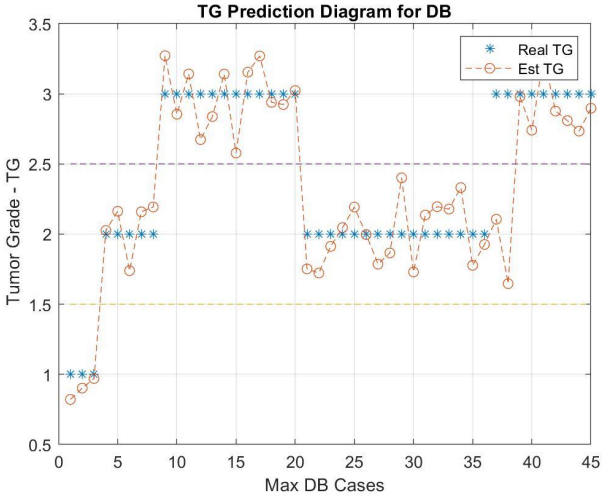
ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

ΑΝΑΛΥΣΗ ΙΑΤΡΙΚΗΣ ΕΙΚΟΝΑΣ ΑΠΟ ΜΑΣΤΟΓΡΑΦΙΕΣ ΜΕ ΧΡΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ



ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΝΤΩΝΙΟΣ ΝΤΙΜΠ

Επιβλέπων: **Δημήτριος - Διονύσιος Κουτσούρης**
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπων: **Δρ. Ουρανία Πετροπούλου**
Ε.ΔΙ.Π. Ε.Μ.Π.

Αθήνα, Φεβρουάριος 2022

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ
ΣΧΟΛΗ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ
ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ
ΤΟΜΕΑΣ ΣΥΣΤΗΜΑΤΩΝ ΜΕΤΑΔΟΣΗΣ ΠΛΗΡΟΦΟΡΙΑΣ
ΚΑΙ ΤΕΧΝΟΛΟΓΙΑΣ ΥΛΙΚΩΝ

ΑΝΑΛΥΣΗ ΙΑΤΡΙΚΗΣ ΕΙΚΟΝΑΣ ΑΠΟ ΜΑΣΤΟΓΡΑΦΙΕΣ ΜΕ ΧΡΗΣΗ ΑΛΓΟΡΙΘΜΩΝ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

ΑΝΤΩΝΙΟΣ ΝΤΙΜΠ

Επιβλέπων: **Δημήτριος - Διονύσιος Κουτσούρης**
Καθηγητής Ε.Μ.Π.

Συνεπιβλέπων: **Δρ. Ουρανία Πετροπούλου**
Ε.ΔΙ.Π. Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 23η Φεβρουαρίου 2022.

.....	.....	.....
Δημήτριος - Διονύσιος Κουτσούρης	**Γιώργος Ματσόπουλος**	**Παναγιώτης Τσανάκας**
Καθηγητής Ε.Μ.Π	Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π

Αθήνα, Φεβρουάριος 2022

ΑΝΤΩΝΙΟΣ ΝΤΙΜΠ

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

Εικόνα Εξωφύλλου

Παρουσίαση διαγράμματος εκτιμώμενων τιμών για τον ιστολογικό δείκτη Tumor Grade και των πραγματικών.

Διάγραμμα που εξήχθη μετά την εκτέλεση του αλγορίθμου σε MATLAB για την πρόβλεψη του ιστολογικού δείκτη Tumor Grade.

Περίληψη

Για την απεικόνιση ύποπτων βλαβών σε ασθενείς με όγκους στο μαστό χρησιμοποιούνται οι μαγνητικές τομογραφίες. Οι θεράποντες ιατροί εξάγουν πληροφορίες από τις απεικονίσεις σχετικά με την βλάβη που παρουσιάζεται στο εκάστοτε ασθενή και αφού κάνουν βιοψία στον όγκο αποφαίνονται για τον τύπο του καρκίνου.

Σκοπός της παρούσας διπλωματικής είναι η δημιουργία αλγορίθμων με χρήση μηχανικής μάθησης για την πρόβλεψη ιστολογικών δεικτών για ασθενείς που έχουν καρκίνο του μαστού ώστε με περαιτέρω ανάπτυξη τους να μπορούν οι θεράποντες ιατροί να εξάγουν τα αποτελέσματα αυτών των δεικτών χωρίς να εκτελούν την βιοψία. Λαμβάνοντας μέρη πληροφοριών που εξήχθησαν από τις μαγνητικές τομογραφίες έγιναν διάφορες προσπάθειες με στόχο την πρόβλεψη αυτών των δεικτών αυτών με όσον το δυνατόν μεγαλύτερη ακρίβεια. Έτσι λοιπόν γράφτηκαν πέντε αλγόριθμοι σε MATLAB κάνοντας χρήση της γραμμικής παρεμβολής αφού πρώτα τα δεδομένα ταξινομήθηκαν με όσο τον δυνατόν πιο αποδοτικό τρόπο ώστε ο αλγόριθμος να εκπαιδευτεί σε δεδομένα με την μεγαλύτερη δυνατή ποικιλία περιστατικών. Αφού έγιναν διάφορες προσπάθειες τα αποτελέσματα ήταν ικανοποιητικά για τους τέσσερις από τους πέντε δείκτες.

Λέξεις κλειδιά: *Μαστογραφία; Μαγνητική Τομογραφία; Μηχανική Μάθηση; Καρκίνος του μαστού; Γραμμική παρεμβολή; MATLAB*

Abstract

MRI scans are used to show suspicious lesions in patients with breast tumors. The treating physicians extract information from the imaging about the damage that occurs in each patient and after doing a biopsy on the tumor, they decide on the type of cancer.

The purpose of this dissertation is to create algorithms using machine learning to predict histological markers for patients with breast cancer so that with their further development the treating physicians can extract the results of these markers without performing the biopsy. Taking parts of the information extracted from the MRI scans, various attempts were made to predict these indicators as accurately as possible. Five algorithms were written in MATLAB using linear interpolation. Before the execution of the linear interpolation the data were sorted in the most efficient way possible so that the algorithm was trained in data with the greatest possible variety of patient cases. After several attempts, the results were satisfactory for four out of five markers.

Keywords: *Mammography; Magnetic resonance; Machine Learning; Breast cancer; Linear interpolation; MATLAB*

ΕΥΧΑΡΙΣΤΙΕΣ

Με την παρούσα διπλωματική κλείνει ο κύκλος σπουδών μου στην σχολή των Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών. Χωρίς την πολύτιμη συμβολή των ανθρώπων που αναγράφονται παρακάτω δεν θα τα είχα καταφέρει, καθώς δεν είναι μόνο το γεγονός ότι η σχολή έχει μεγάλο βαθμό δυσκολίας αλλά η ψυχική μου υγεία αποτέλεσε σοβαρό εμπόδιο στην απόκτηση του διπλώματος.

Για αρχή θα ήθελα να ευχαριστήσω τον κύριο Δημήτριο Κουτσούρη, Καθηγητή Ε.Μ.Π. για την ευκαιρία που μου προσέφερε στην ανάθεση αυτής της πολύ δημιουργικής και εκπαιδευτικής διπλωματικής στο Εργαστήριο Βιοϊατρικής Τεχνολογίας του Εθνικού Μετσόβιου Πολυτεχνείου. Επιπρόσθετα, θα ήθελα να ευχαριστήσω και τον κύριο Γεώργιο Ματσόπουλο, Καθηγητή Ε.Μ.Π. και τον κύριο Παναγιώτη Τσανάκα, Καθηγητή Ε.Μ.Π. για το χρόνο που διέθεσαν στην αξιολόγηση της διπλωματικής μου

Επιπλέον, θα ήθελα να ευχαριστήσω ιδιαίτερα την Δρ. Ουρανία Πετροπούλου, Ε.ΔΙ.Π. Ε.Μ.Π., καθώς και τον Δρ. Παναγιώτη Κατρακάζα, υπεύθυνο ερευνητικών δραστηριοτήτων στην εταιρεία Zelus I.K.E.. Χωρίς την πολύτιμη υποστήριξή τους δεν θα είχα καταφέρει να φτάσω στο τέλος, καθώς δεν με υποστήριξαν μόνο ακαδημαϊκά αλλά και ψυχολογικά σε πολύ δύσκολες στιγμές σαν να με γνώριζαν χρόνια. Αυτοί οι άνθρωποι ήταν δίπλα μου, έδειξαν ανθρωπιά και συμπόνια και έδωσαν μια ευκαιρία σε έναν άνθρωπο που ήταν κυριολεκτικά ‹‹πεσμένος κάτω››.

Ακόμη, θα ήθελα να ευχαριστήσω την κυρία Ευαγγελία Πανουργιά, μόνιμη Λέκτορα της Ιατρικής Σχολής στο Ε.Κ.Π.Α. η οποία όχι μόνο με βοήθησε στην διπλωματική παρέχοντας τα κατάλληλα δεδομένα για την διεκπεραίωση της αλλά είχε την υπομονή να μου εξηγεί διάφορα πράγματα σχετικά με τον καρκίνο του μαστού και με ενέπνευσε δίνοντας μου στόχο και υλικό για να καταφέρω αυτό το αποτέλεσμα.

Επιπρόσθετα, θα ήθελα να ευχαριστήσω από την ψυχή μου τον ψυχίατρό μου Δρ. Ιωάννη Δούμο και την ψυχολόγο μου κυρία Κωνσταντίνα Μυλωνά που δεν με βοήθησαν απλώς να αντιμετωπίσω το πρόβλημά μου αλλά με υποστήριξαν και με έβγαλαν από στιγμές πολύ δύσκολες. Στάθηκαν δίπλα μου και με ‹‹πίστεψαν›› σαν να ήταν φίλοι και συγγενείς μαζί και μοιράστηκα μαζί τους πολλά από τα προβλήματά μου σε σημείο που φοβάμαι ότι μπορεί και να τους έβλαψα. Με συμβούλεψαν και με βγάλανε από καταστάσεις που εύχομαι να μην βιώσει κανείς.

Επίσης, θα ήθελα να ευχαριστήσω ιδιαίτερα τους εξής ανθρώπους τους οποίους θεωρώ μέντορες και φίλους μου: Θεοδώρα Πετρουδάκη, Γεώργιο – Θεόδωρο Στάθη, Ν.Σ., Ε.Χ., Παναγιώτη Μοίρα, Γεώργιο Βογιατζή, Αγγελική Παπαϊωάννου, Βασίλη Μανωλόπουλο,

Νίκο Ορφανό και Νίκο Κελεσίδη οι οποίοι με στηρίξαν απεριόριστα ακαδημαϊκά και ψυχολογικά με όλη τους της υπομονή και με δέχτηκαν με όλα τα αρνητικά του χαρακτήρα μου. Με ενέπνευσαν να προχωρήσω με αυτοπεποίθηση και επιμονή στο στόχο μου.

Ακόμη, θα ήθελα να ευχαριστήσω από την καρδιά μου τον οικογενειακό μου κύκλο και ιδιαίτερα τους γονείς, την αδερφή μου, τις 2 μου θείες, την γιαγιά μου, τον νονό μου, την νονά μου και τα ξαδέρφια μου που ενώ πολλές φορές δεν καταλάβαιναν τι μου συνέβαινε, με βοήθησαν όπως μπορούσαν για να προχωρήσω τις σπουδές μου και να αντιμετωπίσω το πρόβλημα υγείας μου. Το γεγονός ότι με στήριξαν με αγάπη και υπομονή είναι ένα από τα ‹‹καλύτερα φάρμακα›› που έχω λάβει ως παθών.

Τέλος, θα ήθελα να ευχαριστήσω τους συμφοιτητές μου, φίλους μου και την γραμματεία που ήταν εκεί πάντα για να με βοηθήσουν και να με εξυπηρετήσουν.

Εύχομαι, κάποια στιγμή στο μέλλον να βρεθώ στην κατάλληλη θέση να ανταποδώσω όλα αυτά και ακόμα περισσότερα σε όλους.

ΠΡΟΛΟΓΟΣ

Στο πρώτο κεφάλαιο παρουσιάζονται επιγραμματικά κάποιοι αλγόριθμοι και τεχνικές που έχουν χρησιμοποιηθεί σε αλγορίθμους μηχανικής μάθησης σχετικά με την πρόβλεψη δεικτών και αποτελεσμάτων στο καρκίνο του μαστού.

Στο δεύτερο κεφάλαιο γίνεται μια πιο ενδελεχής αναφορά στην έρευνα και ανάπτυξη που έχει γίνει σε αλγορίθμους μηχανικής μάθησης και νευρωνικά δίκτυα για την πρόβλεψη δεικτών και αποτελεσμάτων για διάφορους τύπους καρκίνου.

Στο τρίτο κεφάλαιο αναγράφονται κάποια στοιχεία για την γλώσσα και το περιβάλλον MATLAB καθώς και τις διαφορετικές προσπάθειες που έγιναν στην προσέγγιση του προβλήματος σε Python μαζί με κώδικα. Έπειτα, εξηγούνται οι αλγόριθμοι που γράφτηκαν σε MATLAB και εξάγουν αποτελέσματα με ικανοποιητική ακρίβεια.

Στο τέταρτο κεφάλαιο παρουσιάζονται τα αποτελέσματα με τα ποσοστά ακρίβειας, γραφικές παραστάσεις.

Έπειτα, στο πέμπτο παρουσιάζεται μια σύνοψη της εργασίας, καθώς και οι δυσκολίες που παρουσιάστηκαν και οι μελλοντικές βλέψεις και επεκτάσεις της ανάπτυξης των αλγορίθμων.

Τέλος, στο έκτο και τελευταίο κεφάλαιο παρατίθεται η βιβλιογραφία που χρησιμοποιήθηκε.

Περιεχόμενα

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ

ΕΥΡΕΤΗΡΙΟ ΠΙΝΑΚΩΝ

Κεφάλαιο 1: Εισαγωγή

Κάθε χρόνο, οι παθολόγοι διαγιγνώσκουν 14 εκατομμύρια νέους ασθενείς με καρκίνο σε όλο τον κόσμο. Αυτά είναι εκατομμύρια άνθρωποι που θα αντιμετωπίσουν χρόνια αβεβαιότητας. Οι παθολόγοι πραγματοποιούν διαγνώσεις και προγνώσεις καρκίνου εδώ και δεκαετίες. Οι περισσότεροι παθολόγοι έχουν ποσοστό επιτυχίας 96-98% για τη διάγνωση του καρκίνου. Σύμφωνα με το Πανεπιστημιακό Νοσοκομείο του Όσλο, η ακρίβεια των προγνώσεων είναι μόνο 60% για τους παθολόγους. Η πρόγνωση είναι το μέρος μιας βιοψίας που έρχεται μετά τη διάγνωση του καρκίνου και προβλέπει την ανάπτυξη της νόσου [1], [2] .

Το επόμενο βήμα στην παθολογία είναι η Μηχανική Μάθηση. Η Μηχανική Μάθηση είναι ένας από τους βασικούς κλάδους της Τεχνητής Νοημοσύνης. Είναι ένα σύστημα που λαμβάνει δεδομένα, βρίσκει μοτίβα, εκπαιδεύεται χρησιμοποιώντας τα δεδομένα και βγάζει ένα αποτέλεσμα [2].

Η Μηχανική Μάθηση έχει τα εξής βασικά πλεονεκτήματα:

Πρώτον, οι μηχανές μπορούν να λειτουργήσουν πολύ πιο γρήγορα από τους ανθρώπους. Η βιοψία συνήθως διαρκεί 10 ημέρες από τον Παθολόγο. Ένας υπολογιστής μπορεί να κάνει χιλιάδες βιοψίες μέσα σε λίγα δευτερόλεπτα.

Οι μηχανές μπορούν να κάνουν κάτι στο οποίο οι άνθρωποι δεν είναι τόσο καλοί. Μπορούν να επαναλάβουν αυτό που κάνουν χιλιάδες φορές χωρίς να εξαντληθούν. Μετά από κάθε επανάληψη, το μηχάνημα επαναλαμβάνει τη διαδικασία για να το κάνει καλύτερα. Το κάνουν και οι άνθρωποι αυτό, και το ονομάζουν πρακτική. Ενώ η πρακτική μπορεί να τελειοποιήσει τις ενέργειες, καμία ποσότητα εξάσκησης δεν μπορεί να βάλει έναν άνθρωπο ακόμη και κοντά στην υπολογιστική ταχύτητα ενός υπολογιστή.

Ένα άλλο πλεονέκτημα είναι η μεγάλη ακρίβεια των μηχανών. Με την έλευση της τεχνολογίας του Διαδικτύου των Πραγμάτων, υπάρχουν τόσα πολλά δεδομένα στον κόσμο που οι άνθρωποι δεν μπορούν να τα εξετάσουν όλα. Εκεί μας βοηθούν οι μηχανές. Μπορούν να δουλέψουν πιο γρήγορα από εμάς και να κάνουν ακριβείς υπολογισμούς και να βρουν μοτίβα στα δεδομένα [2].

Σύντομη επεξήγηση των κατηγοριών της μηχανικής μάθησης

Υπάρχουν δύο μεγάλες κατηγορίες Μηχανικής Μάθησης,

- Εποπτευόμενη μάθηση
- Εκμάθηση χωρίς επίβλεψη

Ένας αλγόριθμος εποπτευόμενης μάθησης είναι ένας αλγόριθμος που «διδάσκεται» από τα δεδομένα που του δίνονται. Το μοντέλο εκπαιδεύεται χρησιμοποιώντας δεδομένα με ετικέτα και στη συνέχεια δοκιμάζεται. Αυτό επαναλαμβάνεται μέχρι να επιτευχθεί το βέλτιστο αποτέλεσμα. Μόλις γίνει αυτό, μπορεί να κάνει προβλέψεις για μελλοντικές περιπτώσεις.

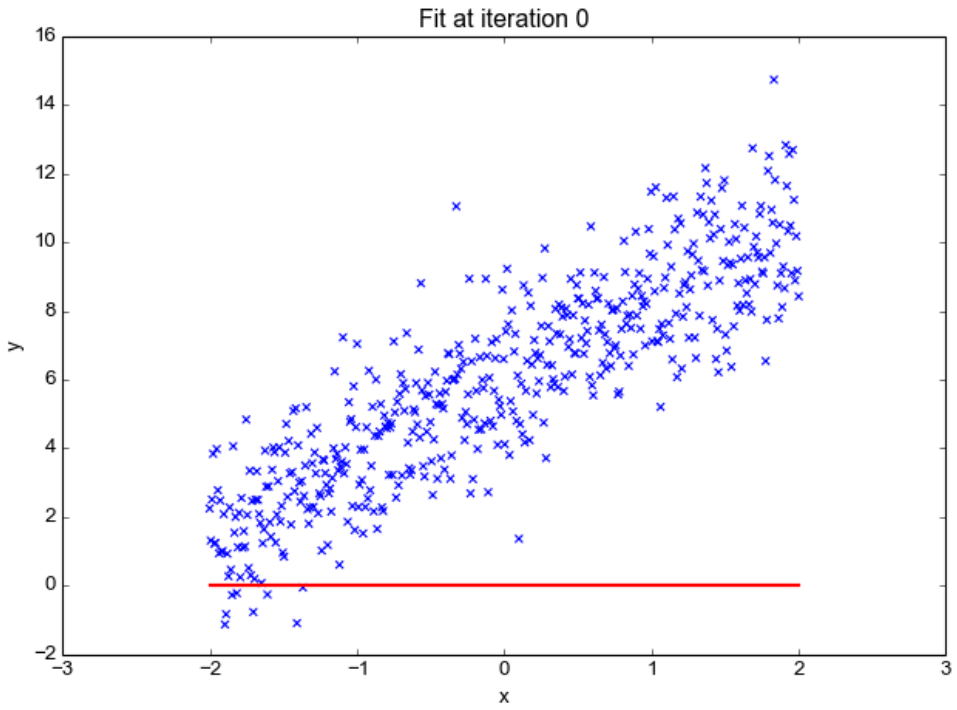
Στη μη εποπτευόμενη μάθηση, τα σύνολα δεδομένων δεν επισημαίνονται. Αντίθετα, είναι δουλειά του μοντέλου να δημιουργήσει μια δομή που ταιριάζει στα δεδομένα βρίσκοντας μοτίβα (όπως ομαδοποιήσεις)[1].

Τεχνικές Μηχανικής Μάθησης

Παλινδρόμηση

Ο κύριος στόχος της παλινδρόμησης είναι να ελαχιστοποιήσει τη συνάρτηση κόστους του μοντέλου. Η συνάρτηση κόστους είναι μια συνάρτηση που υπολογίζει την απόσταση μεταξύ της υπόθεσης για την τιμή x και της πραγματικής τιμής x. Βασικά, δείχνει πόσο μακριά είναι το αποτέλεσμα από την πραγματική απάντηση.

Το όλο θέμα της παλινδρόμησης είναι να βρεθεί μια πολυδιάστατη εξίσωση που ελαχιστοποιεί τη συνάρτηση κόστους για να δημιουργήσει την καλύτερη δυνατή σχέση μεταξύ των σημείων δεδομένων.



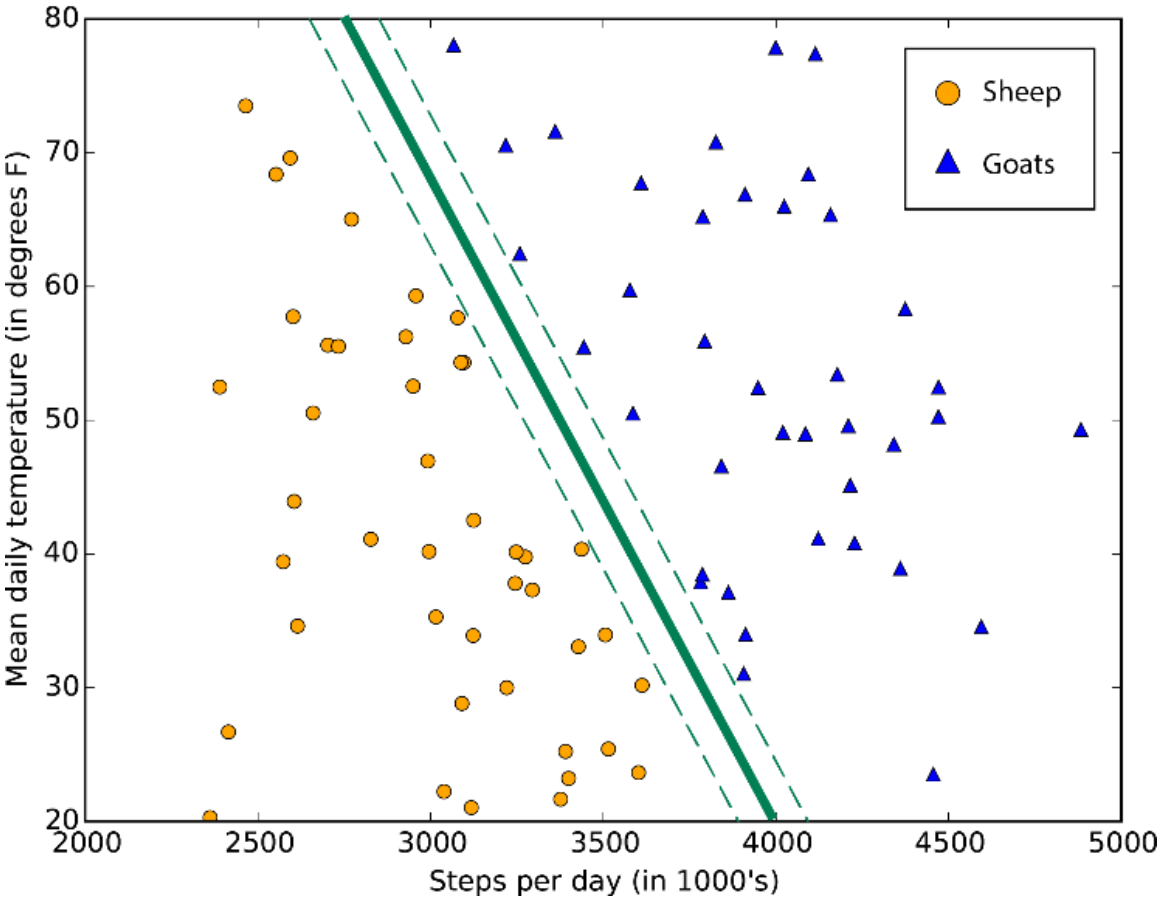
Εικόνα 1: Προσέγγιση των δεδομένων με γραμμική παλινδρόμηση[2]

Ξεκινά με μια τυχαία γραμμή χωρίς συσχέτιση που επαναλαμβάνεται χρησιμοποιώντας κάθοδο βασισμένη στην κλίση (Gradient Descent), που είναι ένας επαναληπτικός αλγόριθμος βελτιστοποίησης πρώτης τάξης που χρησιμοποιείται για την εύρεση ενός τοπικού ελάχιστου/μέγιστου μιας δεδομένης συνάρτησης. Σε αυτόν τον αλγόριθμο, η συνάρτηση κόστους μειώνεται από το μοντέλο που προσαρμόζει τις παραμέτρους του. Καθώς, η κάθοδος βασισμένη στην κλίση μειώνει τη συνάρτηση κόστους όλο και πιο χαμηλά, η προσέγγιση γίνεται όλο και πιο ακριβής [2].

Ταξινόμηση

Τα εποπτευόμενα μοντέλα μάθησης μπορούν να κάνουν περισσότερα από την απλή παλινδρόμηση. Μία από τις πιο χρήσιμες εργασίες της Μηχανικής Μάθησης είναι η ταξινόμηση.

Οι αλγόριθμοι ταξινόμησης δημιουργούν όρια μεταξύ σημείων δεδομένων ταξινομώντας τα ως μια συγκεκριμένη ομάδα, ανάλογα με τα χαρακτηριστικά τους που ταιριάζουν με τις παραμέτρους του μοντέλου.

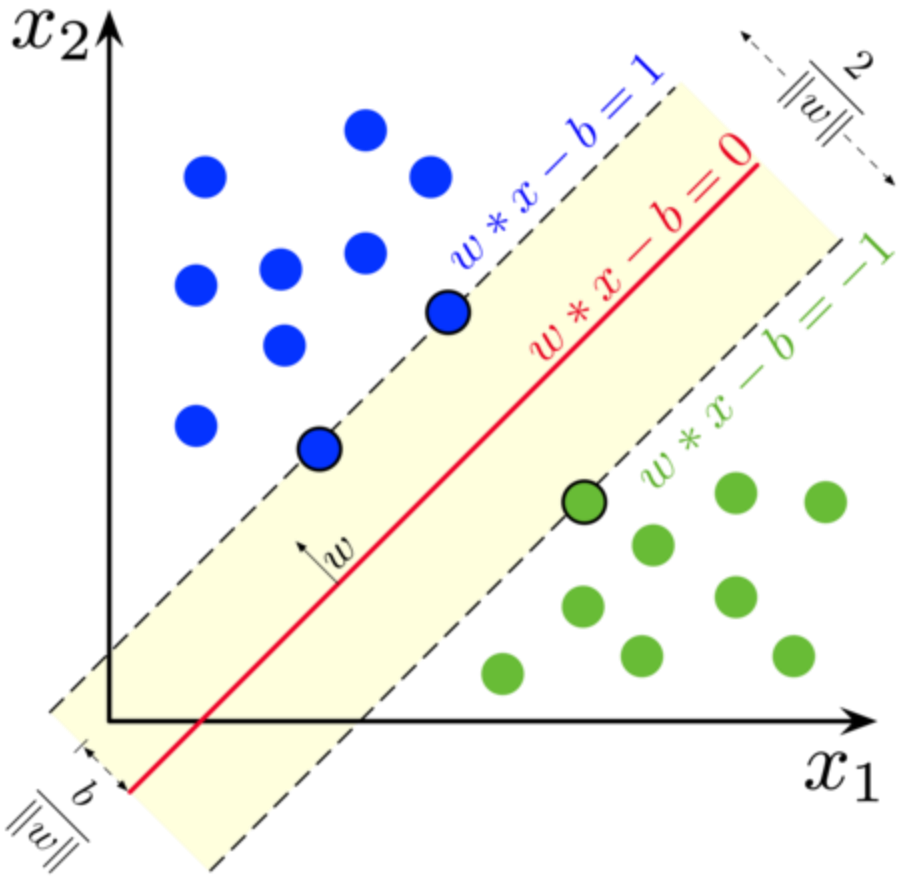


Εικόνα 2: Διαχωρισμός 2 διαφορετικών ζώων ανάλογα με το πόσα βήματα κάνουν σε μια μέρα ανάλογα με την μέση θερμοκρασία της ημέρας [2].

Το όριο μεταξύ των κλάσεων δημιουργείται χρησιμοποιώντας μια διαδικασία που ονομάζεται λογιστική παλινδρόμηση. Η συνάρτηση κόστους επίσης χρησιμοποιείται στην κατηγοριοποίηση [2].

Supported Vector Machines (SVM)

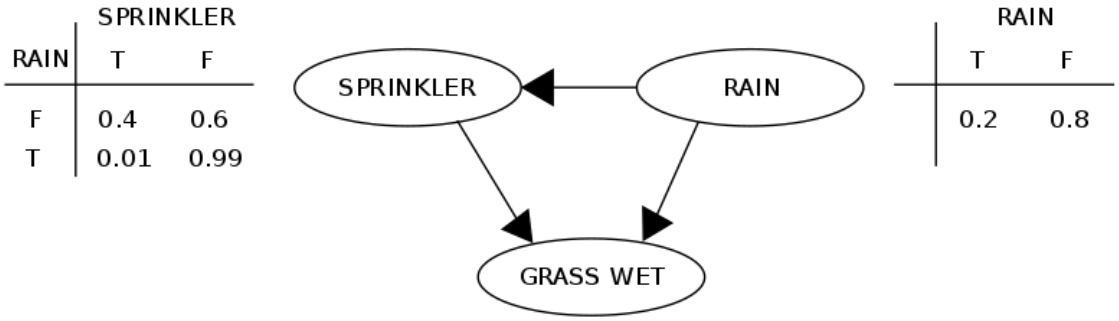
Τα SVM είναι εποπτευόμενοι αλγόριθμοι μάθησης που χρησιμοποιούνται τόσο στην ταξινόμηση όσο και στην παλινδρόμηση. Ο στόχος ενός αλγορίθμου SVM είναι να ταξινομήσει τα δεδομένα δημιουργώντας ένα όριο με το μεγαλύτερο δυνατό περιθώριο μεταξύ του ίδιου και των δεδομένων [2].



Εικόνα 3: Κατηγοριοποίηση δεδομένων από αλγόριθμο SVM [2]

Μπευζιανά Δίκτυα (Bayesian Networks)

Τα Μπευζιανά Δίκτυα είναι ταξινομητές παρόμοιοι με τα δέντρα αποφάσεων. Η διαφορά είναι ότι τα δίκτυα αυτά δείχνουν εκτιμήσεις πιθανοτήτων και όχι προβλέψεις. Το σύνολο δεδομένων των μεταβλητών και οι υπό όρους εξαρτήσεις τους εμφανίζονται σε μια οπτική μορφή που ονομάζεται κατευθυνόμενο ακυκλικό γράφημα (acyclic graph)[2].



RAIN	SPRINKLER: T	SPRINKLER: F
F	0.4	0.6
T	0.01	0.99

RAIN: T	RAIN: F
0.2	0.8

SPRINKLER	RAIN	GRASS WET: T	GRASS WET: F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

Εικόνα 4: Στην εικόνα απεικονίζονται οι δύο λόγοι για τους οποίους το γρασίδι είναι βρεγμένο: είναι είτε από τη βροχή είτε από το ψεκαστήρα. Χρησιμοποιώντας ένα μοντέλο Μπευζιανού Δικτύου, μπορούν να βρεθούν οι πιθανότητες κάθε πιθανού σεναρίου[2].

Κεφάλαιο 2: Βιβλιογραφική Ανασκόπηση

Ορισμός των ιστολογικών δεικτών:

Tumor Grade[1]: Το tumor grade είναι η περιγραφή ενός όγκου που βασίζεται στο πόσο παθολογικά φαίνονται τα καρκινικά κύτταρα και ο ιστός του στο μικροσκόπιο. Είναι ένας δείκτης του πόσο γρήγορα ένας όγκος είναι πιθανό να αναπτυχθεί και να εξαπλωθεί. Εάν τα κύτταρα του όγκου και η οργάνωση του ιστού του είναι κοντά σε αυτά των φυσιολογικών κυττάρων και ιστών, ο όγκος ονομάζεται ‹‹καλά διαφοροποιημένος››. Αυτοί οι όγκοι τείνουν να αναπτύσσονται και να εξαπλώνονται με πιο αργό ρυθμό από τους όγκους που είναι ‹‹αδιαφοροποίητοι›› ή ‹‹κακώς διαφοροποιημένοι››, οι οποίοι έχουν κύτταρα με ανώμαλη εμφάνιση και μπορεί να στερούνται δομών φυσιολογικού ιστού. Με βάση αυτές και άλλες διαφορές στη μικροσκοπική εμφάνιση, οι γιατροί αποδίδουν έναν αριθμητικό «βαθμό» στους περισσότερους καρκίνους. Οι παράγοντες που χρησιμοποιούνται για τον προσδιορισμό του βαθμού του όγκου μπορεί να ποικίλλουν μεταξύ διαφορετικών τύπων καρκίνου.

ER[2]: Οι καρκίνοι του μαστού που έχουν υποδοχείς οιστρογόνων ονομάζονται ER-θετικοί (ή ER+) καρκίνοι.

PR[2]: Οι καρκίνοι του μαστού με υποδοχείς προγεστερόνης ονομάζονται PR-θετικοί (ή PR+) καρκίνοι.

c-erbB2[3]: Υποδοχέας τυροσίνης-πρωτεϊνικής κινάσης erbB-2, γνωστός και ως c-erbB2, είναι μια πρωτεΐνη που βοηθά τα καρκινικά κύτταρα του μαστού να αναπτυχθούν γρήγορα. Τα καρκινικά κύτταρα του μαστού με υψηλότερα από τα φυσιολογικά επίπεδα c-erbB2 ονομάζονται c-erbB2-θετικά. Αυτοί οι καρκίνοι τείνουν να αναπτύσσονται και να εξαπλώνονται πιο γρήγορα από τους καρκίνους του μαστού που είναι αρνητικοί στο c-erbB2, αλλά είναι πολύ πιο πιθανό να ανταποκριθούν στη θεραπεία με φάρμακα που στοχεύουν την πρωτεΐνη c-erbB2. Η c-erbB2 ονομάζεται και αλλιώς HER2.

Ki-67[4]: Το Ki-67 είναι μια πρωτεΐνη στα κύτταρα που αυξάνεται καθώς προετοιμάζονται να διαιρεθούν σε νέα κύτταρα. Μια διαδικασία χρώσης μπορεί να μετρήσει το ποσοστό των καρκινικών κυττάρων που είναι θετικά για Ki-67. Όσο περισσότερα θετικά κύτταρα υπάρχουν, τόσο πιο γρήγορα διαιρούνται και σχηματίζουν νέα κύτταρα. Στον καρκίνο του μαστού, ένα αποτέλεσμα μικρότερο από 10% θεωρείται χαμηλό, 10-20% οριακό και υψηλό εάν είναι μεγαλύτερο από 20%.

---

[1] https://www.cancer.gov/about-cancer/diagnosis-staging/prognosis/tumor-grade-fact-sheet
[2] https://www.cancer.net/research-and-advocacy/asco-care-and-treatment-recommendations-patients/estrogen-and-progesterone-receptor-testing-breast-cancer#:~:text=If%20breast%20cancer%20cells%20have,called%20ER%2FPR%2Dnegative
[3] https://www.cancer.gov/publications/dictionaries/cancer-terms/def/c-erbb-2-positive
[4] https://www.breastcancer.org/symptoms/diagnosis/rate_grade

<u>Επεξήγηση χαρακτηριστικών που χρησιμοποιούνται ως είσοδο στους αλγορίθμους:</u>

Tumor type (τύπος καρκίνου)

- *IDC (Invasive Ductal Carcinoma)*: διηθητικός πορογενής καρκίνος, ο συχνότερος καρκίνος μαστού
- *DCIS (Ductal Carcinoma in Situ)*: ενδοπορικός καρκίνος μαστού, πιο πρώιμος καρκίνος

Morphology (μορφολογία)

- *NME (Non Mass Enhancement)*: μη μαζόμορφη ενίσχυση, ο καρκίνος δε δημιουργεί μάζα άλλα τα καρκινικά κύτταρα εναλλάσσονται με φυσιολογικά
- *MASS*: 3-διάστατη μάζα

Borders (όρια βλάβης)

- Irregular: ανώμαλα όρια
- Spiculated, ακτινωτές προσεκβολές
- *Smooth*: Ομαλά όρια

Tumor size (μέγεθος όγκου)

- < 1.5 cm
- 1.5 – 2.5 cm
- > 2.5 cm

Curve morphology (μορφολογία καμπύλης αιμάτωσης της βλάβης)

- Type 1
- Type 2
- Type 3

Curve type (είδος καμπύλης αιμάτωσης)

- Suspicious (type 2 and 3)
- Benign (type 1)

ADC (Apparent Diffusion Coefficient, συντελεστής διάχυσης της βλάβης)

- Low (χαμηλό σήμα)
- High (υψηλό σήμα)

Σύμφωνα με το Cancer Statistics του 2019, στις ΗΠΑ εκτιμήθηκαν συνολικά 1,762,450 νέα περιστατικά ασθενών με καρκίνο. Περίπου 891,480 αφορούσαν γυναίκες και το 30% αυτών των περιστατικών αφορούσαν ασθενείς με καρκίνο του μαστού[5]. Παρατηρώντας, αυτά τα στατιστικά, είναι φανερό πως εκτός από την έγκαιρη πρόβλεψη μιας ανερχόμενης βλάβης ο ιστολογικός προσδιορισμός ώστε να γίνει η σωστή διάγνωση του τύπου του καρκίνου του μαστού είναι απαραίτητη. Για αυτό το λόγο πολλοί επιστήμονες για την εξαγωγή αυτών των δεδομένων, για την απεικόνιση των ύποπτων όγκων καθώς και την εκτίμηση του τύπου του όγκου στράφηκαν στην ανάπτυξη αλγορίθμων Τεχνητής Νοημοσύνης (Artificial Intelligence). Αυτοί οι αλγόριθμοι χρησιμοποιούν Μηχανική Μάθηση (Machine Learning) καθώς και μεθόδους Βαθιάς Μάθησης (Deep Learning) για την πρόβλεψη αποτελεσμάτων καθώς και την δημιουργία Όρασης από Υπολογιστές (Computer Vision).

Ορισμός Τεχνητής Νοημοσύνης, Μηχανικής Μάθησης, Βαθιά Μάθησης και Όραση Υπολογιστών

- **Τεχνητή Νοημοσύνη** είναι το πεδίο της επιστήμης που συνδυάζει την επιστήμη των υπολογιστών και εκτενή σετ από δεδομένα με σκοπό την επίλυση των προβλημάτων[6]. Είναι δηλαδή η επιστήμη εκείνη που με την χρήση αλγορίθμων και δεδομένων μπορεί να ενεργοποιήσει ένα πρόγραμμα να μαθαίνει, να αντιλαμβάνεται και να επιλύει ένα πρόβλημα με τρόπο τέτοιο σαν να μιμείται την ανθρώπινη σκέψη.
- **Μηχανική Μάθηση** είναι ο κλάδος της Τεχνητής Νοημοσύνης και της επιστήμης των υπολογιστών που επικεντρώνεται στην χρήση των δεδομένων και των αλγορίθμων με σκοπό την μίμηση του τρόπου που μαθαίνουν οι άνθρωποι και σταδιακά, στο πρόγραμμα που εφαρμόζεται, να βελτιώνει την ακρίβεια των αποτελεσμάτων του στην επίλυση ενός προβλήματος[7].
- **Βαθιά Μάθηση** είναι ένα υποσύνολο της Μηχανικής Μάθησης, που πρακτικά είναι ένα νευρωνικό δίκτυο 3 ή παραπάνω στρωμάτων. Αυτά τα νευρωνικά δίκτυα προσπαθούν να προσομοιώσουν την συμπεριφορά του ανθρώπινου εγκεφάλου με σκοπό την εκμάθηση από τον αλγόριθμο μεγάλων ποσοτήτων δεδομένων. Ουσιαστικά δηλαδή τα νευρωνικά δίκτυα είναι δίκτυα που αποτελούνται από στρώματα με κόμβους. Τα στρώματα αποτελούνται από 3 μέρη: το στρώμα εισόδου (input layer), ένα ή περισσότερα κρυμμένα στρώματα (hidden layers) και το στρώμα εξόδου (output layers). Κάθε κόμβος συνδέεται με έναν άλλο και υπάρχει κάποιο βάρος και κάποιο κατώφλι. Εάν η έξοδος ενός κόμβου είναι μεγαλύτερη από το προκαθορισμένο κατώφλι τότε ο κόμβος ενεργοποιείται και τα δεδομένα περνούν στο επόμενο στρώμα. Διαφορετικά, δεν περνούν δεδομένα[89].

[5] https://www.cancer.org/research/cancer-facts-statistics/all-cancer-facts-figures/cancer-facts-figures-2019.html

[6] https://www.ibm.com/cloud/learn/what-is-artificial-intelligence

[7] https://www.ibm.com/cloud/learn/machine-learning

[8] https://www.ibm.com/cloud/learn/deep-learning

[9] https://www.ibm.com/cloud/learn/neural-networks

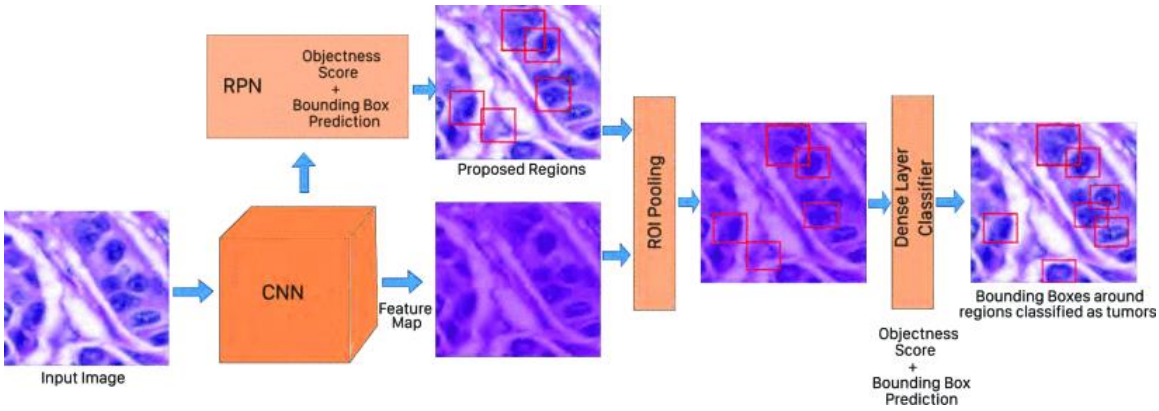
- **Όραση υπολογιστών** είναι το πεδίο της Τεχνητής Νοημοσύνης που ενεργοποιεί τους υπολογιστές και τα υπολογιστικά συστήματα να εξάγουν χρήσιμες πληροφορίες από ψηφιακές εικόνες, βίντεο και άλλες οπτικές εισόδους. Έπειτα από την εξαγωγή των δεδομένων αναλαμβάνει την επίλυση του προβλήματος ή δίνει συστάσεις για την επίλυση αυτού[10].

Παρακάτω θα γίνει παρουσίαση τέτοιων αλγορίθμων που χρησιμοποιούν τεχνολογίες από τις προαναφερθείσες τεχνικές και επιστημονικούς τομείς με σκοπό τον εντοπισμό ύποπτων βλαβών σε ιατρικές απεικονίσεις ή την προσπάθεια διάγνωσης του τύπου του όγκου:

Εντοπισμός όγκου σε ιστοπαθολογικές εικόνες μαστών με την χρήση Faster R-CNN.

Οι δημιουργοί του αλγορίθμου [3] χρησιμοποιήσαν ένα σετ δεδομένων από το Breast Cancer Histopathological Annotation and Diagnosis (BreCaHAD) [4] που αποτελείται από 162 εικόνες που σχολιάζονται για διαφορετικές κατηγορίες από παθολόγους. Οι εικόνες είναι ανάλυσης 1360x1024 και ένδειξη παρουσίας του όγκου φαίνεται με μπλε κουκίδες. Για τον εντοπισμό των όγκων ο αλγόριθμος Faster R-CNN χρησιμοποιήθηκε έναντι των R-CNN και Fast-CNN. Ο λόγος που χρησιμοποιήθηκε ο Faster-CNN είναι διότι ο R-CNN αλγόριθμος χρησιμοποιεί το ConvNet για τον εντοπισμό αντικειμένων με αποτέλεσμα τα εξαγόμενα χαρακτηριστικά να καταλαμβάνουν μεγαλύτερη χωρητικότητα και η εκπαίδευση του αλγορίθμου καθώς και ο εντοπισμός των όγκων να απαιτούν περισσότερο χρόνο. Ο Fast R-CNN επιλύει πολλά από τα προβλήματα του R-CNN, καθώς δεν απαιτεί την αποθήκευση των εξαγόμενων χαρακτηριστικών, εντοπίζει τους όγκους με μεγαλύτερη ταχύτητα και ακρίβεια και η εκπαίδευση του είναι single-stage με αποτέλεσμα να απαιτεί λιγότερο χρόνο. Ο Faster R-CNN ουσιαστικά αποδίδει καλύτερα σε σχέση με τους άλλους δυο αλγορίθμους καθώς δεν χρησιμοποιεί επιλεκτική αναζήτηση για την δημιουργία περιοχών εντοπισμού. Αντ' αυτού χρησιμοποιεί RPN (Regional Proposal Networks) και δημιουργεί ένα συγκεκριμένο αριθμό περιοχών προς οριοθέτηση. Εν συνεχεία, το RPN μεταβιβάζει τα δεδομένα στο ROI pooling εφαρμόζει τις οριοθετήσεις γύρω από το όγκους που προτείνει το ConvNet. Τέλος, η έξοδος του ROI Pooling δίνεται ως είσοδος στο πλήρως Ενοποιημένο Πυκνό Στρώμα (Fully Connected Dense Layer) που στην έξοδο αυτού του σταδίου γίνεται και ο εντοπισμός του όγκου. Παρακάτω φαίνεται και ένα διάγραμμα με την ροή που ακολουθεί ο αλγόριθμος:

[10] https://www.ibm.com/se-en/topics/computer-vision



Εικόνα 5: Ροή λειτουργίας του Faster R-CNN [3].

Τα αποτελέσματα του αλγορίθμου φαίνονται στον παρακάτω πίνακα:

Image Names	Precision				Sensitivity				IoU				F-score			
	L	LN	S	SN	L	LN	S	SN	L	LN	S	SN	L	LN	S	SN
Case 1-06	0.17	0.33	0.52	0.51	0.01	0.01	0.70	0.77	0.50	0.60	0.72	0.69	0.02	0.02	0.60	0.61
Case 1-07	0.25	0	0.50	0.43	0.01	0	0.50	0.45	0.52	0	0.70	0.68	0.02	0	0.50	0.44
Case 1-08	0	0	0.48	0.38	0	0	0.67	0.63	0	0	0.69	0.68	0	0	0.56	0.48
Case 2-06	0.50	0	0.36	0.26	0.03	0	0.57	0.54	0.55	0	0.72	0.68	0.05	0	0.44	0.36
Case 2-08	1	1	0.62	0.59	0.01	0.06	0.63	0.67	0.55	0.62	0.75	0.71	0.03	0.11	0.62	0.62
Case 3-08	0	0.20	0.33	0.25	0	0.02	0.65	0.54	0	0.56	0.70	0.76	0	0.04	0.43	0.34
Case 3-10	0	0	0.22	0.22	0	0	0.47	0.47	0	0	0.71	0.77	0	0	0.30	0.30
Case 4-08	0	0	0.55	0.51	0	0	0.57	0.60	0	0	0.67	0.67	0	0	0.56	0.55
Case 9-07	0.25	0.25	0.42	0.32	0.02	0.02	0.75	0.64	0.55	0.58	0.70	0.70	0.03	0.03	0.54	0.43
Case 13-11	0.25	0	0.35	0.30	0.03	0	0.54	0.56	0.54	0	0.68	0.68	0.05	0	0.43	0.39
Max.	1	1	0.62	0.59	0.03	0.06	0.75	0.77	0.55	0.62	0.75	0.77	0.05	0.11	0.62	0.62
Min.	0	0	0.22	0.22	0	0	0.47	0.45	0	0	0.67	0.67	0	0	0.30	0.30
Mean	0.24	0.18	0.43	0.38	0.01	0.01	0.60	0.59	0.32	0.24	0.71	0.70	0.02	0.02	0.50	0.45
Std.	0.31	0.32	0.12	0.13	0.01	0.02	0.09	0.09	0.27	0.30	0.02	0.04	0.02	0.03	0.10	0.11

Εικόνα 6: Εκτίμηση του Faster R-CNN μοντέλου [3]

Υπόμνημα: L: εκπαίδευση με μεγάλου μεγέθους εικόνες (1360x1024 pixels), LN: εκπαίδευση με μεγάλου μεγέθους εικόνες (1000x1000 pixels) και κανονικοποιημένες ως προς το χρώμα, S: εκπαίδευση με μικρές εικόνες (256x256 pixels), SN: εκπαίδευση με μικρού μεγέθους εικόνες (256x256 pixels) και κανονικοποιημένες ως προς το χρώμα.

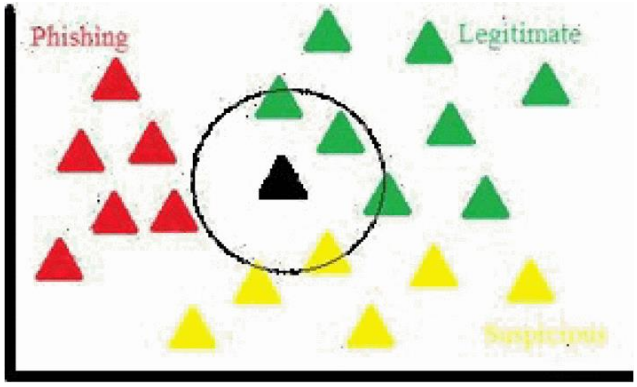
Εντοπισμός του καρκίνου του μαστού και πρόβλεψη του με Μηχανική Μάθηση.

Η μεθοδολογία των ερευνητών στο [5] ήταν να χρησιμοποιήσουν 3 είδη γνωστούς αλγορίθμους για την εξαγωγή αποτελεσμάτων:

- KNN (K-Nearest Neighbors)
- CNN (Convolutional Neural Network)
- SVM (Support-Vector Machine)

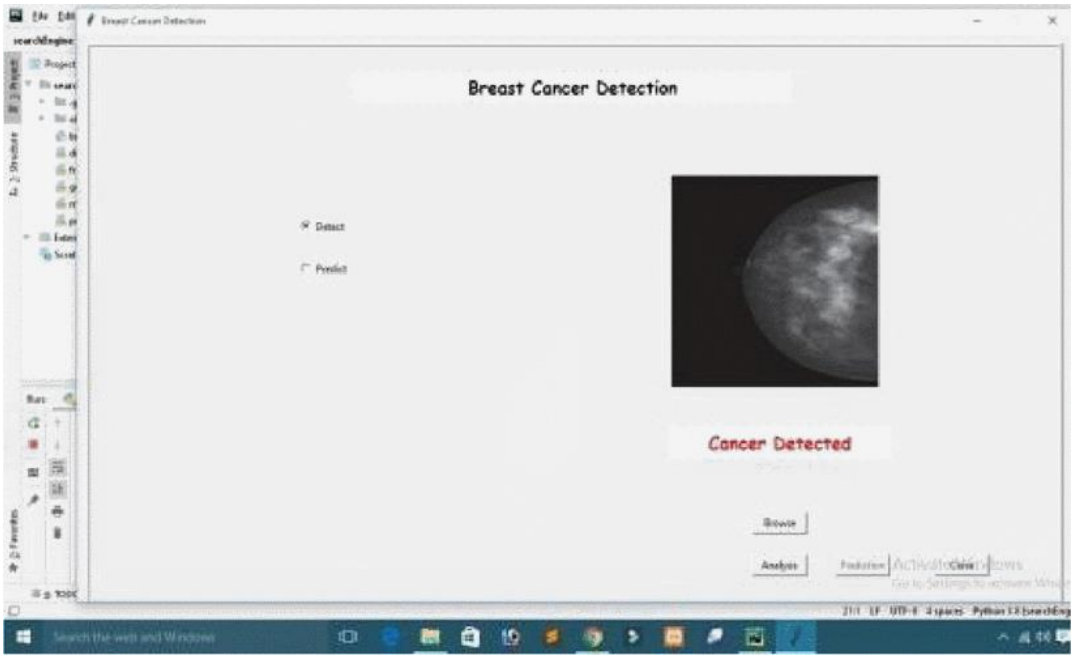
Στην αρχή χρησιμοποιούνται οι μαγνητικές τομογραφίες των ασθενών ως είσοδο στο πρόγραμμα και να γίνει μια προεργασία στις απεικονίσεις ώστε να έρθουν σε μορφή που θα μπορούν να εισαχθούν στο Συνελικτικό Νευρωνικό Δίκτυο (CNN). Αυτό το βήμα περιλαμβάνει πολλαπλή διοχέτευση εικόνων και έπειτα γίνεται η τμηματοποίηση των όγκων από τις απεικονίσεις όπου αυτή είναι αναγκαία. Το δεύτερο βήμα είναι η εξαγωγή των χαρακτηριστικών του όγκου από την μορφολογία του είτε με εποπτικό τρόπο (supervised) ή μη (unsupervised). Εκεί χρησιμοποιείται το στάδιο που περιλαμβάνει το CNN. Το τελευταίο βήμα είναι ο υποβιβασμός χρησιμοποιώντας το Support Vector Machine, που συνήθως ονομάζεται SVM, το οποίο τοποθετεί μια εικόνα στην αντίστοιχη κλάση και με ένα πλήρως συνδεδεμένο επίπεδο που χρησιμοποιεί μια συνάρτηση ενεργοποίησης όπως η Softmax[11]. Τα Συνελικτικά Νευρωνικά Δίκτυα χρησιμοποιούνται για την εύρεση προτύπων (patterns) στις εικόνες. Στα πρώτα στρώματα του νευρωνικού αυτού δικτύου γίνεται η αναγνώριση γραμμών και γωνιών στις εικόνες. Καθώς προχωράει το δίκτυο την προσπέλαση του σε νευρώνες πιο κρυφούς από τους αρχικούς γίνεται και πιο λεπτομερής εξαγωγή πληροφοριών για τα απεικονιζόμενα αντικείμενα της μαγνητικής και κατά συνέπεια του καρκινικού όγκου. Έπειτα γίνεται μια ομαδοποίηση των στοιχείων που εντοπίστηκαν. Εκεί λοιπόν χρησιμοποιείται ο αλγόριθμος του $K_{\text{-στου}}$ – Κοντινότερου Γείτονα (KNN). Ουσιαστικά αυτός ο αλγόριθμος ανήκει στην ομάδα των αλγορίθμων Μηχανικής Μάθησης για αναγνώριση προτύπων και χρησιμοποιεί σετ δεδομένων για να βρει τον κ-οστο πιο κοντινό γείτονα σε επόμενα δείγματα. Η θεωρία της γειτνιάζουσας μεθόδου (adjoint method[12]) χρησιμοποιείται για να περιγράψει το σετ από τα δεδομένα που χρησιμοποιούνται για εκπαίδευση, που στην συνέχεια αποτελούν και το περιορισμό για την αναζήτηση επιπλέον γειτόνων κατά την εκτέλεση του αλγορίθμου. Παρακάτω φαίνεται και μια εικόνα που αναπαριστά πως ελέγχει ο αλγόριθμος τους γείτονες:

[11] https://developers.google.com/machine-learning/crash-course/multi-class-neural-networks/softmax
[12] PDE-constrained optimization and the adjoint method
https://cs.stanford.edu/~ambrad/adjoint_tutorial.pdf



Εικόνα 7: KNN αλγόριθμος κατά την εκτέλεσή του [5]

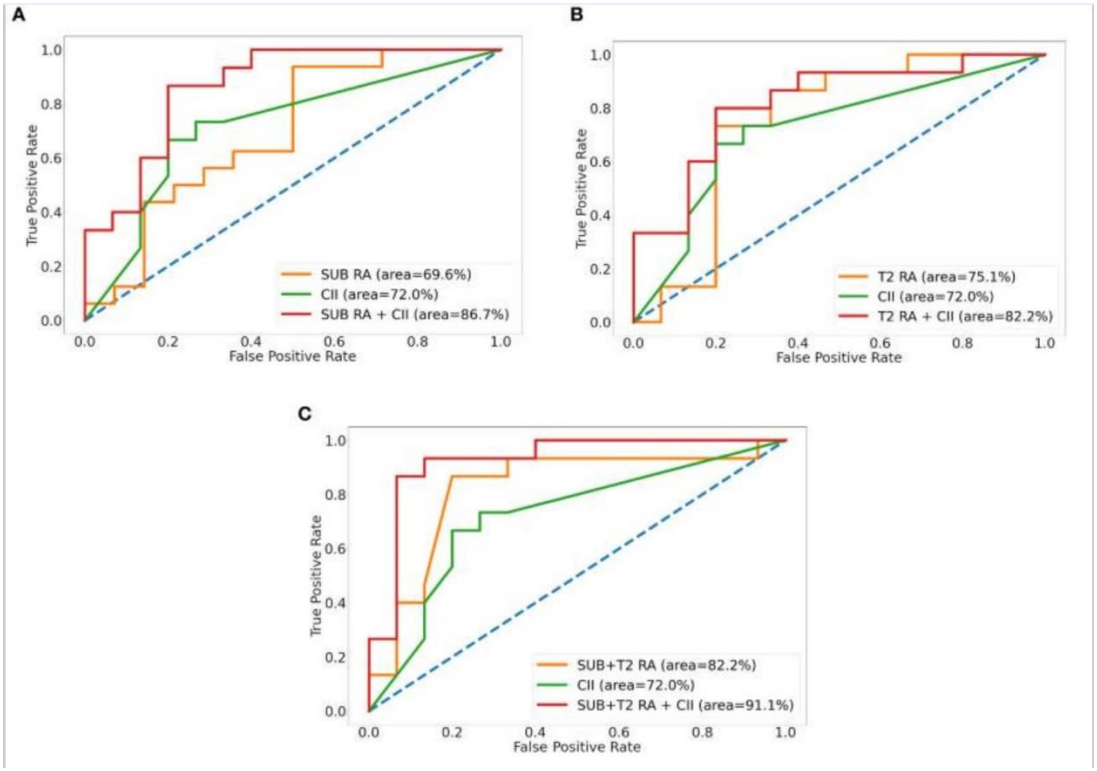
Τέλος χρησιμοποιείται ο SVM αλγόριθμος που είναι ένα εποπτικό σύστημα Μηχανικής Μάθησης το οποίο μπορεί να επιλύσει προβλήματα υποβάθμισης και παλινδρόμησης. Χρησιμοποιήθηκε λόγω την υψηλής ακρίβειας του. Όπως αναφέρεται στην δημοσίευση ο SVM έχει απόδοση 98%, ο CNN 95% ενώ ο KNN 73%. Στην συνέχεια φαίνεται το πρόγραμμα κατά την εκτέλεση του:



Εικόνα 8: Στιγμιότυπο κατά την εκτέλεση του προγράμματος [5]

Κατηγοριοποίηση επιπρόσθετων αλλοιώσεων που έχουν εντοπιστεί σε μαγνητικές τομογραφίες ασθενών με καρκίνο του μαστού κάνοντας χρήση ραδιομικής ανάλυσης και Μηχανικής Μάθησης.

Ο σκοπός δημιουργίας των αλγορίθμων της ερευνητικής εργασίας [6] είναι η μελέτη της σκοπιμότητας χρήσης ραδιομικής ανάλυσης (radiomics analysis) με αλγορίθμους μηχανικής μάθησης πάνω σε μαγνητικές τομογραφίες σε ασθενείς που έχουν ήδη πρωτοπαθή καρκίνου του μαστού για την εύρεση και διάκριση κακοήθων αλλοιώσεων από καλοήθεις. Σε αυτήν την έρευνα χρησιμοποιήθηκαν 174 αλλοιώσεις από απεικόνιση μαγνητικών τομογραφιών, εκ των οποίων οι 86 ήταν καλοήθεις και 88 κακοήθεις, από 158 ασθενείς που είχαν ομόπλευρο πρωτοπαθή καρκίνο του μαστού. Τα δεδομένα χωρίστηκαν σε τυχαία σε αναλογία 80:20, όπου το 80% χρησιμοποιήθηκε για την εκπαίδευση του αλγορίθμου και το 20% για την επιβεβαίωση της ακρίβειάς του. Τα ραδιομικά χαρακτηριστικά που εξήχθησαν από τις 3 περιοχές ενδιαφέροντος (ενδοκαρκινική, περιογκική, συνδυασμένη), χρησιμοποιώντας σταθμισμένες εικόνες T1 κορεσμένες σε λίπος που ελήφθησαν αφαιρώντας τις εικόνες πριν από την αντίθεση και την εικόνα με στάθμιση T2, δόθηκαν στον αλγόριθμο ως είσοδο στο SVM (Support Vector Machine) για την δυαδική κατηγοριοποίηση. Χρησιμοποιήθηκαν Δέντρα Αποφάσεων για να κατασκευαστεί ο κατηγοριοποιητής κάνοντας χρήση χαρακτηριστικών κλινικής απεικονιστικής ερμηνείας (Clinical Imaging Interpreation, CII) που αξιολογούνται από ακτινολόγους. Τα αποτελέσματα της έρευνας από τις ενδοκαρκινικές περιοχές ενδιαφέροντος έδειξαν ακρίβεια και (Area Under Receiver Operating Characteristics, AUROC) 73.3%, 69.6% για το μοντέλο ραδιομικής ανάλυσης, 70.0%, 75.1% για το T2 και 73.3%, 72.0% για το CII μοντέλο. Η διαγνωστική επίδοση αυξήθηκε όταν όταν τα ραδιομικά χαρακτηριστικά συνδυάστηκαν με τα CII χαρακτηριστικά από δεδομένα από πολυπαραμετρικές μαγνητικές τομογραφίες όπου η ακρίβεια έφτασε στο 86.7% και το AUROC στο 91.1%. Ενώ σε εξωτερικό τεστ που διεξήχθη η ακρίβεια και το AUROC των SUB+T2 και radiomics analysis+CII μοντέλου ήταν 80.6% και 91.4% αντίστοιχα. Παρακάτω παραθέτονται και τα διαγράμματα των αποτελεσμάτων:



Εικόνα 9: Λειτουργικές χαρακτηριστικές καμπύλες δέκτη ραδιομικής ανάλυσης (RA), κλινική απεικονιστική ερμηνεία (CII), συνδυαστικά μοντέλα for SUB **(A)**, T2 **(B)**, SUB+T2 **(C)**. SUB, **εικόνες T1 κορεσμένες σε λίπος που ελήφθησαν αφαιρώντας τις εικόνες πριν από την αντίθεση (SUB)**; T2, T2-weighted εικόνες. Λειτουργικές χαρακτηριστικές καμπύλες δέκτη ραδιομικής ανάλυσης (RA), κλινική απεικονιστική ερμηνεία (CII), συνδυαστικά μοντέλα for SUB **(A)**, T2 **(B)** SUB+T2 **(C)**. SUB, **εικόνες T1 κορεσμένες σε λίπος που ελήφθησαν αφαιρώντας τις εικόνες πριν από την αντίθεση**; T2, T2-weighted εικόνες [6].

Τριφασικό DeepBRCA-A Framework βασισμένο σε Βαθιά Μάθηση για αναγνώριση βιοδεικτών για κατηγοριοποίηση καρκίνου του μαστού.

Για αυτή την ερευνητική εργασία [7] χρησιμοποιήθηκε το σύνολο των δεδομένων από το Cancer Genome Atlas (TCGA)[13], το οποίο αποτελεί ένα πρόγραμμα συλλογής δεδομένων γονιδιωματικής του καρκίνου, χαρακτήρισε μοριακά πάνω από 20000 πρωτοπαθείς καρκίνους και τα ταίριαξε με φυσιολογικά δείγματα που καλύπτουν 33 τύπους καρκίνου. Περιλαμβάνει 1218 ασθενείς με καρκίνο του μαστού. Για κάθε ασθενή, υπάρχει γονιδιακή

[13] https://www.cancer.gov/about-nci/organization/ccg/research/structural-genomics/tcga

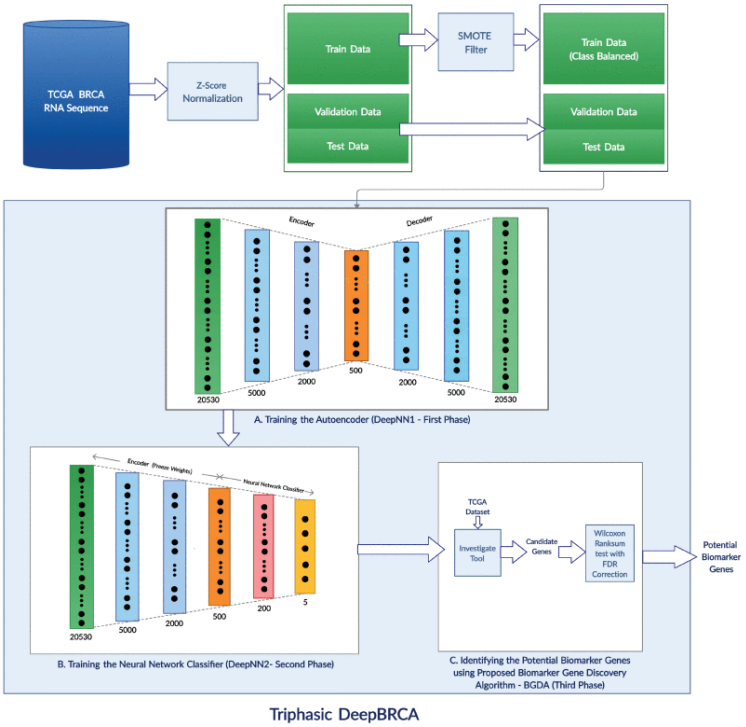
πληροφορία για 20530 γονίδια μαζί με συνοδευτική κλινική πληροφορία. Ουσιαστικά ο αλγόριθμος χωρίζεται σε 3 φάσεις:

- Σχεδιασμός ταξινομητή DeepNN1
- Δίκτυο κατηγοριοποίησης DeepNN2
- Εύρεση της υποκατηγορίας βιοδεικτών

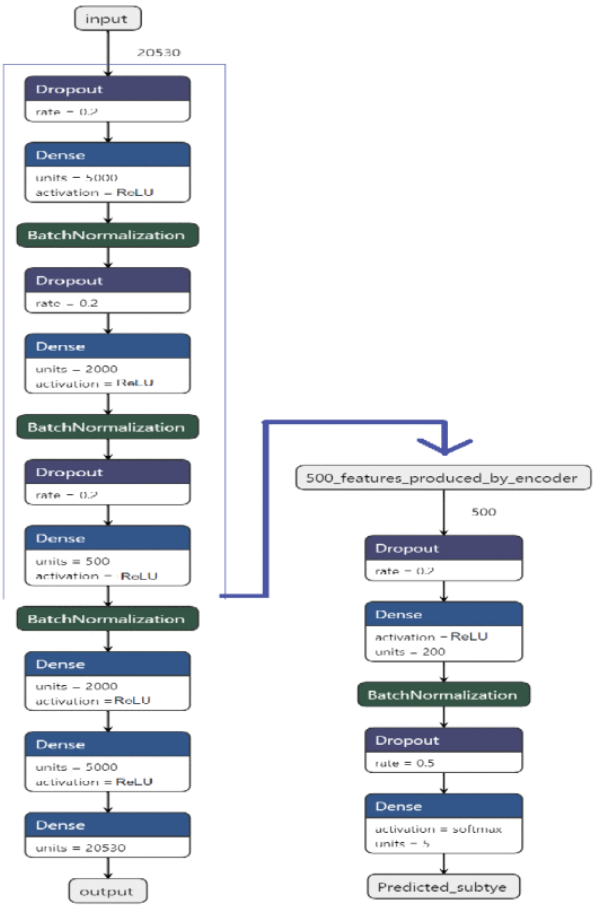
Στην πρώτη φάση χρησιμοποιείται ο ταξινομητής DeepNN1. Η αρχιτεκτονική του αυτόματου κωδικοποιητή περιλαμβάνει 6 πυκνά επίπεδα. Το πρώτο , το δεύτερο και το τρίτο επίπεδο μαζί αποτελούνται από 5000, 2000 και 500 κόμβους αντίστοιχα. Το τέταρτο, το πέμπτο και το έκτο αποτελούνται από 2000, 5000 και 20530 κόμβους αντίστοιχα. Ο αριθμος 20530 υποδηλώνει τον αριθμό των γονιδίων των οποίων η τιμή έκφρασης είναι διαθέσιμη για κάθε ασθενή. Επιπλέον ένας συντελεστής εγκατάλειψης 0.2 χρησιμοποιήθηκε στον αυτόματου κωδικοποιητή (autoencoder) για να προστατεύει το νευρωνικό δίκτυο από υπερπροσαρμογή. Η συνάρτηση ενεργοποίησης ReLU έχει χρησιμοποιηθεί σε όλα τα προαναφερθέντα επίπεδα. Το νευρωνικό δίκτυο στη δεύτερη φάση (DeepNN2) περιλαμβάνει δύο πυκνά στρώματα που έχουν 200 και 5 μονάδες αντίστοιχα. Το δίκτυο χρησιμοποιεί τη συνάρτηση ενεργοποίησης ReLU στο πρώτο πυκνό επίπεδο, ενώ στο τελικό στρώμα έχει υλοποιηθεί τη συνάρτηση ενεργοποίησης Softmax για να διευκολύνει την διαδικασία ταξινόμησης. Ενσωματώθηκε ένας συντελεστής εγκατάλειψης 0.20 μετά το επίπεδο εισόδου και 0.50 μετά το πρώτο κρυφό στρώμα για να προστατευθεί το δίκτυο από την υπερπροσαρμογή. Για την αντιμετώπιση του προβλήματος της εσωτερικής μετατόπισης συμμεταβλητών, εφαρμόζεται ένα στρώμα κανονικοποίησης μετά το πυκνό στρώμα. Έχοντας εκπαιδευτεί το δίκτυο αυτόματου κωδικοποιητή στην πρώτη φάση, δεδομένων των δεδομένων γονιδιακής έκφρασης (NumGenes=20530) για ένα ασθενή, η αντίστοιχη αναπαράσταση που περιλαμβάνει ένα διάνυσμα μεγέθους 500 εξάγεται από το δίκτυο κωδικοποιητών του DeepNN1. Αυτό το κωδικοποιημένο διάνυσμα μεγέθους 500 στοιχείων χρησιμοποιείται στο δίκτυο ταξινόμησης του DeepNN2 για την πρόβλεψη του τύπου του καρκίνου του μαστού. Στην τρίτη φάση και τελευταία φάση, στόχος είναι να εντοπισθούν τα γονίδια που έχουν σημαντική συμβολή στην επίτευξη των αποτελεσμάτων ταξινόμησης. Για το σκοπό αυτό, προτάθηκε ένας αλγόριθμος ανακάλυψης γονιδίων βιοδεικτών (BGDA) που αξιοποιεί τον ταξινομητή νευρωνικών δικτύων της δεύτερης φάσης χρησιμοποιώντας μεθόδους διάδοσης σχετικότητας που είναι διαθέσιμες στο εργαλείο iNNvestigate[14]. Έχουν αξιοποιηθεί έξι μέθόδοι του εργαλείου iNNvestigate για τον εντοπισμό των γονιδίων που σχετίζονται με την ταξινόμηση του υποτύπου του καρκίνου του μαστού. Αυτές οι μέθοδοι χρησιμοποιούνται για τον επεξήγηση της λειτουργίας της ‹‹συμπεριφοράς›› και των αποτελεσμάτων του νευρωνικού δικτύου. Για έναν δεδομένο υποτύπο (έστω HER2) και μια δεδομένη μέθοδο ανάλυσης (ας πούμε, καθοδηγούμενη οπίσθια διάδοση), επιλέχθηκαν τα 250 πιο σημαντικά γονίδια που συνέβαλαν στον ταξινομημένο υποτύπο του. Για κάθε υποκατηγορία, διατηρήθηκαν μόνο εκείνα τα γονίδια που υπήρχαν σε τουλάχιστον 30% των ασθενών.

[14] https://github.com/albermax/innvestigate

Παρακάτω φαίνονται κάποια σχήματα που υποδεικνύουν τη λειτουργία των τριών φάσεων του αλγορίθμου καθώς και την αποτελεσματικότητά του σε σχέση με άλλους αλγορίθμους:



Εικόνα 10: Επισκόπηση της ροής λειτουργίας βαθιάς μάθησης του τριφασικού DeepBRCA για ανακάλυψη γονιδίων βιοδεικτών. Η συστηματική αναπαράσταση του πλαισίου περιλαμβάνει τρία κύρια στοιχεία: Αυτόματος κωδικοποιητής για μειωμένη αναπαράσταση γονιδιακής έκφρασης, ταξινομητή νευρωνικού δικτύου τροφοδοσίας για διαστρωμάτωση υποτύπου καρκίνου του μαστού και ανάλυση ταξινομητή νευρικού δικτύου της δεύτερης φάσης για πιθανή ανακάλυψη υπογραφής γονιδίου βιοδείκτη [7].



Εικόνα 11: Διάγραμμα ροής για τις 2 πρώτες φάσεις που αποτελούνται από τα νευρωνικά δίκτυα DeepNN1 και DeepNN2 [7].

Research Group	Results Type of Omic Data	Genes	Accuracy
Proposed Model- *Triphasic DeepBRCA*	RNA Sequence Gene Expression	54	**0.899 ± 0.04**
Zhang et al. [20]	RNA Sequence Gene Expression	47	0.863
List et al. [19]	RNA Sequence Gene Expression Methylation Data RNA Sequence and Methylation Data	53 38 275	0.869 0.753 0.878
Gao et al. [21]	RNA Sequence Gene Expression	1000	≈ 0.80

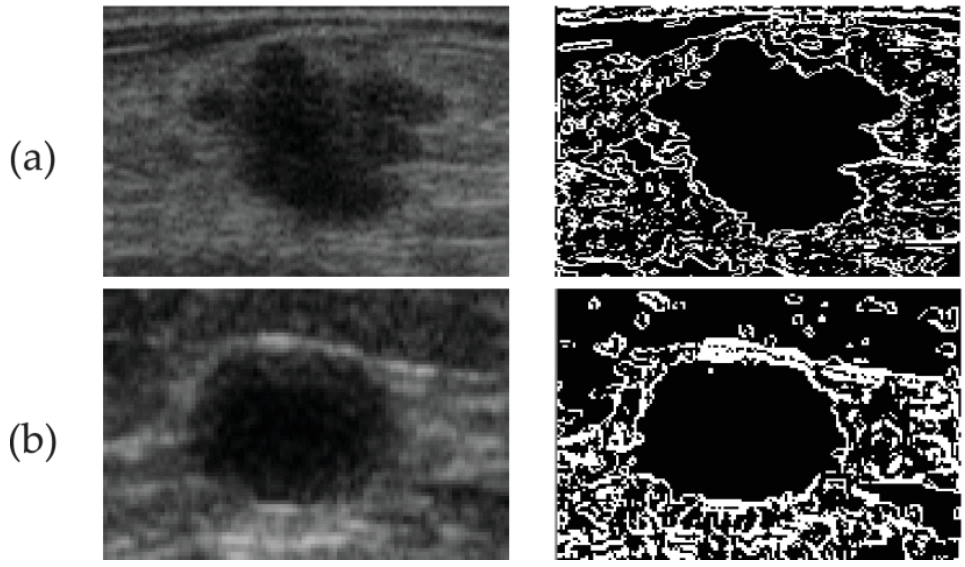
Εικόνα 12: Σύγκριση αποτελεσμάτων του DeepBRCA με τους αλγορίθμους των Zhang et al, List et al και Gao et al [7].

Ταξινόμηση μάζας μαστού με χρήση αλγόριθμου eLFA με βάση το μοντέλο βαθιάς μάθησης CRNN.

Οι ερευνητές αυτού του αλγορίθμου [8] ανέπτυξαν το μοντέλο eLFA-CRNN που αναλύει τα γραμμικά τμήματα της μάζας στην υπερηχητική εικόνα μαστού για να εκτιμήσει εάν η μάζα του μαστού είναι καλοήθης ή κακοήθης. Έτσι, ονομάζεται ανάλυση χαρακτηριστικών γραμμής (LFA). Το αρχικό LFA λαμβάνει υπόψη το θετικό εύρος όλων των τμημάτων γραμμής. Το eLFA διευρύνει την έκφρασή του, λαμβάνοντας υπόψη και ένα αρνητικό εύρος. Το γράμμα "e" στο eLFA υποδηλώνει την επέκταση της αρχικής έκδοσης του LFA. Ο προτεινόμενος αλγόριθμος eLFA εκτελεί την διαδικασία ανίχνευσης άκρων στην εικόνα εισόδου για να προσδιορίσει τους τύπους των γραμμών που αποτελούν το αντικείμενο. Τα δεδομένα που υπολογίζονται χρησιμοποιώντας τον αλγόριθμο ανίχνευσης ακμών είναι δυαδικά και οι τύποι γραμμών που προσδιορίζονται χωρίζονται σε 16 τύπους μέσω συνέλιξης με προκαθορισμένα φίλτρα. Τα ταξινομημένα δεδομένα τύπου γραμμής δημιουργούνται ως δεδομένα εκμάθησης σε έναν χάρτη χαρακτηριστικών 16×32 που ονομάζεται χάρτης eLFA. Το μοντέλο eLFA-CRNN έχει σχεδιαστεί και εφαρμόζεται για την ταξινόμηση του καρκίνου του μαστού για την εκμάθηση του χάρτη eLFA. Οι ακόλουθες ενότητες δείχνουν τη λεπτομερή επεξεργασία του αλγορίθμου eLFA.

- Η διαδικασία ανίχνευσης άκρων του eLFA-CRNN για την εξαγωγή πληροφοριών τμήματος γραμμής

Αυτή η ενότητα εισάγει τον αλγόριθμο ανίχνευσης άκρων που μετατρέπει μια εικόνα μαστού υπερήχων σε πληροφορίες γραμμικού τμήματος. Η εικόνα 13 παρουσιάζει την μετατροπή:



Εικόνα 13: Μάζα υπερηχητικής εικόνας μαστού και η άκρη του. (α) Κακοήθης εικόνα. (β) Καλοήθης εικόνα [8] .

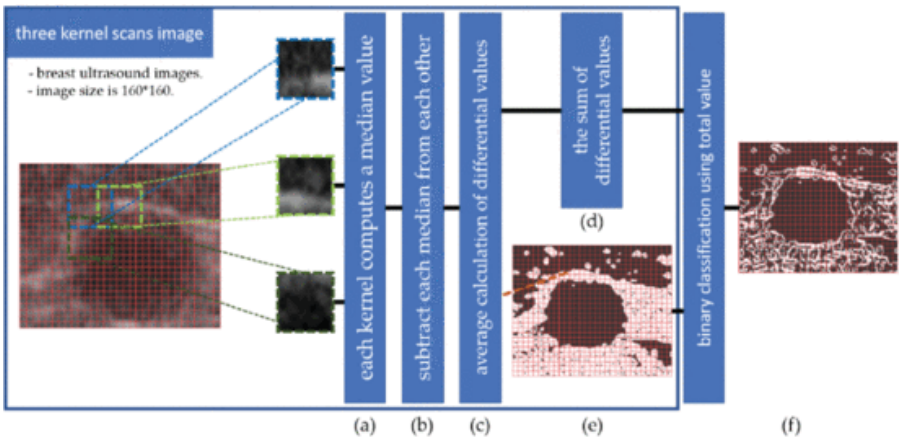
Ο αλγόριθμος ανιχνεύει μια άκρη σε έξι βήματα. Όλες οι περιοχές της υπερηχητικής εικόνας του μαστού σαρώνονται μέσω τριών φίλτρων, τα οποία ορίζουν τις περιοχές όπως παρουσιάζονται από κάτω:

$P_o = [x(i+r,j+s); (r,s) \in A], (i, j) \in Z^2$

$P_h = [x(i+1+r,j+s); (r,s) \in A], (i+1, j) \in Z^2$

$P_v = [x(i+r,j+1+s); (r,s) \in A], (i, j+1) \in Z^2$

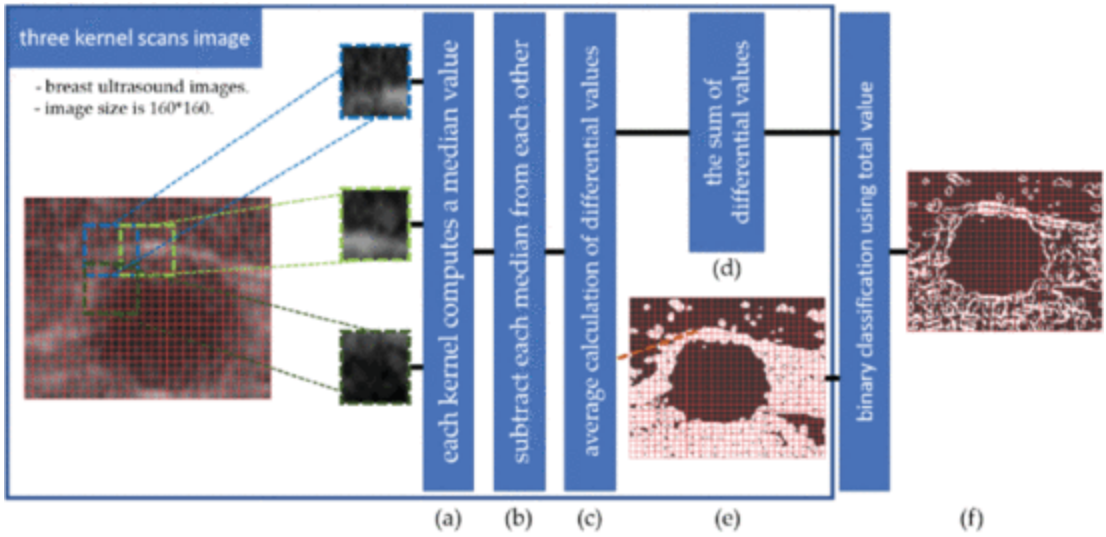
Παρακάτω φαίνεται απεικονιστικά πως γίνεται η διαδικασία επιλογής των άκρων και άρα της οριοθέτησης:



Εικόνα 14: Διαδικασία ανίχνευσης άκρων μέσω υπερηχητικής εικόνας μαστού [8]

- Δημιουργία χάρτη eLFA μέσω της ταξινόμησης τμήματος γραμμής βάσει πληροφοριών άκρων

Εδώ έχει γίνει ανίχνευση ενός άκρου της υπερηχητικής εικόνας του μαστού. Μια σειρά μοτίβων εφαρμόστηκε στις πληροφορίες άκρων που ανιχνεύθηκαν για να ταξινομηθούν τα τμήματα γραμμής σε 16 τύπους. Η εικόνα 14 παρουσιάζει τη διαδικασία εφαρμογής προτύπων:



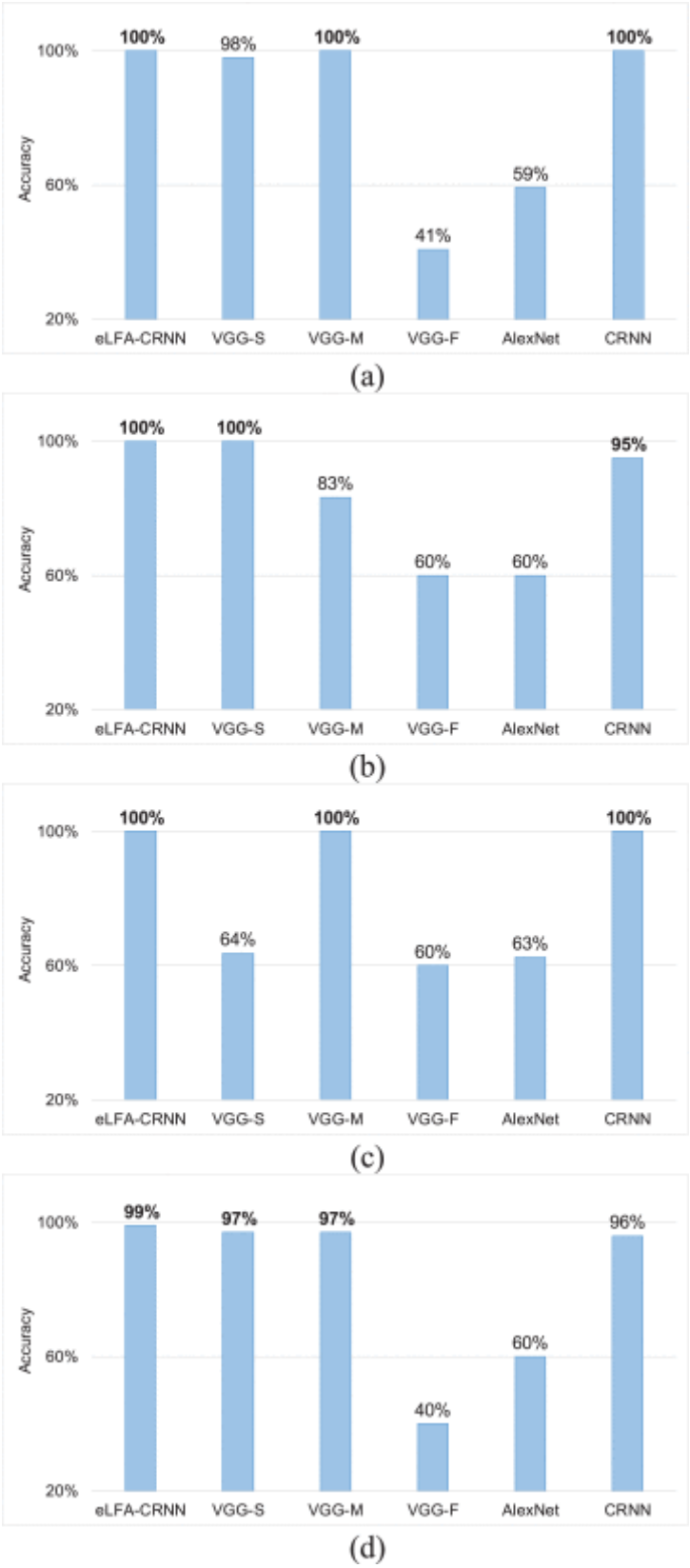
Εικόνα 15: Διαδικασία δημιουργίας χαρτών eLFA. (α) Βήμα δημιουργίας χάρτη. (β) Το βήμα για την εκμάθηση του χάρτη που δημιουργήθηκε[8].

Όπως φαίνεται στην Εικόνα 15(α), απαιτούνται τρία βήματα για τη δημιουργία ενός χάρτη eLFA. Το βήμα 1 είναι να διαιρεθεί η εικόνα των άκρων σε 16 κομμάτια. Το δεύτερο βήμα είναι να εφαρμοστεί η συνέλιξη σε κάθε διαιρεμένο κομμάτι χρησιμοποιώντας ένα φίλτρο. Το τελευταίο βήμα είναι να αναλυθούν οι παράμετροι του χάρτη χαρακτηριστικών που εξήχθησαν από το φίλτρο, να γίνει συλλογή αυτών και να δημιουργήσουν εν τέλι τον τελικό χάρτη χαρακτηριστικών. Ο χάρτης χαρακτηριστικών που σχεδιάζεται στη διαδικασία ονομάζεται χάρτης eLFA. Χρησιμοποιείται ως είσοδος στο βήμα εκμάθησης που φαίνεται στην Εικόνα 15(β). Το παραδοσιακό μοντέλο CRNN τροποποιήθηκε για να επιτρέψει την εκμάθηση του χάρτη eLFA.

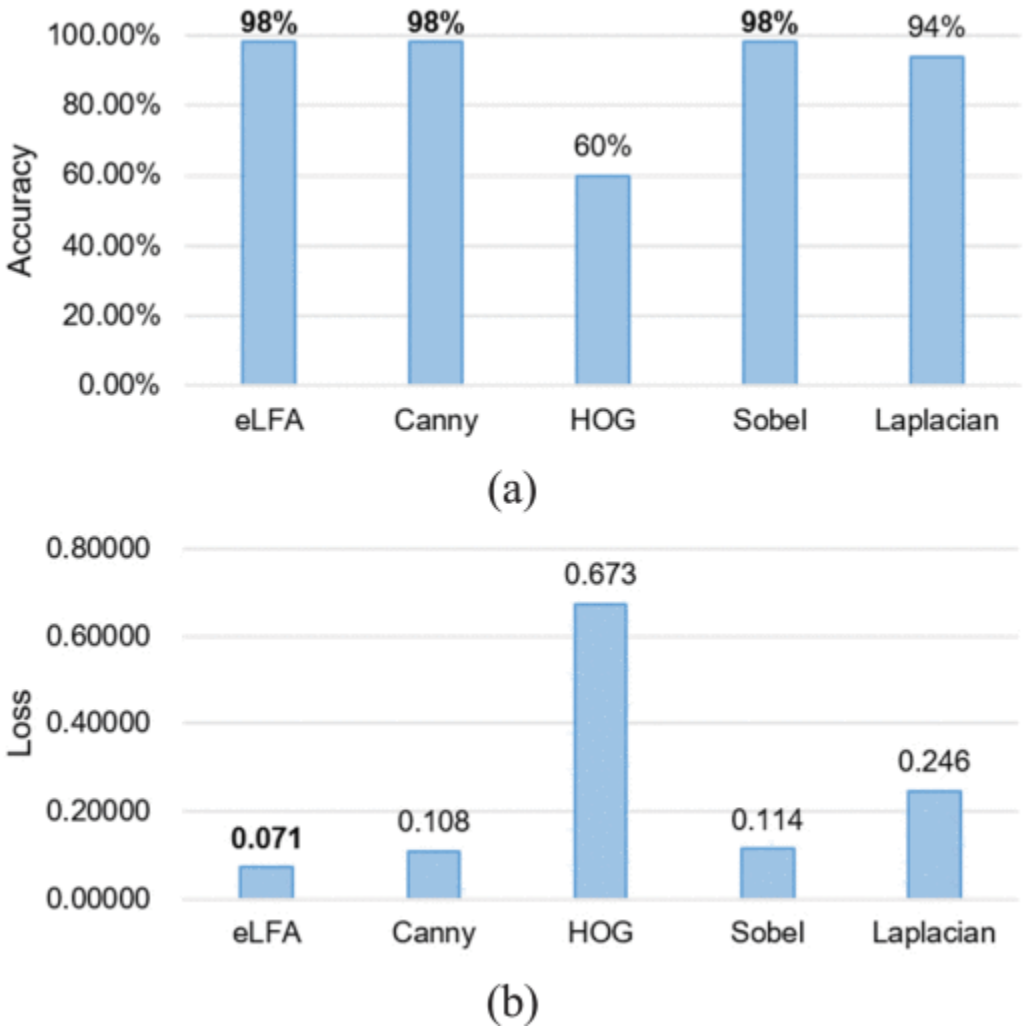
- Σχεδιασμός μοντέλου CRNN για εκμάθηση χάρτη eLFA

Σε αυτή την έρευνα δεν χρησιμοποιείται το κλασικό μοντέλο CRNN από αποτελείται από 7 στρώματα Αντίθετα το μοντέλο που αναπτύχθηκε, eLFA-CRNN, επεξεργάζεται εκ των προτέρων τις πληροφορίες ακμών χρησιμοποιώντας τον αλγόριθμο eLFA, έτσι ώστε να μην γίνεται η χρήση πολλαπλών επιπέδων συνέλιξης. Ένας πολύ συγκεντρωμένος χάρτης eLFA μπορεί να επηρεάσει αρνητικά την ταξινόμηση καθώς μπορεί να είναι πολύ μικρός. Έτσι, αυτή η μελέτη προσπάθησε να εκφράσει έναν χάρτη eLFA με διάφορους τρόπους μέσω ενός στρώματος συνέλιξης που εξάγει πληροφορίες ακμών και κατεύθυνσης ελέγχοντας τις παραμέτρους του φίλτρου. Εξισώνοντας το μέγεθος ενός φίλτρου με αυτό του χάρτη eLFA, μπορεί να ληφθεί μια έξοδος παρόμοια με αυτή του χάρτη eLFA. Σε αυτή τη μελέτη, χρησιμοποιήθηκε ένα στρώμα συνέλιξης για τη μετατροπή της μορφής έκφρασης, αντί για εξαγωγή πληροφοριών ακμών και κατεύθυνσης. Έτσι, το στρώμα 2 είχε το ίδιο μέγεθος φίλτρου με τα δεδομένα εισόδου, και δημιουργήθηκαν 64 διαφορετικές μορφές έκφρασης. Ο χάρτης χαρακτηριστικών που σχεδιάστηκε στο στρώμα 2 αποτρέπει την υπερπροσαρμογή μέσω των επιπέδων 3 και 7. Ο χάρτης χαρακτηριστικών που σχεδιάστηκε στο στρώμα 3 ευθυγραμμίζεται εκ νέου έτσι ώστε το μέγεθός του να αλλάξει σε 512 και στη συνέχεια να μεταβεί σε ένα επαναλαμβανόμενο επίπεδο. Τα επίπεδα 5 και 6 είναι επαναλαμβανόμενα επίπεδα που χρησιμοποιούν έναν τύπο κυψέλης Gated Recurrent Units (GRU). Η επαναλαμβανόμενη δομή λαμβάνει διαδοχικά μια ποικιλία χαρτών χαρακτηριστικών και αναλύονται οι κοινές περιοχές τους. Το στρώμα 5 έχει 64 μονάδες και το στρώμα 6 έχει 32. Το τελικό αποτέλεσμα σχεδιάζεται στο στρώμα 7. Το μοντέλο εκμάθησης χρησιμοποιείται για να προσδιοριστεί εάν μια μάζα που παρατηρείται σε μια υπερηχητική εικόνα μαστού είναι «καλοήθης» ή «κακοήθης».

Παρακάτω φαίνεται η απόδοση του αλγορίθμου σε πειραματικά περιβάλλον και συγκριτικά με άλλους αλγορίθμους:



Εικόνα 16: Αποτελέσματα σε πειραματικά δεδομένα [8]



Εικόνα 17: Αποτελέσματα μέτρησης ακρίβειας για αλγόριθμους ανίχνευσης ακμών. (α) Ακρίβεια. (β) Απώλεια [8]

Αποδοτικός αλγόριθμος AdaBoost υποβοηθούμενος από Βαθιά Μάθηση για ανίχνευση καρκίνου του μαστού και πρώιμη διάγνωση .

Ο σκοπός αυτής της μελέτης [9] είναι να βελτιώσει την πρόβλεψη της πρόγνωσης του όγκου και να βελτιώσει μια βαθύτερη ταξινόμηση των αποτελεσμάτων. Η μέθοδος αυτής της έρευνας δείχνει μια εποπτευόμενη εκμάθηση ταξινομητή και μια διαδικασία εκμάθησης χαρακτηριστικών χωρίς επίβλεψη σε σύγκριση με προηγούμενες προσεγγίσεις ταξινομητή. Το πλήρως συνδεδεμένο συνελικτικό στρώμα έχει χρησιμοποιηθεί για επιλογή χαρακτηριστικών, εξαγωγή χαρακτηριστικών, ανίχνευση, τμηματοποίηση και ταξινόμηση για την αξιολόγηση διαφορετικών σταδίων καρκίνου του μαστού.

Σε αυτήν την περίπτωση, το CNN μαθαίνει την πλήρη ανάλυση κάθε εικονοστοιχείου από τα πραγματικά δεδομένα εισόδου για να επιτύχει πιο ακριβή κατάτμηση από pixel σε pixel, ειδικά στις άκρες των αντικειμένων. Για το σκοπό αυτό, τα στρώματα υποδειγματοληψίας και συγκέντρωσης του δικτύου μπορούν να εξαλειφθούν και τα συνελικτικά στρώματα μπορούν να εξαγάγουν και να μάθουν τα πλήρη χωρικά χαρακτηριστικά του σήματος εισόδου.
Σε αυτό το άρθρο, έχει προταθεί ο αποτελεσματικός αλγόριθμος Adaboost με τη βοήθεια Deep Learning (DLA-EABA) για την ανίχνευση του καρκίνου του μαστού. Για να επιτευχθεί υψηλή ακρίβεια, το CNN απαιτεί εκτεταμένα δεδομένα για εκπαίδευση Επειδή δεν υπάρχουν πολλά διαθέσιμα δεδομένα, τα τελικά δεδομένα που χρησιμοποιήθηκαν για εκπαίδευση και έρευνα είναι από τον ιστότοπο https://wiki.cancerimagingarchive.net/.

- **Βήμα 1**: Αυτόματος Κωδικοποιητής και Ανάλυση Αποκωδικοποιητή για Ταξινόμηση

Η προτεινόμενη μέθοδος εκμάθησης χαρακτηριστικών περιλαμβάνει την χρήση πολλών στοιβαγμένων αυτόματων κωδικοποιητών με σκοπό τη δημιουργία ενός βαθύ συνελικτικού νευρωνικού, με τρόπο ιεραρχικό. Ο μη γραμμικός μετασχηματισμός (NLT) μπορεί να ληφθεί από τη συνδυασμένη απεικόνιση των πραγματικών δεδομένων $\tilde{Y}$ ως είσοδο, Τα τμήματα κωδικοποιητή και αποκωδικοποιητή του αυτόματου κωδικοποιητή περιέχουν πολλαπλά NLT ως εξής:

$$g^{(1)}= \rho(\omega^{(1)} \tilde{Y} + a^{(1)})$$

$$g^{(i)}=\rho(\omega^{(i)}g^{(i-1)}+a^{(i)}),\ i=1,2,\ldots m$$

Όπως φαίνεται στην παραπάνω εξίσωση όπου το m δείχνει τον αριθμό των στρώσεων και το ρ τη συνάρτηση ενεργοποίησης. Τα $g^{(i)}$, $\omega^{(i)}$ και $a^{(i)}$ αντιπροσωπεύουν τον πίνακα βάρους, το κρυφό διάνυσμα και το διάνυσμα πόλωσης στο io στρώμα αντίστοιχα.

- **Βήμα 2**: Αλγόριθμος Adaboost για την εκμάθηση ταξινόμησης

Ο αλγόριθμος AdaBoost χρησιμοποιείται στην εκμάθηση ταξινομητή της προβλεπόμενης προσέγγισης για την εκπαίδευση ταξινομητών που βασίζονται σε έναν αλγόριθμο εποπτευόμενης μάθησης για τον υπολογισμό μιας δυαδικής ταξινόμησης που διαιρεί καλύτερα τις θετικές και τις αρνητικές περιπτώσεις.

- **Βήμα 3:** Επίπεδο Βαθιού συνελικτικού νευρωνικού δικτύου

Το επίπεδο συνέλιξης έχει χρησιμοποιηθεί για τη μετατροπή των πυρήνων σε κομμάτια εικόνας. Το φίλτρο απόρριψης του πυρήνα είναι η παρακάτω συνάρτηση:

$F_m=(F_1^m, F_2^m, F_3^m,\dots F_{e_g^m}^m)$

- **Βήμα 4:** Παλινδρόμηση Softmax

Αυτή είναι μια διαδικασία ταξινόμησης που γενικεύει πολυωνυμικά προβλήματα με λογιστική παλινδρόμηση. Η παλινδρόμηση Softmax, μια γραμμική παλινδρόμηση που παράγει ακατέργαστες βαθμολογίες κλάσεων, δημιουργεί μια κατανομή πιθανότητας κλάσης.

Παρατηρήθηκε από τα πειραματικά αποτελέσματα ακρίβεια 97,2%, ευαισθησία 98,3% και ειδικότητα 96,5% σε σύγκριση με άλλα υπάρχοντα συστήματα.

Σκοπός αυτής της διπλωματικής εργασίας είναι η ανάπτυξη μιας εφαρμογής σε γλώσσα MATLAB, η οποία θα λαμβάνει 5 συγκεκριμένα χαρακτηριστικά (Morphology, Borders, Tumor Size, Curve Morphology και ADC (Apparent Diffusion Coefficient)), που γίνονται εξαγωγή από μαγνητικές τομογραφίες ασθενών με κάποιο ύποπτο όγκο στο στήθος, και κατά την εκτέλεση του θα προβλέπει ως αποτέλεσμα κάποιους ιστολογικούς δείκτες (Tumor Grade, ER (estrogen receptor), PR (hormone progesterone), CERB-2 (Receptor tyrosine-protein kinase erbB-2) και Ki-67 (πρωτεΐνη Ki-67)) οι οποίοι υπό κανονικές συνθήκες οι ιατροί αποφαίνονται για τις τιμές τους αν κάνουν βιοψία και τους παρατηρήσουν κάτω από το μικροσκόπιο. Αυτό μπορεί να γίνει με την χρήση κάποιων μαθηματικών μοντέλων που θα αναλυθούν παρακάτω και αφού ο αλγόριθμος έχει εκπαιδευτεί σε δεδομένα που γνωρίζουμε ήδη τα αποτελέσματά τους. Το ποσοστό επιτυχίας των προς πρόβλεψη χαρακτηριστικών γίνεται με βάση τα αποτελέσματα που έχει βγάλει σαν έξοδο ο αλγόριθμος σε σύγκριση με τα υπολειπόμενα δεδομένα, αυτά δηλαδή που γνωρίζουμε τα αποτελέσματά τους αλλά δεν χρησιμοποιήθηκαν για την εκπαίδευση του αλγορίθμου.

Κεφάλαιο 3: Μεθοδολογία

Σε αυτό το κεφάλαιο θα γίνει επεξήγηση των εργαλείων, αλγορίθμων, γλωσσών προγραμματισμού που χρησιμοποιήθηκαν για την εξαγωγή αποτελεσμάτων.

3.1 Γλώσσα προγραμματισμού και περιβάλλον MATLAB

Το MATLAB είναι μια γλώσσα υψηλού επιπέδου για μηχανικούς και επιστήμονες. Ενσωματώνει υπολογισμούς, οπτικοποίηση και προγραμματισμό σε ένα εύχρηστο περιβάλλον όπου τα προβλήματα και οι λύσεις εκφράζονται με γνωστές μαθηματικές εκφράσεις. Οι τυπικές χρήσεις περιλαμβάνουν:

- Μαθηματικούς υπολογισμούς, όπως ολοκληρώματα, πράξεις με πίνακες, κ.α.
- Μοντελοποίηση, προσομοίωση και προτυποποίηση
- Ανάλυση δεδομένων, εξερεύνηση και οπτικοποίηση
- Επιστημονικά και μηχανικά γραφήματα
- Ανάπτυξη εφαρμογών, συμπεριλαμβανομένης της δημιουργίας γραφικής διεπαφής χρήστη (GUI)

Το MATLAB είναι ένα διαδραστικό σύστημα του οποίου το βασικό στοιχείο δεδομένων είναι ένας πίνακας που δεν απαιτεί διαστασιολόγηση. Αυτό επιτρέπει να λύνονται πολλά τεχνικά υπολογιστικά προβλήματα, ειδικά αυτά με τυποποιήσεις μήτρας και διανύσματος, πολύ πιο γρήγορα από ότι θα χρειαζόταν για να γραφτεί ένα πρόγραμμα σε μια γλώσσα όπως η Fortran[15].

Το όνομα MATLAB προέρχεται από τις λέξεις MATrix LABoratory και αναπτύχθηκε από τον Cleve Moler, ο οποίος ήθελε να αναπτύξει ένα πιο εύκολο σύστημα υπολογισμού πράξεων πινάκων για τους φοιτητές του από το LINPACK[16] και το EISPACK[17] της Fortran. Έτσι τον Δεκέμβριο του 1984, παρουσιάστηκε στο IEEE Conference on Decision and Control στο Las Vegas το PC-MATLAB, ενώ νωρίτερα υπήρχαν κάποιες εκδόσεις οι οποίες ήταν απλά ένας διαδραστικός τρόπος υπολογισμού πινάκων[18].

Πλεονεκτήματα της Matlab[19]:

- Ευκολία στη χρήση
- Υποστήριξη στις 3 βασικά λειτουργικά συστήματα (Windows, Linux, MacOS)
- Ανεξαρτησία όσον αφορά την πλατφόρμα της απεικόνισης γραφικών παραστάσεων
- Γραφικό περιβάλλον εργασίας

[15] https://cimss.ssec.wisc.edu/wxwise/class/aos340/spr00/whatisMATLAB.htm
[16] https://www.netlib.org/linpack/
[17] https://www.netlib.org/eispack/
[18] https://www.mathworks.com/company/newsletters/articles/a-brief-history-of-MATLAB.html
[19] https://www.javatpoint.com/advantages-and-disadvantages-of-matlab

- Ύπαρξη compiler, εκτός του διερμηνέα (interpreter)

Μειονεκτήματα της Matlab:

- Κατά κύριο λόγο η χρήση του MATLAB, και μέσα από το περιβάλλον MATLAB, γίνεται με την χρήση του διερμηνέα (interpreter), κάτι που μερικές φορές καθιστά αργή την εκτέλεση.
- Κόστος αγοράς.

3.2 Οργάνωση δεδομένων

Τα δεδομένα που ελήφθησαν αφορούσαν 77 ασθενείς που για λόγους ιατρικού απορρήτου έχουν αφαιρεθεί τα προσωπικά τους στοιχεία και έχουν αντικατασταθεί από ένα αύξοντα αριθμό. Τα δεδομένα είναι σε μορφή Excel και περιλαμβάνουν 30 στήλες με δεδομένα οι οποίες είναι οι εξής: Αύξων αριθμός του ασθενούς, ιστορικό, οικογενειακό ιστορικό, ηλικία, BIRADS, αν ο όγκος είναι καλοήθης, Tumor Type, Morphology, Borders, Tumor Size, perifocal edema, T2 WI, curve morphology, curve type, breast density, BPE, Feeding Vessel, Inter. Enhancement, Diffusion, ADC, Focality, Tumor Grade, ER, PR, CERB-2, Ki-67, P63, E-Cadherin, Axillary Lymph Nodes, Other Findings. Οι τελευταιες 4 στήλες δεν περιείχαν καθόλου δεδομένα και ούτε αποτελούν μεταβλητές για εξαγωγή αποτελεσμάτων. Να σημειωθεί ότι από τους 77 ασθενείς, αποκλείστηκαν 20 οι οποίοι στο excel στην στήλη BENIGN, είχαν συμπληρωμένα δεδομένα, που υποδηλώνει την καλοήθεια του όγκου ή το ότι ο ασθενής βρίσκεται σε στάδιο μετά την θεραπεία αντιμετώπισης του κακοήθη όγκου. Επίσης, όπως αναφέρθηκε και στο κεφάλαιο 2 τα δεδομένα εισόδου των αλγορίθμων είναι οι 5 από αυτές τις στήλες, οι οποίες είναι: Morphology, Borders, Tumor Size, Curve Morphology και το ADC, ενώ οι πέντε προς εκτίμηση μεταβλητές είναι: Tumor Grade, ER, PR, CERB-2 και Ki-67. Για το σκοπό αυτό και προς διευκόλυνση της εισαγωγής των δεδομένων στους αλγορίθμους κατασκευάστηκαν 5 διαφορετικά excel αρχεία που κάθε φορά περιείχαν τις 5 μεταβλητές εισόδου και μια από τις προς πρόβλεψη μεταβλητές. Αφού, δημιουργήθηκαν τα 5 excel αρχεία έγινε έλεγχος στις στήλες που περιείχαν τα δεδομένα εισόδου και εξόδου αν είναι όλα συμπληρωμένα με τιμές. Παρατηρήθηκε λοιπόν, πως από σε κάποια δεν υπήρχαν τιμές και έτσι στο αρχείο για την πρόβλεψη της μεταβλητής Tumor Grade έχουμε 45 ασθενείς διαθέσιμους για εκπαίδευση και πρόβλεψη των δεδομένων από τους οποίους μπορούμε να ελέγξουμε και την ορθότητά του αλγορίθμου στη πρόβλεψη και 5 στους οποίους δεν έχουμε καθόλου την τιμή στην προς πρόβλεψη μεταβλητή και απλά θα χρησιμοποιηθούν για πρόβλεψη χωρίς να ξέρουμε την ορθότητά του αποτελέσματος. Στις τιμές που δεν είναι γνωστή η έξοδος χειροκίνητα τοποθετήθηκε η τιμή μηδέν. Παρακάτω υπάρχει ένας πίνακας που εξηγεί αναλυτικά τα ωφέλιμα δεδομένα:

Πίνακας 1: Δεδομένα προς χρήση.

Μεταβλητή για πρόβλεψη	**Ασθενείς με ολοκληρωμένα Δεδομένα**	**Δεδομένα που δεν γνωρίζουμε το αποτέλεσμά τους**	**Άθροισμα όλων των ασθενών που θα αξιοποιηθούν από τους αλγορίθμους**
Tumor Grade	45	5	50
ER	49	2	51
PR	49	2	51
CERB-2	46	4	50
Ki-67	44	6	50

17: https://introcs.cs.princeton.edu/java/71boolean/

ANONYMOUS	HISTORY	FAMILY HISTORY	AGE	BIRADS	BENIGN	MALIGNANT = Tumor Type	MORPHOLOGY	BORDERS	TUMOR SIZE
1	PALP.RT.1ST MAMMO.U/S SUSP	NEG.	51	5		IDC	MASS	IRR	1.5
2			63	5		IDC	MASS	SPIC.	2.2
3			43	5		IDC	NME	IRR	6.5
4	PAPL.LT	POS.	33	5		IDC	MASS	IRR	1.6
5	PALP.LT	POS.	41	4		ILC	NME	IRR	2.8
6	PALP.LT		55	5		IDC	MASS	SPIC.	3.8
7	?	?	62	5		ILC	MASS	IRR	0.7
8	PALP/MAMMO RT		43	5		IDC	NME	IRR	6.8
9			39	5		IDC	MASS	IRR	1.9
10	SUSP.MICROCAL.MAMMO .RT.CA.PERTON-OVAR	POS.	60	4		IDC	NME	IRR	1.5
11			55	5		IDC	MASS	IRR	1.2
12	postchemo-ERPR+		68	6		IDC	MASS	SPIC.	1.2
13	SUSP.U/S RT.	POS	56	5		ILC	MASS	IRR	2
14			47	4		DCIS	NME	IRR	7
15			36	5		IDC	MASS	IRR	1.6
16			54	5		IDC	NME	IRR	3.7
17			60	5		IDC	MASS	IRR	3
18			41	5		IDC	MASS	SPIC.	2.1
19			33	4		DCIS	NME	SPIC.	6
20			44	5		IDC+DCIS	MASS	IRR	1.2
21			47	5		IDC +DCIS	MASS	SPIC.	2.5
22			45	5		IDC	MASS	IRR	0.9
23			54	4		IDC	MASS	IRR	1.5
24			41	4		DCIS	NME	IRR	2.4
25			46	4		IDC	MASS	SPIC.	1.3
26			37	4		IDC+DCIS	NME	IRR	6
27			45	5		IDC+DCIS	MASS	IRR	3.5
28			39	5		IDC+DCIS	MASS	IRR	3.6
29			48	4		OTHER(SOLID PAPILLARY)	MASS	SPIC.	1.6
30	BREAST CA 11/16 NEOADJ.	NO	40	6		IDC	NME	IRR	5.2
31	PALP.BREAST CA LT	NO	52	6		IDC	NME	IRR	5.3
32			37	5		IDC + DCIS	NME	IRR	6.2
33			51	4		IDC+DCIS	MASS	IRR	0.7

Εικόνα 18: Στιγμιότυπο 1 από το ολικό excel αρχείο.

CURVE MORPHOLOGY	CURVE TYPE	BREAST DENSITY	BPE	FEEDING VESSEL	INTER.ENHANCEMENT	DIFFUSION	ADC	FOCALITY	TUMOR GRADE	ER	PR	CERB-2	Ki-67	P63	E-CADHERIN	AXILLARY LYMPH NODES	OTHER FINDINGS
3	SUS	D	MIN	Y	RIM	HIGH	LOW?	U	3	N	N	Y+++	5				
3	SUS	B	MIN	N	HOMO	LOW?	?	U	3	Y	Y	Y++	15				
1	BEN.	B	MARK	Y	HETER.	LOW	HIGH	MF	3	N	N	Y++	50				
3	SUS	D	MARK	Y	HETER.	HIGH	LOW	U	3	Y	Y	Y	95				
1	BEN.	D	MIN	N	HOMOG.	HIGH	LOW	U	3	Y	Y	N	90				
	SUSP	D	MIN						2	Y	Y	N	15				
3	SUSP	B	MARK	N	HETER.	NA	NA	U	2	Y++	N	Y++	10				
3	SUSP	D	MIN	Y	HETERO.	HIGH	LOW	U	2	Y	Y	N	15				
2	SUSP	B	MIN	N	HETERO.	HIGH	HIGH	MF		Y++	Y++	N					
2	SUSP	C	MOD	Y	RIM	HIGH	LOW	U	3	N	N	Y	30				
2	SUSP	C	MIN	Y	HETER	HIGH	LOW	U	3	N	N	N	90				
3	SUSP	C	MIN	N	RIM	HIGH	HIGH	MF	2	Y	Y	N	5				
3	SUSP	C	MOD.	Y	RIM	HIGH	LOW	MF	2	Y++	Y+++	Y	10				
2	SUS	D	MARK	Y	HETER	HIGH	HIGH?	MC	3	Y+	N	N/A					
3	SUS	D	MARK	Y	INHOMOG	HIGH	LOW	MF	2	Y+++	Y+++	N	25				
2	SUS.	D	MIN.	Y	RIM	HIGH	LOW	U	3	Y+	Y+	Y	40				
3	BEN.	C	MIN	N	INHOMOG.	NA	NA	U	2	Y	Y	N	20				
	SUS	D	MIN	N				MF	3	N	N	Y++	47				
1	BEN.	D	MIN	Y	INHOMOG.	HIGH	HIGH	U	3	Y+++	Y+++	Y+	15				
1	BEN.	B	MIN.	Y	INHOMOG.	HIGH	LOW	MF	2	Y++	Y++	Y+++	8				
2	SUS	D	MARK	Y	INHOM.	HIGH	LOW	MC	3	N	N	Y++	22				
2	SUS	D	MARK	Y	INHOM	HIGH	LOW	MC	3	Y	Y	Y++	40				
3	SUS	D	MIN	N	INHOMOG	HIGH	LOW	U	3	Y	N	Y++	80				
3	SUS.	D	MIN.	Y	RIM	HIGH	LOW	U	2	Y++	Y++	N	4				
2	SUSP	D	MIN	Y	INHOM	HIGH	LOW	MF	2	Y+++	Y++	N	20				
2	SUSP	D	MARK	Y	INHOMOG	HIGH	HIGH	MF	3	Y+++	Y++	Y	40				

Εικόνα 19: Στιγμιότυπο 2 από το ολικό excel αρχείο.

MORPHOLOGY	BORDERS	TUMOR SIZE	CURVE MORPHOLOGY	ADC	TUMOR GRADE
MASS	IRR	0.70	2.00	HIGH	1
MASS	IRR	1.30	3.00	HIGH	1
MASS	IRR	0.60	3.00	LOW	1
MASS	IRR	1.50	2.00	LOW	2
NME	IRR	2.80	2.00	HIGH	2
MASS	SPIC	2.00	3.00	LOW	2
MASS	IRR	3.00	3.00	HIGH	2
MASS	SPIC	2.10	3.00	LOW	2
MASS	SPIC	1.20	3.00	LOW	2
NME	IRR	6.50	3.00	LOW	3
MASS	IRR	1.60	3.00	LOW	3
MASS	SPIC	3.80	3.00	LOW	3
NME	IRR	6.80	1.00	HIGH	3
MASS	IRR	1.90	3.00	LOW	3
NME	IRR	1.50	1.00	LOW	3
MASS	IRR	1.60	2.00	LOW	3
NME	IRR	3.70	2.00	LOW	3
NME	SPIC	6.00	2.00	HIGH	3
MASS	SPIC	2.50	2.00	LOW	3
NME	IRR	2.40	1.00	HIGH	3
NME	IRR	6.00	2.00	LOW	3
MASS	SMOOTH	6.70	3.00	LOW	2
MASS	SPIC	1.30	1.00	LOW	2
NME	IRR	5.20	2.00	LOW	2
NME	IRR	6.20	3.00	LOW	2

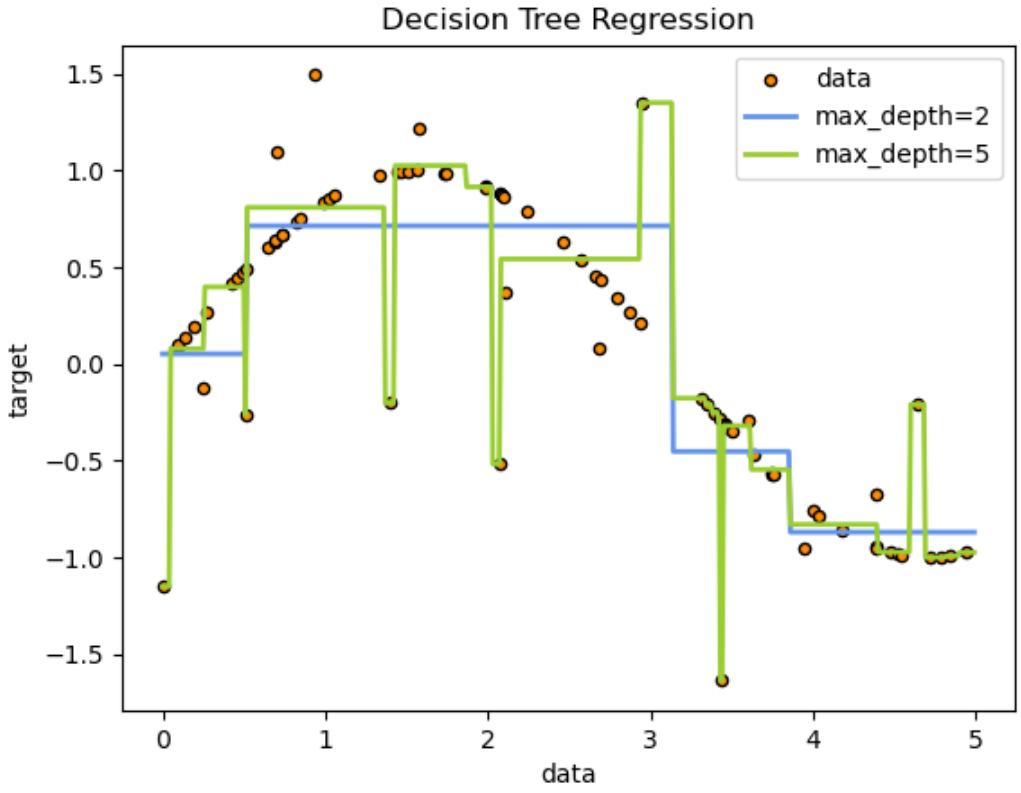
Εικόνα 20: Στιγμιότυπο από το excel αρχείο για την πρόβλεψη του δείκτη Tumor Grade.

Αφού τα δεδομένα οργανώθηκαν σωστά παρακάτω θα αναφερθούν οι ανεπιτυχείς προσπάθειες που έγιναν για την πρόβλεψη των ιστολογικών δεικτών:

3.3 Προσπάθεια με χρήση Python και εφαρμογή του ταξινομητή Δέντρων Αποφάσεων (Decision Trees Classifier) και Παλινδρόμησης των Δέντρων Αποφάσεων (Decision Trees Regression):

Επεξήγηση των Δέντρων Αποφάσεων

Τα Δέντρα Αποφάσεων (DT) είναι μια μη παραμετρική εποπτευόμενη μέθοδος μηχανικής μάθησης που χρησιμοποιείται για ταξινόμηση και παλινδρόμηση. Ο στόχος είναι να δημιουργηθεί ένα μοντέλο που προβλέπει την τιμή μιας μεταβλητής στόχου μαθαίνοντας απλούς κανόνες απόφασης που συνάγονται από τα χαρακτηριστικά δεδομένων. Ένα δέντρο μπορεί να θεωρηθεί ως μια τμηματικά σταθερή προσέγγιση. Για παράδειγμα, στην παρακάτω εικόνα φαίνεται πως τα δέντρα αποφάσεων μαθαίνουν από δεδομένα να προσεγγίζουν μια καμπύλη ημιτόνου με ένα σύνολο κανόνων απόφασης if-then-else. Όσο πιο ‹‹βαθύ›› είναι το δέντρο, τόσο πιο περίπλοκοι είναι οι κανόνες απόφασης και τόσο πιο κατάλληλο είναι το μοντέλο[20]:



Εικόνα 21: Προσπάθεια προσέγγισης του ημιτόνου από το Δέντρο Αποφάσεων[20].

[20] https://scikit-learn.org/stable/modules/tree.html

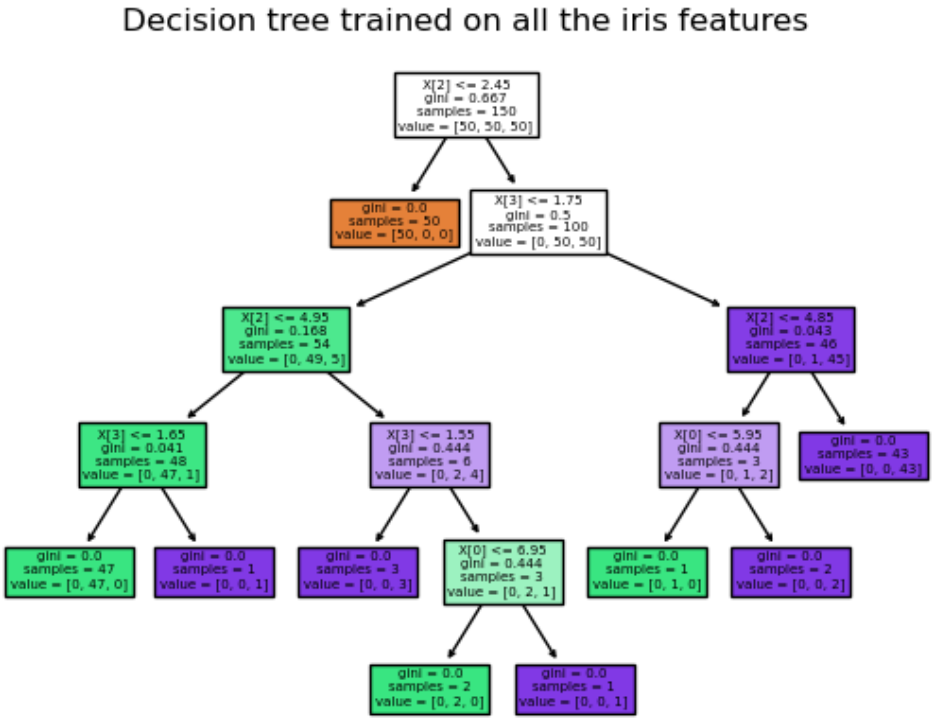
Πλεονεκτήματα των Δέντρων Αποφάσεων :

- Εύκολα στην κατανόηση και στην ερμηνεία. Τα δέντρα μπορούν να οπτικοποιηθούν.
- Απαιτούν λίγη προετοιμασία δεδομένων. Άλλες τεχνικές απαιτούν συχνά κανονικοποίηση δεδομένων, πρέπει να δημιουργηθούν εικονικές μεταβλητές και να αφαιρεθούν κενές τιμές.
- Η πολυπλοκότητα ενός Δέντρου Αποφάσεων (δηλαδή, η πρόβλεψη δεδομένων) είναι λογαριθμική ως προς τον αριθμό των σημείων δεδομένων που χρησιμοποιούνται για την εκπαίδευση του δέντρου.
- Ικανά να χειρίζονται τόσο αριθμητικά όσο και κατηγορικά δεδομένα. Ωστόσο, η εφαρμογή scikit-learn δεν υποστηρίζει κατηγορικές μεταβλητές προς το παρόν. Άλλες τεχνικές είναι συνήθως εξειδικευμένες στην ανάλυση συνόλων δεδομένων που έχουν μόνο έναν τύπο μεταβλητής.
- Ικανά να χειριστούν προβλήματα πολλαπλών εξόδων.
- Χρησιμοποιούν ένα μοντέλο ‹‹λευκού κουτιού››. Εάν μια δεδομένη κατάσταση είναι παρατηρήσιμη σε ένα μοντέλο, η εξήγηση της συνθήκης εξηγείται εύκολα με δυαδική λογική Boolean[17]. Αντίθετα, σε ένα μοντέλο μαύρου κουτιού (π.χ. σε ένα τεχνητό νευρωνικό δίκτυο), τα αποτελέσματα μπορεί να είναι πιο δύσκολο να ερμηνευτούν.
- Δυνατότητα επικύρωσης ενός μοντέλου με τη χρήση στατιστικών δοκιμών. Αυτό καθιστά δυνατό τον υπολογισμό της αξιοπιστίας του μοντέλου.
- Αποδίδει καλά ακόμα κι αν οι παραδοχές του παραβιάζονται κάπως από το πραγματικό μοντέλο από το οποίο δημιουργήθηκαν τα δεδομένα.

Μειονεκτήματα των Δέντρων Αποφάσεων:

- Μπορούν να δημιουργηθούν υπερβολικά πολύπλοκα δέντρα που δεν γενικεύουν καλά τα δεδομένα. Αυτό ονομάζεται υπερπροσαρμογή (overfitting). Μηχανισμοί όπως το κλάδεμα, δηλαδή της μείωσης του μεγέθους του δέντρου, ο καθορισμός του ελάχιστου αριθμού δειγμάτων που απαιτούνται σε έναν κόμβο φύλλων ή ο καθορισμός του μέγιστου βάθους του δέντρου είναι απαραίτητοι για την αποφυγή αυτού του προβλήματος.
- Τα δέντρα αποφάσεων μπορεί να είναι ασταθή επειδή μικρές παραλλαγές στα δεδομένα μπορεί να έχουν ως αποτέλεσμα τη δημιουργία ενός εντελώς διαφορετικού δέντρου. Αυτό το πρόβλημα μετριάζεται χρησιμοποιώντας δέντρα απόφασης μέσα σε ένα σύνολο.

- Οι προβλέψεις των δέντρων απόφασης δεν είναι ούτε ομαλές ούτε συνεχείς, αλλά τμηματικά σταθερές προσεγγίσεις όπως φαίνεται στην Εικόνα 15. Επομένως, δεν είναι αποδοτικά στην παρέκταση (extrapolation), δηλαδή στην εκτίμηση με βάση την τάση των προηγούμενων μεταβλητών.
- Το πρόβλημα της εκμάθησης ενός δέντρου βέλτιστων αποφάσεων είναι γνωστό ότι είναι NP-complete υπό διάφορες πτυχές βελτιστοποίησης και ακόμη και για απλές έννοιες. Κατά συνέπεια, οι πρακτικοί αλγόριθμοι μάθησης του δέντρου αποφάσεων βασίζονται σε ευρετικούς αλγόριθμους όπως ο ‹‹άπληστοι›› αλγόριθμος (greedy algorithms) όπου λαμβάνονται τοπικά βέλτιστες αποφάσεις σε κάθε κόμβο. Τέτοιοι αλγόριθμοι δεν μπορούν να εγγυηθούν την επιστροφή του γενικά βέλτιστου δέντρου αποφάσεων. Αυτό μπορεί να μετριαστεί με την εκπαίδευση πολλών δέντρων σε έναν εκπαιδευόμενο συνόλου, όπου τα χαρακτηριστικά και τα δείγματα δειγματίζονται τυχαία με αντικατάσταση.



Εικόνα 22: Απεικόνιση του Δέντρου Αποφάσεων μετά την εκτέλεση του αλγορίθμου[20]

Για αρχή εγκαταστάθηκαν κάποια python πακέτα τα οποία θα βοηθούσαν στην ανάγνωση excel αρχείων και την αποθήκευση και οργάνωση των δεδομένων σε dataframes. Έπειτα, χωρίστηκαν τα δεδομένα σε train και test. Τέλος, κλήθηκε ο classifier. Παρακάτω παρατίθεται ο κώδικας που χρησιμοποιήθηκε:

```
1. import xlrd
2. from sklearn.model_selection import train_test_split
3. import pandas as pd
4. from sklearn import metrics
5. import numpy as np
6. from sklearn.tree import DecisionTreeClassifier
7. from sklearn.preprocessing import StandardScaler
8.
9.
10. loc = ("C:\\Users\\anton\\OneDrive\\Desktop\\book_1_ki_67_test.xls")
11. wb = xlrd.open_workbook(loc)
12. sheet = wb.sheet_by_index(0)
13. morphology_set = []
14. borders_set = []
15. tumor_size_set = []
16. curve_morphology_set = []
17. adc = []
18. ki_67_set = []
19.
20. for i in range(1, sheet.nrows, 1):
21.     morphology_set.append(sheet.cell_value(i, 0))
22.     borders_set.append(sheet.cell_value(i, 1))
23.     tumor_size_set.append(sheet.cell_value(i, 2))
24.     curve_morphology_set.append(sheet.cell_value(i, 3))
25.     adc.append(sheet.cell_value(i, 4))
26.     ki_67_set.append(sheet.cell_value(i, 5))
27.
28. morphology_set = pd.DataFrame(morphology_set)
29. borders_set = pd.DataFrame(borders_set)
30. tumor_size_set = pd.DataFrame(tumor_size_set)
31. curve_morphology_set = pd.DataFrame(curve_morphology_set)
32. adc = pd.DataFrame(adc)
33. ki_67_set = pd.DataFrame(ki_67_set)
34.
35.
36. frames = [morphology_set, borders_set, tumor_size_set, curve_ morphology_set,
    adc_set]
37. input_data = pd.concat(frames, axis=1)
38. input_data = pd.DataFrame(input_data)
```

```
39. input_data = input_data.set_axis(['a', 'b', 'c', 'd', 'e'], axis=1, inplace=False)
40. input_data = pd.get_dummies(input_data, columns = ['a', 'b', 'd', 'e'])
41. input_data_train, input_data_test, ki_67_set_train, ki_67_set_test =
    train_test_split(input_data, ki_67_set, test_size=0.1, random_state=0, shuffle=0)
42. sc_X = StandardScaler()
43. input_data_train = sc_X.fit_transform(input_data_train)
44. input_data_test = sc_X.transform(input_data_test)
45.
46. classifier = DecisionTreeClassifier(criterion="gini", max_depth = None,
    random_state=0)
47. classifier = classifier.fit(input_data_train, ki_67_set_train)
48. y_pred = classifier.predict(input_data_test)
49.
50.
51. print(y_pred)
52. print(ki_67_set_test)
53. print('AccuracyScore:',metrics.accuracy_score(ki_67_set_test,y_pred))
```

Προκειμένου να βγουν ορθότερα αποτελέσματα με μεγαλύτερη ακρίβεια χρησιμοποιήθηκε one-hot encoding με την εντολή *pd.get_dummies*. Το one-hot encoding ουσιαστικά παρατηρεί σε κάθε στήλη ποιες είναι οι διαθέσιμες τιμές και δημιουργεί για κάθε τιμή μια κατηγορία στην οποία για κάθε γραμμή, ασθενή δηλαδή, συμπληρώνει την τιμή 1 αν ο ασθενής έχει την συγκεκριμένη τιμή ή 0 αν δεν την έχει. Παρακάτω φαίνεται ένα στιγμιότυπο από το πως διαμορφώθηκαν τα δεδομένα:

```
C:\Users\anton\PycharmProjects\pythonProject\venv\Scripts\python.exe C:/Users/anton/PycharmProjects/pythonProject/main.py
      c  a_MASS  a_MASS+NME  a_NME  b_IRR  ...  d_1.0  d_2.0  d_3.0  e_HIGH  e_LOW
0   3.6       1          0      0      1  ...      0      0      1       0      1
1   1.5       0          0      1      1  ...      1      0      0       0      1
2   1.0       1          0      0      1  ...      0      0      1       0      1
3   2.0       1          0      0      1  ...      0      0      1       1      0
4   0.8       1          0      0      1  ...      0      0      1       0      1
5   0.5       1          0      0      0  ...      0      0      1       0      1
6   1.9       1          0      0      1  ...      0      0      1       0      1
7   0.6       1          0      0      1  ...      0      0      1       0      1
8   1.6       1          0      0      0  ...      0      0      1       0      1
9   3.8       1          0      0      0  ...      0      0      1       0      1
10  0.7       0          0      1      1  ...      0      0      1       0      1
11  1.3       1          0      0      1  ...      0      0      1       1      0
12  3.3       0          1      0      1  ...      0      0      1       0      1
13  2.0       1          0      0      1  ...      0      0      1       0      1
14  3.2       0          0      1      1  ...      0      0      1       1      0
15  6.2       0          0      1      1  ...      0      0      1       0      1
16  6.0       0          0      1      1  ...      0      1      0       0      1
17  1.2       1          0      0      1  ...      0      0      1       0      1
18  1.5       1          0      0      1  ...      0      1      0       0      1
19  1.6       1          0      0      1  ...      0      0      1       0      1
20  6.7       0          0      1      1  ...      1      0      0       1      0
21  6.5       0          0      1      1  ...      0      1      0       0      1
22  1.3       1          0      0      1  ...      0      1      0       0      1
23  5.3       0          0      1      1  ...      0      1      0       1      0
24  1.6       1          0      0      0  ...      0      0      1       0      1
25  2.5       1          0      0      0  ...      0      1      0       0      1
26  3.5       1          0      0      1  ...      0      1      0       0      1
27  0.8       1          0      0      1  ...      0      0      1       1      0
28  6.7       1          0      0      0  ...      0      0      1       0      1
29  2.7       1          0      0      0  ...      0      0      1       0      1
30  3.5       1          0      0      0  ...      0      0      1       0      1
31  0.7       1          0      0      1  ...      0      1      0       1      0
32  2.8       0          0      1      1  ...      0      1      0       1      0
33  1.3       1          0      0      0  ...      1      0      0       0      1
34  2.1       1          0      0      0  ...      0      0      1       0      1
```

Εικόνα 23: Διαμόρφωση δεδομένων μετά το One Hot Encoding

Επίσης εκτός από το one-hot encoding χρησιμοποιήθηκε και η τυποποίηση χαρακτηριστικών (standardize features) των δεδομένων εισόδου που γίνονται χρήση και για την εκπαίδευση αλλά και για την πρόβλεψη. Η τυποποίηση χαρακτηριστικών γίνεται με σκοπό την αφαίρεση της μέσης τιμής και την κλιμάκωση (scaling) της διακύμανσης μονάδας. Παρ' όλα ταύτα τα αποτελέσματα δεν είναι ικανοποιητικά και κυμαίνονται 0% - 20% για το Ki-67. Το ποσοστό επιτυχίας επετεύχθη με ένα μέγεθος δεδομένων προς δοκιμή της τάξεως του 10%.

Έγιναν επιπλέον προσπάθειες δίνοντας επιπλέον παραμέτρους στην εντολή του ταξινομητή. Συγκεκριμένα αναλύονται ποιες παράμετροι έκαναν κάποια διαφορά στο ποσοστό ακρίβειας είτε προς το καλύτερο είτε προς το χειρότερο:

❖ **Ki-67**

Πίνακας 2: Δοκιμές decision tree classifier για το Ki-67 για κριτήριο gini και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
gini (default)	None (default)	0	20
gini (default)	2	0	0
gini (default)	3	0	20
gini (default)	4	0	20
gini (default)	5	0	20
gini (default)	6	0	0
gini (default)	7	0	0
gini (default)	8	0	20
gini (default)	9	0	20
gini (default)	10	0	20
gini (default)	11	0	20
gini (default)	>11	0	20

Πίνακας 3: Δοκιμές decision tree classifier για το Ki-67 για κριτήριο entropy και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
entropy	None (default)	0	11.11
entropy	2	0	11.11
entropy	3	0	11.11
entropy	4	0	0
entropy	5	0	11.11
entropy	6	0	0
entropy	7	0	11.11
entropy	8	0	11.11
entropy	9	0	11.11
entropy	10	0	11.11
entropy	11	0	11.11
entropy	>11	0	11.11

❖ **TUMOR GRADE**

Πίνακας 4: Δοκιμές decision tree classifier για το Tumor Grade για κριτήριο gini και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
gini (default)	None (default)	0	80
gini (default)	2	0	20
gini (default)	3	0	20
gini (default)	4	0	40
gini (default)	5	0	80
gini (default)	6	0	80
gini (default)	7	0	80
gini (default)	8	0	80
gini (default)	9	0	80
gini (default)	10	0	80
gini (default)	11	0	80
gini (default)	>11	0	80

Πίνακας 5: Δοκιμές decision tree classifier για το Tumor Grade για κριτήριο entropy και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
entropy	None (default)	0	40
entropy	2	0	40
entropy	3	0	20
entropy	4	0	40
entropy	5	0	60
entropy	6	0	40
entropy	7	0	60
entropy	8	0	60
entropy	9	0	40
entropy	10	0	40
entropy	11	0	40
entropy	>11	0	40

Μέγιστο ποσοστό επιτυχίας: 80%

❖ **ER, PR, CERB-2**

Πίνακας 6: Δοκιμές decision tree classifier για το ER, PR, CERB-2 για κριτήριο gini και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
gini (default)	None (default)	0	60
gini (default)	2	0	60
gini (default)	3	0	60
gini (default)	4	0	60
gini (default)	5	0	40
gini (default)	6	0	80
gini (default)	7	0	40
gini (default)	8	0	60
gini (default)	9	0	60
gini (default)	10	0	60
gini (default)	11	0	60
gini (default)	>11	0	60

Πίνακας 7: Δοκιμές decision tree classifier για το ER, PR, CERB-2 για κριτήριο entropy και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**Ακρίβεια (%)**
entropy	None (default)	0	60
entropy	2	0	60
entropy	3	0	60
entropy	4	0	60
entropy	5	0	40
entropy	6	0	80
entropy	7	0	40
entropy	8	0	60
entropy	9	0	60
entropy	10	0	60
entropy	11	0	60
entropy	>11	0	60

Μέγιστο ποσοστό επιτυχίας: 80%

Αξίζει να σημειωθούν τα εξής πράγματα:

1) Έγιναν δοκιμές και με χρήση άλλων παραμέτρων συνδυαστικά με τις παραμέτρους που αναφέρθηκαν στους πίνακες. Οι παράμετροι αυτοί ήταν οι εξής:

   min_samples_split, min_samples_leaf, min_weight_fraction_leaf, max_features, max_leaf_nodes, min_impurity_decrease, class_weight, ccp_alpha. **Δεν αναφέρθηκαν στους πίνακες καθώς δεν έφεραν καμία μεταβολή στην ακρίβεια.**

2) Παρατηρήθηκε πως η σειρά με την οποία ήταν τοποθετημένοι οι ασθενείς, η οποία ήταν αυτή από το ολικό excel, έπρεπε να αλλάξει καθώς το δείγμα που λάμβανε για εκπαίδευση ο αλγόριθμος δεν ήταν αρκετά καλό για να αποδώσει όσον το δυνατόν μεγαλύτερο ποσοστό επιτυχίας. Έγιναν αρκετές δοκιμές με ανακατατάξεις των ασθενών και τελικά αποφασίστηκε να τοποθετηθούν οι ασθενείς με τρόπο τέτοιο ώστε να έχει δείγματα ο αλγόριθμος από ποικίλες περιπτώσεις στο κομμάτι που διαλέγει για εκπαίδευση.

Επεξήγηση των παραμέτρων της συνάρτησης που χρησιμοποιήθηκαν:

- *Criterion* (επιλογές: “gini”, “entropy”}. Η παράμετρος για τη μέτρηση της ποιότητας ενός διαχωρισμού . Υποστηριζόμενα κριτήρια είναι το "gini" για την Gini impurity και η "entropy" για το κέρδος πληροφοριών. Το Gini Impurity[5] είναι η πιθανότητα εσφαλμένης ταξινόμησης ενός τυχαία επιλεγμένου στοιχείου στο σύνολο δεδομένων, εάν είχε επισημανθεί τυχαία σύμφωνα με την κατανομή κλάσεων στο σύνολο δεδομένων. Η εντροπία[6] παρέχει ένα μέτρο της μέσης ποσότητας πληροφοριών που απαιτείται για την αναπαράσταση ενός γεγονότος που προέρχεται από μια κατανομή πιθανότητας για μια τυχαία μεταβλητή.
- *max_depth* (επιλογές: None ή κάποιος ακέραιος) Το μέγιστο βάθος του δέντρου. Αν χρησιμοποιηθεί η επιλογή None, τότε οι κόμβοι επεκτείνονται μέχρι να γίνουν όλα τα φύλλα καθαρά ή έως ότου όλα τα φύλλα περιέχουν λιγότερα από min_samples_split δείγματα, δηλαδή έως ότου τα φύλλα να περιέχουν τον ελάχιστο αριθμό δειγμάτων που απαιτούνται για τον διαχωρισμό ενός εσωτερικού κόμβου.

Σε αυτό το σημείο έγινε αλλαγή από DecisionTreeClasifier σε DecisionTreeRegressor. Παρακάτω φαίνεται ο κώδικας που χρησιμοποιήθηκε:

```
1. import xlrd
2. from sklearn.model_selection import train_test_split
3. import pandas as pd
4. from sklearn import metrics
5. import numpy as np
6. from sklearn.tree import DecisionTreeRegressor
7. from sklearn.preprocessing import StandardScaler
8.
9. loc= ("C:\\Users\\anton\\OneDrive\\Desktop\\book_1_ki_67_test.xls")
10. wb = xlrd.open_workbook(loc)
11. sheet = wb.sheet_by_index(0)
12. morphology_set = []
13. borders_set = []
14. tumor_size_set = []
15. curve_morphology_set = []
16. adc = []
17. ki_67_set = []
18.
19. for i in range(1, sheet.nrows, 1):
20.    morphology_set.append(sheet.cell_value(i, 0))
21.   borders_set.append(sheet.cell_value(i, 1))
22.   tumor_size_set.append(sheet.cell_value(i, 2))
23.   curve_morphology_set.append(sheet.cell_value(i, 3))
24.   adc.append(sheet.cell_value(i, 4))
25.   ki_67_set.append(sheet.cell_value(i, 5))
26.
27. morphology_set = pd.DataFrame(morphology_set)
28. borders_set = pd.DataFrame(borders_set)
29. tumor_size_set = pd.DataFrame(tumor_size_set)
30. curve_morphology_set = pd.DataFrame(curve_morphology_set)
31. adc = pd.DataFrame(adc)
32. ki_67_set = pd.DataFrame(ki_67_set)
33. frames = [morphology_set, borders_set, tumor_size_set, curve_morphology_set,
    adc]
34. input_data = pd.concat(frames, axis=1)
35. input_data = pd.DataFrame(input_data)
36. input_data = input_data.set_axis(['a', 'b', 'c', 'd', 'e'], axis=1, inplace=False)
37. input_data = pd.get_dummies(input_data, columns = ['a', 'b', 'd', 'e'])
38. ki_67_set = pd.get_dummies(ki_67_set)
39. input_data_train, input_data_test, ki_67_set_train, ki_67_set_test =
    train_test_split(input_data, ki_67_set, test_size=0.1, random_state=0, shuffle=0)
```

```
sc_X = StandardScaler()
input_data_train = sc_X.fit_transform(input_data_train)
input_data_test = sc_X.transform(input_data_test)
classifier = DecisionTreeRegressor(criterion="squared_error", max_depth =
    None, random_state=0)
regressor = regressor.fit(input_data_train, ki_67_set_train)
y_pred = regressor.predict(input_data_test)


print(y_pred)
print(ki_67_set_test)
print('Mean Absolute Error:', metrics.mean_absolute_error(ki_67_set_test,
    y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(ki_67_set_test, y_pred))
print('Root Mean Squared Error:',
    np.sqrt(metrics.mean_squared_error(ki_67_set_test, y_pred)))
print('R^2:', metrics.r2_score(ki_67_set_test, y_pred))

```

❖ **Ki-67**

Πίνακας 8: Δοκιμές decision tree regressor για το Ki-67 για κριτήριο squared_error και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**MAE**	**MSE**	**RMSE**	**R^2**
squared_error (default)	None (default)	0	24.4	1030.8	32.10	-2.38
squared_error (default)	2	0	24.4	1030.8	32.10	-2.38
squared_error (default)	3	0	25.31	975.90	31.23	-2.21
squared_error (default)	4	0	27.45	1077.31	32.82	-2.54
squared_error (default)	5	0	26.45	1051.92	32.43	-2.45

Πίνακας 9: Δοκιμές decision tree regressor για το Ki-67 για κριτήριο poisson και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**MAE**	**MSE**	**RMSE**	**R^2**
poisson	None (default)	0	16.2	718.6	26.81	-1.36
poisson	2	0	14.4	306.46	17.51	-0.007
poisson	3	0	14.27	309.11	17.58	-0.016
poisson	4	0	12.58	298.67	17.28	-0.018

poisson	5	0	17.38	505.29	22.48	-0.66

❖ **TUMOR GRADE**

Πίνακας 10: Δοκιμές decision tree regressor για το Tumor Grade για κριτήριο squared_error και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**MAE**	**MSE**	**RMSE**	**R^2**
squared_error (default)	None (default)	0	0.2	0.2	0.45	0.167
squared_error (default)	2	0	0.538	0.312	0.559	-0.301
squared_error (default)	3	0	0.559	0.364	0.603	-0.519

Παρατηρήθηκε πως για τον δείκτη tumor grade το καλύτερο κριτήριο είναι το squared_error για max_depth=None. Το κριτήριο poisson έδινε ίδια αποτελέσματα.

❖ **ER, PR, CERB-2**

Πίνακας 11: Δοκιμές decision tree regressor για το ER, PR, CERB-2 για κριτήριο squared_error και διαφορετικές τιμές max_depth.

criterion	**max_depth**	**random_state**	**MAE**	**MSE**	**RMSE**	**R^2**
squared_error (default)	None (default)	0	0.4	0.4	0.632	-0.666
squared_error (default)	2	0	0.42	0.277	0.526	-0.155
squared_error (default)	3	0	0.415	0.248	0.498	-0.034
squared_error (default)	4	0	0.303	0.235	0.485	0.022
squared_error (default)	5	0	0.347	0.29	0.538	-0.207

Πίνακας 12: Δοκιμές decision tree regressor για το ER, PR, CERB-2 για κριτήριο poisson και διαφορετικές τιμές max_depth.

criterion	max_depth	random_state	MAE	MSE	RMSE	R^2
poisson	None (default)	0	0.333	0.189	0.435	0.210
poisson	2	0	0.429	0.267	0.517	-0.115
poisson	3	0	0.413	0.274	0.524	-0.145
poisson	4	0	0.366	0.217	0.466	0.09
poisson	5	0	0.333	0.189	0.435	-0.210

Παρατηρήθηκε πως για τους δείκτες er, pr, cerb-2 το καλύτερο κριτήριο είναι το poisson για max_depth=None, ενώ για τιμές του max_depth ≥ 5 το αποτελέσματα που λαμβάνονται είναι ίδια με αυτά με την τιμή να είναι στο None.

- *Criterion* (επιλογές: "squared_error", "friedman_mse", "absolute_error", "poisson"):
  Η λειτουργία για τη μέτρηση της ποιότητας ενός διαχωρισμού. Τα υποστηριζόμενα κριτήρια είναι "squared_error" για το μέσο τετράγωνο σφάλμα, το οποίο ισούται με τη μείωση της διακύμανσης ως κριτήριο επιλογής χαρακτηριστικών και ελαχιστοποιεί την απώλεια L2 χρησιμοποιώντας τον μέσο όρο κάθε τερματικού κόμβου, "friedman_mse", ο οποίος χρησιμοποιεί μέσο τετράγωνο σφάλμα με βαθμολογία βελτίωσης Friedman για πιθανους διαχωρισμούς, "absolute_error" για το μέσο απόλυτο σφάλμα, το οποίο ελαχιστοποιεί την απώλεια L1 χρησιμοποιώντας τη διάμεσο κάθε τερματικού κόμβου και "poisson" που χρησιμοποιεί μείωση στην απόκλιση Poisson.

Παρατηρήθηκε πως διαφορά στα αποτελέσματα έκανε η αλλαγή από squared_error σε poisson, χωρίς όμως και πάλι τα αποτελέσματα να είναι ενθαρρυντικά.

Αναφορικά με τα αποτελέσματα που εξήχθησαν από την παρεμβολή του Δέντρου Αποφάσεων παραθέτονται οι τύποι υπολογισμού για το MSE, RMSE, MAE, R^2 οι οποίοι αναφέρονται στους πίνακες 3 – 13, που είναι παράμετροι υπολογισμού ακριβείας και προσαρμογής των αποτελεσμάτων στα δεδομένα:

Mean Squared Error: Μέσο Τετραγωνικό Σφάλμα

- $$\mathbf{MSE} = \frac{\sum_1^{\mathrm{n}}(\text{πραγματική τιμή} - \text{προβλεπόμενη τιμή})^2}{\text{Πλήθος δεδομένων}}$$

Root Mean Squared Error: Τετραγωνική Ρίζα του Μέσου Τετραγωνικού Σφάλματος

- $\mathbf{RMSE} = \sqrt{\mathrm{MSE}}$

Mean Absolute Error: Μέσο Απόλυτο Σφάλμα

- $\mathbf{MAE} = \frac{\sum_1^n |(\text{πραγματική τιμή} - \text{προβλεπόμενη τιμή})|}{\text{Πλήθος δεδομένων}}$

R^2

- $\mathbf{R^2} = 1 - \frac{\sum_1^n (\text{πραγματική τιμή} - \text{προβλεπόμενη τιμή})^2}{\sum_1^n (\text{πραγματική τιμή} - \text{μέση τιμή})^2}$

3.4 Προσπάθεια με χρήση Python και χρήση της συνάρτησης polynomial.polynomial.polyfit του πακέτου numpy

Σε αυτό το σημείο έγινε προσπάθεια της δημιουργίας κώδικα που ουσιαστικά με βάση τα δεδομένα θα έβρισκε ποιο πολυώνυμο θα ταίριαζε καλύτερα στα δεδομένα και με αυτό θα έκανε προσέγγιση τους συντελεστές τους. Αυτό θα επιτυγχανόταν με την συνάρτηση **numpy.polynomial.polynomial.polyfit** του πακέτου ***numpy***. Όμως με βάση αναφορές στο Github[18,19] περιορίζεται σε μονοδιάστατους πίνακες για τα δεδομένα εισόδου και έτσι μετά από μερικές προσπάθειες εγκαταλείφθηκε η ιδέα.

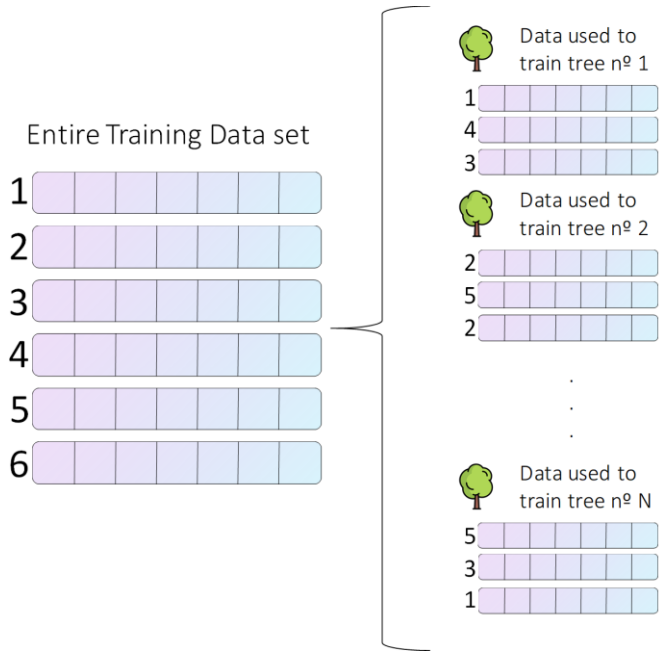
3.5 Προσπάθεια με χρήση Python και εφαρμογή του ταξινομητή Τυχαίου Δάσους (Random Forest Classifier) και Παλινδρόμησης των Τυχαίου Δάσους (Random Forest Regression):

Επεξήγηση Τυχαίου Δάσους

Ένα τυχαίο δάσος είναι ένας μετα-εκτιμητής (meta estimator) που ταιριάζει σε έναν αριθμό ταξινομητών δέντρων αποφάσεων σε διάφορα υποδείγματα του συνόλου δεδομένων και χρησιμοποιεί τον μέσο όρο για να βελτιώσει την προγνωστική ακρίβεια και τον έλεγχο της υπερπροσαρμογής (overfitting). Στον κόσμο της Τεχνητής Νοημοσύνης, τα μοντέλα Random Forest είναι ένα είδος μη παραμετρικών μοντέλων που μπορούν να χρησιμοποιηθούν τόσο για παλινδρόμηση όσο και για ταξινόμηση. Οι μέθοδοι συνόλου περιλαμβάνουν τη χρήση πολλών ‹‹μαθητών›› - δέντρων για τη βελτίωση της απόδοσης οποιουδήποτε από αυτών ξεχωριστά. Αυτές οι μέθοδοι μπορούν να περιγραφούν ως τεχνικές που χρησιμοποιούν μια ομάδα αδύναμων ‹‹μαθητών›› - δέντρων (αυτοί που κατά

μέσο όρο επιτυγχάνουν ελαφρώς καλύτερα αποτελέσματα από ένα τυχαίο μοντέλο) μαζί, προκειμένου να δημιουργήσουν ένα ισχυρότερο, συγκεντρωτικό μοντέλο - ‹‹δάσος››.

Ένα από τα κύρια μειονεκτήματα των Decision Trees είναι ότι είναι πολύ επιρρεπή στην υπερπροσαρμογή: τα καταφέρνουν καλά με τα δεδομένα εκπαίδευσης, αλλά δεν είναι τόσο ευέλικτα για να κάνουν προβλέψεις σε δείγματα που δεν έχουν αντιμετωπίσει ξανά. Αν και υπάρχουν λύσεις για αυτό, όπως το κλάδεμα των δέντρων (pruning), αυτό μειώνει την προγνωστική τους ισχύ. Γενικά είναι μοντέλα με μέτρια προκατάληψη και υψηλή διακύμανση, αλλά είναι απλά και εύκολα στην ερμηνεία. Τα μοντέλα Random Forest συνδυάζουν την απλότητα των Decision Trees με την ευελιξία και τη δύναμη ενός μοντέλου συνόλου. Σε ένα δάσος από δέντρα, ξεχνάμε την υψηλή διακύμανση ενός συγκεκριμένου δέντρου και ανησυχούμε λιγότερο για κάθε μεμονωμένο στοιχείο, έτσι μπορούμε να αναπτύξουμε δέντρα που έχουν μεγαλύτερη προγνωστική δύναμη από ένα κλαδεμένο (pruned tree). Παρόλο που τα μοντέλα Random Forest δεν προσφέρουν τόσο μεγάλη ικανότητα πρόβλεψης όσο ένα μόνο δέντρο, η απόδοσή τους είναι πολύ καλύτερη και δεν χρειάζεται να ανησυχούμε τόσο για την καλύτερη δυνατή ρύθμιση των παραμέτρων του δάσους όπως κάνουμε με τα μεμονωμένα δέντρα. Παρακάτω φαίνεται μια εικόνα που δείχνει πως αναπτύσσονται από ένα σύνολο δεδομένων τα διάφορα δέντρα του τυχαίου δάσους[21]:



Εικόνα 24: Random Forest: Πως από τα δεδομένα δημιουργούνται τα δέντρα του δάσους[21].

[21] https://towardsdatascience.com/random-forest-explained-7eae084f3ebe

```
1.  import numpy as np
2.  import xlrd
3.  import pandas as pd
4.  from sklearn.ensemble import RandomForestClassifier
5.  from sklearn.model_selection import train_test_split
6.  from sklearn.preprocessing import StandardScaler
7.
8.  loc = ("C:\\Users\\anton\\OneDrive\\Desktop\\book_1_ki_67_test.xls")
9.  wb = xlrd.open_workbook(loc)
10. sheet = wb.sheet_by_index(0)
11.
12. morphology_set = []
13. borders_set = []
14. tumor_size_set = []
15. curve_morphology_set = []
16. adc = []
17. ki_67_set = []
18.
19.
20. for i in range(1, sheet.nrows, 1):
21.     morphology_set.append(sheet.cell_value(i, 0))
22.     borders_set.append(sheet.cell_value(i, 1))
23.     tumor_size_set.append(sheet.cell_value(i, 2))
24.     curve_morphology_set.append(sheet.cell_value(i, 3))
25.     adc.append(sheet.cell_value(i, 4))
26.     ki_67_set.append(sheet.cell_value(i, 5))
27.
28. morphology_set = pd.DataFrame(morphology_set)
29. borders_set = pd.DataFrame(borders_set)
30. tumor_size_set = pd.DataFrame(tumor_size_set)
31. curve_morphology_set = pd.DataFrame(curve_morphology_set)
32. adc = pd.DataFrame(adc)
33. ki_67_set = pd.DataFrame(ki_67_set)
34.
35. frames = [morphology_set, borders_set, curve_morphology_set, adc]
36. input_data = pd.concat(frames, axis=1)
37. input_data = pd.DataFrame(input_data)
38. input_data = input_data.set_axis(['a', 'b', 'c', 'd'], axis=1, inplace=False)
39. input_data = pd.get_dummies(input_data, columns = ['a', 'b', 'c', 'd'])
40. ki_67_set = pd.get_dummies(ki_67_set)
41. input_data_train, input_data_test, ki_67_set_train, ki_67_set_test =
    train_test_split(input_data, ki_67_set, test_size=0.2, random_state=0, shuffle=0)
42.
43. sc_X = StandardScaler()
```

```
input_data_train = sc_X.fit_transform(input_data_train)
input_data_test = sc_X.transform(input_data_test)

classifier = RandomForestClassifier(criterion="entropy", max_depth = None,
random_state=0)
classifier = classifier.fit(input_data_train, ki_67_set_train)
y_pred = classifier.predict(input_data_test)


print(y_pred)
print(ki_67_set_test)
print("Accuracy:",metrics.accuracy_score(ki_67_set_test, y_pred))
```

Παρακάτω παρατίθεται ο κώδικας για το RandomForestRegressor:

```
import numpy as np
import xlrd
import pandas as pd
from sklearn.ensemble import RandomForestRegressor # Import Decision Tree
Classifier
from sklearn import metrics #Import scikit-learn metrics module for accuracy
calculation
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler

loc = ("C:\\Users\\anton\\OneDrive\\Desktop\\book_1_ki_67_test.xls")
wb = xlrd.open_workbook(loc)
sheet = wb.sheet_by_index(0)

morphology_set = []
borders_set = []
tumor_size_set = []
curve_morphology_set = []
adc = []
ki_67_set = []


for i in range(1, sheet.nrows, 1):
    morphology_set.append(sheet.cell_value(i, 0))
    borders_set.append(sheet.cell_value(i, 1))
    tumor_size_set.append(sheet.cell_value(i, 2))
    curve_morphology_set.append(sheet.cell_value(i, 3))
    adc.append(sheet.cell_value(i, 4))
    ki_67_set.append(sheet.cell_value(i, 5))

morphology_set = pd.DataFrame(morphology_set)
```

```
borders_set = pd.DataFrame(borders_set)
tumor_size_set = pd.DataFrame(tumor_size_set)
curve_morphology_set = pd.DataFrame(curve_morphology_set)
adc = pd.DataFrame(adc)
ki_67_set = pd.DataFrame(ki_67_set)

frames = [morphology_set, borders_set, curve_morphology_set, adc]
input_data = pd.concat(frames, axis=1)
input_data = pd.DataFrame(input_data)
input_data = input_data.set_axis(['a', 'b', 'c', 'd'], axis=1, inplace=False)
input_data = pd.get_dummies(input_data, columns = ['a', 'b', 'c', 'd'])
ki_67_set = pd.get_dummies(ki_67_set)
input_data_train, input_data_test, ki_67_set_train, ki_67_set_test =
    train_test_split(input_data, ki_67_set, test_size=0.2, random_state=0, shuffle=0)
ki_67_set_train = np.ravel(ki_67_set_train)
sc_X = StandardScaler()
input_data_train = sc_X.fit_transform(input_data_train)
input_data_test = sc_X.transform(input_data_test)

classifier = RandomForestRegressor(criterion="squared_error", max_depth =
    None, random_state=0)
classifier = classifier.fit(input_data_train, ki_67_set_train)
y_pred = classifier.predict(input_data_test)


print(y_pred)
print(ki_67_set_test)
print('Mean Absolute Error:', metrics.mean_absolute_error(ki_67_set_test,
    y_pred))
print('Mean Squared Error:', metrics.mean_squared_error(ki_67_set_test, y_pred))
print('Root Mean Squared Error:',
    np.sqrt(metrics.mean_squared_error(ki_67_set_test, y_pred)))
print('R^2:', metrics.r2_score(ki_67_set_test, y_pred))
```

Παρ' ότι πρόκειται για μια διαφορετική μέθοδο, τα αποτελέσματα είναι ακριβώς τα ίδια με αυτά τών Decision Trees Classifier και Regressor.

Συζήτηση

Είναι φανερό πως λόγω της φύσης των δεδομένων που περιλαμβάνει δεδομένα χωρισμένα σε κατηγορίες αλλά και δεδομένα που οι τιμές είναι αριθμητικές και συνεχείς οι προηγούμενοι αλγόριθμοι αποτυγχάνουν να μας δώσουν αποτελέσματα με υψηλό ποσοστό επιτυχίας.

3.6 Προσπάθεια με χρήση MATLAB και εφαρμογή μεθόδου Γραμμικής Παρεμβολής:

Επεξήγηση των Γραμμικής Παρεμβολής

Η ανάλυση παλινδρόμησης είναι μια στατιστική μεθοδολογία που μας επιτρέπει να προσδιορίσουμε την ισχύ και τη σχέση δύο μεταβλητών. Η παλινδρόμηση δεν περιορίζεται σε δύο μεταβλητές, θα μπορούσαμε να έχουμε 2 ή περισσότερες μεταβλητές που δείχνουν μια σχέση μεταξύ τους. Τα αποτελέσματα από την παλινδρόμηση βοηθούν στην πρόβλεψη μιας άγνωστης τιμής ανάλογα με τη σχέση με τις προβλεπόμενες μεταβλητές. Για παράδειγμα, το ύψος και το βάρος κάποιου συνήθως έχουν σχέση. Γενικά, οι ψηλότεροι άνθρωποι τείνουν να ζυγίζουν περισσότερο. Θα μπορούσαμε να χρησιμοποιήσουμε την ανάλυση παλινδρόμησης για να βοηθήσουμε στην πρόβλεψη του βάρους ενός ατόμου, δεδομένου του ύψους του. Όταν υπάρχει μία μεμονωμένη μεταβλητή εισόδου, η παλινδρόμηση αναφέρεται ως Απλή Γραμμική Παλινδρόμηση. Χρησιμοποιούμε τη μεμονωμένη μεταβλητή (ανεξάρτητη) για να μοντελοποιήσουμε μια γραμμική σχέση με τη μεταβλητή στόχο (εξαρτημένη). Αυτό το κάνουμε προσαρμόζοντας ένα μοντέλο για να περιγράψουμε τη σχέση. Εάν υπάρχουν περισσότερες από προβλεπόμενες μεταβλητές, η παλινδρόμηση αναφέρεται ως Πολλαπλή Γραμμική Παλινδρόμηση[22].

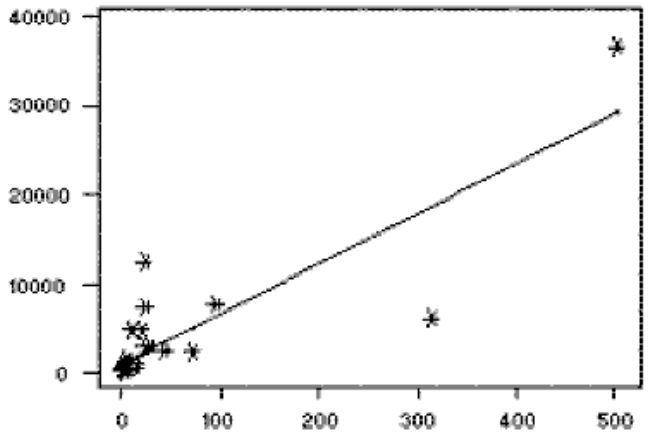
Πριν προσαρμοστεί ένα γραμμικό μοντέλο σε παρατηρούμενα δεδομένα, θα πρέπει πρώτα να καθοριστεί εάν υπάρχει ή όχι σχέση μεταξύ των μεταβλητών ενδιαφέροντος. Αυτό δεν σημαίνει απαραίτητα ότι η μία μεταβλητή προκαλεί την άλλη, αλλά ότι υπάρχει κάποια σημαντική συσχέτιση μεταξύ των δύο μεταβλητών. Ένα διάγραμμα διασποράς (scatterplot) μπορεί να είναι ένα χρήσιμο εργαλείο για τον προσδιορισμό της σχέσης μεταξύ δύο μεταβλητών. Εάν δεν φαίνεται να υπάρχει συσχέτιση μεταξύ των προτεινόμενων επεξηγηματικών και εξαρτημένων μεταβλητών (δηλαδή, το διάγραμμα διασποράς δεν δεικνύει αυξητικές ή φθίνουσες τάσεις), τότε η προσαρμογή ενός μοντέλου γραμμικής παλινδρόμησης στα δεδομένα πιθανότατα δεν θα παρέχει ένα χρήσιμο μοντέλο.

Ένα πολύτιμο αριθμητικό μέτρο συσχέτισης μεταξύ δύο μεταβλητών είναι ο συντελεστής συσχέτισης, ο οποίος είναι μια τιμή μεταξύ -1 και 1 που υποδεικνύει την ισχύ της συσχέτισης των παρατηρούμενων δεδομένων για τις δύο μεταβλητές. Μια γραμμή γραμμικής παλινδρόμησης έχει μια εξίσωση της μορφής $Y = a + bX$, όπου X είναι η επεξηγηματική μεταβλητή και Y είναι η εξαρτημένη μεταβλητή. Η κλίση της ευθείας είναι b και a είναι η τομή (η τιμή του y όταν $x = 0$).

Η πιο κοινή μέθοδος για την προσαρμογή μιας γραμμής παλινδρόμησης είναι η μέθοδος των ελαχίστων τετραγώνων. Αυτή η μέθοδος υπολογίζει την καλύτερη προσαρμοσμένη γραμμή για τα παρατηρούμενα δεδομένα ελαχιστοποιώντας το άθροισμα των τετραγώνων των κατακόρυφων αποκλίσεων από κάθε σημείο δεδομένων προς τη γραμμή (αν ένα σημείο βρίσκεται ακριβώς στην προσαρμοσμένη γραμμή, τότε η κατακόρυφη απόκλιση είναι 0). Επειδή οι αποκλίσεις πρώτα τετραγωνίζονται και μετά αθροίζονται, δεν υπάρχουν

[22] https://towardsdatascience.com/linear-regression-explained-1b36f97b7572

ακυρώσεις μεταξύ θετικών και αρνητικών τιμών[23]. Παρακάτω στην εικόνα φαίνεται ένα παράδειγμα πως παρεμβάλλεται η ευθεία στα παρατηρούμενα δεδομένα:



Εικόνα 25: Ευθεία Γραμμικής Παρεμβολής σε παρατηρούμενα δεδομένα[24].

Η παρακάτω υλοποίηση προσεγγίζει με μια μέθοδο εκπαίδευσης των δεδομένων και προσθήκης αυτών στην βάση εκπαίδευσης για βελτίωση της ακρίβειας, ενώ χρησιμοποιήθηκε η Γραμμική Παρεμβολή στην οποία έγινε η προσέγγιση των συντελεστών του πολυωνύμου με την μέθοδο ελαχίστων τετραγώνων. Ο αλγόριθμος που κατασκευάστηκε έχει κάποιους διακόπτες λειτουργίας πριν προβεί στην Γραμμική Παλινδρόμηση. Στον παρακάτω πίνακα εξηγούνται αυτές οι λειτουργίες:

Πίνακας 13: «Διακόπτες›› λειτουργίας του αλγορίθμου σε MATLAB

Λειτουργίες	**Συνοπτική Επεξήγηση**
db_sorting	Ενεργοποιεί την ταξινόμηση της βάσης δεδομένων
db_sort_once	Επαναλαμβανόμενη ή μη ταξινόμηση της βάσης
db_ins_freq	Προσθήκη ασθενή στη βάση σύμφωνα με την συχνότητα εμφάνισης των δεδομένων εισόδου
db_rand_start	Επιλογή τυχαίου ασθενή για εκκίνηση
db_redcd_thsd	Ενεργοποίηση μειωμένου κατωφλίου στην νόρμα υπολογισμού.

Στις πρώτες γραμμές του κώδικα (29 – 40) υπάρχουν οι επιλογές του πίνακα με σκοπό την προεπιλογή τους πριν την εκτέλεση του κώδικα καθώς και η επιλογή του μεγέθους της βάσης για την εκπαίδευση του μοντέλου με την μεταβλητή db_size:

```
29. db_sorting='y';       % Enables/disables DB sorting (y/n)
30. db_sort_once='n';     % DB sorting one time only (y/n)
31. max_db_iter=5000;  % Max Iterations for repetitive DB sorting
```

[23] http://www.stat.yale.edu/Courses/1997-98/101/linreg.htm

```
32.
33. db_ins_freq='n';      % Enables insertion in DB according frequencies (y/n)
34. db_rand_start='n';    % Select Random first entry for DB (y/n)
35. db_redcd_thsd='n';    % Enables/disables reduced threshold for norm
36. db_dist_thsd=2.0;     % Max norm distance for DB sorting
37.
38. db_size=21;                  % Number of cases used in model
39. max_db_size=45;              % Max number of cases included in DB
40. extra_cases=5;               % Define extra cases excluded from DB for pure
    prediction
```

Η μεταβλητή extra_cases περιλαμβάνει το πλήθος των ασθενών που στην προς πρόβλεψη μεταβλητή δεν υπάρχει τιμή και θα τα χρησιμοποιήσει και αυτά για να τα προβλέψει χωρίς να συμπεριληφθούν και αυτά στο ποσοστό ακρίβειας μιας και δεν ξέρουμε τις πραγματικές τιμές τους.

Έπειτα στις γραμμές 45 - 62, με τις μεταβλητές cat1_d1, cat1_d2, … cat6_d3 γίνεται ο ορισμός των τιμών που εμφανίζονται μέσα στις στήλες/κατηγορίες. Εξαίρεση στον ορισμό κατηγοριών αποτελεί το Tumor Size το οποίο λόγω του ότι είναι συνεχής, αριθμητική μεταβλητή αποθηκεύεται στο db_data_num.

```
45. cat1_d1='MASS';
46. cat1_d2='NME';
47. cat1_d3='MASS+NME';
48.
49. cat2_d1='IRR';
50. cat2_d2='SMOOTH';
51. cat2_d3='SPIC';
52.
53. cat4_d1=1.00;
54. cat4_d2=2.00;
55. cat4_d3=3.00;
56.
57. cat5_d1='HIGH';
58. cat5_d2='LOW';
59.
60. cat6_d1=1.00;
61. cat6_d2=2.00;
62. cat6_d3=3.00;
```

Στις γραμμές 67 – 68 γίνεται ο ορισμός των κατωφλίων για το Tumor Grade. Η μεταβλητή thsd1_max=1.5 είναι το κατώφλι για τιμές Tumor Grade 0 έως 1 και η μεταβλητή thsd2_max=2.5 είναι για τιμές 2 και μεγαλύτερες του 2.5 αντιστοιχούν για Tumor Grade ίσο με 3.

```
67. thsd1_max=1.5;
68. thsd2_max=2.5;
```

Στην γραμμή 77 με την εντολή [db_data_num, db_data_txt, db_data_raw]=xlsread('dataxl_tg.xls') γίνεται το διάβασμα των δεδομένων από το excel.

Στις γραμμές 79 - 96 γίνεται αρχικοποίηση των μετρητών για τα δεδομένα που είναι κατηγορίες με τις μεταβλητές cat1_d1_c, cat1_d2_c, …, cat6_d3_c.

```
79.   cat1_d1_c=0;
80.   cat1_d2_c=0;
81.   cat1_d3_c=0;
82.
83.   cat2_d1_c=0;
84.   cat2_d2_c=0;
85.   cat2_d3_c=0;
86.
87.   cat4_d1_c=0;
88.   cat4_d2_c=0;
89.   cat4_d3_c=0;
90.
91.   cat5_d1_c=0;
92.   cat5_d2_c=0;
93.
94.   cat6_d1_c=0;
95.   cat6_d2_c=0;
96.   cat6_d3_c=0;
```

Στις γραμμές 98 – 132 γίνεται η καταμέτρηση του πλήθους των τιμών που βρίσκονται στις στήλες. Για παράδειγμα, στην στήλη Morphology η κατηγορία MASS εμφανίζεται 38 φορές, ενώ το NME 20 φορές. Το ίδιο γίνεται για όλες τις στήλες και για όλες τις κατηγορίες.

```
98. for j=1:1:5
99.       for i=2:1:max_db_size+1
100.          if (j==1)
101.             if (strcmp(db_data_raw(i,j),cat1_d1)==1)
102.                cat1_d1_c=cat1_d1_c+1;
103.             elseif (strcmp(db_data_raw(i,j),cat1_d2)==1)
104.                cat1_d2_c=cat1_d2_c+1;
105.             elseif (strcmp(db_data_raw(i,j),cat1_d3)==1)
106.                cat1_d3_c=cat1_d3_c+1;
107.             end;
108.          elseif (j==2)
```

```
109.            if (strcmp(db_data_raw(i,j),cat2_d1)==1)
110.              cat2_d1_c=cat2_d1_c+1;
111.            elseif (strcmp(db_data_raw(i,j),cat2_d2)==1)
112.              cat2_d2_c=cat2_d2_c+1;
113.            elseif (strcmp(db_data_raw(i,j),cat2_d3)==1)
114.              cat2_d3_c=cat2_d3_c+1;
115.            end;
116.          elseif (j==4)
117.            if (db_data_num(i-1,2)==cat4_d1)
118.              cat4_d1_c=cat4_d1_c+1;
119.            elseif (db_data_num(i-1,2)==cat4_d2)
120.              cat4_d2_c=cat4_d2_c+1;
121.            elseif (db_data_num(i-1,2)==cat4_d3)
122.              cat4_d3_c=cat4_d3_c+1;
123.            end;
124.          elseif (j==5)
125.            if (strcmp(db_data_raw(i,j),cat5_d1)==1)
126.              cat5_d1_c=cat5_d1_c+1;
127.            elseif (strcmp(db_data_raw(i,j),cat5_d2)==1)
128.              cat5_d2_c=cat5_d2_c+1;
129.            end;
130.          end;
131.        end;
132.      end;
```

Το ίδιο γίνεται και για την μεταβλητή που πρόκειται να γίνει η πρόβλεψη με τις εντολές στις γραμμές 134 – 142. Αυτό γίνεται σε όλη την βάση δεδομένων, δηλαδή σε όλους τους ασθενής που γνωρίζουμε πλήρως τα δεδομένα τους.

```
134.  for i=1:1:max_db_size+1
135.        if (db_data_num(i,4)==cat6_d1)
136.          cat6_d1_c=cat6_d1_c+1;
137.        elseif (db_data_num(i,4)==cat6_d2)
138.          cat6_d2_c=cat6_d2_c+1;
139.        elseif (db_data_num(i,4)==cat6_d3)
140.          cat6_d3_c=cat6_d3_c+1;
141.        end;
142.      end;
```

Στις γραμμές 147 – 164 γίνεται η αντιστοίχιση της συχνότητας εμφάνισης ανά κατηγορία σε κάθε στήλη με εντολές όπως freq_cat1_d1=cat1_d1_c/db_size/2. Αυτό αφορά μόνο τα δεδομένα εισόδου. Είναι δηλαδή σαν μια αντιστοιχία-χαρτογράφηση την ολικής εμφάνισης των κατηγοριών από όλη την βάση στην βάση που επρόκειτο να χρησιμοποιήσουμε για εκπαίδευση. Το διαιρούμε δια 2 για να έρθει γύρω από το κέντρο του διαστήματος.

```
147.    weight1=cat1_d1_c^2+cat1_d2_c^2+cat1_d3_c^2;
148.    freq_cat1_d1=cat1_d1_c/db_size/2;
149.    freq_cat1_d2=freq_cat1_d1+cat1_d2_c/db_size/2;
150.    freq_cat1_d3=freq_cat1_d2+cat1_d3_c/db_size/2;
151.
152.    weight2=cat2_d1_c^2+cat2_d2_c^2+cat2_d3_c^2;
153.    freq_cat2_d1=cat2_d1_c/db_size/2;
154.    freq_cat2_d2=freq_cat2_d1+cat2_d2_c/db_size/2;
155.    freq_cat2_d3=freq_cat2_d2+cat2_d3_c/db_size/2;
156.
157.    weight4=cat4_d1_c^2+cat4_d2_c^2+cat4_d3_c^2;
158.    freq_cat4_d1=cat4_d1_c/db_size/2;
159.    freq_cat4_d2=freq_cat4_d1+cat4_d2_c/db_size/2;
160.    freq_cat4_d3=freq_cat4_d2+cat4_d3_c/db_size/2;
161.
162.    weight5=cat5_d1_c^2+cat5_d2_c^2;
163.    freq_cat5_d1=cat5_d1_c/db_size/2;
164.    freq_cat5_d2=freq_cat5_d1+cat5_d2_c/db_size/2;
```

Στις γραμμές 166 – 204 αντικαθιστά τις κατηγορίες ανά στήλη με την συχνότητα που βρήκε στις γραμμές 147 – 164 και το αποθηκεύει στον πίνακα data_num.

```
166.  for j=1:1:6
167.      for i=2:1:max_db_size+extra_cases+1
168.        if (j==1)
169.          if (strcmp(db_data_raw(i,j),cat1_d1)==1)
170.            data_num(i-1,j)=freq_cat1_d1;
171.          elseif (strcmp(db_data_raw(i,j),cat1_d2)==1)
172.            data_num(i-1,j)=freq_cat1_d2;
173.          elseif (strcmp(db_data_raw(i,j),cat1_d3)==1)
174.            data_num(i-1,j)=freq_cat1_d3;
175.          end;
176.        elseif (j==2)
177.          if (strcmp(db_data_raw(i,j),cat2_d1)==1)
178.            data_num(i-1,j)=freq_cat2_d1;
179.          elseif (strcmp(db_data_raw(i,j),cat2_d2)==1)
180.            data_num(i-1,j)=freq_cat2_d2;
181.          elseif (strcmp(db_data_raw(i,j),cat2_d3)==1)
182.            data_num(i-1,j)=freq_cat2_d3;
183.          end;
184.        elseif (j==3)
185.          data_num(i-1,j)=db_data_num(i-1,j-2)/max(db_data_num(1:db_size,1));
186.        elseif (j==4)
187.          if (db_data_num(i-1,2)==cat4_d1)
188.            data_num(i-1,j)=freq_cat4_d1;
189.          elseif (db_data_num(i-1,2)==cat4_d2)
```

```
190.              data_num(i-1,j)=freq_cat4_d2;
191.            elseif (db_data_num(i-1,2)==cat4_d3)
192.              data_num(i-1,j)=freq_cat4_d3;
193.            end;
194.          elseif (j==5)
195.            if (strcmp(db_data_raw(i,j),cat5_d1)==1)
196.              data_num(i-1,j)=freq_cat5_d1;
197.            elseif (strcmp(db_data_raw(i,j),cat5_d2)==1)
198.              data_num(i-1,j)=freq_cat5_d2;
199.            end;
200.          elseif (j==6)
201.            data_num(i-1,j)=db_data_num(i-1,4);
202.          end;
203.        end;
204.    end;
```

Στη γραμμή 209 είναι η συνθήκη: if (db_sorting=='y'), που αποτελεί και το πρώτο διακόπτη λειτουργίας του αλγορίθμου. Αν λοιπόν ισχύει η συνθήκη γίνεται η δημιουργία και η αρχικοποίηση σε μηδενικά ενός πίνακα γραμμή db_mark στη γραμμή 211 που έχει μέγεθος όσο όλοι οι ασθενείς μαζί, και αυτοί που έχουν συμπληρωμένα δεδομένα και αυτοί που έχουν ελλιπή δεδομένα στην προς πρόβλεψη μεταβλητή. Ο πίνακας αυτός θα υποδηλώνει αν ο ασθενής αυτός είναι στην βάση της εκπαίδευσης ή όχι. Αν είναι τότε στο αντίστοιχο κελί θα υπάρχει η τιμή 1. Η επόμενη συνθήκη στην γραμμή 213 αφορά το αν εκτός από την ταξινόμηση της βάσης ο χρήστης θέλει να εκκινήσει με κάποιο τυχαίο ασθενή. Αν ισχύει η συνθήκη, όταν αυτός επιλεχθεί με την εντολή rand_entp=floor(rand*max_db_size), ενημερώνεται ο πίνακας db_mark, ενώ σε ένα πίνακα γραμμή με την ονομασία db_member αποθηκεύεται και ποιος ήταν. (Ο db_member δείχνει ουσιαστικά αν είναι μέλος κάποιος ασθενής στη βάση δεδομένων εκπαίδευσης.) Αν δεν ισχύει η συνθήκη, τα στοιχεία στο πίνακα data_num του 1ου ασθενούς αποθηκεύονται σε ένα πίνακα γραμμή data_num_test και το db_mark αλλάζει στο πρώτο κελί του την τιμή από 0 σε 1, καθώς πλέον έχει ήδη επιλεχθεί ο πρώτος ασθενής της βάσης, και ο πίνακας db_member έχει το 1ο ασθενή του excel ως αποθηκευμένο.

Στη γραμμή 226 εμφανίζεται η συνθήκη, που αποτελεί και τον δεύτερο διακόπτη λειτουργίας, if (db_sort_once=='y'). Αν ισχύει η συνθήκη, από στις γραμμές 227 – 233 γίνεται υπολογισμός των συντελεστών με την μέθοδο των ελαχίστων τετραγώνων, επανυπολογίζοντας κάθε φορά, καθώς προσθέτει και νέους ασθενείς από την συνολική βάση. Στη γραμμή 233, υπολογίζεται με χρήση νόρμας η απόσταση των παλιών συντελεστών με τους νέους. Η διαδικασία αυτή ολοκληρώνεται όταν ο μετρητής db_cnt γίνει ίσος με το πλήθος των ασθενών στην βάση εκπαίδευσης.

```
227.  for j=2:1:max_db_size
228.        if (db_cnt<db_size)
229.
     [a_test_old,flag]=lsqr(data_num_test(1:db_cnt,1:5),data_num_test(1:db_cnt,6));
```

```
230.        data_num_test(db_cnt+1,1:6)=data_num(j,1:6);
231.
    [a_test_next,flag]=lsqr(data_num_test(1:db_cnt+1,1:5),data_num_test(1:db_cnt+1
    ,6));
232.        %norm_test=norm(a_test_old-a_test_next);
233.        norm_test=abs(norm(a_test_old)-norm(a_test_next));
```

Στη γραμμή 236 εμφανίζεται η τρίτη λειτουργία του αλγορίθμου η οποία αν ισχύει η συνθήκη if (db_ins_freq=='y') τότε ο αλγόριθμος δεν εξετάζει μόνο την νόρμα απόστασης για τους συντελεστές του πολυωνύμου αλλά και το πόσοι έχουν εισαχθεί στην βάση για να για να ελέγξει αν ισχύουν τα κριτήρια πυκνότητας. Ο έλεγχος γίνεται από την στήλη της μεταβλητής που θα γίνει η πρόβλεψη. Στις γραμμές 249 – 286 γίνεται έλεγχος αν έχει γίνει overfitting σε κάθε κατηγορία. Στην περίπτωση που το (db_ins_freq=='n') τότε βάζει τον συγκεκριμένο ασθενή μόνο μέσα από τον έλεγχο της νόρμας. Αξίζει να σημειωθεί ότι στην γραμμή 280 η συνθήκη if ((norm_test<=db_dist_thsd) && (db_mark(j)==0) && (ins_db_flag==1)) γίνεται έλεγχος της νόρμας απόστασης σε σχέση με ένα μεταβλητό κατώφλι που επανυπολογίζεται κάθε φορά προστίθεται ένας ασθενής στη βάση και παράλληλα ελέγχεται αν έχει ήδη συμπεριληφθεί ο ασθενής καθώς και έλεγχος στην συχνότητα εμφάνισης.

```
236.  if (db_ins_freq=='y')
237.        db_cat6_d1_c=0;
238.        db_cat6_d2_c=0;
239.        db_cat6_d3_c=0;
240.        for kk=1:1:length(db_member)
241.          if (data_num(db_member(kk),6)==cat6_d1)
242.            db_cat6_d1_c=db_cat6_d1_c+1;
243.          elseif (data_num(db_member(kk),6)==cat6_d2)
244.            db_cat6_d2_c=db_cat6_d2_c+1;
245.          elseif (data_num(db_member(kk),6)==cat6_d3)
246.            db_cat6_d3_c=db_cat6_d3_c+1;
247.          end;
248.        end;
249.        if (data_num(j,6)==cat6_d1)
250.          if (db_cat6_d1_c<=floor(cat6_d1_c*db_size/max_db_size))
251.            ins_db_flag=1;
252.          else
253.            ins_db_flag=0;
254.          end;
255.        elseif (data_num(j,6)==cat6_d2)
256.          if (db_cat6_d2_c<=floor(cat6_d2_c*db_size/max_db_size))
257.            ins_db_flag=1;
258.          else
259.            ins_db_flag=0;
260.          end;
```

```
261.          elseif (data_num(j,6)==cat6_d3)
262.            if (db_cat6_d3_c<=floor(cat6_d3_c*db_size/max_db_size))
263.              ins_db_flag=1;
264.            else
265.              ins_db_flag=0;
266.            end;
267.          end;
268.          else
269.            ins_db_flag=1;
270.          end;
271.
272.          if (db_redcd_thsd=='y')
273.            if ((norm_test<=db_dist_thsd/(db_size+1-length(db_member))) &&
    (db_mark(j)==0) && (ins_db_flag==1))
274.              db_cnt=db_cnt+1;
275.              db_mark(j)=1;
276.              db_member(db_cnt)=j;
277.              data_num_test(db_cnt,1:6)=data_num(j,1:6);
278.            end;
279.          else
280.            if ((norm_test<=db_dist_thsd) && (db_mark(j)==0) &&
    (ins_db_flag==1))
281.              db_cnt=db_cnt+1;
282.              db_mark(j)=1;
283.              db_member(db_cnt)=j;
284.              data_num_test(db_cnt,1:6)=data_num(j,1:6);
285.            end;
286.          end;
```

Στην γραμμή 292 – 365 αφορά την επαναληπτική ταξινόμηση της βάσης (iterated sorting). Δηλαδή ποιο συγκεκριμένα αν η συνθήκη της γραμμής 226 if (db_sort_once=='n'). Στις γραμμές 293-297 γίνεται έλεγχος αν έγινε εκκίνηση με κάποιο τυχαίο ασθενή ή με τον πρώτο. Στην 299 η συνθήκη while ((db_cnt<db_size) && (k<max_db_iter)) ελέγχεται όσο δεν έχει ξεπεραστεί το μέγιστο πλήθος db_size (βάση εκπαίδευσης) και όσο δεν έχει υπερβεί ο αλγόριθμος το με΄γιστο πλήθος επαναλήψεων που ορίστηκε στη γραμμή 31 με την μεταβλητή max_db_iter. Ουσιαστικά ο αλγόριθμος μέχρι και την γραμμή 365 δεν κάνει κάτι διαφορετικό από ότι γινόταν στις γραμμές 226 – 290.

```
292.    if (db_rand_start=='y')
293.      k=rand_entp;
294.    else
295.      k=0;
296.    end;
297.    reent=0;
298.    while ((db_cnt<db_size) && (k<max_db_iter))
```

```
299.        k=k+1;
300.        max_iter=k+1;
301.        j=mod(k,max_db_size)+1;
302.        if (db_cnt<db_size)
303.
    [a_test_old,flag]=lsqr(data_num_test(1:db_cnt,1:5),data_num_test(1:db_cnt,6));
304.          data_num_test(db_cnt+1,1:6)=data_num(j,1:6);
305.
    [a_test_next,flag]=lsqr(data_num_test(1:db_cnt+1,1:5),data_num_test(1:db_cnt+1
    ,6));
306.          %norm_test=norm(a_test_old-a_test_next);
307.          norm_test=abs(norm(a_test_old)-norm(a_test_next));
308.
309.          % Insertion in DB  according frequencies
310.          if (db_ins_freq=='y')
311.          db_cat6_d1_c=0;
312.          db_cat6_d2_c=0;
313.          db_cat6_d3_c=0;
314.          for kk=1:1:length(db_member)
315.            if (data_num(db_member(kk),6)==cat6_d1)
316.              db_cat6_d1_c=db_cat6_d1_c+1;
317.            elseif (data_num(db_member(kk),6)==cat6_d2)
318.              db_cat6_d2_c=db_cat6_d2_c+1;
319.            elseif (data_num(db_member(kk),6)==cat6_d3)
320.              db_cat6_d3_c=db_cat6_d3_c+1;
321.            end;
322.          end;
323.          if (data_num(j,6)==cat6_d1)
324.            if (db_cat6_d1_c<=floor(cat6_d1_c*db_size/max_db_size))
325.              ins_db_flag=1;
326.            else
327.              ins_db_flag=0;
328.            end;
329.          elseif (data_num(j,6)==cat6_d2)
330.            if (db_cat6_d2_c<=floor(cat6_d2_c*db_size/max_db_size))
331.              ins_db_flag=1;
332.            else
333.              ins_db_flag=0;
334.            end;
335.          elseif (data_num(j,6)==cat6_d3)
336.            if (db_cat6_d3_c<=floor(cat6_d3_c*db_size/max_db_size))
337.              ins_db_flag=1;
338.            else
339.              ins_db_flag=0;
340.            end;
341.          end;
```

```
342.          else
343.            ins_db_flag=1;
344.          end;
345.
346.          if (db_redcd_thsd=='y')
347.            if ((norm_test<=db_dist_thsd/(db_size+1-length(db_member))) &&
    (db_mark(j)==0) && (ins_db_flag==1))
348.              db_cnt=db_cnt+1;
349.              db_mark(j)=1;
350.              db_member(db_cnt)=j;
351.              data_num_test(db_cnt,1:6)=data_num(j,1:6);
352.            end;
353.          else
354.            if ((norm_test<=db_dist_thsd) && (db_mark(j)==0) &&
    (ins_db_flag==1))
355.              db_cnt=db_cnt+1;
356.              db_mark(j)=1;
357.              db_member(db_cnt)=j;
358.              data_num_test(db_cnt,1:6)=data_num(j,1:6);
359.            end;
360.          end;
361.
362.        end;
363.      end;
364.  end;
```

Οπότε καταλήγει στις γραμμές 367 – 372 όπου αν η μεταβλητή db_cnt, που ουσιαστικά έχει προσμετρήσει το πλήθος των ασθενών που τελικά μπήκαν στην βάση εκπαίδευση, είναι μικρότερη από το db_size που όρισε ο χρήστης στην γραμμή 38, τότε αυτό σημαίνει ότι το κατώφλι db_dist_thsd, που επέλεξε ο χρήστης είναι πολύ αυστηρό και άρα δεν συγκεντρώνεται ο προκαθορισμένος αριθμός ασθενών για την βάση εκπαίδευσης. Άρα ουσιαστικά το κατώφλι που ορίζεται, έχει σκοπό την ευρεία ποικιλία διαφορετικών συνδυασμών δεδομένων από την συνολική βάση. Το ανώτατο πιο χαλαρό όριο από παρατήρηση είναι το 4.5. Στην περίπτωση που θέλουμε να μειωθεί το κατώφλι αυτό, μέσα σε ανεκτά όρια, αυτό που ουσιαστικά γίνεται είναι να δίνεται στο αλγόριθμο για εκπαίδευση ασθενείς από την ίδια κατηγορία Tumor Grade και πιθανά αυτό να οδηγήσει σε μία όχι και τόσο καλή ακρίβεια στην πρόβλεψη.

```
367.  if (db_cnt<db_size)
368.      disp(' Not enough DB members after sorting!');
369.      disp(' Relax db_dist_thsd!');
370.      disp(db_cnt);
371.      return;
372.    end;
```

Από την γραμμή 399 – 471 γίνεται υπολογισμός του παράγοντα κλιμάκωσης (scaling factor) για τα δεδομένα. Συγκεκριμένα, στις γραμμές 404 – 409 γίνεται πρώτα κλιμάκωσης στα δεδομένα της βάσης εκπαίδευσης. Από τις γραμμές 410 – 471 γίνεται η κλιμάκωση σε όλη την υπόλοιπη βάση. Πιο συγκεκριμένα, στις γραμμές 419 – 425 γίνεται υπολογισμός των συντελεστών A του πολυωνύμου για τους ασθενείς έξω από την βάση εκπαίδευσης και αποθηκεύονται, υποθετικά είτε αυτοί ανήκουν στην κατηγορία Tumor Grade ίσο 1, είτε ίσο με 2, είτε ίσο με 3. Στις γραμμές 423 – 425 γίνεται ο υπολογισμός των αποστάσεων του προβλεπόμενου από το αντίστοιχο πραγματικό της βάσης. Στην γραμμή 426 - 465 γίνεται έλεγχος ποιος είναι ο ελαχίστη νόρμα που υπολογίστηκαν πριν και αποθηκεύτηκαν στα tt(1), tt(2), tt(3). Στην περίπτωση που το tt(1) είναι το μικρότερο τότε αποθηκεύει τα tt(2) και tt(3) στα xtt(1) και xtt(2) αντίστοιχα για να βρεθεί ποιος είναι ο αμέσος μικρότερος από τα tt(2) και tt(3). Ο αμέσος επόμενος ελάχιστος αποθηκεύεται με την εντολή στην γραμμή 442 ([v,ind]=min(xtt)). Στην συνθήκη της γραμμής 449 ((v>tt(2)+dist_thsd*max_db_size/i)) γίνεται ο έλεγχος αν ο δεύτερος μικρότερος απέχει ένα ικανοποιητικό κατώφλι από τον πρώτο το οποίο υπολογίζεται από την νόρμα tt(2) προσθέτοντας τον παράγοντα dist_thsd*max_db_size/i. Το dist_thsd = 0.025 και έχει βγεί με παρατήρηση και δοκιμή και αποτυχία. Το max_db_size, είναι το πλήθος όλης της βάσης των ασθενών χωρίς τους ασθενείς που έχουν ελλιπή δεδομένα, και το i είναι ο δείκτης από το for loop που τρέχει από το 1 σε όλη τη βάση (συμπεριλαμβανομένου και αυτών που έχουν ελλιπή δεδομένα. Αν ισχύει η συνθήκη της γραμμής 449 τότε δίνεται scaling factor (παράγοντας κλιμάκωσης) ίσο με 1 καθώς αφορά περίπτωση ασθενή με Tumor Grade ίσο με 1. Αλλιώς θα δωθεί scaling factor ίσο με 2 ή ίσο με 3. Εφόσον, δεν ισχύει η συνθήκη της γραμμής 426 τότε στις γραμμές 439 και 452, εκτελείται ο κώδικας αν το tt(2) είναι η μικρότερη νόρμα ή το tt(3) είναι η μικρότερη νόρμα αντίστοιχα και εκτελούνται οι αντίστοιχες διαδικασίες όπως τις γραμμές 429 – 438.

```
399.  temp_num1(1:db_size,1)=data_num(1:db_size,6);
400.    temp_num2(1:db_size,1)=data_num(1:db_size,6);
401.    temp_num3(1:db_size,1)=data_num(1:db_size,6);
402.
403.    dist_thsd=0.0025;
404.    for i=1:1:max_db_size+extra_cases
405.      if (i<=db_size)
406.        scalef(i)=data_num(i,6);
407.        for j=1:1:5
408.          data_num(i,j)=data_num(i,j)*scalef(i);
409.        end;
410.      else
411.        temp_num1(i,1)=1;
412.        temp_num2(i,1)=2;
413.        temp_num3(i,1)=3;
414.        temp_data_num1(i,1:5)=data_num(i,1:5);
415.        temp_data_num2(1:i-1,1:5)=data_num(1:i-1,1:5);
416.        temp_data_num2(i,1:5)=2*data_num(i,1:5);
```

```
417.          temp_data_num3(1:i-1,1:5)=data_num(1:i-1,1:5);
418.          temp_data_num3(i,1:5)=3*data_num(i,1:5);
419.          [a_val,flag]=lsqr(data_num(1:db_size,1:5),scalef(1:db_size)');
420.          [a1,flag]=lsqr(temp_data_num1(1:i,1:5),temp_num1);
421.          [a2,flag]=lsqr(temp_data_num2(1:i,1:5),temp_num2);
422.          [a3,flag]=lsqr(temp_data_num3(1:i,1:5),temp_num3);
423.          tt(1)=norm(a1-a_val);
424.          tt(2)=norm(a2-a_val);
425.          tt(3)=norm(a3-a_val);
426.          if (min(tt)==tt(1))
427.            xtt(1)=tt(2);
428.            xtt(2)=tt(3);
429.            [v,ind]=min(xtt);
430.            if (v>tt(1)+dist_thsd*max_db_size/i)
431.              scalef(i)=1;
432.            else
433.              if (ind==1)
434.                scalef(i)=2;
435.              else
436.                scalef(i)=3;
437.              end;
438.            end;
439.          elseif (min(tt)==tt(2))
440.            xtt(1)=tt(1);
441.            xtt(2)=tt(3);
442.            [v,ind]=min(xtt);
443.            if (v>tt(2)+dist_thsd*max_db_size/i)
444.              scalef(i)=2;
445.            else
446.              if (ind==1)
447.                scalef(i)=1;
448.              else
449.                scalef(i)=3;
450.              end;
451.            end;
452.          else
453.            xtt(1)=tt(1);
454.            xtt(2)=tt(2);
455.            [v,ind]=min(xtt);
456.            if (v>tt(3)+dist_thsd*max_db_size/i)
457.              scalef(i)=3;
458.            else
459.              if (ind==1)
460.                scalef(i)=1;
461.              else
462.                scalef(i)=2;
```

```
463.              end;
464.            end;
465.          end;
466.          a=a_val;
467.          for j=1:1:5
468.            data_num(i,j)=data_num(i,j)*scalef(i);
469.          end;
470.        end;
471.      end;
```

Στις γραμμές 476 – 485 εκτελείται η γραμμική παρεμβολή.

```
476.    for i=db_size+1:1:max_db_size+extra_cases
477.        est_val(i-db_size)=a'*data_num(i,1:5)';
478.        if (est_val(i-db_size)<thsd1_max)
479.          est_tg(i-db_size)=1;
480.        elseif ((est_val(i-db_size)>=thsd1_max) && (est_val(i-
    db_size)<thsd2_max))
481.          est_tg(i-db_size)=2;
482.        else
483.          est_tg(i-db_size)=3;
484.        end;
485.      end;
```

Στις γραμμές 487 – 493 αποθηκεύονται στον πίνακα diff_tg η διαφορά του εκτιμώμενου από αυτό που έγινε πρόβλεψη. Αν η διαφορά είναι 0 τότε πετύχαμε ακριβώς το αποτέλεσμα. Αν όχι, και τα αποτελέσματα υπολογίζεται και αποθηκεύεται η διαφορά.

```
487.  for i=db_size+1:1:max_db_size+extra_cases
488.        if (i<=max_db_size)
489.          diff_tg(i-db_size)=est_tg(i-db_size)-data_num(i,6);
490.        else
491.          diff_tg(i-db_size)=0;
492.        end;
493.      end;
```

Στις γραμμές 495 – 503 γίνεται η καταμέτρηση των αποτελεσμάτων που έγινε επιτυχής η πρόβλεψη τους και αυτών που απέτυχαν.

```
495.  pred_succ=0;
496.     pred_fail=0;
497.     for i=1:1:max_db_size-db_size
498.        if (diff_tg(i)==0)
499.          pred_succ=pred_succ+1;
500.        else
```

```
501.          pred_fail=pred_fail+1;
502.        end;
503.     end;
```

Από τις γραμμές 520 – 580 γίνεται εκτύπωση των αποτελεσμάτων με τις εκτιμώμενες και πραγματικές τιμές καθώς και την διαφορά εκτίμησης και πραγματικής τιμής, των προεπιλογών του χρήστη, του πλήθους των επαναλήψεων που εκτελέστηκαν συνολικά στην ταξινόμηση, του ποσοστού επιτυχίας, του πλήθος των ασθενών που ήταν γνωστά και πλήρη τα αποτελέσματά τους, ποιος ήταν ο πρώτος ασθενής που καταχωρήθηκε στην βάση εκπαίδευση (αν δηλαδή ξεκίνησε ο αλγόριθμος από τον πρώτο ή από κάποιον τυχαίο), καθώς και τα αποτελέσματα που βρέθηκαν στους ασθενής που είχαν ελλείψεις στην προς πρόβλεψη μεταβλητή (extra cases).

Τα αποτελέσματα είναι ικανοποιητικά για τους 4 από τους 5 δείκτες (Tumor Grade, ER, PR, CERB-2) στους οποίους χρησιμοποιήθηκε ακριβώς ο ίδιος αλγόριθμος. Όσον αφορά το Ki-67 έγινε μια προσπάθεια με παρόμοιο αλγόριθμο και με χρήση της polyfit, αλλά τα αποτελέσματα είχαν πολύ χαμηλά ποσοστά επιτυχίας. Παρακάτω παρατίθεται ο κώδικας για το Ki-67 καθώς και τα αποτελέσματα του:

```
1.  %-----------------------------------------------------------------
2.  %--                         Data Processing Environment
3.  %-----------------------------------------------------------------
4.  % Name              :       Antonios Ntib
5.  % Version/Com       :       v1.0  --> Initial Version
6.  %                           v1.1  --> Characterization based on frequenices
7.  %                           v1.2  --> Calibration of data_num values
8.  %                           v1.3  --> Remove pivoting
9.  %                           v1.4  --> Different algorithms for scalef
    calculations
10. %                           v1.9  --> Norm distance algorithm for scalef
11. %                           v1.10 --> Sort DB algorithm
12. %                           v1.11 --> Population in DB according TG
    frequenices
13. %                           v1.12 --> Addition of pure prediction for extra
    samples
14. %                           v1.13 --> Cat5 reduced to Low/High classes
15. % Date              :       21/12/2021
16. %-----------------------------------------------------------------
17.
18. clc;
19. clear all;
20. close all;
21.
22. %-----------------------------------------------------------------
23. %--                         Parameters
```

```
24. %-------------------------------------------------------------------
25.
26. %-------------------------------------------------------------------
27. % DB Sorting Parameters
28.
29. db_size=30;                    % Number of cases used in model
30. max_db_size=44;                % Max number of cases included in DB
31. extra_cases=6;                 % Define extra cases excluded from DB for pure
    predictioin
32.
33. %-------------------------------------------------------------------
34. % Definition of Classification Zones
35.
36. poly_ord=10;
37. thsd_max=5;
38.
39. %-------------------------------------------------------------------
40. % Definition of Categories
41.
42. cat1_d1='MASS';
43. cat1_d2='NME';
44. cat1_d3='MASS+NME';
45.
46. cat2_d1='IRR';
47. cat2_d2='SMOOTH';
48. cat2_d3='SPIC';
49.
50. cat4_d1=1.00;
51. cat4_d2=2.00;
52. cat4_d3=3.00;
53.
54. cat5_d1='HIGH';
55. cat5_d2='LOW';
56.
57. %-------------------------------------------------------------------
58. %--                                Model Code
59. %-------------------------------------------------------------------
60.
61. %-------------------------------------------------------------------
62. % Read data from xls file and frequencies for each characterization
63.
64.    [db_data_num, db_data_txt, db_data_raw]=xlsread('dataxl_ki_67.xls');
65.
66.    cat1_d1_c=0;
67.    cat1_d2_c=0;
68.    cat1_d3_c=0;
```

```
69.
70.   cat2_d1_c=0;
71.   cat2_d2_c=0;
72.   cat2_d3_c=0;
73.
74.   cat4_d1_c=0;
75.   cat4_d2_c=0;
76.   cat4_d3_c=0;
77.
78.   cat5_d1_c=0;
79.   cat5_d2_c=0;
80.
81.   for j=1:1:5
82.     for i=2:1:max_db_size+1
83.       if (j==1)
84.         if (strcmp(db_data_raw(i,j),cat1_d1)==1)
85.           cat1_d1_c=cat1_d1_c+1;
86.         elseif (strcmp(db_data_raw(i,j),cat1_d2)==1)
87.           cat1_d2_c=cat1_d2_c+1;
88.         elseif (strcmp(db_data_raw(i,j),cat1_d3)==1)
89.           cat1_d3_c=cat1_d3_c+1;
90.         end;
91.       elseif (j==2)
92.         if (strcmp(db_data_raw(i,j),cat2_d1)==1)
93.           cat2_d1_c=cat2_d1_c+1;
94.         elseif (strcmp(db_data_raw(i,j),cat2_d2)==1)
95.           cat2_d2_c=cat2_d2_c+1;
96.         elseif (strcmp(db_data_raw(i,j),cat2_d3)==1)
97.           cat2_d3_c=cat2_d3_c+1;
98.         end;
99.       elseif (j==4)
100.          if (db_data_num(i-1,2)==cat4_d1)
101.            cat4_d1_c=cat4_d1_c+1;
102.          elseif (db_data_num(i-1,2)==cat4_d2)
103.            cat4_d2_c=cat4_d2_c+1;
104.          elseif (db_data_num(i-1,2)==cat4_d3)
105.            cat4_d3_c=cat4_d3_c+1;
106.          end;
107.        elseif (j==5)
108.          if (strcmp(db_data_raw(i,j),cat5_d1)==1)
109.            cat5_d1_c=cat5_d1_c+1;
110.          elseif (strcmp(db_data_raw(i,j),cat5_d2)==1)
111.            cat5_d2_c=cat5_d2_c+1;
112.          end;
113.        end;
114.      end;
```

```
115.    end;
116.
117.  %-----------------------------------------------------------------
118.  % Map numerical values on characterization
119.
120.    weight1=cat1_d1_c^2+cat1_d2_c^2+cat1_d3_c^2;
121.    freq_cat1_d1=cat1_d1_c/db_size/2;
122.    freq_cat1_d2=freq_cat1_d1+cat1_d2_c/db_size/2;
123.    freq_cat1_d3=freq_cat1_d2+cat1_d3_c/db_size/2;
124.
125.    weight2=cat2_d1_c^2+cat2_d2_c^2+cat2_d3_c^2;
126.    freq_cat2_d1=cat2_d1_c/db_size/2;
127.    freq_cat2_d2=freq_cat2_d1+cat2_d2_c/db_size/2;
128.    freq_cat2_d3=freq_cat2_d2+cat2_d3_c/db_size/2;
129.
130.    weight4=cat4_d1_c^2+cat4_d2_c^2+cat4_d3_c^2;
131.    freq_cat4_d1=cat4_d1_c/db_size/2;
132.    freq_cat4_d2=freq_cat4_d1+cat4_d2_c/db_size/2;
133.    freq_cat4_d3=freq_cat4_d2+cat4_d3_c/db_size/2;
134.
135.    weight5=cat5_d1_c^2+cat5_d2_c^2;
136.    freq_cat5_d1=cat5_d1_c/db_size/2;
137.    freq_cat5_d2=freq_cat5_d1+cat5_d2_c/db_size/2;
138.
139.    for j=1:1:6
140.      for i=2:1:max_db_size+extra_cases+1
141.        if (j==1)
142.          if (strcmp(db_data_raw(i,j),cat1_d1)==1)
143.            data_num(i-1,j)=freq_cat1_d1;
144.          elseif (strcmp(db_data_raw(i,j),cat1_d2)==1)
145.            data_num(i-1,j)=freq_cat1_d2;
146.          elseif (strcmp(db_data_raw(i,j),cat1_d3)==1)
147.            data_num(i-1,j)=freq_cat1_d3;
148.          end;
149.        elseif (j==2)
150.          if (strcmp(db_data_raw(i,j),cat2_d1)==1)
151.            data_num(i-1,j)=freq_cat2_d1;
152.          elseif (strcmp(db_data_raw(i,j),cat2_d2)==1)
153.            data_num(i-1,j)=freq_cat2_d2;
154.          elseif (strcmp(db_data_raw(i,j),cat2_d3)==1)
155.            data_num(i-1,j)=freq_cat2_d3;
156.          end;
157.        elseif (j==3)
158.          data_num(i-1,j)=db_data_num(i-1,j-2)/max(db_data_num(1:db_size,1));
159.        elseif (j==4)
160.          if (db_data_num(i-1,2)==cat4_d1)
```

```
161.              data_num(i-1,j)=freq_cat4_d1;
162.            elseif (db_data_num(i-1,2)==cat4_d2)
163.              data_num(i-1,j)=freq_cat4_d2;
164.            elseif (db_data_num(i-1,2)==cat4_d3)
165.              data_num(i-1,j)=freq_cat4_d3;
166.            end;
167.          elseif (j==5)
168.            if (strcmp(db_data_raw(i,j),cat5_d1)==1)
169.              data_num(i-1,j)=freq_cat5_d1;
170.            elseif (strcmp(db_data_raw(i,j),cat5_d2)==1)
171.              data_num(i-1,j)=freq_cat5_d2;
172.            end;
173.          elseif (j==6)
174.            data_num(i-1,j)=db_data_num(i-1,4);
175.          end;
176.        end;
177.      end;
178.
179.  %-----------------------------------------------------------------
180.  % Interploation using Polyfit
181.
182.  %    for i=1:1:max_db_size+extra_cases
183.  %      x1=data_num(i,1);
184.  %      x2=data_num(i,2);
185.  %      x3=data_num(i,3);
186.  %      x4=data_num(i,4);
187.  %      x5=data_num(i,5);
188.  %      x_val(i)=(x1*x5)/(x2*x4)+x3;
189.  %      if (i<=max_db_size)
190.  %        y_val(i)=data_num(i,6);
191.  %      else
192.  %        y_val(i)=0;
193.  %      end;
194.  %      x_val(i)=1+2*y_val(i)+x_val(i);
195.  %    end;
196.  %    a=polyfit(x_val(1:db_size),y_val(1:db_size),poly_ord);
197.  %    t1(:,1)=x_val';
198.  %    t1(:,2)=y_val';
199.
200.  %    for i=1:1:max_db_size+extra_cases
201.  %      x1=data_num(i,1);
202.  %      x2=data_num(i,2);
203.  %      x3=data_num(i,3);
204.  %      x4=data_num(i,4);
205.  %      x5=data_num(i,5);
206.  %      if (i<=db_size)
```

```
207. %         x_val(i)=(x1*x5)/(x2*x4)+x3;
208. %         y_val(i)=data_num(i,6);
209. %         x_val(i)=1+2*y_val(i)+x_val(i);
210. %       else
211. %         temp_x(1:db_size)=x_val(1:db_size);
212. %         temp_y(1:db_size)=y_val(1:db_size);
213. %         a_val=polyfit(x_val(1:db_size),y_val(1:db_size),poly_ord);
214. %         for k=1:1:100
215. %           clc;
216. %           x_val(i)=(x1*x5)/(x2*x4)+x3;
217. %           temp_x(db_size+1)=1+2*k+x_val(i);
218. %           temp_y(db_size+1)=k;
219. %
    a_temp=polyfit(temp_x(1:db_size+1),temp_y(1:db_size+1),poly_ord);
220. %           norm_a(k)=norm(a_val-a_temp);
221. %         end;
222. %         [v,ind]=min(norm_a);
223. %         x_val(i)=(x1*x5)/(x2*x4)+x3;
224. %         x_val(i)=1+2*ind+x_val(i);
225. %         y_val(i)=ind;
226. %       end;
227. %     end;
228. %     a=polyfit(x_val(1:db_size),y_val(1:db_size),poly_ord);
229. %     t1(:,1)=x_val';
230. %     t1(:,2)=y_val';
231.
232.     for i=1:1:max_db_size+extra_cases
233.       x1=data_num(i,1);
234.       x2=data_num(i,2);
235.       x3=data_num(i,3);
236.       x4=data_num(i,4);
237.       x5=data_num(i,5);
238.       x_val(i)=(x1*x5)/(x2*x4)+x3;
239.       if (i<=db_size)
240.         y_val(i)=data_num(i,6);
241.       else
242.         a_val=polyfit(x_val(1:db_size),y_val(1:db_size),poly_ord);
243.         y_val(i)=0;
244.         for j=1:1:poly_ord+1
245.           y_val(i)=y_val(i)+a_val(j)*x_val(i)^(poly_ord+1-j);
246.         end;
247.         y_val(i)=floor(y_val(i));
248.         if (y_val(i)<0)
249.           y_val(i)=0;
250.         elseif (y_val(i)>100)
251.           y_val(i)=100;
```

```
252.         end;
253.       end;
254.     end;
255.     for i=1:1:max_db_size+extra_cases
256.       x_val(i)=1+2*y_val(i)+x_val(i);
257.     end;
258.     a=polyfit(x_val(1:db_size),y_val(1:db_size),poly_ord);
259.     t1(:,1)=x_val';
260.     t1(:,2)=y_val';
261.     clc;
262.
263.  %-------------------------------------------------------------------
264.  % Prediction using Regression Model for max db cases
265.
266.     k=0;
267.     for i=db_size+1:1:max_db_size+extra_cases
268.       k=k+1;
269.       est_ki(k)=0;
270.       for j=1:1:poly_ord+1
271.         est_ki(k)=est_ki(k)+a(j)*x_val(i)^(poly_ord+1-j);
272.       end;
273.       est_ki(k)=floor(est_ki(k));
274.       if (abs(est_ki(k))>100)
275.         est_ki(k)=100;
276.       end;
277.     end;
278.
279.     model_est(1:max_db_size+extra_cases)=0;
280.     for i=1:1:max_db_size+extra_cases
281.       for j=1:1:poly_ord+1
282.         model_est(i)=model_est(i)+a(j)*x_val(i)^(poly_ord+1-j);
283.       end;
284.       model_est(i)=floor(model_est(i));
285.       if (abs(model_est(i))>100)
286.         model_est(i)=100;
287.       end;
288.     end;
289.
290.     for i=db_size+1:1:max_db_size+extra_cases
291.       if (i<=max_db_size)
292.         diff_ki(i-db_size)=est_ki(i-db_size)-data_num(i,6);
293.         if (abs(diff_ki(i-db_size))<thsd_max)
294.           diff_ki(i-db_size)=0;
295.         end;
296.       else
297.         diff_ki(i-db_size)=0;
```

```
298.       end;
299.     end;
300.
301.     pred_succ=0;
302.     pred_fail=0;
303.     for i=1:1:max_db_size-db_size
304.       if (diff_ki(i)==0)
305.         pred_succ=pred_succ+1;
306.       else
307.         pred_fail=pred_fail+1;
308.       end;
309.     end;
310.
311.       temp(:,1)=db_size+1:1:max_db_size+extra_cases;
312.       temp(:,2)=est_ki(1:1:max_db_size+extra_cases-db_size);
313.       temp(1:1:max_db_size-db_size,3)=data_num(db_size+1:1:max_db_size,6);
314.       temp(max_db_size+1-db_size:1:max_db_size+extra_cases-
    db_size,3)=est_ki(max_db_size+1-db_size:1:max_db_size+extra_cases-db_size);
315.       temp(:,4)=diff_ki(1:1:max_db_size+extra_cases-db_size);
316.
317.  disp('-------------------------------------------------');
318.  disp(' Model Results  ');
319.  disp('-------------------------------------------------');
320.  disp('-------------------------------------------------');
321.  disp(' Cases ysed for Model ');
322.  disp(db_size);
323.  disp(' Total Number of Cases in DB ');
324.  disp(max_db_size);
325.  disp(' Model Success Ratio (%) ');
326.  disp(100*pred_succ/(max_db_size-db_size));
327.  disp('-------------------------------------------------');
328.  disp(' Analytic Estimation on Cases ');
329.  disp(' DB - Est - Ki67 - Diff ');
330.  disp('-------------------------------------------------');
331.  disp(temp(1:1:max_db_size-db_size,:));
332.  disp('-------------------------------------------------');
333.  disp('                          Extra Cases                          ');
334.  disp('-------------------------------------------------');
335.  disp(temp(max_db_size-db_size+1:1:max_db_size-db_size+extra_cases,:));
336.
337.  figure(1)
338.  gcf;
339.  shg;
340.  i=1:1:max_db_size;
341.  plot(i,data_num(i,6),'-*',i,model_est(i),'--o');
342.  grid;
```

```
343.  title(' Ki 67 Prediction Diagram for DB ');
344.  xlabel(' Max DB Cases ');
345.  ylabel(' Ki 67');
346.  legend('Real Ki 67','Est Ki 67');
```

Αποτελέσματα:

Model Results

Cases used for Model

30

Total Number of Cases in DB

44

Model Success Ratio (%)

7.1429

Analytic Estimation on Cases

DB - Est - Ki67 - Diff

---

31 34 40 -6

32 43 4 39

33 35 5 30

34 2 8 -6

35 32 10 22

36 39 15 24

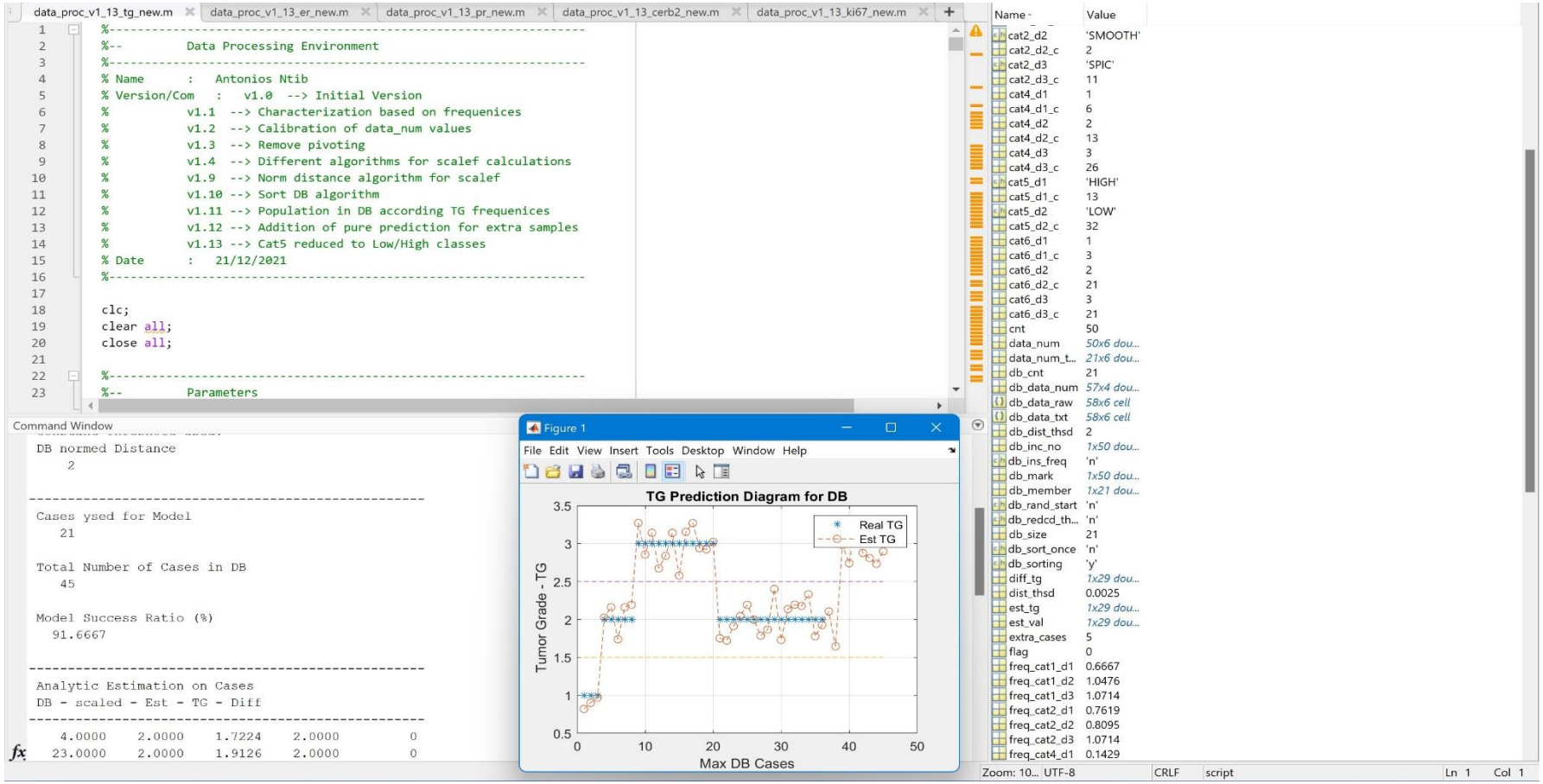
37 33 20 13

38	33	25	8
39	20	40	-20
40	27	30	0
41	43	5	38
42	32	50	-18
43	34	60	-26
44	32	90	-58

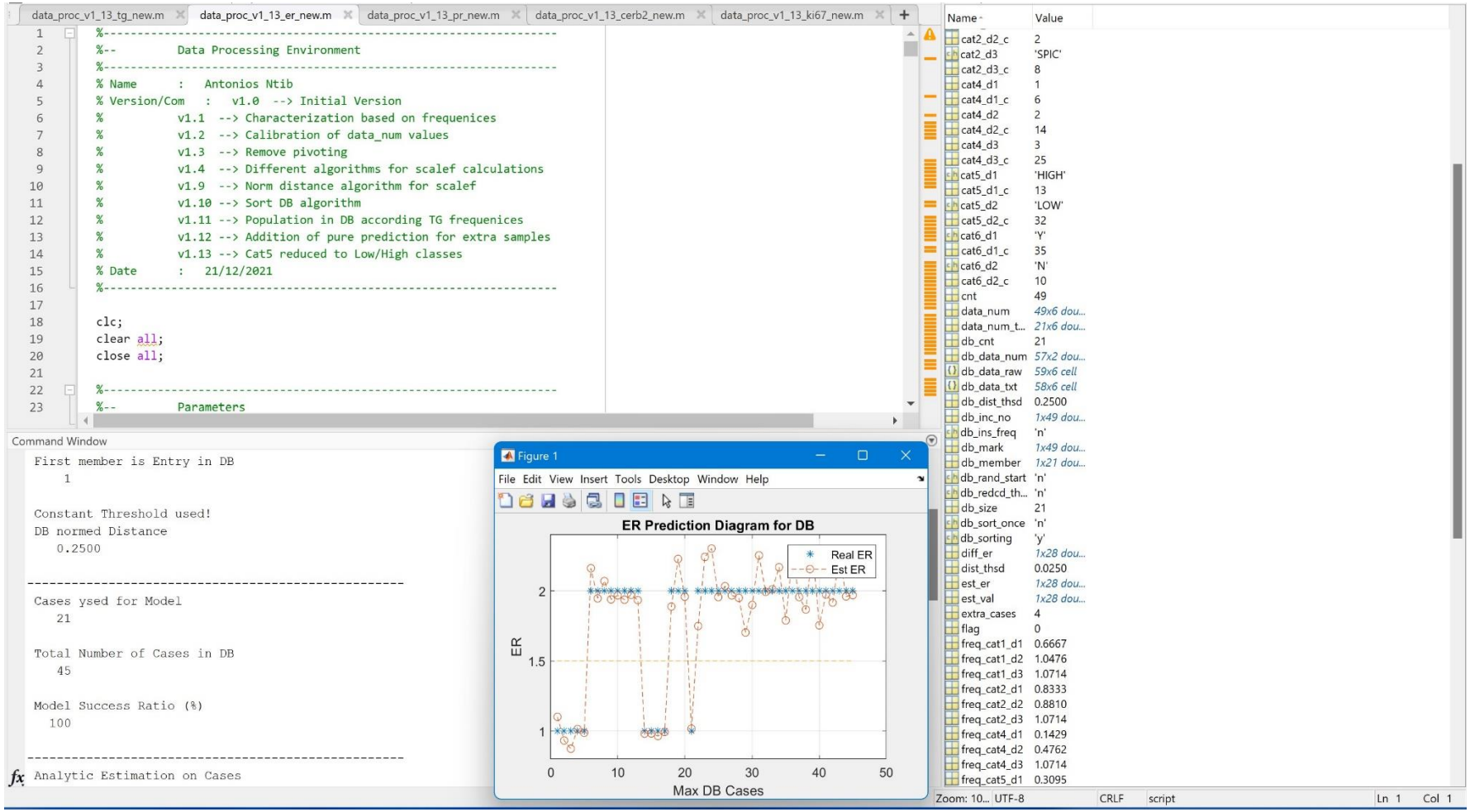
Extra Cases

45	29	29	0
46	21	21	0
47	32	32	0
48	5	5	0
49	35	35	0
50	30	30	0

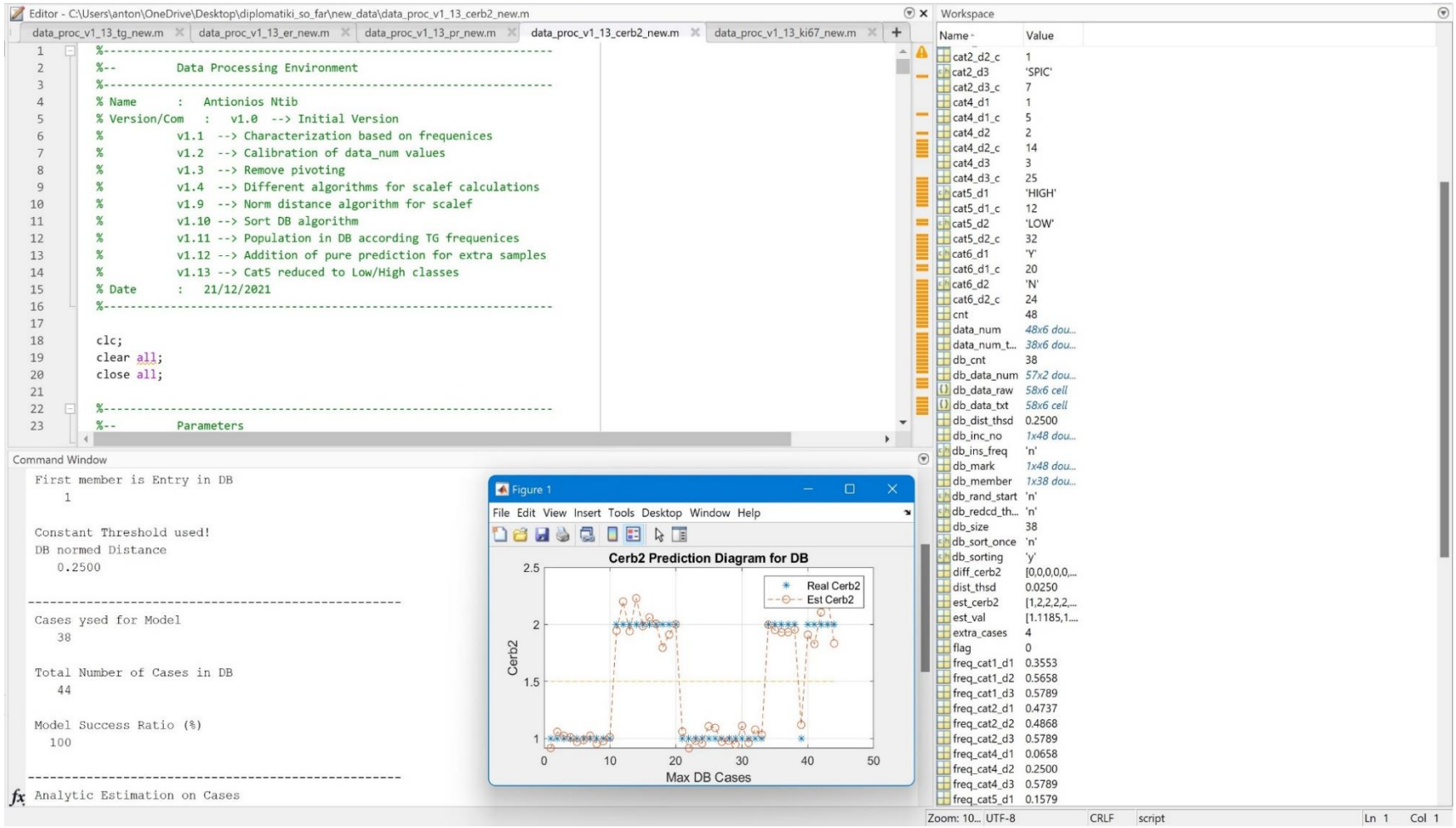
3.7 Στιγμιότυπα από την εκτέλεση των αλγορίθμων



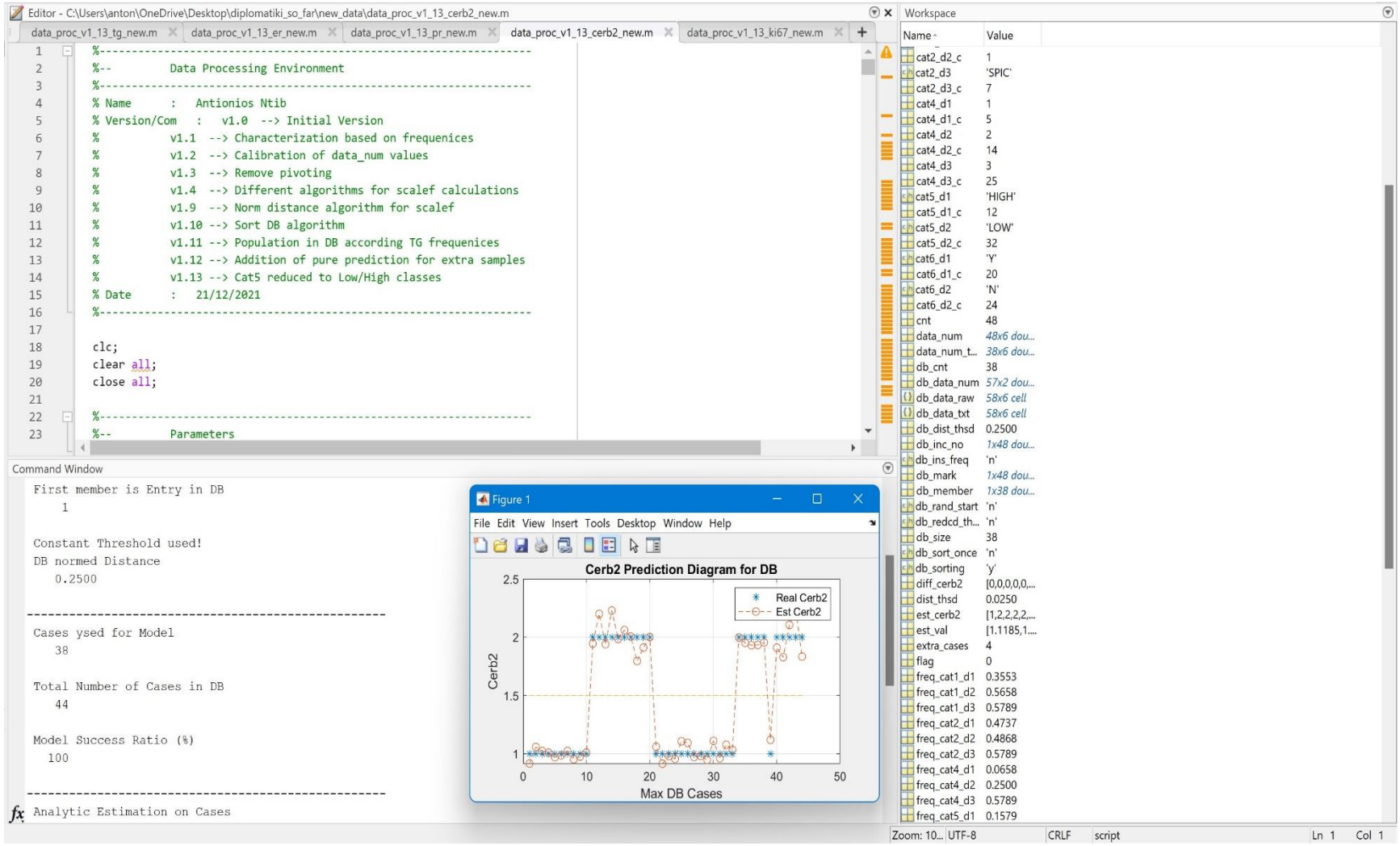
Εικόνα 26: Στιγμιότυπο μετά από την εκτέλεση του κώδικα για τον δείκτη Tumor Grade.



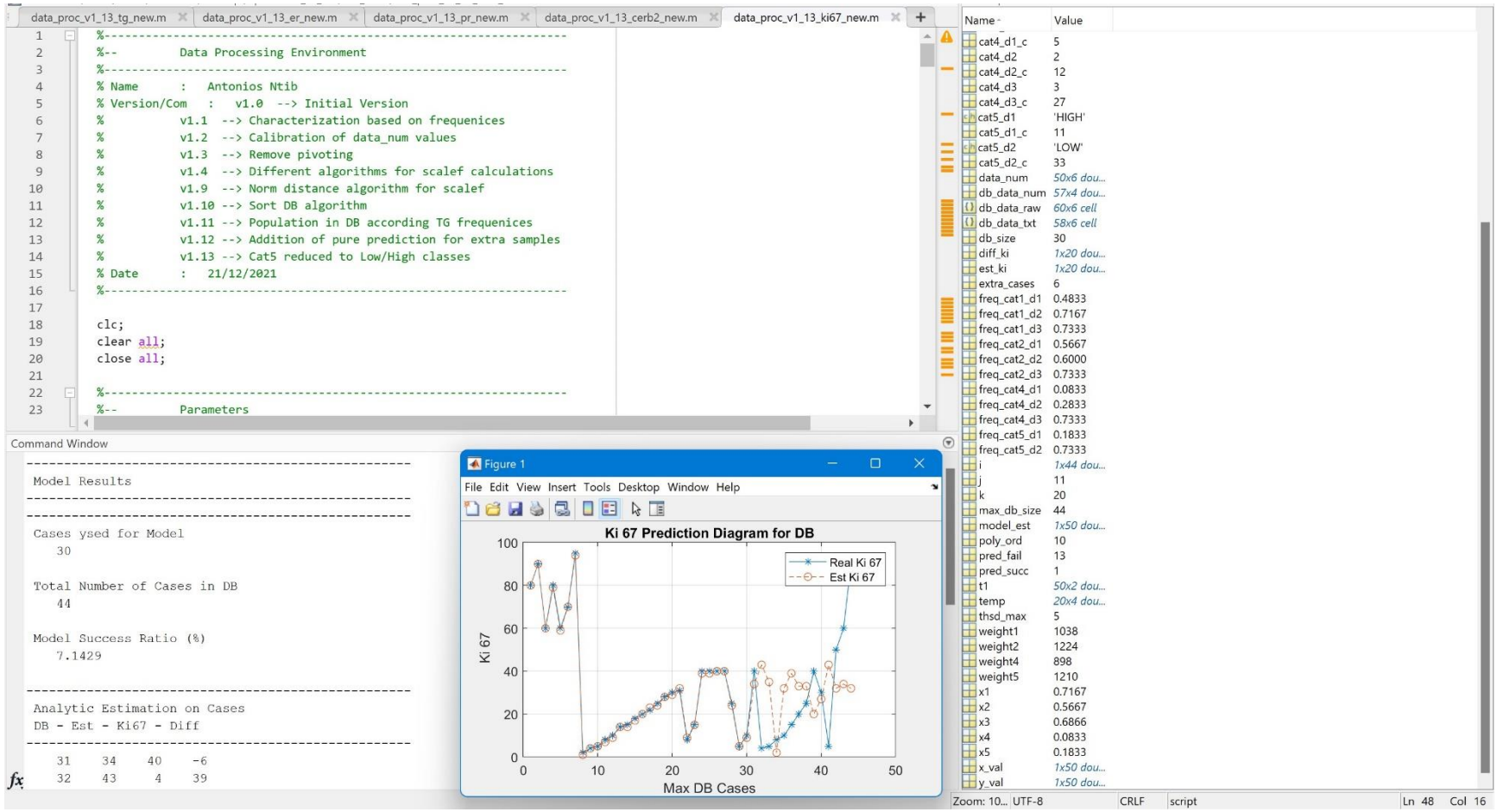
Εικόνα 27: Στιγμιότυπο μετά από την εκτέλεση του κώδικα για τον δείκτη ER.



Εικόνα 28: Στιγμιότυπο μετά από την εκτέλεση του κώδικα για τον δείκτη PR.



Εικόνα 29: Στιγμιότυπο μετά από την εκτέλεση του κώδικα για τον δείκτη CERB-2.



Εικόνα 30: Στιγμιότυπο μετά από την εκτέλεση του κώδικα για τον δείκτη Ki-67.

Κεφάλαιο 4: Αποτελέσματα

Σε αυτό το κεφάλαιο θα παρουσιαστούν και αναλυθούν τα αποτελέσματα από τους αλγορίθμους που κατασκευάστηκαν στο MATLAB.

- **TUMOR GRADE**

Model Results

DB sorting Enabled!

DB not sorted using frequencies

DB elements sorted iterated!

Max Number of Iterations

22

First member is Entry in DB

1

Constant Threshold used!

DB normed Distance

2

Cases ysed for Model

21

Total Number of Cases in DB

45

Model Success Ratio (%)

91.6667

Analytic Estimation on Cases

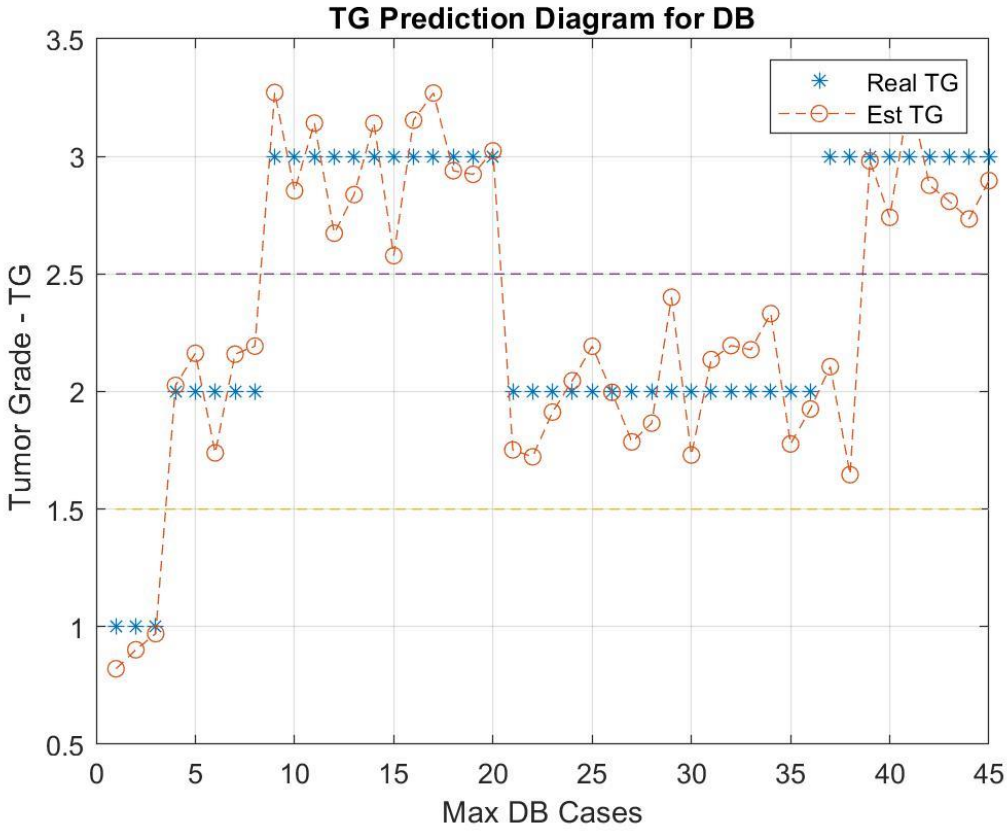
DB - scaled - Est - TG - Diff

4.0000	2.0000	1.7224	2.0000	0
23.0000	2.0000	1.9126	2.0000	0
24.0000	2.0000	2.0460	2.0000	0
25.0000	2.0000	2.1926	2.0000	0
26.0000	2.0000	1.9967	2.0000	0
27.0000	2.0000	1.7861	2.0000	0
28.0000	2.0000	1.8657	2.0000	0
29.0000	2.0000	2.4012	2.0000	0
30.0000	2.0000	1.7300	2.0000	0
31.0000	2.0000	2.1363	2.0000	0
32.0000	2.0000	2.1956	2.0000	0
33.0000	2.0000	2.1780	2.0000	0
34.0000	2.0000	2.3316	2.0000	0
35.0000	2.0000	1.7771	2.0000	0
36.0000	2.0000	1.9259	2.0000	0
37.0000	2.0000	2.1060	3.0000	-1.0000
38.0000	2.0000	1.6466	3.0000	-1.0000
39.0000	3.0000	2.9807	3.0000	0
40.0000	3.0000	2.7409	3.0000	0
41.0000	3.0000	3.2670	3.0000	0
42.0000	3.0000	2.8774	3.0000	0
43.0000	3.0000	2.8092	3.0000	0

44.0000	3.0000	2.7339	3.0000	0
45.0000	3.0000	2.8971	3.0000	0

Extra Cases

46.0000	3.0000	2.9002	3.0000	0
47.0000	3.0000	2.9705	3.0000	0
48.0000	3.0000	3.2898	3.0000	0
49.0000	3.0000	3.2170	3.0000	0
50.0000	3.0000	2.8004	3.0000	0



Εικόνα 31: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το Tumor Grade και των πραγματικών.

Χρησιμοποιήθηκαν 45 ασθενείς για την εκτέλεση του αλγορίθμου εκ των οποίων οι 21 χρησιμοποιήθηκαν για εκπαίδευση. Παρατηρείται πως έχοντας ενεργοποιημένη την επιλογή για ταξινόμηση της βάσης και απενεργοποιημένες όλες τις άλλες επιλογές που αφορούν ταξινόμηση με βάση την συχνότητα, τυχαία επιλογή ασθενή για εκκίνηση του αλγορίθμου, ταξινόμηση της βάσης μόνο μια φορά καθώς και μείωση του κατωφλίου που αφορά τον υπολογισμό της νόρμας για την διαφορά των τιμών μεταξύ εκτιμώμενων και πραγματικών, το ποσοστό ακρίβειας είναι περίπου 92.67%. Στο κομμάτι που αναγράφεται ως Extra Cases αφορά εκείνους τους ασθενείς που δεν είναι γνωστή ποια είναι η πραγματική τιμή και απλά ο αλγόριθμος εκτίμησε την τιμή του δείκτη Tumor Grade. Αυτά τα αποτελέσματα δεν έχουν συμπεριληφθεί στον υπολογισμό της ακρίβειας, καθ' ότι όπως αναφέρθηκε δεν είναι γνωστή η πραγματική τιμή. Επιπρόσθετα στο διάγραμμα φαίνονται πόσο απέχουν οι πραγματικές τιμές από τις εκτιμώμενες, ενώ η κίτρινη και η μωβ διακεκομμένη γραμμή, παράλληλες στον άξονα Max DB Cases, καθορίζουν τα όρια στα οποία ανήκουν οι εκτιμώμενες τιμές. Δηλαδή, πιο συγκεκριμένα, η περιοχή κάτω από την κίτρινη γραμμή αφορά τους ασθενείς με Tumor Grade ίσο με 1, μεταξύ της κίτρινης και μωβ γραμμής ίσο με 2, ενώ πάνω από την μωβ γραμμή ίσο με 3. Αυτές οι γραμμές είναι ουσιαστικά η οπτική αναπαράσταση των κατωφλίων που χρησιμοποιεί ο κώδικας για να προσδιορίσει με ακρίβεια σε ποια τιμή Tumor Grade αντιστοιχίζεται ο κάθε ασθενής. Τέλος, όπως φαίνεται στον πίνακα Analytic Estimation on Cases, υπάρχουν οι στήλες DB, scaled , Est, TG και Diff. Η στήλη DB φανερώνει τον αριθμό του ασθενή που χρησιμοποιήθηκε για εκτίμηση, το scaled σε ποια κατηγόρια ανήκει η εκτίμηση, το Est την ακριβή τιμή της εκτίμησης, το TG την πραγματική τιμή και το Diff πόσο απείχε η κατηγορία του Tumor Grade που ανήκει η εκτίμηση από την πραγματική τιμή. Όπου υπάρχει στην στήλη Diff τιμή διάφορη του μηδενός, αυτό σημαίνει ότι ο αλγόριθμος απέτυχε να προσδιορίσει σωστά το Tumor Grade.

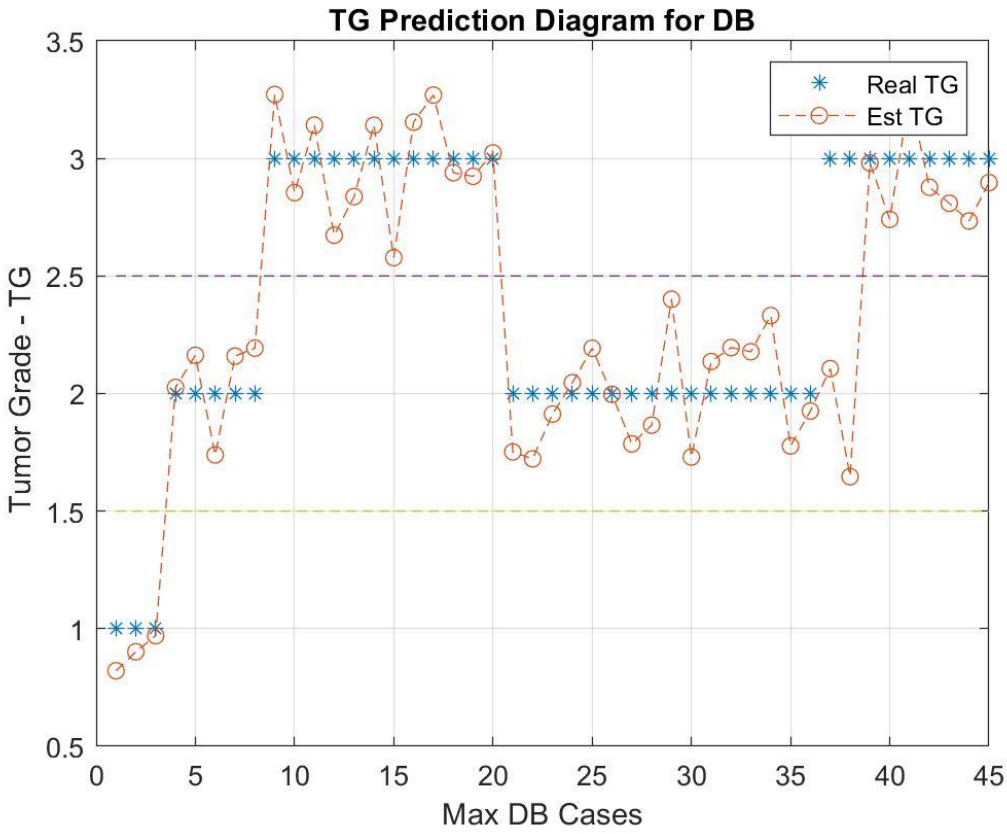
Ενδεικτικές εκτελέσεις του αλγορίθμου με διαφορετικές προεπιλογές:

- ✓ **Τυχαία επιλογή ασθενή (ασθενής 41) για εκκίνηση :**

```
--------------------------------------------------
 Model Results
--------------------------------------------------
 DB sorting Enabled!
 DB not sorted using frequencies
 DB elements sorted iterated!
 Max Number of Iterations
   65

 Random First Entry in DB
   41

 Constant Threshold used!
 DB normed Distance
    2
```

```
--------------------------------------------------
 Cases ysed for Model
   21

 Total Number of Cases in DB
   45

 Model Success Ratio (%)
   54.1667

--------------------------------------------------
 Analytic Estimation on Cases
 DB - scaled - Est - TG - Diff
--------------------------------------------------
    1.0000    2.0000    1.6829    1.0000    1.0000
    2.0000    2.0000    1.8602    1.0000    1.0000
    3.0000    2.0000    1.9800    1.0000    1.0000
   21.0000    2.0000    1.9927    3.0000   -1.0000
   22.0000    2.0000    1.6751    2.0000         0
   23.0000    2.0000    1.8728    2.0000         0
   24.0000    2.0000    2.0377    2.0000         0
   25.0000    2.0000    2.1925    2.0000         0
   26.0000    2.0000    1.9646    2.0000         0
   27.0000    2.0000    1.7674    2.0000         0
   28.0000    2.0000    1.8855    2.0000         0
   29.0000    2.0000    2.5019    2.0000    1.0000
   30.0000    2.0000    1.7296    2.0000         0
   31.0000    2.0000    2.1106    2.0000         0
   32.0000    2.0000    2.2808    2.0000         0
   33.0000    2.0000    2.1724    2.0000         0
   34.0000    2.0000    2.3886    2.0000         0
   35.0000    3.0000    2.7312    2.0000    1.0000
   36.0000    3.0000    2.9362    2.0000    1.0000
   37.0000    3.0000    3.0983    3.0000         0
   38.0000    3.0000    2.4087    3.0000   -1.0000
   39.0000    2.0000    2.0239    3.0000   -1.0000
   40.0000    2.0000    1.8112    3.0000   -1.0000
   42.0000    2.0000    1.9462    3.0000   -1.0000

--------------------------------------------------
          Extra Cases
--------------------------------------------------
   46.0000    2.0000    1.9687    2.0000         0
   47.0000    2.0000    1.9716    2.0000         0
   48.0000    2.0000    2.1949    2.0000         0
   49.0000    2.0000    2.1840    2.0000         0
```

50.0000 2.0000 1.8560 2.0000 0



Εικόνα 32: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το Tumor Grade και των πραγματικών με τυχαία επιλογή ασθενή.

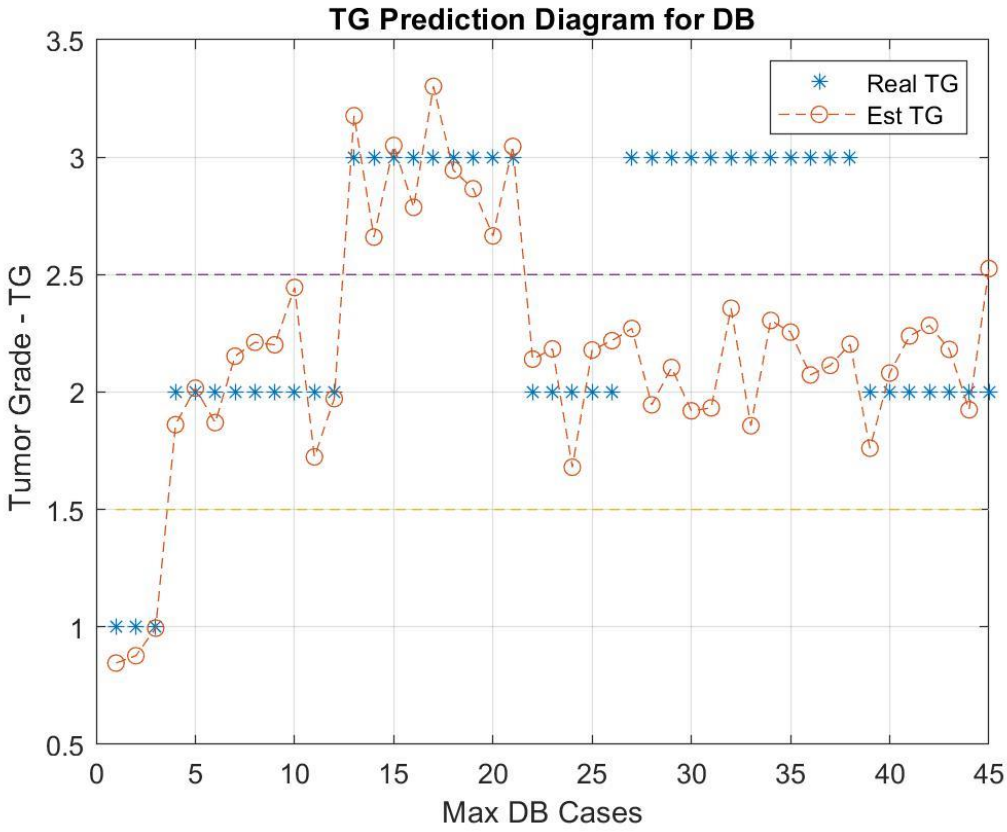
✓ **Τυχαία επιλογή ασθενή (ασθενής 28) για εκκίνηση :**

```
-------------------------------------------------
 Model Results
-------------------------------------------------
 DB sorting Enabled!
 DB not sorted using frequencies
 DB elements sorted iterated!
 Max Number of Iterations
   49

 Random First Entry in DB
```

```
    28

 Constant Threshold used!
 DB normed Distance
     2

-------------------------------------------------
 Cases ysed for Model
    21

 Total Number of Cases in DB
    45

 Model Success Ratio (%)
   45.8333

-------------------------------------------------
 Analytic Estimation on Cases
 DB - scaled - Est - TG - Diff
-------------------------------------------------
    5.0000    2.0000    2.1404    2.0000         0
    6.0000    2.0000    2.1837    2.0000         0
    7.0000    2.0000    1.6801    2.0000         0
    8.0000    2.0000    2.1793    2.0000         0
    9.0000    2.0000    2.2189    2.0000         0
   10.0000    2.0000    2.2709    3.0000   -1.0000
   11.0000    2.0000    1.9457    3.0000   -1.0000
   12.0000    2.0000    2.1046    3.0000   -1.0000
   13.0000    2.0000    1.9202    3.0000   -1.0000
   14.0000    2.0000    1.9325    3.0000   -1.0000
   15.0000    2.0000    2.3571    3.0000   -1.0000
   16.0000    2.0000    1.8567    3.0000   -1.0000
   17.0000    2.0000    2.3049    3.0000   -1.0000
   18.0000    2.0000    2.2554    3.0000   -1.0000
   19.0000    2.0000    2.0728    3.0000   -1.0000
   20.0000    2.0000    2.1135    3.0000   -1.0000
   21.0000    2.0000    2.2039    3.0000   -1.0000
   22.0000    2.0000    1.7609    2.0000         0
   23.0000    2.0000    2.0811    2.0000         0
   24.0000    2.0000    2.2390    2.0000         0
   25.0000    2.0000    2.2841    2.0000         0
   26.0000    2.0000    2.1819    2.0000         0
   27.0000    2.0000    1.9246    2.0000         0
   29.0000    2.0000    2.5257    2.0000    1.0000

-------------------------------------------------
```

Extra Cases

46.0000	2.0000	1.9808	2.0000	0
47.0000	2.0000	2.2253	2.0000	0
48.0000	2.0000	2.2189	2.0000	0
49.0000	2.0000	2.3533	2.0000	0
50.0000	2.0000	1.9559	2.0000	0



Εικόνα 33: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το Tumor Grade και των πραγματικών με τυχαία επιλογή ασθενή - Επανεκτέλεση.

Παρατηρούμε πως το ποσοστό ακρίβειας αλλάζει καθώς ο αλγόριθμος κάθε φορά λαμβάνει διαφορετικό δείγμα για εκπαίδευση.

- ✓ **Χρήση της προεπιλογής για εισαγωγή στην βάση εκπαίδευσης με βάση την συχνότητα εμφάνισης και χωρίς ταξινόμηση:**

Model Results

Cases ysed for Model

21

Total Number of Cases in DB

45

Model Success Ratio (%)

79.1667

Analytic Estimation on Cases

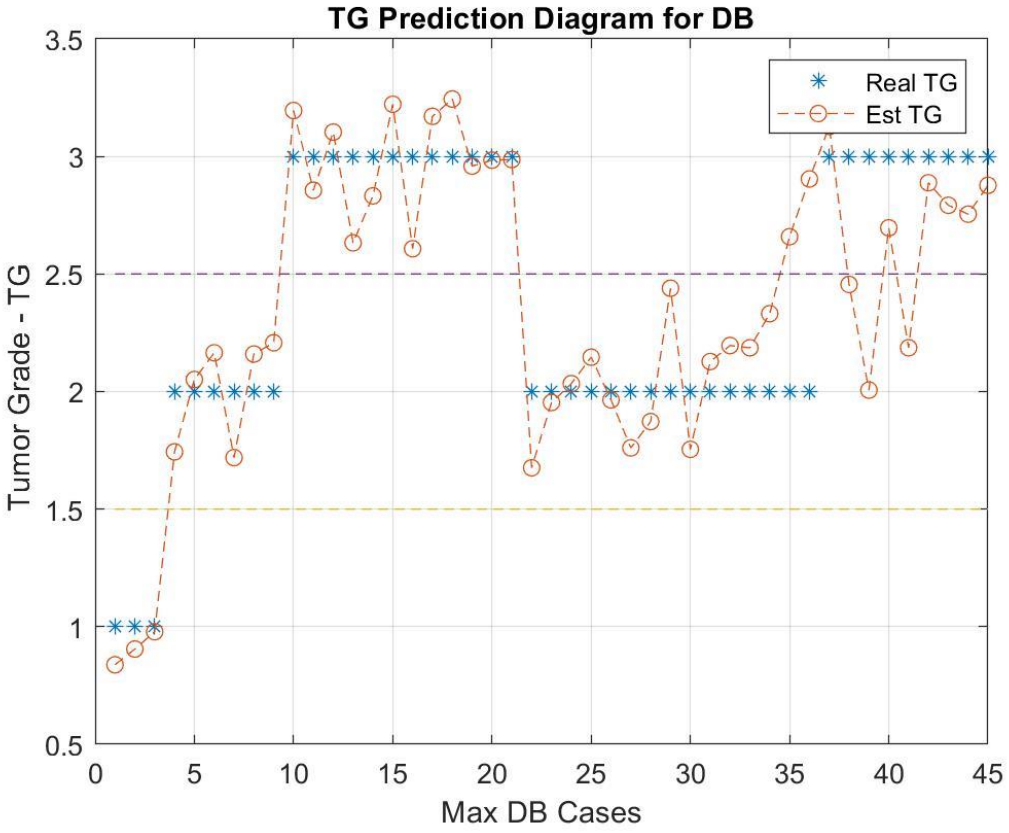
DB - scaled - Est - TG - Diff

22.0000	2.0000	1.6758	2.0000	0
23.0000	2.0000	1.9535	2.0000	0
24.0000	2.0000	2.0337	2.0000	0
25.0000	2.0000	2.1463	2.0000	0
26.0000	2.0000	1.9645	2.0000	0
27.0000	2.0000	1.7605	2.0000	0
28.0000	2.0000	1.8724	2.0000	0
29.0000	2.0000	2.4393	2.0000	0
30.0000	2.0000	1.7543	2.0000	0

31.0000	2.0000	2.1276	2.0000	0
32.0000	2.0000	2.1957	2.0000	0
33.0000	2.0000	2.1862	2.0000	0
34.0000	2.0000	2.3312	2.0000	0
35.0000	3.0000	2.6585	2.0000	1.0000
36.0000	3.0000	2.9041	2.0000	1.0000
37.0000	3.0000	3.1275	3.0000	0
38.0000	3.0000	2.4556	3.0000	-1.0000
39.0000	2.0000	2.0061	3.0000	-1.0000
40.0000	3.0000	2.6963	3.0000	0
41.0000	2.0000	2.1862	3.0000	-1.0000
42.0000	3.0000	2.8881	3.0000	0
43.0000	3.0000	2.7922	3.0000	0
44.0000	3.0000	2.7545	3.0000	0
45.0000	3.0000	2.8770	3.0000	0

Extra Cases

46.0000	3.0000	2.9201	3.0000	0
47.0000	3.0000	2.9822	3.0000	0
48.0000	3.0000	3.3113	3.0000	0
49.0000	3.0000	3.2584	3.0000	0
50.0000	3.0000	2.7411	3.0000	0



Εικόνα 34: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το Tumor Grade και των πραγματικών με εισαγωγή ασθενών με βάση την συχνότητα εμφάνισης στην βάση εκπαίδευσης και χωρίς ταξινόμηση.

❖ **ER**

Model Results

DB sorting Enabled!

DB not sorted using frequencies

DB elements sorted iterated!

Max Number of Iterations

104

First member is Entry in DB

1

Constant Threshold used!

DB normed Distance

0.2500

Cases ysed for Model

21

Total Number of Cases in DB

45

Model Success Ratio (%)

100

Analytic Estimation on Cases

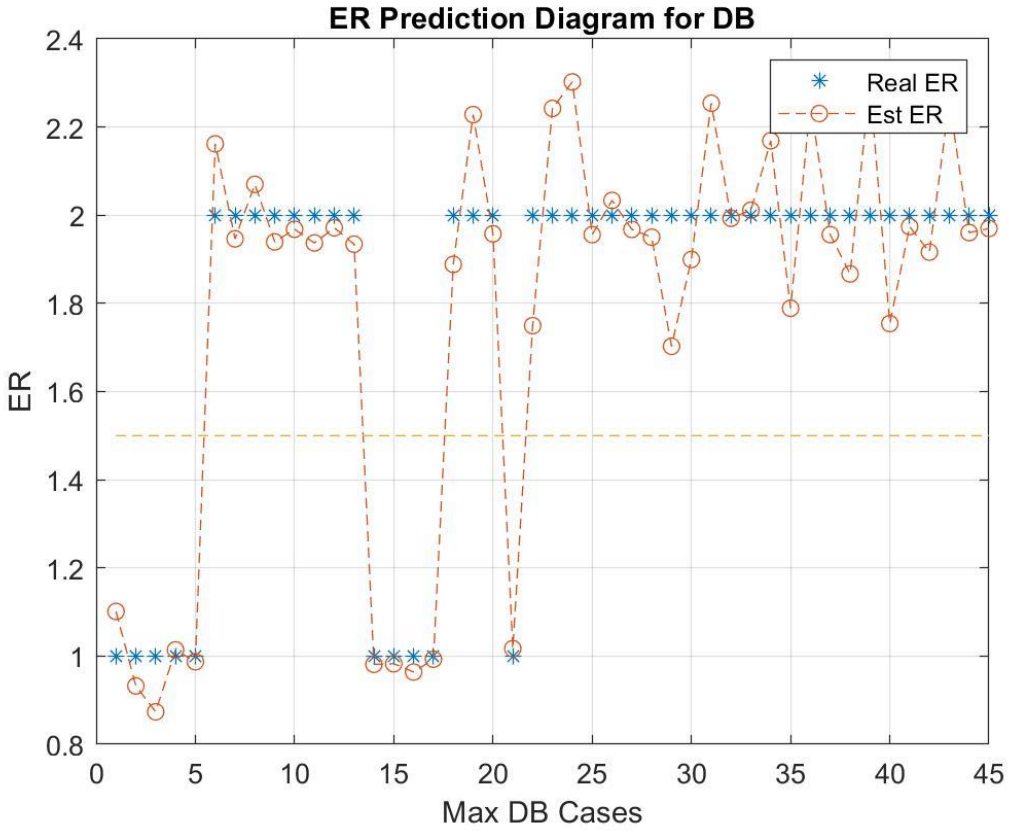
DB - scaled - Est - ER - Diff

6.0000	2.0000	1.7495	2.0000	0
15.0000	2.0000	2.2423	2.0000	0
16.0000	2.0000	2.3025	2.0000	0
17.0000	2.0000	1.9559	2.0000	0
18.0000	2.0000	2.0336	2.0000	0
19.0000	2.0000	1.9678	2.0000	0
20.0000	2.0000	1.9501	2.0000	0
21.0000	2.0000	1.7026	2.0000	0

22.0000	2.0000	1.8996	2.0000	0
23.0000	2.0000	2.2540	2.0000	0
24.0000	2.0000	1.9931	2.0000	0
25.0000	2.0000	2.0114	2.0000	0
26.0000	2.0000	2.1689	2.0000	0
27.0000	2.0000	1.7890	2.0000	0
29.0000	2.0000	2.2540	2.0000	0
30.0000	2.0000	1.9559	2.0000	0
33.0000	2.0000	1.8669	2.0000	0
34.0000	2.0000	2.2979	2.0000	0
35.0000	2.0000	1.7542	2.0000	0
36.0000	2.0000	1.9742	2.0000	0
37.0000	2.0000	1.9162	2.0000	0
38.0000	2.0000	2.2576	2.0000	0
40.0000	2.0000	1.9605	2.0000	0
41.0000	2.0000	1.9699	2.0000	0

Extra Cases

46.0000	2.0000	2.2600	2.0000	0
47.0000	2.0000	2.2094	2.0000	0
48.0000	2.0000	2.0540	2.0000	0
49.0000	2.0000	1.8979	2.0000	0



Εικόνα 35: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το ER και των πραγματικών.

Χρησιμοποιήθηκαν 45 ασθενείς για την εκτέλεση του αλγορίθμου εκ των οποίων οι 21 χρησιμοποιήθηκαν για εκπαίδευση. Παρατηρείται πως έχοντας ενεργοποιημένη την επιλογή για ταξινόμηση της βάσης και απενεργοποιημένες όλες τις άλλες επιλογές που αφορούν ταξινόμηση με βάση την συχνότητα, τυχαία επιλογή ασθενή για εκκίνηση του αλγορίθμου, ταξινόμηση της βάσης μόνο μια φορά καθώς και μείωση του κατωφλίου που αφορά τον υπολογισμό της νόρμας για την διαφορά των τιμών μεταξύ εκτιμώμενων και πραγματικών, το ποσοστό ακρίβειας είναι 100%.

- ✓ **Ενδεικτική εκτέλεση όπου ο αλγόριθμος δεν μπορεί να εξάγει αποτελέσματα λόγω συνδυασμού προεπιλογών και αυστηρού κατωφλίου**

Not enough DB members after sorting!

Relax db_dist_thsd!

5

Παρατηρείται πως με ενεργοποιημένη την ταξινόμηση της βάσης καθώς και της επιλογής για εισαγωγή περιπτώσεων ασθενών με βάση την συχνότητα εμφάνισης το κατώφλι που ορίζει την μέγιστη δυνατή διαφορά της νόρμας μεταξύ πραγματικής και εκτιμώμενης τιμής περιορίζει αρκετά το πόσες περιπτώσεις θα εισαχθούν στην βάση για εκπαίδευση. Αυτό τερματίζει το αλγόριθμο απρόσμενα καθώς με πενιχρό πλήθος ασθενών για εκπαίδευση ο αλγόριθμος δεν πρόκειται να προβλέψει με ικανοποιητική ακρίβεια τον δείκτη. Αν αυξήσουμε το κατώφλι, δηλαδή αλλάξουμε την τιμή της μεταβλητής db_dist_thsd από 0.25 σε 0.35 ο αλγόριθμος εκτελείται κανονικά όμως το ποσοστά ακρίβειας δεν είναι και τα καλύτερα δυνατά:

Model Results

DB sorting Enabled!

Db sorting according Frequencies

DB elements sorted iterated!

Max Number of Iterations

70

First member is Entry in DB

1

Constant Threshold used!

DB normed Distance

0.3500

Cases ysed for Model

21

Total Number of Cases in DB

45

Model Success Ratio (%)

50

Analytic Estimation on Cases

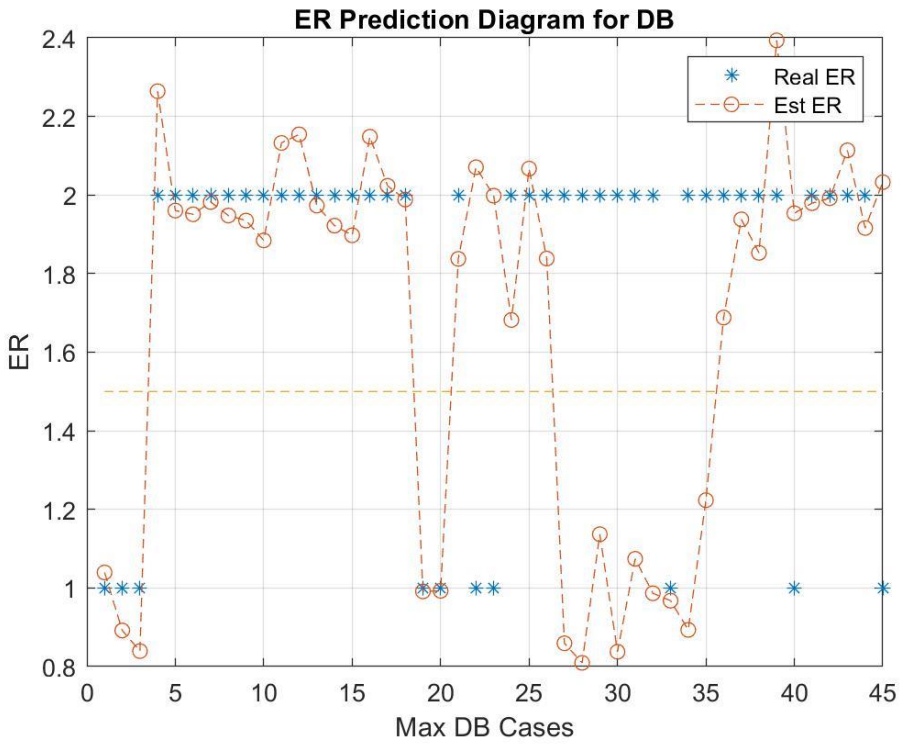
DB - scaled - Est - ER - Diff

4.0000	2.0000	2.0707	1.0000	1.0000
5.0000	2.0000	1.9983	1.0000	1.0000
6.0000	2.0000	1.6819	2.0000	0
9.0000	2.0000	2.0672	2.0000	0
18.0000	2.0000	1.8383	2.0000	0
20.0000	1.0000	0.8592	2.0000	-1.0000
21.0000	1.0000	0.8095	2.0000	-1.0000
26.0000	1.0000	1.1368	2.0000	-1.0000
27.0000	1.0000	0.8376	2.0000	-1.0000
29.0000	1.0000	1.0741	2.0000	-1.0000
30.0000	1.0000	0.9865	2.0000	-1.0000
32.0000	1.0000	0.9676	1.0000	0
33.0000	1.0000	0.8934	2.0000	-1.0000
34.0000	1.0000	1.2233	2.0000	-1.0000
35.0000	2.0000	1.6882	2.0000	0
36.0000	2.0000	1.9380	2.0000	0
37.0000	2.0000	1.8528	2.0000	0
38.0000	2.0000	2.3935	2.0000	0
39.0000	2.0000	1.9538	1.0000	1.0000
40.0000	2.0000	1.9793	2.0000	0
41.0000	2.0000	1.9919	2.0000	0

43.0000	2.0000	2.1136	2.0000	0
44.0000	2.0000	1.9160	2.0000	0
45.0000	2.0000	2.0326	1.0000	1.0000

Extra Cases

46.0000	2.0000	2.3362	2.0000	0
47.0000	2.0000	2.0885	2.0000	0
48.0000	2.0000	2.1053	2.0000	0
49.0000	1.0000	0.9439	1.0000	0



Εικόνα 36: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το ER και των πραγματικών με χρήση εισαγωγής ασθενών στη βάση με γνώμονα την συχνότητα εμφάνισης καθώς και κατώφλι νόρμας 0.35 αντί του 0.25.

Αξίζει να επισημανθεί πως εδώ οι τιμές δεν είναι αριθμητικές αλλά οι επιλογές Yes ή No, το οποίο και έχουμε αντιστοιχίσει στις τιμές 1 και 2 αντίστοιχα. Επίσης το κατώφλι για τον ορισμό των περιοχών που ορίζει στον αν είναι Yes ή No είναι η κίτρινη διακεκομμένη γραμμή, παράλληλη στο Max DB cases, και είναι ορισμένη στο 1.5.

❖ **PR**

Model Results

DB sorting Enabled!

DB not sorted using frequencies

DB elements sorted iterated!

Max Number of Iterations

64

First member is Entry in DB

1

Constant Threshold used!

DB normed Distance

0.2500

Cases ysed for Model

23

Total Number of Cases in DB

45

Model Success Ratio (%)

100

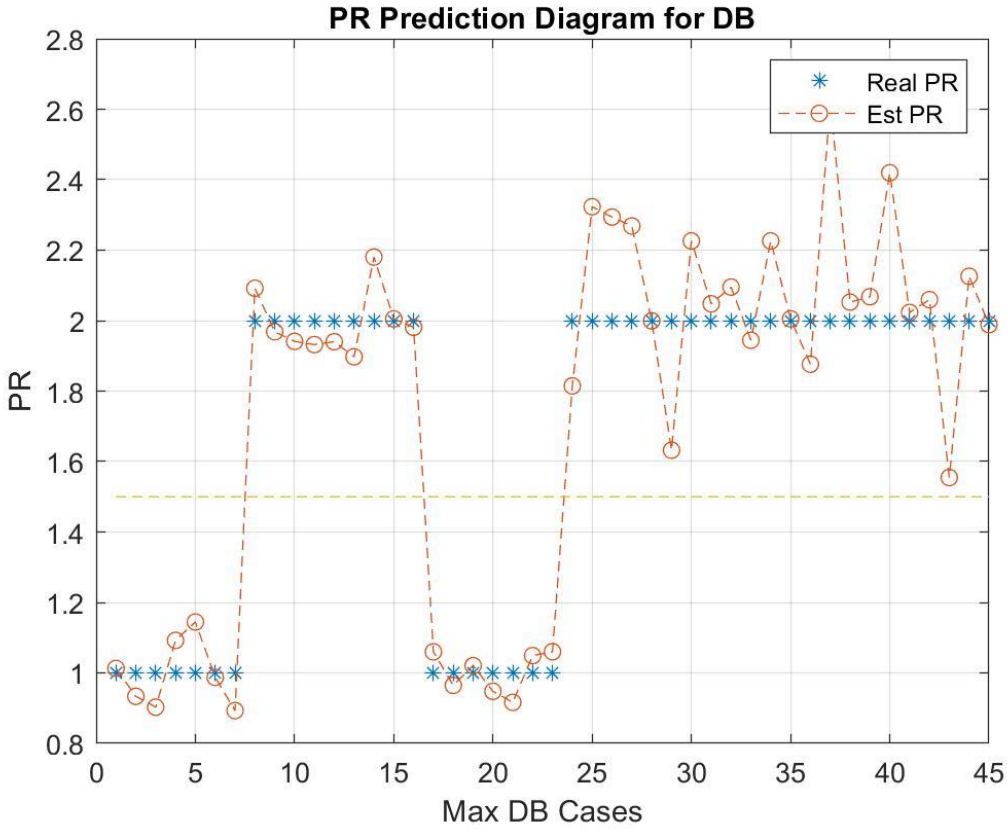
Analytic Estimation on Cases

DB - scaled - Est - PR - Diff

8.0000	2.0000	1.8146	2.0000	0
11.0000	2.0000	2.3233	2.0000	0
13.0000	2.0000	2.2940	2.0000	0
20.0000	2.0000	2.2686	2.0000	0
21.0000	2.0000	1.9993	2.0000	0
22.0000	2.0000	1.6321	2.0000	0
23.0000	2.0000	2.2262	2.0000	0
24.0000	2.0000	2.0477	2.0000	0
25.0000	2.0000	2.0952	2.0000	0
27.0000	2.0000	1.9443	2.0000	0
29.0000	2.0000	2.2262	2.0000	0
30.0000	2.0000	2.0049	2.0000	0
33.0000	2.0000	1.8763	2.0000	0
34.0000	2.0000	2.6212	2.0000	0
36.0000	2.0000	2.0525	2.0000	0
37.0000	2.0000	2.0679	2.0000	0
38.0000	2.0000	2.4197	2.0000	0
40.0000	2.0000	2.0232	2.0000	0
41.0000	2.0000	2.0597	2.0000	0
42.0000	2.0000	1.5546	2.0000	0
43.0000	2.0000	2.1258	2.0000	0
44.0000	2.0000	1.9886	2.0000	0

Extra Cases

46.0000	2.0000	2.4498	2.0000	0
47.0000	2.0000	2.0528	2.0000	0
48.0000	2.0000	2.2849	2.0000	0
49.0000	2.0000	2.0203	2.0000	0



Εικόνα 37: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το PR και των πραγματικών.

Χρησιμοποιήθηκαν 45 ασθενείς για την εκτέλεση του αλγορίθμου εκ των οποίων οι 23 χρησιμοποιήθηκαν για εκπαίδευση. Ομοίως με προηγουμένως, έχοντας ενεργοποιημένη την επιλογή για ταξινόμηση της βάσης και απενεργοποιημένες όλες τις άλλες επιλογές που αφορούν ταξινόμηση με βάση την συχνότητα, τυχαία επιλογή ασθενή για εκκίνηση του

αλγορίθμου, ταξινόμηση της βάσης μόνο μια φορά καθώς και μείωση του κατωφλίου που αφορά τον υπολογισμό της νόρμας για την διαφορά των τιμών μεταξύ εκτιμώμενων και πραγματικών, παρατηρείται πως το ποσοστό ακρίβειας είναι στο 100%. Μια ακόμη ομοιότητα είναι και το γεγονός ότι πάλι το αποτέλεσμα για αυτό τον δείκτη είναι οι τιμές Yes ή No, οπότε ακολουθήθηκε το ίδιο μοτίβο με την αντιστοίχιση τους στις τιμές 1 ή 2. Το κατώφλι που διαχωρίζει τις 2 περιοχές επιλογών είναι ορισμένο στο 1.5. Οτιδήποτε κάτω από το 1.5 θεωρείται No και πάνω από το 1.5 θεωρείται Yes.

- **CERB-2**

Model Results

DB sorting Enabled!

DB not sorted using frequencies

DB elements sorted iterated!

Max Number of Iterations

121

First member is Entry in DB

1

Constant Threshold used!

DB normed Distance

0.2500

Cases ysed for Model

38

Total Number of Cases in DB

44

Model Success Ratio (%)

100

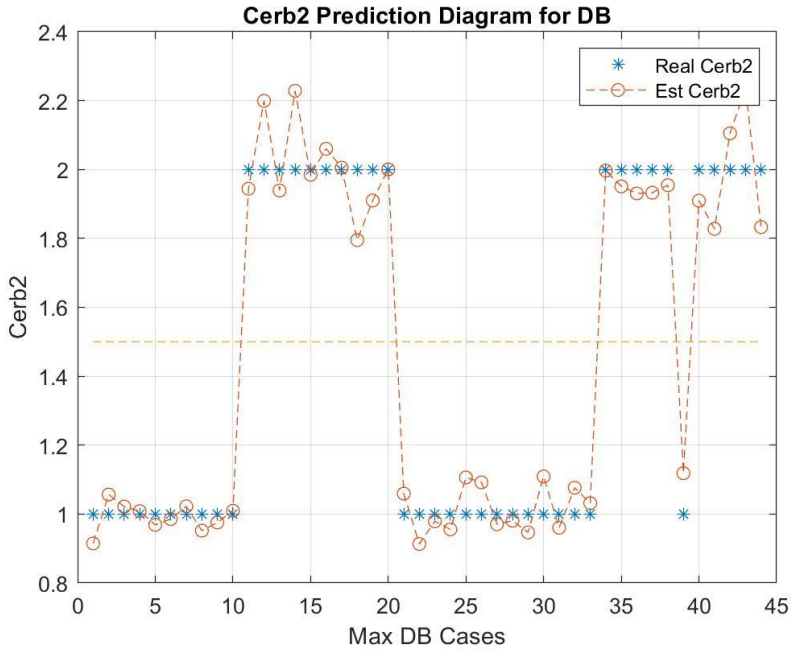
Analytic Estimation on Cases

DB - scaled - Est - Cerb2 - Diff

10.0000	1.0000	1.1185	1.0000	0
14.0000	2.0000	1.9095	2.0000	0
16.0000	2.0000	1.8278	2.0000	0
18.0000	2.0000	2.1051	2.0000	0
39.0000	2.0000	2.2370	2.0000	0
42.0000	2.0000	1.8329	2.0000	0

Extra Cases

45.0000	2.0000	1.9997	2.0000	0
46.0000	2.0000	2.2044	2.0000	0
47.0000	2.0000	2.2811	2.0000	0
48.0000	2.0000	1.9656	2.0000	0



Εικόνα 38: Παρουσίαση διαγράμματος εκτιμώμενων τιμών για το CERB-2 και των πραγματικών.

Χρησιμοποιήθηκαν 44 ασθενείς για την εκτέλεση του αλγορίθμου εκ των οποίων οι 38 χρησιμοποιήθηκαν για εκπαίδευση. Έχοντας ενεργοποιημένη την επιλογή για ταξινόμηση της βάσης και απενεργοποιημένες όλες τις άλλες επιλογές που αφορούν ταξινόμηση με βάση την συχνότητα, τυχαία επιλογή ασθενή για εκκίνηση του αλγορίθμου, ταξινόμηση της βάσης μόνο μια φορά καθώς και μείωση του κατωφλίου που αφορά τον υπολογισμό της νόρμας για την διαφορά των τιμών μεταξύ εκτιμώμενων και πραγματικών, παρατηρείται πως το ποσοστό ακρίβειας είναι στο 100%. Τέλος και εδώ η εκτίμηση του δείκτη είναι με τιμές Yes ή No, οπότε ακολουθήθηκε ακριβώς η ίδια στρατηγική με τα ER και PR.

Κεφάλαιο 5: Σύνοψη, Περιορισμοί, Μελλοντικές Επεκτάσεις

Σε αυτό το κεφάλαιο θα συζητηθούν συνοπτικά οι αλγόριθμοι, καθώς και οι περιορισμοί και οι επεκτάσεις που θα μπορούσαν να γίνουν για μια πιο ολοκληρωμένη πρόβλεψη των δεικτών και διάγνωση των ασθενών.

5.1 Σύνοψη

Στόχος αυτής της διπλωματικής ήταν η ανάπτυξη αλγορίθμων για την πρόβλεψη ιστολογικών δεικτών ασθενών που ανιχνεύονταν με κάποιο ύποπτο όγκο στο μαστό. Τα στοιχεία εξάγονταν από μαγνητικές τομογραφίες και καταγράφονταν σε ένα αρχείο excel. Υπήρχαν συνολικά 26 στήλες με δεδομένα για 77 ασθενείς. Για την παρούσα διπλωματική χρησιμοποιήθηκαν 45 ασθενείς, σε κάποιες περιπτώσεις 44 και 5 από τις 26 στήλες που περιείχαν δεδομένα. Αυτές ήταν: Morphology, Borders, Tumor Size, Curve Morphology και το ADC. Στόχος ήταν οι πρόβλεψη των πέντε ιστολογικών δεικτών: Tumor Grade, ER, PR, CERB-2 και Ki-67. Το πρώτο βήμα ήταν η δημιουργία 5 excel αρχείων που θα περιείχαν το καθένα τις 5 στήλες (Morphology, Borders, Tumor Size, Curve Morphology, ADC) και μια επιπλέον στήλη που θα περιείχε μια από τις 5 μεταβλητές για πρόβλεψη. Στη συνέχεια για τους δείκτες Tumor Grade, ER, PR, CERB-2, που η διαδικασία που ακολουθούν οι αλγόριθμοι είναι η ίδια, γίνεται η αποθήκευση τους σε πίνακες και γίνεται ταξινόμηση ανά συχνότητα εμφάνισης καθώς και η κανονικοποίηση τους. Οι αλγόριθμοι λαμβάνουν περίπου 25 ασθενείς (διαφέρει λίγο το πλήθος των ασθενών ανά κατηγορία πρόβλεψης) για εκπαίδευση και στη συνέχεια με τους εναπομείναντες ασθενείς γίνεται η πρόβλεψη. Επειδή χρησιμοποιείται το μοντέλο Γραμμικής Παρεμβολής για πρόβλεψη τα αποτελέσματα έχουν αποκλίσεις από τις ακριβείς τιμές. Παρ' όλα ταύτα έχουν οριστεί κατώφλια για την κάθε κατηγορία αν τα αποτελέσματα βρίσκονται στα διαστήματα των κατωφλίων τότε τα αποτελέσματα είναι ακριβή με υψηλό ποσοστό ακρίβειας. Τέλος, γίνεται η πρόβλεψη των ασθενών που δεν είναι γνωστά τα αποτελέσματα των δεικτών τους. Όσον αφορά τον ιστολογικό δείκτη Ki-67 παρ' ότι έγινε μια προσέγγιση με αλγόριθμο που προσπαθεί να προβλέψει τα αποτελέσματα κατασκευάζοντας κάθε φορά πολυώνυμα, τα αποτελέσματα δεν είναι ενθαρρυντικά.

5.2 Περιορισμοί

Όπως για κάθε μοντέλο που κατασκευάζεται για πρόβλεψη κάποιας μεταβλητής μέσω κάποιων δεδομένων, το μεγαλύτερο πρόβλημα αποτελεί το πλήθος των δεδομένων. Οι αλγόριθμοι έχουν δοκιμαστεί για ένα μικρό ποσό δεδομένων. Αυτό έγκειται στο γεγονός ότι υπάρχει πολύ μεγάλη δυσκολία στην συγκέντρωση τέτοιου είδους πληροφοριών καθώς δεν υπάρχει κάποιο ηλεκτρονικό σύστημα καταχώρησης στο οποίο θα καταγράφονται όλες αυτοί οι παράμετροι, πρώτον για χρήση από τους θεράποντες ιατρούς και κατά δεύτερον για χρήση στην έρευνα, εφ' όσον ο ασθενής το επιθυμεί.

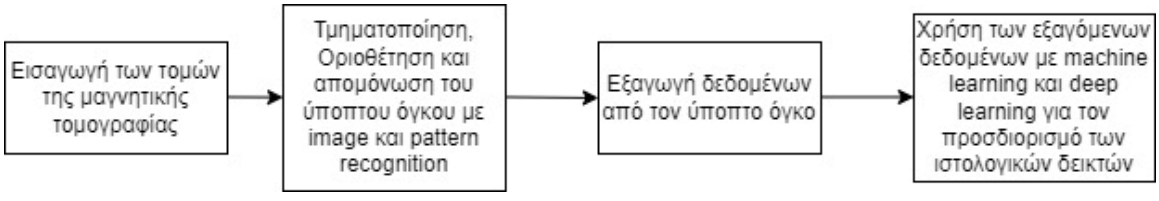
Ένα ακόμη πρόβλημα που παρουσιάστηκε είναι το γεγονός ότι τέτοιου είδους προσπάθειες βρίσκονται σε πρώιμο στάδιο, ενώ αυτές που έχουν επιδείξει πολύ υψηλά ποσοστά επιτυχίας είναι σε πολύ περιορισμένο αριθμό. Αυτό καθιστά δύσκολο το γεγονός της εύρεσης κάποιου εναύσματος για την επιλογή των κατάλληλων αλγορίθμων καθώς και την βελτίωση των υπαρχόντων για μελλοντική χρήση στην έρευνα.

Επιπρόσθετα, ένα πρόβλημα που έπρεπε να επιλυθεί ήταν το γεγονός ότι τα δεδομένα εισόδου και εξόδου ήταν και κατηγορικά καθώς και συνεχής αριθμητικές τιμές. Αυτό με βάση τους υπάρχοντες αλγορίθμους καθιστά πολύ δύσκολη την πρόβλεψη, καθώς συνήθως οι αλγόριθμοι λειτουργούν καλύτερα όταν έχουν ως είσοδο αλλά και να προβλέψουν ένα είδος δεδομένων και όχι μικτά.

5.3 Μελλοντικές Επεκτάσεις

Όπως ειπώθηκε και στο κεφάλαιο 6.2 πολύ μεγάλη ανάγκη αποτελεί η εύρεση δεδομένων για την εκπαίδευση και την δοκιμή του αλγορίθμου στην πρόβλεψη. Επιπρόσθετα για τον δείκτη Ki-67, αυτό αποτελεί αναγκαίο βήμα για την βελτίωση της ακρίβειας του αλγορίθμου. Επίσης, μια διαφορετική προσέγγιση του προβλήματος θα ήταν και η χρήση νευρωνικών δικτύων μέσω Deep Learning κάτι το οποίο πολύ πιθανόν θα βελτίωνε σημαντικά την ακρίβεια ειδικά του δείκτη Ki-67. Σημαντικό, επίσης να επισημανθεί ότι η εύρεση ενός ή περισσοτέρων νευρωνικών δικτύων με μια προσαρμοσμένη συνάρτηση ενεργοποίησης, ίσως να ήταν και αυτό που θα επίλυε το πρόβλημα κατά ένα πολύ μεγάλο βαθμό. Τέλος, μια ακόμη προσέγγιση όσον αφορά τα νευρωνικά δίκτυα, θα μπορούσαν να χρησιμοποιηθούν και με επικουρικό τρόπο στους υπάρχοντες αλγορίθμους καθώς θα μπορούσαν μέσα από το ολικό excel αρχείο να δοκίμαζαν επιλεκτικά κάποιες δευτερεύοντες στήλες ως δεδομένα εισαγωγής με μια συνάρτηση ενεργοποίησης και να βελτίωναν έτσι την ακρίβεια των εξαγόμενων αποτελεσμάτων.

Ένα πολύ σημαντικό βήμα ωστόσο σε μια ολοκληρωμένη και πιο αυτοματοποιημένη πρόβλεψη θα ήταν η κατασκευή και ενός αλγορίθμου στον οποίον θα εισάγονταν οι τομές από τις τρισδιάστατες μαγνητικές τομογραφίες στους μαστούς και με χρήση αναγνώρισης προτύπων και εικόνων (image recognition, pattern recognition) θα γινόταν η τμηματοποίηση και η οριοθέτηση του ύποπτου όγκου από την ολική εικόνα **και από εκεί** θα εξάγονταν τα διάφορα χαρακτηριστικά, που αυτή την στιγμή λαμβάνονται μέσω ενός excel αρχείου. Δηλαδή, αυτές οι στήλες δεδομένων που ένα περιορισμένο πλήθος τους δίνονται ως είσοδο στον αλγόριθμο να εξάγονται από την εικόνα που θα έχει οριοθετηθεί αυτόματα από ένα αλγόριθμο που θα έχει κατασκευαστεί για αυτήν ακριβώς την διαδικασία. Παρακάτω παρατίθεται και ένα διάγραμμα για την διαδικασία που θα μπορούσε να ακολουθηθεί:



Εικόνα 39: Διάγραμμα ροής για την πλήρη αυτοματοποίηση της διαδικασίας πρόβλεψης.

Τέλος, η δημιουργία κάποιου γραφικού περιβάλλοντος στο οποίο οι ιατροί θα μπορούσαν να έχουν την δυνατότητα να εισάγουν εύκολα και γρήγορα τις τομογραφίες καθώς και να συμπληρώνουν στοιχεία που δεν είναι δυνατόν να προβλεφθούν όπως η ηλικία, το ιστορικό του ασθενούς και το οικογενειακό ιστορικό τα οποία πιθανά θα συνεισφέραν και στην ακρίβεια της πρόβλεψης ή και ακόμα και στην διόρθωση των εξαχθέντων αποτελεσμάτων αν αυτά δεν ταιριάζουν απόλυτα με τα πραγματικά δεδομένα τα οποία εξάγονται από τους ίδιους με την φυσική εξέταση και την βιοψία.

Κεφάλαιο 6: Βιβλιογραφία

[1] K. Bera, K. A. Schalper, D. L. Rimm, V. Velcheti, and A. Madabhushi, 'Artificial intelligence in digital pathology — new tools for diagnosis and precision oncology', *Nat. Rev. Clin. Oncol.*, vol. 16, no. 11, pp. 703–715, Nov. 2019, doi: 10.1038/s41571-019-0252-y.

[2] J. E. E. Arthur, 'Using Machine Learning on an Imbalanced Cancer Dataset', *ETD Collect. Univ. Tex. El Paso*, pp. 1–57, Jan. 2020.

[3] P. Harrison and K. Park, 'Tumor Detection In Breast Histopathological Images Using Faster R-CNN', in *2021 International Symposium on Medical Robotics (ISMR)*, Nov. 2021, pp. 1–7. doi: 10.1109/ISMR48346.2021.9661483.

[4] A. Aksac, D. J. Demetrick, T. Ozyer, and R. Alhajj, 'BreCaHAD: a dataset for breast cancer histopathological annotation and diagnosis', *BMC Res. Notes*, vol. 12, no. 1, p. 82, Feb. 2019, doi: 10.1186/s13104-019-4121-7.

[5] A. Chauhan, H. Kharpate, Y. Narekar, S. Gulhane, T. Virulkar, and Y. Hedau, 'Breast Cancer Detection and Prediction using Machine Learning', in *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, Sep. 2021, pp. 1135–1143. doi: 10.1109/ICIRCA51532.2021.9544687.

[6] H. Lee *et al.*, 'Classification of MR-Detected Additional Lesions in Patients With Breast Cancer Using a Combination of Radiomics Analysis and Machine Learning', *Front. Oncol.*, vol. 11, 2021, Accessed: Feb. 17, 2022. [Online]. Available: https://www.frontiersin.org/article/10.3389/fonc.2021.744460

[7] S. Rajpal, M. Agarwal, V. Kumar, A. Gupta, and N. Kumar, 'Triphasic DeepBRCA-A Deep Learning-Based Framework for Identification of Biomarkers for Breast Cancer Stratification', *IEEE Access*, vol. 9, pp. 103347–103364, 2021, doi: 10.1109/ACCESS.2021.3093616.

[8] C.-M. Kim, R. C. Park, and E. J. Hong, 'Breast Mass Classification Using eLFA Algorithm Based on CRNN Deep Learning Model', *IEEE Access*, vol. 8, pp. 197312–197323, 2020, doi: 10.1109/ACCESS.2020.3034914.

[9] J. Zheng, D. Lin, Z. Gao, S. Wang, M. He, and J. Fan, 'Deep Learning Assisted Efficient AdaBoost Algorithm for Breast Cancer Detection and Early Diagnosis', *IEEE Access*, vol. 8, pp. 96946–96954, 2020, doi: 10.1109/ACCESS.2020.2993536.