριμένες γονιδιακές μεταλλάξεις, όπως είναι του γονιδίου EGFR. Η σύνδεση μεταξύ μιας μετάλλαξης και της αντίστοιχης ιστοπαθολογίκής εικόνας μπορεί να οδηγήσει σε πολύτιμα συμπεράσματα που εν δυνάμει μπορούν να οδηγήσουν σε ταχύτερη και ακριβέστερη διάγνωση.

Στην αρχή, το πρόβλημα αναλύεται από βιολογική και υπολογιστική πλευρά, προκειμένου να προσδιοριστεί η καλύτερη δυνατή προσέγγιση. Ακολούθως, αναπτύσσονται δύο διαφορετικές υπολογιστικές μέθοδοι με σκοπό την ανίχνευση χαρακτηριστικών στις ιστολογικές εικόνες ασθενών που χαρακτηρίζονται από σωματική μετάλλαξη στο EGFR. Η πρώτη προσέγγιση γίνεται με τη χρήση ενός βαθέος νευρωνικού δικτύου για αναγνώριση εικόνων και επιτυγχάνει ένα αρκετά καλό ποσοστό ταξινόμησης. Ωστόσο, απαιτείται περαιτέρω ανάλυση για την κατανόηση των παραγόμενων χαρακτηριστικών, κάτι το οποίο δε μπορεί να γίνει μέσω του νευρωνικού, και επομένως εφαρμόζεται η χρήση ψηφιακών φίλτρων για τη μελέτη υφής των εικόνων. Η μέθοδος αυτή επιτυγχάνει ένα ακόμα καλύτερο ποσοστό επιτυχίας, το οποίο δηλώνει την ύπαρξη μορφολογικής διαφοροποίησης του ιστού στην περίπτωση μετάλλαξης στο γονίδιο EGFR.

Ταυτοχρόνως, εξετάζονται οι υποτύποι αδενοκαρκινώματος, που αποτυπώνουν μοτίβα από χαρακτηριστικά έκφρασης γονιδίων, ως προς τη διαφοροποίηση της υφής στις αντίστοιχες διαφάνειες ιστών. Αυτό γίνεται στη προσπάθεια διάκρισης των χαρακτηριστικών που ανιχνεύονται μεταξύ ενός συγκεκριμένου υποτύπου, του TRU, ο οποίος έχει αποδειχθεί ότι συνδέεται με τις μεταλλάξεις στο EGFR, και τα χαρακτηριστικά της υφής που σχετίζονται με τη μετάλλαξη του EGFR.

Τέλος, παρουσιάζονται λεπτομερώς τα παραπάνω αποτελέσματα, συγκλίνοντας στην ύπαρξη μιας σαφούς σχέσης μεταξύ της σωματικής μετάλλαξης του γονιδίου EGFR και της ιστοπαθολογικής αλλοίωσης του αντίστοιχου καρκινικού ιστού.

Λέξεις κλειδιά

Classification, Digital Filtering, Deep Learning, Computational Biology, Precision Medicine, Image Analysis, Computer Vision, Pattern Recognition, Medical Data, EGFR, TCGA, Mutation, Cancer, Adenocarcinoma, Expression

Abstract

This thesis studies the effect that a somatic mutation has on the texture format of a tumor histopathology slide. We examine specifically the case of the EGFR mutation on Adenocarcinoma Lung Cancer type, with use of the latest computational methods. Adenocarcinoma Lung Cancer accounts for about 40% of all lung cancers and during the latest years, many targeted therapies related to specific gene mutations, such as the EGFR gene, have been developed. The link between the mutation and the histopathology slide can lead to faster, more accurate diagnosis as well as valuable pattern detection.

The problem is firstly addressed and analyzed both biologically and computationally, to determine the best possible approach. Following this, two different computational methods are developed with the purpose of detecting texture feature within tissue slides characterized by an EGFR mutation. The first method used is Convolutional Neural Networks for image recognition and achieves a good classification rate. Further analysis for the origin of the produced features is needed though, and therefore, the second method of Digital Image Texture Analysis is applied. That achieves an even better success rate, which strongly implies the existence of texture features connected to the EGFR mutation.

At the same time, LUAD gene expression subtypes, are also explored in terms of texture differences on their corresponding tissue slides. This is performed mainly to distinguish the features detected between a specific subtype, TRU, which is enriched with EGFR mutation, and EGFR mutation related texture features.

Finally, a thorough presentation of the above results is made, which all conclude that there is a distinct connection between the presence of an EGFR somatic mutation and its effects on the texture appearance of the tissue slide.

Key words

Classification, Digital Filtering, Deep Learning, Computational Biology, Precision Medicine, Image Analysis, Computer Vision, Pattern Recognition, Medical Data, EGFR, TCGA, Mutation, Cancer, Adenocarcinoma, Expression

Ευχαριστίες

Η παρούσα διπλωματική εργασία εκπονήθηκε στο πλαίσιο του προπτυχιακού προγράμματος σπουδών της σχολής Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών του Εθνικού Μετσόβιου Πολυτεχνείου, σε συνεργασία με την Ιατρική Σχολή του Πανεπιστημίου της Νέας Υόρκης.

Θα ήθελα αρχικά να ευχαριστήσω τον Δρ. Αριστοτέλη Τσιρίγο, Αναπληρωτή Καθηγητή Παθολογίας και Διευθυνής του Κέντρου Εφαρμοσμένης Βιοπληροφορικής στην Ιατρική Σχολή του Πανεπιστημίου της Νέας Υόρκης, για τη δυνατότητα που μου έδωσε να γίνω μέλος του εργαστηρίου του και να εργαστώ σε ένα άκρως ενδιαφέροντα τομέα. Η καθοδήγηση και υποστήριξή του υπήρξε καθοριστική για την εκπόνηση της εργασίας μου, ενώ η άψογη συνεργασία μας, η μεταδοτικότητα του και το ενδιαφέρον του για αυτόν τον τομέα ήταν ο καταλύτης για να αγαπήσω και εγώ το αντικείμενο και να ασχοληθώ με αυτό.

Στη συνέχεια θα ήθελα να ευχαριστήσω τον Δρ. Γεώργιο Ματσόπουλο, Καθηγητή Ε.Μ.Π, καθώς και τον Δρ. Βασίλη Κορφιάτη, μέλος του εργαστηρίου, για την πολύτιμη βοήθεια τους και συμβολή του τόσο στο επιστημονικό και τεχνικό κομμάτι της εργασίας αλλά και σε ό,τι αφορά γενικότερες συμβουλές για τον τομέα της βιοπληροφορικής. Επιπλέον, οφείλω ιδιαίτερες ευχαριστίες στον Δρ. Λεωνίδα Αλεξόπουλο, Αναπληρωτή Καθηγητή Ε.Μ.Π. για την άψογη συνεργασία μας, την απαραίτητη καθοδήγησή του καθώς κυρίως και για την ένταξη μου στο εργαστήριο του, μέσω του οποίου εκτέθηκα σε άλλα projects και γνώρισα πολλά ενδιαφέροντα άτομα.

Η διπλωματική αυτή εργασία δε θα ήταν πραγματικότητα χωρίς το Δρ. Θεόδωρο Σακελλαρόπουλο, συνδετικό κρίκο όλων των παραπάνω κόσμων. Του οφείλω ιδιαίτερες ευχαριστίες για το χρόνο που αφιέρωσε και τη θεμελιώδη συνεισφορά του. Η εμπειρία και οι γνώσεις του με βοήθησαν πάρα πολλές φορές να προχωρήσω σε δύσκολα σημεία. Τον ευχαριστώ ιδιαίτερα και για την πνευματική στήριξη και τις εξαιρετικές συμβουλές του για την πορεία μου στο νέο αυτό για μένα τομέα.

Τέλος, η εργασία αυτή είναι αφιερωμένη στην οικογένεια μου, την αδερφή μου Ισιδώρα, και όλους τους φίλους μου που βρίσκονταν κοντά μου σε όλη τη διάρκεια των 5 τελευταίων ετών, με στηρίζουν διαρκώς, και τους είμαι ευγνώμων για όλα όσα έχω πετύχει και όσα ζήσαμε μαζί.

Μαρία Αμαλία Τουρνή,

Αθήνα, 27η Σεπτεμβρίου 2018

Acknowledgements

This diploma thesis was written as a part of the undergraduate studies at the school of Electrical and Computer Engineering at the National Technical University of Athens, in collaboration with the New York University Medical School.

I would like to start by thanking Dr. Aristotelis Tsirigos, Associate Professor of Pathology and Director of the Center for Applied Bioinformatics at the New York University Medical School, for giving me the opportunity to become a member of his laboratory and work in a highly interesting field. His thorough guidance and support, our excellent cooperation, and moreover his vast interest in this field were the catalyst for me to also love the subject and decide to dive further into it.

Following, I would like to thank my supervisor Dr. Georgios Matsopoulos, Professor of NTUA, and Dr. Vassilis Korfatis, member of Dr. Matsopoulos' laboratory, for their valuable help and contribution to the scientific and technical part of the work, as well as providing general and apt advice on the field of bioinformatics.

In addition, I would like to express my special thanks to Dr. Leonidas Alexopoulos, Associate Professor NTUA for our excellent cooperation, his necessary guidance, as well as including me in his laboratory, via which I became exposed to other projects and met many interesting people.

Another important person is Dr. Theodore Sakellaropoulos, a link of all the above worlds. I owe him particular thanks for the time he devoted and his fundamental contribution. His experience and knowledge have helped me many times to move on from difficult points. I also thank him very much for his overall support and his excellent advice on my progress in this new, for me, area.

Finally, this work is dedicated to my family, my sister Isidora, and all my friends who have been with me throughout the last 5 years, and are always supporting me. I am grateful for everything that I have achieved with them being by my side and for what we experienced together.

Maria Amalia Tourni,

Athens, September 27, 2018

Contents

List of Tables

List of Figures

Chapter 1

Εκτεταμένη Περίληψη

Στο κεφάλαιο αυτό δίνεται μια αναλυτική περίληψη της διπλωματικής εργασίας. Θα παρουσιαστούν συνοπτικά βασικές έννοιες θεωρίας και στη συνέχεια θα δοθεί έμφαση στα αποτελέσματα.

1.1 Θεωρητικό Υπόβαθρο

Σε αυτό το κομμάτι θα παρουσιαστούν βασικές βιολογικές έννοιες και υπολογιστικές τεχνικές που χρησιμοποιούνται στην εργασία. Στο κεφάλαιο 3 ακολουθεί αναλυτική περιγραφή και περαιτέρω εμπλουτισμός των εννοιών.

1.1.1 Βιολογικοί Ορισμοί

Καρκίνος του Πνεύμονα

Ο καρκίνος του πνεύμονα αποτελεί το 2ο πιο συχνό είδος καρκίνου, και την 1η πιο κοινή αιτία θανάτου λόγω καρκίνου στις Ηνωμένες Πολιτείες, σύμφωνα με το American Cancer Society [8]. Υπάρχουν 2 κύριες κατηγορίες: Ο μη μικροκυτταρικός καρκίνος του πνεύμονα (NSCLC), ο οποίος ισχύει για το 80-85% των περιπτώσεων και ο μικροκυτταρικός καρκίνος (SCLC) που ισχύει για τις υπόλοιπες. Ο μη μικροκυτταρικός καρκίνος του πνεύμονα έχει τρεις κύριους υποτύπους: το πλακώδες καρκίνωμα, το αδενοκαρκίνωμα και το καρκίνωμα του πνεύμονα από μεγάλα κύτταρα.

Γονιδιακή Μετάλλαξη

Η γονιδιακή μετάλλαξη ορίζεται ως μια μόνιμη αλλαγή σε ένα τμήμα της αλυσίδας DNA, η οποία αποτελεί ένα γονίδιο. Μια μετάλλαξη μπορεί να είναι είτε κληρονομική, η οποία έχει προέλθει από τον γονέα και εντοπίζεται σε όλα τα κύτταρα του οργανισμού, είτε επίκτητη-σωματική, που έχει προκληθεί κάποια στιγμή στο κύκλο ζωής του οργανισμού, εντοπίζεται μόνο σε συγκεκριμένα κύτταρα και δε μεταδίδεται κληρονομικά.[9]

Ιστοπαθολογικές Εικόνες

Οι ιστοπαθολογικές εικόνες είναι λεπτές φέτες ιστού που προέρχονται από εγχείρηση ή από κάποια βιοψία, οι οποίες έπειτα από κατάλληλη χρώση, τοποθετούνται κάτω από το μικροσκόπιο για τη μελέτη τους και την εξαγωγή συμπερασμάτων από τον εκάστοτε ειδικό. Πλέον, οι εικόνες αυτές μπορούν να ψηφιοποιηθούν και να αποθηκευτούν ηλεκτρονικά, με πολύ μεγάλη ανάλυση.[10]

Γονίδιο EGFR

Το EGFR (Epidermal growth factor receptor) είναι μια διαμεμβρανική πρωτεΐνη, μέλος της οικογένειας των υποδοχέων της τυροσινικής κινάσης και αποτελεί βασικό σηματοδότη στην ενεργοποίηση διαφόρων κυτταρικών μοριακών μονοπατιών. Έχει διαπιστωθεί ότι το γονίδιο που είναι υπεύθυνο για την παραγωγή της πρωτεΐνης EGFR εκφράζεται σε περισσότερο από το 60% των καρκίνων του πνεύμονα

μη μικροκυτταρικού τύπου, το οποίο το καθιστά έναν ιδανικό παράγοντα μελέτης για την εύρεση στοχευμένης θεραπείας.[11]

Μέθοδος Ανάγνωσης DNA & RNA

Η μέθοδος ανάγνωσης (''αλληλούχησης'') του DNA ορίζει τη διαδικασία καταγραφής της σειράς των νουκλεοτιδίων ενός δεδομένου θραύσματος DNA. Οι χρησιμοποιούμενες μέθοδοι στοχεύουν στον προσδιορισμό των τεσσάρων βάσεων - αδενίνη, γουανίνη, κυτοσίνη και θυμίνη - σε ένα σκέλος DNA. Η ανάγνωση ενός ολόκληρου γονιδιώματος είναι μια περίπλοκη διαδικασία που περιλαμβάνει την εύρεση της αλληλουχίας μικρότερων, επικαλυπτόμενων τμημάτων του DNA και το συνδυασμό τους σε μία μοναδική τελική ακολουθία. Η ακολουθία του DNA περιέχει όλες τις πληροφορίες που καθορίζουν τη λειτουργία ενός οργανισμού. Η καταγραφή της και η αποκωδικοποίηση των πληροφοριών που εμπεριέχει είναι καταλυτικός παράγοντας για την κατανόηση της.[12]

Σε ότι αφορά την ανάγνωση RNA, οι πληροφορίες που εξάγουμε είναι παρόμοιες με τις αντίστοιχες του DNA. Ωστόσο, μπορεί να εμπεριέχει πολύτιμες πληροφορίες για την κατάσταση του εξεταζόμενου ιστού, καθώς θα αποκαλύψει τις αλληλουχίες που εκφράζονται ενεργά μέσα στο κύτταρο.

Έκφραση Γονιδίου

Γονιδιακή έκφραση είναι η διαδικασία της μεταγραφής του κώδικα γονιδίου του DNA σε κώδικα RNA προκειμένου να αρχίσει η παραγωγή της επιθυμητής πρωτεΐνης με σκοπό να επιτελεστεί μια συγκεκριμένη ενέργεια. Υπολογίζεται ως ο αριθμός των RNA αντιγράφων που παράγονται μέσα στο κύτταρο, τα οποία οδηγούν σε αντίστοιχη ποσότητα στην παραγωγή της κωδικοποιημένης πρωτεΐνης. [13]

Υποτύποι Αδενοκαρκινώματος

Αρκετές μελέτες που έχουν επεξεργαστεί τα αποτελέσματα ανάλυσης μεταγραφικών λειτουργιών των κυττάρων αδενοκαρκινώματος, όπως αυτά έχουν προκύψει από την ανάγνωση DNA & RNA, κατέληξαν στη δημιουργία τριων κλάσεων που βασίζονται και σε δεδομένα γονιδιακής έκφρασης και μορφολογικά χαρακτηριστικά του καρκινικού όγκου. Οι Eric A. Collisson et al [14] πρότειναν την ακόλουθη ονοματολογία :

1. Terminal respiratory unit (TRU, formerly bronchioid)
2. Proximal-inflammatory (PI, formerly squamoid)
3. Proximal-proliferative (PP, formerly magnoid)

Αρκετές μελέτες έχουν ανακαλύψει συσχετισμούς μεταξύ των κλάσεων και κλινικών αποτελεσμάτων. Για παράδειγμα, ο υποτύπος TRU έχει αποδειχθεί συσχετισμένος με την πλειοψηφία των ογκολογικών δειγμάτων που παρουσιάζουν μετάλλαξη στο γονίδιο EGFR.

1.1.2 Υπολογιστικές Μέθοδοι

Στο κομμάτι αυτό θα παρουσιάσουμε τα δύο βασικά υπολογιστικά μοντέλα που χρησιμοποιήσαμε. Αναλυτική περιγραφή μπορεί να βρεθεί στο κεφάλαιο 4.

Μηχανική Μάθηση & Βαθιά Νευρωνικά Δίκτυα

Οι αλγόριθμοι μηχανικής μάθησης χρησιμοποιούνται για την πρόβλεψη και κατηγοριοποίηση ενός συνόλου δεδομένων σε ένα γνωστό ή άγνωστο αριθμό τάξεων. Αυτό επιτυγχάνεται είτε με τη μάθηση της σχέσης που υπάρχει μεταξύ των χαρακτηριστικών και των τιμών-στόχων του συνόλου δεδομένων, είτε δημιουργώντας ένα νέο σύνολο κατηγοριών που ομαδοποιεί τα δεδομένα με τον καλύτερο δυνατό

τρόπο. Η επιτυχία της πρόβλεψης εξετάζεται σε ένα σύνολο άγνωστων δεδομένων με σκοπό την σωστή πρόβλεψη κατηγοριοποίησης. Οι δύο κύριοι τρόποι μηχανικής μάθησης είναι η επιβλεπόμενη μάθηση, που οι κατηγορίες είναι γνωστές, και η μη επιβλεπόμενη μάθηση, που οι κατηγορίες είναι άγνωστες.

Ένας αλγόριθμος μηχανικής μάθησης, που μπορεί να εφαρμοστεί και στις δύο κατηγορίες, είναι τα Τεχνητά Νευρωνικά Δίκτυα. Η αρχιτεκτονική τους είναι εμπνευσμένη από τα βιολογικά νευρωνικά δίκτυα του νευρικού συστήματος. Κατά την εκπαίδευσή τους μπορούν να αναλύουν και να ανιχνεύουν πρότυπα και χαρακτηριστικά των δεδομένων εισόδου, που μπορεί να είναι δύσκολο ή και αδύνατο να βρεθούν από ανθρώπους.

Μια υποκατηγορία νευρωνικών δικτύων είναι τα βαθιά - συνελικτικα νευρωνικά δίκτυα. Τα συνελικτικα νευρωνικά δίκτυα (CNN) έχουν σχεδιαστεί κυρίως για χρήση σε ανεπεξέργαστα δεδομένα, καθώς λόγω της πολυπλοκότητας τους, μπορούν να ανιχνεύουν και να αποκωδικοποιούν ευκολότερα πιο περίπλοκα χαρακτηριστικά.

Δύο ακόμη αλγόριθμοι μηχανικής μάθησης που χρησιμοποιήθηκαν στο πλαίσιο της εργασίας είναι το Δέντρο Αποφάσεων και η Λογιστική Παλινδρόμηση. Περισσότερη ανάλυση για τους αλγορίθμους αυτούς γίνεται στο κεφάλαιο 4.

Ψηφιακά Φίλτρα Εξαγωγής Χαρακτηριστικών Εικόνας

Η ψηφιακή επεξεργασία εικόνας με τη χρήση φίλτρων αποτελεί μια άκρως διαδεδομένη τεχνική για την ανίχνευση χαρακτηριστικών υφής σε εικόνες με τη χρήση αλγορίθμων. Η υφή μιας εικόνας εκφράζεται από τη χωρική κατανομή των χρωμάτων και των τόνων στην εικόνα. Η ποσοτικοποίηση της γίνεται με τη χρήση μεθόδων που υπολογίζονται άμεσα ή έμμεσα από τους τόνους της εικόνας. Μέσω αυτών των τεχνικών, μπορεί να προκύψουν και συμπεράσματα ανίχνευσης μοτίβων που το ανθρώπινο μάτι θα δυσκολευόταν να αναγνωρίσει.

Μια κλασσική διαδικασία ψηφιακής αναγνώρισης υφής, η οποία εφαρμόστηκε στην εργασία αυτή, είναι η εξής :

1. Ψηφιακή Τμηματοποίηση Εικόνας: Η τεχνική αυτή χρησιμοποιείται με σκοπό τον διαχωρισμό περιοχών της εικόνας οι οποίες παρουσιάζουν μορφολογικό ενδιαφέρον, σε σύγκριση με όσες μπορούν να παραληφθούν και να θεωρηθούν ”φόντο”.

2. Χρήση Ψηφιακών Φίλτρων σε εικόνες: Μια εικόνα, στη μορφή ενός διανύσματος, περνά ως είσοδος σε γραμμικά ή μη γραμμικά φίλτρα, με στόχο τη παραγωγή ενός διανύσματος που περιγράφει μοναδικά τις λεπτομέρειες υφής κάθε τμήματός της. Το διάνυσμα αυτό μπορεί αργότερα να χρησιμοποιηθεί ως είσοδος σε αλγόριθμους ταξινόμησης και / ή ομαδοποίησης που ολοκληρώνουν τη διαδικασία ανίχνευσης χαρακτηριστικών υφής.

   Τα φίλτρα τα οποία χρησιμοποιήθηκαν στην εργασία αυτή, η αναλυτική μαθηματική περιγραφή των οποίων βρίσκεται στο κεφάλαιο 4, είναι τα εξής :

   (a) Discrete Wavelet Transform.
   (b) Gabor Kernels
   (c) Haralick Texture Filters
   (d) Fractal Dimension Filters
   (e) Local Binary Patterns

3. Κατηγοριοποίηση Χαρακτηριστικών Υφής: Έχοντας ως είσοδο ένα ενοποιημένο διάνυσμα, όπως προκύπτει από τα φίλτρα που επιλέγονται να εφαρμοστούν στην εικόνα, στη συνέχεια με τη χρήση ενός αλγορίθμου ταξινόμησης, μπορούμε να δημιουργήσουμε ένα σύστημα ανίχνευσης της κατηγοριοποίησης των χαρακτηριστικών υφής των δεδομένων, μέ τη χρήση μηχανικής μάθησης.

1.2 Κεντρικός άξονας εργασίας

Η εργασία είναι βασισμένη στη δουλειά που παρουσιάζεται στο άρθρο "Classification and Mutation Prediction from Non-Small Cell Lung Cancer Histopathology Images using Deep Learning" [7] που δημοσιεύθηκε το 2018.

Η ανάλυση ιστοπαθολογικών εικόνων καρκινικού κυτταρικού ιστού των πνευμόνων είναι μια από τις βασικές μεθόδους που χρησιμοποιούν οι παθολόγοι για να αξιολογήσουν το στάδιο, τον τύπο και τους υποτύπους καρκίνου των πνευμόνων.

Από τη δουλειά που προηγήθηκε, παραλάβαμε ένα βαθύ νευρωνικό δίκτυο το οποίο, με είσοδο μια ιστοπαθολογική εικόνα και ένα πίνακα χαρακτηριστικών που αντιστοιχούν στην συγκεκριμένη εικόνα εκπαιδεύτηκε επιτυχώς σε 2 διαφορετικά προβλήματα αναγνώρισης:

1. Αναγνώριση είδους καρκίνου: αδενοκαρκίνωμα, πλακώδες καρκίνωμα, φυσιολογικός ιστός. Επιτυχία αναγνώρισης 0.97 AUC (Area Under Curve) .

2. Πρόβλεψη των 10 πιο συχνών μεταλλάξεων στο αδενοκαρκίνωμα. Από αυτό προέκυψε ότι 6 από αυτά τα γονίδια - STK11, EGFR, FAT1, SETBP1, KRAS, TP53 - μπορούν να προβλεφθούν με ακρίβεια 0.733 - 0.856 AUC.

Σε αυτή τη μελέτη επικεντρωθήκαμε στο δεύτερο αποτέλεσμα και προσπαθήσαμε να εντοπίσουμε μορφολογικά χαρακτηριστικά του ιστού τα οποία είναι άγνωστα ή μη καταγεγραμμένα επισήμως και εκφράζουν μεταγραφικές αλλαγές και μεταλλάξεις.

Συγκεκριμένα, καταφέραμε να επιβεβαιώσουμε με δύο διαφορετικούς τρόπους (χρήση Βαθέος Νευρωνικού Δικτύου και χρήση φίλτρων για τη μελέτη υφής και εξαγωγή χαρακτηριστικών των εικόνων) την ύπαρξη μορφολογικής διαφοροποίησης του ιστού στην περίπτωση ύπαρξης μετάλλαξης στο γονίδιο EGFR.

1.3 Δεδομένα

Τα δεδομένα που χρησιμοποιήσαμε προέρχονται από το ερευνητικό πρόγραμμα Cancer Genome Atlas (TCGA). Το TCGA περιλαμβάνει δεδομένα που περιγράφουν καρκινικούς και φυσιολογικούς ιστούς για παραπάνω από 11.000 ασθενείς και είναι ανοιχτό για την ακαδημαϊκή και ερευνητική κοινότητα.

1.3.1 Δεδομένα Μεταλλάξεων

Χρησιμοποιήσαμε δύο ειδών δεδομένα προερχόμενα από αλληλούχηση DNA ασθενών με αδενοκαρκίνωμα. Έτσι, ανά ασθενή έχουμε δεδομένα :

- Μεταλλάξεων: Από αυτά αντλούμε την πληροφορία για την ύπαρξη μετάλλαξης στο EGFR γονίδιο, για 361 ασθενείς. Από αυτούς, 43 εμφανίζουν τη μετάλλαξη.

- Γονιδιακής Έκφρασης : Από αυτά αντλούμε την πληροφορία έκφρασης συγκεκριμένων γονιδίων, για 374 ασθενείς. Με βάση αυτά τα δεδομένα και με τη διαδικασία που περιγράφεται στο Collisson et al[14], κατηγοριοποιήσαμε τους ασθενείς σε τέσσερις υποτύπους έκφρασης: TRU, PP, PI & Unknown (στην τελευταία τοποθετήθηκαν όσοι δεν πληρούσαν το όριο κατηγοριοποίησης για την ένταξη σε όλες τις κατηγορίες).

Για κάθε ασθενή από τους παραπάνω, χρησιμοποιήθηκε η αντίστοιχη ιστοπαθολογικη εικόνα. Ωστόσο, επειδή το μέγεθος μιας εικόνας είναι πάρα πολύ μεγάλο, κάθε μια τεμαχίστηκε σε 512 x 512 pixels μη επικαλυπτόμενα πλακίδια.

1.4 Αποτελέσματα Μηχανικής Μάθησης με Βαθύ Νευρωνικό Δίκτυο

1.4.1 Πρόβλεψη Μετάλλαξης EGFR

Στο κομμάτι αυτό θα παρουσιάσουμε περιληπτικά τα αποτελέσματα της εκπαίδευσης με τη χρήση του νευρωνικού δικτύου, χωρίς λεπτομέρειες για ακριβή μεγέθη και παραμετροποιήσεις του δικτύου. Η διαδικασία περιγράφεται αναλυτικά στο κεφάλαιο 6.

Για την πρόβλεψη μετάλλαξης με βαθύ νευρωνικό δίκτυο χρησιμοποιήθηκε το δίκτυο Google Inception v3, όπως αυτό τροποποιήθηκε από τον Nicolas Coudray [7].

Ως είσοδο χρησιμοποιήθηκαν οι εικόνες, τεμαχισμένες σε πλακίδια, μαζί με την κλάση που ανήκει ο αντίστοιχος ασθενής. Σε αυτό το σημείο έγινε μια αντιστοίχιση, ώστε να κρατήσουμε ως δεδομένα εισόδου μόνο τα πλακίδια των οποίων ο ιστός έχει χαρακτηριστεί ως αδενοκαρκίνωμα και προβλέφθηκε σωστά ως αδενοκαρκίνωμα από την εκπαίδευση αναγνώρισης του είδους του καρκίνου από το βαθύ νευρωνικό.

Η δοκιμή του εκπαιδευμένου πλέον δικτύου έγινε με ποσοστό επιτυχίας 73%, το οποίο προδιαθέτει ότι υπάρχει κάποιο μορφολογικό χαρακτηριστικό που να διαχωρίζει τον καρκινικό ιστό που περιέχει μετάλλαξη στο γονίδιο EGFR, από αυτόν που δεν έχει.

Ωστόσο, με τη χρήση τεχνικών βαθιάς μάθησης, είναι αρκετά περίπλοκο και δύσκολο να βρεθούν ποια είναι αυτά τα χαρακτηριστικά που δημιουργούν αυτή τη διαφοροποίηση. Για το λόγο αυτό, στη συνέχεια, θα χρησιμοποιήσουμε τεχνικές αναγνώρισης υφής εικόνας μέσω φίλτρων με σκοπό την πιο ολοκληρωμένη πρόβλεψη.

1.4.2 Πρόβλεψη υποτύπου αδενοκαρκινώματος

Όπως αναφέρθηκε, έχει αποδειχθεί ότι οι ασθενείς που ανήκουν στον υποτύπο αδενοκαρκινώματος TRU, έχουν μεγάλη πιθανότητα να έχουν μετάλλαξη στο γονίδιο EGFR. Γι' αυτό, χρησιμοποιήσαμε την ίδια δομή του νευρωνικού δικτύου, ώστε να εξετάσουμε εάν η θεωρητική συσχέτιση των υποτύπων με ιστοπαθολογικά χαρακτηριστικά επιβεβαιώνεται και κατά πόσο αυτό επηρεάζει την ανίχνευση της μετάλλαξης του EGFR στην εικόνα.

Το αποτέλεσμα της εκπαίδευσης έδειξε ότι ο TRU υπότυπος ανιχνεύεται στα χαρακτηριστικά της εικόνας με ακρίβεια 85%, είτε εξετάζεται μόνος του είτε μαζί με τους υπόλοιπους υποτύπους. Ωστόσο, συγκρίνοντας τους ασθενείς που προβλέφθηκαν επιτυχημένα στο πρόβλημα της μετάλλαξης στο EGFR, και τους ασθενείς που προβλέφθηκαν επιτυχημένα στην κλάση TRU, βρίσκουμε ότι υπάρχει επικάλυψη αλλά όχι απόλυτη ταύτιση, το οποίο σημαίνει ότι κάποια χαρακτηριστικά που ανιχνεύονται στην περίπτωση του EGFR οφείλονται και σε ξεχωριστούς παράγοντες, μη σχετικούς με την κλάση TRU.

1.5 Αποτελέσματα Χρήσης Ψηφιακών Φίλτρων για Ανίχνευση Χαρακτηριστικών Υφής Εικόνας

Στη συνέχεια, προκειμένου να κατανοήσουμε καλύτερα τα χαρακτηριστικά που διαφοροποιούν τον ιστό που εμφανίζει EGFR μετάλλαξη από εκείνον που δεν εμφανίζει, θα εφαρμόσουμε στις εικόνες μια σειρά από ψηφιακά φίλτρα που στοχεύουν στον εντοπισμό των χαρακτηριστικών αυτών. Η διαδικασία χωρίζεται σε 3 στάδια:

1. Τμηματοποίηση Εικόνας

   Χρησιμοποιήθηκε η τεχνική της δυαδικής κατωφλίασης, η οποία ορίζει μια σταθερή τιμή κατωφλιού σε όλη την εικόνα, πάνω από την οποία τα εικονοστοιχεία λαμβάνουν την τιμή 1 (255) και κάτω από την οποία την τιμή 0 (0). Επιλέξαμε για κάθε πλακίδιο κατώφλι την τιμή 150, από το εύρος 0-255.

2. Εξαγωγή χαρακτηριστικών εικόνας με τη χρήση φίλτρων

   Στη συνέχεια, περάσαμε κάθε πλακίδιο από φίλτρα και των 5 κατηγοριών που αναφέρθηκαν. Κάθε φίλτρο στοχεύει στην ανίχνευση ξεχωριστών χαρακτηριστικών. Από τη διαδικασία αυτή προέκυψε ένα διάνυσμα 34 χαρακτηριστικών.

3. Ταξινόμηση χαρακτηριστικών εικόνας με τη χρήση επιβλεπόμενης μάθησης

   Στη συνέχεια εφαρμόσαμε αλγορίθμους επιβλεπόμενης μηχανικής μάθησης, με είσοδο το διάνυσμα των χαρακτηριστικών ανά πλακίδιο και την σήμανση εάν το πλακίδιο ανήκει σε ασθενή που εμφανίζει EGFR μετάλλαξη ή όχι, με σκοπό την ανίχνευση της διαφοροποίησης ανάμεσα στις 2 κατηγορίες.

   Ο αλγόριθμος που πέτυχε την καλύτερη ακρίβεια πρόβλεψης ήταν το δέντρο αποφάσεων, το οποίο κατάφερε να προβλέψει τις δύο ξεχωριστές κατηγορίες με ακρίβεια 91%. Παρατηρούμε ότι, η ακρίβεια πρόβλεψης χαρακτηριστικών υφής που διαφοροποιούν τις εικόνες μεταξύ των δύο κλάσεων είναι αρκετά μεγάλη, γεγονός το οποίο επιβεβαιώνει την αρχική μας πρόβλεψη.

Επομένως συμπεραίνουμε πως υπάρχουν χαρακτηριστικά που διαφοροποιούν τις ιστοπαθολογικές εικόνες και τα οποία μπορούν να προβλέψουν με αρκετά καλή ακρίβεια, την μετάλλαξη του γονιδίου EGFR μόνο από την ύπαρξη της εικόνας και χωρίς την ύπαρξη δεδομένων αλληλούχησης DNA & RNA.

Chapter 2

Introduction

At this chapter we will make a brief presentation of the biological problem we are trying to solve. We will mention some basic key knowledge about Lung Cancer, it's subtypes and disease characteristics, and also details about the work preceded, which this thesis is based on. Finally, we will give a rough structure of the thesis.

2.1 Biological Problem – Motivation

Lung Cancer is measured as the second most common cancer, and the first by far most common cancer related cause of death per year in the United States, according to the American Cancer Society [8]. It is commonly diagnosed at people after the age of 65. There are two main types of lung cancer : Non-small cell lung cancer (NSCLC), which applies to 80-85 % of the cases, and Small cell lung cancer (SCLC) which applies to the rest. The treatment process for each case is completely different.[15]

2.1.1 Non-small cell lung cancer (NCSLC)

All lung cancers that are not recognised as small cell lung cancers, are considered non-small cell lung cancer type .There are three main subtypes: Adenocarcinoma, Squamous cell (epidermoid) carcinoma and Large cell (undifferentiated) carcinoma.

- **Adenocarcinoma** : This types represents around 40% of lung cancer and is met at the early versions of the cells. It is the most common lung cancer in both smokers (current and former) and non-smokers and the most likely to be found in younger patients. Usually it is found in the outer parts of the lung and can be discovered at an early stage, before its spread. A subcategory called "adenocarcinoma in situ" presents a better outcome than the rest for most of the cases. Lung Adenocarcinoma has also been studied in terms of clusters and subtypes associated with the transcriptional profile it's patients, and three basic subtypes have been defined, which we will later analyze.

- **Squamous cell (epidermoid) carcinoma** : This type represents around 25-30% of lung cancer and is found in early stages of squamous cells, the flat cells that line the inside of the airways of the lung. Most of the time they are linked with smoking.

- **Large cell (undifferentiated) carcinoma** : This types represents around 10-15% of lung cancer. In most cases it's hard to treat as it tends to develop in any part of the lung and spreads rather quickly.

There are also a few other subtypes, such as adenosquamous carcinoma and sarcomatoid carcinoma, which are not that common.

2.1.2 Causes

Smoking is by far the most common cause of lung cancer, with 80% of the lung cancer deaths to be related to smoking. Regular cigarettes, cigars and pipe smoking all have the same lung cancer risk, as well as secondhand smoking with a lower rate.

The second most common cause is exposure to Radon, a radioactive gas that one cannot smell, taste or see. It is measured as the leading cause of lung cancer in non-smokers. Other risk factors are exposure to asbestos, arsenic in drinking water, air pollution and of course family history of lung cancer.

2.1.3 Diagnosis

A chest radiograph is the first examination one can take, which can reveal an obvious mass or an abnormality that implies the existence of cancer. Afterwards, a lung biopsy is performed in order to diagnose details around the type and stage of cancer. The sample is analyzed through histopathology, a microscopic examination of the tissue, for which we will refer later in detail. This produces histological slides that can be analyzed by doctors in order to make useful observations.

Information such as cancer type can nowadays be discovered by doctors only by looking at the picture. However, the of detection other information such as driver mutations are nowadays of high priority as they seem to play a crucial role in applying targeted therapies which prove to be in many cases far more successful than the standard empirical chemotherapy techniques. [16].

Digital Image Recognition of histopathology slides is a necessary tool that can be used in detecting such patters, as it is highly likely that some of them can be recognized just by looking at the texture of the image. That means that there is some texture feature that distinguishes cases from one another, but it is too complicated for the human eye yet to see.

2.2 Previous Work – TCGA Project Cancer Genome Atlas

This thesis is based on the work presented at ”Classification and Mutation Prediction from Non-Small Cell Lung Cancer Histopathology Images using Deep Learning” [7], published at 2018.

The histopathological images analysis of lung cancer cell tissue is one of the basic methods used by physicians to assess the stage, type and subtypes of lung cancer. The two most prevalent subtypes of lung cancer, which belong to the type of non-small cell lung cancer, are adenocarcinoma and squamous carcinoma.

From the work that preceded we received a deep neural network which, given a histopathological image and a table of attributes corresponding to the particular image, all obtained from the TCGA database, was successfully trained in 2 different pattern recognition problems:

1. Cancer type recognition: adenocarcinoma, squamous cell carcinoma, normal tissue. Success rate 0.97 AUC (Area Under Curve).

2. Prediction of the 10 most common mutations in adenocarcinoma. Six of these gene mutations - STK11, EGFR, FAT1, SETBP1, KRAS, TP53 - can be predicted with an accuracy of 0.733-0.856 AUC.

In this study we focused on the second result and tried to identify morphological features of the lung tissue that are unknown or not officially recorded and express transcriptional changes and mutations.

In detail, Epidermal growth factor receptor (EGFR) a transmembrane protein was found associated with many lung cancer cases and two targeted treatments have been discovered and put to use[11]. We will study the existence of morphological differentiation of the tissue in the case of a mutation in the EGFR gene in two different ways: Using a Deep Neural Network and using filters to study texture and extraction of image characteristics.

2.2.1 TCGA Project Cancer Genome Atlas

The Cancer Genome Atlas (TCGA) [17], a collaboration between the National Cancer Institute (NCI) and National Human Genome Research Institute (NHGRI), has generated comprehensive, multi- dimensional maps of the key genomic changes in 33 types of cancer. The TCGA dataset, containing 2.5 petabytes of data describing tumor tissue and matched normal tissues from more than 11,000 patients, is publically available and has been used widely by the research community. The data have contributed to more than a thousand studies of cancer by independent researchers and to the TCGA research network publications.

TCGA created a genomic data analysis pipeline that can effectively collect, select, and analyze human tissues for genomic alterations on a very large scale. For this purpose also, the GDC Data Portal was created, as a data platform that provides cancer researchers and bioinformaticians the opportunity to easily search and download cancer data for analysis [6].

2.3 Thesis Structure

The following Diploma Thesis consists of 7 chapters :

1. Chapter 1: Extensive summary of the subject of this thesis and it's results, in greek.
2. Chapter 2: Introduction to the Biological Problem and Engineering Solutions. A small presentation of the Biological Problem is made, as well as a reference to the previous work that this thesis was based on.
3. Chapter 3: Thorough Presentation of the Problem and the Data used. A more detailed analysis of the biological terms as well as the type of data used is made.
4. Chapter 4: Presentation of the theoretical background used for our Digital Image Filtering and Machine Learning applications and Classification as well as evaluation methods.
5. Chapter 5: Presentation of the data-preprocessing methods and implementation of the computational methods used
6. Chapter 6: Presentation of the results from both techniques, in words and visuals.
7. Chapter 7: Summary of results and presentation of future work suggested based on our results.

Chapter 3

Biological Background - Data Description

This chapter focuses on the biological background information needed for this thesis,in order to understand the biological data used as well as some important terms.

3.1 Biological Background

Following, is a small presentation of some basic biological terms necessary for understanding in depth the object of this thesis.

3.1.1 Gene Mutation

A gene mutation is a permanent alteration of a part of a DNA sequence that represents a gene. A different version of the same gene is created and is called allele. They are classified in two major categories :

1. Hereditary mutations : They are the mutations inherited from the parents and exist in ever cell of the body throughout the whole life of the person. If the DNA of a parent has a mutation, the resulting fertilized egg will receive this mutation and carry it in each of its cells.

2. Acquired (or somatic) mutations : These mutations are caused at some point in a person's life, due to environmental parameters or errors during DNA copy creation of cell division. They exist only in certain cells of the body and if it happens in somatic cells (all cells expect sperm and egg cells), it cannot be passed from one generation to another.

At a DNA Level, the main types of DNA changes are [9]:

1. Silent mutation: The mutation changes one codon (a sequence of three DNA or RNA nucleotides that corresponds with a specific amino acid) for an amino acid into another codon for that same amino acid.

2. Missense mutation: The codon for one amino acid is replaced by a codon for another amino acid.

3. Nonsense mutation: The codon for one amino acid is replaced by a translation termination (stop) codon.

4. Frameshift mutation: Single base additions or deletion that cause a shift in the DNA polymerase and creates a far different protein than the original one.

5. Chromosome Rearrangements : A chromosome represents the way the day is structured within the cell. Changes such as deletion or replication of a part can lead to changes for multiple genes.

The following picture best describes the different main types of mutations at DNA level:

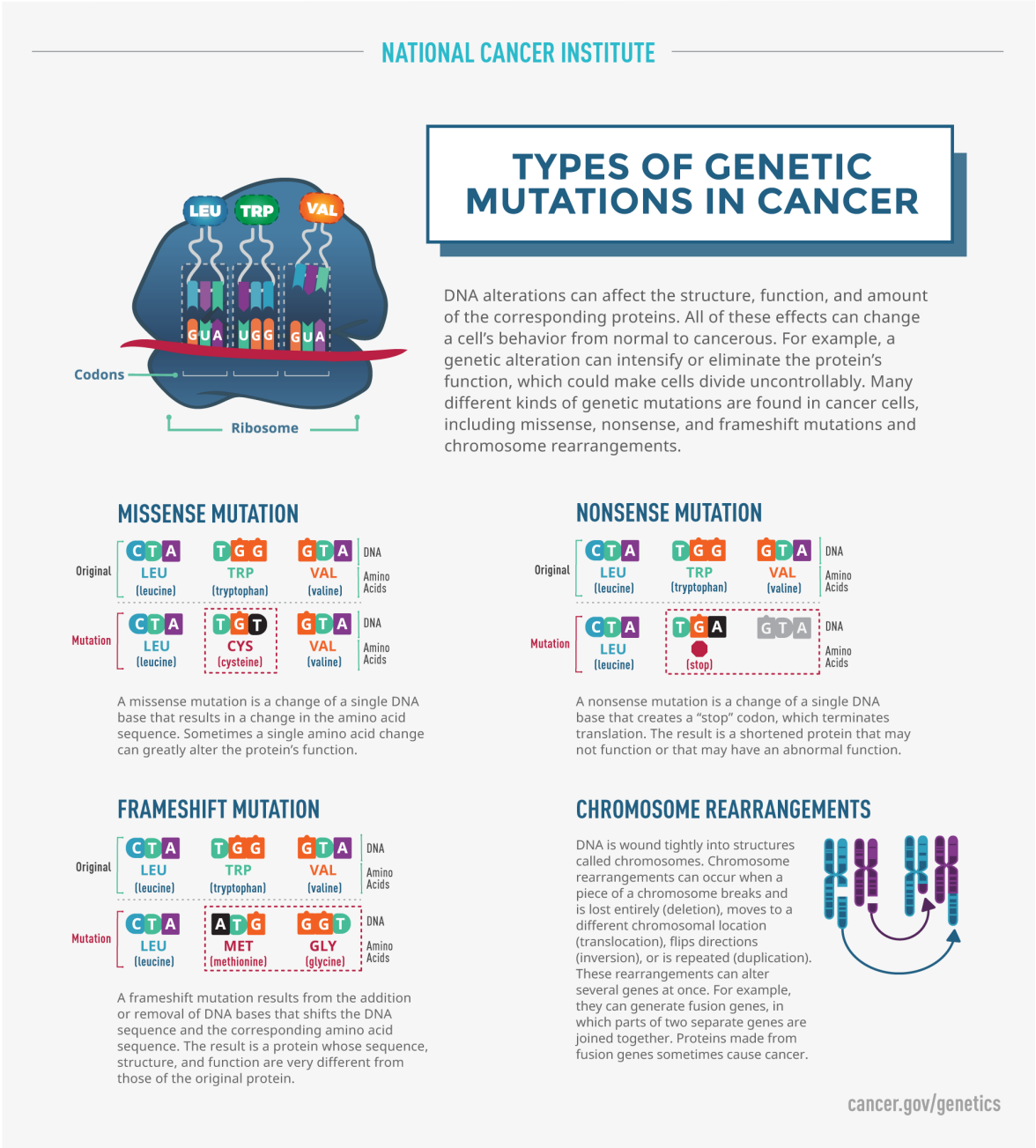


Figure 3.1: Types of Genetic Mutations in Cancer, from cancer.gov/genetics

3.1.2 Histopathology Images

Pathology Images are thin slices of tissue acquired from surgery or biopsy, that are cut from a sample and then put under a microscope for examination from pathologists, after preparing and staining them accordingly.

Staining is used in order to visualize cellular components within the image. The most popular form of staining developed and used for more than 100 years is Hematoxylin-Eosin staining. H&E Staining is essential in recognizing tissue types and the morphologic changes that assist in modern cancer diagnosis. Hematoxylin stains cell nucleic acids blue, while Eosin stains cytoplasm and connective tissue pink. That happens because hematoxylin is basic/positive and therefore it binds to basophilic substances such as DNA & RNA, whereas eosin is acidic / negative and therefore it binds to acidophilic substances such as positively charged amino axic chains. This procedure makes it easier for the pathologists to recognize patterns and complete the tissue diagnosis.[18]

Afterwards, the two dimensional slides are analyzed by doctors in order to make conclusions about the type and stage of cancer as well as other information. It is a procedure that requires high accuracy and experience. Nowadays, these slides are digitized and stored in a digital form. However, they usually have a very large size (10,000 to over 100,000 pixels in each dimension) and can be very noisy in the amount of information they contain, which makes analyzing them a rather challenging task for the pathologists. [10]

During the last few years, the dramatic advances in computational resources and digital image analysis algorithms have created new tools which contribute to a computer-assisted diagnosis. Computational analysis of such detailed images can assist pathologist in making faster, more precise and detailed diagnosis. Moreover, it plays a key role in detecting features and patterns in such images that are not yet distinguishable by the human eye. Such patters may prove groundbreaking for patient-targeted treatments, drug discoveries and new era personalized medicine. [19]

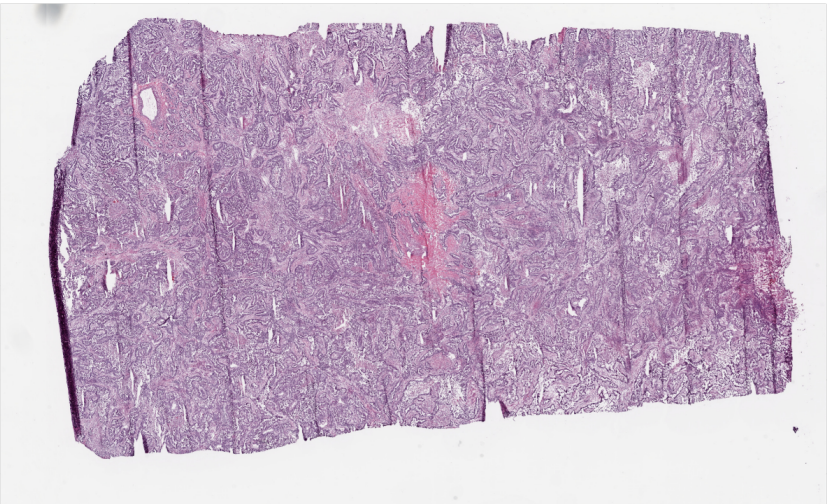


Figure 3.2: Sample of a Lung Adenocarcinoma Histopathology Image

3.1.3 Sequencing Data

Sequencing is used in genetics and biochemistry, in order to determine the primary structure of an unbranched biopolymer (DNA, RNA, protein, carbohydrate). After sequencing is performed, a sequence (symbolic linear depiction) is created which contains and summarizes much of the molecule's atomic-level structure.

3.1.4 DNA Sequencing

DNA Sequencing defines the process of exploring the nucleotide order of a given DNA fragment. The methods used aim in determining the four bases -adenine, guanine, cytosine, and thymine— in a strand of DNA.

The sequencing of an entire genome is a complicated process that includes sequencing smaller pieces of DNA and combining them in a single sequence. There are various, older and newer techniques used. Two of the most used methods are Sanger sequencing and Shotgun sequencing. In both of them, the sequencing of overlapping fragments is used in order to assemble larger regions of DNA and, eventually, entire chromosomes.

The most recent technologies used for sequencing are called Next Generation Sequencing. What differentiates them from the older techniques are the following :

1. Highly parallel: many sequencing reactions take place at the same time
2. Micro scale: reactions are tiny and many can be done at once on a chip
3. Fast: because reactions are done in parallel, results are ready much faster
4. Low-cost: sequencing a genome is cheaper than with Sanger sequencing
5. Shorter length: reads typically range from 50 -700 nucleotides in length

The DNA sequence contains all the information needed for living beings to survive and reproduce. Discovering the sequence and understanding the information it encodes is crucial in understanding the way organisms function. Sequencing research is becoming more and more important, as it is highly connected with discoveries in areas such as drug discovery, diagnosis and treatment of currently unmapped aspects of diseases, biotechnology evolutions and many more. [12]

3.1.5 RNA Sequencing

Since RNA is produced through the transcription of DNA, the information we can extract from RNA is similar to the one extracted from DNA. However RNA sequencing can give us valuable information for the status of the examined tissue, as it will reveal the sequences that are actively expressed in the cells. Nowadays, RNA-Seq uses recently developed deep-sequencing technologies and is widely used for gene expression detection. [20]

3.1.6 Gene Expression

The DNA stored in the nucleus of the cell, is split in smaller sequences which are called genes. Genes dictate certain functions of the cell by encoding the protein that orders each function. Gene expression is the process of transcription of the DNA gene code into RNA code in order to start the production of the ordered protein and therefore complete a certain action. It is measured as the number of RNA copies that are produced inside the cell, which lead to the production of the encoded protein accordingly. [13]

Gene expression is the process by which the genetic code - the nucleotide sequence - of a gene is used to direct protein synthesis and produce the structures of the cell.

Measuring Gene expression is an important process at it can provide us with valuable information about the function of the cell and most importantly, if it is normal or not. Sequencing technology - RNA Seq allows the calculation of the exact number of RNA arrays in a sample, whereas another technique such as the DNA microarray [21], calculates the number of genes in many samples.

For the purpose of this thesis, we used data of transcriptomic profiling of RNA-Seq-based expression extraction with the use of the HTSeq framework [22], for patients of LUAD adenocarcinoma, provided publicly by the Genomic Data Commons Data Portal .

3.1.7 Molecular subtypes of lung adenocarcinoma

Transcriptional analysis and profiling can lead to important discoveries such as the changes caused by driver mutations or provide information for tumors classified according to their profile.

Several studies have worked on lung adenocarcinoma transcriptional clustering, creating three (3) clusters,which defined transcriptional subtypes, beased on tumor morphological features, named, respectively, after the focus of each research.[23, 24, 25, 26]. Eric A. Collisson et al [14] suggested the following naming for the three clusters, taking into account histopathological [27], anatomic and mutational load classifications :

1. Terminal respiratory unit (TRU, formerly bronchioid) subtype
2. Proximal-inflammatory (PI, formerly squamoid) subtype
3. Proximal-proliferative (PP, formerly magnoid) subtype

For the above three clusters, there have been many studies which revealed associations between them and clinical outcomes and genomic alterations. For example. the PP subtype is found enriched in the KRAS gene mutation and depleted at STK11 tumor suppressor gene. The PI subtype was characterized by solid histopathology and co-mutation of NF1 and TP53. Finally, the TRU subtype was found correlated with the majority of the EGFR-mutated tumors. As the clusters are derived from histopathological features, TRU enrichment in the EGFR gene [28] gives us a clue and a motive to further investigate the presence of the EGFR mutation on tissue slides' characteristics.

3.1.8 EGFR

Epidermal growth factor receptor (EGFR) is a transmembrane protein with cytoplasmic kinase activity that transduces important growth factor signaling from the extracellular milieu to the cell.

It had been found that the EGFR gene is expressed in more than 60% of non small cell lung cancers, which makes it an ideal study factor for tumor targeted treatment. Such a treatment has been developed and approved and are based on Inhibitors that target the kinase domain of EGFR. Specifically, Tyrosine kinase inhibitors (TKIs) have a high response rate in adenocarcinoma cases. [11]

3.2 Dataset Description

3.2.1 Image Data (image extraction - staining - colors)

In this study we used whole slide images of hematoxylin and eosin stained histopathology images, as described above, from the TCGA Dataset, obtained by excision. The images corresponded to LUAD lung cancer patients and were obtained from the Cancer Digital Slide Archive [29]

3.2.2 Sequencing Data

Our dataset comes from the NCI Genomic Data Commons [30] which provides the research community with an online platform for uploading, searching, viewing and downloading cancer-related data.

We downloaded sequencing data containing information about gene mutations for a total of 361 patients from the LUAD TCGA database. We used the below selected filters :

- Project: TCGA LUAD
- Data Category: Simple Nucleotide Variation
- Data Type: Masked Somatic Mutation

We also downloaded Transcriptomic data of 374 patients, which contained information about gene expression per patient.

In the next chapters, we will further analyze the processing and use of this data.

Chapter 4

Theoretical Background

The purpose of this chapter is to briefly present the technical background of the techniques and methods used for the purpose of this thesis. A reference is made both on Image Digital Filtering techniques as well as Machine learning techniques. Evaluation methods used are also analyzed.

4.1 Image Digital Texture Filtering

In computer science, Digital Image Filtering is the processing of images with the use of algorithms made to detect texture characteristics within the image. Image texture is prescribed as an image obeying some statistical properties. These techniques can be extremely useful as, apart from information like Edge Detection, which is recognizable from the human eye, they can help us draw conclusions about important image features, which the human eye would find difficult to recognize.

A standard process of image texture recognition is the following :

1. Image Segmentation : A technique used to reduce image noise and focus on important texture information
2. Image Filtering : The core of the process. The image is passed as a vector through filters that extract the texture information, also as a vector.
3. Texture characteristics Classification : Classification and/or clustering algorithms can then be applied to the results from the filters, with the purpose of defining texture attributes and drawing conclusions.

Following, the above process will be presented in detail.

4.1.1 Digital Image Segmentation

Image Segmentation Techniques are used in order to separate the image in areas of interest (foreground) and background. Information such as histograms, pixel intensity and area similarity are important for defining the image segments. Following, the technique Thresholding is presented, which was used at this project.

Thresholding

Thresholding is the process in which one or more thresholds of brightness intensity are set as a criterion for categorizing each pixel according to their value of brightness (higher or lower) [31]. Thresholding is mostly used in cases where the area of interest has a clear brightness difference from the rest of the image and a relative uniformity. Every object of the image that has a relatively big size creates a pixel distribution near the average value at the image histogram. There are 2 categories of thresholding : Global and Adaptive.

In Global Thresholding, a constant value T is used as a threshold and applied to all pixels. Any pixel value below T is given a zero value, or one otherwise. Global thresholding is the most common and fastest techniques of image segmentation and is normally used to get rid of small perturbations in the image before processing. However, it is not recommended for more complex images with more than one objects of interest, or objects with non-uniform brightness. Some of the most common subtypes are Binary Threshold, Inverse Binary Threshold, Truncate Threshold, To Zero Threshold, Inverse to Zero Threshold.

In Adaptive Thresholding, the threshold T is not the same for the whole image, but depends from the brightness of each area. At the beginning, a constant value is set as a starting point and then from each pixel, the value of their brightness as it derives from a filter used to blur the image, is added. Therefore, in the brighter areas higher threshold is used, whereas in the darker areas that value is much lower. This techniques are mostly successful in images with higher noise. Some of the most common subtypes are Otsu Method, Mixture Models and Multispectral Thresholding.

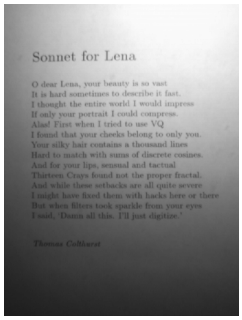
(a) original image

(b) low threshold

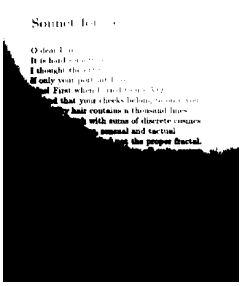
(c) correct threshold

(d) high threshold

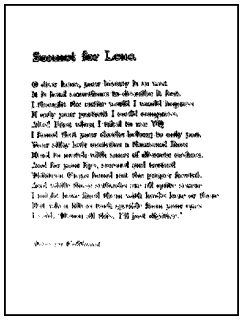
Figure 4.1: Example of Global Thresholding.



(a) original image



(b) Binary Thresholding



(c) Adaptive Thresholding

Figure 4.2: Example of Adaptive Thresholding.

4.1.2 Digital Filters

In Image Digital Filtering, an image vector is passed as input to linear or non linear filters, with the goal of producing a vector that uniquely describes the texture details of each image segment as output. This vector can be later used as input in classification and/or clustering algorithms which complete the texture detection process.

In detail, the filter is a mathematical model that generates output values, depending on the values assigned to a set of Kernels (small matrix). Each kernel (window) is placed around each pixel and makes a mathematical calculation (convolution), taking into account the neighborhood area around the pixel and stores the result in pixel position in a new ”image” that is a ”bulk” response of the filter. This is repeated for all pixels in the original image. We can either sum up these results for a more general conclusion for the image, or keep local values and make conclusions about specific areas. [32]

There are three types of approaches [33]:

1. Structural approach: A texture is a set of texture elements (size, orientation, randomness), or texels, occurring in some regular or repeated pattern
2. Statistical approach : Characterize texture using statistical measures computed from grayscale intensities (or colors) alone. A technique most commonly used, as it can be applied to all kinds of images, with low computational cost (ex. Edge Detection).
3. Fourier approach : Texture detection algorithms, based on the Fourier transformation.

During this thesis, we implemented and used the following texture filtering algorithms:

- Haralick

  Haralick filters [34] [35] are based on the process of defining texture characteristics that are interpretable by the human eye. Therefore, the image texture is considered a quantification of the spatial variation of grey tone values. In 1973, Haralick et al. introduced the use of gray level co-occurrence matrices (GLCM). Tone and Texture have a co-depended relationship. When an area has a small variation of features of discrete grey tone, then tone dominates this area. Respectively, when the area has a wide variation of grey tone features, texture dominates this area.

  The Haralick texture recognition method is based on the joint probability distributions in pairs of pixels. GLCM contain the frequency of each gray level at a pixel which its location is fixed relative to it’s neighbor pixels. Each resolution cell has 8 neighbor cells, as shown at fig 4.3. Each neighbor defines different matrices according to the angle of difference and the distance from the center cell.

  Some of the basic features extracted from Haralick filters (with $g(i, j)$ as the element in cell i, j of a a normalized GLCM) are :

  - Energy/Angular Second Moment: $f_1 = \sum_{i,j} g(i, j)^2$
  - Entropy: $f_2 = -\sum_{i,j} g(i, j) \log_2 g(i, j)$, or 0 if $g(i, j) = 0$
  - Correlation: $f_3 = \sum_{i,j} \frac{(i-\mu)(j-\mu)g(i,j)}{\sigma^2}$
  - Inverse Difference Moment: $f_4 = \sum_{i,j} \frac{1}{1+(i-j)^2} g(i, j)$
  - Inertia: $f_5 = \sum_{i,j}(i - j)^2 g(i, j)$
  - Cluster Shade: $f_6 = \sum_{i,j}((i - \mu) + (j - \mu))^3 g(i, j)$
  - Cluster Prominence: $f_7 = \sum_{i,j}((i - \mu) + (j - \mu))^4 g(i, j)$

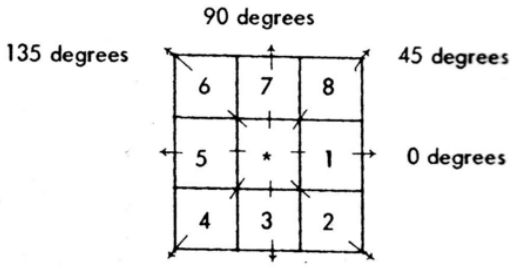


Figure 4.3: 3x3 window definition and spatial relationship for calculating Haralick texture measures. Pixel 1 and 5 are 0° (horizontal) nearest neighbors to the center pixel; pixel 2 and 6 are 135° nearest neighbors; pixels 3 and 7 are 90° nearest neighbors, pixel 4 and 8 are 45° nearest neighbors to the center pixel

  - Haralick's Correlation : $f_8 = \frac{\sum_{i,j}(i,j)g(i,j) - \mu_t^2}{\sigma_t^2}$ where μ_t and σ_t are the mean and standard deviation of the row (or column, due to symmetry) sums.

Also, $\mu =$ (weighted pixel average) $= \sum_{i,j} i \cdot g(i,j) = \sum_{i,j} j \cdot g(i,j)$ (due to matrix summetry), and

$\sigma =$ (weighted pixel variance) $= \sum_{i,j}(i-\mu)^2 \cdot g(i,j) = \sum_{i,j}(j-\mu)^2 \cdot g(i,j)$ (due to matrix summetry)

Haralick filters are one of the most common texture recognition filters, as they can be used in a wide variety of images.

- Wavelets Filter

  Wavelets are defined as mathematical functions that localize both on a time and frequency basis. Wavelets have varying frequency, limited duration and zero average value. An example is shown at figure 4.4, in comparison with a Sinusoid function. The wavelet transformation is a time-frequency transformation, with the use of a wavelet's function. It is similar to the Fourier transformation, with the difference that Fourier cannot localize in time and frequency at the same time. A wavelet transformation can be either discrete or continuous. A discrete wavelet transform (DWT) is any wavelet transform for which the wavelets are discretely sampled.

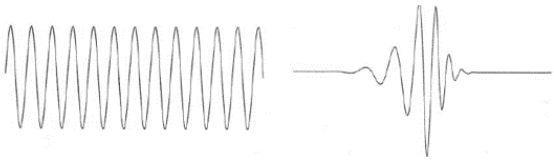


Figure 4.4: Left : Sinusoid signal. Right : Wavelet signal

There are many types of wavelet functions. Mathematically, wavelets are defined as following:

  - Base functions [36] [37]

$$f(t) = \sum_{jk} c_{jk}\psi_{jk}(t)$$

ψ_{jk} functions are a set of base functions defined for wavelets, such as sin() and cos() are for the Fourier function. The span of ψ_{jk} functions is the vector space S which contains all of the functions f(t) that can be represented by ψ_{jk}.

We can construct the base function by applying translations and scalings to the "mother wavelet" ψ_{jk}.

$$\psi(s,\tau,t) = \frac{1}{\sqrt{s}}\psi(\frac{t-\tau}{s})$$

If we define the wavelet basis as $s = 2^{-j}$ and $\tau = k * 2^{-j}$, then :

$$\psi(s,\tau,t) = 2^{\frac{j}{2}}\psi(2^j t - k) = \psi_{jk}(t)$$

The coefficients $c_(jk)$ are given by $c_(jk) = [W_\psi f](2^{-j}, k2{-}j)$.

– Digital Transformation

In the Digital Wavelets Transformation, the samples are first passed through a low pass filter (convolutional computation) and simultaneously, the signal is decomposed by a high pass filter. The approximation coefficients are the lowpass representation of the signal and the details are the wavelet coefficients. At each subsequent level, the approximation coefficients are divided into a coarser approximation (lowpass) and highpass (detail) part. The detail coefficient implies the edges of the image, therefore the characterization. [38]

For the purpose of this thesis we implemented the Wavelet transformation with the use of PyWavelets[39]

- Gabor Filter

Gabor filters are widely used in areas such as texture analysis, edge detection, feature extraction etc. They are special classes of bandpass (linear) filters, which means they keep a certain 'band' of frequencies and reject the others.

Gabor filters function in a similar way as the one we previously described for conventional filters. A convolution kernel, which is an array of pixels with an assigned weight is the core of this filter. The array is then slid over each pixel and performs a convolution operation. A Gabor filter responds to edges and texture changes. which means that the filter has a distinguishing value at the spatial location of that feature. [40]

The mathematical representation of the filter is :

$$g(x,y;\lambda,\theta,\phi,\gamma) = \exp\left(-\frac{x'^2+\gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda}+\phi\right)$$

where:

$$x' = x\cos\theta + y\sin\theta$$

$$y' = -x\sin\theta + y\cos\theta$$

In this equation, λ defines the Wavelength, i.e. the number of cycles/pixel, θ defines the orientation, i.e. the angle of the normal to the sinusoid, ϕ is the phase offset of the sinusoid, $gamma < 1$ is the spatial aspect ratio and σ is the standard deviation of the Gaussian function used in the Gabor filter.

Optimizing the parameters above can lead to correct and multiple feature extraction (figure 3.3 4.5)

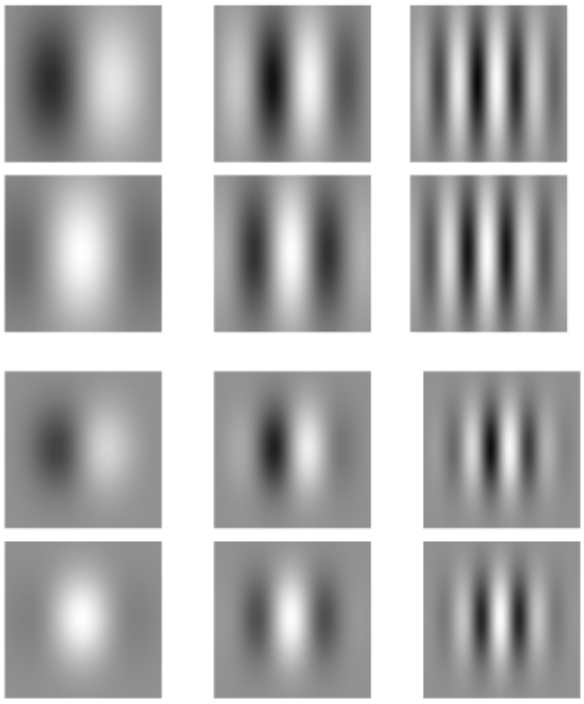


Figure 4.5: Gabor filters with different combinations of σ, θ and ϕ

- Local Binary Patterns

  The Local Binary Patterns (LBP) filter is a simple but highly efficient texture operator. It characterizes each pixel of an image, by thresholding the neighborhood of the pixel, and converting the result to a binary number. Local Binary Patterns are based on the Texture Spectrum model from 1990 [41] [42].

  The concept of function of a LBP filter is the following :

  1. The examined window is divided into cells (for ex. 16x16 pixels per cell).
  2. Each pixel of the cell is compared with it's 8 neighbor cells along a circle.
  3. If the examined pixel's value is higher than it's neighbor, that corresponds to a value of 0, or 1 otherwise. In that way, an 8 digit binary number is formed, which uniquely describes the cell. An example of such calculation is shown at figure 4.6.
  4. A histogram of the frequency of each number occurring is then produced for the cell.
  5. All histograms from all the cells are concatenated and produce a feature vector representing the entire window.

  The produced feature vector can afterwards be used by classification algorithms, in order to perform texture analysis or computer vision tasks. At figure 4.6 an example of an LBP calculation is shown.

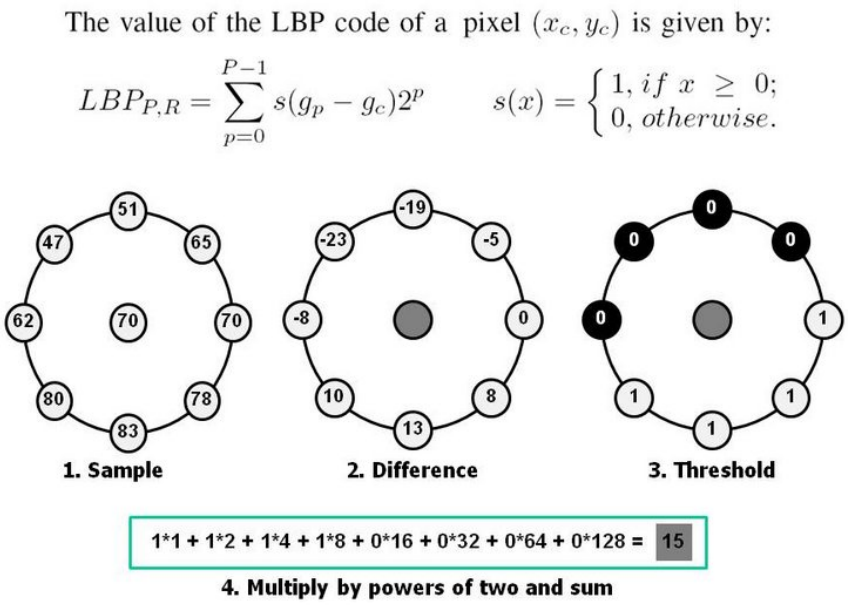


Figure 4.6: Example of LBP calculation

- Fractal Dimensions

  In order to understand the Fractal dimensions, we first need to understand the different meaning of the word 'dimension' for this occasion. Specifically, irregular geometric objects (fractals) use the Hausdorff Dimension.[43]

  Let's start with an object with a Euclidean dimension D. If we reduce its liner size by 1/r, in each spatial direction, it's measure would equal now $N = r^D$. An example is shown at figure 4.7

  If we take the log of the above equation, we have $\log(N) = Dlog(r) => D = log(N)/log(r)$. The difference now is that D can also equal a fraction and not only and integer as in the Euclidean distance.

  In terms of texture recognition, calculating the fractal dimensions of an image can help us understand how compact an object is, i.e. how wrinkled or complex a surface is. A method used to calculate fractal dimensions over images is box counting:

  In box counting, we cover the item with many boxes of a specific size and then count the number of boxes needed to fully cover the item. We test the above with various box sizes. Scaling the number of boxes with the size gives us the estimation of the item's fractal dimension.

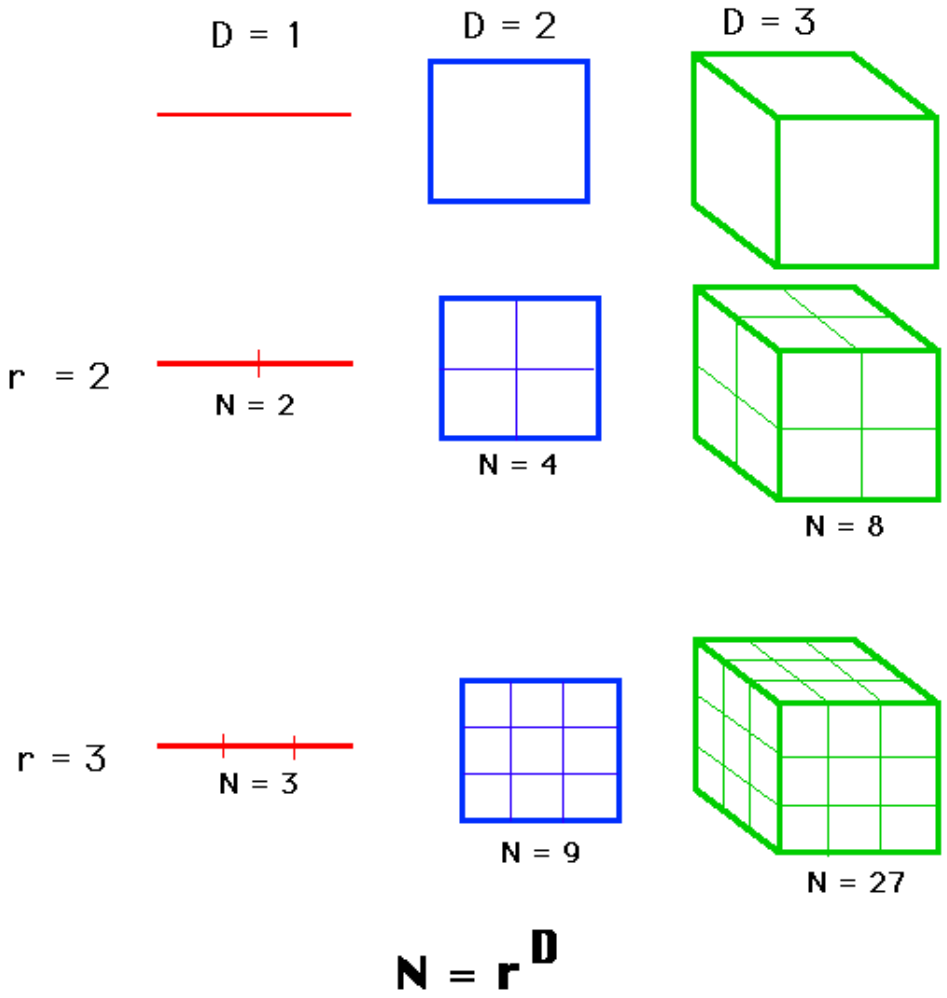


Figure 4.7: Example of conversion from Euclidean to Hausdorff dimension

4.2 Machine Learning

Machine Learning algorithms are used to predict and categorize a data set over a known or unknown number of classes. They achieve that by either learning the relationship between the feature variables and the target variables of the dataset, or by creating their own set of classes which achieves the best possible accuracy of prediction. The accuracy is tested upon a test set, the target variables of which the algorithm is going to predict. There are 2 main types of machine learning algorithms :

- Supervised Learning Algorithms, also known as Classification Algorithms
- Unsupervised Learning Algorithms also known as Clustering Algorithms

In supervised learning algorithms, the classifier receives as input a set with known labels for all the data and tries to calibrate it's parameters as best as possible, in order to achieve the highest possible success rate of predicting the labels. A dataset with unknown labels - for the classifier - is used later on to test that rate.

In unsupervised learning algorithms / clustering, the algorithm receives only the feature variables and, according to some criteria, it tries to create clusters of features with the same attributes. The key difference is that, since there are no known classes beforehand, there is no straightforward way to evaluate the accuracy of the clustering and test whether it's the best possible combination. [44]

Following, we will present some of the most important supervised classification algorithms.

4.2.1 Artificial Neural Networks

According to the inventor of one of the first neurocomputers, Dr. Robert Hecht-Nielsen, an Artificial Neural Network(ANN) is computing system made up of a number of simple, highly interconnected processing elements, which process information by their dynamic state response to external inputs. The architecture of the ANN is inspired by the biological neural networks of the nervous system. Their use is to analyze and detect patterns and features inside the given input data, which might be hard or impossible for the humans to explore. They do that by "learning" how to analyze the data and creating rules that apply to them, rather than being taught specific instruction to solve the problem. This is also however the sensitive spot of this technology, as its operation and the way it decides to solve a problem can be unpredictable.

A basic Neuron - A perceptron

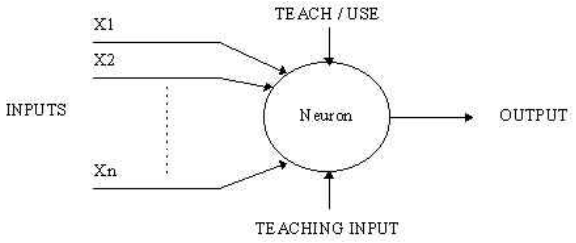


Figure 4.8: Representation of a basic Neuron.

As you can see at figure 4.8, a basic neuron, also called a perceptron, is a device with multiple inputs and one output. Let's assume this is neuron j. It receives an input $p_j(t)$,where t is a discrete time parameter, from the previous neurons that are connected to it. The key attributes of the neuron are the following :

1. An activation function $a_j t$
2. A threshold θ_j
3. An activation function f. f computes the new activation at a given time as $a_j(t+1) = f(a_j(t), \theta_j, p_j(t))$
4. An output function $o_j(t) = f_{out}(a_j(t))$

Each connection between the neurons j and its previous neurons i has a weight w_{ji} assigned to it. The input of the neuron is then computed as $\sum w_{ij} o_i$. The weight describes how much each input affects the output of the neuron. In other words, it describes the impact of the feature that the previous neuron detects in the feature that the current neuron is attempting to detect. The adjustment of these weights is part of the learning process and defined by deviation of the current output from the target output of the network.[45]

The activation function that computes the relationship between the input and the output of the neuron belongs to one of the following three categories :

1. Linear (or ramp), where the output is proportional to the total weighted output
2. Threshold, where the output is binary between 2 specified values, depending on the input value and whether it is greater of less than a threshold value.
3. Sigmoid, where the output depends on the input in a non linear way.

[46]

A Neural Network

A neural network consists of multiple neurons connected in multiple layers.[47] There are 3 layer categories :

1. The input layer. The neurons receive as input the raw input data without weights
2. The hidden layers. A Neural Network can have one or more hidden layers of neurons. These neurons function as described above.
3. The output layer. The neurons receive as input the output of the hidden layers' neurons. They might differ from the rest of the neurons, depending on the way we want the data output presented.

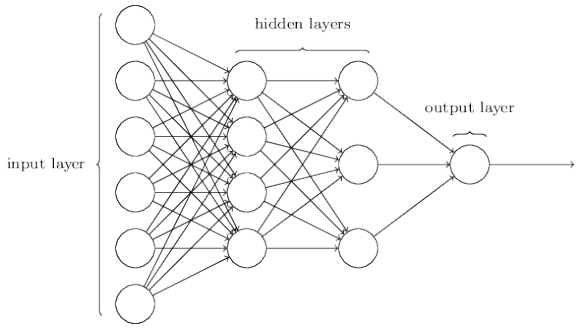


Figure 4.9: Representation of a basic Neural Network. Source : www.neuralnetworksanddeeplearning.com

The network is trained for a N number of epochs (cycles of trainings).

When each prediction is made, the ANN uses a loss function to calculate the error of the prediction. The most simple loss function is :

$$\text{loss function} = \|\text{true label} - \text{predicted label}\|$$

The most commonly used loss function is the mean squared error:

$$E = \frac{1}{P}\sum p = 1p\|d^p - y^p\|^2$$

,where y is the output and d is the true label.

The goal of the training is to adjust the weights of the network in order to minimize the loss function. An important algorithm for this purpose is the Back Propagation Algorithm. Back propagation is the method used to distribute the error of each prediction back to the neurons of the network and adjust the weights accordingly. It is basically a core function that supports the learning of the network.

In order to define the change of the weights, every time we calculate the derivate of the loss function for a small change ∂W of the weights. The goal is then to minimize the loss derivate. For example, a positive loss value with a prediction smaller that the true label will produce a negative derivative, which means that an increase in the weights is needed in order to approach the local minimum. In math, this is written as follows:

$$\frac{dw_{ij}}{dt} = -\frac{\partial E}{\partial W_{ij}}$$

For the evaluation of the previous equation, we set a new metric for each neuron, the delta δ metric:

$$\delta_i = -\frac{\partial E}{\partial U}$$

where u is the output of each neuron, calculated as $u = w^T x$ with x being the input vector and w the weight vector of the neuron.

With the use of the δ equation, we can now calculate the derivate as :

$$\frac{dw_{ij}}{dt} = \frac{\partial E}{\partial u_i}\frac{\partial u_i}{\partial w_{ij}} = -\delta_i \frac{\partial u_i}{\partial w_{ij}}$$

Now the second derivate is easily calculated as $\alpha_j^k(l-1)$ for $j \neq 0$ and 1 otherwise.

The δ function :

- At the output layer :

$$\delta_i(L) = (d_i - y_i)f'(u_k(L))$$

- At any other previous layer:

$$\delta_i(L) = \sum_{\mu=1}^{N(L+1)} \delta_\mu(L+1)w_{\mu i}f'(u_i(l))$$

  Therefore, the error for each layer L depends from the error of the next layer L+1, which means that the error is transfered backwards throughout the network

The final equation for updating the weights between the i and j neurons is :

$$w_{ij}(l, k+1) = w_{ij}(l, k) + \beta\delta_i^k\alpha_j^k(l-1)$$

β is a parameter called learning step and sets the rate at which the weights will change. Finding the correct balance between a large or small value of β is crucial for the performance of the training, as it defines the convergence of the algorithm.

The back propagation algorithm stops when the error reaches a certain minimum threshold set from the beginning of the training.

Learning Example - Summarization

Let's take the example of recognizing a data set of black and white handwritten digits. The size of input data and therefore the number of neurons of the input layer is 256, each corresponding to a small section of the image and it's color tone. The output layer consists of 10 neurons, one for each digit and there is an X number of hidden layers.

For the training of the network, the weights are initialized randomly and images are fed to the neural network, together with the correct classification label of the image. When the image goes through the network, the output of the neurons is used as an input to an error function which calculates the difference between the current output and the desired output. Afterwards, the weights' values are adjusted according to the error function output and the next images are given as input, until the error function is minimized.

4.2.2 Decision Trees (DT)

Decision Tree learning is one of the most popular supervised classification algorithms. A decision tree is a tree where each node represents a feature(attribute), each link(branch) represents a decision(rule) and each leaf represents an outcome(categorical or continues value). In other words, it represents a model that implements decision making rules in features extracted by the input data, in order to predict as accurately as possible the target value. There are two types of Decision Tree models: Classification Trees, where the target variables take discrete values, and Regression Trees, where the target variables

take continuous values. The term Classification And Regression Tree (CART) was introduced in 1984 by L. Breiman [48] and is used to describe both categories.

A plain example is shown at figure 4.10. This DT uses data that describe passengers from the Titanic and tries to determine whether a patient has a higher change of being a survivor or not. The three features taken into account for this decision are sex, age and sibsp (number of spouses or children aboard). According to this graph, one had a good chance of surviving if they were a female, or a male younger than 9.5 years with less than 2.5 siblings.

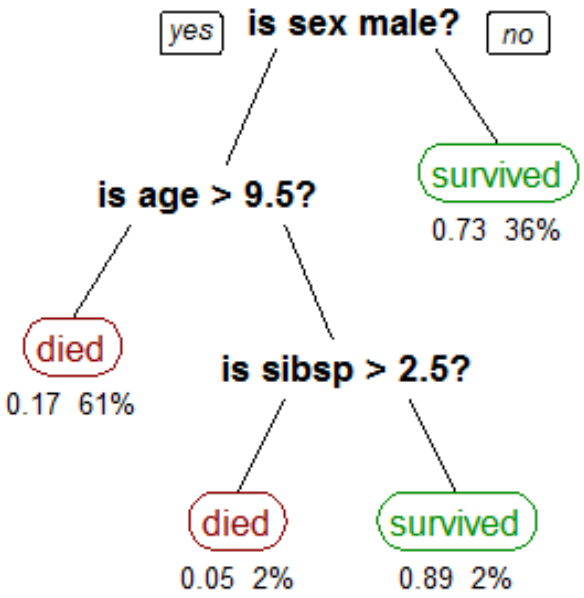


Figure 4.10: Representation of a basic Decision Tree. Source : www.wikipedia.com

There are a few metrics used to determine the creation and split of a DT. Two of them are the Gini impurity and the Information Gain.

The Gini Impurity metric is used in CART algorithms and is based on minimizing the impurity of the DT. In other words, Gini Impurity describes the probability of a randomly chosen element from the dataset to be misclassified, if a random label was assigned to it according to the distribution of labels in the subset. If p_i is the presence of same class inputs inside a particular group, the Gini Impurity is calculated as :

$$G = \sum p_i * \sum (1 - p_i)$$

The goal of the algorithm is the minimization of G. A perfect classification can be achieved only when a group contains a class completely, which means p_i is 1 or 0, and therefore G equals zero. The worst scenario is when a a 50–50 split of classes appears in a group and then p_i and therefore G equals 0.5.[49].

Information Gain metrics uses entropy in order to determine the impurity of the classification. Entropy is defined as :

$$E(S) = -p_1 \log p_1 - p_2 \log p_2 ... - p_n \log p_n = \sum p_i * log p_i$$

where p_i equals the percentage of a class i present in the child node that results from a split in the tree and $\sum p_i = 1$.

Let's see an example that best describes what entropy is. We have 2 baskets each filled with 50 easter eggs, covered by a sheet. These easter eggs are painted either blue or red. We reveal the baskets and start counting the distribution of the colors. We find that the first basket has 50 blue colored eggs, whereas the second basket has 25 blue and 25 red colored eggs. The first basket has an entropy of 0 as there is no randomness in the set and the second basket has an entropy of 1, as there is complete randomness.

Information gain is defined as how much would the entropy of the dataset be reduced, if we applied a certain type of partition. Mathematically:

$$IG(T, a) = E(T) - E(T|a)$$

In other words, it's the parent's T entropy minus the weighted childrens' entropy when splitting according to an attribute a. The attribute that maximizes the difference, the information gain on the examined node is selected. We repeat this process for each impure node until the DT is completed.

Another important question when creating a DT, is when to stop splitting. In datasets that have a large number of features, creating a link for each feature can lead to large, deep and complex DTs. A technique used to prevent this is selecting a maximum depth of the tree, which is the path from the root to the furthest leaf.

Decision Trees are a widely preferred Machine Learning Algorithm, as they are easy to comprehend and implement, can be used for many types of data, both categorical and numerical and most importantly, they function in a white box, which means the process of exporting a result can be easily explained using boolean logic. However there are still some disadvantages in this method. The most important is that the problem of creating an optimal decision tree is NP-complete, even for simple cases. Therefore, in practice, heuristic methods are used, such as the greedy algorithm, where locally optimal decisions are made at each node.[50]

4.2.3 Logistic Regression (LR)

Logistic Regression is a statistical method that is widely used in Machine Learning. In LR, a categorical output is determined from the linear combination of one or more dependent variables. The goal is to find the best linear model that connects the dependent variables with the desired output.[51] There are three types of Logistic Regression [52]:

1. Binary LR: There are only two output values (ex. 0 - 1 , False - True)
2. Multinomial LR: Three or more output values, unordered (ex.
3. Ordinal LR: Three or more output values, ordered (ex. rating 1-5)

A (non-) linear decision boundary / threshold can be set, which determines the prediction of a class. Binary logistic regression is used to predict the odds of being a case based on the values of the independent variables (predictors). The odds are defined as the probability that a particular outcome is a case divided by the probability that it is a noncase.

$$odds = \frac{p}{1-p} = \frac{\text{probability of presence of characteristics}}{\text{probability of absence of characteristics}}$$

The logarithm of the odds is defined as "logit" :

$$logit = \ln(\frac{p}{1-p})$$

The logit output is then fitted to the predictors. The predicted value of the logit is converted back into predicted odds via the inverse of the natural logarithm, namely the exponential function. That means that although the final output is categorical, LR calculates the continuous variable odds which states whether the characteristic is present. [53]

4.2.4 Balancing Methods

Most of the classification algorithms are made to perform their best with the input of a dataset balanced among all classes. However, especially in the field of biology, it is uncommon to find a dataset that is, from its creation, completely balanced. In the case of two classes, conventional classifiers give the same amount of attention in both classes, without taking into account the relative distribution of each class. That leads to a higher miss rate for the minority class, as the overall accuracy is mostly optimized on the majority class. This can lead to false and even dangerous conclusions if, for example the minority class is a high risk patient group which will be classified as normal.

The most common solution to this problem is changing the balances of the classes in the dataset before the training. This can be done with either over-sampling, which means artificially generating data from the minority class, or under-sampling, which means selecting fewer data from the majority class to be used during training. There are also methods that are based in the combination of those two techniques.

Oversampling has the benefit of keeping all the information from the original dataset and feeding it to training. However, duplicating or generating more data in order to create two balanced classes can lead to creating a huge amount of data, perhaps more than needed for the training, which effect training speed and memory usage.

Random Undersampling is another technique, which randomly selects a subset of the majority class to be used for training, in order to balance data between the two classes. However, this can also lead to important rules from the majority dataset being missed, as the selected amount of samples containing it was not enough for the classifier to recognize a pattern.

A more precise method is to enforce informed under- or over- sampling techniques, which apply a number of rules for selecting the data, in order to avoid loosing important information or overfitting in the case of oversampling. [54] [55]

4.3 Deep Learning

Deep Learning is a subcategory of Machine Learning that is based on recognizing multiple simple representations of more complex and raw data. Computational models that apply deep learning consist of multiple nonlinear processing layers, each of them trained to transform the representation they receive as input into a representation of a higher abstraction. Deep Leaning architectures have been applied to fields as computer vision, speech recognition, object detection, natural language processing and many others such as drug discovery and genomics.

Let's take for example a simple image recognition with the use of a deep learning model, as shown on Figure 4.11. The image is digitally represented as set of pixel values. However, for a computer, the mapping of these pixels to a human understandable object recognition is a complex procedure. Deep Learning mechanisms help break down this process in smaller, easier computations that, pipelined, compose the final result. The image pixels are put as input in the first layer, the visible layer, and are the only variables we are able to observe beforehand. Afterwards, a series of hidden layers are responsible for detecting features of the image and decide which of these features contain important information. Here, the first set of layers starts by detecting edges inside the image, for example by comparing nearby pixel brightness. The second layer receives this representation as input and detects the arrangements of the edges and contours. Then, the third layer assembles these observations into larger combinations that correspond to specific objects. Finally the last layer identifies the objects detected on the image according to a set of known labels.[1]

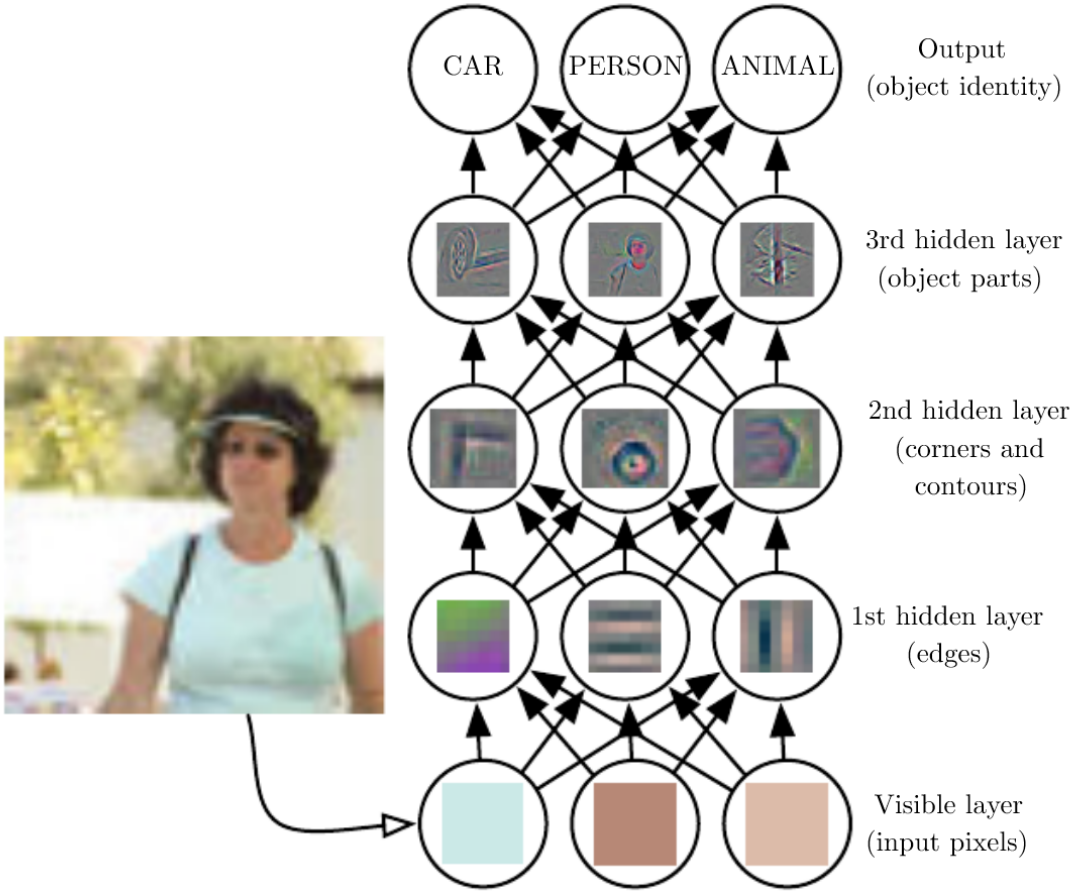


Figure 4.11: Example of Deep Learning Image Processing[1]

4.3.1 Deep Learning Networks

A Deep Learning Network (DNN) is an Artificial Neural Network with multiple layers between the input and the output layer, therefore the ”deep” characterization. In continuation with what we described above for ANNs, DNNs follow the process of weight manipulation, also called credit assignment in order to output the desired results.[56]

Deep Learning Networks manage to find complex mathematical relationships between input and output data, both linear and non linear, and compose representations of features difficult to be detected by more shallow networks. The most basic form is a Feed-forward network, in which the data go through each layer, from the input to the output, without any looping back. Its function is similar to the one described for the multilayer perceptron. Two other commonly used models are Convolutional NNs and Recurrent NNs. Recurrent NNs are used mainly for speech recognition, as they receive the input as a sequence which helps keep the relationship between the sequential data points.

4.3.2 Convolutional Neural Networks

Convolutional Neural Networks (CNN)[1, 57] are feed-forward ANNs that are mainly designed to process raw data structured in multiple arrays, such as 2D images. What differentiates CNNs as a category of DNNs, is the type of layers used in the hidden layers section. A standar CNN’s hidden layer consists of convolutional layers, pooling layers and fully connected layers.

The convolutional layers are the main body of feature extraction of the network. The neurons are connected in way so as to organize feature detection maps. All neurons inside a feature map share the same weight and each unit is connected to a field of neurons from a previous feature map via a set of trainable weights known as filter bank. The input image x is convolved with the weights of the feature map W_k . The result is then passed through a non-linear activation function, and thus the final output y_k is produced.

$$y_k = f(W_k * x)$$

The activation function allows for the nonlinear features to be extracted from the convolutional layer, One of the most common activation functions is ReLu, which is defined as

$$f(x) = max(x, 0)$$

.

The pooling layers receive the outputs of the activation layer and try to reduce the dimension of features by merging similar features together to one, single output. One common technique used mostly in the past is average pooling aggregation, which means that the input values from the same small image part were averaged and set as input for the next layer. However, max pooling, which states that the max value of the input set is fed into the next layer, is more widely used nowadays . [58]. The difference between max and average pooling can be seen at Figure 4.12.

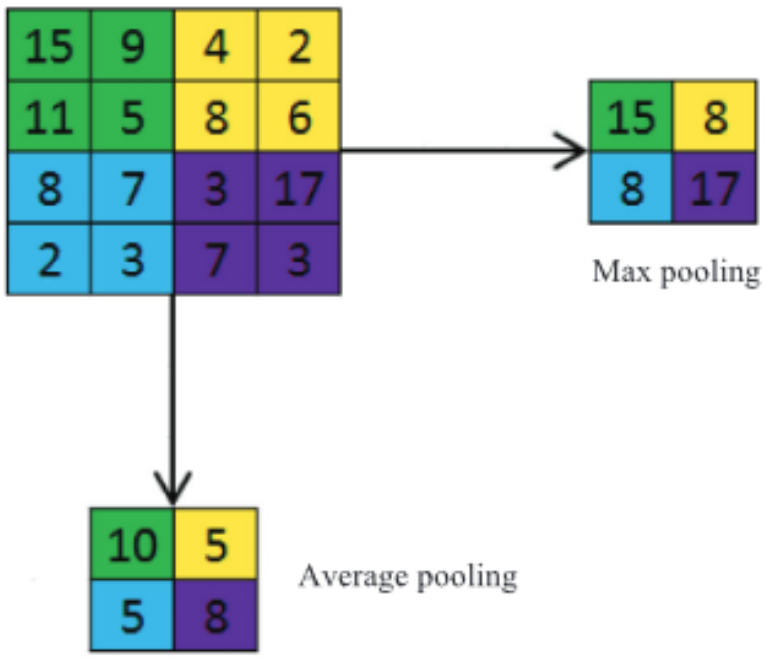


Figure 4.12: Average Pooling vs Max Pooling. A 4x4 image is passed through 2x2 filters in the convolutional layer. Maximum pooling outputs the maximum value of each 2x2 region, whereas average pooling outputs the average between the values of each 2x2 region. [2]

Sets of Convolutional and pooling layers are often stacked repeatedly one after the other multiple times, in order to improve feature detection. Afterwards, a set of fully connected neurons, known as a fully connected layer, connected to all of the activation functions in the previous layer, is responsible for making the high-level reasoning. At this stage the neurons will try to interpret the features detected and classify them.

In the end a normalization or loss layer is placed, which calculates the difference between the predicted value and the expected value (during the training). Different functions used are the soft-max function, which computes the success of a single class in between all exclusive classes, sigmoid function, which computes K independent probabilities of K different classes in a range of [0,1], and euclidean loss, used for regressing to real valued labels.

The training of the CNN is done as the training of an ANN, using the back - propagation algorithm.

For this theses, we used the Deep CNN ”Inception v3” developed by Google, as it was modified for the purpose of the ”Classification and Mutation Prediction from Non-Small Cell Lung Cancer Histopathology Images using Deep Learning” [7] paper.

4.3.3 Inception v3 Model

In 2014 M. Lin discovered that training multiple conv. layers simultaneously and stacking their feature maps linked with an MPL, produced a non-linear transformation, substituting the standard way of using linear transformations' convolutional layers followed by a nonlinear activation function. This idea is also known as the ”inception modules”[59].

C. Szegedy used this idea and developed a new model called the ”GoogLeNet”, known also as Inception v1[60]. Inception v1 consists of 22 layers of ”inception modules”, each of which consists of 1x1, 3x3,5x5 conv. layers and a 3x3 max pooling layer. This manages to increase the sparsity of the model and construct different patters. The feature maps produced from each module are chained and given as input to the next module. This model won the 2014 ImageNet Large-Scale Visual Recognition Challenge [61]. Figure 4.13 shows the structure of an inception module with dimension reductions and Figure 4.14 shows the architecture of GoogLeNet.

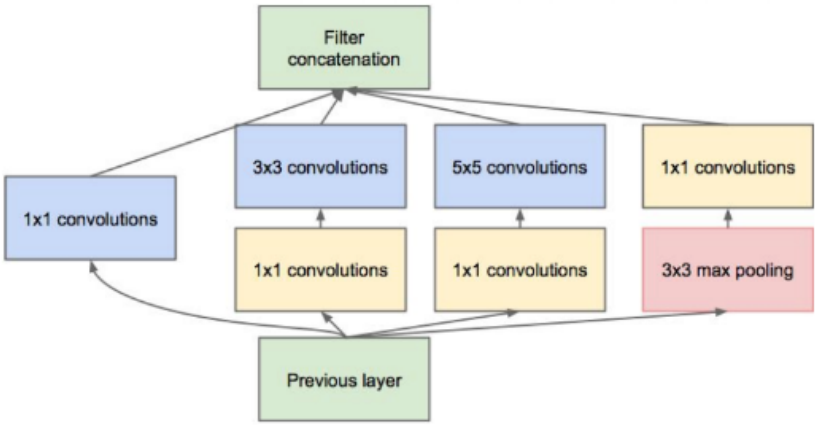


Figure 4.13: Inception Module [3]

Inception v2 and later on Inception v3 models were introduced one year later, at 2015, and are based on the architecture of Inception v1, of course with some optimizations and changes.[4] Such are the replacement of the 5x5 filters with two 3x3 filters consisting of 3x3 convolution and a 3x1 fully connected layer, which decreased the number of parameters used in each module and thus reduced computational time and cost. In v3, fine tuning and changes that improved analysis for images with higher precision, reached a 3.58% error rate over the 2012 ImageNet challenge. Figure 4.15 shows the architecture of Inception v3 Model.

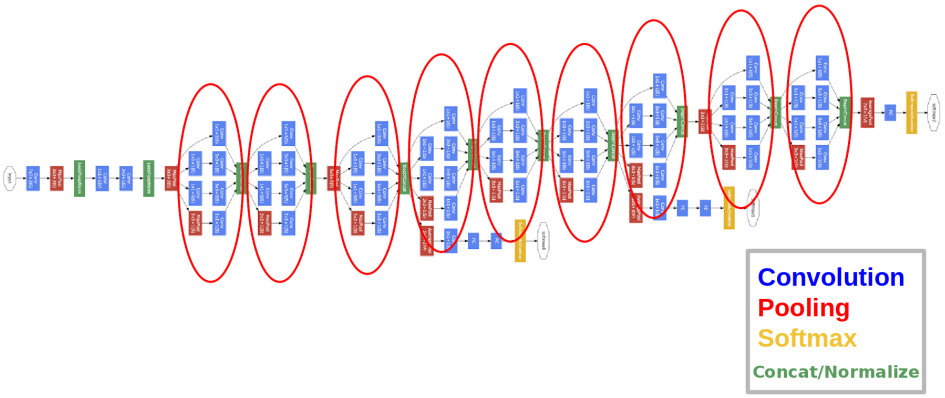


Figure 4.14: GoogLeNet Architecture [3]

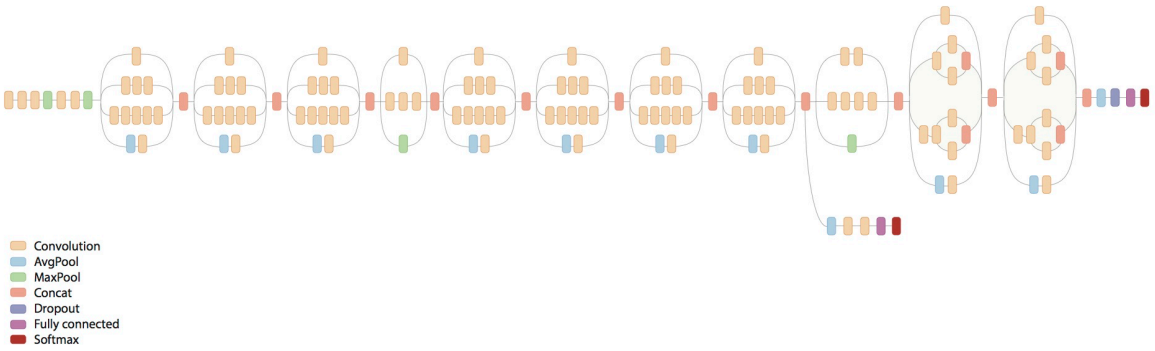


Figure 4.15: Inception v3 Architecture [4]

4.4 Evaluation Methods

4.4.1 Confusion Matrix

A confusion matrix [62] is a metric technique used for evaluating classification algorithms' results. In the case we have two classes, we define them as positive and negative glass. The confusion matrix will give the following results :

- Number of samples that belong to the positive class and were identified correctly as positive class - True Positive (TP)
- Number of samples that belong to the negative class and were identified correctly as negative class - True Negative (TN)
- Number of samples that belong to the positive class and were identified wrongly as negative class - False Positive (FP)
- Number of samples that belong to the negative class and were identified wrongly as positive class - False Negative (FN)

The accuracy of the classification is then calculated as :

$$\frac{TP + TN}{TP + TN + FN + FP}$$

Other important metrics are :

Precision = Proportion of positive identifications that was correct: $\frac{TP}{TP+FP}$

Recall/Sensitivity/True positive rate (tp) = Proportion of actual positives that was identified correctly: $\frac{TP}{TP+FN}$

False positive rate (fp) = Proportion of negative values incorrectly classified in comparison to the total negative values : $\frac{FP}{FP+TN}$

Specificity = $1 - fp$

In the case of mutli-class evaluation, for k classes:
Micro-averaged result =

$$\frac{\sum_{i=1}^{k} TP_k}{\sum_{i=1}^{k} TP_k + FP_k}$$

Macro-averaged result =

$$\frac{\sum_{i=1}^{k} Precision_k}{k}$$

4.4.2 ROC Curve & AUC - Area Under Curve

ROC (receiver operating characteristic) Curves are a very popular graph for measuring and visualizing the performance of a classifier [63]. The process of creating a ROC Curve is the following :

Given what we described above, in the case of a dataset with a two classes, if we train a classifier to recognize the dataset, we can easily measure the output and compare it with the input to produce the confusion matrix. Therefore, for each sample we have the original class and the predicted class, which all together populate the numbers of TP, TN, FP and FN.
Then, we can compute the true positive rates and false positive rates of the classification, according to the mathematical types presented above.

A ROC curve is a 2d graph, with true positive rate on the x-axis and false positive rate on the y-axis.

At figure 4.16 we can see an example of a ROC curve. The output of the classifier is the curve, whereas the y=x axis is the output of a random classifier, that classifies the dataset with a 50% chance for each class. The higher the curve, the longer the distance from the diagonal axis, and the better the classification. The area defined under the curve is used as the measure of the classification's accuracy and is called 'Area Under Curve'(AUC).

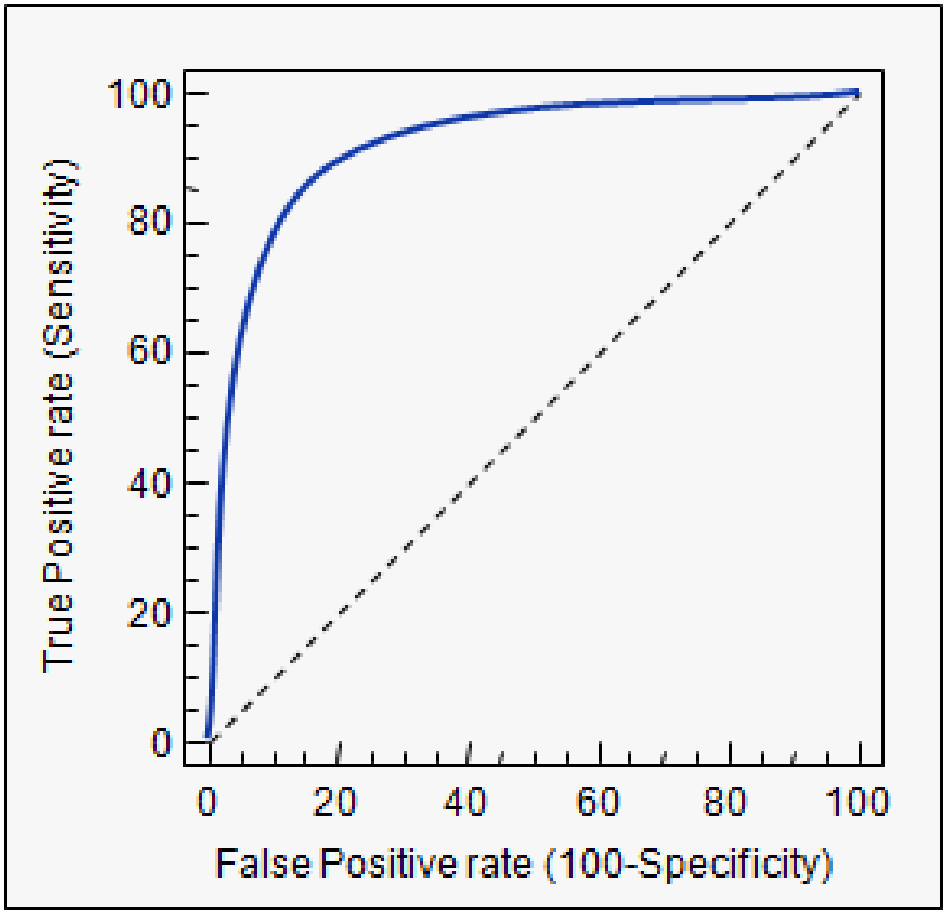


Figure 4.16: Example of a ROC Curve[5]

Chapter 5

Data Preprocessing - Implementation

The purpose of this chapter is to briefly present the procedure we followed for preparing the data used, and analyze the specific implementation of the algorithms and techniques described at chapter 4, on this thesis.

5.1 Dataset Preparation

As mentioned at chapter 3, we downloaded our data from the NCI Genomic Data Commons[30].

5.1.1 Sequencing Data Pre-processing

We downloaded and used gene mutation and expression data of a total number of 361 and 374 LUAD Lung Cancer Patients, respectively.

Mutation Data Format

The data came in the form of a MAF (Mutation Annotation Format) file. A MAF is a tab-delimited text file with aggregated mutation information from VCF Files, which contains the reports of somatic variants produced by the GDC DNA-Seq somatic variant-calling pipeline, comparing a set of matched tumor/normal alignments. Also, our data was in the form of Somatic MAFs (*somatic.maf), otherwise known as Masked Somatic Mutation files, which are further processed to remove lower quality and potential germline variants. Together with the MAF file, we had access to a meta-data file, which contained important information, such as the patient condition, vital stage, cancer stage etc.

The Somatic MAF file contained a total number of 208180 samples. Each sample carried information for a specific gene mutation of a unique patient. The most important information acquired from the dataset is:

- Hugo Symbol: The Hugo Symbol of the genes, according to the HGNC Database. [64] Hugo symbols are strings which are always in caps.
- Entrez Gene Id: Another form of label of the genes, according to the NCBI Database [65].
- Chromosome: The affected chromosome (chr1)
- Variant Classification: Translational effect of variant allele (Silent Mutation, RNA, Missense Mutation and other types).
- Variant Type: Mutation Type
- Tumor Sample Barcode: Aliquot barcode for the tumor sample. The barcode is a collection of identifiers, each of which specifically identifies a TCGA data element.

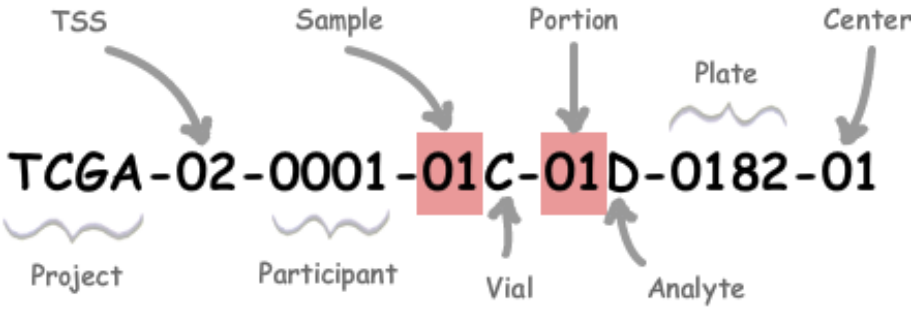


Figure 5.1: A TCGA Tumor Sample Barcode. Source: NCI GDC Documentation [6]

According to figure 5.1, the barcode carries important information about order numbers and also vital information such as the Sample Type, which implies whether it's a Tumor or Healthy Tissue sample. The first 12 characters are used to uniquely define the patient.

- Tumor Sample UUID: GDC aliquot UUID for tumor sample
- BIOTYPE: Biotype of the transcript (for ex. protein coding, processed transcript)
- IMPACT: The impact modifier for the consequence type (Low, Moderate, High, Modifier)
- FILTER: Copied from input VCF. This includes filters implemented directly by the variant caller and other external software used in the DNA-Seq pipeline.

Mutation Data Filtering

With the use of the information described above, The steps we applied are the following:

1. We kept only filtered (PASS) Non Silent Mutations (Variant Classification different than 'Silent mutation', which means mutations that cause to a change of the DNA chain).
2. We dropped also 290 samples that had an Entrez Gene ID of 0 and an Unmapped Hugo Symbol.

At this stage, one can extract multiple information, both per gene type and per patient type. For the purpose of this thesis, we set the 'Hugo Symbol' feature equal to 'EGFR' and we were able to detect the number of cases with an EGFR non silent mutation, which is the object of our research. From the 361 unique patients, the number of patients that presented an EGFR mutation is 43.

Expression Data Format

For each patient we downloaded a txt file with the calculated expression signal of a series of genes, as measured from the mRNA sequence from each patient's tumor sample, as well as a metadata file with information about the patient just as described above. As a patient identifier, the first 12 characters of the Tumor Sample barcode is used.

Expression Data Filtering

In order to assign expression subtype labels to each of the patients, we replicated the work described at Collisson et al. [14]. The process was the following :

Gene expression data were median centered per gene. With the application of a nearest centroid predictor [66], a subtype was assigned to each tumor speciment, using previously published predictor centroids [25, 23]. The process was limited to the genes common to the predictor and the TCGA cohort

and with the use of the Pearson Correlation, the subtybe with the maximum correlation coefficient was chosen for each sample.

The difference we added to the above procedure is that we created a fourth subtype called Unknown, to which we classified the samples with a variance lower than 0.1 between the 3 common classes. The final dataset is :

Subtype	Number of samples
TRU	171
PP	40
PI	102
Unknown	61

Table 5.1: Number of patient per subtype

5.1.2 Image data Pre-processing

For each patient ID from the sequencing data, there is a matching stained histopathology image of a tissue sample. The images were obtained from the Genomic Data Commons Database [6].

According to the result of the Sequencing Data in terms of the presence of an EGFR mutation, we were able to match and also label each of these images. At figure 5.2, a tissue slide of a patient with an EGFR mutation is presented, whereas at figure 5.3 , we see a picture of a tissue slide of a patient that did not acquire an EGFR mutation.

Figure 5.2: Histopathology Image of Patient no TCGA-86-8074, labeled as EGFR Mutated.

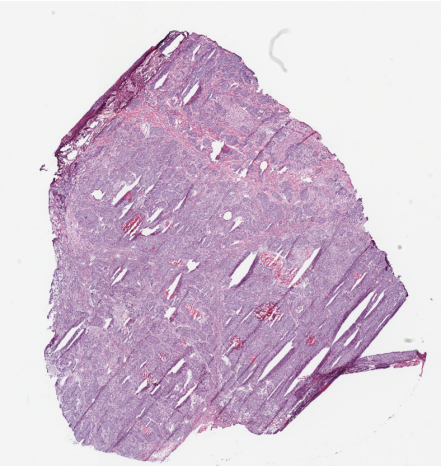
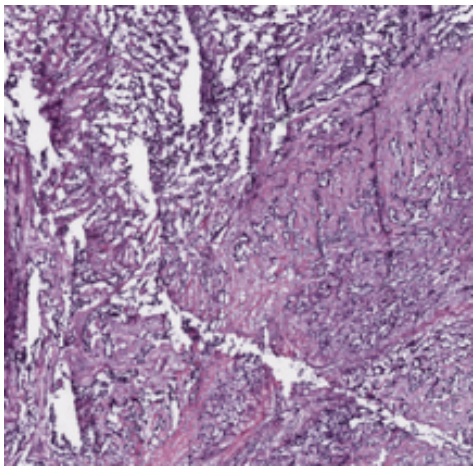
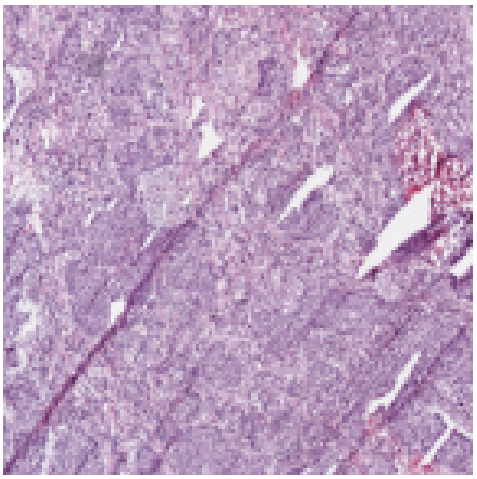


Figure 5.3: Histopathology Image of Patient no TCGA-55-8620, labeled as non EGFR Mutated.

As we mentioned in the Histopathology slides theory, these images, created with a magnification of x20 to x40, have a significantly large size due to the high resolution and magnification accuracy (from 10.000 pixels to 100.000 in each of the two dimensions). This makes them difficult to manage and process. For this reason, each image was split in non overlapping tiles of size 512x512 px, with the use of the OpenSlide tool [67], as modified by Coudray, Nicolas, et al [7]. In that way, we ended up with multiple tiles per image. Because of the different size of the images, there were images with tens of tiles and images with thousands of tiles. At figure 5.4 an example of such tiles is shown, for both of the categories.



(a) EGFR Mutated Tile



(b) EGFR Non Mutated Tile

Figure 5.4: Histopathology Image tiles, 512x512 px

One problem we faced was the fact that the image consists of smaller areas that may differ with each other in terms of diagnosis. Specifically, there are areas of the image that are recognized as normal tissue and areas that are labeled as tumor tissue. Within the same image, there can be tumor tissue of more than one type of lung cancer (for ex. an area of adenocarcinoma and an area of squamous cell type). However, our sequencing data specifically contained data from patients identified with adenocarcinoma lung cancer. Therefore, for each image we acquired the information from Coudray, Nicolas, et al [7], which included per tile classification (normal, luad, lusc) and also per tile prediction result from the CNN (correctly predicted, falsely predicted). According to this information, we filtered the slides and kept only those that where labeled as luad slides and at the same time, were also predicted

as luad tiles by the CNN. In that way, we were able to keep the parts of each image that actually contain the information correlating with our sequencing data.

The total number of tiles that was ultimately used was :

Mutated	Non Mutated	Total
36.045	198.562	234.607

Table 5.2: EGFR Mutated and Non Mutated Tiles Split

Subtype	Number of tiles
TRU	113690
PP	29035
PI	80208
Unknown	35715
Total	258648

Table 5.3: Number of tiles per subtype

5.2 EGFR Deep Learning Implementation

We begin by using a Deep Learning network in order to identify a potential distinction between the EGFR mutant slides and non mutant slides.

We used the Inception v3 model[68], written in TensorFlow, as modified from Coudray, Nicolas, et al [7] in order to recognize tumor vs normal histopathology slides. TensorFlow is an open source software library for high performance numerical computation [69], easily used with GPU modules. EGFR mutation was treated as a binary classification. We used the technique described for the ”Identification of gene mutations”.

Firstly, the tiles were split into 70% training , 15% validation and 15% testing data. The splitting was performed at patient/slide level, which means that all tiles from the same patient were assigned to the same set of data, in order to avoid overlaps. Then, a second split happened, in which the tiles were converted to TFRecords, which is the tensorflow form of data used as input for the CNN, and labeled according to a file which matched the ID of the patient with the EGFR mutation information.

As mentioned in Theory, the inception v3 architecture includes the use of inception models which consist of multiple convolution layers of different kernel size and a max pooling layer. For the initial 5 convolution nodes, 2 max pooling layers combine their results and output them to 11 stacks of such inception modules. The network ends with a fully connected layer, followed by a either a softmax or a sigmoid output layer. For the EGFR gene, we used the sigmoid output layer with only one positive outcome : EGFR mutation.

We also used the following parameters as defined: The loss function equals the cross entropy between predicted probability and the true class labels, and the RMSProp[70] optimization was used, with a learning rate of 0.1,a weight decay of 0.9, momentum of 0.9, and epsilon of 1.0. We trained the network for a total of 500,000 iterations. In order to avoid over fitting, we run validation tests on a regular basis and calculated the cross-entropy loss function on the train and validation dataset, and finally used the model with the best validation score.

For testing the network, we performed test on the tiles and then aggregated in order to output a result for the whole tissue slide.

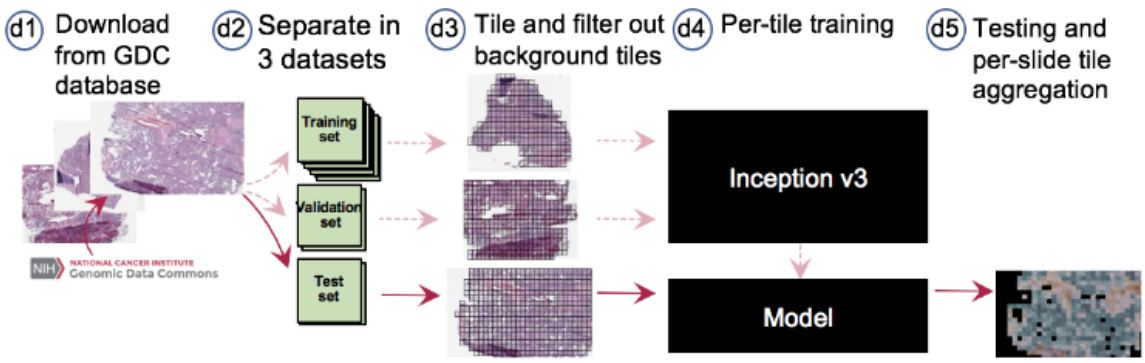


Figure 5.5: Image obtained from Coudray, Nicolas, et al [7]. The slides obtained from the GDC Database were separated into 3 different data sets, then tiled and only luad identified tiles were kept. The tiles were used as input for the full training of the modified inception v3 model. Finally ,testing on the tiles and aggregation over them in order to conclude the accuracy of the prediction over the whole slide was made.

The pipeline of the training can be observed at figure 5.5

Consecutively, we performed a full training of the network described above, on the computing resources at the High Performance Computing Facility at NYU Langone Medical Center.

5.3 LUAD Molecular Subtypes Training Implementation

For the prediction of the 4 subtypes from histopathology images, we used a similar process as the one described above.

The Image data pre-processing was done in the exact same way, leading to a total number of 258648 tiles split of which 113690 is TRU positive, 29035 is PP positive, 80208 is PI positive and 35715 is Unknown.

Again ,we used the Inceptionv3 model, with the same fine tunning and with a sigmoid output layer to predict the four classes.

5.4 Digital Image Filtering Implementation

For the process of the Digital Image Filtering we will follow consists of 3 steps:

1. Image Segmentation
2. Image Filtering & Feature Extraction
3. Feature Classification

The theory of the above steps was thoroughly analyzed at chapter 4, therefore we will focus on describing the pipeline we used from the initial images to the final classification output and the specific parameters we chose for each step. Figure 5.6 shows a diagram of that pipelined procedure.

The subject of our research remains exactly the same : The detection of an EGFR mutation case texture differentiation of histopathology tissue slides. Again, as the initial tissue slides' size is quite large, we decided to use the same tiles created from the procedure described for the Deep Learning method.

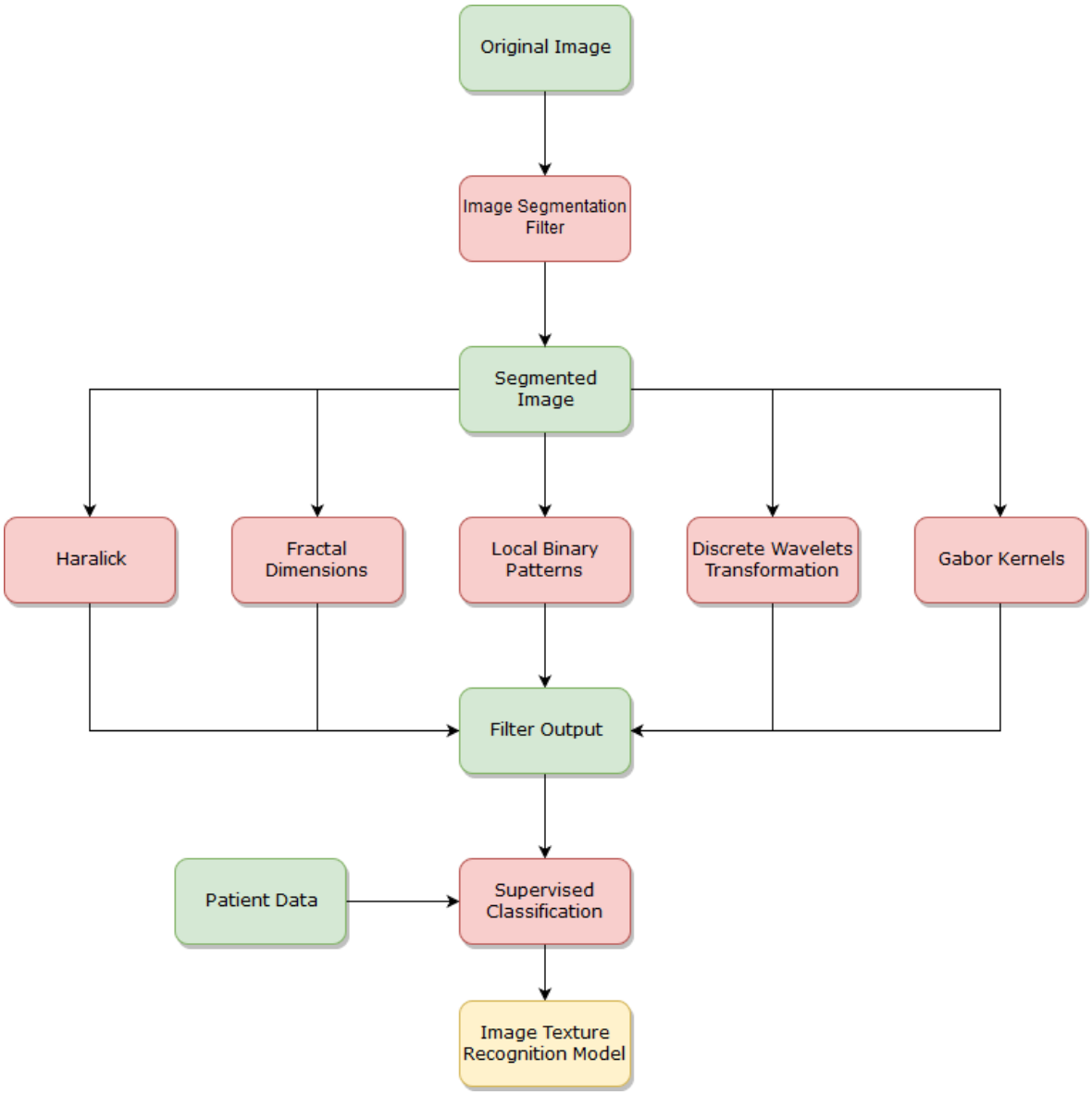


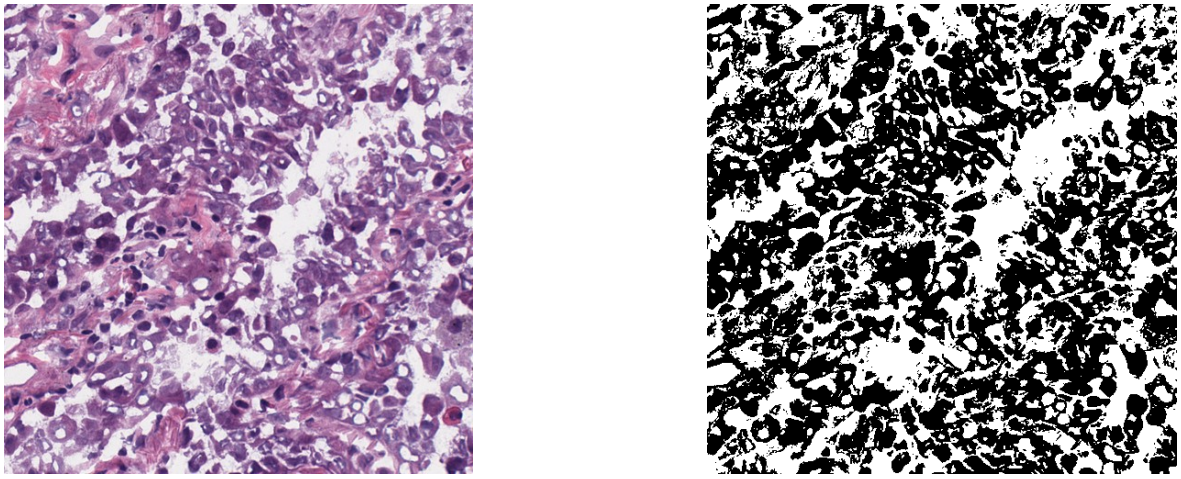
Figure 5.6: Image Texture Recognition Pipeline, followed for each tile. At first, we pass each tile from a segmentation filter. Then we collect the output and pass it on to 5 image texture recognition digital filters. We concatenate the output, label it according to the patient data and pass it on to a supervised classifier, in order to train it to detect features connected with a patient's EGFR mutation possibility.

5.4.1 Image Segmentation

As mentioned, this technique is commonly used in order to separate areas within the same image that present a morphological interest from those that can be omitted and are recognized as background noise.

The technique of thresholding was chosen, as it is a very simple and highly effective method for this purpose. Each tile was passed though a filter with a binary threshold b , with b valued from 0 to 255. Choosing the right threshold is an important factor, as it is crucial that most of the background noise is filtered, while at the same time important features of the image are not lost. After several tryouts, we set the threshold for each tile at b = 150. All the pixels with a value below 150 are colored white (255), whereas all the pixels above 150 are colored black (0). At figure 5.7, an example of an

actual tile before and after segmentation is presented.



(a) Original Tile **(b)** Segmented Tile

Figure 5.7: Image Tile Segmentation, original and segmented at a threshold of 150

5.4.2 Image Filtering & Feature Extraction

This is the most important part of the process. The methodology is that each tile will pass through a feature extraction filter and from each filter, a vector of attributes related to the tile, will be produced. These vectors will be combined, forming a dataset of features, which will be later used for the image classification.

The filters we used are the ones explained at chapter 4, with the following parameters :

1. Gabor Filter. We used the implementation of gabor filters as defined at the scikit-image python library [71, 72]. The parameters we set are the following :

   - theta: range(4) / 4. *π
   - sigma: 1,3
   - frequency: 0.05, 0.25

   That produces a total of 16 gabor kernels.

2. Haralick Filter. We used the implementation of grey-level co-occurrence matrix at the scikit-image python libray [72, 73]. After we compute the gray level co-occurrence matrix, we select the following 5 features : dissimilarity, correlation, homogeneity, ASM, energy.

3. Local Binary Patterns Filter. We used the implementation of local binary patterns at the scikit-image python libray[72, 74]. We set a radius of 3, and therefore we included in the calculations of each window 8*3 number of points/pixels. Also, we set the method of determining the pattern as 'ror', which is the calculation of the gray scale cycle for each window, taking into account a rotation invariant which denotes the shift to the right from the reference point. We produce a total of 20 features, as the bins of the histogram created.

4. Fractal Dimension Filters. Here we set an initial box size of 2 with a scaling factor of 2. The final result is one feature containing the fractal dimension calculation.

5. Wavelets Filter. We used the implementation of wavelets as defined at the PyWavelets python library [75]. Also, the 2nd wavelet from the Daubechies wavelet family was used. Daubechies

is considered the most popular wavelet family for texture feature detection. They are a family of orthogonal wavelets defining a discrete wavelet transform and characterized by a maximal number of vanishing moments for some given support. [76] The key difference is that the Daubechies wavelet uses overlapping windows, therefore the results include all changes between pixel intensities. We produce 2 features, the average and detail coefficients.

We concatenate the above features in a total dataset of 44 features for 234607 samples/tiles. After dropping features that equal zero for all tiles, we are left with 34 filter features.

The important detail at this part is this is a one-time calculation. In other words, the filters are calculated upon the images using as input only the pixel values and without taking into account any metadata or labels of the picture. That means that, once the final dataset of all feature outputs is created, it is eligible to be analyzed and answer any type of question we want to ask about a specific attribute we want to study. Therefore, we could choose to use them to train a classifier for detecting differentiations of any type of gene mutations or tumor stage information, gene expression information etc. For the purpose of this thesis, and in cohesion with the above technique of deep learning, we focused on detecting an EGFR mutation texture feature.

5.4.3 Features Classification

As mentioned, a way to interpret the texture features calculated is to create a classifier that will detect valuable relationships between the texture data and known patient data. For this purpose, we assigned to each feature vector calculated for each tile a binary label stating whether the patient matching with this tile presented an EGFR mutation or not. We implemented and tried two classification methods, a decision tree and logistic regression.

For the code implementation we used the python Scikit-Learn library [77].

Decision Tree Classification Implementation

The theory of the Decision Algorithm was presented at chapter 4.

As our dataset is highly imbalanced, in order to achieve a non biased result, we implemented the technique of imbalanced learning. Specifically, we used Random - Under Sampling from imbalanced-learn library [78] compatible with scikit-learn. We set the ratio equal to 1, which means that all data from the minority class (EGFR mutated) equal data from the majority class(EGFR non mutated) will be selected.

Using the grid search algorithm from the scikit learn library, which implements an exhaustive search over specified parameter values, we tested the following values assigned to the Decision Tree classifier :

- Criterion: {Gini, Entropy}. They define the use of the Gini impurity or the Information Gain metric
- Splitter: {Best', Random}. At best split, the random tree split is done on the most relevant feature. At random split, the random tree split is done at a completely random feature.
- Maximum depth : [2-9]. This parameters sets the maximum depth of the tree from 2 to 9, without allowing it to overpass it.

A plot that describes the performance of each combination for training the classifier is shown at figure 5.8

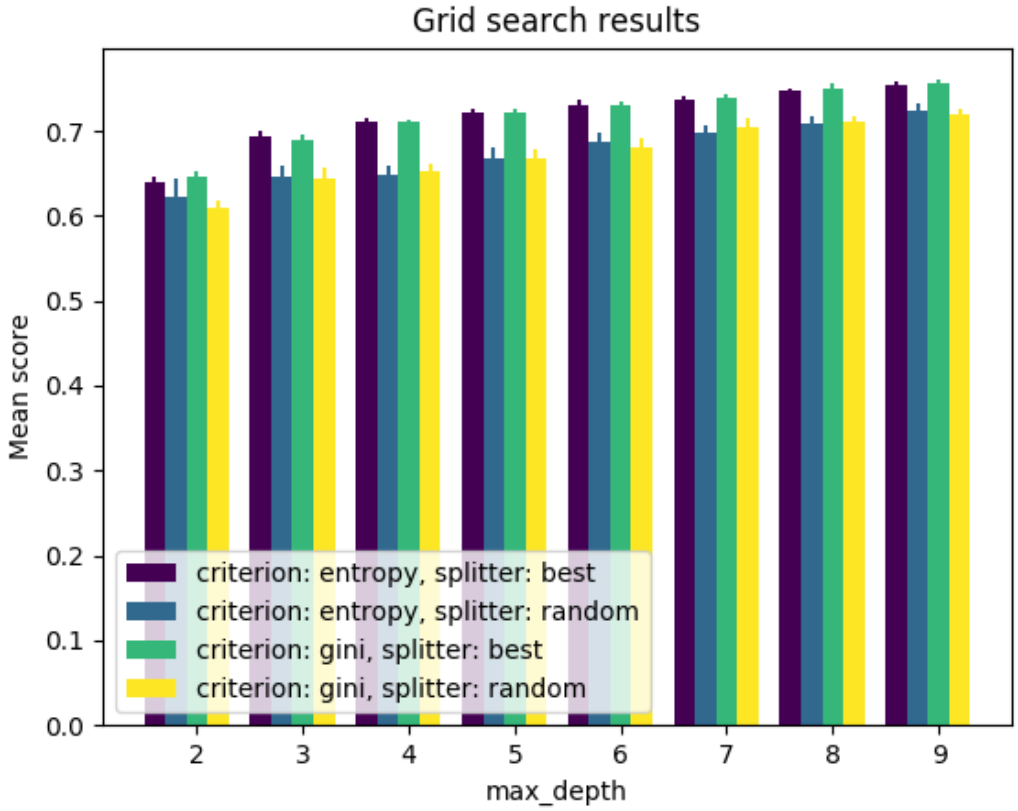


Figure 5.8: Plot results of sklearn grid search for Decision Tree Classifier

As a metric for the evaluation, the mean score of accuracy is computed over each combination. As we can also see from the graph, the best combination that was selected is Gini, Best, 9, with a mean accuracy of 0.733. As we observe from the graph, the maximum depth of the tree had an almost linear relationship with the accuracy result, achieving the best performance when it equals 9. The depth was not increased more however, as with deeper trees we noticed there was no increase of the performance. Moreover, we see that splitting at the best feature is more efficient than splitting at a random feature, in every combination. Finally, the Gini impurity and the Information Gain metrics seem to have a really close performance score, however after several repetitions, the Gini Impurity metric always performed better, even with a really small difference.

Afterwards, we proceed to the training with the above parameters selected. We use 80% (288 patients, 220566 tiles) of data of each class for training and the rest for testing (73 patients, 14041 tiles). Splitting is done at patient level, as the original labeling of our data is also done at patient level.

Following, we fit the classifier with the specific parameters as they emerged from the above analysis.

5.4.4 Logistic Regression Implementation

Another classifier we used is Logistic Regression. The theory of Logistic Regression was presented at chapter 4.

The process we followed is exactly the one described for the Decision Tree classifier, with the only difference being the parameters set for the classifier. For Logistic Regression, we fine-tuned the parameter C, which is the Inverse of regularization strength. Regularization is applying a penalty to increasing the magnitude of parameter values in order to avoid overfitting. The larger the C, the less

likely the parameters will be increased in magnitude, in order to adjust for small perturbations in the data. The values assigned to C for testing are :

$C = [60, 70, 80, 90, 100, 110]$

At figure 5.9, we can see that eventually, changing the values of C lead to zero effect on the training accuracy. Therefore, a random value of C was chosen each time.

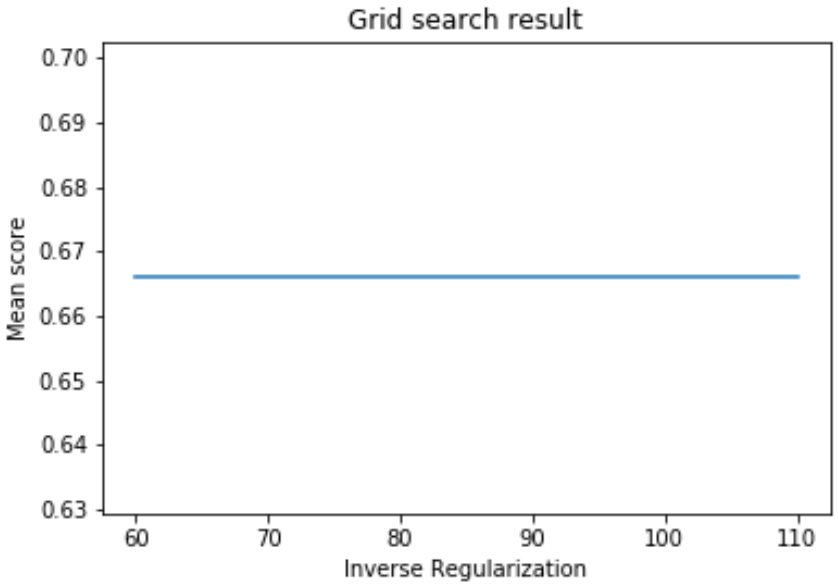


Figure 5.9: Plot results of sklearn grid search for Logistic Regression

Chapter 6

Results

In this chapter, after having presented all the theoretical background and techniques used for the purpose of this thesis, the core results of our work will be analyzed.

6.1 Deep Learning Results

6.1.1 EGFR Training Results

The first approach we followed in order to identify the existence of an image texture differentiation in the case of an EGFR mutant patient, was the use of a Convolutional Neural Network.

After the training was competed, we performed tests on the trained network, and the result of one can be seen at figure 6.1

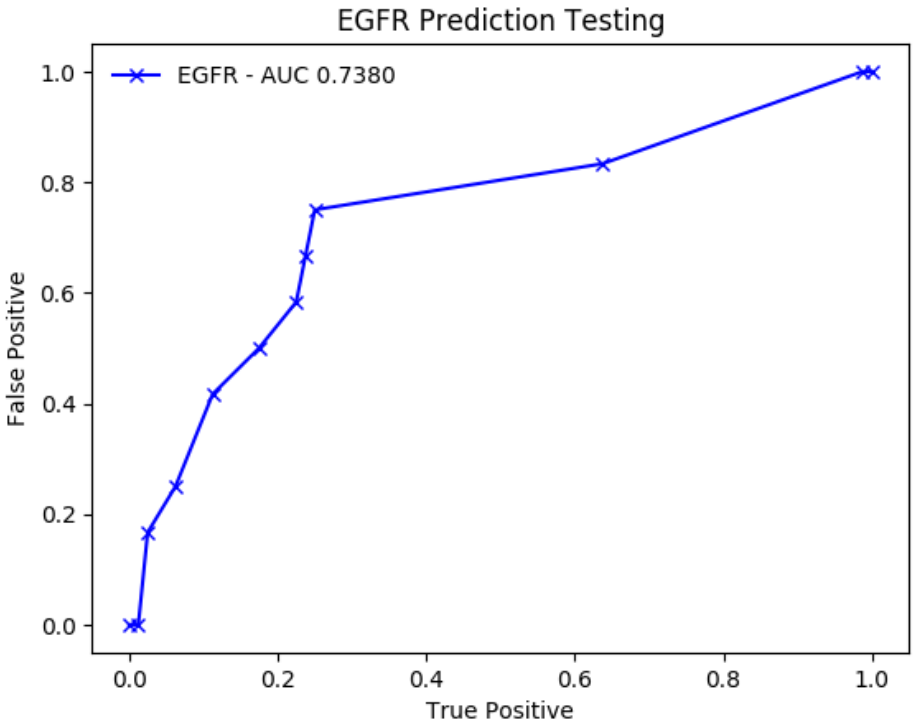


Figure 6.1: ROC Curve of EGFR CNN Training, AUC per Tile

From the above graph we can observe that there is a clear indication of some kind of texture features that create a different pattern of the tumor tissue, in the case of an EGFR mutation. In detail, the network classifies with a success rate of 74% the images into the two given categories. This is an extremely valuable discovery, as the detection of an EGFR mutation is an important factor for

determining various parameters in the decision of the treatment that will be decided for the patient [79].

However, Deep Learning techniques are characterized in many cases as ”black boxes”. That means, that their decision making strategy is so complex, that in most cases it is hard or sometimes even impossible to define what are the features that the CNN has learned, recognizes and classifies a sample accordingly. For that reason, it is a technique still often avoided by the medical community when it comes to vindicating a medical conclusion.

Therefore, in order to further explore why the CNN distinguishes these two categories, we decided to explore two more approaches. The first is to explore the connection of the EGFR gene with the TRU subtype and whether they are correlated in terms of histopathology texture. We do that by training the CNN to recognize the 4 basic molecular subtypes and then analyze it's results related to the EGFR gene mutation. Moreover, we decided to test the connection of an EGFR mutation with the histopathology slide, using a different, more traditional and acceptable technique, that of Digital Image Filtering.

6.1.2 LUAD Molecular Subtypes Training Result

According to what described above, we performed a Deep Learning training, targeted to recongize the four molecular LUAD subtypes. The results of a full training can be seen at figures 6.2 and 6.3, per tile and per slide accordingly.

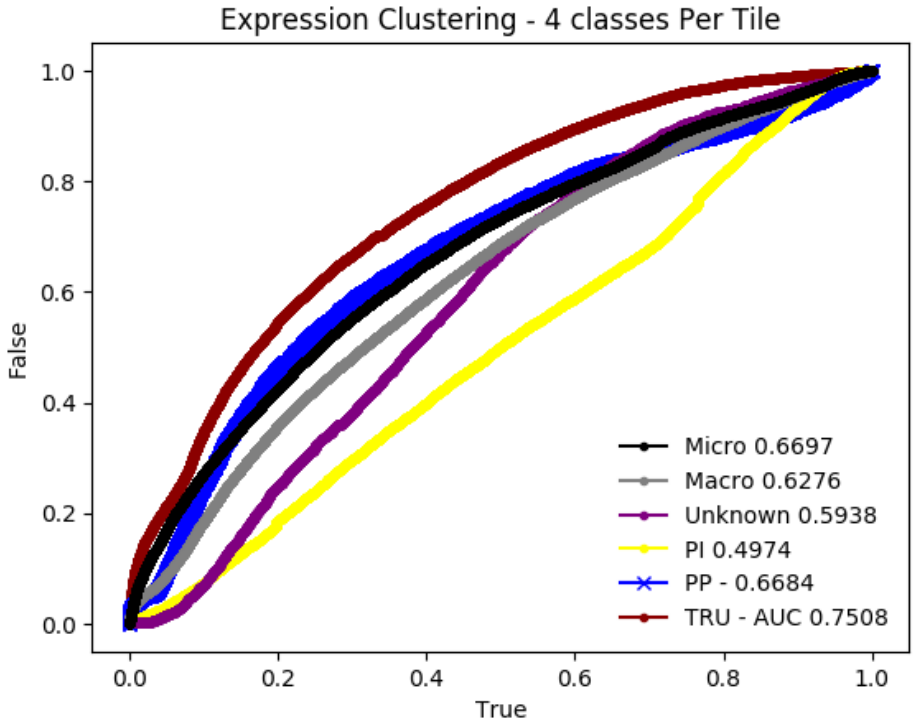


Figure 6.2: ROC Curve of Expression Subtypes CNN Training, AUC per Tile

From the two figures we observe the following :

- The TRU subtype is predicted with a 75% accuracy per tile and 85% accuracy per slide. That confirms the theory behind the creation of these subtypes, as derived also from histopathology features. However, we should not confuse the detection of the TRU subtype with the EGFR gene mutation detection. TRU subtype is indeed enriched in EGFR gene mutation, but it is not identified only by it.

  This is something we confirm by looking a the relationship between the two datasets. We used a total number of 86 patients to test the subtypes' classifier. From those, 36 were labeled as

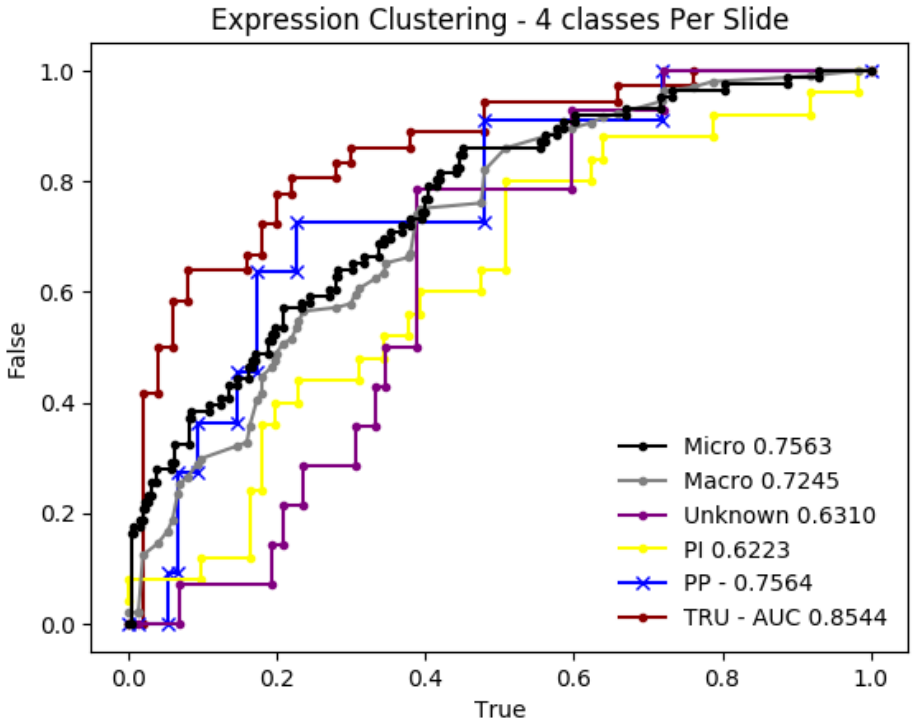


Figure 6.3: ROC Curve of Expression Subtypes CNN Training, AUC per Slide

TRU and from those, the 30 were correctly classified as TRU. Combining the above data with EGFR mutation data, we see that only 12 of 36 and 11 of 30 are EGFR mutated, which confirms the theory of EGFR mutation enrichment on the TRU subtype but also states clearly their distinction.

- The PP subtype is predicted with a 66% accuracy per tile and 75% accuracy per slide. Although, per slide, the classifier achieves a significant success rate, it is not enough to give us a clear picture the subtypes' detection within the image, but rather just a hint.
- The PI subtype is predicted with a 50% accuracy per tile and 62% accuracy per slide. We can safely say that the classifier didn't manage to predict histopathology features connected with the PI subtype.
- The Unknown subtype is predicted with a 60% accuracy per tile and 63% accuracy per slide. We didn't expect high rate at this category, which was confirmed, as it was created mainly with the purpose of maintaining the three other classes more cohesive.

Given the positive result of the TRU subtype, we decided to test a binary classification between the TRU subtype and the rest.Therefore, we split our data in two categories :

- Patients classified as TRU LUAD subtype
- Patients classified as PP, PI or Unknown LUAD subtype

The data used for these two categories are presented at table 6.1.

Following the exact same procedure as for the EGFR binary training, the testing results are presented at figures 6.4 and 6.5 .

As we can see, the model was able to distinct with a good 74% accuracy per tile and 85% per slide, a TRU subtype related tissue slide from a non TRU related one. This is a very positive result as

TRU Patients	Non TRU Patients
171	203
TRU Positive Tiles	Non TRU Positive tiles
113690	144958

Table 6.1: TRU / Non TRU number of tiles and slides

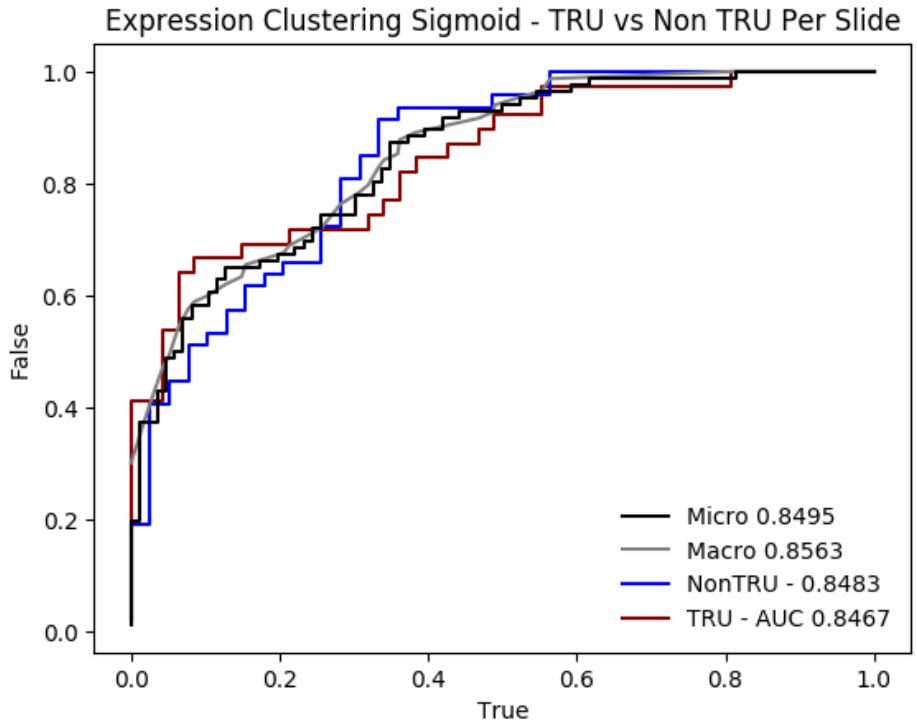


Figure 6.4: Expression Clustering Training ROC Curve - TRU vs Non TRU per Slide

it confirms the theory behind the creation of the expression subtypes and, most importantly, our model.

To conclude, we managed to confirm the relevance between the expression subtypes and histopathology features, and, more importantly for the purpose of this thesis, confirm the relevant distinction between the histopathology features found for TRU positive cases and EGFR mutation positive cases. Therefore, we continue with exploring the EGFR mutation connected features with the use of Digital Image Recognition techniques.

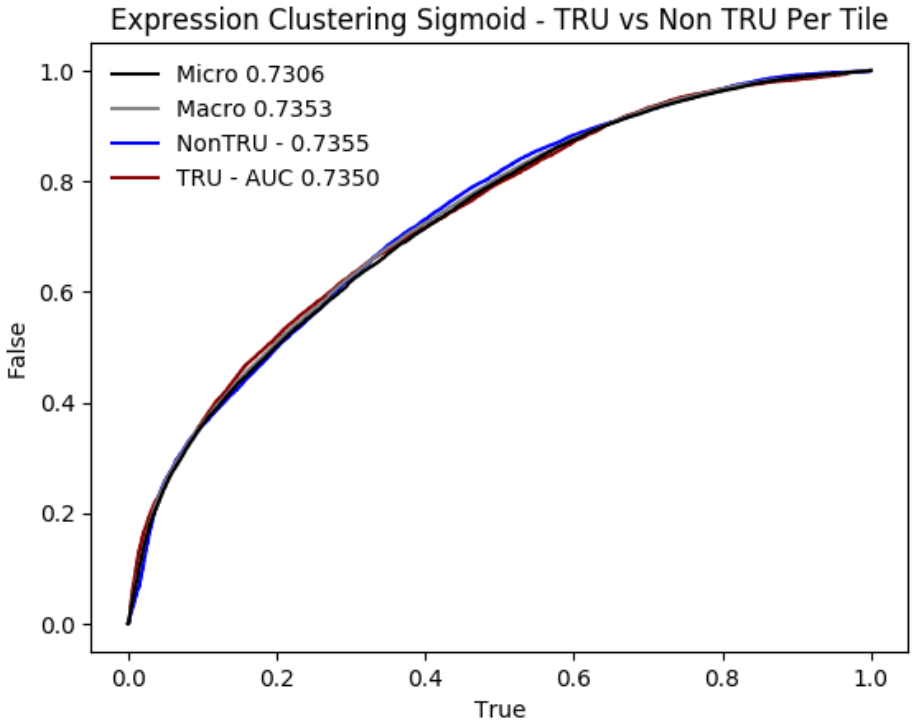


Figure 6.5: Expression Clustering Training ROC Curve - TRU vs Non TRU per Tile

6.2 Digital Image Filtering Results

In order to further validate and explore the result of the Deep Learning network for the EGFR mutation histopathology slide texture differentiation, we proceeded with the use of Digital Image filtering. We specifically use the procedure described at chapter 5 and produce a total of 34 features for each tile. Following, we present the results of the two different algoriths, Decision Tree Classifier & Logistic Regression.

6.2.1 Decision Tree Classifier Results

After training the Decision Tree algorithm according to the process presented at chapter 5, we test it to a new dataset of 73 patients, unseen by the algorithm

The metric results of the testing are presented at table 6.2. The confusion matrix and the ROC curve produced from test data are shown at figures 6.6 and 6.7.

	MEAN	STD
AUC	0.913353	0.005614
Accuracy	0.868744	0.008484
G-Mean	0.909212	0.006181
Precision	1.000000	0.000000
Recall	0.826705	0.011228
Sensitivity	0.826705	0.011228
Specificity	1.000000	0.000000

Table 6.2: Results Micro Decision Tree

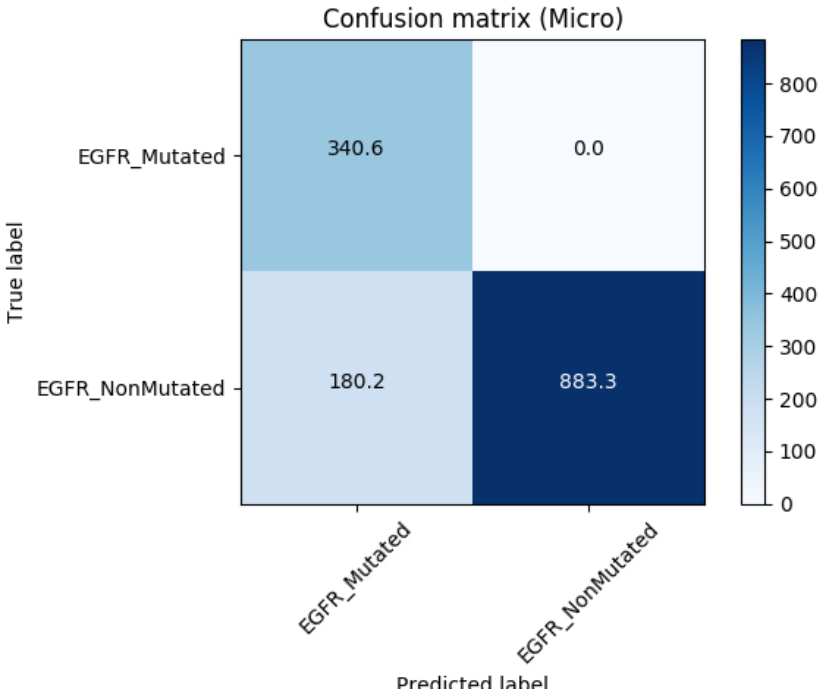


Figure 6.6: Micro Confusion Matrix for testing on Decision Tree Classifier

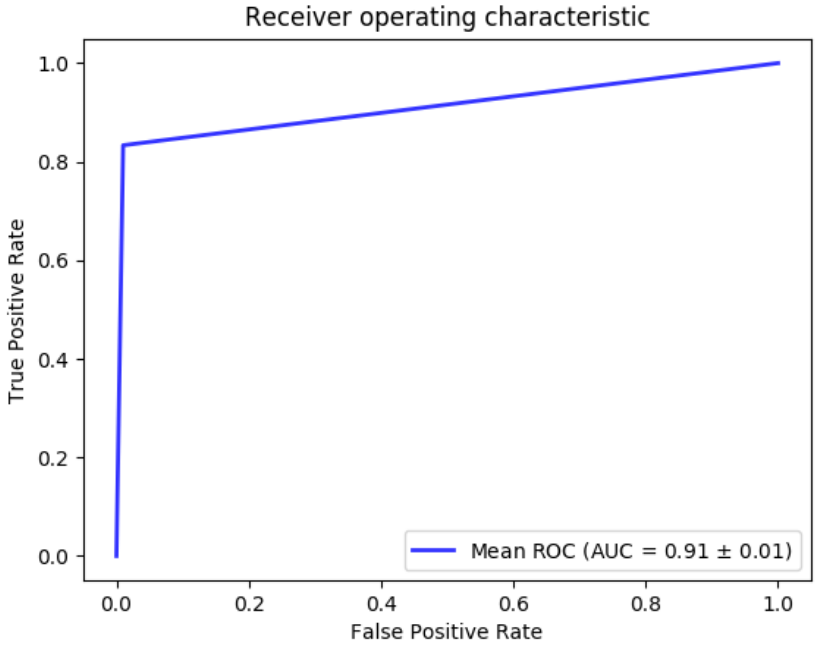


Figure 6.7: ROC Curve for Decision Tree Classifier

The results produced are quite encouraging. In detail, the classifier predicts the case of an EGFR mutation depending only on the image texture features with an AUC of 0.91. That means that we can be 91 % sure that there is a combination of texture characteristics that identify tumor slides from patients that have an EGFR mutation, in comparison to those that do not.

Logistic Regression Results

Another classifier we used is Logistic Regression. The implementation of Logistic Regression was presented at chapter 5.

The presented results are for C=100.

The performance of the Logistic Regression Algorithm is shown at table 6.3 and figures 6.8 and 6.9.

	MEAN	STD
AUC	0.661665	0.011207
Accuracy	0.663267	0.010663
G-Mean	0.661537	0.011300
Precision	0.858768	0.007026
Recall	0.664781	0.013375
Sensitivity	0.664781	0.013375
Specificity	0.658549	0.019681

Table 6.3: Results Micro Logistic Regression

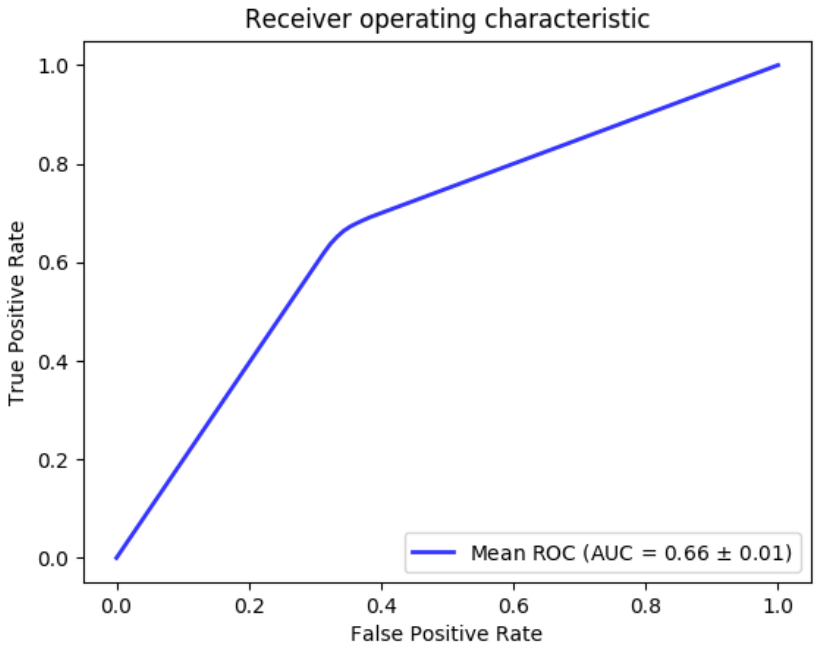


Figure 6.8: ROC Curve for Logistic Regression

An AUC of 0.66 or 66% is achieved with the use of this classifier. We observe that, although this algorithm also detects a texture change in the case of an EGFR mutation, it does not outperform the algorithm of the Decision Tree. Therefore, we will keep the result as defined from the Decision Tree algorithm.

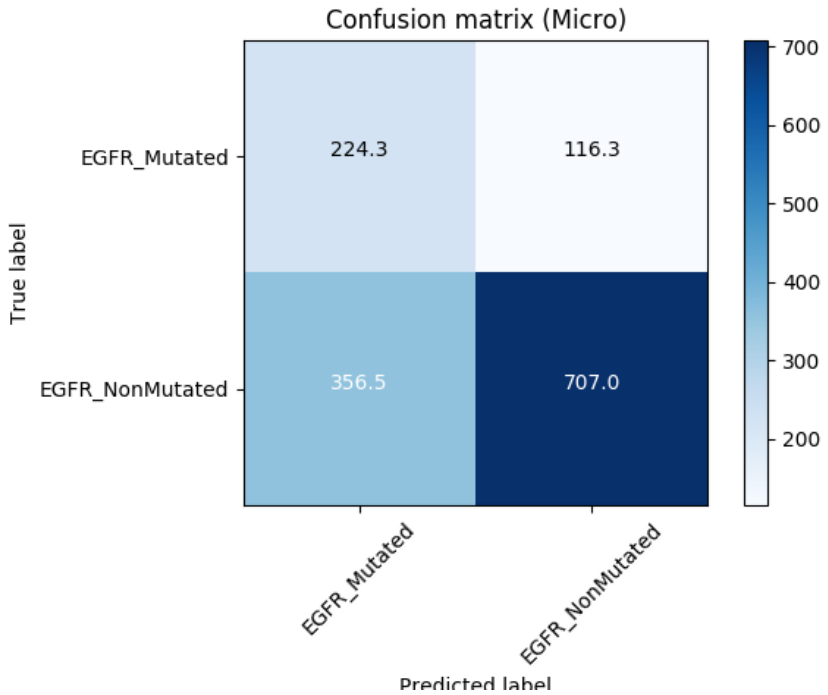


Figure 6.9: Confusion Matrix for Logistic Regression

Chapter 7

Epilogue

7.1 Synopsis and Conclusions

According to the World Health Organization [80], Lung Cancer is the first most common cancer related cause of death worldwide. In 2015, it was responsible for 1.69 million deaths alone. Therefore, the need for extensive research for new and better ways of diagnosis and treatment is imperative.

The computational analysis of histopathology tumor slides is a relatively new field of research that, given the latest advanced computational resources available, can be a catalyst for exploring new aspects of the disease and leading to advanced types of treatment.

For this thesis, we considered as basis the approved targeted therapies of lung cancer that were build around the EGFR gene mutation and tried to explore whether the impact that the mutation caused leads eventually to a texture change of the histopathological tumor opsis.

After trying two different methods, Deep Learning and Image Digital Filtering Analysis, we can be confident that the presence of an EGFR mutation within the cells and the effects in the cells' function that this change brings, leads to a computationally recognizable change of the tumors' texture optical view.

We also tested the detection of the LUAD subtypes within the histopathology images, which was successful for one of the three subtypes, the TRU subtype. Due to the fact that TRU is proven to be highly enriched in the EGFR mutation, we tested the relationship between the texture features detected in each case and proved that they overlap but don't exactly identify with each other.

7.2 Future Work

There is a lot of potential for future work regarding this topic and many different possible directions that could be taken :

- First and foremost, the analysis we performed for the EGFR gene can be performed for each gene mutation that presents a medical interest, especially those that are related to targeted gene therapies.
- Another area of research is testing of other machine learning methods or use of the same methods with different configurations.
- Suggested is the use of other texture filtering techniques, both in terms of new filter algorithms as well as modification of the current filters, in order to produce multiple vectors and then keep the ones contributing the most.

- Also, further analysis can be made in the results of the texture filters' process. It would be interesting to explore the exact texture features that differentiate the two classes and convert them to a human recognizable form. In that way they can be easily understood by the medical community and finally used more efficiently in the process of diagnosis.

Bibliography

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

[2] Waseem Rawat and Zenghui Wang. Deep convolutional neural networks for image classification: A comprehensive review. Neural Computation, 29(9):2352–2449, sep 2017.

[3] Leonardo Araujo dos Santos. Googlenet.

[4] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jonathon Shlens, and Zbigniew Wojna. Rethinking the inception architecture for computer vision. CoRR, abs/1512.00567, 2015.

[5] Frank Schoonjans. Roc curve analysis with medcalc, Sep 2018.

[6] Genomic data commons data portal. https://portal.gdc.cancer.gov/.

[7] Nicolas Coudray, Paolo Santiago Ocampo, Theodore Sakellaropoulos, Navneet Narula, Matija Snuderl, David Fenyö, Andre L. Moreira, Narges Razavian, and Aristotelis Tsirigos. Classification and mutation prediction from non-small cell lung cancer histopathology images using deep learning. Nature Medicine, 2018.

[8] American cancer society. https://www.cancer.org/.

[9] Anthony Griffiths. The Molecular Basis of Mutation.

[10] David A Gutman, Jake Cobb, Dhananjaya Somanna, Yuna Park, Fusheng Wang, Tahsin Kurc, Joel H Saltz, Daniel J Brat, Lee A D Cooper, and Jun Kong. Cancer digital slide archive: an informatics resource to support integrated in silico analysis of TCGA pathology data. Journal of the American Medical Informatics Association, 20(6):1091–1098, nov 2013.

[11] Gillian Bethune, Drew Bethune, Neale Ridgway, and Zhaolin Xu. Epidermal growth factor receptor (egfr) in lung cancer: an overview and update. J Thorac Dis, 2(1):48–51, Mar 2010. jtd-02-01-048[PII].

[12] RYE CONNIE. Biology. OpenStax, 2017.

[13] Bruce Alberts, Dennis Bray, Karen Hopkin, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, and Peter Walter. Essential cell biology. Garland Science, 2013.

[14] Eric A. Collisson, Joshua D. Campbell, Angela N. Brooks, Alice H. Berger, William Lee, Juliann Chmielecki, David G. Beer, Leslie Cope, Chad J. Creighton, Ludmila Danilova, Li Ding, Gad Getz, Peter S. Hammerman, D. Neil Hayes, Bryan Hernandez, James G. Herman, John V. Heymach, Igor Jurisica, Raju Kucherlapati, David Kwiatkowski, Marc Ladanyi, Gordon Robertson, Nikolaus Schultz, Ronglai Shen, Rileen Sinha, Carrie Sougnez, Ming-Sound Tsao, William D. Travis, John N. Weinstein, Dennis A. Wigle, Matthew D. Wilkerson, Andy Chu, Andrew D. Cherniack, Angela Hadjipanayis, Mara Rosenberg, Daniel J. Weisenberger, Peter W. Laird, Amie Radenbaugh, Singer Ma, Joshua M. Stuart, Lauren Averett Byers, Stephen B. Baylin, Ramaswamy Govindan, Matthew Meyerson, Mara Rosenberg, Stacey B. Gabriel, Kristian Cibulskis, Carrie Sougnez, Jaegil Kim, Chip Stewart, Lee Lichtenstein, Eric S. Lander, Michael S.

Lawrence, Gad Getz, Cyriac Kandoth, Robert Fulton, Lucinda L. Fulton, Michael D. McLellan, Richard K. Wilson, Kai Ye, Catrina C. Fronick, Christopher A. Maher, Christopher A. Miller, Michael C. Wendl, Christopher Cabanski, Li Ding, Elaine Mardis, Ramaswamy Govindan, Chad J. Creighton, David Wheeler, Miruna Balasundaram, Yaron S. N. Butterfield, Rebecca Carlsen, Andy Chu, Eric Chuah, Noreen Dhalla, Ranabir Guin, Carrie Hirst, Darlene Lee, Haiyan I. Li, Michael Mayo, Richard A. Moore, Andrew J. Mungall, Jacqueline E. Schein, Payal Sipahimalani, Angela Tam, Richard Varhol, A. Gordon Robertson, Natasja Wye, Nina Thiessen, Robert A. Holt, Steven J. M. Jones, Marco A. Marra, Joshua D. Campbell, Angela N. Brooks, Juliann Chmielecki, Marcin Imielinski, Robert C. Onofrio, Eran Hodis, Travis Zack, Carrie Sougnez, Elena Helman, Chandra Sekhar Pedamallu, Jill Mesirov, Andrew D. Cherniack, Gordon Saksena, Steven E. Schumacher, Scott L. Carter, Bryan Hernandez, Levi Garraway, Rameen Beroukhim, Stacey B. Gabriel, Gad Getz, Matthew Meyerson, Angela Hadjipanayis, Semin Lee, Harshad S. Mahadeshwar, Angeliki Pantazi, Alexei Protopopov, Xiaojia Ren, Sahil Seth, Xingzhi Song, Jiabin Tang, Lixing Yang, Jianhua Zhang, Peng-Chieh Chen, Michael Parfenov, Andrew Wei Xu, Netty Santoso, Lynda Chin, Peter J. Park, Raju Kucherlapati, Katherine A. Hoadley, J. Todd Auman, Shaowu Meng, Yan Shi, Elizabeth Buda, Scot Waring, Umadevi Veluvolu, Donghui Tan, Piotr A. Mieczkowski, Corbin D. Jones, Janae V. Simons, Matthew G. Soloway, Tom Bodenheimer, Stuart R. Jefferys, Jeffrey Roach, Alan P. Hoyle, Junyuan Wu, Saianand Balu, Darshan Singh, Jan F. Prins, J.S. Marron, Joel S. Parker, D. Neil Hayes, Charles M. Perou, Jinze Liu, Leslie Cope, Ludmila Danilova, Daniel J. Weisenberger, Dennis T. Maglinte, Philip H. Lai, Moiz S. Bootwalla, David J. Van Den Berg, Timothy Triche Jr, Stephen B. Baylin, Peter W. Laird, Mara Rosenberg, Lynda Chin, Jianhua Zhang, Juok Cho, Daniel DiCara, David Heiman, Pei Lin, William Mallard, Douglas Voet, Hailei Zhang, Lihua Zou, Michael S. Noble, Michael S. Lawrence, Gordon Saksena, Nils Gehlenborg, Helga Thorvaldsdottir, Jill Mesirov, Marc-Danie Nazaire, Jim Robinson, Gad Getz, William Lee, B. Arman Aksoy, Giovanni Ciriello, Barry S. Taylor, Gideon Dresdner, Jianjiong Gao, Benjamin Gross, Venkatraman E. Seshan, Marc Ladanyi, Boris Reva, Rileen Sinha, S. Onur Sumer, Nils Weinhold, Nikolaus Schultz, Ronglai Shen, Chris Sander, Sam Ng, Singer Ma, Jingchun Zhu, Amie Radenbaugh, Joshua M. Stuart, Christopher C. Benz, Christina Yau, David Haussler, Paul T. Spellman, Matthew D. Wilkerson, Joel S. Parker, Katherine A. Hoadley, Patrick K. Kimes, D. Neil Hayes, Charles M. Perou, Bradley M. Broom, Jing Wang, Yiling Lu, Patrick Kwok Shing Ng, Lixia Diao, Lauren Averett Byers, Wenbin Liu, John V. Heymach, Christopher I. Amos, John N. Weinstein, Rehan Akbani, Gordon B. Mills, Erin Curley, Joseph Paulauskis, Kevin Lau, Scott Morris, Troy Shelton, David Mallery, Johanna Gardner, Robert Penny, Charles Saller, Katherine Tarvin, William G. Richards, Robert Cerfolio, Ayesha Bryant, Daniel P. Raymond, Nathan A. Pennell, Carol Farver, Christine Czerwinski, Lori Huelsenbeck-Dill, Mary Iacocca, Nicholas Petrelli, Brenda Rabeno, Jennifer Brown, Thomas Bauer, Oleg Dolzhanskiy, Olga Potapova, Daniil Rotin, Olga Voronina, Elena Nemirovich-Danchenko, Konstantin V. Fedosenko, Anthony Gal, Madhusmita Behera, Suresh S. Ramalingam, Gabriel Sica, Douglas Flieder, Jeff Boyd, JoEllen Weaver, Bernard Kohl, Dang Huy Quoc Thinh, George Sandusky, Hartmut Juhl, Edwina Duhig, Peter Illei, Edward Gabrielson, James Shin, Beverly Lee, Kristen Rogers, Dante Trusty, Malcolm V. Brock, Christina Williamson, Eric Burks, Kimberly Rieger-Christ, Antonia Holway, Travis Sullivan, Dennis A. Wigle, Michael K. Asiedu, Farhad Kosari, William D. Travis, Natasha Rekhtman, Maureen Zakowski, Valerie W. Rusch, Paul Zippile, James Suh, Harvey Pass, Chandra Goparaju, Yvonne Owusu-Sarpong, John M. S. Bartlett, Sugy Kodeeswaran, Jeremy Parfitt, Harmanjatinder Sekhon, Monique Albert, John Eckman, Jerome B. Myers, Richard Cheney, Carl Morrison, Carmelo Gaudioso, Jeffrey A. Borgia, Philip Bonomi, Mark Pool, Michael J. Liptay, Fedor Moiseenko, Irina Zaytseva, Hendrik Dienemann, Michael Meister, Philipp A. Schnabel, Thomas R. Muley, Martin Peifer, Carmen Gomez-Fernandez, Lynn Herbert, Sophie Egea, Mei Huang, Leigh B. Thorne, Lori Boice, Ashley Hill Salazar, William K. Funkhouser, W. Kimryn Rathmell, Rajiv Dhir, Samuel A. Yousem, Sanja Dacic, Frank Schneider, Jill M. Siegfried,

Richard Hajek, Mark A. Watson, Sandra McDonald, Bryan Meyers, Belinda Clarke, Ian A. Yang, Kwun M. Fong, Lindy Hunter, Morgan Windsor, Rayleen V. Bowman, Solange Peters, Igor Letovanec, Khurram Z. Khan, Mark A. Jensen, Eric E. Snyder, Deepak Srinivasan, Ari B. Kahn, Julien Baboud, David A. Pot, Kenna R. Mills Shaw, Margi Sheth, Tanja Davidsen, John A. Demchok, Liming Yang, Zhining Wang, Roy Tarnuzzer, Jean Claude Zenklusen, Bradley A. Ozenberger, Heidi J. Sofia, William D. Travis, Richard Cheney, Belinda Clarke, Sanja Dacic, Edwina Duhig, William K. Funkhouser, Peter Illei, Carol Farver, Natasha Rekhtman, Gabriel Sica, James Suh, and Ming-Sound Tsao. Comprehensive molecular profiling of lung adenocarcinoma. Nature, 511(7511):543–550, jul 2014.

[15] Robert L. Keith. Lung carcinoma: Tumors of the lungs.

[16] Bryan A. Chan and Brett G.M. Hughes. Targeted therapy for non-small cell lung cancer: current standards and the promise of the future. Translational Lung Cancer Research, 4(1), 2014.

[17] The cancer genome atlas. https://cancergenome.nih.gov/.

[18] A. H. Fischer, K. A. Jacobson, J. Rose, and R. Zeller. Hematoxylin and eosin staining of tissue and cell sections. Cold Spring Harbor Protocols, 2008(6):pdb.prot4986–pdb.prot4986, may 2008.

[19] M.N. Gurcan, L.E. Boucheron, A. Can, A. Madabhushi, N.M. Rajpoot, and B. Yener. Histopathological image analysis: A review. IEEE Reviews in Biomedical Engineering, 2:147–171, 2009.

[20] Zhong Wang, Mark Gerstein, and Michael Snyder. RNA-seq: a revolutionary tool for transcriptomics. Nature Reviews Genetics, 10(1):57–63, jan 2009.

[21] Mark Schena, Dari Shalon, Ronald W. Davis, and Patrick O. Brown. Quantitative monitoring of gene expression patterns with a complementary dna microarray. Science, 270(5235):467–470, 1995.

[22] Simon Anders, Paul Theodor Pyl, and Wolfgang Huber. Htseq—a python framework to work with high-throughput sequencing data. Bioinformatics, 31(2):166–169, 2015.

[23] Matthew D Wilkerson, Xiaoying Yin, Vonn Walter, Ni Zhao, Christopher R Cabanski, Michele C Hayward, C Ryan Miller, Mark A Socinski, Alden M Parsons, Leigh B Thorne, et al. Differential pathogenesis of lung adenocarcinoma subtypes involving sequence mutations, copy number, chromosomal instability, and methylation. PloS one, 7(5):e36530, 2012.

[24] Arindam Bhattacharjee, William G Richards, Jane Staunton, Cheng Li, Stefano Monti, Priya Vasa, Christine Ladd, Javad Beheshti, Raphael Bueno, Michael Gillette, et al. Classification of human lung carcinomas by mrna expression profiling reveals distinct adenocarcinoma subclasses. Proceedings of the National Academy of Sciences, 98(24):13790–13795, 2001.

[25] D Neil Hayes, Stefano Monti, Giovanni Parmigiani, C Blake Gilks, Katsuhiko Naoki, Arindam Bhattacharjee, Mark A Socinski, Charles Perou, and Matthew Meyerson. Gene expression profiling reveals reproducible human lung adenocarcinoma subtypes in multiple independent patient cohorts. Journal of Clinical Oncology, 24(31):5079–5090, 2006.

[26] David G Beer, Sharon LR Kardia, Chiang-Ching Huang, Thomas J Giordano, Albert M Levin, David E Misek, Lin Lin, Guoan Chen, Tarek G Gharib, Dafydd G Thomas, et al. Gene-expression profiles predict survival of patients with lung adenocarcinoma. Nature medicine, 8(8):816, 2002.

[27] William D Travis, Elisabeth Brambilla, Masayuki Noguchi, Andrew G Nicholson, Kim R Geisinger, Yasushi Yatabe, David G Beer, Charles A Powell, Gregory J Riely, Paul E Van Schil, et al. International association for the study of lung cancer/american thoracic society/european

respiratory society international multidisciplinary classification of lung adenocarcinoma. Journal of thoracic oncology, 6(2):244–285, 2011.

[28] Michael R. Peterson, Zhe Piao, Lyudmila A. Bazhenova, Noel Weidner, and Eunhee S. Yi. Terminal respiratory unit type lung adenocarcinoma is associated with distinctive EGFR immunoreactivity and EGFR mutations. Applied Immunohistochemistry & Molecular Morphology, 15(3):242–247, sep 2007.

[29] Cancer digital slide archive. http://cancer.digitalslidearchive.net/.

[30] Robert L. Grossman, Allison P. Heath, Vincent Ferretti, Harold E. Varmus, Douglas R. Lowy, Warren A. Kibbe, and Louis M. Staudt. Toward a shared vision for cancer genomic data. New England Journal of Medicine, 375(12):1109–1112, sep 2016.

[31] Rafael C Gonzalez and Richard E Woods. Thresholding. Digital Image Processing, pages 595–611, 2002.

[32] Jamie Ludwig. Image convolution. http://web.pdx.edu/~jduh/courses/Archive/geog481w07/Students/Ludwig_ImageConvolution.pdf.

[33] Rajesh Rao. Computer vision (uw cse 455).

[34] Nourhan Zayed and Heba A. Elnemr. Statistical analysis of haralick texture features to discriminate lung abnormalities. International Journal of Biomedical Imaging, 2015:1–7, 2015.

[35] Robert M Haralick, Karthikeyan Shanmugam, et al. Textural features for image classification. IEEE Transactions on systems, man, and cybernetics, (6):610–621, 1973.

[36] Amara Graps. An introduction to wavelets. IEEE Comput. Sci. Eng., 2(2):50–61, June 1995.

[37] Yves Meyer. Wavelets and operators, volume 1. Cambridge university press, 1995.

[38] Dipalee Gupta and Siddhartha Choubey. Discrete wavelet transform for image processing. International Journal of Emerging Technology and Advanced Engineering, 4(3):598–602, 2015.

[39] G Lee, R Gommers, F Wasilewski, K Wohlfahrt, A O'Leary, and Nahrstaedt H. Pywavelets - wavelet transforms in python, 2006.

[40] Jason Corso. Linear filters and image processing.

[41] Dong-Chen He and Li Wang. Texture unit, texture spectrum, and texture analysis. IEEE transactions on Geoscience and Remote Sensing, 28(4):509–512, 1990.

[42] Li Wang and Dong-Chen He. Texture classification using texture spectrum. Pattern Recognition, 23(8):905–910, 1990.

[43] Fractals and the Fractal Dimension.

[44] Charu C Aggarwal. Data classification: algorithms and applications. CRC Press, 2014.

[45] B Yegnanarayana. Artificial neural networks. PHI Learning Pvt. Ltd., 2009.

[46] Christos Stergiou and Dimitrios Siganos. Neural networks.

[47] Michael A. Nielsen. Neural Networks and Deep Learning. Determination Press, 2015.

[48] L. Breiman, J. Friedman, C.J. Stone, and R.A. Olshen. Classification and Regression Trees. The Wadsworth and Brooks-Cole statistics-probability series. Taylor & Francis, 1984.

[49] Prashant Gupta. Decision trees in machine learning – towards data science, May 2017.

[50] Laurent Hyafil and Ronald L. Rivest. Constructing optimal binary decision trees is NP-complete. Information Processing Letters, 5(1):15–17, may 1976.

[51] Chao-Ying Joanne Peng, Kuk Lida Lee, and Gary M. Ingersoll. An introduction to logistic regression analysis and reporting. The Journal of Educational Research, 96(1):3–14, sep 2002.

[52] Saishruthi Swaminathan. Logistic regression - detailed overview – towards data science, Mar 2018.

[53] DW Hosmer and S Lemeshow. Applied logistic regression., 2nd edn.(wiley: New york.). NY, USA, 2000.

[54] H. He and E. A. Garcia. Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering, 21(9):1263–1284, Sept 2009.

[55] M. Mostafizur Rahman and D. N. Davis. Addressing the class imbalance problem in medical datasets. International Journal of Machine Learning and Computing, pages 224–228, 2013.

[56] Jürgen Schmidhuber. Deep learning in neural networks: An overview. Neural Networks, 61:85–117, jan 2015.

[57] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. Nature, 521(7553):436–444, may 2015.

[58] Marc' Aurelio Ranzato, Y-Lan Boureau, and Yann LeCun. Sparse feature learning for deep belief networks. In Proceedings of the 20th International Conference on Neural Information Processing Systems, NIPS'07, pages 1185–1192, USA, 2007. Curran Associates Inc.

[59] Min Lin, Qiang Chen, and Shuicheng Yan. Network in network. CoRR, abs/1312.4400, 2013.

[60] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, jun 2015.

[61] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. International Journal of Computer Vision (IJCV), 115(3):211–252, 2015.

[62] Stephen V. Stehman. Selecting and interpreting measures of thematic classification accuracy. Remote Sensing of Environment, 62(1):77–89, oct 1997.

[63] Tom Fawcett. An introduction to ROC analysis. Pattern Recognition Letters, 27(8):861–874, jun 2006.

[64] Bethan Yates, Bryony Braschi, Kristian A. Gray, Ruth L. Seal, Susan Tweedie, and Elspeth A. Bruford. Genenames.org: the HGNC and VGNC resources in 2017. Nucleic Acids Research, 45(D1):D619–D625, oct 2016.

[65] D. Maglott, J. Ostell, K. D. Pruitt, and T. Tatusova. Entrez gene: gene-centered information at NCBI. Nucleic Acids Research, 39(Database):D52–D57, nov 2010.

[66] Nearest centroid predictor. http://cancer.unc.edu/nhayes/publications/adenocarcinoma.2012/wilkerson.2012.LAD.predic.

[67] Adam Goode, Benjamin Gilbert, Jan Harkes, Drazen Jukic, and Mahadev Satyanarayanan. Openslide: A vendor-neutral software foundation for digital pathology. Journal of pathology informatics, 4, 2013.

[68] Tensorflow. tensorflow/models.

[69] Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Yangqing Jia, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.

[70] Geoffrey Hinton, Nitish Srivastava, and Kevin Swersky. Lecture 6a overview of mini-□batch gradient descent.

[71] Gabor filter banks for texture classification. http://scikit-image.org/docs/dev/auto_examples/features_detection/plot_gabor.html.

[72] Stéfan van der Walt, Johannes L. Schönberger, Juan Nunez-Iglesias, François Boulogne, Joshua D. Warner, Neil Yager, Emmanuelle Gouillart, and Tony Yu. scikit-image: image processing in python. PeerJ, 2:e453, jun 2014.

[73] Glcm texture features. http://scikit-image.org/docs/dev/auto_examples/features_detection/plot_glcm.html.

[74] Local binary pattern for texture classification. http://scikit-image.org/docs/dev/auto_examples/features_detection/plot_local_binary_pattern.html.

[75] Gregory R. Lee, Ralf Gommers, Kai Wohlfahrt, Filip Wasilewski, Aaron O'Leary, Holger Nahrstaedt, David Menéndez Hurtado, Thomas Arildsen, Helder Oliveira, Ankit Agrawal, SylvainLan, Michel Pelletier, Matthew Brett, Frank Yu, Daniel M. Pelt, Saket Choudhary, Daniele Tricoli, Check Your Git Settings!, Asnt, Mike DePalatis, Michael Marino, Mark Harfouche, Jonathan Dan, Jakirkham, Jacopo Antonello, Dawid Laszuk, Daniel Goertzen, Balint Reczey, and 0-Tree. Pywavelets/pywt: Pywavelets v1.0.0, 2018.

[76] Ingrid Daubechies. Ten Lectures on Wavelets. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA, 1992.

[77] 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.

[78] Guillaume Lemaître, Fernando Nogueira, and Christos K. Aridas. Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning. Journal of Machine Learning Research, 18(17):1–5, 2017.

[79] C. Bartholomew, L. Eastlake, P. Dunn, and D. Yiannakis. EGFR targeted therapy in lung cancer; an evolving story. Respiratory Medicine Case Reports, 20:137–140, 2017.

[80] World health organization. http://www.who.int/news-room/fact-sheets/detail/cancer.