



ΔΙΕΘΝΕΣ  
ΠΑΝΕΠΙΣΤΗΜΙΟ  
ΤΗΣ ΕΛΛΑΔΟΣ

Σχολή Μηχανικών  
Τμήμα Μηχανικών Παραγωγής και Διοίκησης

### Διπλωματική Εργασία

---

Ταξινόμηση του καρκίνου του μαστού με  
μεθόδους μηχανικής μάθησης, βάσει δεδομένων  
που εξήχθησαν από παρακέντηση

---

Εκπόνηση:  
Πανιτσίδης Λάζαρος  
ΑΕΜ: 2016/101

Επίβλεψη:  
Δρ. Παπαδοπούλου  
Φωτεινή

Δηλώνω υπεύθυνα ότι το παρόν κείμενο αποτελεί προϊόν προσωπικής μελέτης και εργασίας και πως όλες οι πηγές που χρησιμοποιήθηκαν για τη συγγραφή της δηλώνονται σαφώς είτε στις παραπομπές είτε στη βιβλιογραφία. Γνωρίζω πως η λογοκλοπή είναι σοβαρότατο παράπτωμα και είμαι ενήμερος για την επέλευση των νόμιμων συνεπειών.

Οι απόψεις και τα συμπεράσματα που περιέχονται σε αυτό το έγγραφο εκφράζουν τον συγγραφέα και δεν πρέπει να ερμηνευτεί ότι αντιπροσωπεύουν τις επίσημες θέσεις του Διεθνούς Πανεπιστημίου της Ελλάδος.

*The greatest glory in living lies not in never falling,  
but in rising every time we fall.*  
— Nelson Mandela

*The best way to predict the future is to create it.*  
— Abraham Lincoln

---

## ΕΥΧΑΡΙΣΤΙΕΣ

---

Πρώτα απ' όλα, θα ήθελα να ευχαριστήσω θερμά την καθηγήτρια κ. Φωτεινή Παπαδοπούλου για το ενδιαφέρον, την καθοδήγηση, την εμπιστοσύνη που μου έδειξε και την στήριξή της καθ' όλη τη διάρκεια της συνεργασίας μας.

Θα ήθελα να εκφράσω την αγάπη και την ευγνωμοσύνη μου στους γονείς μου, Κωνσταντίνο και Χρυσή, καθώς και στην αδερφή μου, Ειρήνη, για τις θυσίες, την αγάπη, την ενθάρρυνση και την υποστήριξή τους όλα αυτά τα χρόνια.

Ένα μεγάλο ευχαριστώ οφείλω στους αγαπημένους μου φίλους, Αλέξανδρο και Σάββα, που είναι μαζί μου στα εύκολα και στα δύσκολα και με στηρίζουν ότι κι αν κάνω. Η ενθάρρυνσή τους με βοήθησε να εξελιχθώ και να γίνω καλύτερος άνθρωπος και επιστήμονας.

Επιπλέον, ευγνωμονώ συγγενείς, φίλους και γνωστούς για την όποια συνεισφορά τους τόσο στην προσωπική όσο και την επαγγελματική μου πορεία και τις όμορφες στιγμές που περάσαμε μαζί.

Τέλος, χρωστάω ένα μεγάλο ευχαριστώ στον κ. Αθανάσιο Μπόγδανο, ο οποίος με το πάθος του για την φυσική και τον τρόπο του, πίστεψε σε μένα και με ώθησε σε αυτό το ταξίδι της μάθησης.

## Περίληψη

Η μηχανική μάθηση είναι ένας ταχέως αναπτυσσόμενος τομέας στο πεδίο της τεχνητής νοημοσύνης που αλλάζει τον τρόπο με τον οποίο επεξεργαζόμαστε, αναλύουμε και κατανοούμε τα δεδομένα. Πρόκειται για μια μέθοδο διδασκαλίας των υπολογιστών να μαθαίνουν από τα δεδομένα, χωρίς να προγραμματίζονται ρητά. Οι αλγόριθμοι μηχανικής μάθησης μπορούν να εκπαιδευτούν ώστε να εντοπίζουν μοτίβα σε δεδομένα, να κάνουν προβλέψεις και να λαμβάνουν αποφάσεις. Υπάρχουν τρεις βασικοί τύποι μηχανικής μάθησης, η μάθηση με επίβλεψη, η μάθηση χωρίς επίβλεψη και η μάθηση ενίσχυσης. Η μάθηση με επίβλεψη χρησιμοποιείται όταν η σωστή έξοδος είναι ήδη γνωστή και ο αλγόριθμος εκπαιδεύεται για να βρίσκει τη σωστή έξοδο για νέες εισόδους. Αυτός είναι ο τύπος που θα χρησιμοποιηθεί στην παρούσα εργασία.

Ο καρκίνος του μαστού είναι ένας τύπος καρκίνου που αναπτύσσεται στα κύτταρα του μαστού. Είναι ο δεύτερος πιο συχνός καρκίνος μεταξύ των γυναικών παγκοσμίως, μετά τον καρκίνο του δέρματος. Η έγκαιρη ανίχνευση και θεραπεία του καρκίνου του μαστού μπορεί να βελτιώσει σημαντικά τις πιθανότητες επιβίωσης. Υπάρχουν διάφορες μέθοδοι για την ανίχνευση του καρκίνου του μαστού, όπως η μαστογραφία, ο υπέροχος και η αναρρόφηση με λεπτή βελόνα (FNA). Η FNA είναι μια διαδικασία κατά την οποία αφαιρείται ένα μικρό δείγμα ιστού του μαστού χρησιμοποιώντας μια λεπτή βελόνα. Το δείγμα εξετάζεται στη συνέχεια κάτω από μικροσκόπιο για να ελεγχθεί για καρκινικά κύτταρα.

Η παρούσα εργασία παρουσιάζει μια ολοκληρωμένη μελέτη σχετικά με την ταξινόμηση του καρκίνου του μαστού με τη χρήση μεθόδων μηχανικής μάθησης. Τα δεδομένα που χρησιμοποιήθηκαν στη μελέτη εξήχθησαν από παρακέντηση με τη μέθοδο αναρρόφησης με λεπτή βελόνα. Στη μελέτη χρησιμοποιήθηκαν συνολικά 13 αλγόριθμοι μηχανικής μάθησης, που είναι οι εξής: Gaussian Naive Bayes, Linear & Quadratic Discriminant Analysis, Ridge Classifier, k-Nearest Neighbors, Support Vector Machines, Decision Tree, Random Forest, Gradient Tree Boosting, AdaBoost & XGBoost, Stochastic Gradient Descent & Multi-Layer Perceptron. Η απόδοση κάθε αλγορίθμου αξιολογήθηκε χρησιμοποιώντας το F1-score ως κύρια μετρική, η οποία είναι ένα μέτρο της ισορροπίας μεταξύ ακρίβειας και ανάκλησης. Χρησιμοποιήθηκαν επίσης πρόσθετες μετρικές όπως η ορθότητα, η ακρίβεια και η ανάκληση.

Επειδή το σύνολο δεδομένων που χρησιμοποιήθηκε στην παρούσα μελέτη ήταν περιορισμένο, χρησιμοποιήθηκε 10-πλή διασταυρούμενη επικύρωση για την αξιολόγηση της απόδοσης των αλγορίθμων. Επιπλέον, για τη ρύθμιση των παραμέτρων των αλγορίθμων χρησιμοποιήθηκε 5-πλή εμφωλευμένη διασταυρούμενη επικύρωση. Η εργασία δίνει έμφαση στη μείωση του αριθμού των χαρακτηριστικών διατηρώντας παράλληλα υψηλή ακρίβεια στα αποτελέσματα. Χρησιμοποιήθηκαν τρία διαφορετικά σύνολα χαρακτηριστικών, μεταξύ των οποίων ένα με όλα τα χαρακτηριστικά, ένα με ένα υποσύνολο επτά χαρακτηριστικών και ένα με χαρακτηριστικά που εξήχθησαν με τη μέθοδο ανάλυσης κύριων συνιστώσων. Τα αποτελέσματα της μελέτης παρέχουν πληροφορίες σχετικά με τις πιο αποτελεσματικές και αποδοτικές μεθόδους μηχανικής μάθησης για την ταξινόμηση του καρκίνου του μαστού. Επιπλέον, η εργασία παρουσιάζει μια συζήτηση σχετικά με τις τεχνικές επιλογής χαρακτηριστικών που χρησιμοποιήθηκαν και την επίδραση της μείωσης του αριθμού των χαρακτηριστικών στην απόδοση των αλγορίθμων.

---

# Title

## Breast Cancer Classification with Machine Learning Methods

### Abstract

Machine learning is a rapidly growing area within the field of artificial intelligence that is changing the way we process, analyze and understand data. It is a method of teaching computers to learn from data, without being explicitly programmed. Machine learning algorithms can be trained to identify patterns in data, make predictions, and make decisions. There are three main types of machine learning, supervised learning, unsupervised learning, and reinforcement learning. Supervised learning is used when the correct output is already known and the algorithm is trained to find the correct output for new inputs. This is the type that is going to be used in this thesis.

Breast cancer is a type of cancer that develops in the cells of the breast. It is the second most common cancer among women worldwide, after skin cancer. Early detection and treatment of breast cancer can significantly improve the chances of survival. There are several methods for detecting breast cancer, including mammography, ultrasound, and fine-needle aspiration (FNA). FNA is a procedure in which a small sample of breast tissue is removed using a thin needle. The sample is then examined under a microscope to check for cancer cells.

This thesis presents a comprehensive study on the classification of breast cancer using machine learning methods. The data used in the study were extracted from paracentesis with the fine-needle aspiration method. A total of 13 machine learning algorithms were employed in the study, including Gaussian Naive Bayes, Linear & Quadratic Discriminant Analysis, Ridge Classifier, k-Nearest Neighbors, Support Vector Machines, Decision Tree, Random Forest, Gradient Tree Boosting, Adaboost & XGBoost, Stochastic Gradient Descent & Multi-Layer Perceptron. The performance of each algorithm was evaluated using the F1-score as the primary metric, which is a measure of the balance between precision and recall. Additional metrics such as accuracy, precision, and recall were also used.

Because the data set used in this study was limited, 10-fold cross validation was used for evaluating the performance of the algorithms. Additionally, a nested 5-fold cross validation was used for tuning the parameters of the algorithms. The thesis emphasizes on the reduction of the number of features while maintaining high accuracy in the results. Three different feature sets were used, including one with all features, one with a subset of seven features, and one with features extracted using the Principal Component Analysis method. The results of the study provide insight into the most effective and efficient machine learning methods for breast cancer classification. Additionally, the thesis presents a discussion on the feature selection techniques used and the impact of reducing the number of features on the performance of the algorithms.

Lazaros Panitsidis  
Industrial Engineering & Management Department  
International Hellenic University, Greece  
February 2023

# Περιεχόμενα

Ευχαριστίες . . . . .	iii
Περίληψη . . . . .	iv
Abstract . . . . .	v
Ακρωνύμια . . . . .	xiii
<b>1 Εισαγωγή</b>	<b>1</b>
1.1 Περιγραφή του Προβλήματος . . . . .	3
1.2 Σκοπός - Συνεισφορά της Διπλωματικής Εργασίας . . . . .	3
1.3 Διάρθρωση της Αναφοράς . . . . .	4
<b>2 Επισκόπηση της Ερευνητικής Περιοχής</b>	<b>5</b>
2.1 Πρόοδος στον κλάδο . . . . .	5
2.2 Δεδομένα . . . . .	6
2.2.1 Εισαγωγή . . . . .	6
2.2.2 Θέση του κυτταρικού πυρήνα . . . . .	7
2.2.3 Πυρηνικά χαρακτηριστικά . . . . .	9
<b>3 Μηχανική και Βαθιά Μάθηση</b>	<b>14</b>
3.1 Εισαγωγή στην Επιστήμη της Μηχανικής Μάθησης . . . . .	15
3.1.1 Βασικές κατηγορίες . . . . .	16
3.1.2 Κατηγορίες των αλγορίθμων βάση την μορφή της εξόδου . . .	18
3.1.3 Γενικές κατηγορίες επιβλεπόμενης μάθησης . . . . .	19
3.1.4 Κατηγορίες επιβλεπόμενης μάθησης για ταξινόμηση . . . . .	20
3.1.5 Αναπαράσταση δεδομένων . . . . .	22
3.2 Μέθοδοι Μηχανικής Μάθησης . . . . .	23
3.2.1 Naive Bayes . . . . .	23
3.2.2 Linear & Quadratic Discriminate Analysis . . . . .	25
3.2.3 Ridge Regression & Classification . . . . .	28
3.2.4 K-Nearest Neighbors . . . . .	29
3.2.5 Support Vector Machine . . . . .	35
3.2.6 Decision Tree . . . . .	38
3.2.7 Random Forest . . . . .	43
3.2.8 Adaptive Boosting . . . . .	45
3.2.9 Stochastic Gradient Descent . . . . .	49
3.2.10 Gradient Tree Boosting . . . . .	56
3.3 Εισαγωγή στα Νευρωνικά Δίκτυα . . . . .	61
3.3.1 Ο (τεχνητός) νευρώνας . . . . .	61

3.3.2	Συναρτήσεις Ενεργοποίησης . . . . .	64
3.3.3	Συναρτήσεις Σφάλματος/Κόστους . . . . .	69
3.3.4	Αλγόριθμος Backpropagation . . . . .	70
3.4	Perceptron Πολλαπλών Επιπέδων . . . . .	74
3.4.1	Multi-layer Perceptron . . . . .	74
3.4.2	Ταξινόμηση . . . . .	75
3.4.3	Όρος Κανονικοποίησης . . . . .	76
3.4.4	Αλγόριθμοι . . . . .	76
3.4.5	Πολυπλοκότητα . . . . .	77
3.4.6	Μαθηματική διατύπωση . . . . .	77
<b>4</b>	<b>Εργαλεία και Τεχνικές</b>	<b>79</b>
4.1	Hardware . . . . .	79
4.2	Εργαλεία Λογισμικού . . . . .	81
4.3	Μετρικές Απόδοσης . . . . .	82
4.3.1	Ορθότητα (Accuracy) . . . . .	82
4.3.2	Ακρίβεια (Precision) . . . . .	83
4.3.3	Ανάκληση (Recall) . . . . .	83
4.3.4	F1-score . . . . .	84
4.4	Γενικευμένη Αξιολόγηση . . . . .	85
4.4.1	Διασταυρούμενη Επικύρωση . . . . .	85
4.4.2	Εμφωλευμένη Διασταυρούμενη Επικύρωση . . . . .	87
4.5	Μέθοδοι Επιλογής Υπερ-Παραμέτρων . . . . .	89
4.5.1	Αναζήτηση Πλέγματος . . . . .	89
4.5.2	Τυχαιοποιημένη Αναζήτηση . . . . .	89
4.5.3	Σύγκριση . . . . .	89
4.5.4	Συμπέρασμα . . . . .	90
4.6	Ανάλυση Κύριων Συνιστωσών με Αποσύνθεση Ιδιαζουσών Τιμών . . . . .	91
4.6.1	Εισαγωγή . . . . .	91
4.6.2	Δεδομένα σε πίνακα . . . . .	91
4.6.3	Ανάλυση Κύριων Συνιστωσών . . . . .	92
4.6.4	Αποσύνθεση Ιδιαζουσών Τιμών . . . . .	94
4.6.5	Σχέση μεταξύ SVD και PCA . . . . .	96
<b>5</b>	<b>Υλοποίηση</b>	<b>98</b>
5.1	Προ-επεξεργασία δεδομένων . . . . .	99
5.1.1	Διάβασμα των δεδομένων . . . . .	99
5.1.2	”Καθαρισμός” των δεδομένων . . . . .	99
5.1.3	Υποδειγματοληψία . . . . .	100
5.2	Οπτικοποίηση . . . . .	101
5.3	Επιλογή Χαρακτηριστικών . . . . .	108
5.3.1	Επιλογή χαρακτηριστικών βάση συσχέτισης . . . . .	108
5.3.2	Μονομεταβλητή επιλογή χαρακτηριστικών . . . . .	115
5.3.3	Αναδρομική εξάλειψη χαρακτηριστικών . . . . .	115
5.3.4	Σημαντικότητα των χαρακτηριστικών . . . . .	117
5.3.5	Ελάχιστος πλεονασμός και μέγιστη συνάφεια (mRMR) . . . . .	119

## ΠΕΡΙΕΧΟΜΕΝΑ

---

5.3.6 Σύνοψη Επιλογής Χαρακτηριστικών . . . . .	120
5.4 Ανάλυση Κύριων Συνιστωσών . . . . .	121
5.5 Αποτελέσματα των αλγορίθμων . . . . .	124
<b>6 Συμπεράσματα, Προβλήματα &amp; Μελλοντικές Επεκτάσεις</b>	<b>130</b>
6.1 Γενικά Συμπεράσματα . . . . .	130
6.2 Προβλήματα . . . . .	131
6.3 Μελλοντικές Επεκτάσεις . . . . .	132
<b>Βιβλιογραφία</b>	<b>134</b>

# Κατάλογος Σχημάτων

2.1	Αρχικά κατά προσέγγιση όρια των κυτταρικών πυρήνων . . . . .	8
2.2	”Φίδια” μετά τη σύγκλιση στα όρια του κυτταρικού πυρήνα . . . . .	10
2.3	Ακτινικές γραμμές που χρησιμοποιούνται για τον υπολογισμό της ομαλότητας . . . . .	11
2.4	Χορδές που χρησιμοποιούνται για τον υπολογισμό της κοιλότητας .	12
2.5	Τμήματα που χρησιμοποιούνται στον υπολογισμό συμμετρίας . . . . .	12
2.6	Ακολουθία Μετρήσεων για υπολογισμό φράκταλ διάστασης . . . . .	13
3.1	Κλάδοι και εφαρμογές της επιστήμης της Τεχνητής Νοημοσύνης . . . . .	15
3.2	Διάγραμμα Venn των βασικών κατηγοριών μηχανικής μάθησης . . . . .	17
3.3	Κατηγορίες των αλγορίθμων βάση την μορφή της εξόδου . . . . .	18
3.4	Απεικόνιση πίσω από τον αλγόριθμο Naive Bayes . . . . .	24
3.5	Γραμμικό όριο απόφασης στον Naive Bayes . . . . .	25
3.6	Ανάλυση Γραμμικής & Τετραγωνικής Διάκρισης . . . . .	26
3.7	Συντελεστές της συνάρτησης κορυφογραμμής ως συνάρτηση κανονικοποίησης . . . . .	28
3.8	Ταξινόμηση αμφίβολου σημείου με τον αλγόριθμο k-NN . . . . .	31
3.9	Διάγραμμα Voronoi . . . . .	33
3.10	Μοναδιαίος κύβος . . . . .	34
3.11	Η κατάρα της διαστατικότητας . . . . .	35
3.12	Λειτουργικά περιθώρια των Μηχανών Διανυσμάτων Γραμμικής Διάκρισης .	36
3.13	Δυαδική ταξινόμηση χρησιμοποιώντας μη γραμμικό SVC με πυρήνα RBF . . . . .	36
3.14	Παλινδρόμηση Δέντρου Αποφάσεων . . . . .	39
3.15	Δομή Δέντρου Αποφάσεων . . . . .	43
3.16	Διαδικασία του Bagging . . . . .	44
3.17	Τυχαία Δάση . . . . .	45
3.18	Βασική Διαφορά Bagging και Boosting . . . . .	46
3.19	Διαδικασία της Ενίσχυσης . . . . .	47
3.20	Προσαρμοστικός Αλγόριθμος Ενίσχυσης . . . . .	48
3.21	Συναρτήσεις απώλειας του SGD . . . . .	51
3.22	Περιγράμματα των διαφορετικών όρων τακτοποίησης σε έναν δισδιάστατο χώρο παραμέτρων . . . . .	52
3.23	Στοχαστική Κλίση Καθόδου . . . . .	54
3.24	Βασική διαφορά XGBoost με LightGBM . . . . .	61
3.25	Βιολογικός Νευρώνας . . . . .	62
3.26	Μαθηματικό μοντέλο του νευρώνα . . . . .	63

3.27 Συνάρτηση Σιγμοειδούς συνάρτησης . . . . .	64
3.28 Συνάρτηση Υπερβολικής Εφαπτωμένης . . . . .	65
3.29 Συνάρτηση Rectified Linear Unit - ReLU . . . . .	66
3.30 Συνάρτηση Leaky ReLU . . . . .	67
3.31 Συνάρτηση Maxout . . . . .	68
3.32 Perceptron πολλαπλών στρωμάτων με 1 κρυφό στρώμα . . . . .	74
3.33 Μεταβαλλόμενη συνάρτηση απόφασης με την τιμή του alpha . . . . .	76
4.1 Τυπική ροή εργασιών διασταυρούμενης επικύρωσης . . . . .	85
4.2 Διασταυρούμενη Επικύρωση . . . . .	86
4.3 Στρωματοποιημένες κ αναδιπλώσεις . . . . .	87
4.4 Εμφωλευμένη Διασταυρούμενη Επικύρωση . . . . .	88
4.5 Διάταξη Πλέγματος και Τυχαία Διάταξη . . . . .	90
4.6 Η αρχική και ασυχέτιστη προβολή 1000 δειγμάτων που προέρχονται από μια πολυμεταβλητή Γκαουσιανή . . . . .	94
4.7 Χαρτογράφηση μοναδιαίας σφαίρας σε (υπερ)έλλειψη . . . . .	95
4.8 Συνιστώσες της αποσύνθεσης της ιδιάζουσας τιμής ενός πίνακα A .	96
4.9 Άλλαγή βάσης και μείωση διαστάσεων . . . . .	97
4.10 Προβολή των μεταβλητών στο πρώτο παραγοντικό επίπεδο . . . . .	97
5.1 Αριθμός δειγμάτων πριν και μετά την υποδειγματοληφία . . . . .	102
5.2 Κατανομές των χαρακτηριστικών . . . . .	103
5.3 Μέσες τιμές των χαρακτηριστικών . . . . .	105
5.4 Τυπικό σφάλμα των χαρακτηριστικών . . . . .	106
5.5 Ακραίες τιμές των χαρακτηριστικών . . . . .	107
5.6 Γραμμική συσχέτιση μεταξύ 2 χαρακτηριστικών . . . . .	108
5.7 Γραμμική συσχέτιση όλων των χαρακτηριστικών . . . . .	109
5.8 Διάγραμμα Venn μεταξύ Εντροπίας και Αμοιβαίας Πληροφορίας .	111
5.9 Αμοιβαία πληροφορία όλων των χαρακτηριστικών σε σχέση με τη διάγνωση . . . . .	112
5.10 Σχέση μεταξύ των χαρακτηριστικών με το μεγαλύτερο MI σκορ . .	113
5.11 Χαρακτηριστικά με αποδεκτή τιμή γραμμικής συσχέτισης . . . . .	114
5.12 Μέσο όρος των F1 σκορ του RFECV . . . . .	116
5.13 Σημαντικότητα χαρακτηριστικών βάση τυχαίων δασών . . . . .	117
5.14 Σημαντικότητα χαρακτηριστικών βάση του αλγορίθμου XGBoost .	118
5.15 Επιλογή διαστάσεων βάσει της αθροιστικής διακύμανσης . . . . .	121
5.16 Αποσυσχέτιση . . . . .	122
5.17 Εξηγούμενη διακύμανση των κύριων συνιστωσών που επιλέχθηκαν .	122
5.18 Αμοιβαία πληροφορία των κύριων συνιστωσών που επιλέχθηκαν .	123
5.19 F1-Score και ο χρόνος εκτέλεσης των αλγορίθμων στις προεπιλεγμένες παραμέτρους . . . . .	127
5.20 F1-Score και ο χρόνος εκτέλεσης των αλγορίθμων στις ρυθμισμένες παραμέτρους . . . . .	128
5.21 Η διαφορά του F1-Score και του χρόνου εκτέλεσης μεταξύ των αλγορίθμων στις ρυθμισμένες και στις προεπιλεγμένες παραμέτρους . . .	129

# Κατάλογος πινάκων

3.1 Πίνακας d - 1 παραδείγματος μοναδιαίου κύβου . . . . .	34
4.1 Βασικές προδιαγραφές του επεξεργαστή Intel® Core™ i5-8400 . . . . .	80
4.2 Βασικές προδιαγραφές της μνήμης HyperX HX426C16FB3/4 . . . . .	80
5.1 Πίνακας επιλογής χαρακτηριστικών . . . . .	120

# Κατάλογος Αλγορίθμων

3.1	AdaBoost.M1 . . . . .	49
3.2	Stochastic gradient descent v1 . . . . .	55
3.3	(“On-line”) Stochastic gradient descent v2 . . . . .	55
3.4	(Batch) Gradient Descent . . . . .	56
3.5	Minibatch (Stochastic) Gradient Descent v1 . . . . .	57
3.6	Minibatch (Stochastic) Gradient Descent v2 . . . . .	57
3.7	Perceptron Algorithm . . . . .	63
3.8	Οπισθοδιάδοση - μαθηματικός συμβολισμός . . . . .	72
3.9	Οπισθοδιάδοση - ψευδοκώδικας . . . . .	73

# Ακρωνύμια Εγγράφου

Παρακάτω παρατίθενται ορισμένα από τα πιο συχνά χρησιμοποιούμενα ακρωνύμια της παρούσας διπλωματικής εργασίας:

FNA	→ Fine Needle Aspiration
AI	→ Artificial Intelligence
ML	→ Machine Learning
WDBC	→ Wisconsin Diagnostic Breast Cancer
NB	→ Naive Bayes
LDA	→ Linear Discriminant Analysis
QDA	→ Quadratic Discriminant Analysis
KNN	→ K-Nearest Neighbors
SVM	→ Support Vector Machine
DT	→ Decision Tree
RF	→ Random Forest
SGD	→ Stochastic Gradient Descent
GBDT	→ Gradient Boosted Decision Trees
NN	→ Neural Network
ANN	→ Artificial Neural Network
MLP	→ Multilayer Perceptron
CPU	→ Central Processing Unit
GPU	→ Graphics Processing Unit
CV	→ Cross Validation
PCA	→ Principal Component Analysis
MI	→ Mutual Information
MRMR	→ Minimum Redundancy Maximum Relevance

# 1

## Εισαγωγή

Αν και τις τελευταίες δεκαετίες έχουν εμφανιστεί διάφοροι ορισμοί της Τεχνητής Νοημοσύνης (TN-AI), ο John McCarthy προσφέρει τον ακόλουθο ορισμό στο άρθρο του "What is Artificial Intelligence ?" [1]:

"Είναι η επιστήμη και η μηχανική της κατασκευής ευφυών μηχανών, ιδιαίτερα ευφυών προγραμμάτων υπολογιστών. Σχετίζεται με το παρόμοιο έργο της χρήσης υπολογιστών για την κατανόηση της ανθρώπινης νοημοσύνης, αλλά η TN δεν χρειάζεται να περιορίζεται σε μεθόδους που είναι βιολογικά παρατηρήσιμες".

Ωστόσο, δεκαετίες πριν από αυτόν τον ορισμό, η γέννηση της συζήτησης για την τεχνητή νοημοσύνη δηλώνεται από το θεμελιώδες έργο του Alan Turing, "Computing Machinery and Intelligence" [2], το οποίο δημοσιεύθηκε το 1950. Σε αυτή την εργασία, ο Τούρινγκ, που συχνά αναφέρεται ως ο "πατέρας της επιστήμης των υπολογιστών", θέτει το εξής ερώτημα: "Μπορούν οι μηχανές να σκέφτονται;". Από εκεί και πέρα, προσφέρει μια δοκιμασία, γνωστή πλέον ως "Δοκιμασία Τούρινγκ", όπου ένας άνθρωπος-ανακριτής θα προσπαθούσε να διακρίνει μεταξύ μιας απάντησης κειμένου από υπολογιστή και ενός ανθρώπου. Αν και το τεστ αυτό έχει υποστεί μεγάλη εξέταση από τη δημοσίευσή του, παραμένει ένα σημαντικό μέρος της ιστορίας της τεχνητής νοημοσύνης, καθώς και μια συνεχιζόμενη έννοια στη φιλοσοφία, καθώς χρησιμοποιεί ιδέες γύρω από τη γλωσσολογία.

Στη συνέχεια, οι Stuart Russell και Peter Norvig προχώρησαν στη δημοσίευση του βιβλίου "Artificial Intelligence: A Modern Approach" [3], το οποίο αποτελεί ένα από τα κορυφαία εγχειρίδια στη μελέτη της τεχνητής νοημοσύνης. Σε αυτό, εμβαθύνουν σε τέσσερις πιθανούς στόχους ή ορισμούς της Τεχνητής Νοημοσύνης, οι οποίοι διαφοροποιούν τα υπολογιστικά συστήματα με βάση τον ορθολογισμό και τη σκέψη έναντι της δράσης:

- Ανθρώπινη προσέγγιση:

## ΚΕΦΑΛΑΙΟ 1. ΕΙΣΑΓΩΓΗ

---

- Συστήματα που σκέφτονται όπως οι άνθρωποι
  - Συστήματα που ενεργούν όπως οι άνθρωποι
- Ιδανική προσέγγιση:
    - Συστήματα που σκέφτονται ορθολογικά
    - Συστήματα που ενεργούν ορθολογικά

Ο ορισμός του Alan Turing θα ανήκε στην κατηγορία των "συστημάτων που ενεργούν όπως οι άνθρωποι".

Στην απλούστερη μορφή της, η τεχνητή νοημοσύνη είναι ένας τομέας, ο οποίος συνδυάζει την επιστήμη των υπολογιστών και ισχυρά σύνολα δεδομένων, για να επιτρέψει την επίλυση προβλημάτων. Περιλαμβάνει επίσης τα επιμέρους πεδία της μηχανικής μάθησης και της βαθιάς μάθησης, τα οποία αναφέρονται συχνά σε συνδυασμό με την τεχνητή νοημοσύνη. Αυτοί οι κλάδοι αποτελούνται από αλγορίθμους TN, οι οποίοι επιδιώκουν να δημιουργήσουν συστήματα εμπειρογνωμόνων με στόχο την πρόβλεψη, την αυτοματοποίηση και τη βελτιστοποίηση εργασιών που ιστορικά έκαναν οι άνθρωποι, όπως η αναγνώριση ομιλίας και προσώπου, η λήψη αποφάσεων και η μετάφραση.

Υπάρχουν τρεις κύριες κατηγορίες TN:

- Τεχνητή Περιορισμένη Νοημοσύνη (Artificial Narrow Intelligence - ANI)
- Τεχνητή Γενική Νοημοσύνη (Artificial General Intelligence - AGI)
- Τεχνητή Υπερ-Νοημοσύνη (Artificial Super Intelligence - ASI)

Η ANI θεωρείται "ασθενής" TN, ενώ οι άλλοι δύο τύποι χαρακτηρίζονται ως "ισχυρή" TN. Η αδύναμη TN ορίζεται από την ικανότητά της να ολοκληρώσει μια πολύ συγκεκριμένη εργασία, όπως η νίκη σε μια παρτίδα σκάκι ή η αναγνώριση ενός συγκεκριμένου ατόμου σε μια σειρά φωτογραφιών. Καθώς προχωράμε σε ισχυρότερες μορφές TN, όπως η AGI και η ASI, η ενσωμάτωση πιο ανθρώπινων συμπεριφορών γίνεται πιο εμφανής, όπως η ικανότητα ερμηνείας του τόνου και των συναισθημάτων. Τα chatbots και οι εικονικοί βοηθοί, όπως η Siri, αγγίζουν την επιφάνεια αυτού του φαινομένου, αλλά εξακολουθούν να αποτελούν παραδείγματα ANI.

Η ισχυρή τεχνητή νοημοσύνη ορίζεται από την ικανότητά της σε σύγκριση με τον άνθρωπο. Η Τεχνητή Γενική Νοημοσύνη (AGI) θα αποδίδει στο ίδιο επίπεδο με έναν άλλο άνθρωπο, ενώ η Τεχνητή Υπερονοημοσύνη (ASI) (γνωστή και ως υπερονοημοσύνη) θα ξεπερνά τη νοημοσύνη και την ικανότητα ενός ανθρώπου. Καμία από τις δύο μορφές Ισχυρής Τεχνητής Νοημοσύνης δεν υπάρχει ακόμη, αλλά η συνεχής έρευνα στον τομέα αυτό συνεχίζεται. Δεδομένου ότι αυτός ο τομέας της Τεχνητής Νοημοσύνης εξακολουθεί να εξελίσσεται ραγδαία, το καλύτερο παράδειγμα που μπορεί να δωθεί για το πώς θα μπορούσε να μοιάζει αυτό, είναι ο χαρακτήρας της Ava στην ταινία "Ex Machina".

## 1.1. ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

---

### 1.1 ΠΕΡΙΓΡΑΦΗ ΤΟΥ ΠΡΟΒΛΗΜΑΤΟΣ

---

Ο καρκίνος είναι ένας γενικός όρος για μια μεγάλη ομάδα ασθενειών που μπορούν να προσβάλουν οποιοδήποτε μέρος του σώματος. Άλλοι όροι είναι οι κακοήθεις όγκοι και τα νεοπλάσματα [4]. Ο καρκίνος χαρακτηρίζεται από την ταχεία εξαπλωση ανώμαλων κυττάρων που ξεπερνούν τα φυσιολογικά τους όρια και στη συνέχεια εισβάλλουν σε παρακείμενα μέρη του σώματος και μπορούν να εξαπλωθούν σε άλλα όργανα. Η διαδικασία αυτή ονομάζεται μετάσταση [4]. Οι μεταστάσεις αποτελούν την κύρια αιτία θανάτων που σχετίζονται με τον καρκίνο. Ο καρκίνος είναι μια παγκόσμια θανατηφόρα ασθένεια. Το 2018, περίπου 9,6 εκατομμύρια άνθρωποι πέθαναν λόγω καρκίνου [4]. Σε παγκόσμιο επίπεδο, ένας στους έξι θανάτους προκαλείται από καρκίνο. Περίπου το 70% των θανάτων από καρκίνο συμβαίνουν σε χώρες χαμηλού και μεσαίου εισοδήματος. Οι αιτίες των θανάτων από καρκίνο περιλαμβάνουν τον δείκτη μάζας σώματος, τη χαμηλή κατανάλωση φρούτων και λαχανικών, την έλλειψη σωματικής δραστηριότητας, τη χρήση καπνού και τη χρήση αλκοόλ. Η χρήση καπνού είναι ο σημαντικότερος παράγοντας κινδύνου για τον καρκίνο και ευθύνεται για το 22% περίπου των θανάτων από καρκίνο [5].

Ο καρκίνος είναι ένας τύπος ασθένειας που προκαλείται από την ανεξέλεγκτη ανάπτυξη κυττάρων στο σώμα. Συχνά αναφέρεται με το όνομα της δομής στην οποία η καρκινική νόσος είναι αποτελεσματική στο σώμα. Ο καρκίνος του μαστού στις γυναίκες είναι ένας τύπος καρκίνου με πολύ υψηλό ποσοστό θνησιμότητας. Στον καρκίνο του μαστού τα ταχέως διαιρούμενα κύτταρα σχηματίζουν μάζες του μαστού. Οι μάζες αυτές ονομάζονται όγκοι. Οι όγκοι χωρίζονται σε δύο ομάδες ως καλοήθεις και κακοήθεις. Οι κακοήθεις όγκοι διεισδύουν στους υγιείς ιστούς του σώματος και τους βλάπτουν. Τα επιβλαβή κύτταρα στο εσωτερικό του όγκου μπορούν να εξαπλωθούν σε διάφορα όργανα του σώματος και να τα βλάψουν. Ως καρκίνος του μαστού νοείται ένας κακοήθης όγκος που τοποθετείται στο μαστό.

Ο καρκίνος του μαστού είναι ο πιο επικίνδυνος καρκίνος που προκαλεί το θάνατο σε γυναίκες ηλικίας 40-55 ετών. Σύμφωνα με τον Παγκόσμιο Οργανισμό Γιγείας, 2,09 εκατομμύρια άνθρωποι διαγνωσκούνται με καρκίνο του μαστού κάθε χρόνο [4]. Ως εκ τούτου, έχουν διεξαχθεί πολλές μελέτες για την έγκαιρη διάγνωση του καρκίνου, ο οποίος προκαλεί τόσο επιβλαβείς επιπτώσεις στον άνθρωπο. Στην παρούσα μελέτη, επιχειρήθηκε η διάγνωση του καρκίνου με τη χρήση των δεδομένων για τον καρκίνο του μαστού του Wisconsin Diagnostic Breast Cancer (WDBC) [6].

### 1.2 ΣΚΟΠΟΣ - ΣΥΝΕΙΣΦΟΡΑ ΤΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

---

Στην παρούσα διπλωματική εργασία μελετάται η χρήση διάφορων αλγορίθμων μηχανικής μάθησης, προκειμένου να διαγνωσθεί ο καρκίνος του μαστού από δεδομένα δειγμάτων βιοφίας με την μεγαλύτερη δυνατή ακρίβεια.

Ένας σημαντικός παράγοντας που λαμβάνεται υπόψη, είναι ο χρόνος εκτέλεσης των αλγορίθμων. Αυτό, γίνεται προσπάθεια να επιτευχθεί επιλέγοντας τα πιο χρήσιμα χαρακτηριστικά και μειώνοντας τις διαστάσεις των δεδομένων.

Λόγω περιορισμένου hardware, επιλέχθηκε ένας μικρός όγκος δεδομένων. Εξαιτίας αυτού, χρησιμοποιούνται τεχνικές που αξιοποιούν τα δεδομένα με τον καλύτερο δυνατό τρόπο για την εκπαίδευση αλλά και τον έλεγχο της απόδοσης των αλγορίθμων.

### 1.3 ΔΙΑΡΘΡΩΣΗ ΤΗΣ ΑΝΑΦΟΡΑΣ

---

Η διάρθρωση της παρούσας διπλωματικής εργασίας είναι η εξής:

- **Κεφάλαιο 2:** Γίνεται ανασκόπηση της ερευνητικής περιοχής που αφορά την διάγνωση του καρκίνου του μαστού με χρήση τεχνικών μηχανικής μάθησης. Επίσης, περιγράφονται τα δεδομένα που θα χρησιμοποιηθούν.
- **Κεφάλαιο 3:** Αρχικά, γίνεται μια εισαγωγή στην επιστήμη της μηχανικής μάθησης και στη συνέχεια περιγράφεται το βασικό θεωρητικό κομμάτι όλων των αλγορίθμων που θα χρησιμοποιηθούν.
- **Κεφάλαιο 4:** Παρουσιάζονται τα εργαλεία που θα χρησιμοποιηθούν και περιγράφεται η θεωρία πίσω από τις τεχνικές αξιολόγησης και επιλογής χαρακτηριστικών.
- **Κεφάλαιο 5:** Αναλυτική περιγραφή της προ-επεξεργασίας των δεδομένων και της διαδικασίας επιλογής χαρακτηριστικών. Στη συνέχεια παρουσιάζονται τα αποτελέσματα των αλγορίθμων.
- **Κεφάλαιο 6:** Αναφέρονται τα τελικά συμπεράσματα. Επίσης, αναλύονται τα προβλήματα που προέκυψαν και προτείνονται θέματα για μελλοντική μελέτη, αλλαγές και επεκτάσεις.

# 2

## Επισκόπηση της Ερευνητικής Περιοχής

### 2.1 ΠΡΟΟΔΟΣ ΣΤΟΝ ΚΛΑΔΟ

---

Ο καρκίνος του μαστού αποτελεί κύρια αιτία θανάτου μεταξύ των γυναικών παγκοσμίως. Η έγκαιρη ανίχνευση του καρκίνου του μαστού μπορεί να βελτιώσει σημαντικά τις πιθανότητες επιτυχούς θεραπείας και επιβίωσης. Μια κοινή μέθοδος για την ανίχνευση του καρκίνου του μαστού είναι η αναρρόφηση με λεπτή βελόνα (FNA), η οποία περιλαμβάνει την εισαγωγή μιας λεπτής βελόνας στον ιστό του μαστού για τη λήψη ενός μικρού δείγματος για εργαστηριακή ανάλυση.

Οι μέθοδοι μηχανικής μάθησης μπορούν να χρησιμοποιηθούν για την ανάλυση δεδομένων που συλλέγονται από δείγματα FNA για την ανίχνευση του καρκίνου του μαστού. Οι μέθοδοι αυτές περιλαμβάνουν την εκπαίδευση αλγορίθμων σε ένα μεγάλο σύνολο δεδομένων από δείγματα FNA, ορισμένα από τα οποία χαρακτηρίζονται ως καρκινικά και άλλα ως καλοήθη. Ο αλγόριθμος μαθαίνει να εντοπίζει μοτίβα στα δεδομένα που σχετίζονται με καρκινικό ιστό. Αφού εκπαιδευτεί, ο αλγόριθμος μπορεί στη συνέχεια να εφαρμοστεί σε νέα δείγματα FNA για να προβλέψει εάν περιέχουν ή όχι καρκινικό ιστό.

Υπάρχουν αρκετοί διαφορετικοί τύποι μεθόδων μηχανικής μάθησης που μπορούν να χρησιμοποιηθούν για την ανίχνευση καρκίνου του μαστού, συμπεριλαμβανομένων των δέντρων απόφασης, των μηχανών διανυσμάτων υποστήριξης και των νευρωνικών δικτύων. Κάθε μία από αυτές τις μεθόδους έχει τα δικά της πλεονεκτήματα και περιορισμούς και η επιλογή της μεθόδου που θα χρησιμοποιηθεί εξαρτάται από τα συγκεκριμένα χαρακτηριστικά των δεδομένων και τους στόχους της ανάλυσης.

Ένα από τα βασικά πλεονεκτήματα της χρήσης της μηχανικής μάθησης για την ανίχνευση του καρκίνου του μαστού είναι ότι μπορεί να συμβάλει στη βελτίωση της ακρίβειας και της αξιοπιστίας της διάγνωσης. Με την αυτοματοποίηση της ανάλυσης των δειγμάτων FNA, η μηχανική μάθηση μπορεί να συμβάλει στη μείωση της πιθανότητας ανθρώπινου σφάλματος και υποκειμενικότητας στη διαδικασία

## ΚΕΦΑΛΑΙΟ 2. ΕΠΙΣΚΟΠΗΣΗ ΤΗΣ ΕΡΕΥΝΗΤΙΚΗΣ ΠΕΡΙΟΧΗΣ

---

διάγνωσης. Επιπλέον, η μηχανική μάθηση μπορεί να βοηθήσει στον εντοπισμό λεπτών μοτίβων στα δεδομένα που μπορεί να είναι δύσκολο να εντοπιστούν από τους ανθρώπινους αναλυτές, οδηγώντας ενδεχομένως σε ακριβέστερη και έγκαιρη ανίχνευση του καρκίνου του μαστού.

Υπάρχουν πολλές μελέτες που έχουν διεξαχθεί στο σύνολο δεδομένων για τον καρκίνο του μαστού WDBC και η επιτυχία τους είναι αρκετά μεγάλη. Οι Quinlan et al. πραγματοποίησαν την πρώτη από αυτές τις μελέτες. Στη μελέτη χρησιμοποιήθηκε δέντρο απόφασης C4.5 για την ταξινόμηση και επιτεύχθηκε επιτυχία 94,74% [7]. Ο ασαφής γενετικός αλγόριθμος χρησιμοποιήθηκε σε μελέτη του Pena Reyes και επιτεύχθηκε επιτυχία 97,36% [8]. Σε μια άλλη μελέτη, οι Nauck και Kruse πέτυχαν επιτυχία 95% χρησιμοποιώντας ασαφείς νευρώνες [9]. Στη μελέτη του Setiono που χρησιμοποίησε νευρωνικά δίκτυα πρόωσης, υπήρξε επιτυχία 98,1% [10]. Σε μια μελέτη των Albrecht et al. χρησιμοποιήθηκε η μέθοδος νευρωνικών δικτύων perceptron και επιτεύχθηκε ποσοστό επιτυχίας 98,8% [11]. Σε μια μελέτη με τη χρήση ασαφούς μεθόδου ομαδοποίησης από τους Abonyi και Szeifert, επιτεύχθηκε επιτυχία 95,57% [12]. Οι Kiyani et al. χρησιμοποιώντας νευρωνικά δίκτυα γενικευμένης παλινδρόμησης πέτυχαν επιτυχία 98,8% [13]. Η μελέτη των Polat και Güneş πέτυχε ποσοστό επιτυχίας 98% [14]. Το 2007, χρησιμοποιήθηκαν από τον Übeyli το νευρωνικό δίκτυο multilayer perceptron (MLPNN), το συνδυασμένο νευρωνικό δίκτυο (CNN), το πιθανοτικό νευρωνικό δίκτυο (PNN), το επαναλαμβανόμενο νευρωνικό δίκτυο (RNN) και η μηχανή διανυσμάτων υποστήριξης (SVM). Σε αυτή τη μελέτη, η μεγαλύτερη επιτυχία επιτεύχθηκε με τη χρήση μηχανών διανυσμάτων υποστήριξης με ποσοστό 99,54% [15]. Στη μελέτη που χρησιμοποιήσει ο Akay μαζί με την επιλογή χαρακτηριστικών και τις μηχανές διανυσμάτων υποστήριξης, επιτεύχθηκε επιτυχία 99,5% [16]. Οι Peng et al. πέτυχαν ποσοστό επιτυχίας 99,50% χρησιμοποιώντας τις μεθόδους φίλτρου και περιτύλιξης [17]. Το 2012, οι Salama et al. πραγματοποίησαν με τις μηχανές διανυσμάτων υποστήριξης, επιτεύχθηκε διαγνωστική επιτυχία 97,71% [18].

## 2.2 ΔΕΔΟΜΕΝΑ

---

Τα αριθμητικά δεδομένα που θα χρησιμοποιηθούν σε αυτήν την εργασία, εξήχθησαν και χρησιμοποιήθηκαν για πρώτη φορά από το πανεπιστήμιο του Wisconsin [19] το 1993.

### 2.2.1 Εισαγωγή

Η διάγνωση των όγκων του μαστού παραδοσιακά διενεργείται με πλήρη βιοψία, μια επεμβατική χειρουργική επέμβαση. Η παρακέντηση με λεπτή βελόνα (FNA) παρέχει έναν τρόπο εξέτασης μικρής ποσότητας ιστού από τον όγκο. Ωστόσο, η διάγνωση με αυτή τη διαδικασία δεν ήταν πάντοτε επιτυχής. Με την προσεκτική εξέταση τόσο των χαρακτηριστικών των μεμονωμένων κυττάρων όσο και των σημαντικών συμφραζόμενων χαρακτηριστικών, όπως το μέγεθος των κυτταρικών σβόλων, οι γιατροί σε ορισμένα εξειδικευμένα ιδρύματα μπόρεσαν να κάνουν επιτυχείς διαγνώσεις χρησιμοποιώντας FNA. Ωστόσο, πολλά διαφορετικά χαρακτηριστικά πι-

στεύεται ότι συσχετίζονται με κακοήθεια και η διαδικασία παραμένει εξαιρετικά υποκειμενική, ανάλογα με την ικανότητα και την εμπειρία του γιατρού. Προκειμένου να αυξηθεί η ταχύτητα, η ορθότητα και η αντικειμενικότητα της διαδικασίας διάγνωσης, χρησιμοποιήθηκαν τεχνικές επεξεργασίας εικόνας και μηχανικής μάθησης.

### 2.2.2 Θέση του κυτταρικού πυρήνα

#### Προετοιμασία εικόνας

Η διαδικασία διάγνωσης ξεκινά με τη λήψη μιας μικρής σταγόνας υγρού από έναν όγκο του μαστού χρησιμοποιώντας μια λεπτή βελόνα. Το αναρροφούμενο υλικό στη συνέχεια εξάγεται σε γυάλινη πλάκα και χρωματίζεται. Η εικόνα για φηφιακή ανάλυση δημιουργείται από μια έγχρωμη βιντεοκάμερα JVC TK-1070U τοποθετημένη πάνω σε μικροσκόπιο Olympus και η εικόνα προβάλλεται στην κάμερα με αντικειμενικό φακό 63x και 2,5x οπτικό. Η εικόνα λαμβάνεται από μια έγχρωμη πλακέτα συλλογής καρέ ComputerEyes/RT (Digital Vision, Inc., Dedham MA 02026) ως αρχείο Targa 512x480, με 8-bit ανά pixel.

#### Διεπαφή χρήστη

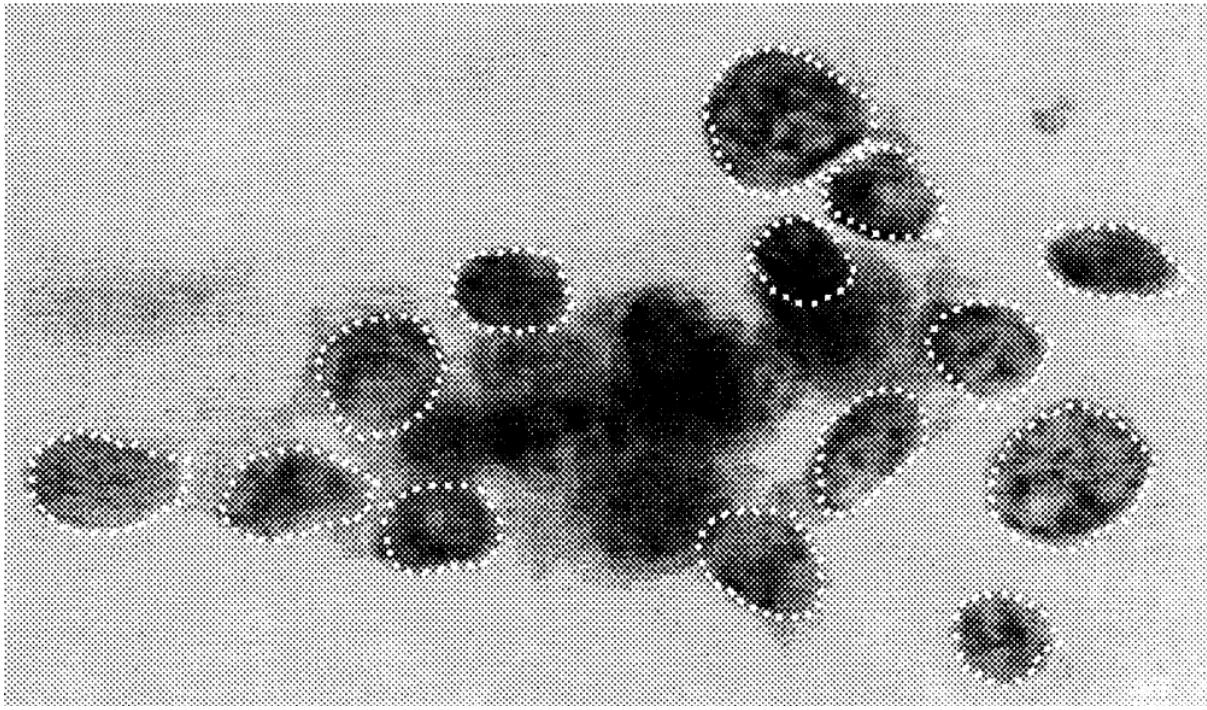
Το πρώτο βήμα για την επιτυχή ανάλυση της φηφιακής εικόνας είναι να καθοριστεί μια ακριβής θέση για κάθε όριο του πυρήνα του κυττάρου. Αναπτύχθηκε μια γραφική διεπαφή χρήστη που επιτρέπει στον χρήστη να εισάγει κατά προσέγγιση αρχικά όρια αρκετών πυρήνων για να παρέχει ένα αντιπροσωπευτικό δείγμα. Η διεπαφή αναπτύχθηκε χρησιμοποιώντας το σύστημα παραθύρων X και το σετ γραφικών στοιχείων Athena σε ένα DECstation 3100. Η ανίχνευση ενός πρόχειρου περιγράμματος ορισμένων ορατών κυτταρικών πυρήνων γίνεται με το ποντίκι. Αυτά τα περιγράμματα φαίνονται στο σχήμα 2.1.

#### Περίγραμμα "φιδιού"

Ξεκινώντας με ένα κατά προσέγγιση όριο που ορίζεται από τον χρήστη ως αρχικοποίηση, το πραγματικό όριο του πυρήνα του κυττάρου εντοπίζεται από ένα μοντέλο ενεργού περιγράμματος γνωστό στη βιβλιογραφία ως "φίδι". Ένα φίδι είναι μια παραμορφώσιμη λωρίδα που επιδιώκει να ελαχιστοποιήσει μια ενεργειακή συνάρτηση που ορίζεται στο μήκος του τόξου μιας κλειστής καμπύλης. Η ενεργειακή συνάρτηση ορίζεται με τέτοιο τρόπο ώστε η ελαχιστη τιμή εμφανίζεται όταν η καμπύλη αντιστοιχεί με ακρίβεια στο όριο ενός πυρήνα κυττάρου. Για να επιτευχθεί αυτό, η ενεργειακή συνάρτηση που πρέπει να ελαχιστοποιηθεί ορίζεται ως η ακόλουθη συνάρτηση μήκους τόξου  $s$ :

$$E = \int_s (\alpha E_{cont}(s) + \beta E_{curv}(s) + \gamma E_{image}(s)) ds$$

Εδώ το  $E$  αντιπροσωπεύει τη συνολική ενέργεια που είναι ενσωματωμένη στο μήκος του τόξου του φιδιού. Ο υπολογισμός ενέργειας είναι ένα σταθμισμένο άθροισμα ενεργειακών όρων  $E_{cont}$ ,  $E_{curv}$  και  $E_{image}$  με αντίστοιχα βάρη  $\alpha$ ,  $\beta$  και  $\gamma$ . Για να



Σχήμα 2.1: Αρχικά κατά προσέγγιση όρια των κυτταρικών πυρήνων [19]

Ο χρήστης σχεδιάζει πρώτα ένα πρόχειρο αρχικό περίγραμμα ορισμένων ορίων του κυτταρικού πυρήνα. Κάθε περίγραμμα χρησιμεύει ως η αρχική θέση για μια παραμορφωμένη λωρίδα που συγκλίνει σε ένα ακριβές όριο του πυρήνα.

απλοποιηθεί η απαραίτητη επεξεργασία, η ενεργειακή συνάρτηση υπολογίζεται σε έναν αριθμό διακριτών σημείων κατά μήκος της καμπύλης και το άθροισμα αυτών των τιμών ελαχιστοποιείται. Οι όροι ενέργειας των συστατικών μετρούν τις ακόλουθες ποσότητες:

- **Συνέχεια  $E_{cont}$**

Αυτός ο όρος έχει κατασκευαστεί για να τιμωρεί τις ασυνέχειες στην καμπύλη. Στη διακριτή περίπτωση, αυτός ο όρος μετρά πόσο ομοιόμορφα απέχουν τα σημεία του "φιδιού". Ας σημειωθεί ότι αυτή είναι μια γεωμετρική ιδιότητα του ίδιου του φιδιού και δεν εξαρτάται από το όριο του πυρήνα που προσδιορίζεται. Η απόσταση από ένα σημείο φιδιού σε έναν από τους γείτονές του βρίσκεται και συγκρίνεται με τη μέση απόσταση μεταξύ γειτονικών σημείων. Το μέγεθος αυτής της διαφοράς είναι τότε  $E_{cont}$ .

- **Καμπυλότητα  $E_{curv}$**

Αυτός ο γεωμετρικός όρος μετρά τις ασυνέχειες στην καμπυλότητα του φιδιού. Οι πυρήνες των κυττάρων είναι λίγο πολύ ελλειφοειδείς. Ως εκ τούτου, σημεία με ασυνήθιστα υψηλή ή χαμηλή καμπυλότητα, σε σύγκριση με έναν κύκλο, τιμωρούνται. Σύμφωνα με αυτή τη γνώση για το πυρηνικό σχήμα, υιοθετήθηκε η ακόλουθη μέθοδος. Πρώτον, βρίσκεται το «κέντρο» του φιδιού (κέντρο μάζας των σημείων του φιδιού). Η απόσταση από ένα σημείο φιδιού στο κέντρο (δηλαδή, μήκος ακτινικής γραμμής) συγκρίνεται στη συνέχεια με

τον μέσο όρο τέτοιων αποστάσεων σε μια "γειτονιά" του σημείου. Το μέγεθος της διαφοράς είναι αυτός ο ενεργειακός όρος  $E_{curv}$ .

- **Εικόνα  $E_{image}$**

Αυτός είναι ο μόνος όρος που συνδέει την απόδοση του φιδιού με την υποκείμενη εικόνα. Η  $E_{image}$  μετρά την ασυνέχεια σε γκρι κλίμακα κατά μήκος του φιδιού. Για να ποσοτικοποιηθεί αυτή η ασυνέχεια, περιστρέφεται η περιοχή της εικόνας που αντιστοιχεί στο σημείο του φιδιού με έναν ανιχνευτή άκρων Sobel και παρατηρείται το μέγεθος της άκρης που προκύπτει. Αυτός ο όρος προσαρμόζεται εκμεταλλευόμενος το γεγονός ότι οι πυρήνες των κυττάρων είναι γενικά πιο σκούροι από το περιβάλλον υλικό. Ως εκ τούτου, το πρότυπο ανίχνευσης ακμών περιστρέφεται έτσι ώστε η αναμενόμενη ακμή να είναι κάθετη στην ακτινική γραμμή του πυρήνα σε αυτό το σημείο. Για παράδειγμα, για ένα σημείο φιδιού ακριβώς πάνω από το κέντρο του πυρήνα, το πρότυπο άκρων

1	2	1
0	0	0
-1	-2	-1

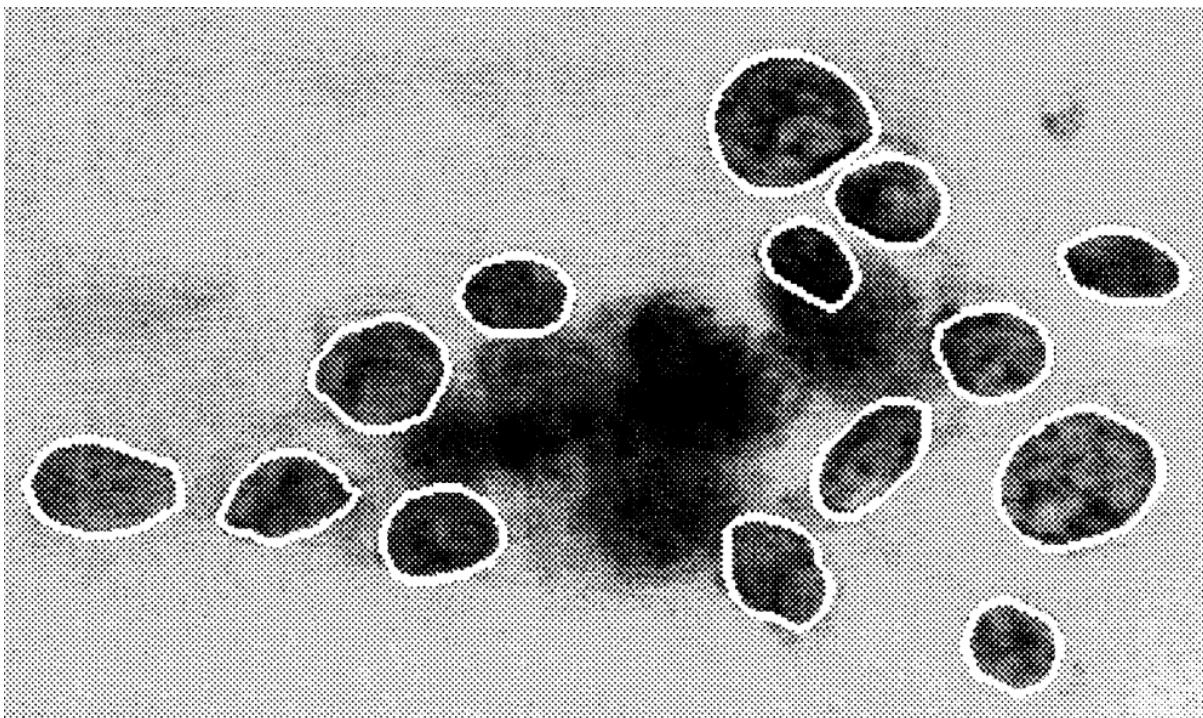
θα εφαρμοζόταν. Με αυτόν τον τρόπο, οι ασυνέχειες γκρι κλίμακας που είναι κάθετες στην ακτινική γραμμή παράγουν την υψηλότερη βαθμολογία ακμών. Η  $E_{image}$  ορίστηκε έτσι ώστε μια απότομη ασυνέχεια ελαχιστοποιεί την ενεργειακή τιμή.

Τα βάρη  $\alpha, \beta$  και  $\gamma$  είναι εμπειρικά παραγόμενες σταθερές. Για καλύτερη απόδοση σε αυτές τις εικόνες, το  $\gamma$  έχει οριστεί κάπως υψηλότερα από τις άλλες για να διασφαλιστεί ότι το φίδι συγχλίνει σε οποιοδήποτε ορατό όριο. Ο όρος καμπυλότητα καθορίζει το σχήμα του φιδιού σε περιπτώσεις χαμηλής αντίθεσης ή μερικής απόφραξης. Ο όρος της συνέχειας δεν καθορίζει το σχήμα, αλλά εμποδίζει τα σημεία φιδιού να συσσωρεύονται κοντά σε περιοχές με την πιο έντονη αντίθεση της κλίμακας του γκρι.

Για τον έλεγχο του χρόνου υπολογισμού, η βέλτιστη τοπική τιμή της ενεργειακής συνάρτησης προσεγγίζεται χρησιμοποιώντας έναν άπληστο αλγόριθμο. Εάν η τιμή της συνάρτησης σε ένα συγκεκριμένο σημείο φιδιού μπορεί να μειωθεί μετακινώντας το σημείο σε ένα γειτονικό pixel, τότε μετακινείται, επηρεάζοντας έτσι πιθανώς τον υπολογισμό της ενέργειας σε άλλα σημεία. Η διαδικασία επαναλαμβάνεται για κάθε σημείο μέχρις ότου όλα τα σημεία να εγκατασταθούν σε ένα τοπικό ελάχιστο της συνάρτησης ενέργειας. Τα αποτελέσματα μιας τυπικής εικόνας φαίνονται στο σχήμα 2.2.

### 2.2.3 Πυρηνικά χαρακτηριστικά

Το διαγνωστικό σύστημα υπολογιστικής όρασης εξάγει δέκα διαφορετικά χαρακτηριστικά από τα όρια των πυρήνων των κυττάρων που δημιουργούνται από



Σχήμα 2.2: "Φίδια" μετά τη σύγκλιση στα όρια του κυτταρικού πυρήνα [19]  
Αυτά τα περιγράμματα είναι η τελική αναπαράσταση των ορίων των κυτταρικών πυρήνων αφού ο χρήστης είναι ικανοποιημένος με τη σύγκλιση των φιδιών. Αυτή η διαδραστική διαδικασία διαρκεί περίπου δύο έως πέντε λεπτά.

φίδια. Όλα τα χαρακτηριστικά είναι αριθμητικά μοντελοποιημένα έτσι ώστε οι μεγαλύτερες τιμές να υποδεικνύουν συνήθως υψηλότερη πιθανότητα κακοήθειας. Τα εξαγόμενα χαρακτηριστικά είναι τα εξής.

#### 1. Ακτίνα

Η ακτίνα ενός μεμονωμένου πυρήνα μετριέται με τον μέσο όρο των μήκους των τμημάτων της ακτινικής γραμμής που ορίζονται από το κέντρο του φιδιού και τα μεμονωμένα σημεία φιδιού.

#### 2. Περίμετρος

Η συνολική απόσταση μεταξύ των σημείων του φιδιού αποτελεί την πυρηνική περίμετρο

#### 3. Εμβαδόν

Η πυρηνική περιοχή μετριέται απλά μετρώντας τον αριθμό των εικονοστοιχείων στο εσωτερικό του φιδιού και προσθέτοντας το μισό από τα pixel στην περίμετρο.

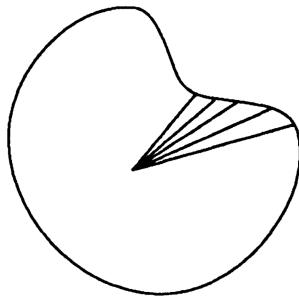
#### 4. Συμπαγής χώρος

Η περίμετρος και το εμβαδόν συνδυάζονται για να δώσουν ένα μέτρο το πόσο συμπαγής είναι ο χώρος των κυτταρικών πυρήνων χρησιμοποιώντας

τον τύπο περίμετρος<sup>2</sup>/εμβαδόν. Αυτός ο αδιάστατος αριθμός ελαχιστοποιείται από έναν κυκλικό δίσκο και αυξάνεται με την ανωμαλία του ορίου. Ωστόσο, αυτό το μέτρο σχήματος αυξάνεται επίσης για τους επιμήκεις κυτταρικούς πυρήνες, οι οποίοι δεν υποδηλώνουν απαραίτητα αυξημένη πιθανότητα κακοήθειας. Το χαρακτηριστικό είναι επίσης πολωμένο προς τα πάνω για μικρά κελιά λόγω της μειωμένης ακρίβειας που επιβάλλεται από την ψηφιοποίηση του δείγματος. Αντισταθμίζουμε το γεγονός ότι καμία μεμονωμένη μέτρηση σχήματος δεν φαίνεται να συλλαμβάνει την ιδέα του ακανόνιστου» χρησιμοποιώντας πολλά διαφορετικά χαρακτηριστικά σχήματος.

### 5. Ομαλότητα

Η ομαλότητα ενός πυρηνικού περιγράμματος ποσοτικοποιείται με τη μέτρηση της διαφοράς μεταξύ του μήκους μιας ακτινικής γραμμής και του μέσου μήκους των γραμμών που την περιβάλλουν. Αυτό είναι παρόμοιο με τον υπολογισμό της ενέργειας καμπυλότητας στα φίδια.



Σχήμα 2.3: Ακτινικές γραμμές που χρησιμοποιούνται για τον υπολογισμό της ομαλότητας [19]

### 6. Κοιλότητα

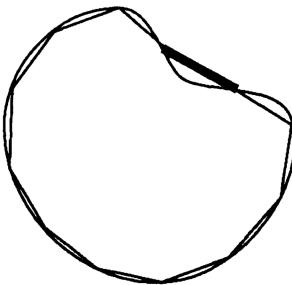
Σε μια περαιτέρω προσπάθεια να συλληφθούν πληροφορίες του σχήματος, μετριέται ο αριθμός και η σοβαρότητα των κοιλοτήτων ή των εσοχών σε έναν πυρήνα κυττάρου. Σχεδιάζονται συγχορδίες μεταξύ μη γειτονικών σημείων φιδιού και μετριέται η έκταση στην οποία το πραγματικό όριο του πυρήνα βρίσκεται στο εσωτερικό κάθε χορδής (βλ. Εικόνα 4). Αυτή η παράμετρος επηρεάζεται σε μεγάλο βαθμό από το μήκος αυτών των συγχορδιών, καθώς οι μικρότερες συγχορδίες αποτυπώνουν καλύτερα τις μικρές κοιλότητες. Επιλέχθηκε να τονιστούν οι μικρές εσοχές, καθώς οι μεγαλύτερες ανωμαλίες σχήματος αποτυπώνονται από άλλα χαρακτηριστικά.

### 7. Κοίλα σημεία

Αυτό το χαρακτηριστικό είναι παρόμοιο με την κοιλότητα αλλά μετρά μόνο τον αριθμό, και όχι το μέγεθος, των κοιλοτήτων του περιγράμματος.

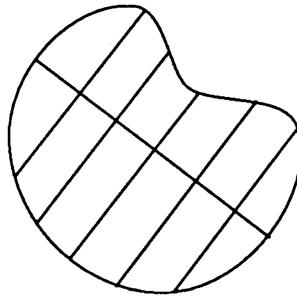
### 8. Συμμετρία

Για να μετρηθεί η συμμετρία, βρίσκεται ο κύριος άξονας ή η μεγαλύτερη χορδή που διασχίζει το κέντρο. Στη συνέχεια μετριέται η διαφορά μήκους μεταξύ



Σχήμα 2.4: Χορδές που χρησιμοποιούνται για τον υπολογισμό της κοιλότητας [19]

των γραμμών που είναι κάθετες στον κύριο άξονα στο όριο του κελιού και στις δύο κατευθύνσεις (εικόνα 5). Ιδιαίτερη προσοχή λαμβάνεται υπόψη για περιπτώσεις όπου ο κύριος άξονας κόβει το όριο του κελιού λόγω κοιλότητας.



Σχήμα 2.5: Τμήματα που χρησιμοποιούνται στον υπολογισμό συμμετρίας [19]

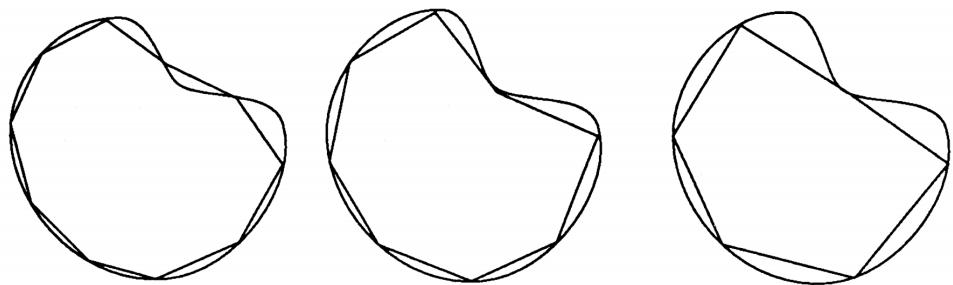
### 9. Διάσταση φράκταλ

Η φράκταλ διάσταση ενός κυττάρου προσεγγίζεται χρησιμοποιώντας την «προσέγγιση της ακτογραμμής» που περιγράφεται από τον Mandelbrot. Η περίμετρος του πυρήνα μετριέται χρησιμοποιώντας ολοένα και μεγαλύτερους «χάρακες». Καθώς αυξάνεται το μέγεθος του χάρακα, μειώνοντας την ακρίβεια της μέτρησης, η παρατηρούμενη περίμετρος μειώνεται (σχήμα 6). Η σχεδίαση αυτών των τιμών σε μια λογαριθμική κλίμακα και η μέτρηση της καθοδικής κλίσης δίνει (το αρνητικό του) μια προσέγγιση στη διάσταση φράκταλ. Όπως συμβαίνει με όλα τα χαρακτηριστικά σχήματος, μια υψηλότερη τιμή αντιστοιχεί σε λιγότερο κανονικό περίγραμμα και επομένως σε μεγαλύτερη πιθανότητα κακοήθειας.

### 10. Υφή

Η υφή του πυρήνα του κυττάρου μετριέται με την εύρεση της διακύμανσης των εντάσεων της κλίμακας του γκρι στα περιεχόμενα εικονοστοιχεία.

Όλα τα χαρακτηριστικά σχήματος επαληθεύτηκαν χρησιμοποιώντας εξιδανικευμένα φανταστικά κύτταρα. Φάνηκε να αυξάνονται καθώς τα όρια έγιναν λιγότερο τακτικά και να μην συσχετίζονται σε μεγάλο βαθμό με το μέγεθος του περιγράμματος.



Σχήμα 2.6: Ακολουθία Μετρήσεων για υπολογισμό φράκταλ διάστασης [19]

Η μέση τιμή, η ακραία (μεγαλύτερη) τιμή και το τυπικό σφάλμα κάθε χαρακτηριστικού υπολογίζονται για κάθε εικόνα. Οι ακραίες τιμές είναι οι πιο διαισθητικά χρήσιμες για το υπό εξέταση πρόβλημα, καθώς μόνο λίγα κακοήθη κύτταρα μπορεί να εμφανιστούν σε ένα δεδομένο δείγμα.

# 3

## Μηχανική και Βαθιά Μάθηση

Η *Μηχανική Μάθηση* (Machine Learning - ML) είναι ένας κλάδος της τεχνητής νοημοσύνης (AI) που επιτρέπει στους υπολογιστές να "αυτο-μαθαίνουν" από δεδομένα εκπαίδευσης και να βελτιώνονται με την πάροδο του χρόνου, χωρίς να προγραμματίζονται ρητά. Οι αλγόριθμοι μηχανικής μάθησης είναι σε θέση να εντοπίζουν μοτίβα στα δεδομένα και να μαθαίνουν από αυτά, προκειμένου να κάνουν τις δικές τους προβλέψεις. Εν ολίγοις, οι αλγόριθμοι και τα μοντέλα μηχανικής μάθησης μαθαίνουν μέσω της εμπειρίας.

Στον παραδοσιακό προγραμματισμό, ένας μηχανικός υπολογιστών γράφει μια σειρά από οδηγίες που καθοδηγούν έναν υπολογιστή πώς να μετατρέψει τα δεδομένα εισόδου σε μια επιθυμητή έξοδο. Οι οδηγίες βασίζονται ως επί το πλείστον σε μια δομή IF-THEN: όταν πληρούνται ορισμένες προϋποθέσεις, το πρόγραμμα εκτελεί μια συγκεκριμένη ενέργεια.

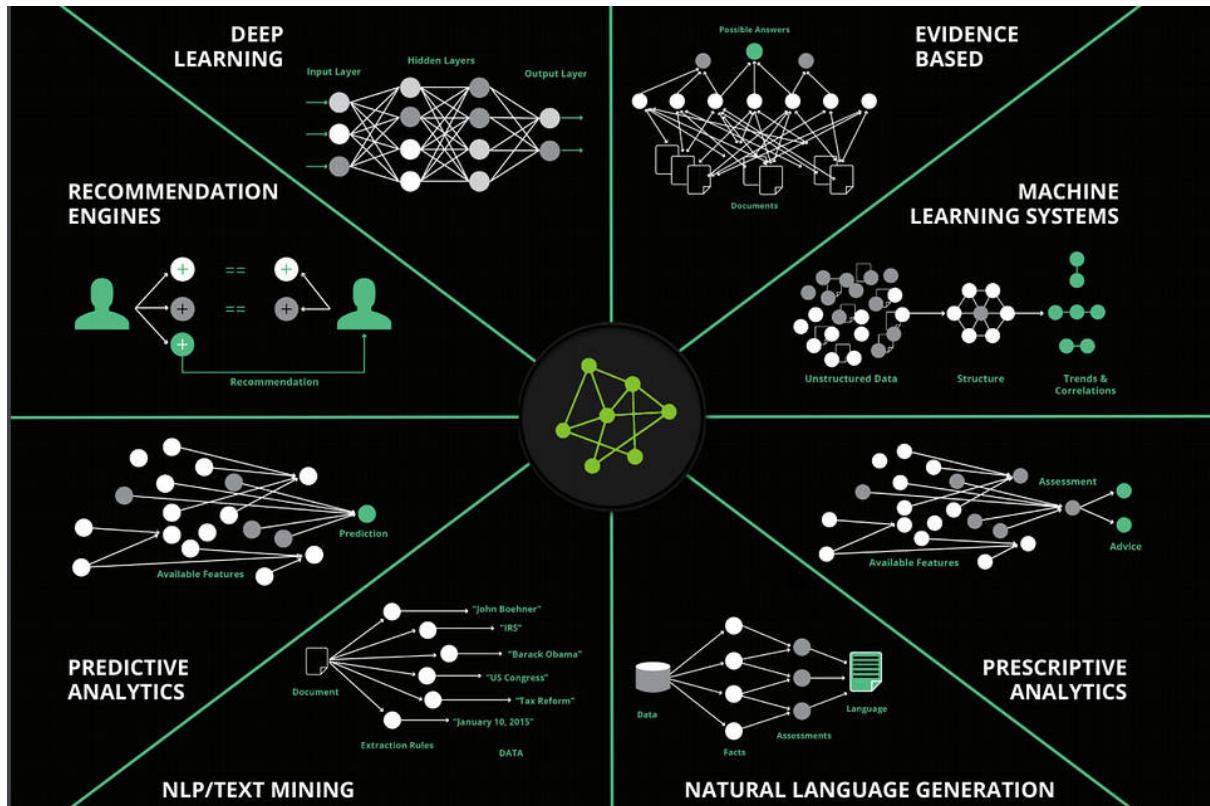
Η μηχανική μάθηση, από την άλλη πλευρά, είναι μια αυτοματοποιημένη διαδικασία που επιτρέπει στις μηχανές να επιλύουν προβλήματα με ελάχιστη ή καθόλου ανθρώπινη συμβολή και να αναλαμβάνουν δράσεις με βάση προηγούμενες παρατηρήσεις.

Αν και η τεχνητή νοημοσύνη και η μηχανική μάθηση χρησιμοποιούνται συχνά εναλλακτικά, πρόκειται για δύο διαφορετικές έννοιες. Η τεχνητή νοημοσύνη είναι η ευρύτερη έννοια - μηχανές που λαμβάνουν αποφάσεις, μαθαίνουν νέες δεξιότητες και επιλύουν προβλήματα με παρόμοιο τρόπο με τους ανθρώπους - ενώ η μηχανική μάθηση είναι ένα υποσύνολο της τεχνητής νοημοσύνης που επιτρέπει στα ευφυή συστήματα να μαθαίνουν αυτόνομα νέα πράγματα από δεδομένα.

Σε αυτό το κεφάλαιο, αρχικά, θα γίνει μια εισαγωγή στην επιστήμη της Μηχανικής Μάθησης με τα βασικά στοιχεία και τις κατηγορίες της. Στη συνέχεια θα περιγραφούν θεωρητικά οι αλγόριθμοι οι οποίοι θα χρησιμοποιηθούν αργότερα για ταξινόμηση. Τέλος, παρουσιάζονται και αναλύονται τεχνικές και αλγόριθμοι που αφορούν τα Νευρωνικά Δίκτυα και πιο συγκεκριμένα το Perceptron πολλαπλών επιπέδων.

## 3.1 ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

Η μηχανική μάθηση έχει γίνει όλο και πιο δημοφιλής τα τελευταία χρόνια, με εφαρμογές σε ένα ευρύ φάσμα τομέων. Με την έλευση των μεγάλων δεδομένων και του ισχυρού υλικού (hardware όπως πολύ ισχυρές κάρτες γραφικών), η μηχανική μάθηση μπόρεσε να επιτύχει καινοτομίες σε πολλούς τομείς και οδήγησε στην ανάπτυξη νέων προϊόντων και υπηρεσιών.



Σχήμα 3.1: Κλάδοι και εφαρμογές της επιστήμης της Τεχνητής Νοημοσύνης [20]

Πολλά από τα προβλήματα που συναντάμε στην καθημερινότητα και όχι μόνο, μέχρι πριν λίγα χρόνια οι άνθρωποι τα έλυναν χειροκίνητα, ενώ σήμερα επιλύονται με αλγορίθμους ML (σχήμα 3.1). Μερικά παραδείγματα περιλαμβάνουν:

- Αναγνώριση ομιλίας - Speech Recognition
- Μηχανική όραση - Computer Vision
  - Αναγνώριση αντικειμένων σε εικόνες - Object Recognition
  - Αναγνώριση και εντοπισμός της θέσης αντικειμένων σε εικόνες - Object Detection
- Αναγνώριση ηλεκτρονικών επιθέσεων στο διαδίκτυο - Cyberattack detection
- Επεξεργασία φυσικής γλώσσας - Natural Language Processing

- Κατανόηση της φυσικής γλώσσας του ανθρώπου - Natural Language Understanding
- Μοντελοποίηση και χρήση της φυσικής γλώσσας του ανθρώπου από μηχανές - Natural Language Generation
- Μηχανές αναζήτησης - Search Engines
- Αναπαράσταση γνώσης - Knowledge Representation
- Ρομποτική
- Ιατρική
  - Προσδιορισμός ασθενειών και διάγνωση
  - Ανακάλυψη και παρασκευή φαρμάκων
  - Εξατομικευμένη ιατρική
  - Πρόβλεψη ξεσπάσματος πανδημίας

#### 3.1.1 Βασικές κατηγορίες

Τα προβλήματα Μηχανικής Μάθησης χωρίζονται σε τρεις μεγάλες κατηγορίες:

- **Μάθηση με επίβλεψη - Supervised Learning**

Η μάθηση με επίβλεψη είναι μια τεχνική μηχανικής μάθησης όπου ένα μοντέλο εκπαιδεύεται σε ένα σύνολο δεδομένων με ετικέτες, πράγμα που σημαίνει ότι η σωστή έξοδος (ή "ετικέτα") είναι ήδη γνωστή για κάθε είσοδο. Το μοντέλο εκπαιδεύεται για να μάθει μια συνάρτηση απεικόνισης, γνωστή και ως υπόθεση, η οποία μπορεί να γενικευτεί σε νέα αθέατα δεδομένα. Η υπόθεση αντιπροσωπεύεται από ένα σύνολο παραμέτρων που βελτιστοποιούνται κατά τη διάρκεια της διαδικασίας εκπαίδευσης με τη χρήση ενός αλγορίθμου βελτιστοποίησης, όπως η κάθοδος κλίσης. Ο στόχος της μάθησης με επίβλεψη είναι να βρεθεί το καλύτερο σύνολο παραμέτρων που ελαχιστοποιεί το σφάλμα πρόβλεψης στο σύνολο εκπαίδευσης. Η μάθηση με επίβλεψη μπορεί να χωριστεί σε δύο κύριες κατηγορίες: ταξινόμηση και παλινδρόμηση.

- **Μάθηση χωρίς επίβλεψη - Unsupervised Learning**

Η μάθηση χωρίς επίβλεψη είναι μια τεχνική μηχανικής μάθησης όπου το μοντέλο δεν λαμβάνει δεδομένα με ετικέτες και πρέπει να βρει μοτίβα ή σχέσεις στα δεδομένα εισόδου από μόνο του. Οι πιο συνηθισμένες προσεγγίσεις για τη μάθηση χωρίς επίβλεψη είναι η ομαδοποίηση και η μείωση της διάστασης. Οι αλγόριθμοι ομαδοποίησης ομαδοποιούν παρόμοια σημεία δεδομένων μαθαίνοντας ένα σύνολο πρωτοτύπων που αντιπροσωπεύουν τις συστάδες. Οι αλγόριθμοι μείωσης της διαστατικότητας μειώνουν τον αριθμό των χαρακτηριστικών στα δεδομένα, καθιστώντας τα ευκολότερα οπτικοποιήσιμα και αναλύσιμα. Η πιο συνηθισμένη προσέγγιση για τη μείωση της διαστατικότητας είναι η χρήση γραμμικών μεθόδων, όπως η ανάλυση κύριων συνιστωσών (PCA) ή η παραγοντοποίηση μη αρνητικών πινάκων (NMF).

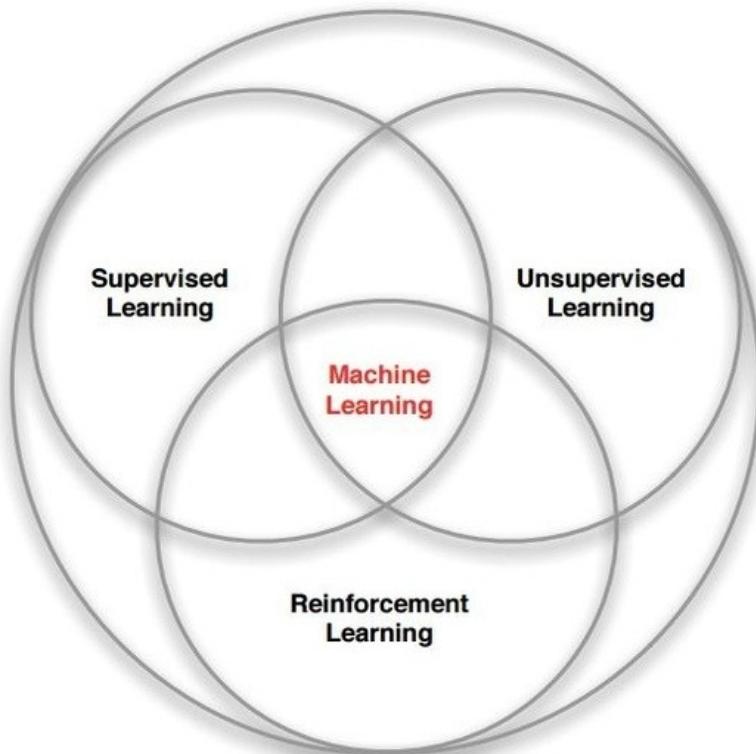
### 3.1. ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

- **Μάθηση με ανταμοιβή - Reinforcement Learning**

Η ενισχυτική μάθηση [21] είναι ένας τύπος μηχανικής μάθησης όπου ένας πράκτορας μαθαίνει να λαμβάνει αποφάσεις αλληλεπιδρώντας με το περιβάλλον του και λαμβάνοντας ανατροφοδότηση με τη μορφή ανταμοιβών ή ποινών. Στόχος του πράκτορα είναι να μεγιστοποιήσει τη σωρευτική ανταμοιβή με την πάροδο του χρόνου μαθαίνοντας μια πολιτική, μια αντιστοίχιση από καταστάσεις σε ενέργειες, που μεγιστοποιεί την αναμενόμενη σωρευτική ανταμοιβή. Η πιο συνηθισμένη προσέγγιση για την ενισχυτική μάθηση είναι η χρήση μιας μεθόδου βασισμένης στην τιμή, όπως η Q-learning, η SARSA, ή μιας μεθόδου βασισμένης στην πολιτική, όπως η REINFORCE.

Συνοπτικά, η επιβλεπόμενη μάθηση επικεντρώνεται στην εκμάθηση μιας συνάρτησης αντιστοίχισης από τις εισόδους στις εξόδους με τη χρήση επισημασμένων δεδομένων, η μη επιβλεπόμενη μάθηση επικεντρώνεται στην ανακάλυψη της εγγενούς δομής στα δεδομένα μέσω ομαδοποίησης ή μείωσης της διάστασης και η μάθηση ενίσχυσης επικεντρώνεται στη λήψη αποφάσεων με τη μεγιστοποίηση των ανταμοιβών με την πάροδο του χρόνου μέσω της εκμάθησης μιας πολιτικής που αντιστοιχίζει καταστάσεις σε ενέργειες.

Πέρα από αυτά που αναφέρθηκαν, υπάρχουν και προβλήματα που είναι υβριδικά, δηλαδή είναι συνδυασμός των παραπάνω. Το σχήμα 3.2 δείχνει τις βασικές κατηγορίες αλγορίθμων ML σε διάγραμμα Venn.



Σχήμα 3.2: Διάγραμμα Venn των βασικών κατηγοριών μηχανικής μάθησης [21]

### 3.1.2 Κατηγορίες των αλγορίθμων βάση την μορφή της εξόδου

Οι αλγόριθμοι μηχανικής μάθησης μπορούν επίσης να χωριστούν σε τρεις κύριες κατηγορίες ανάλογα με τη μορφή της εξόδου που θα παράγουν [22][23]:

- **Αλγόριθμοι ταξινόμησης:**

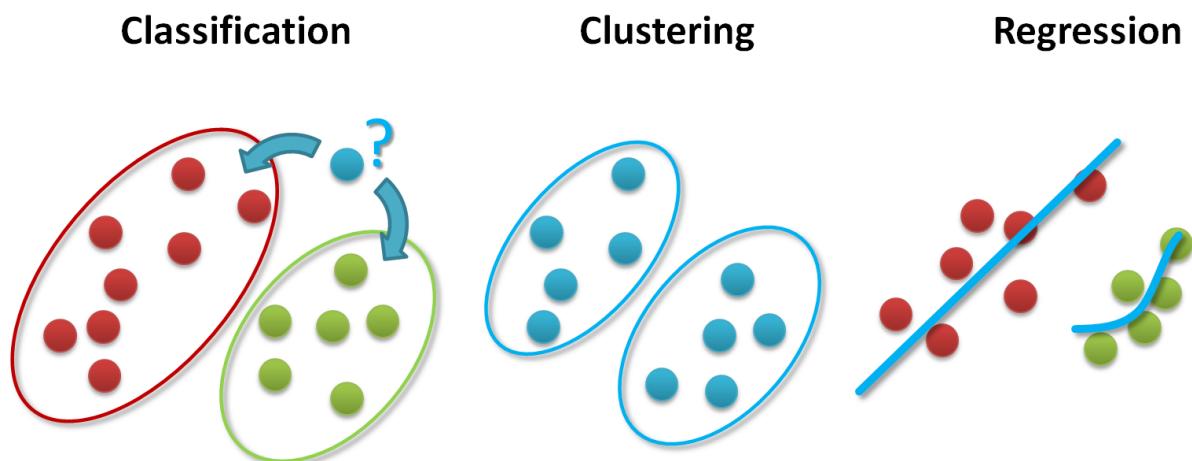
Αυτοί οι αλγόριθμοι χρησιμοποιούνται για την πρόβλεψη διακριτών τιμών (όπως ναι ή όχι) με βάση ένα ή περισσότερα χαρακτηριστικά εισόδου. Ο στόχος είναι να βρεθεί ένα όριο απόφασης που διαχωρίζει τις διαφορετικές κλάσεις στα δεδομένα. Παραδείγματα αλγορίθμων ταξινόμησης περιλαμβάνουν τη λογιστική παλινδρόμηση, τους k-κοντινότερους γείτονες, τα δέντρα αποφάσεων και τις μηχανές διανυσμάτων υποστήριξης.

- **Αλγόριθμοι ομαδοποίησης:**

Αυτοί οι αλγόριθμοι χρησιμοποιούνται για την ομαδοποίηση παρόμοιων περιπτώσεων στα δεδομένα σε συστάδες ή ομάδες. Η συσταδοποίηση είναι μια μορφή μάθησης χωρίς επίβλεψη, καθώς τα σημεία δεδομένων δεν είναι επισημασμένα και ο αλγόριθμος προσπαθεί να βρει τη δομή των δεδομένων από μόνος του. Παραδείγματα αλγορίθμων ομαδοποίησης περιλαμβάνουν την k-means, την ιεραρχική ομαδοποίηση και την ομαδοποίηση με βάση την πυκνότητα.

- **Αλγόριθμοι παλινδρόμησης:**

Οι αλγόριθμοι αυτοί χρησιμοποιούνται για την πρόβλεψη συνεχών τιμών (όπως μια τιμή ή μια θερμοκρασία) με βάση ένα ή περισσότερα χαρακτηριστικά εισόδου. Ο στόχος είναι να βρεθεί η καλύτερη γραμμή ή συνάρτηση προσαρμογής που περιγράφει τη σχέση μεταξύ των εισόδων και της εξόδου. Παραδείγματα αλγορίθμων παλινδρόμησης περιλαμβάνουν τη γραμμική παλινδρόμηση, την πολυωνυμική παλινδρόμηση και την παλινδρόμηση διανυσμάτων υποστήριξης.



Σχήμα 3.3: Κατηγορίες των αλγορίθμων βάση την μορφή της εξόδου [23]

### 3.1. ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

---

Κάθε κατηγορία έχει το δικό της σύνολο αλγορίθμων και η επιλογή του καλύτερου αλγορίθμου εξαρτάται από τα χαρακτηριστικά των δεδομένων και του εκάστοτε προβλήματος.

Αξίζει να σημειωθεί ότι ορισμένοι αλγόριθμοι μπορούν να χρησιμοποιηθούν για πολλαπλές εργασίες, για παράδειγμα, ο αλγόριθμος k-κοντινότεροι γείτονες μπορεί να χρησιμοποιηθεί τόσο για εργασίες ταξινόμησης όσο και για εργασίες παλινδρόμησης.

#### 3.1.3 Γενικές κατηγορίες επιβλεπόμενης μάθησης

Οι αλγόριθμοι επιβλεπόμενης μάθησης μπορούν γενικά να χωριστούν σε δύο κύριες κατηγορίες με βάση τον τρόπο λειτουργίας τους: παραμετρικοί αλγόριθμοι και μη παραμετρικοί αλγόριθμοι [24].

- **Παραμετρικοί αλγόριθμοι:**

Αυτοί οι αλγόριθμοι κάνουν υποθέσεις σχετικά με την υποκείμενη κατανομή πιθανοτήτων των δεδομένων και μαθαίνουν ένα σύνολο παραμέτρων που ταιριάζουν καλύτερα στα δεδομένα. Μόλις μάθουν τις παραμέτρους, ο αλγόριθμος μπορεί να κάνει προβλέψεις για νέα σημεία δεδομένων. Παραδείγματα παραμετρικών αλγορίθμων περιλαμβάνουν τη γραμμική παλινδρόμηση, τη λογιστική παλινδρόμηση και τις μηχανές διανυσμάτων υποστήριξης.

- **Μη παραμετρικοί αλγόριθμοι:**

Αυτοί οι αλγόριθμοι κάνουν λίγες ή καθόλου υποθέσεις σχετικά με την υποκείμενη κατανομή πιθανοτήτων των δεδομένων και αντ' αυτού μαθαίνουν έναν γενικό κανόνα από τα δεδομένα. Αυτοί οι αλγόριθμοι είναι πιο ευέλικτοι και μπορούν να προσαρμοστούν σε μεγαλύτερο εύρος κατανομών δεδομένων. Παραδείγματα μη παραμετρικών αλγορίθμων περιλαμβάνουν τα δέντρα αποφάσεων, τα τυχαία δάση και τους k-κοντινότερους γείτονες.

Τόσο οι παραμετρικοί όσο και οι μη παραμετρικοί αλγόριθμοι έχουν τα δικά τους πλεονεκτήματα και μειονεκτήματα. Οι παραμετρικοί αλγόριθμοι είναι υπολογιστικά λιγότερο πολύπλοκοι και εύκολα ερμηνεύσιμοι, αλλά μπορεί να είναι ευαίσθητοι στις υποθέσεις που γίνονται σχετικά με την κατανομή των δεδομένων. Οι μη παραμετρικοί αλγόριθμοι είναι πιο ευέλικτοι και μπορούν να προσαρμοστούν σε ένα ευρύτερο φάσμα κατανομών δεδομένων, αλλά είναι πιο πολύπλοκοι υπολογιστικά και μπορεί να είναι πιο δύσκολο να ερμηνευθούν.

Μια πρόσθετη ταξινόμηση μπορεί να βασιστεί στη δομή του μοντέλου :

- **Αλγόριθμοι βασισμένοι σε περιπτώσεις:**

Αυτοί οι αλγόριθμοι μαθαίνουν από τις περιπτώσεις ή τα παραδείγματα στα δεδομένα εκπαίδευσης. Δεν δημιουργούν ρητά ένα μοντέλο, αλλά αντιθέτως συγκρίνουν τα νέα δεδομένα με τα παραδείγματα στο σύνολο εκπαίδευσης και χρησιμοποιούν αυτή την ομοιότητα για να κάνουν προβλέψεις. Ο k-NN είναι ένα παράδειγμα αυτού του τύπου αλγορίθμου.

- **Αλγόριθμοι βασισμένοι σε μοντέλα:**

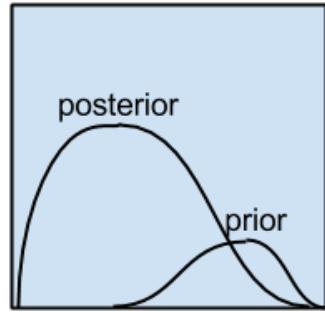
Αυτοί οι αλγόριθμοι μαθαίνουν ένα μοντέλο από τα δεδομένα εκπαίδευσης το οποίο μπορεί να χρησιμοποιηθεί για την πραγματοποίηση προβλέψεων σε νέα δεδομένα. Το μοντέλο μπορεί να αναπαρασταθεί ως ένα σύνολο παραμέτρων, ένα σύνολο κανόνων απόφασης ή ένα νευρωνικό δίκτυο. Παραδείγματα αλγορίθμων που βασίζονται σε μοντέλα περιλαμβάνουν τη γραμμική παλινδρόμηση, τη λογιστική παλινδρόμηση και τα νευρωνικά δίκτυα.

### 3.1.4 Κατηγορίες επιβλεπόμενης μάθησης για ταξινόμηση

Πιο συγκεκριμένα, οι αλγόριθμοι επιβλεπόμενης μάθησης για ταξινόμηση μπορούν να ομαδοποιηθούν ανάλογα με τις ομοιότητές τους σχετικά με τον τρόπο λειτουργίας τους [22][25]:

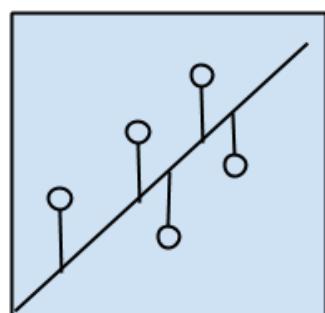
#### Bayesian ταξινομητές

Αυτοί οι αλγόριθμοι βασίζονται στο θεώρημα του Bayes, το οποίο παρέχει έναν τρόπο ενημέρωσης των πεποιθήσεών μας σχετικά με την κατάσταση του κόσμου δεδομένων νέων δεδομένων. Παραδείγματα ταξινομητών Bayes περιλαμβάνουν τον Naive Bayes, τις Γκαουσιανές Διαδικασίες και τα Δίκτυα Bayes. Αυτοί οι αλγόριθμοι είναι πολύ ερμηνεύσιμοι και εύκολο να ενημερωθούν με νέα δεδομένα, αλλά μπορεί να είναι υπολογιστικά εντατικοί.



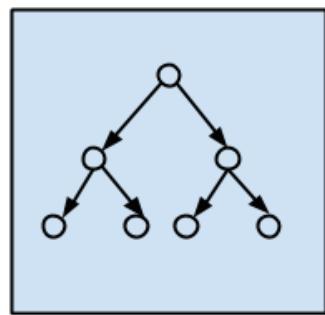
#### Γραμμικοί ταξινομητές

Οι αλγόριθμοι αυτοί βασίζονται σε γραμμικές εξισώσεις και κάνουν προβλέψεις με βάση το γραμμικό συνδυασμό των χαρακτηριστικών εισόδου. Παραδείγματα γραμμικών ταξινομητών περιλαμβάνουν τη λογιστική παλινδρόμηση και τη γραμμική ανάλυση διάκρισης. Αυτοί οι αλγόριθμοι είναι απλοί και εύκολα ερμηνεύσιμοι, αλλά είναι περιορισμένοι ως προς την ικανότητά τους να μοντελοποιούν σύνθετα όρια αποφάσεων.



#### Ταξινομητές δέντρων απόφασης

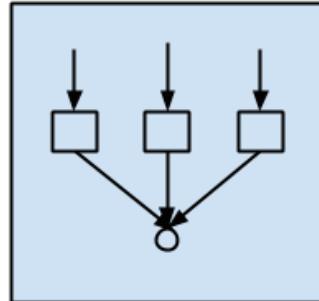
Αυτοί οι αλγόριθμοι χρησιμοποιούν μια δομή που μοιάζει με δέντρο για να αναπαραστήσουν τις αποφάσεις και τις πιθανές συνέπειές τους. Κάθε εσωτερικός κόμβος στο δέντρο αναπαριστά ένα χαρακτηριστικό και κάθε κόμβος φύλλου αναπαριστά μια ετικέτα κλάσης. Παραδείγματα αλγορίθμων δέντρων απόφασης περιλαμβάνουν τους ID3, C4.5 και CART. Αυτοί οι αλγόριθμοι είναι απλοί στην κατανόηση και την ερμηνεία, αλλά μπορούν εύκολα να υπερπροσαρμόσουν τα δεδομένα εκπαίδευσης.



### 3.1. ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΗΣ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

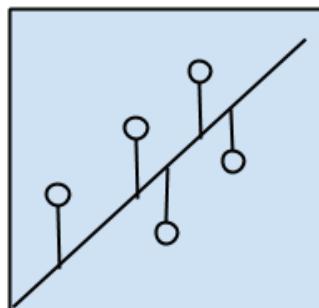
#### Ταξινομητές συνόλου

Αυτοί οι αλγόριθμοι χρησιμοποιούν πολλαπλά μοντέλα για να κάνουν προβλέψεις. Μπορούν να χωριστούν σε δύο τύπους: bagging και boosting. Οι μέθοδοι bagging, όπως το Random Forest, εκπαιδεύουν πολλαπλά μοντέλα ανεξάρτητα και συνδυάζουν τις προβλέψεις τους με πλειοφηφική φημοφορία ή μέσο όρο. Οι μέθοδοι Boosting, όπως το AdaBoost, εκπαιδεύουν διαδοχικά πολλαπλά μοντέλα, όπου κάθε μοντέλο προσπαθεί να διορθώσει τα λάθη των προηγούμενων μοντέλων. Αυτοί οι αλγόριθμοι μπορούν συχνά να βελτιώσουν την απόδοση των μεμονωμένων μοντέλων μειώνοντας την υπερβολική προσαρμογή και αυξάνοντας την ποικιλομορφία των μοντέλων.



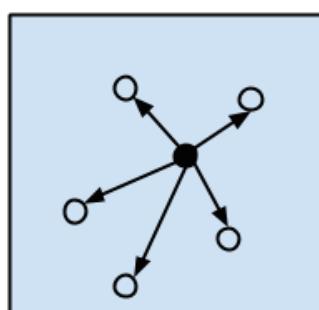
#### Ταξινομητές μηχανών διανυσμάτων υποστήριξης (SVM)

Αυτοί οι αλγόριθμοι βασίζονται στην έννοια της εύρεσης του γραμμικού ταξινομητή μέγιστου περιθωρίου. Βρίσκουν το καλύτερο όριο απόφασης που διαχωρίζει τις κλάσεις με το μεγαλύτερο περιθώριο. Οι SVM μπορούν επίσης να χρησιμοποιηθούν για μη γραμμική ταξινόμηση με τη χρήση συναρτήσεων πυρήνα. Αυτοί οι αλγόριθμοι είναι ισχυροί για γραμμική και μη γραμμική ταξινόμηση, αλλά μπορεί να είναι ευαίσθητοι στην επιλογή του πυρήνα και στην κλίμακα των εισόδων.



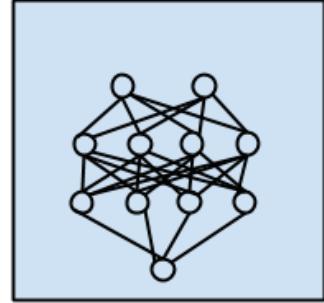
#### Ταξινομητές με βάση την απόσταση

Αυτοί οι αλγόριθμοι ταξινομούν μια περίπτωση με βάση την απόστασή της από τις περιπτώσεις του συνόλου εκπαίδευσης. Ο αλγόριθμος k-κοντινότεροι γείτονες (k-NN) είναι μια μέθοδος βασισμένη στην απόσταση που ταξινομεί τα νέα παραδείγματα με βάση την πλειοψηφική κλάση των k κοντινότερων περιπτώσεων στο σύνολο εκπαίδευσης.



### Ταξινομητές νευρωνικών δικτύων

Αυτοί οι αλγόριθμοι βασίζονται στη δομή και τη λειτουργία του ανθρώπινου εγκεφάλου και αποτελούνται από διασυνδεδεμένους κόμβους επεξεργασίας που ονομάζονται τεχνητοί νευρώνες. Τα νευρωνικά δίκτυα μπορούν να χωρίστονται σε διάφορους τύπους, όπως τα νευρωνικά δίκτυα τροφοδότησης, τα αναδρομικά και τα συνελικτικά νευρωνικά δίκτυα. Χρησιμοποιούνται σε ένα ευρύ φάσμα εφαρμογών, όπως η ταξινόμηση εικόνων, η αναγνώριση ομιλίας και η επεξεργασία φυσικής γλώσσας. Αυτοί οι αλγόριθμοι είναι ισχυρά μοντέλα που μπορούν να μάθουν εξαιρετικά μη γραμμικές και πολύπλοκες σχέσεις, αλλά μπορεί να είναι δύσκολο να ερμηνευθούν και μπορεί να απαιτούν μεγάλο όγκο δεδομένων και υπολογιστικών πόρων.



Εν συντομίᾳ, οι Bayesian ταξινομητές είναι πολύ ερμηνεύσιμοι και εύκολο να ενημερωθούν με νέα δεδομένα, αλλά μπορεί να είναι υπολογιστικά εντατικοί. Οι γραμμικοί ταξινομητές είναι απλοί και εύκολα ερμηνεύσιμοι, αλλά είναι περιορισμένοι ως προς την ικανότητά τους να μοντελοποιούν σύνθετα όρια αποφάσεων. Οι ταξινομητές δέντρων απόφασης είναι απλοί στην κατανόηση και την ερμηνεία, αλλά μπορούν εύκολα να υπερπροσαρμόσουν τα δεδομένα εκπαίδευσης. Οι ταξινομητές συνόλου μπορούν συχνά να βελτιώσουν την απόδοση των μεμονωμένων μοντέλων μειώνοντας την υπερπροσαρμογή και αυξάνοντας την ποικιλομορφία των μοντέλων. Οι ταξινομητές που βασίζονται στην απόσταση δεν κάνουν υποθέσεις σχετικά με την κατανομή των δεδομένων, είναι εύκολοι στην ερμηνεία, αλλά μπορεί να είναι υπολογιστικά ακριβοί για μεγάλα σύνολα δεδομένων. Οι ταξινομητές νευρωνικών δικτύων είναι ισχυρά μοντέλα που μπορούν να μάθουν εξαιρετικά μη γραμμικές και πολύπλοκες σχέσεις, αλλά μπορεί να είναι δύσκολο να ερμηνευθούν και μπορεί να απαιτούν μεγάλο όγκο δεδομένων και υπολογιστικών πόρων. Οι ταξινομητές SVM είναι ισχυροί για γραμμική και μη γραμμική ταξινόμηση, αλλά μπορεί να είναι ευαίσθητοι στην επιλογή του πυρήνα και στην κλίμακα των εισόδων.

#### 3.1.5 Αναπαράσταση δεδομένων

Η μορφή της αναπαράστασης των δεδομένων αποτελεί σημαντικό παράγοντα για την απόδοση των αλγορίθμων μηχανικής μάθησης, διότι επηρεάζει την ευαισθησία του αλγορίθμου στην υποκείμενη δομή των δεδομένων.

Η αναπαράσταση δεδομένων μπορεί να λάβει πολλές μορφές, συμπεριλαμβανομένων των αριθμητικών, των κατηγορικών και των δεδομένων κειμένου. Τα αριθμητικά δεδομένα, όπως οι συνεχείς ή διακριτές μετρήσεις, μπορούν να αναπαράσταθούν με τη χρήση ακέραιων αριθμών, κωδικοποίησης ενός ψηφίου ή δυαδικής κωδικοποίησης. Δεδομένα κειμένου, όπως κείμενα φυσικής γλώσσας, μπορούν να αναπαρασταθούν με τη

## 3.2. ΜΕΘΟΔΟΙ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

---

χρήση ενσωμάτωσης λέξεων, όπως το word2vec ή το GloVe, ή με αναπαραστάσεις bag-of-words [26].

Είναι επίσης σημαντικό να ληφθεί υπόψη η κλίμακα και η κατανομή των δεδομένων. Για παράδειγμα, η κανονικοποίηση ή η τυποποίηση των δεδομένων μπορεί να βελτιώσει την απόδοση ορισμένων αλγορίθμων που είναι ευαίσθητοι στην κλίμακα των χαρακτηριστικών εισόδου.

Επιπλέον, η αναπαράσταση των δεδομένων μπορεί να βελτιωθεί με τεχνικές εξαγωγής χαρακτηριστικών, επιλογής χαρακτηριστικών και μείωσης της διαστατικότητας, οι οποίες μπορούν να χρησιμοποιηθούν για τη μετατροπή των ακατέργαστων δεδομένων σε μια πιο κατατοπιστική και συμπαγή αναπαράσταση που είναι πιο κατάλληλη για τη συγκεκριμένη εργασία μηχανικής μάθησης.

Συνολικά, η μορφή της αναπαράστασης των δεδομένων είναι ένα κρίσιμο βήμα στον αγωγό (pipeline) μηχανικής μάθησης, καθώς καθορίζει την ποιότητα και το πληροφοριακό περιεχόμενο των δεδομένων που θα χρησιμοποιήσει ο αλγόριθμος για να μάθει.

## 3.2 ΜΕΘΟΔΟΙ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ

---

Σε αυτήν την ενότητα θα περιγραφούν αναλυτικά οι αλγόριθμοι μηχανικής μάθησης που θα εφαρμοστούν στη συνέχεια.

### 3.2.1 Naive Bayes

Οι μέθοδοι Naive Bayes [27] είναι ένα σύνολο αλγορίθμων επιβλεπόμενης μάθησης που βασίζονται στην εφαρμογή του θεωρήματος του Bayes με την "αφελή" υπόθεση της ανεξαρτησίας υπό όρους μεταξύ κάθε ζεύγους χαρακτηριστικών, δεδομένης της τιμής της μεταβλητής κλάσης. Ένας πιο περιγραφικός όρος για το υποκείμενο μοντέλο πιθανότητας θα ήταν το "ανεξάρτητο μοντέλο χαρακτηριστικών". Ένας "αφελής" ταξινομητής Bayes [28] υποθέτει ότι η παρουσία (ή απουσία) ενός συγκεκριμένου χαρακτηριστικού μιας κλάσης δεν σχετίζεται με την παρουσία (ή απουσία) οποιουδήποτε άλλου χαρακτηριστικού, δεδομένης της μεταβλητής κλάσης. Το θεώρημα του Bayes δηλώνει την ακόλουθη σχέση [29], δεδομένης της μεταβλητής κλάσης  $y$  και του εξαρτημένου διανύσματος χαρακτηριστικών  $x_1$  μέσω  $x_n$ :

$$P(y \mid x_1, \dots, x_n) = \frac{P(y) P(x_1, \dots, x_n \mid y)}{P(x_1, \dots, x_n)}$$

Χρησιμοποιώντας την "αφελή" υπόθεση ανεξαρτησίας υπό όρους ότι :

$$P(x_\alpha \mid y, x_1, \dots, x_{\alpha-1}, x_{\alpha+1}, \dots, x_n) = P(x_\alpha \mid y)$$

για όλα τα χαρακτηριστικά  $\alpha$ , αυτή η σχέση απλοποιείται σε :

$$P(y \mid x_1, \dots, x_n) = \frac{P(y) \prod_{\alpha=1}^n P(x_\alpha \mid y)}{P(x_1, \dots, x_n)}$$

Από τη στιγμή που το  $P(x_1, \dots, x_n)$  είναι σταθερό δεδομένης της εισόδου, μπορεί να χρησιμοποιηθεί ο ακόλουθος κανόνας ταξινόμησης:

$$P(y \mid x_1, \dots, x_n) \propto P(y) \prod_{\alpha=1}^n P(x_\alpha \mid y)$$

↓

$$\hat{y} = \arg \max_y P(y) \prod_{\alpha=1}^n P(x_\alpha \mid y)$$

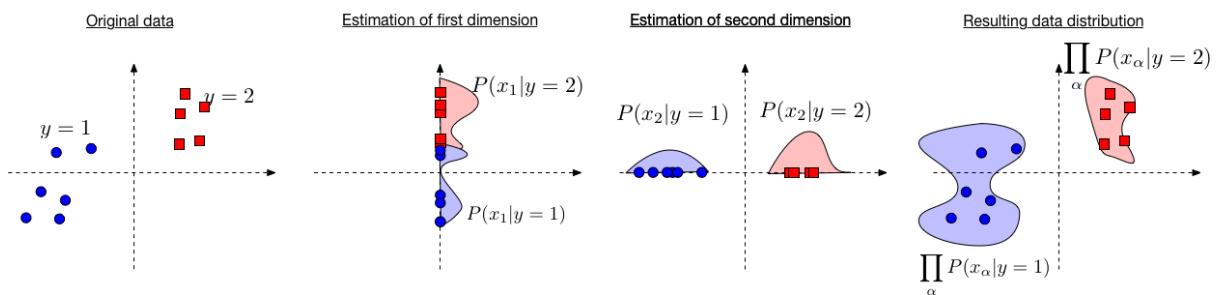
και μπορεί να χρησιμοποιηθεί η Μέγιστη A Posteriori (MAP) εκτίμηση για να υπολογιστούν τα  $P(y)$  και  $P(x_\alpha \mid y)$ , όπου το πρώτο είναι τότε η σχετική συχνότητα της κλάσης  $y$  στο σετ εκπαίδευσης.

Οι διαφορετικοί "αφελείς" ταξινομητές Bayes διαφέρουν χυρίως ως προς τις υποθέσεις που κάνουν σχετικά με την κατανομή του  $P(x_\alpha \mid y)$ .

Παρά τις φαινομενικά υπεραπλουστευμένες υποθέσεις τους, οι "αφελείς" ταξινομητές Bayes έχουν λειτουργήσει αρκετά καλά σε πολλές πραγματικές καταστάσεις, όπως είναι η περίφημη ταξινόμηση εγγράφων και το φιλτράρισμα ανεπιθύμητων μηνυμάτων. Απαιτούν μια μικρή ποσότητα δεδομένων εκπαίδευσης για την εκτίμηση των απαραίτητων παραμέτρων.

Οι Naive Bayes μαθητές (learners) και ταξινομητές μπορεί να είναι εξαιρετικά γρήγοροι σε σύγκριση με πιο εξελιγμένες μεθόδους. Η αποσύνδεση των κατανομών χαρακτηριστικών υπό όρους κλάσης σημαίνει ότι κάθε κατανομή μπορεί να εκτιμηθεί ανεξάρτητα ως μονοδιάστατη κατανομή όπως φαίνεται στο σχήμα 3.4. Αυτό με τη σειρά του βοηθά στην άμβλυνση των προβλημάτων που προκύπτουν από την "κατάρα" της διάστασης.

Από την άλλη πλευρά, αν και ο "αφελής" Bayes είναι γνωστός ως ένας αξιοπρεπής ταξινομητής, είναι γνωστό ότι είναι κακός εκτιμητής, συνεπώς δεν θα πρέπει να χρησιμοποιείται για εύρεση πιθανότητας γεγονότος.



Σχήμα 3.4: Απεικόνιση πίσω από τον αλγόριθμο Naive Bayes [29]

Υπολογισμός του  $P(x_\alpha \mid y)$  ανεξάρτητα σε κάθε διάσταση (δύο μεσαίες εικόνες) και στη συνέχεια λαμβάνεται μια εκτίμηση της πλήρους κατανομής υποθέτοντας ανεξαρτησία υπό όρους  $P(\mathbf{x}|y) = \prod_\alpha P(x_\alpha|y)$  (δεξιά εικόνα).

### Gaussian Naive Bayes

Στον ταξινομητή Gaussian Naive Bayes που θα χρησιμοποιηθεί στη συνέχεια, η κατανομή των χαρακτηριστικών θεωρείται ότι είναι Γκαουσιανή, δηλαδή (η πλήρης

κατανομή)  $P(\mathbf{x}|y) \sim \mathcal{N}(\mu_y, \Sigma_y)$  όπου  $\Sigma_y$  είναι διαγώνιος πίνακας συνδιακύμανσης με  $[\Sigma_y]_{\alpha,\alpha} = \sigma_{\alpha,y}^2$ . Έτσι η συνάρτηση πυκνότητας πιθανότητας γίνεται :

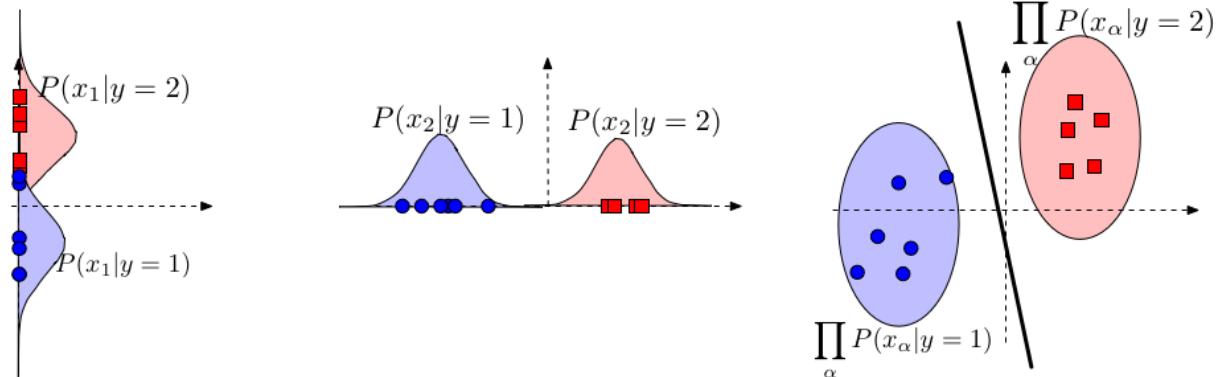
$$P(x_\alpha | y = c) = \mathcal{N}(\mu_{\alpha c}, \sigma_{\alpha c}^2) = \frac{1}{\sqrt{2\pi}\sigma_{\alpha c}} e^{-\frac{1}{2}\left(\frac{x_\alpha - \mu_{\alpha c}}{\sigma_{\alpha c}}\right)^2}$$

Οι παράμετροι των κατανομών υπολογίζονται για κάθε διάσταση και κλάση ανεξάρτητα. Οι κατανομές Gauss έχουν μόνο δύο παραμέτρους, τον μέσο όρο και τη διακύμανση. Ο μέσος όρος  $\mu_{\alpha,y}$  υπολογίζεται από τη μέση τιμή του χαρακτηριστικού της διάστασης  $\alpha$  από όλα τα δείγματα με ετικέτα  $y$ . Η (τετραγωνική) τυπική απόκλιση είναι απλώς η απόκλιση αυτής της εκτίμησης.

$$\begin{aligned} \mu_{\alpha c} &\leftarrow \frac{1}{n_c} \sum_{i=1}^n I(y_i = c) x_{i\alpha} & \text{όπου } n_c = \sum_{i=1}^n I(y_i = c) \\ \sigma_{\alpha c}^2 &\leftarrow \frac{1}{n_c} \sum_{i=1}^n I(y_i = c) (x_{i\alpha} - \mu_{\alpha c})^2 \end{aligned}$$

### Ο Naive Bayes ως γραμμικός ταξινομητής

Ο Naive Bayes οδηγεί σε ένα γραμμικό όριο απόφασης σε πολλές κοινές περιπτώσεις [29]. Στο σχήμα 3.5 φαίνεται η περίπτωση όπου το  $P(x_\alpha|y)$  είναι Γκαουσιανό και όπου το  $\sigma_{\alpha,c}$  είναι πανομοιότυπο για όλα τα  $c$  (αλλά μπορεί να διαφέρει μεταξύ των διαστάσεων (= χαρακτηριστικών)  $\alpha$ ). Το όριο των ελλειψώνειδών δείχνει περιοχές ίσων πιθανοτήτων  $P(\mathbf{x}|y)$ . Η γραμμή απόφασης υποδεικνύει το όριο απόφασης όπου  $P(y=1|\mathbf{x}) = P(y=2|\mathbf{x})$ .



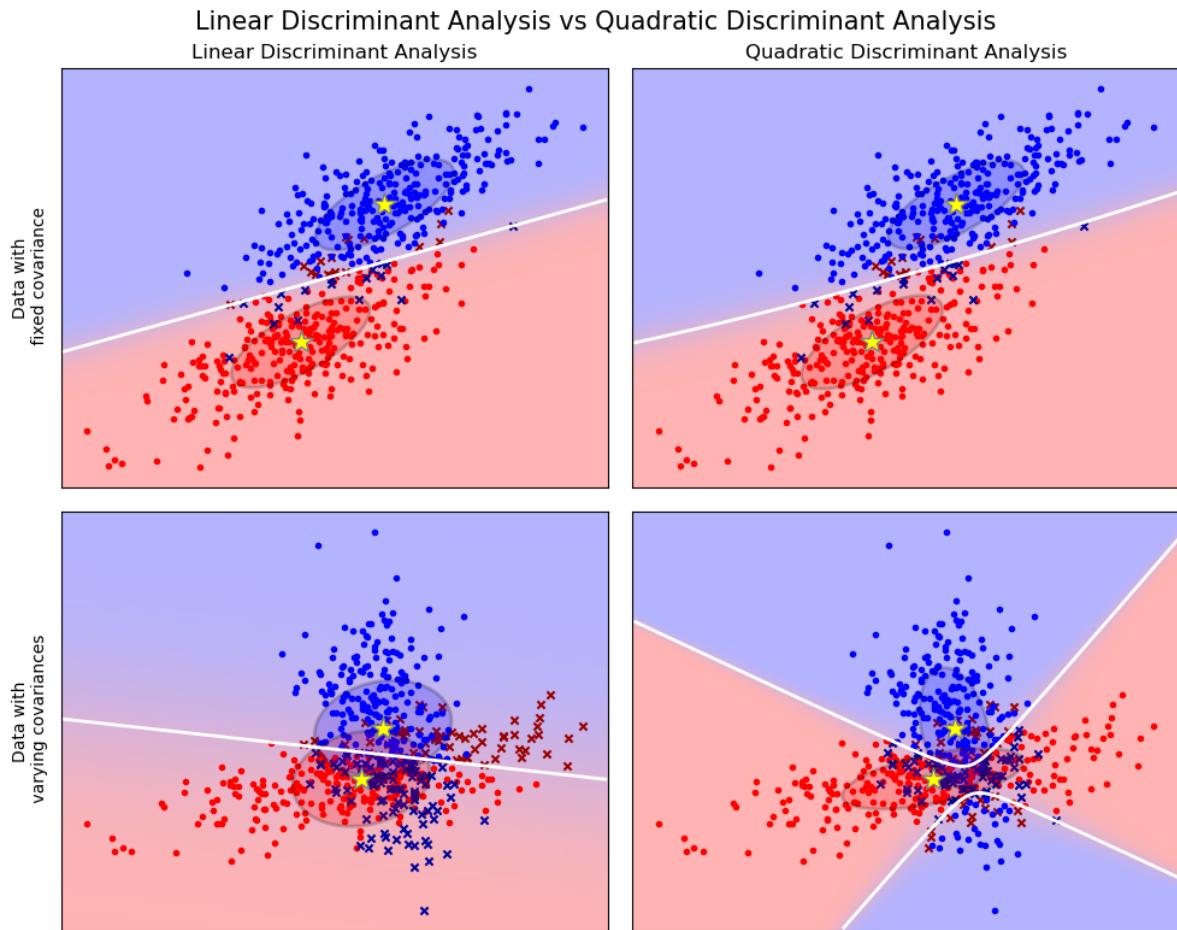
Σχήμα 3.5: Γραμμικό όριο απόφασης στον Naive Bayes [29]

### 3.2.2 Linear & Quadratic Discriminate Analysis

Η Ανάλυση Γραμμικής Διάκρισης (LDA) και η Ανάλυση Τετραγωνικής Διάκρισης (QDA) είναι δύο κλασικοί ταξινομητές, με μια γραμμική και μια τετραγωνική επιφάνεια απόφασης, όπως υποδηλώνουν τα ονόματά τους αντίστοιχα.

### ΚΕΦΑΛΑΙΟ 3. ΜΗΧΑΝΙΚΗ ΚΑΙ ΒΑΘΙΑ ΜΑΘΗΣΗ

Αυτοί οι ταξινομητές είναι ελκυστικοί επειδή έχουν λύσεις κλειστής μορφής που μπορούν εύκολα να υπολογιστούν, είναι εγγενώς πολυχλασικοί, έχουν αποδειχθεί ότι λειτουργούν καλά στην πράξη και δεν έχουν υπερπαράμετρους για συντονισμό.



Σχήμα 3.6: Ανάλυση Γραμμικής & Τετραγωνικής Διάκρισης [30]

Το σχήμα 3.6 δείχνει τα όρια απόφασης για τη Ανάλυση Γραμμικής Διάκρισης και την Ανάλυση Τετραγωνικής Διάκρισης. Η κάτω σειρά δείχνει ότι η Ανάλυση Γραμμικής Διάκρισης μπορεί να μάθει μόνο γραμμικά όρια, ενώ η Ανάλυση Τετραγωνικής Διάκρισης μπορεί να μάθει τετραγωνικά όρια και επομένως είναι πιο ευέλικτη.

#### Μαθηματική διατύπωση των ταξινομητών LDA και QDA

Τόσο το LDA όσο και το QDA μπορούν να προκύψουν από απλά πιθανοτικά μοντέλα [31] που μοντελοποιούν την κατανομή υπό όρους κλάσης των δεδομένων  $P(X | y = k)$  για κάθε κλάση. Στη συνέχεια, οι προβλέψεις μπορούν να ληφθούν χρησιμοποιώντας τον κανόνα του Bayes, για κάθε δείγμα εκπαίδευσης  $x \in \mathcal{R}^d$ :

$$P(y = k|x) = \frac{P(x|y = k)P(y = k)}{P(x)} = \frac{P(x|y = k)P(y = k)}{\sum_l P(x|y = l) \cdot P(y = l)}$$

και επιλέγουμε την κλάση  $k$  που μεγιστοποιεί αυτή την εκ των υστέρων (a posteriori) πιθανότητα.

Πιο συγκεκριμένα, για Ανάλυση Γραμμικής και Τετραγωνικής Διάκρισης, το  $P(x|y)$  μοντελοποιείται ως πολυμεταβλητή Γκαουσιανή κατανομή με πυκνότητα :

$$P(x|y = k) = \frac{1}{(2\pi)^{d/2} |\Sigma_k|^{1/2}} \exp \left( -\frac{1}{2}(x - \mu_k)^t \Sigma_k^{-1} (x - \mu_k) \right)$$

όπου το  $d$  είναι ο αριθμός των χαρακτηριστικών.

### Ανάλυση Τετραγωνικής Διάκρισης (QDA)

Σύμφωνα με το παραπάνω μοντέλο, ο λογάριθμος του a posteriori είναι :

$$\begin{aligned} \log P(y = k|x) &= \log P(x|y = k) + \log P(y = k) + Cst \\ &= -\frac{1}{2} \log |\Sigma_k| - \frac{1}{2}(x - \mu_k)^t \Sigma_k^{-1} (x - \mu_k) + \log P(y = k) + Cst, \end{aligned}$$

όπου ο σταθερός όρος  $Cst$  αντιστοιχεί στον παρονομαστή  $P(x)$ , επιπρόσθετα με τους άλλους σταθερούς όρους της Γκαουσιανής. Η προβλεπόμενη κλάση είναι αυτή που μεγιστοποιεί αυτό το log-posterior.

**Σημείωση : Συσχέτιση με Gaussian Naive Bayes** Εάν στο μοντέλο QDA υποθέσει κανείς ότι οι πίνακες συνδιακύμανσης είναι διαγώνιοι, τότε οι είσοδοι θεωρούνται υπό όρους ανεξάρτητες σε κάθε κλάση και ο ταξινομητής που προκύπτει είναι ισοδύναμος με τον ταξινομητή Gaussian Naive Bayes.

### Ανάλυση Γραμμικής Διάκρισης (LDA)

Η LDA είναι μια ειδική περίπτωση της QDA, όπου οι Γκαουσιανές για κάθε κλάση υποτίθεται ότι μοιράζονται τον ίδιο πίνακα συνδιακύμανσης :  $\Sigma_k = \Sigma$  για όλα τα  $k$ . Αυτό μειώνει το log posterior σε :

$$\log P(y = k|x) = -\frac{1}{2}(x - \mu_k)^t \Sigma^{-1} (x - \mu_k) + \log P(y = k) + Cst.$$

Ο όρος  $(x - \mu_k)^t \Sigma^{-1} (x - \mu_k)$  αντιστοιχεί στην απόσταση Mahalanobis<sup>1</sup> μεταξύ του δείγματος  $x$  και του μέσου όρου  $\mu_k$ . Η απόσταση Mahalanobis λέει πόσο κοντά είναι το  $x$  από το  $\mu_k$ , ενώ επίσης υπολογίζει τη διακύμανση κάθε χαρακτηριστικού. Το LDA μπορεί λοιπόν να ερμηνευτεί ότι εκχωρεί  $x$  στην κλάση της οποίας ο μέσος όρος είναι ο πλησιέστερος ως προς την απόσταση Mahalanobis, ενώ επίσης υπολογίζει τις προηγούμενες πιθανότητες κλάσης.

To log-posterior του LDA μπορεί συνεπώς να διατυπωθεί [32] ως :

$$\log P(y = k|x) = \omega_k^t x + \omega_{k0} + Cst.$$

όπου  $\omega_k = \Sigma^{-1} \mu_k$  και  $\omega_{k0} = -\frac{1}{2} \mu_k^t \Sigma^{-1} \mu_k + \log P(y = k)$ . Οι ποσότητες αυτές αντιστοιχούν στις ιδιότητες του συντελεστή και της παρεμβολής, αντίστοιχα.

<sup>1</sup>[https://en.wikipedia.org/wiki/Mahalanobis\\_distance](https://en.wikipedia.org/wiki/Mahalanobis_distance)

Από τον παραπάνω τύπο, είναι σαφές ότι το LDA έχει μια γραμμική επιφάνεια απόφασης. Στην περίπτωση του QDA, δεν υπάρχουν υποθέσεις για τους πίνακες συνδιακύμανσης  $\Sigma_k$  των Γκαουσιανών, που αυτό οδηγεί σε τετραγωνικές επιφάνειες απόφασης.

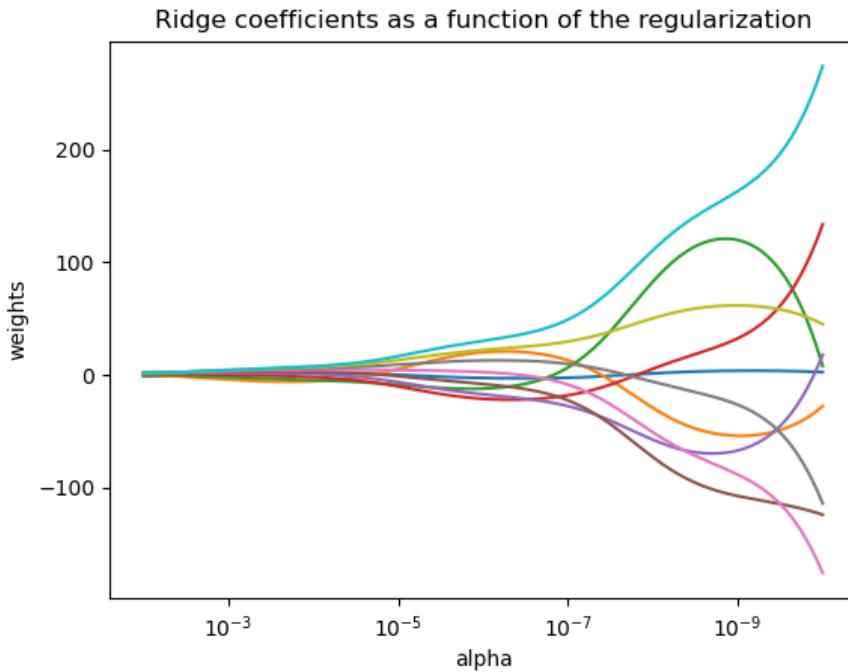
### 3.2.3 Ridge Regression & Classification

#### Regression

Η παλινδρόμηση κορυφογραμμής αντιμετωπίζει ορισμένα από τα προβλήματα των συνήθων ελάχιστων τετραγώνων επιβάλλοντας μια ποινή στο μέγεθος των συντελεστών. Οι συντελεστές κορυφογραμμής ελαχιστοποιούν ένα τιμωρημένο υπολειπόμενο άθροισμα τετραγώνων:

$$\min_w \|Xw - y\|_2^2 + \alpha \|w\|_2^2$$

Η παράμετρος πολυπλοκότητας  $\alpha \geq 0$  ελέγχει το μέγεθος της συρρίκνωσης: όσο μεγαλύτερη είναι η τιμή του  $\alpha$ , τόσο μεγαλύτερη είναι η ποσότητα της συρρίκνωσης και έτσι οι συντελεστές γίνονται πιο εύρωστοι ως προς τη συγγραμμικότητα<sup>2</sup>.



Σχήμα 3.7: Συντελεστές της συνάρτησης κορυφογραμμής ως συνάρτηση κανονικοποίησης [33]

<sup>2</sup>Η συγγραμμικότητα (collinearity) είναι μια γραμμική σχέση μεταξύ δύο ανεξάρτητων μεταβλητών. Για παράδειγμα δύο μεταβλητές είναι συγγραμμικές αν υπάρχει μια ακριβής γραμμική σχέση μεταξύ τους

## Classification

Ο παλινδρομητής Ridge έχει μια παραλλαγή ταξινομητή. Αυτός ο ταξινομητής μετατρέπει πρώτα τους δυαδικούς στόχους σε  $\{-1, 1\}$  και στη συνέχεια αντιμετωπίζει το πρόβλημα ως παλινδρόμηση, βελτιστοποιώντας τον ίδιο στόχο όπως παραπάνω. Η προβλεπόμενη κλάση αντιστοιχεί στο πρόσημο της πρόβλεψης του παλινδρομητή. Για ταξινόμηση πολλαπλών κλάσεων, το πρόβλημα αντιμετωπίζεται ως παλινδρόμηση πολλαπλών εξόδων και η προβλεπόμενη κλάση αντιστοιχεί στην έξοδο με την υψηλότερη τιμή.

Μπορεί να φαίνεται αμφίβολη η χρήση μιας (τιμωρούμενης) απώλειας ελάχιστων τετραγώνων σε ένα μοντέλο ταξινόμησης αντί για τις πιο "παραδοσιακές" απώλειες (π.χ. logistic ή hinge loss). Ωστόσο, στην πράξη, όλα αυτά τα μοντέλα μπορούν να οδηγήσουν σε παρόμοια αποτελέσματα διασταυρούμενης επικύρωσης, ενώ η τιμωρούμενη απώλεια ελάχιστων τετραγώνων που χρησιμοποιείται από τον ταξινομητή Ridge επιτρέπει μια πολύ διαφορετική επιλογή των αριθμητικών λύσεων με διαφορετικά προφίλ υπολογιστικής απόδοσης.

Ο ταξινομητής Ridge μπορεί να είναι σημαντικά ταχύτερος από π.χ. την Λογιστική παλινδρόμηση με μεγάλο αριθμό κλάσεων επειδή μπορεί να υπολογίσει τον πίνακα προβολής  $(X^T X)^{-1} X^T$  μόνο μία φορά.

Αυτός ο ταξινομητής αναφέρεται μερικές φορές ως Μηχανές Διανυσμάτων Υποστήριξης Ελάχιστων Τετραγώνων (Least Squares Support Vector Machines [34]) με γραμμικό πυρήνα.

### 3.2.4 K-Nearest Neighbors

Ο αλγόριθμος k-πλησιέστερων γειτόνων, γνωστός και ως kNN ή k-NN [35], είναι ένας μη παραμετρικός, εποπτευόμενος ταξινομητής εκμάθησης, ο οποίος χρησιμοποιεί την εγγύτητα για να κάνει ταξινόμησεις ή προβλέψεις σχετικά με την ομαδοποίηση ενός μεμονωμένου σημείου δεδομένων. Ενώ μπορεί να χρησιμοποιηθεί είτε για προβλήματα παλινδρόμησης είτε για προβλήματα ταξινόμησης, συνήθως χρησιμοποιείται ως αλγόριθμος ταξινόμησης, με βάση την υπόθεση ότι παρόμοια σημεία μπορούν να βρεθούν το ένα κοντά στο άλλο.

Αξίζει επίσης να σημειωθεί ότι ο αλγόριθμος kNN είναι επίσης μέρος μιας οικογένειας μοντέλων «τεμπέλικης εκμάθησης», που σημαίνει ότι αποθηκεύει μόνο ένα σύνολο δεδομένων εκπαίδευσης σε σύγκριση με το στάδιο εκπαίδευσης. Αυτό σημαίνει επίσης ότι όλος ο υπολογισμός γίνεται όταν μελετάται μια ταξινόμηση ή πρόβλεψη. Δεδομένου ότι βασίζεται σε μεγάλο βαθμό στη μνήμη για την αποθήκευση όλων των δεδομένων εκπαίδευσης, αναφέρεται επίσης ως μέθοδος μάθησης που βασίζεται σε στιγμιότυπα ή στη μνήμη.

Αν και δεν είναι τόσο δημοφιλής όσο ήταν κάποτε, εξακολουθεί να είναι ένας από τους πρώτους αλγόριθμους που μαθαίνει κανείς στην επιστήμη δεδομένων λόγω της απλότητας και της ακρίβειάς του. Ωστόσο, καθώς μεγαλώνει ένα σύνολο δεδομένων, το kNN γίνεται όλο και πιο αναποτελεσματικό, διακυβεύοντας τη συνολική απόδοση του μοντέλου. Χρησιμοποιείται συνήθως για απλά συστήματα συστάσεων, αναγνώριση προτύπων, εξόρυξη δεδομένων, προβλέψεις χρηματοοικονομικών αγορών, ανίχνευση εισβολής και πολλά άλλα.

### Ταξινόμηση

Για προβλήματα ταξινόμησης, αποδίδεται μια ετικέτα κλάσης με βάση την πλειοψηφία, δηλαδή χρησιμοποιείται η ετικέτα που αναπαρίσταται πιο συχνά γύρω από ένα δεδομένο σημείο δεδομένων όπως φαίνεται στο σχήμα 3.8. Ενώ αυτό θεωρείται τεχνικά "πολλαπλή ψηφοφορία", ο όρος "ψηφοφορία της πλειοψηφίας" χρησιμοποιείται πιο συχνά στη βιβλιογραφία. Η διάκριση μεταξύ αυτών των ορολογιών είναι ότι η «ψηφοφορία με πλειοψηφία» απαιτεί τεχνικά πλειοψηφία μεγαλύτερη του 50%, η οποία λειτουργεί κυρίως όταν υπάρχουν μόνο δύο κατηγορίες. Όταν υπάρχουν πολλές κλάσεις, π.χ. τέσσερις κατηγορίες, δεν χρειάζεται απαραίτητα το 50% των ψήφων για να βγει ένα συμπέρασμα σχετικά με μια τάξη. Θα μπορούσε να οριστεί μια ετικέτα τάξης με ψήφο μεγαλύτερη από 25%.

Πιο τυπικά, ας υποτεθεί μια συνάρτηση στόχου  $f(x) = y$  που εκχωρεί μια ετικέτα κλάσης  $y \in \{1, \dots, t\}$  σε ένα δείγμα εκπαίδευσης [36],

$$f : \mathbb{R}^{\geq} \rightarrow \{1, \dots, t\}.$$

(Συνήθως, χρησιμοποιείται το γράμμα  $k$  για να δηλωθεί ο αριθμός των κλάσεων, αλλά στο πλαίσιο του  $k$ -NN, θα ήταν πολύ μπερδεμένο.)

Υποθέτοντας ότι προσδιορίστηκαν οι  $k$  πλησιέστεροι γείτονες ( $\mathcal{D}_{||} \subseteq \mathcal{D}$ ) ενός αιμφίβολου σημείου (query point)  $x^{[q]}$ ,

$$\mathcal{D}_k = \{\langle x^{[1]}, f(x^{[1]}) \rangle, \dots, \langle x^{[k]}, f(x^{[k]}) \rangle\},$$

η υπόθεση  $k$ NN μπορεί να οριστεί ως

$$h(x^{[q]}) = \underset{y \in \{1, \dots, t\}}{\operatorname{argmax}} \sum_{i=1}^k \delta(y, f(x^{[i]})).$$

Εδώ, το  $\delta$  υποδηλώνει τη συνάρτηση Δέλτα Kronecker

$$\delta(a, b) = \begin{cases} 1, & \text{εάν } a = b, \\ 0, & \text{εάν } a \neq b. \end{cases}$$

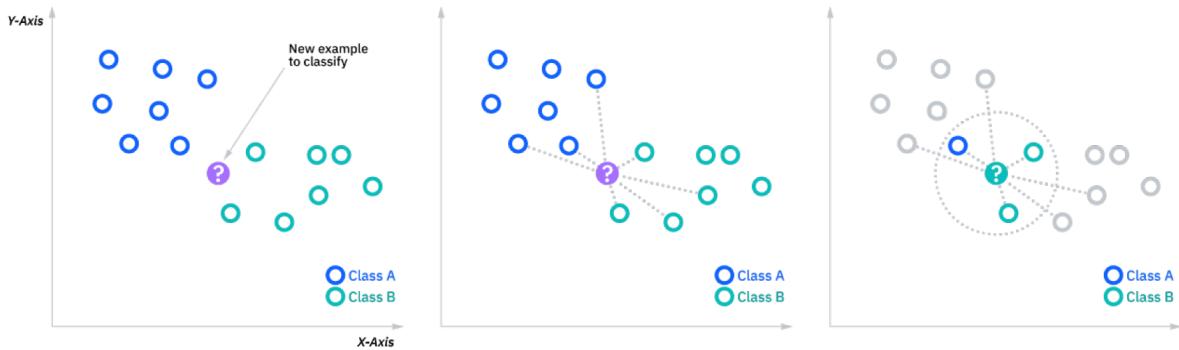
Ένας απλούστερος συμβολισμός είναι χρησιμοποιώντας από την στατιστική την επικρατούσα τιμή ή κορυφή (mode):

$$h(x^{[t]}) = mode(\{f(x^{[1]}), \dots, f(x^{[k]})\}).$$

### Παλινδρόμηση

Η γενική ιδέα του  $k$ NN για παλινδρόμηση είναι η ίδια όπως και στην ταξινόμηση: πρώτον, βρίσκονται οι  $k$  πλησιέστεροι γείτονες στο σύνολο δεδομένων. Δεύτερον, γίνεται μια πρόβλεψη με βάση τις ετικέτες των  $k$  πλησιέστερων γειτόνων. Ωστόσο, στην παλινδρόμηση, η συνάρτηση στόχου παίρνει πραγματικές τιμές αντί για διακριτές όπως συμβαίνει στην ταξινόμηση [36]:

$$f : \mathbb{R}^{\geq} \rightarrow \mathbb{R}.$$



Σχήμα 3.8: Ταξινόμηση αμφίβολου σημείου με τον αλγόριθμο k-NN [35]

Μια κοινή προσέγγιση για τον υπολογισμό του συνεχούς στόχου είναι ο υπολογισμός του αριθμητικού μέσου ή αλλιώς τη μέση τιμή (mean ή average) της τιμής του στόχου πάνω στους  $k$  πλησιέστερους γείτονες,

$$h(x^{[t]}) = \frac{1}{k} \sum_{i=1}^k f(x^{[i]}).$$

Ως εναλλακτική λύση στον μέσο όρο των τιμών στόχου των  $k$  πλησιέστερων γειτόνων για την πρόβλεψη της επικέτας ενός αμφίβολου σημείου, συνηθίζεται επίσης να χρησιμοποιείται η διάμεσος (median).

### Μετρικές Απόστασης

Προκειμένου να καθοριστεί ποια σημεία δεδομένων είναι πιο κοντά σε ένα δεδομένο αμφίβολο σημείο, θα πρέπει να υπολογιστεί η απόσταση μεταξύ του αμφίβολου σημείου και των άλλων σημείων δεδομένων. Αυτές οι μετρήσεις απόστασης βοηθούν στο σχηματισμό ορίων απόφασης, τα οποία χωρίζουν τα αμφίβολα σημεία σε διαφορετικές περιοχές. Συνήθως, τα όρια απόφασης απεικονίζονται με διαγράμματα Voronoi όπως φαίνεται στο σχήμα 3.9.

Αν και υπάρχουν πολλά μέτρα απόστασης τα οποία μπορούν να επιλεγούν, θα αναφερθούν τα πιο βασικά [35]:

- **Ευκλείδια (p = 2)**

Αυτό είναι το πιο συχνά χρησιμοποιούμενο μέτρο απόστασης και περιορίζεται σε διανύσματα πραγματικής αξίας. Χρησιμοποιώντας τον παρακάτω τύπο, μετρά μια ευθεία γραμμή μεταξύ του σημείου ερωτήματος και του άλλου σημείου που μετράται.

$$d_{Euclidean}(x^{[a]}, x^{[b]}) = \sqrt{\sum_{j=1}^m (x_j^{[a]} - x_j^{[b]})^2}$$

- **Μανχάταν (p = 1)**

Αυτή είναι επίσης μια άλλη δημοφιλής μέτρηση απόστασης, η οποία μετρά την απόλυτη τιμή μεταξύ δύο σημείων. Αναφέρεται επίσης ως απόσταση ταξί ή απόσταση μπλοκ πόλης, καθώς συνήθως απεικονίζεται με ένα πλέγμα, που απεικονίζει

πώς μπορεί κανείς να πλοηγγηθεί από τη μια διεύθυνση στην άλλη μέσω των δρόμων της πόλης.

$$d_{Manhattan}(x^{[a]}, x^{[b]}) = \sum_{j=1}^m |x_j^{[a]} - x_j^{[b]}|$$

- **Μινκόβσκι**

Αυτό το μέτρο απόστασης είναι η γενικευμένη μορφή των μετρήσεων της Ευκλείδειας απόστασης και της απόστασης Μανχάταν. Η παράμετρος,  $p$ , στον παρακάτω τύπο, επιτρέπει τη δημιουργία άλλων μετρήσεων απόστασης. Η Ευκλείδεια απόσταση αντιπροσωπεύεται από αυτόν τον τύπο όταν το  $p$  είναι ίσο με δύο και η απόσταση του Μανχάταν συμβολίζεται με  $p$  ίσο με ένα.

$$d_{Minkowski}(x^{[a]}, x^{[b]}) = \left( \sum_{j=1}^m |x_j^{[a]} - x_j^{[b]}|^p \right)^{1/p}$$

- **Χάμινγκ**

Αυτή η τεχνική χρησιμοποιείται συνήθως με διανύσματα Boolean ή συμβολοσειρών, προσδιορίζοντας τα σημεία όπου τα διανύσματα δεν ταιριάζουν. Ως αποτέλεσμα, έχει επίσης αναφερθεί ως μέτρηση επικάλυψης. Αυτό μπορεί να αναπαρασταθεί με τον ακόλουθο τύπο:

$$d_{Hamming}(x^{[a]}, x^{[b]}) = \sum_{j=1}^m |x_j^{[a]} - x_j^{[b]}|$$

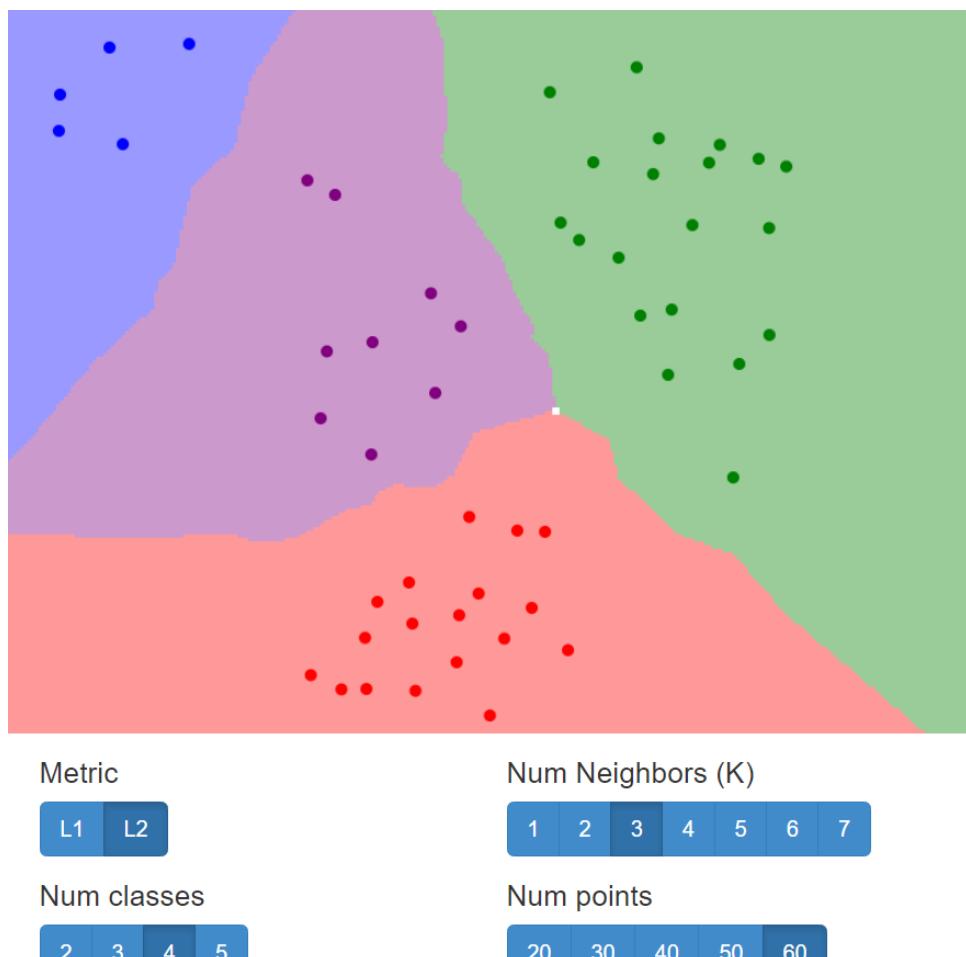
$x = y \rightarrow d = 0$   
 $x \neq y \rightarrow d \neq 1$

## Η κατάρα της διαστατικότητας

Ο αλγόριθμος  $k$ NN είναι ιδιαίτερα επιρρεπής στην κατάρα της διαστατικότητας [36]. Στη μηχανική μάθηση, η κατάρα της διαστατικότητας αναφέρεται σε σενάρια με σταθερό μέγεθος δειγμάτων εκπαίδευσης, αλλά με αυξανόμενο αριθμό διαστάσεων και εύρος τιμών χαρακτηριστικών σε κάθε διάσταση σε ένα χώρο χαρακτηριστικών υψηλών διαστάσεων. Στον  $k$ NN ένας αυξανόμενος αριθμός διαστάσεων γίνεται όλο και πιο προβληματικός επειδή όσο περισσότερες διαστάσεις προστίθενται, τόσο μεγαλύτερος πρέπει να είναι ο όγκος στον υπερχώρο για να συλλάβει έναν σταθερό αριθμό γειτόνων. Καθώς ο όγκος μεγαλώνει και μεγαλώνει, οι "γείτονες" γίνονται όλο και λιγότερο "παρόμοιοι" με το αμφίβολο σημείο, καθώς είναι πλέον όλοι σχετικά μακριά από το σημείο αυτό, λαμβάνοντας υπόψη όλες τις διαφορετικές διαστάσεις που περιλαμβάνονται κατά τον υπολογισμό των αποστάσεων ανά ζεύγη.

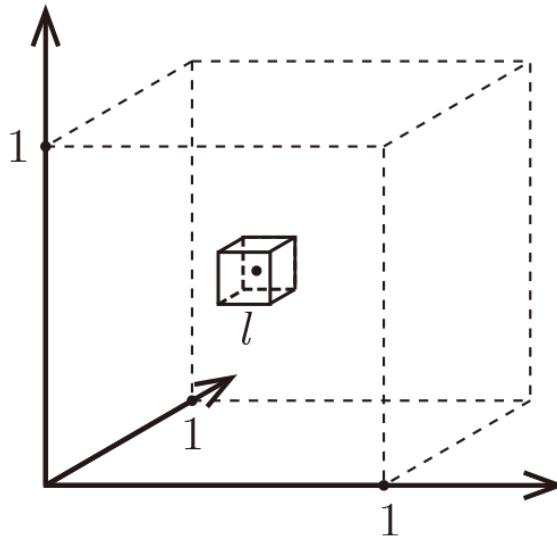
Αυτό μπορεί να δειχτεί με ένα απλό παράδειγμα [37]. Έστω ότι υπάρχουν σημεία ομοιόμορφα τυχαία μέσα στον μοναδιαίο κύβο (που φαίνεται στο σχήμα 3.10) και θα διερευνηθεί πόσο χώρο θα καταλάβουν οι  $k$  πιο κοντινοί γείτονες ενός σημείου δοκιμής μέσα σε αυτόν τον κύβο.

### 3.2. ΜΕΘΟΔΟΙ ΜΗΧΑΝΙΚΗΣ ΜΑΘΗΣΗΣ



Σχήμα 3.9: Διάγραμμα Voronoi  
<http://vision.stanford.edu/teaching/cs231n-demos/knn/>

Τυπικά, ας υποτεθεί ο μοναδιαίος κύβος  $[0, 1]^d$ . Όλα τα δεδομένα εκπαίδευσης δειγματίζονται ομοιόμορφα μέσα σε αυτόν τον κύβο, δηλαδή για κάθε  $i$ ,  $x_i \in [0, 1]^d$ , και λαμβάνονται υπόψη οι  $k = 10$  πλησιέστεροι γείτονες ενός τέτοιου σημείου δοκιμής.



Σχήμα 3.10: Μοναδιαίος κύβος [37]

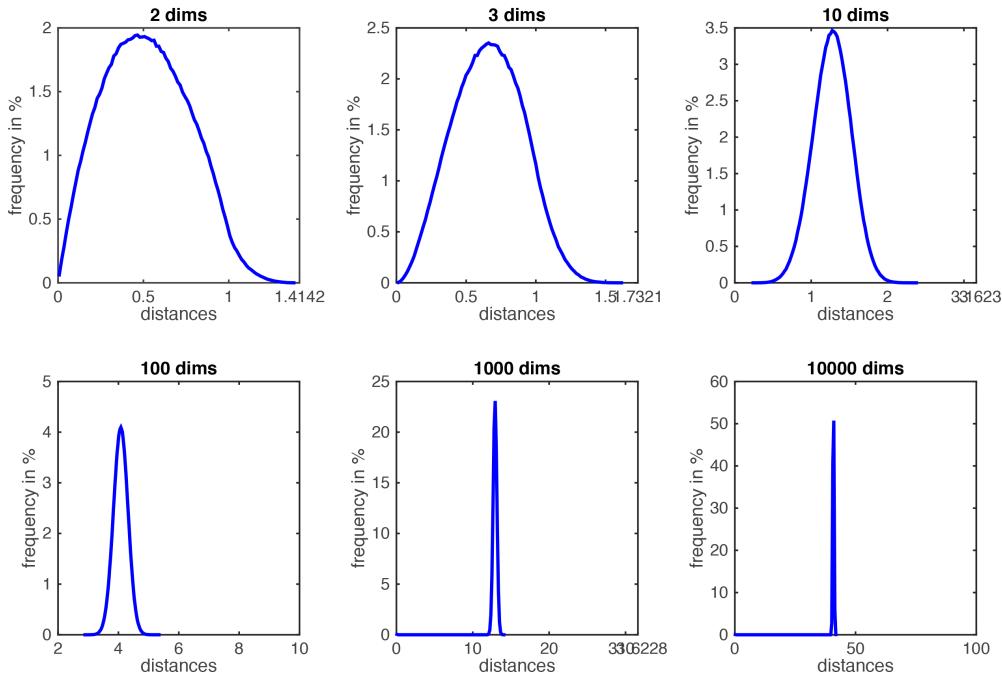
Έστω  $\ell$  το μήκος ακμής του μικρότερου υπερκύβου που περιέχει όλους τους  $k$ -πλησιέστερους γείτονες ενός σημείου δοκιμής. Στη συνέχεια  $\ell^d \approx \frac{k}{n}$  και  $\ell \approx \left(\frac{k}{n}\right)^{1/d}$ . Για  $n = 1000$ , το  $\ell$  μεγαλώνει αρκετά όπως φαίνεται στον πίνακα 3.1.

d	l
1	0.1
10	0.63
100	0.955
1000	0.9954

Πίνακας 3.1: Πίνακας d - 1 παραδείγματος μοναδιαίου κύβου

Οπότε όταν το  $d \gg 0$ , χρειάζεται σχεδόν ολόκληρος ο χώρος για να βρεθεί το 10-NN. Αυτό αναλύει τις υποθέσεις  $k$ -NN, επειδή τα  $k$ -NN δεν είναι ιδιαίτερα πιο κοντά (και επομένως πιο παρόμοια) από οποιαδήποτε άλλα σημεία δεδομένων στο σύνολο εκπαίδευσης.

Τα ιστογράμματα στο σχήμα 3.11 δείχνουν τις κατανομές όλων των αποστάσεων ανά ζεύγη μεταξύ τυχαία κατανεμημένων σημείων εντός τετραγώνων μονάδας  $d$ -διαστάσεων. Καθώς ο αριθμός των διαστάσεων  $d$  αυξάνεται, όλες οι αποστάσεις συγκεντρώνονται σε ένα πολύ μικρό εύρος.



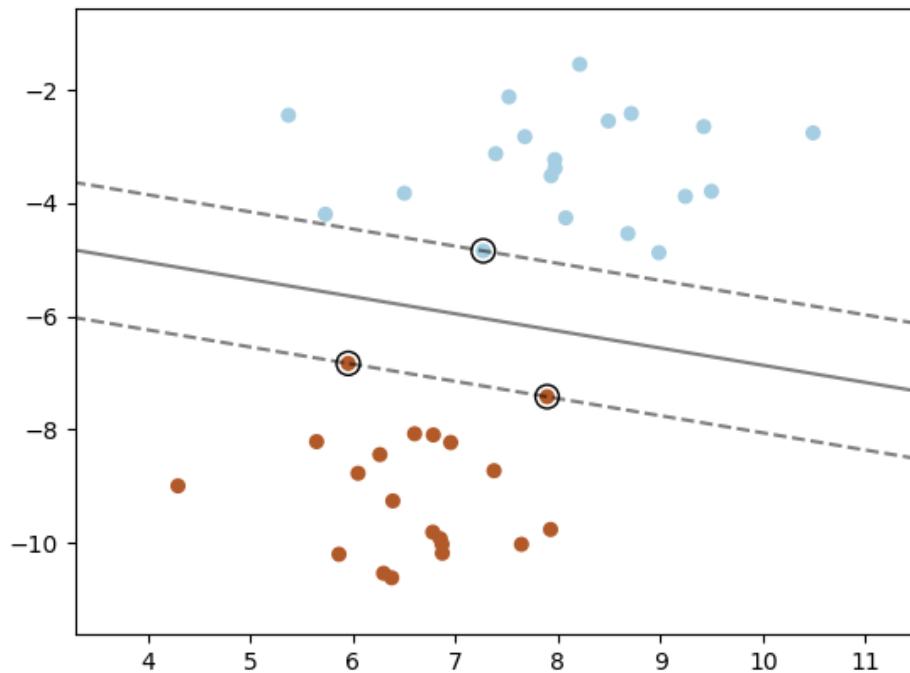
Σχήμα 3.11: Η κατάρα της διαστατικότητας [37]

Θα μπορούσε κανείς να σκεφτεί ότι θα βοηθούσε η αύξηση του αριθμού των δειγμάτων εκπαίδευσης,  $n$ , έως ότου οι πλησιέστεροι γείτονες είναι πραγματικά κοντά στο σημείο δοκιμής. Πόσα σημεία δεδομένων θα χρειαζόταν τέτοια ώστε το  $\ell$  να γίνει πραγματικά μικρό; Το  $\ell = \frac{1}{10} = 0.1 \Rightarrow n = \frac{k}{\ell^d} = k \cdot 10^d$ , το οποίο αυξάνεται εκθετικά! Για  $d > 100$  θα χρειαζόταν πολύ περισσότερα σημεία δεδομένων από ό,τι υπάρχουν ηλεκτρόνια στο σύμπαν...

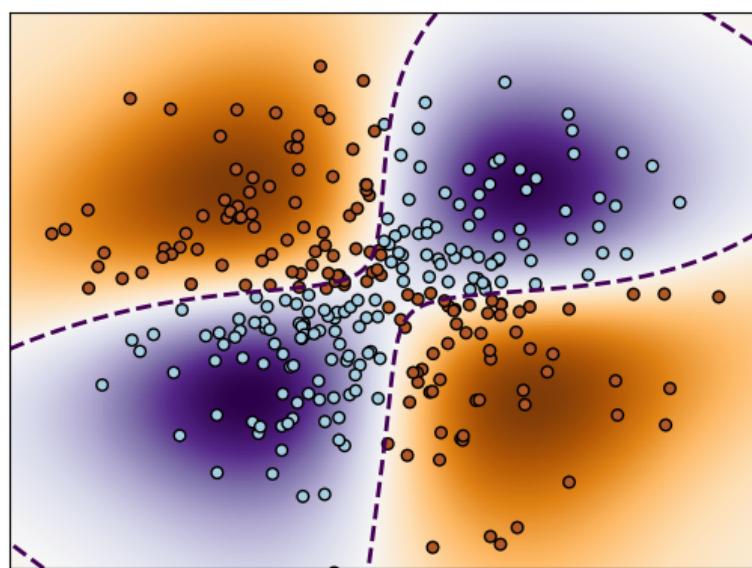
### 3.2.5 Support Vector Machine

Μια **Μηχανή Διανυσμάτων Υποστήριξης** [38] κατασκευάζει ένα υπερ-επίπεδο ή ένα σύνολο υπερ-επίπεδων σε ένα χώρο υψηλών ή άπειρων διαστάσεων, το οποίο μπορεί να χρησιμοποιηθεί για ταξινόμηση, παλινδρόμηση ή άλλα προβλήματα. Διαστηματικά, ένας καλός διαχωρισμός επιτυγχάνεται από το υπερ-επίπεδο που έχει τη μεγαλύτερη απόσταση από τα πλησιέστερα σημεία δεδομένων εκπαίδευσης οποιασδήποτε κατηγορίας (το λεγόμενο λειτουργικό περιθώριο), αφού γενικά όσο μεγαλύτερο είναι το περιθώριο τόσο μικρότερο είναι το σφάλμα γενίκευσης του ταξινομητή. Το σχήμα 3.12 δείχνει τη συνάρτηση απόφασης για ένα γραμμικά διαχωρίσιμο πρόβλημα, με τρία δείγματα στα όρια του περιθώριού, που ονομάζονται «διανύσματα υποστήριξης»:

Γενικά, όταν το πρόβλημα δεν είναι γραμμικά διαχωρίσιμο, τα διανύσματα υποστήριξης είναι τα δείγματα εντός των ορίων του περιθώριού όπως φαίνεται στο σχήμα 3.13.



Σχήμα 3.12: Λειτουργικά περιθώρια των Μηχανών Διανυσμάτων Γύποστήριξης [39]



Σχήμα 3.13: Δυαδική ταξινόμηση χρησιμοποιώντας μη γραμμικό SVC με πυρήνα RBF [40]

Ο χρωματικός χάρτης απεικονίζει τη συνάρτηση απόφασης που μαθαίνει το SVC.

## Συναρτήσεις Πυρήνα

Η συνάρτηση πυρήνα είναι μια μέθοδος που χρησιμοποιείται για την μετατροπή των δεδομένων εισόδου στην απαιτούμενη μορφή ανάλογα με το πρόβλημα. Ο "Πυρήνας" (Kernel) χρησιμοποιείται λόγω ενός συνόλου μαθηματικών συναρτήσεων που χρησιμοποιούνται στις Μηχανές Διανυσμάτων Γύποστήριξης για τον χειρισμό των δεδομένων. Έτσι, η συνάρτηση πυρήνα γενικά μετασχηματίζει τα δεδομένα από το σετ εκπαίδευσης έτσι ώστε μια μη γραμμική επιφάνεια απόφασης να μπορεί να μετασχηματιστεί σε μια γραμμική εξίσωση σε μεγαλύτερο αριθμό χωρικών διαστάσεων. Βασικά, επιστρέφει το εσωτερικό γινόμενο μεταξύ δύο σημείων σε μια τυπική διάσταση χαρακτηριστικών. Η συνάρτηση πυρήνα μπορεί να είναι οποιοδήποτε από τα παρακάτω :

- Γραμμική :  $\langle x, x' \rangle$ .
- Πολυωνυμική : Αντιπροσωπεύει την ομοιότητα των διανυσμάτων στο σετ εκπαίδευσης σε ένα χώρο χαρακτηριστικών πάνω από τα πολυώνυμα των αρχικών μεταβλητών που χρησιμοποιούνται στον πυρήνα.  $(\gamma \langle x, x' \rangle + r)^d$ , όπου το  $d$  ορίζεται από την παράμετρο βαθμού του πολυωνύμου  $r$ .
- Συνάρτηση ακτινικής βάσης πυρήνα Gauss (RBF) : Χρησιμοποιείται όταν δεν υπάρχει προηγούμενη γνώση σχετικά με τα δεδομένα.  $\exp(-\gamma \|x - x'\|^2)$ , με  $\gamma > 0$ .
- Σιγμοειδής : Αυτή η συνάρτηση είναι ισοδύναμη με ένα μοντέλο νευρωνικού δικτύου perceptron δύο επιπέδων, το οποίο χρησιμοποιείται ως συνάρτηση ενεργοποίησης για τους τεχνητούς νευρώνες.  $\tanh(\gamma \langle x, x' \rangle + r)$ .

## Μαθηματική διατύπωση

Δεδομένων των διανυσμάτων εκπαίδευσης  $x_i \in \mathbb{R}^p, i = 1, \dots, n$ , σε δύο κλάσεις, και ένα διάνυσμα  $y \in \{1, -1\}^n$ , ο στόχος είναι να βρεθούν τα  $w \in \mathbb{R}^p$  και  $b \in \mathbb{R}$  τέτοια ώστε η πρόβλεψη που δίνεται από τον τύπο  $\text{sign}(w^T \phi(x) + b)$  να είναι σωστή για τα περισσότερα δείγματα.

Το SVC επιλύει το ακόλουθο πρωταρχικό πρόβλημα :

$$\min_{w, b, \zeta} \frac{1}{2} w^T w + C \sum_{i=1}^n \zeta_i$$

που υπόκειται στο  $y_i(w^T \phi(x_i) + b) \geq 1 - \zeta_i$ ,

$$\zeta_i \geq 0, i = 1, \dots, n$$

Η μεγιστοποίηση του περιθωρίου γίνεται διαισθητικά (ελαχιστοποιώντας το  $\|w\|^2 = w^T w$ ), ενώ επιφέρεται ποινή όταν ένα δείγμα ταξινομείται εσφαλμένα ή εντός του ορίου περιθωρίου. Στην ιδανική περίπτωση, η τιμή  $y_i(w^T \phi(x_i) + b)$  θα ήταν  $\geq 1$  για όλα τα δείγματα, γεγονός που υποδεικνύει μια τέλεια πρόβλεψη. Άλλα τα προβλήματα συνήθως δεν είναι πάντα τέλεια διαχωρίσιμα με ένα υπερεπίπεδο, επομένως επιτρέπεται σε ορισμένα δείγματα να βρίσκονται σε απόσταση  $\zeta_i$  από

το σωστό όριο περιθωρίου. Ο όρος ποινής  $C$  ελέγχει την ισχύ αυτής της ποινής και, ως εκ τούτου, λειτουργεί ως παράμετρος αντίστροφης ταχτοποίησης.

Το διπλό πρόβλημα ως προς το πρωταρχικό είναι

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha - e^T \alpha$$

που υπόκειται στο  $y^T \alpha = 0$

$$0 \leq \alpha_i \leq C, i = 1, \dots, n$$

όπου  $e$  είναι το διάνυσμα όλων και  $Q$  είναι ένας  $n$  επί  $n$  θετικός ημικαθορισμένος πίνακας,  $Q_{ij} \equiv y_i y_j K(x_i, x_j)$ , όπου  $K(x_i, x_j) = \phi(x_i)^T \phi(x_j)$  είναι ο πυρήνας. Οι όροι  $\alpha_i$  ονομάζονται διπλοί συντελεστές και περιορίζονται πάνω από το  $C$ . Αυτή η διπλή αναπαράσταση υπογραμμίζει το γεγονός ότι τα διανύσματα εκπαίδευσης αντιστοιχίζονται έμμεσα σε έναν υψηλότερο (ίσως άπειρο) διαστατικό χώρο από τη συνάρτηση  $\phi$ <sup>3</sup>.

Μόλις λυθεί το πρόβλημα βελτιστοποίησης, η έξοδος της συνάρτησης απόφασης για ένα δεδομένο δείγμα  $x$  γίνεται :

$$\sum_{i \in SV} y_i \alpha_i K(x_i, x) + b,$$

και η προβλεπόμενη κλάση αντιστοιχεί στο πρόσημο της. Χρειάζεται μόνο να αθροιστούν τα διανύσματα υποστήριξης (δηλαδή τα δείγματα που βρίσκονται εντός του περιθωρίου) επειδή οι διπλοί συντελεστές  $\alpha_i$  είναι μηδέν για τα άλλα δείγματα.

### 3.2.6 Decision Tree

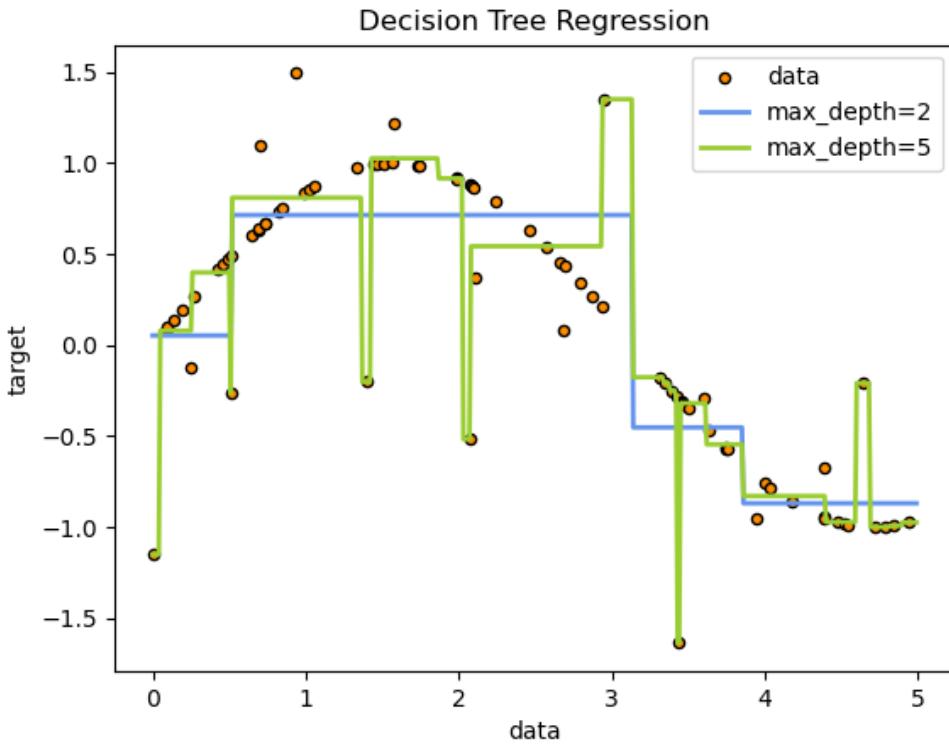
Τα Δέντρα Αποφάσεων (DT) [41] είναι μια μη παραμετρική εποπτευόμενη μέθοδος μάθησης που χρησιμοποιείται για ταξινόμηση και παλινδρόμηση. Ο στόχος είναι να δημιουργηθεί ένα μοντέλο που προβλέπει την τιμή μιας μεταβλητής στόχου (κλάσης) μαθαίνοντας απλούς κανόνες απόφασης που συνάγονται από τα χαρακτηριστικά των δεδομένων. Ένα δέντρο μπορεί να θεωρηθεί ως μια τμηματικά σταθερή προσέγγιση.

Για παράδειγμα, στο σχήμα 3.14, τα δέντρα αποφάσεων μαθαίνουν από δεδομένα να προσεγγίζουν μια καμπύλη ημιτόνου με ένα σύνολο κανόνων απόφασης εάν-τότε-άλλο (if-then-else). Όσο πιο βαθύ είναι το δέντρο, τόσο πιο περίπλοκοι είναι οι κανόνες απόφασης και τόσο πιο προσαρμοσμένο είναι το μοντέλο στα δεδομένα.

Μερικά πλεονεκτήματα των δέντρων απόφασης είναι :

- Απλό στην κατανόηση και στην ερμηνεία. Τα δέντρα μπορούν να οπτικοποιηθούν.

<sup>3</sup>[https://en.wikipedia.org/wiki/Kernel\\_method](https://en.wikipedia.org/wiki/Kernel_method)



Σχήμα 3.14: Παλινδρόμηση Δέντρου Αποφάσεων [42]

- Απαιτεί λίγη προετοιμασία δεδομένων. Άλλες τεχνικές συχνά απαιτούν κανονικοποίηση των δεδομένων, πρέπει να δημιουργηθούν εικονικές μεταβλητές και να αφαιρεθούν κενές τιμές.
- Το κόστος χρήσης του δέντρου (δηλαδή, η πρόβλεψη δεδομένων) είναι λογαριθμικό ως προς τον αριθμό των σημείων δεδομένων που χρησιμοποιούνται για την εκπαίδευση του δέντρου.
- Ικανό να χειρίζεται τόσο αριθμητικά όσο και κατηγοριακά δεδομένα.
- Ικανό να χειριστεί προβλήματα πολλαπλών εξόδων.
- Χρησιμοποιεί ένα μοντέλο "λευκού κουτιού". Εάν μια δεδομένη κατάσταση είναι παρατηρήσιμη σε ένα μοντέλο, η εξήγηση της συνθήκης εξηγείται εύκολα με τη λογική boolean. 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 να ληφθεί υπόψη η αξιοπιστία του μοντέλου.
- Αποδίδει καλά ακόμα κι αν οι παραδοχές του παραβιάζονται κάπως από το αληθινό μοντέλο από το οποίο δημιουργήθηκαν τα δεδομένα.

Μερικά μειονεκτήματα των δέντρων απόφασης είναι :

- Οι μαθητές (learners) του δέντρου αποφάσεων μπορούν να δημιουργήσουν υπερβολικά πολύπλοκα δέντρα που δεν γενικεύουν καλά τα δεδομένα. Αυτό ονομάζεται υπερπροσαρμογή. Μηχανισμοί όπως το "κλάδεμα", δηλαδή 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 στο σχήμα 3.14. Επομένως, δεν είναι καλά στην παρέκταση<sup>4</sup>.
- Οι πρακτικοί αλγόριθμοι μάθησης δέντρων αποφάσεων βασίζονται σε ευρετικούς αλγόριθμους όπως ο αλγόριθμος greedy όπου λαμβάνονται τοπικά βέλτιστες αποφάσεις σε κάθε κόμβο. Τέτοιοι αλγόριθμοι δεν μπορούν να εγγυηθούν ότι θα επιστρέψουν το συνολικά βέλτιστο δέντρο αποφάσεων. Αυτό μπορεί να μετριαστεί με την εκπαίδευση πολλών δέντρων σε έναν εκπαιδευόμενο συνόλου (ensemble learner), όπου τα χαρακτηριστικά και τα δείγματα δειγματοληπτούνται τυχαία με αντικατάσταση.
- Υπάρχουν έννοιες που είναι δύσκολο να μαθευτούν από τα δέντρα αποφάσεων, επειδή αυτά δεν τις εκφράζουν εύκολα, όπως προβλήματα XOR, ισοτιμίας ή πολυπλέκτη.
- Οι μαθητές (learners) του δέντρου αποφάσεων δημιουργούν προκατειλημμένα δέντρα εάν κυριαρχούν ορισμένες κλάσεις. Επομένως, συνιστάται η εξισορόπηση του συνόλου δεδομένων πριν από την προσαρμογή με το δέντρο αποφάσεων.

### Μαθηματική διατύπωση

Με δεδομένο το διάνυσμα εκπαίδευσης  $x_i \in R^n$ ,  $i = 1, \dots, l$  και το διάνυσμα της κλάσης  $y \in R^l$ , ένα δέντρο αποφάσεων [31] χωρίζει αναδρομικά τον χώρο χαρακτηριστικών έτσι ώστε τα δείγματα με τις ίδιες "ετικέτες"<sup>5</sup> (labels) ή παρόμοιες τιμές στόχου<sup>6</sup> (target) να ομαδοποιούνται μαζί.

Ας υποτεθεί ότι τα δεδομένα στον κόμβο  $m$  αντιπροσωπεύονται από το  $Q_m$  με  $n_m$  δείγματα. Για κάθε υποψήφιο διαχωρισμό  $\theta = (j, t_m)$  που αποτελείται από ένα

<sup>4</sup><https://en.wikipedia.org/wiki/Extrapolation>

<sup>5</sup>Ετικέτα (label) : αλγηθινό αποτέλεσμα του στόχου. Στην εποπτευόμενη μάθηση, οι ετικέτες στόχου είναι γνωστές για το σύνολο δεδομένων εκπαίδευσης αλλά όχι για τα δεδομένα ελέγχου.

<sup>6</sup>Στόχος (target) : τελικό αποτέλεσμα που προσπαθεί το μοντέλο να προβλέψει, γνωστό και ως  $y$ . Μπορεί να είναι κατηγοριακός (π.χ. καλοί/θης ή κακοί/θης καρκίνος) ή συνεχής (π.χ. τιμή σπιτιού).

χαρακτηριστικό  $j$  και ένα κατώφλι  $t_m$ , τα δεδομένα διαχωρίζονται σε  $Q_m^{left}(\theta)$  και  $Q_m^{right}(\theta)$  υποσύνολα

$$Q_m^{left}(\theta) = \{(x, y) | x_j \leq t_m\}$$

$$Q_m^{right}(\theta) = Q_m \setminus Q_m^{left}(\theta)$$

Η ποιότητα ενός υποψήφιου διαχωρισμού ενός κόμβου  $m$  υπολογίζεται στη συνέχεια χρησιμοποιώντας μια συνάρτηση ακαθαρσίας (impurity function) ή συνάρτηση απώλειας  $H()$ , η επιλογή της οποίας εξαρτάται από την εργασία που επιλύεται (ταξινόμηση ή παλινδρόμηση)

$$G(Q_m, \theta) = \frac{n_m^{left}}{n_m} H(Q_m^{left}(\theta)) + \frac{n_m^{right}}{n_m} H(Q_m^{right}(\theta))$$

Οι παράμετροι που πρέπει να επιλεχθούν, πρέπει να ελαχιστοποιούν την ακαθαρσία

$$\theta^* = \operatorname{argmin}_{\theta} G(Q_m, \theta)$$

Αναδρομικά για τα υποσύνολα  $Q_m^{left}(\theta^*)$  και  $Q_m^{right}(\theta^*)$  έως ότου επιτευχθεί το μέγιστο επιτρεπόμενο βάθος,  $n_m < \min_{samples}$  ή  $n_m = 1$ .

### Κριτήρια ταξινόμησης

Εάν ένας στόχος είναι ένα αποτέλεσμα ταξινόμησης που παίρνει τιμές  $0, 1, \dots, K-1$ , για τον κόμβο  $n$ , έστω

$$p_{mk} = \frac{1}{n_m} \sum_{y \in Q_m} I(y = k)$$

είναι η αναλογία των παρατηρήσεων της κλάσης  $k$  στον κόμβο  $m$ . Εάν ο  $m$  είναι τερματικός κόμβος, η πιθανότητα για αυτήν την περιοχή ορίζεται σε  $p_{mk}$ . Κοινές μετρικές της ακαθαρσίας είναι οι ακόλουθες.

$$\text{Gini} : H(Q_m) = \sum_k p_{mk}(1 - p_{mk})$$

$$\text{Log Loss ή Εντροπία} : H(Q_m) = - \sum_k p_{mk} \log(p_{mk})$$

**Σημείωση:** Το κριτήριο της εντροπίας υπολογίζει την εντροπία Shannon των πιθανών κλάσεων. Ως πιθανότητά τους παίρνει τις συχνότητες της κλάσης των σημείων από τα δεδομένα εκπαίδευσης που έχουν φτάσει σε ένα συγκεκριμένο φύλλο  $m$ . Η χρήση της εντροπίας Shannon ως κριτήριο διαχωρισμού κόμβου δέντρου ισοδυναμεί με την ελαχιστοποίηση της λογαριθμικής απώλειας (log loss) (επίσης γνωστή ως διασταυρούμενη εντροπία και πολυωνυμική απόκλιση) μεταξύ των πραγματικών "ετικετών"  $y_i$  και των πιθανολογικών προβλέψεων  $T_k(x_i)$  του μοντέλου δέντρου  $T$  για την κλάση  $k$ . Η λογαριθμική απώλεια ενός μοντέλου δέντρου  $T$  που υπολογίζεται σε ένα σύνολο δεδομένων  $D$  ορίζεται ως εξής :

$$\text{LL}(D, T) = -\frac{1}{n} \sum_{(x_i, y_i) \in D} \sum_k I(y_i = k) \log(T_k(x_i))$$

όπου  $D$  είναι το σύνολο δεδομένων εκπαίδευσης από  $n$  ζευγάρια  $(x_i, y_i)$ .

Σε ένα δέντρο ταξινόμησης, οι προβλεπόμενες πιθανότητες κλάσης εντός των κόμβων των φύλλων είναι σταθερές, δηλαδή: για κάθε  $(x_i, y_i) \in Q_m$ , έχει :  $T_k(x_i) = p_{mk}$  για κάθε κλάση  $k$ .

Αυτή η ιδιότητα καθιστά δυνατή την επανεγγραφή του  $\text{LL}(D, T)$  ως το άθροισμα των εντροπιών Shannon που υπολογίζεται για κάθε φύλλο του δέντρου  $T$  του σταθμισμένου με τον αριθμό των σημείων δεδομένων εκπαίδευσης που έφτασαν σε κάθε φύλλο :

$$\text{LL}(D, T) = \sum_{m \in T} \frac{n_m}{n} H(Q_m)$$

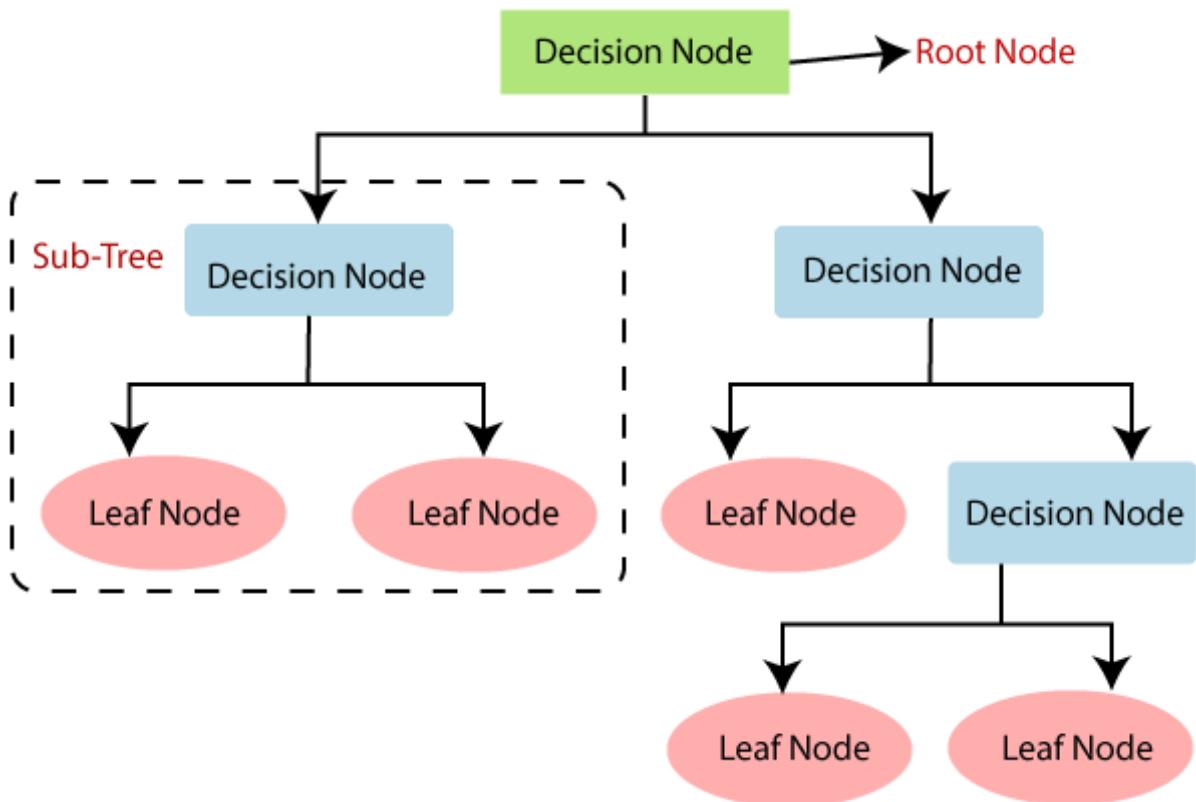
### Κλάδεμα ελάχιστου κόστους-πολυπλοκότητας

Το κλάδεμα ελάχιστου κόστους-πολυπλοκότητας είναι ένας αλγόριθμος που χρησιμοποιείται για το κλάδεμα ενός δέντρου για να αποφευχθεί η υπερπροσαρμογή [41] (overfitting). Αυτός ο αλγόριθμος παραμετροποιείται με την  $\alpha \geq 0$ , γνωστή ως παράμετρος πολυπλοκότητας. Η παράμετρος πολυπλοκότητας χρησιμοποιείται για τον καθορισμό του μέτρου κόστους-πολυπλοκότητας,  $R_\alpha(T)$  ενός δεδομένου δέντρου  $T$  :

$$R_\alpha(T) = R(T) + \alpha |\tilde{T}|$$

όπου  $|\tilde{T}|$  είναι ο αριθμός των τερματικών κόμβων μέσα στο δέντρο  $T$  και  $R(T)$  παραδοσιακά ορίζεται ως ο συνολικός ρυθμός εσφαλμένης ταξινόμησης των τερματικών κόμβων. Εναλλακτικά, μπορεί να χρησιμοποιηθεί η συνολική σταθμισμένη "ακαθαρσία" (impurity) δείγματος των τερματικών κόμβων για  $R(T)$ . Όπως φαίνεται παραπάνω, η ακαθαρσία ενός κόμβου εξαρτάται από το κριτήριο. Το κλάδεμα ελάχιστου κόστους-πολυπλοκότητας βρίσκει το υποδέντρο του  $T$  που ελαχιστοποιεί το  $R_\alpha(T)$ .

Το μέτρο πολυπλοκότητας κόστους ενός μεμονωμένου κόμβου είναι  $R_\alpha(t) = R(t) + \alpha$ . Το κλαδί,  $T_t$ , ορίζεται ότι είναι ένα δέντρο όπου ο κόμβος  $t$  είναι η ρίζα του. Γενικά, η ακαθαρσία ενός κόμβου είναι μεγαλύτερη από το άθροισμα των ακαθαρσιών των τερματικών κόμβων του,  $R(T_t) < R(t)$ . Ωστόσο, το μέτρο πολυπλοκότητας κόστους ενός κόμβου,  $t$ , και του κλαδιού του,  $T_t$ , μπορεί να είναι ίσο ανάλογα με το  $\alpha$ . Το αποτελεσματικό  $\alpha$  ενός κόμβου ορίζεται ως η τιμή όπου είναι ίσα,  $R_\alpha(T_t) = R_\alpha(t)$  ή  $\alpha_{eff}(t) = \frac{R(t) - R(T_t)}{|T|-1}$ . Ένας μη τερματικός κόμβος με τη μικρότερη τιμή  $\alpha_{eff}$  είναι ο πιο αδύναμος κρίκος και θα "κλαδευτεί". Αυτή η διαδικασία σταματά όταν το ελάχιστο του κλαδευμένου δέντρου  $\alpha_{eff}$  είναι μεγαλύτερο από την παράμετρο  $\alpha$ .



Σχήμα 3.15: Δομή Δέντρου Αποφάσεων [43]

### 3.2.7 Random Forest

#### Μέθοδοι Bagging

Στους αλγόριθμους συνόλου (ensemble), οι μέθοδοι Bagging (βγαίνει από τα αρχικά του Bootstrap Aggregation ή συνάθροιση bootstrap) σχηματίζουν μια κατηγορία αλγορίθμων που φτιάχνουν πολλές περιπτώσεις ενός εκτιμητή μαύρου κουτιού σε τυχαία υποσύνολα του αρχικού συνόλου εκπαίδευσης και στη συνέχεια συγκεντρώνουν τις μεμονωμένες προβλέψεις τους για να σχηματίσουν μια τελική πρόβλεψη όπως φαίνεται στο σχήμα 3.16. Αυτές οι μέθοδοι χρησιμοποιούνται ως τρόπος μείωσης της διακύμανσης ενός βασικού εκτιμητή (π.χ., ενός δέντρου αποφάσεων), εισάγοντας την τυχαιοποίηση στη διαδικασία κατασκευής του και στη συνέχεια δημιουργώντας ένα σύνολο από αυτό. Σε πολλές περιπτώσεις, οι μέθοδοι bagging αποτελούν έναν πολύ απλό τρόπο βελτίωσης σε σχέση με ένα μεμονωμένο μοντέλο, χωρίς να απαιτείται η προσαρμογή του βασικού αλγορίθμου. Καθώς παρέχουν έναν τρόπο για τη μείωση της υπερπροσαρμογής, αυτές οι μέθοδοι λειτουργούν καλύτερα με ισχυρά και πολύπλοκα μοντέλα (π.χ. πλήρως ανεπτυγμένα δέντρα αποφάσεων), σε αντίθεση με τις μεθόδους ενίσχυσης που συνήθως λειτουργούν καλύτερα με αδύναμα μοντέλα (π.χ. ρηχά δέντρα απόφασης).

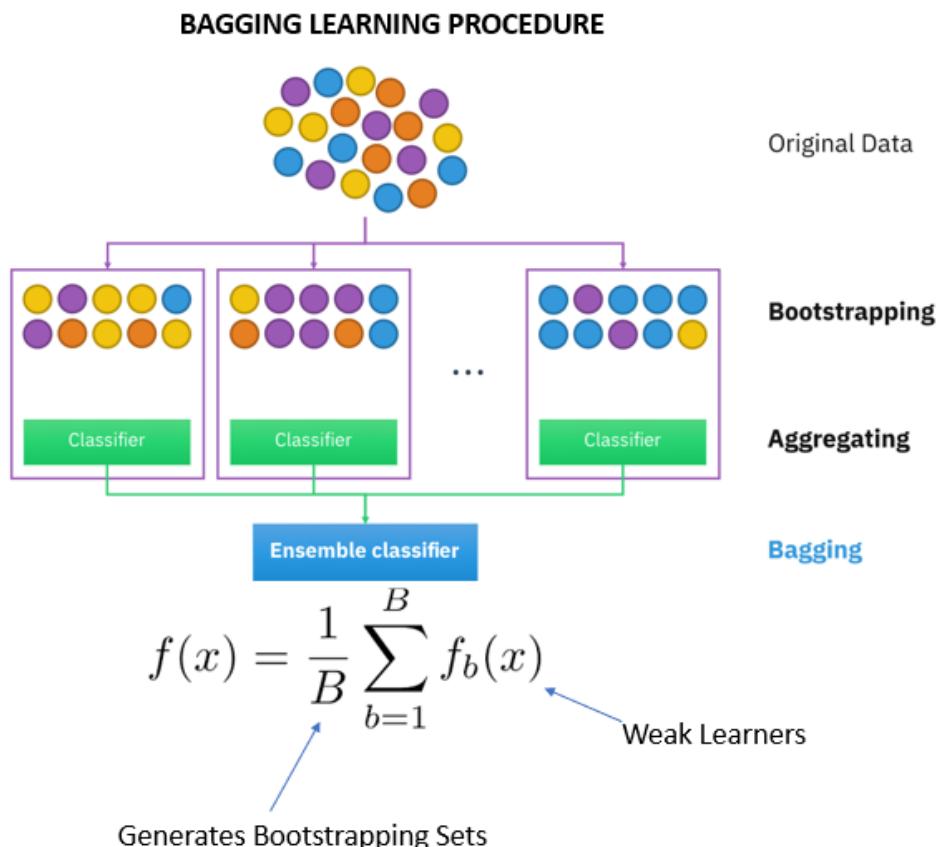
Οι μέθοδοι bagging διαφέρουν μεταξύ τους κυρίως λόγω του τρόπου με τον οποίο βγάζουν τυχαία υποσύνολα από το σετ εκπαίδευσης :

- Όταν τα τυχαία υποσύνολα του συνόλου δεδομένων βγαίνουν ως τυχαία υπο-

σύνολα των δειγμάτων, τότε αυτός ο αλγόριθμος είναι γνωστός ως Επικόλληση (Pasting) [44].

- Όταν τα δείγματα λαμβάνονται με αντικατάσταση, τότε η μέθοδος είναι γνωστή ως Bagging [45].
- Όταν τα τυχαία υποσύνολα του συνόλου δεδομένων βγαίνουν ως τυχαία υποσύνολα των χαρακτηριστικών, τότε η μέθοδος είναι γνωστή ως Τυχαίοι Υποχώροι (Random Subspaces) [46].
- Τέλος, όταν οι βασικοί εκτιμητές εκπαιδεύονται σε υποσύνολα τόσο δειγμάτων όσο και χαρακτηριστικών, τότε η μέθοδος είναι γνωστή ως Τυχαία Τμήματα (Random Patches) [47].

Πιο αναλυτικά, το Bagging αναφέρεται και ως Bootstrapping, το οποίο είναι μια τεχνική επαναδειγματοληψίας που περιλαμβάνει επανειλημμένη λήψη δειγμάτων από τα αρχικά δεδομένα με αντικατάσταση. Ο όρος ”με αντικατάσταση”, σημαίνει ότι το ίδιο σημείο δεδομένων μπορεί να συμπεριληφθεί πολλές φορές στο σύνολο δεδομένων που δημιουργήθηκε με επαναδειγματοληψία.



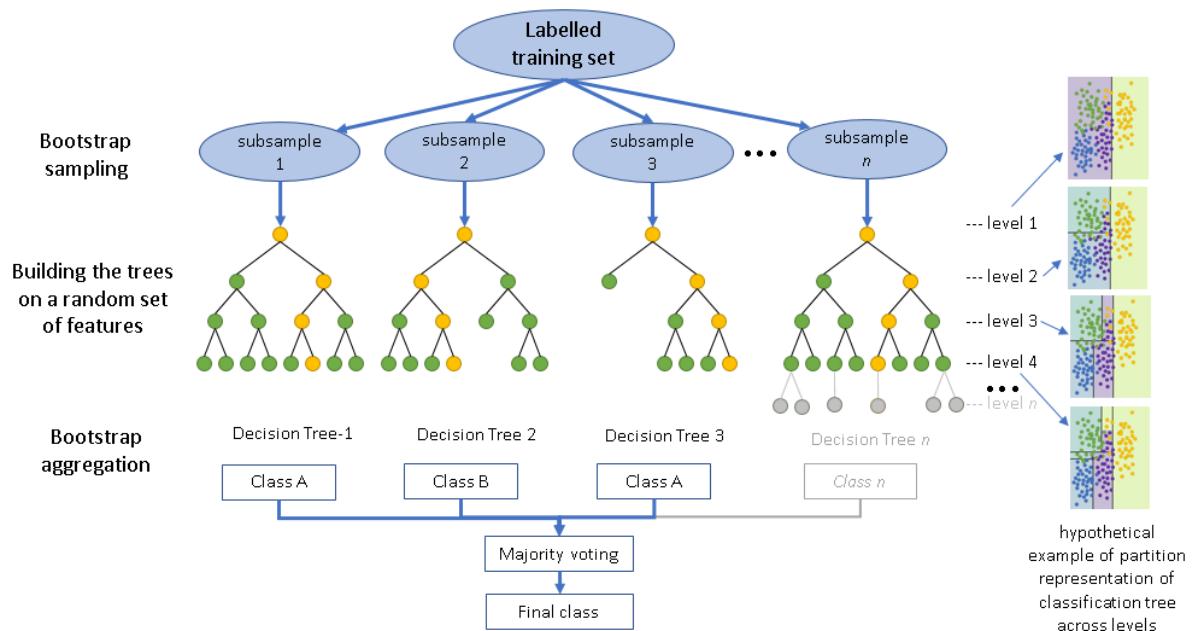
Σχήμα 3.16: Διαδικασία του Bagging [48]

## Τυχαία Δάση

Στα τυχαία δάση, κάθε δέντρο του συνόλου δημιουργείται από ένα δείγμα που έχει τραβηγχτεί με αντικατάσταση (δηλαδή, ένα δείγμα bootstrap) από το σετ εκπαίδευσης.

Επιπλέον, κατά τον διαχωρισμό κάθε κόμβου κατά την κατασκευή ενός δέντρου, ο βέλτιστος διαχωρισμός βρίσκεται είτε από όλα τα χαρακτηριστικά της εισόδου είτε από ένα τυχαίο υποσύνολο αυτών.

Ο σκοπός αυτών των δύο πηγών τυχαιότητας είναι να μειώσουν τη διακύμανση του εκτιμητή δασών. Πράγματι, τα μεμονωμένα δέντρα απόφασης παρουσιάζουν συνήθως υψηλή διακύμανση και τείνουν να υπερπροσαρμόζονται. Αυτή η προστιθέμενη τυχαιότητα στα δάση αποδίδει δέντρα απόφασης με αποσυνδεδεμένα σφάλματα πρόβλεψης. Λαμβάνοντας τον μέσο όρο αυτών των προβλέψεων, ορισμένα σφάλματα μπορούν να ακυρωθούν. Τα τυχαία δάση επιτυγχάνουν μειωμένη διακύμανση συνδυάζοντας διαφορετικά δέντρα όπως φαίνεται στο σχήμα 3.17, μερικές φορές με το κόστος μιας μικρής αύξησης της προκατάληψης. Στην πράξη, η μείωση της διακύμανσης είναι συχνά σημαντική, οπότε προκύπτει ένα συνολικά καλύτερο μοντέλο.

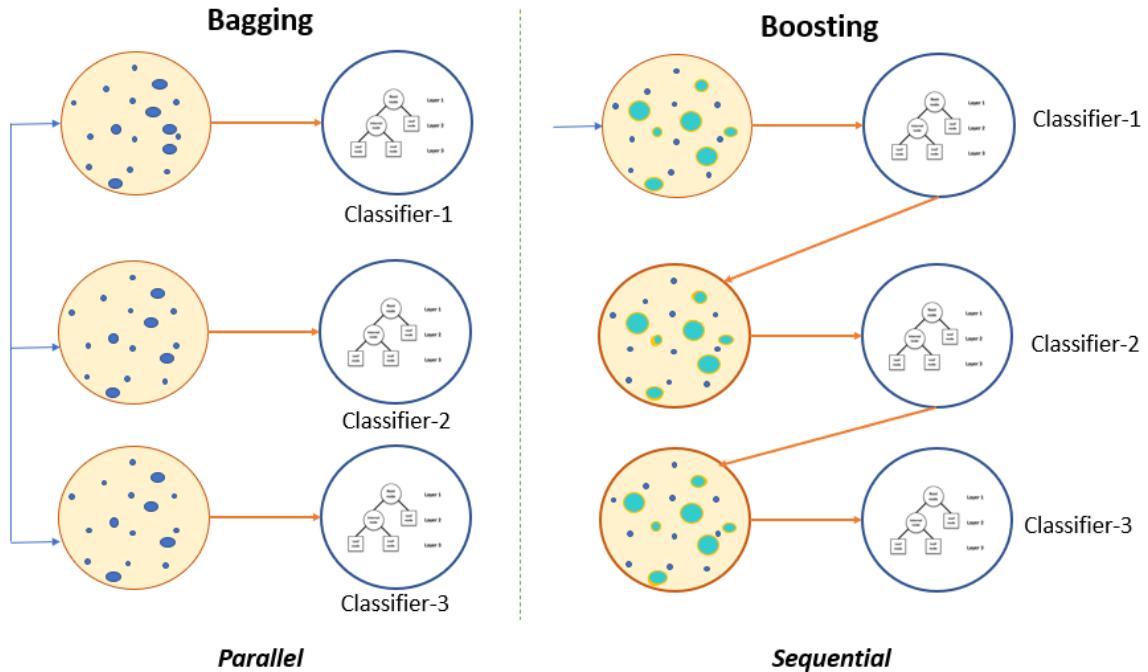


Σχήμα 3.17: Τυχαία Δάση [49]

## 3.2.8 Adaptive Boosting

Το Boosting (ενίσχυση) είναι μια πολύ "ισχυρή" ιδέα μάθησης που εισήχθη για πρώτη φορά το 1997 από τους Freund και Schapire [50]. Αρχικά σχεδιάστηκε για προβλήματα ταξινόμησης, αλλά αργότερα μελετήθηκε και επεκτάθηκε και στην παλινδρόμηση. Το κίνητρο για αυτή την προσέγγιση ήταν μια διαδικασία που να συνδυάζει τα αποτελέσματα πολλών αδύναμων ταξινομητών (ή μαθητών) για να

δημιουργήσει μια ισχυρή «επιτροπή» όπως φαίνεται στο σχήμα 3.19. Από αυτή την άποψη, φαίνεται πως η ενίσχυση έχει ομοιότητα με το bagging (το οποίο αναφέρθηκε στην ενότητα 3.2.7) και άλλες προσεγγίσεις που βασίζονται σε επιτροπές, όμως στην πραγματικότητα, η σύνδεσή τους είναι στην καλύτερη περίπτωση επιφανειακή και η ενίσχυση είναι θεμελιωδώς διαφορετική (σχήμα 3.18).



Σχήμα 3.18: Βασική Διαφορά Bagging και Boosting [48]

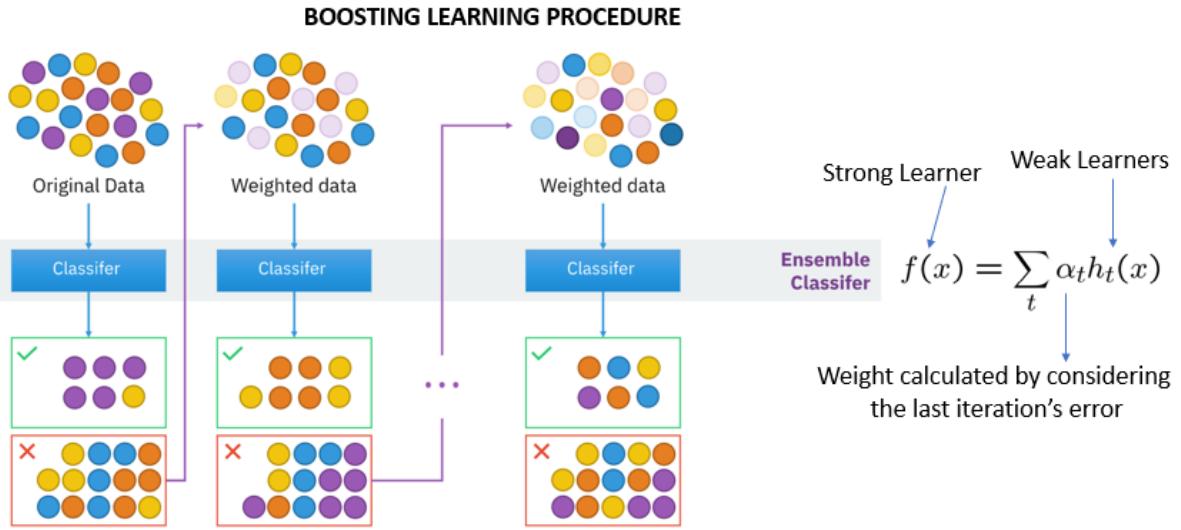
Όπως φαίνεται και στο σχήμα η διαδικασία της εκπαίδευσης των ταξινομητών στο Bagging γίνεται παράλληλα, ενώ στην Ενίσχυση γίνεται διαδοχικά.

Σε αυτό το σημείο θα περιγραφεί ο πιο δημοφιλής αλγόριθμος ενίσχυσης, που πρωτοδημοσιεύτηκε από τους ίδιους, και ονομάζεται "AdaBoost.M1" [51]. Ας υποτεθεί ένα πρόβλημα δύο κλάσεων, με τη μεταβλητή εξόδου κωδικοποιημένη ως  $Y \in \{-1, 1\}$ . Δεδομένου ενός διανύσματος μεταβλητών πρόβλεψης  $X$ , ένας ταξινομητής  $G(X)$  παράγει μια πρόβλεψη λαμβάνοντας μία από τις δύο τιμές  $\{-1, 1\}$ . Το ποσοστό σφάλματος στο δείγμα εκπαίδευσης είναι

$$\overline{err} = \frac{1}{N} \sum_{i=1}^N I(y_i \neq G(x_i)),$$

και το αναμενόμενο ποσοστό σφάλματος στις μελλοντικές προβλέψεις είναι  $E_{XY} I(Y \neq G(X))$ .

Αδύναμος ταξινομητής είναι αυτός του οποίου το ποσοστό σφάλματος είναι ελαφρώς καλύτερο από την τυχαία εικασία. Ο σκοπός της ενίσχυσης είναι η διαδοχική εφαρμογή του ασθενούς αλγόριθμου ταξινόμησης σε επανείλημμένα τροποποιημένες εκδόσεις των δεδομένων, παράγοντας έτσι μια ακολουθία αδύναμων ταξινομητών  $G_m(x)$ ,  $m = 1, 2, \dots, M$ . Στη συνέχεια, οι προβλέψεις από όλους αυτούς τους



Σχήμα 3.19: Διαδικασία της Ενίσχυσης [48]

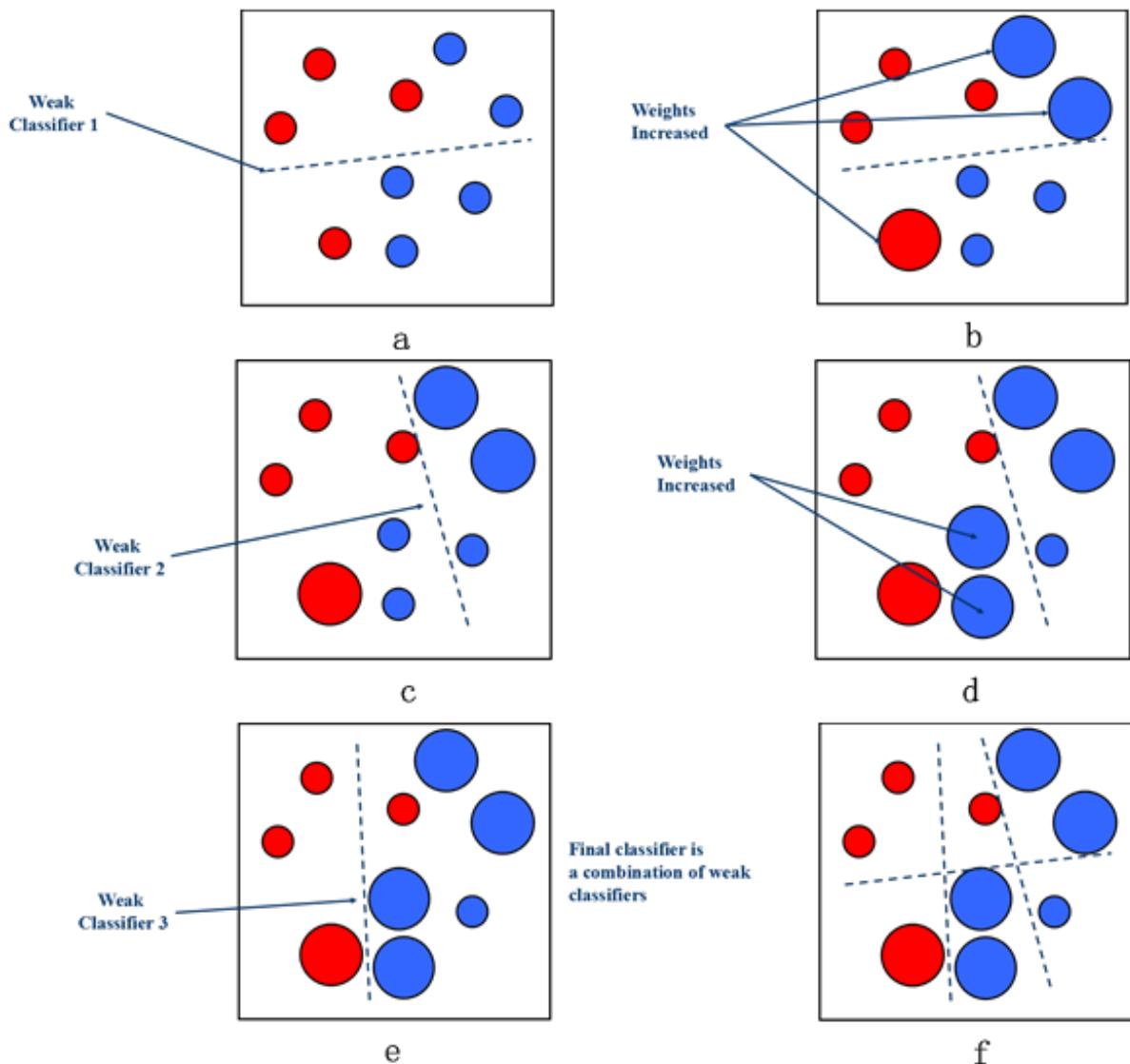
αδύναμους ταξινομητές συνδυάζονται με σταθμισμένη πλειοψηφία για να παραχθεί η τελική πρόβλεψη :

$$G(x) = \text{sign} \left( \sum_{m=1}^M a_m G_m(x) \right).$$

Εδώ τα  $a_1, a_2, \dots, a_M$  υπολογίζονται από τον αλγόριθμο ενίσχυσης και σταθμίζουν τη συνεισφορά κάθε αντίστοιχου  $G_m(x)$  [31]. Η επίδρασή τους είναι να δίνουν μεγαλύτερη επιρροή στους πιο ακριβείς ταξινομητές στην ακολουθία. Στο σχήμα 3.20, φαίνεται ότι οι ταξινομητές εκπαιδεύονται σε σταθμισμένες εκδόσεις του συνόλου δεδομένων και στη συνέχεια συνδυάζονται για να παράγουν μια τελική πρόβλεψη.

Οι τροποποιήσεις δεδομένων σε κάθε βήμα ενίσχυσης συνίστανται στην εφαρμογή βαρών  $w_1, w_2, \dots, w_N$  σε κάθε μία από τις παρατηρήσεις εκπαίδευσης  $(x_i, y_i), i = 1, 2, \dots, N$ . Αρχικά όλα τα βάρη ορίζονται σε  $w_i = 1/N$ , έτσι ώστε το πρώτο βήμα απλά να εκπαιδεύει τον ταξινομητή στα δεδομένα με τον συνήθη τρόπο. Για κάθε διαδοχική επανάληψη  $m = 2, 3, \dots, M$  τα βάρη παρατήρησης τροποποιούνται ξεχωριστά και ο αλγόριθμος ταξινόμησης εφαρμόζεται ξανά στις σταθμισμένες παρατηρήσεις. Στο βήμα  $m$ , εκείνες οι παρατηρήσεις που ταξινομήθηκαν λανθασμένα από τον ταξινομητή  $G_{m-1}(x)$  που προκλήθηκαν στο προηγούμενο βήμα, τα βάρη τους θα αυξηθούν, ενώ τα βάρη μειώνονται για εκείνα που ταξινομήθηκαν σωστά. Έτσι, καθώς προχωρούν οι επαναλήψεις, οι παρατηρήσεις που είναι δύσκολο να ταξινομηθούν σωστά λαμβάνουν ολοένα αυξανόμενη επιρροή. Κάθε διαδοχικός ταξινομητής αναγκάζεται έτσι να επικεντρωθεί σε εκείνες τις παρατηρήσεις εκπαίδευσης που χάνονται από τους προηγούμενους στη σειρά.

Ο αλγόριθμος 3.1 δείχνει τις λεπτομέρειες του αλγόριθμου AdaBoost.M1. Ο τρέχων ταξινομητής  $G_m(x)$  επάγεται στις σταθμισμένες παρατηρήσεις στη γραμμή 2α. Το σταθμισμένο ποσοστό σφάλματος που προκύπτει υπολογίζεται στη γραμμή 2β.



Σχήμα 3.20: Προσαρμοστικός Αλγόριθμος Ενίσχυσης [52]

Η γραμμή 2γ υπολογίζει το βάρος  $a_m$  που δίνεται στο  $G_m(x)$  για την παραγωγή του τελικού ταξινομητή  $G(x)$  (γραμμή 3). Στη γραμμή 2δ τα επιμέρους βάρη καθεμιάς από τις παρατηρήσεις ενημερώνονται για την επόμενη επανάληψη. Οι παρατηρήσεις που ταξινομούνται εσφαλμένα από το  $G_m(x)$  έχουν τα βάρη τους να κλιμακώνονται με έναν παράγοντα  $\exp(a_m)$ , αυξάνοντας τη σχετική επιρροή τους για την επαγωγή του επόμενου ταξινομητή  $G_{m+1}(x)$  στην ακολουθία.

---

### Αλγόριθμος 3.1 AdaBoost.M1.

---

1. Αρχικοποίηση των βαρών των παρατηρήσεων  $w_i = 1/N$ ,  $i = 1, 2, \dots, N$
  2. Για  $m = 1$  μέχρι  $M$ :
    - (α') Προσαρμογή ενός ταξινομητή  $G_m(x)$  στα δεδομένα εκπαίδευσης χρησιμοποιώντας βάρη  $w_i$ .
    - (β') Υπολογισμός του σφάλματος
$$\text{err}_m = \frac{\sum_{i=1}^N w_i I(y_i \neq G_m(x_i))}{\sum_{i=1}^N w_i}.$$
    - (γ') Υπολογισμός του βάρους  $\alpha_m = \log((1 - \text{err}_m)/\text{err}_m)$ .
    - (δ') Ενημέρωση των βαρών  $w_i \leftarrow w_i \cdot \exp[\alpha_m \cdot I(y_i \neq G_m(x_i))]$ ,  $i = 1, 2, \dots, N$ .
  3. Έξοδος  $G(x) = [\sum_{m=1}^M \alpha_m G_m(x)]$ .
- 

### 3.2.9 Stochastic Gradient Descent

Η Στοχαστική Κλίση Καθόδου (SGD) [53] είναι μια απλή αλλά πολύ αποτελεσματική προσέγγιση για την τοποθέτηση γραμμικών ταξινομητών και παλινδρομητών υπό κυρτές συναρτήσεις απώλειας όπως οι (γραμμικές) Μηχανές Διανυσμάτων Υποστήριξης και Λογιστική Παλινδρόμηση. Παρόλο που το SGD υπάρχει εδώ και πολύ καιρό στην κοινότητα της μηχανικής μάθησης, έχει λάβει μεγάλη προσοχή πρόσφατα στο πλαίσιο της εκμάθησης μεγάλης κλιμακας.

Το SGD έχει εφαρμοστεί με επιτυχία σε μεγάλης κλιμακας και "αραιά" (sparse) προβλήματα μηχανικής μάθησης που συναντώνται συχνά στην ταξινόμηση κειμένων και στην επεξεργασία φυσικής γλώσσας. Δεδομένου ότι τα δεδομένα είναι αραιά, οι ταξινομητές σε αυτόν τον αλγόριθμο προσαρμόζονται εύκολα σε προβλήματα με περισσότερα από  $10^5$  παραδείγματα εκπαίδευσης και περισσότερα από  $10^5$  χαρακτηριστικά.

#### Πολυπλοκότητα

Το κύριο πλεονέκτημα του SGD είναι η αποτελεσματικότητά του, η οποία είναι γραμμική ως προς τον αριθμό των παραδειγμάτων εκπαίδευσης. Εάν το  $X$  είναι

ένας πίνακας μεγέθους  $(n, p)$ , η εκπαίδευση έχει κόστος  $O(kn\bar{p})$ , όπου  $k$  είναι ο αριθμός των επαναλήψεων (εποχές) και  $\bar{p}$  είναι ο μέσος αριθμός μη μηδενικών χαρακτηριστικών ανά δείγμα. Ωστόσο, πρόσφατα θεωρητικά αποτελέσματα δείχνουν ότι ο χρόνος εκτέλεσης, για να επιτευχθεί κάποια επιθυμητή ακρίβεια βελτιστοποίησης, δεν αυξάνεται καθώς αυξάνεται το μέγεθος του σετ εκπαίδευσης.

### Μαθηματική διατύπωση

Δίνεται ένα σύνολο παραδειγμάτων εκπαίδευσης  $(x_1, y_1), \dots, (x_n, y_n)$  όπου  $x_i \in \mathbf{R}^m$  και  $y_i \in \mathcal{R}$  ( $y_i \in \{-1, 1\}$  για ταξινόμηση), και στόχος είναι να εκπαιδευτεί μια συνάρτηση γραμμικής βαθμολόγησης  $f(x) = w^T x + b$  με παραμέτρους μοντέλου  $w \in \mathbf{R}^m$  και τομής  $b \in \mathbf{R}$ . Η πρόβλεψη για δυαδική ταξινόμηση γίνεται κοιτάζοντας το πρόσημο της  $f(x)$ . Για να βρεθούν οι παράμετροι του μοντέλου, ελαχιστοποιείται το κανονικοποιημένο σφάλμα εκπαίδευσης που δίνεται από

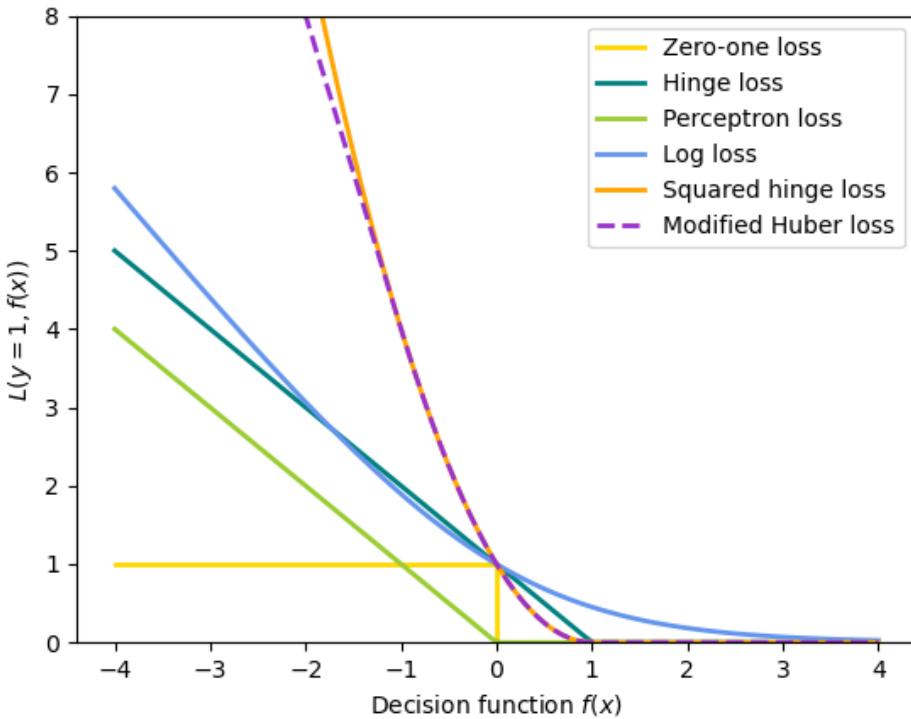
$$E(w, b) = \frac{1}{n} \sum_{i=1}^n L(y_i, f(x_i)) + \alpha R(w)$$

όπου  $L$  είναι μια συνάρτηση απώλειας που μετρά την (λανθασμένη) προσαρμογή του μοντέλου και  $R$  είναι ένας όρος τακτοποίησης (γνωστός και ως ποινή) που τιμωρεί την πολυπλοκότητα του μοντέλου. Το  $\alpha > 0$  είναι μια μη αρνητική υπερ-παράμετρος που ελέγχει την ισχύ τακτοποίησης. Διαφορετικές επιλογές για το  $L$  συνεπάγουν διαφορετικούς ταξινομητές ή παλινδρομητές:

- Hinge (soft-margin): ισοδύναμο με την Ταξινόμηση Διανυσμάτων Υποστήριξης.  $L(y_i, f(x_i)) = \max(0, 1 - y_i f(x_i))$ .
- Perceptron:  $L(y_i, f(x_i)) = \max(0, -y_i f(x_i))$ .
- Modified Huber:  $L(y_i, f(x_i)) = \max(0, 1 - y_i f(x_i))^2$  και  $L(y_i, f(x_i)) = -4y_i f(x_i)$  σε διαφορετική περίπτωση.
- Log Loss: ισοδύναμο με τη Λογιστική Παλινδρόμηση.  $L(y_i, f(x_i)) = \log(1 + \exp(-y_i f(x_i)))$ .
- Squared Error: Γραμμική Παλινδρόμηση (Ridge ή Lasso, εξαρτάται από το  $R$ ).  $L(y_i, f(x_i)) = \frac{1}{2}(y_i - f(x_i))^2$ .
- Huber: λιγότερο ευαίσθητο σε ακραίες τιμές από τα ελάχιστα τετράγωνα. Ισοδυναμεί με τα ελάχιστα τετράγωνα όταν  $|y_i - f(x_i)| \leq \varepsilon$ , και  $L(y_i, f(x_i)) = \varepsilon|y_i - f(x_i)| - \frac{1}{2}\varepsilon^2$  διαφορετικά.
- Epsilon-Insensitive: (soft-margin) ισοδύναμο με την Παλινδρόμηση Διανυσμάτων Υποστήριξης.  $L(y_i, f(x_i)) = \max(0, |y_i - f(x_i)| - \varepsilon)$ .

Όλες οι παραπάνω συναρτήσεις απώλειας μπορούν να θεωρηθούν ως ένα ανώτερο όριο στο σφάλμα λανθασμένης ταξινόμησης (απώλεια μηδέν-ένα) όπως φαίνεται στο σχήμα 3.21.

Μερικές δημοφιλείς επιλογές για τον όρο τακτοποίησης  $R$  (παράμετρος ποινής) περιλαμβάνουν:



Σχήμα 3.21: Συναρτήσεις απώλειας του SGD [54]

- L2 norm:  $R(w) := \frac{1}{2} \sum_{j=1}^m w_j^2 = ||w||_2^2$
- L1 norm:  $R(w) := \sum_{j=1}^m |w_j|$ , που οδηγεί σε αραιές λύσεις.
- Elastic Net:  $R(w) := \frac{\rho}{2} \sum_{j=1}^n w_j^2 + (1 - \rho) \sum_{j=1}^m |w_j|$ , ένας κυρτός συνδυασμός L2 και L1, όπου  $\rho = 1 - l1_{ratio}$  με  $0 \leq l1_{ratio} \leq 1$ .

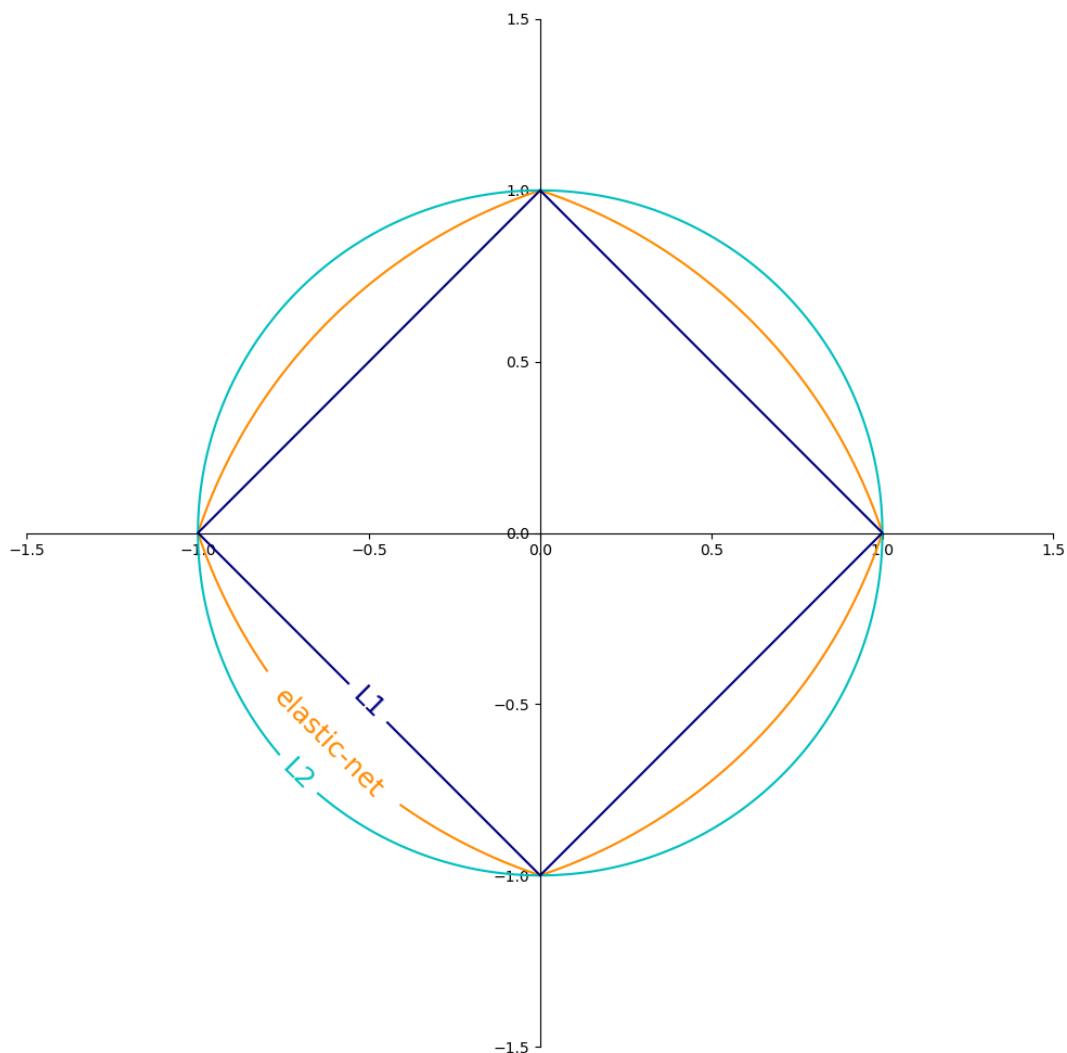
Το σχήμα 3.22 δείχνει τα περιγράμματα των διαφορετικών όρων τακτοποίησης σε έναν δισδιάστατο χώρο παραμέτρων ( $m = 2$ ) όταν  $R(w) = 1$ .

### Λειτουργία του αλγορίθμου SGD

Η στοχαστική κλίση καθόδου είναι μια μέθοδος βελτιστοποίησης για προβλήματα βελτιστοποίησης χωρίς περιορισμούς. Σε αντίθεση με την κλίση καθόδου (παρτίδας), το SGD προσεγγίζει την πραγματική κλίση του  $E(w, b)$  εξετάζοντας ένα μεμονωμένο παράδειγμα εκπαίδευσης κάθε φορά.

Η κλάση SGDClassifier (του sklearn) εφαρμόζει μια ρουτίνα εκμάθησης SGD πρώτης τάξης. Ο αλγόριθμος επαναλαμβάνεται πάνω στα παραδείγματα εκπαίδευσης και για κάθε παράδειγμα ενημερώνει τις παραμέτρους του μοντέλου σύμφωνα με τον κανόνα ενημέρωσης που δίνεται από

$$w \leftarrow w - \eta \left[ \alpha \frac{\partial R(w)}{\partial w} + \frac{\partial L(w^T x_i + b, y_i)}{\partial w} \right]$$



Σχήμα 3.22: Περιγράμματα των διαφορετικών όρων ταχτοποίησης σε έναν δισδιάστατο χώρο παραμέτρων [54]

όπου η είναι ο ρυθμός εκμάθησης που ελέγχει το μέγεθος του βήματος στον χώρο παραμέτρων. Η τομή  $b$  ενημερώνεται με παρόμοιο τρόπο αλλά χωρίς κανονικοποίηση (και με πρόσθετη διάσπαση για αραιούς πίνακες).

Ο ρυθμός μάθησης η μπορεί να είναι είτε σταθερός είτε σταδιακά μειούμενος. Για ταξινόμηση, το προεπιλεγμένο πρόγραμμα ρυθμού εκμάθησης δίνεται από

$$\eta^{(t)} = \frac{1}{\alpha(t_0 + t)}$$

όπου  $t$  είναι το χρονικό βήμα (υπάρχουν συνολικά  $n$  δείγματα \* ή επαναλήψεις = χρονικά βήματα), το  $t_0$  προσδιορίζεται με βάση μια ευρετική<sup>7</sup> (heuristic) που προτείνεται από τον Léon Bottou έτσι ώστε οι αναμενόμενες αρχικές ενημερώσεις να είναι συγχρίσιμες με το αναμενόμενο μέγεθος των βαρών (αυτό υποθέτοντας ότι ο κανόνας των δειγμάτων εκπαίδευσης είναι περίπου 1).

### Παραλλαγές του αλγορίθμου

Υπάρχουν πολλές διαφορετικές παραλλαγές [57] της (στοχαστικής) κλίσης καθόδου, όπως το τράβηγμα ενός δείγματος κάθε φορά με αντικαταστάσεις ή η επανάληψη σε εποχές και το τράβηγμα ενός ή περισσότερων δειγμάτων εκπαίδευσης χωρίς αντικατάσταση. Ο στόχος αυτής της γρήγορης καταγραφής είναι να περιγράψει εν συντομίᾳ τις διαφορετικές προσεγγίσεις.

- **Stochastic Gradient Descent v1**

Έστω ότι

$$\mathcal{D} = (\langle \mathbf{x}^{[1]}, y^{[1]} \rangle, \langle \mathbf{x}^{[2]}, y^{[2]} \rangle, \dots, \langle \mathbf{x}^{[n]}, y^{[n]} \rangle) \in (\mathbb{R}^m \times \{0, 1\})^n$$

είναι το σύνολο δεδομένων που αποτελείται από  $n$  παραδείγματα εκπαίδευσης με χαρακτηριστικά  $x_j^{[i]}$  και στόχους ή ετικέτες κλάσεων  $y^{[i]}$ .

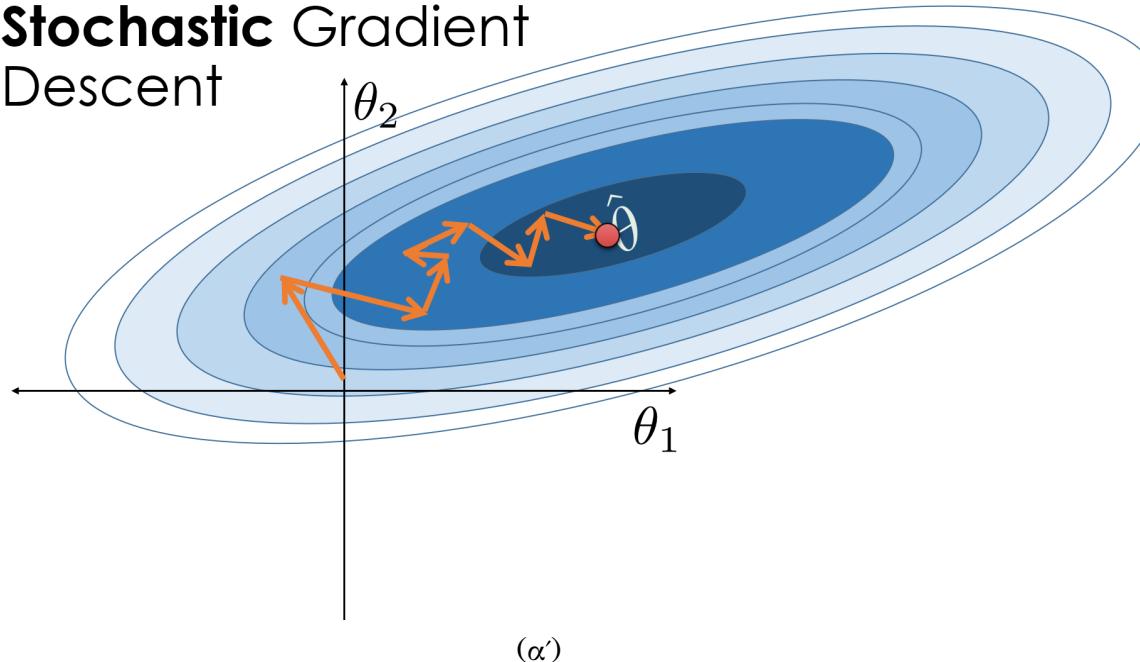
Για την χρήση της «αληθινής» στοχαστικής κλίσης καθόδου, τραβιούνται τυχαία παραδείγματα με αντικατάσταση. Ο φευδοκώδικας φαίνεται στον αλγόριθμο 3.2.

**Σημείωση:** Ενώ αυτή είναι η «πιο στοχαστική» παραλλαγή λόγω της ανεξαρτησίας κατά τη δειγματοληψία, η οποία είναι επομένως η πιο χρησιμη παραλλαγή στο πλαίσιο της Στατιστικής, δεν συνηθίζεται να χρησιμοποιείται καθ' αυτόν τον τρόπο στην Επιστήμη των Υπολογιστών και στη Μηχανική Μάθηση (αυτό είναι πιθανό να οφείλεται σε εμπειρικούς λόγους απόδοσης έναντι στατιστικών εγγυήσεων).

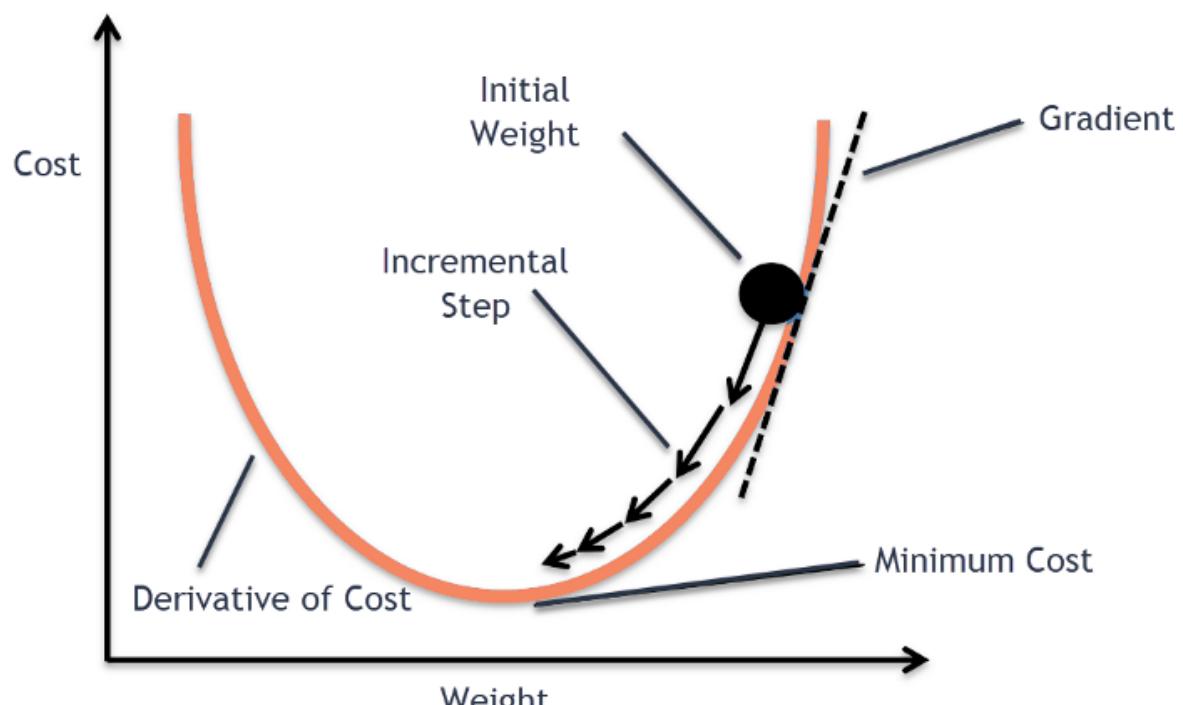
- (“On-line”) **Stochastic Gradient Descent v2**

<sup>7</sup>Μια ευρετική ή ευριστική (από το αρχαίο ελληνικό εύρισκω) είναι οποιαδήποτε προσέγγιση για την επίλυση προβλημάτων ή την αυτοανακάλυψη που χρησιμοποιεί μια πρακτική μέθοδο που δεν είναι εγγυημένη ότι είναι βέλτιστη, τέλεια ή ορθολογική, αλλά είναι ωστόσο επαρκής για να επιτευχθεί ένας άμεσος, βραχυπρόθεσμος στόχος ή προσέγγιση. Όπου η εύρεση της βέλτιστης λύσης είναι αδύνατη ή μη πρακτική, μπορούν να χρησιμοποιηθούν ευρετικές μέθοδοι για να επιταχυνθεί η διαδικασία εύρεσης μιας ικανοποιητικής λύσης. Η ευρετική μπορεί να είναι νοητικές συντομεύσεις που διευκολύνουν το γνωστικό φορτίο της λήψης μιας απόφασης.

## Stochastic Gradient Descent



(α')



(β')

Σχήμα 3.23: (α) Η Στοχαστική Κλίση Καθόδου σε τρισδιάστατο χώρο. [55] (β) Απεικόνιση της διαδικασίας εύρεσης του ελάχιστου κόστους. [56]

---

**Αλγόριθμος 3.2 Stochastic gradient descent v1**


---

```

1: procedure STOCHASTIC GRADIENT DESCENT
2:   αρχικοποίηση:  $w \leftarrow 0^{m-1}$ ,  $b \leftarrow 0$ 
3:   for  $t \in [1, \dots, T]$  do
4:     τράβηγμα ενός τυχαίου δείγματος με αντικατάσταση:  $\langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D}$ 
5:     υπολογισμός πρόβλεψης:  $\hat{y}^{[i]} \leftarrow h(\mathbf{x}^{[i]})$ 
6:     υπολογισμός απώλειας:  $\mathcal{L}^{[i]} \leftarrow L(\hat{y}^{[i]}, y^{[i]})$ 
7:     υπολογισμός των κλίσεων:  $\Delta w \leftarrow -\nabla_{\mathcal{L}^{[i]}} w$ ,  $\Delta b \leftarrow -\frac{\partial \mathcal{L}^{[i]}}{\partial b}$ 
8:     ενημέρωση παραμέτρων:  $w \leftarrow w + \Delta w$ ,  $b \leftarrow +\Delta b$ 
9:   end for
10: end procedure

```

---

Στην πράξη, δεδομένου ότι συνήθως ο πειραματισμός γίνεται με δείγματα σταθερού μεγέθους και πρέπει να αξιοποιηθούν με τον καλύτερο τρόπο όλα τα διαθέσιμα δεδομένα εκπαίδευσης, συνήθως χρησιμοποιείται η έννοια των «εποχών». Στο πλαίσιο της μηχανικής μάθησης, μια εποχή σημαίνει «ένα πέρασμα πάνω από το σύνολο δεδομένων εκπαίδευσης». Συγκεκριμένα, αυτό που διαφέρει από τον προηγούμενο αλγόριθμο 3.2, είναι ότι στις επαναλήψεις που γίνονται στο σετ εκπαίδευσης, τραβιούνται τυχαία δείγματα χωρίς αντικατάσταση. Ο αλγόριθμος είναι ο 3.3:

---

**Αλγόριθμος 3.3 (“On-line”) Stochastic gradient descent v2**


---

```

1: procedure (“ON-LINE”) STOCHASTIC GRADIENT DESCENT
2:   αρχικοποίηση:  $w \leftarrow 0^{m-1}$ ,  $b \leftarrow 0$ 
3:   for εποχή  $\in [1, \dots, E]$  do
4:     ανακάτεμα του  $\mathcal{D}$  για την αποτροπή “κύκλων”
5:     for  $\forall \langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D}$  do
6:       υπολογισμός πρόβλεψης:  $\hat{y}^{[i]} \leftarrow h(\mathbf{x}^{[i]})$ 
7:       υπολογισμός απώλειας:  $\mathcal{L}^{[i]} \leftarrow L(\hat{y}^{[i]}, y^{[i]})$ 
8:       υπολογισμός των κλίσεων:  $\Delta w \leftarrow -\nabla_{\mathcal{L}^{[i]}} w$ ,  $\Delta b \leftarrow -\frac{\partial \mathcal{L}^{[i]}}{\partial b}$ 
9:       ενημέρωση παραμέτρων:  $w \leftarrow w + \Delta w$ ,  $b \leftarrow +\Delta b$ 
10:    end for
11:   end for
12: end procedure

```

---

**Σημείωση:** Αυτή η παραλλαγή δεν ονομάζεται «επίσημα» «on-line» στοχαστική κλίση. Ωστόσο, η προσέγγιση που χρησιμοποιεί εποχές είναι η πιο κοινή παραλλαγή της στοχαστικής κλίσης καθόδου. Επίσης, σε πλαίσιο βιβλιογραφία, ο όρος “on-line” χρησιμοποιείται στο πλαίσιο της κλίσης καθόδου εάν χρησιμοποιείται μόνο ένα παράδειγμα εκπαίδευσης τη φορά για τον υπολογισμό της απώλειας και την ενημέρωση των παραμέτρων.

- (Batch) Gradient Descent

### ΚΕΦΑΛΑΙΟ 3. ΜΗΧΑΝΙΚΗ ΚΑΙ ΒΑΘΙΑ ΜΑΘΗΣΗ

---

Η κατά παρτίδα κλίση καθόδου ή απλώς η ”κλίση καθόδου” είναι η καθοριστική (όχι στοχαστική) παραλλαγή. Εδώ, οι παράμετροι ενημερώνονται σε σχέση με την απώλεια που υπολογίζεται σε όλα τα παραδείγματα εκπαίδευσης. Αν και οι ενημερώσεις δεν είναι θορυβώδεις, γίνεται μόνο μία ενημέρωση ανά εποχή, η οποία μπορεί να είναι λίγο αργή εάν το σύνολο δεδομένων είναι μεγάλο. Ο αλγόριθμος είναι ο 3.4:

---

#### Αλγόριθμος 3.4 (Batch) Gradient Descent

---

```
1: procedure (BATCH) GRADIENT DESCENT
2:   αρχικοποίηση:  $w \leftarrow 0^{m-1}$ ,  $b \leftarrow 0$ 
3:   for εποχή  $\in [1, \dots, E]$  do
4:     ανακάτεμα του  $\mathcal{D}$  για την αποτροπή ”κύκλων”
5:     for  $\forall \langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D}$  do
6:       υπολογισμός πρόβλεψης:  $\hat{y}^{[i]} \leftarrow h(\mathbf{x}^{[i]})$ 
7:     end for
8:     υπολογισμός απώλειας:  $\mathcal{L} \leftarrow \frac{1}{n} \sum_{i=1}^n L(\hat{y}^{[i]}, y^{[i]})$ 
9:     υπολογισμός των κλίσεων:  $\Delta w \leftarrow -\nabla_{\mathcal{L}} w$ ,  $\Delta b \leftarrow -\frac{\partial \mathcal{L}}{\partial b}$ 
10:    ενημέρωση παραμέτρων:  $w \leftarrow w + \Delta w$ ,  $b \leftarrow +\Delta b$ 
11:  end for
12: end procedure
```

---

#### • Minibatch (Stochastic) Gradient Descent v1

Η κλίση καθόδου σε μικρές παρτίδες είναι μια παραλλαγή της στοχαστικής κλίσης καθόδου που προσφέρει μια ωραία αντιστάθμιση μεταξύ των στοχαστικών εκδόσεων που εκτελούν ενημερώσεις με βάση τον αλγόριθμο 3.2 και τον αλγόριθμο 3.3. Εδώ, η απώλεια προσεγγίζεται με βάση ένα μικρότερο δείγμα του σετ εκπαίδευσης, το οποίο επιτρέπει να γίνουν περισσότερες ενημερώσεις ανά εποχή σε σύγκριση με την στοχαστική κλίση παρτίδας. Από την άλλη πλευρά, η προσέγγιση της απώλειας δεν είναι τόσο θορυβώδης όσο στον αλγόριθμο 3.2 και στον αλγόριθμο 3.3, καθώς χρησιμοποιούνται περισσότερα δείγματα εκπαίδευσης. Τέλος, μπορεί επίσης να χρησιμοποιηθεί ένας διανυσματοποιημένος κώδικας (όπως στην κλίση καθόδου παρτίδας).

#### • Minibatch (Stochastic) Gradient Descent v2

Τέλος, η (ενδεχομένως) πιο κοινή παραλλαγή της στοχαστικής κλίσης καθόδου – πιθανώς λόγω ανώτερης (εμπειρικά) απόδοσης – είναι ένας συνδυασμός μεταξύ του αλγορίθμου στοχαστικής κλίσης καθόδου που βασίζεται σε εποχές (αλγόριθμος 3.3) και της κλίσης καθόδου σε μίνι-παρτίδες (αλγόριθμος 3.5). Ο αλγόριθμος είναι ο 3.6:

### 3.2.10 Gradient Tree Boosting

To Gradient Tree Boosting ή Gradient Boosted Decision Trees (GBDT) [58] είναι μια γενίκευση της ενίσχυσης σε αυθαίρετες διαφοροποιήσιμες συναρτήσεις απωλειών. Η GBDT είναι μια ακριβής και αποτελεσματική διαδικασία που μπορεί να χρησιμοποιηθεί τόσο για προβλήματα παλινδρόμησης όσο και για προβλήματα ταξινόμησης σε διάφορους τομείς.

**Αλγόριθμος 3.5 Minibatch (Stochastic) Gradient Descent v1**

```

1: procedure MINIBATCH (STOCHASTIC) GRADIENT DESCENT v1
2:   αρχικοποίηση:  $w \leftarrow 0^{m-1}$ ,  $b \leftarrow 0$ 
3:   for  $t \in [1, \dots, T]$  do
4:     for  $i \in [1, \dots, m]$  (όπου  $m$  το μέγεθος της μίνι-παρτίδας) do
5:       τράβηγμα ενός τυχαίου δείγματος με αντικατάσταση:  $\langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D}$ 
6:     end for
7:     υπολογισμός απώλειας:  $\mathcal{L} \leftarrow \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{[i]}, y^{[i]})$ 
8:     υπολογισμός των κλίσεων:  $\Delta \mathbf{w} \leftarrow -\nabla_{\mathcal{L}} \mathbf{w}$ ,  $\Delta b \leftarrow -\frac{\partial \mathcal{L}}{\partial b}$ 
9:     ενημέρωση παραμέτρων:  $\mathbf{w} \leftarrow \mathbf{w} + \Delta \mathbf{w}$ ,  $b \leftarrow +\Delta b$ 
10:   end for
11: end procedure

```

**Αλγόριθμος 3.6 Minibatch (Stochastic) Gradient Descent v2**

```

1: procedure MINIBATCH (STOCHASTIC) GRADIENT DESCENT v2
2:   αρχικοποίηση:  $w \leftarrow 0^{m-1}$ ,  $b \leftarrow 0$ 
3:   for εποχή  $\in [1, \dots, E]$  do
4:     ανακάτευμα του  $\mathcal{D}$  για την αποτροπή "κύκλων"
5:     for  $i \in [1, \dots, m]$  (όπου  $m$  το μέγεθος της μίνι-παρτίδας) do
6:       τράβηγμα τυχαίου δείγματος χωρίς αντικατάσταση:  $\langle \mathbf{x}^{[i]}, y^{[i]} \rangle \in \mathcal{D}$ 
7:     end for
8:     υπολογισμός απώλειας:  $\mathcal{L} \leftarrow \frac{1}{m} \sum_{i=1}^m L(\hat{y}^{[i]}, y^{[i]})$ 
9:     υπολογισμός των κλίσεων:  $\Delta \mathbf{w} \leftarrow -\nabla_{\mathcal{L}} \mathbf{w}$ ,  $\Delta b \leftarrow -\frac{\partial \mathcal{L}}{\partial b}$ 
10:    ενημέρωση παραμέτρων:  $\mathbf{w} \leftarrow \mathbf{w} + \Delta \mathbf{w}$ ,  $b \leftarrow +\Delta b$ 
11:   end for
12: end procedure

```

### Μαθηματική διατύπωση

Αρχικά θα παρουσιαστεί το GBRT για παλινδρόμηση και στη συνέχεια θα περιγραφεί λεπτομερώς η περίπτωση της ταξινόμησης.

### Regression

Οι παλινδρομητές GBRT είναι προσθετικά μοντέλα των οποίων η πρόβλεψη  $\hat{y}_i$  για μια δεδομένη είσοδο  $x_i$  έχει την ακόλουθη μορφή :

$$\hat{y}_i = F_M(x_i) = \sum_{m=1}^M h_m(x_i)$$

όπου το  $h_m$  είναι εκτιμητές που ονομάζονται αδύναμοι μαθητές στο πλαίσιο της ενίσχυσης. Το Gradient Tree Boosting χρησιμοποιεί παλινδρομητές δέντρων αποφάσεων σταθερού μεγέθους ως αδύναμους μαθητές. Η σταθερά  $M$  αντιστοιχεί στον αριθμό αυτών των αδύναμων μαθητών.

Παρόμοιο με άλλους αλγόριθμους ενίσχυσης, ένα GBRT είναι χτισμένο με άπληστο τρόπο :

$$F_m(x) = F_{m-1}(x) + h_m(x),$$

όπου το δέντρο  $h_m$  που προστέθηκε πρόσφατα τοποθετείται για να ελαχιστοποιηθεί ένα άθροισμα απώλειών  $L_m$ , δεδομένου του προηγούμενου συνόλου  $F_{m-1}$  :

$$h_m = \arg \min_h L_m = \arg \min_h \sum_{i=1}^n l(y_i, F_{m-1}(x_i) + h(x_i)),$$

όπου το  $l(y_i, F(x_i))$  ορίζεται από την παράμετρο της απώλειας.

Από προεπιλογή, το αρχικό μοντέλο  $F_0$  επιλέγεται ως η σταθερά που ελαχιστοποιεί την απώλεια: για απώλεια ελαχίστων τετραγώνων, αυτός είναι ο εμπειρικός μέσος όρος των τιμών-στόχων. Το αρχικό μοντέλο μπορεί επίσης να καθοριστεί μέσω του ορίσματος `init`.

Χρησιμοποιώντας μια προσέγγιση Taylor πρώτης τάξης, η τιμή του  $l$  μπορεί να προσεγγιστεί ως εξής:

$$l(y_i, F_{m-1}(x_i) + h_m(x_i)) \approx l(y_i, F_{m-1}(x_i)) + h_m(x_i) \left[ \frac{\partial l(y_i, F(x_i))}{\partial F(x_i)} \right]_{F=F_{m-1}}.$$

**Σημείωση:** Εν συντομίᾳ, μια προσέγγιση Taylor πρώτης τάξης λέει ότι  $l(z) \approx l(a) + (z - a) \frac{\partial l(a)}{\partial a}$ . Εδώ, το  $z$  αντιστοιχεί σε  $F_{m-1}(x_i) + h_m(x_i)$ , και το  $a$  αντιστοιχεί στο  $F_{m-1}(x_i)$ .

Η ποσότητα  $\left[ \frac{\partial l(y_i, F(x_i))}{\partial F(x_i)} \right]_{F=F_{m-1}}$  είναι η παράγωγος της απώλειας ως προς τη δεύτερη παράμετρό της, που υπολογίζεται ως  $F_{m-1}'(x)$ . Είναι εύκολο να υπολογιστεί για οποιοδήποτε δεδομένο  $F_{m-1}(x_i)$  σε κλειστή μορφή αφού η απώλεια είναι διαφοροποιήσιμη. Συμβολίζεται ως  $g_i$ .

Αφαιρώντας τους σταθερούς όρους, έχουμε :

$$h_m \approx \arg \min_h \sum_{i=1}^n h(x_i) g_i$$

Αυτό ελαχιστοποιείται εάν το  $h(x_i)$  προσαρμόζεται για να προβλέψει μια τιμή που είναι ανάλογη με την αρνητική κλίση  $-g_i$ . Επομένως, σε κάθε επανάληψη, προσαρμόζεται ο εκτιμητής  $h_m$  για να προβλέψει τις αρνητικές κλίσεις των δειγμάτων. Οι διαβαθμίσεις ενημερώνονται σε κάθε επανάληψη. Αυτό μπορεί να θεωρηθεί ως κάποιο είδος κλίσης καθόδου σε ένα λειτουργικό χώρο.

**Σημείωση:** Για κάποιες απώλειες, π.χ. η ελάχιστη απόλυτη απόκλιση (LAD) όπου οι διαβαθμίσεις είναι  $\pm 1$ , οι τιμές που προβλέπονται από ένα προσαρμοσμένο  $h_m$  δεν είναι αρκετά ακριβείς: το δέντρο μπορεί να εξάγει μόνο ακέραιες τιμές. Ως αποτέλεσμα, οι τιμές των φύλλων του δέντρου  $h_m$  τροποποιούνται μόλις τοποθετηθεί το δέντρο, έτσι ώστε οι τιμές των φύλλων να ελαχιστοποιούν την απώλεια  $L_m$ . Η ενημέρωση εξαρτάται από την απώλεια: για την απώλεια LAD, η τιμή ενός φύλλου ενημερώνεται στη διάμεσο των δειγμάτων σε αυτό το φύλλο.

## Classification

Η ενίσχυση κλίσης για ταξινόμηση είναι πολύ παρόμοια με την περίπτωση παλινδρόμησης. Ωστόσο, το άθροισμα των δέντρων  $F_M(x_i) = \sum_m h_m(x_i)$  δεν είναι ομοιογενές σε μια πρόβλεψη: δεν μπορεί να είναι κλάση, αφού τα δέντρα προβλέπουν συνεχείς τιμές.

Η αντιστοίχιση από την τιμή  $F_M(x_i)$  σε μια κλάση ή μια πιθανότητα εξαρτάται από την απώλεια. Για το log-loss, η πιθανότητα ότι το  $x$  ανήκει στη θετική κλάση μοντελοποιείται ως  $p(y_i = 1|x_i) = \sigma(F_M(x_i))$  όπου  $\sigma$  είναι η σιγμοειδής ή η συνάρτηση εξόδου.

Για την ταξινόμηση πολλαπλών κλάσεων, τα δέντρα K (για τις κατηγορίες K) χτίζονται σε κάθε μία από τις M επαναλήψεις. Η πιθανότητα ότι το  $x_i$  ανήκει στην κλάση k μοντελοποιείται ως softmax των τιμών  $F_{M,k}(x_i)$ .

Αξιοσημείωτο είναι το ότι ακόμη και για μια εργασία ταξινόμησης, ο υποεκτιμητής  $h_m$  εξακολουθεί να είναι ένας παλινδρομητής, όχι ένας ταξινομητής. Αυτό συμβαίνει επειδή οι υποεκτιμητές εκπαιδεύονται να προβλέπουν (αρνητικές) κλίσεις, οι οποίες είναι πάντα συνεχείς ποσότητες.

## Συναρτήσεις Απώλειας

- Παλινδρόμηση

- **Squared error** : Η συνηθισμένη επιλογή για παλινδρόμηση λόγω των ανώτερων υπολογιστικών ιδιοτήτων του. Το αρχικό μοντέλο δίνεται με τον μέσο όρο των τιμών-στόχων.
- **Least absolute deviation** : Μια ισχυρή συνάρτηση απώλειας για παλινδρόμηση. Το αρχικό μοντέλο δίνεται από τη διάμεσο των τιμών-στόχων.

- **Huber** : Μια άλλη ισχυρή συνάρτηση απώλειας που συνδυάζει ελάχιστα τετράγωνα και ελάχιστη απόλυτη απόκλιση. Το α χρησιμοποιείται για να ελεγχθεί η ευαισθησία σε σχέση με τις ακραίες τιμές.
- **Quantile** : Μια συνάρτηση απώλειας για παλινδρόμηση ποσοστοιχιών. Το  $0 < \alpha < 1$  χρησιμοποιείται για να καθοριστεί το ποσοστό. Αυτή η συνάρτηση απώλειας μπορεί να χρησιμοποιηθεί για τη δημιουργία διαστημάτων πρόβλεψης.

- **Ταξινόμηση**

- **Binary log-loss** : Η διωνυμική συνάρτηση απώλειας αρνητικής λογαριθμικής πιθανότητας για δυαδική ταξινόμηση. Παρέχει εκτιμήσεις πιθανοτήτων. Το αρχικό μοντέλο δίνεται από το log odds-ratio.
- **Multi-class log-loss** : Η πολυωνυμική συνάρτηση αρνητικής απώλειας log-likelihood για ταξινόμηση πολλαπλών κλάσεων με αμοιβαία αποκλειστικές κλάσεις. Παρέχει εκτιμήσεις πιθανοτήτων. Το αρχικό μοντέλο δίνεται από την προηγούμενη πιθανότητα κάθε κλάσης. Σε κάθε επανάληψη, πρέπει να δημιουργηθούν τόσα δέντρα παλινδρόμησης όσα είναι και ο αριθμός των κλάσεων, που καθιστά το GBRT μάλλον αναποτελεσματικό για σύνολα δεδομένων με μεγάλο αριθμό κλάσεων.
- **Exponential loss** : Η ίδια συνάρτηση απώλειας με τον AdaBoostClassifier. Λιγότερο ανθεκτικά σε δείγματα με εσφαλμένη επισήμανση από το 'log-loss'. Μπορεί να χρησιμοποιηθεί μόνο για δυαδική ταξινόμηση.

### Συρρίκνωση μέσω του ρυθμού μάθησης

Ο Friedman [58] πρότεινε μια απλή στρατηγική κανονικοποίησης που κλιμακώνει τη συμβολή κάθε αδύναμου μαθητή με έναν σταθερό παράγοντα  $\nu$  :

$$F_m(x) = F_{m-1}(x) + \nu h_m(x)$$

Η παράμετρος  $\nu$  ονομάζεται επίσης ρυθμός εκμάθησης επειδή κλιμακώνει το μήκος του βήματος της διαδικασίας κλίσης καθόδου.

Η παράμετρος  $\nu$  αλληλεπιδρά έντονα με τον αριθμό των αδύναμων μαθητών που θα προσαρμοστούν στα δεδομένα. Οι μικρότερες τιμές του ρυθμού μάθησης απαιτούν μεγαλύτερο αριθμό αδύναμων μαθητών για να διατηρήσουν ένα σταθερό σφάλμα εκπαίδευσης. Εμπειρικά στοιχεία υποδεικνύουν ότι οι μικρές τιμές του ρυθμού μάθησης ευνοούν για το καλύτερο σφάλμα δοκιμής. Συνίσταται [31] να οριστεί ο ρυθμός εκμάθησης σε μια μικρή σταθερά (π.χ.  $\nu \leq 0.1$ ) και ο αριθμός των εκτιμητών να επιλεγεί με πρόωρη διακοπή.

### eXtreme Gradient Boosting & Light Gradient Boosting Machine

Δύο από τους πιο δημοφιλείς αλγόριθμους που βασίζονται σε αυτή τη λογική και θα χρησιμοποιηθούν παρακάτω, είναι ο XGBoost και ο LightGBM.

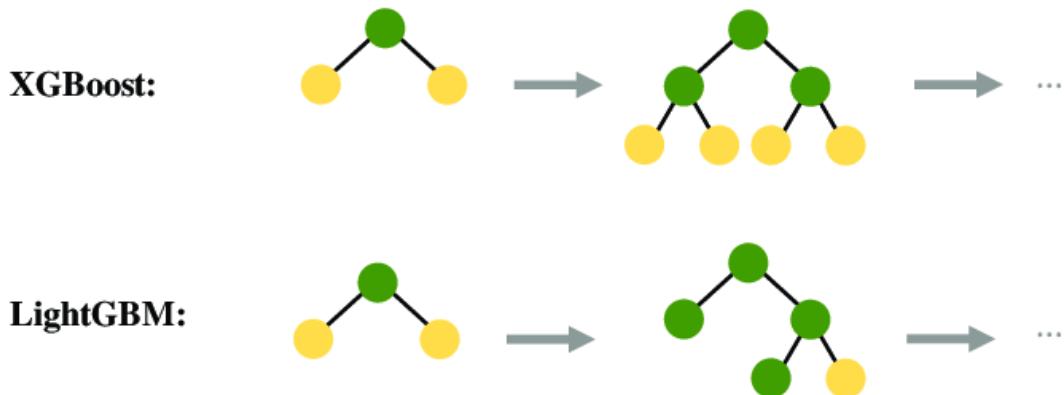
- **eXtreme Gradient Boosting**

Το XGBoost είναι ένας αλγόριθμος μηχανικής μάθησης που εστιάζει στην ταχύτητα υπολογισμού και στην απόδοση του μοντέλου. Εισήχθη από τον Tianqi Chen [59] και επί του παρόντος αποτελεί μέρος μιας ευρύτερης εργαλειοθήκης από την DMLC (Distributed Machine Learning Community). Ο αλγόριθμος μπορεί να χρησιμοποιηθεί τόσο για εργασίες παλινδρόμησης όσο και για εργασίες ταξινόμησης και έχει σχεδιαστεί για να λειτουργεί με μεγάλα και περίπλοκα σύνολα δεδομένων.

- **Light Gradient Boosting Machine**

Παρόμοια με το XGBoost, το LightGBM [60] (από τη Microsoft) είναι ένα κατανεμημένο πλαίσιο υψηλής απόδοσης που χρησιμοποιεί δέντρα αποφάσεων για εργασίες κατάταξης, ταξινόμησης και παλινδρόμησης.

Σε αντίθεση με τη level-wise (οριζόντια) ανάπτυξη στο XGBoost, το LightGBM πραγματοποιεί φυλλομετρική (κάθετη) ανάπτυξη (σχήμα 3.24) που έχει ως αποτέλεσμα μεγαλύτερη μείωση των απωλειών και με τη σειρά της μεγαλύτερη ακρίβεια ενώ είναι πιο γρήγορο. Άλλα αυτό μπορεί επίσης να οδηγήσει σε υπερπροσαρμογή στα δεδομένα εκπαίδευσης που θα μπορούσαν να χειριστούν χρησιμοποιώντας την παράμετρο μέγιστου βάθους που καθορίζει πού θα συμβεί ο διαχωρισμός. Ως εκ τούτου, το XGBoost είναι σε θέση να δημιουργήσει πιο στιβαρά μοντέλα από το LightGBM.



Σχήμα 3.24: Βασική διαφορά XGBoost με LightGBM [61]

## 3.3 ΕΙΣΑΓΩΓΗ ΣΤΑ ΝΕΥΡΩΝΙΚΑ ΔΙΚΤΥΑ

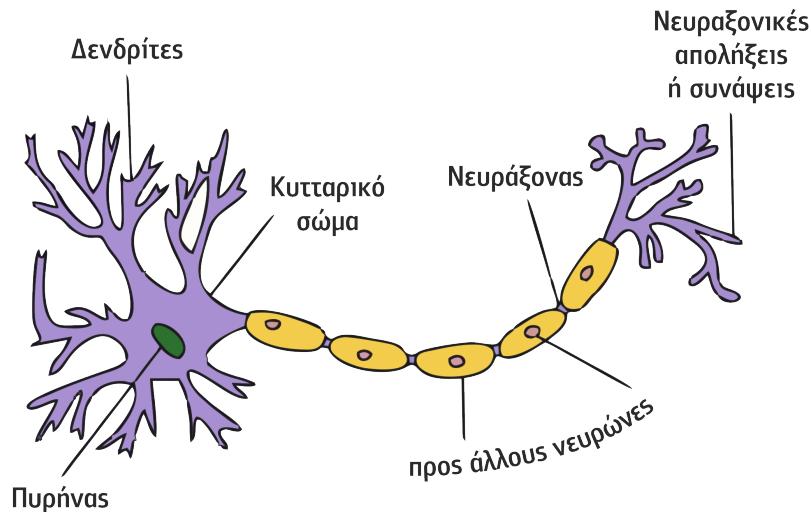
### 3.3.1 Ο (τεχνητός) νευρώνας

Το ανθρώπινο νευρικό σύστημα αποτελείται από τον εγκέφαλο, ο οποίος λαμβάνει και επεξεργάζεται πληροφορίες, και από υποδοχείς και φορείς που μετατρέπουν τα ερεθίσματα σε ηλεκτρικά ερεθίσματα και αποκρίσεις, αντίστοιχα. Περίπου 86 δισεκατομμύρια νευρώνες βρίσκονται στο ανθρώπινο νευρικό σύστημα και συνδέονται μεταξύ τους με περίπου  $10^{14} - 10^{15}$  συνάψεις. Ο εγκέφαλος περιέχει περίπου 10 δισεκατομμύρια νευρώνες στον φλοιό και 60 τρισεκατομμύρια συνάψεις ή συνδέσεις μεταξύ τους, γεγονός που του επιτρέπει να λειτουργεί αποτελεσματικά με

### ΚΕΦΑΛΑΙΟ 3. ΜΗΧΑΝΙΚΗ ΚΑΙ ΒΑΘΙΑ ΜΑΘΗΣΗ

ενεργειακή απόδοση περίπου 10-16 Joule ανά λειτουργία ανά δευτερόλεπτο.

Κάθε νευρώνας δέχεται σήματα εισόδου από τους δενδρίτες του και παράγει σήματα εξόδου κατά μήκος του (μοναδικού) άξονά του. Ο άξονας τελικά διακλαδίζεται και συνδέεται μέσω συνάψεων με δενδρίτες άλλων νευρώνων σχήμα 3.25.



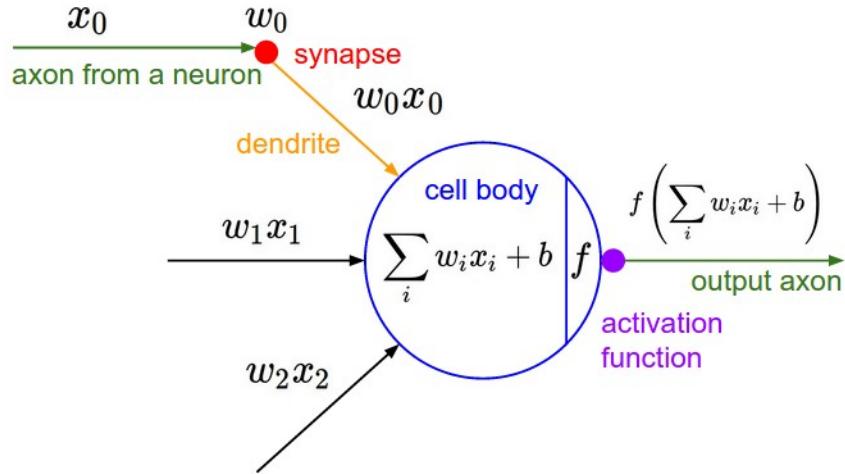
Σχήμα 3.25: Βιολογικός νευρώνας [62]

Το αντίστοιχο μαθηματικό μοντέλο του νευρώνα, φαίνεται στο σχήμα 3.26. Στο υπολογιστικό μοντέλο ενός νευρώνα [63], τα σήματα που ταξιδεύουν κατά μήκος των αξόνων ( $x_i$ ) αλληλεπιδρούν πολλαπλασιαστικά ( $w_i * x_i$ ) με τους δενδρίτες του άλλου νευρώνα με βάση τη συναπτική ισχύ στη συγκεκριμένη σύναψη ( $w_i$ ). Η ιδέα είναι ότι οι συναπτικές εντάσεις (τα βάρη  $w_i$ ) μπορούν να μάθουν και να ελέγχουν την ένταση της επιρροής (και την κατεύθυνσή της: διεγερτική (θετικό βάρος) ή ανασταλτική (αρνητικό βάρος)) ενός νευρώνα σε έναν άλλο.

Οι συνολικές συνεισφορές συνδυάζονται μαζί, μαζί με μια προκατάληψη  $b$  (bias), σχηματίζοντας το άθροισμα  $b + \sum_i w_i x_i$ . Αυτή η τιμή περνάει στη συνέχεια σε μια μη γραμμική συνάρτηση ενεργοποίησης  $f$  για να παράγει την έξοδο  $f(b + \sum_i w_i x_i)$ . Αυτή είναι η ενεργοποίηση του νευρώνα και η τιμή αυτή είναι που περνάει σε άλλους νευρώνες που συνδέονται με τον άξονά του.

Ο αλγόριθμος perceptron είναι ένας αλγόριθμος γραμμικού ταξινομητή, που σημαίνει ότι διαχωρίζει το χώρο εισόδου σε δύο περιοχές με ένα γραμμικό όριο απόφασης. Εκπαιδεύεται χρησιμοποιώντας έναν αλγόριθμο μάθησης με επίβλεψη, όπως η κάθιδος κλίσης, ο οποίος προσαρμόζει τα βάρη και την προκατάληψη με βάση το σφάλμα μεταξύ της προβλεπόμενης εξόδου και της πραγματικής εξόδου.

Ο αλγόριθμος αρχικοποιεί τα βάρη  $w$  και την προκατάληψη  $b$  στο μηδέν, και στη συνέχεια, για κάθε παράδειγμα εκπαίδευσης  $(x_i, y_i)$ , ελέγχει αν το παράδειγμα έχει ταξινομηθεί εσφαλμένα από το τρέχον όριο απόφασης που αντιπροσωπεύεται από τα βάρη και την προκατάληψη. Εάν ναι, τα βάρη και η προκατάληψη ενημερώνονται προς την κατεύθυνση της σωστής ταξινόμησης. Ο αλγόριθμος συνεχίζει μέχρι να ταξινομηθούν σωστά όλα τα παραδείγματα ή να επιτευχθεί ένα κριτήριο διακοπής. Η τελική έξοδος είναι τα μαθημένα βάρη και η προκατάληψη. Αξίζει να σημειωθεί



Σχήμα 3.26: Μαθηματικό μοντέλο του νευρώνα [63]

---

**Άλγοριθμος 3.7 Perceptron Algorithm**


---

**Είσοδος:** Δεδομένα εκπαίδευσης  $\mathcal{D} = \{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ , όπου  $x_i \in \mathbb{R}^m$  και  $y_i \in \{-1, 1\}$

**Αρχικοποίηση:**  $w \leftarrow \mathbf{0}$ ,  $b \leftarrow 0$

**Επανάλαβε:**

```

for  $i = 1$  ως  $n$  do
    if  $y_i(\mathbf{w}^T \mathbf{x}_i + b) \leq 0$  then
         $w \leftarrow w + y_i \mathbf{x}_i$ 
         $b \leftarrow b + y_i$ 
    end if
end for

```

**Μέχρι:** Σύγκλιση

**Έξοδος:**  $w, b$

---

ότι ο αλγόριθμος 3.7 είναι η βασική εκδοχή του αλγορίθμου perceptron και δεν είναι εγγυημένη η σύγκλιση, γι' αυτό και η συνθήκη σύγκλισης δεν προσδιορίζεται στον φευδοκώδικα. Για την εγγύηση της σύγκλισης χρησιμοποιείται μια επέκταση του αλγορίθμου perceptron που ονομάζεται "Pocket Algorithm".

### 3.3.2 Συναρτήσεις Ενεργοποίησης

Η συνάρτηση ενεργοποίησης είναι ένα σημαντικό μέρος ενός νευρωνικού δικτύου και παίζει καθοριστικό ρόλο στον καθορισμό της ικανότητας του δικτύου να μαθαίνει και να γενικεύει σε νέα δεδομένα. Η επιλογή της σωστής συνάρτησης ενεργοποίησης μπορεί να είναι καθοριστική για την απόδοση ενός πολυεπίπεδου perceptron (MLP) σε μια εργασία ταξινόμησης.

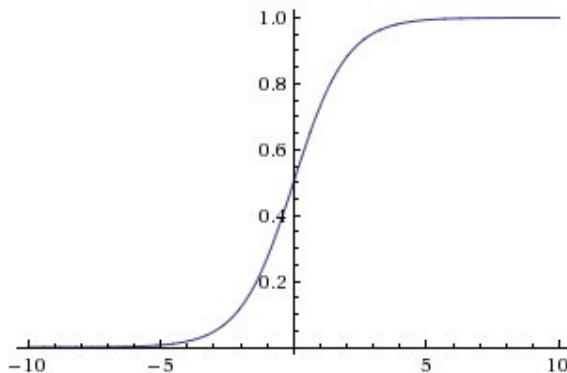
Παρακάτω παρουσιάζονται οι πιο συνηθισμένες συναρτήσεις ενεργοποίησης[63]:

- **Σιγμοειδής - Sigmoid**

Η σιγμοειδής συνάρτηση φαίνεται στο σχήμα 3.27 και έχει τη μαθηματική μορφή

$$\sigma(x) = 1/(1 + e^{-x}).$$

Παίρνει έναν πραγματικό αριθμό και τον "συμπιέζει" σε ένα εύρος μεταξύ 0 και 1. Συγκεκριμένα, οι μεγάλοι αρνητικοί αριθμοί γίνονται 0 και οι μεγάλοι θετικοί αριθμοί γίνονται 1. Η σιγμοειδής συνάρτηση έχει χρησιμοποιηθεί συχνά ιστορικά, καθώς έχει μια ωραία ερμηνεία ως ο ρυθμός πυροδότησης ενός νευρώνα: από καθόλου πυροδότηση (0) έως πλήρως κορεσμένη πυροδότηση σε μια υποτιθέμενη μέγιστη συχνότητα (1).



Σχήμα 3.27: Συνάρτηση Σιγμοειδούς συνάρτησης [63]

Στην πράξη, η σιγμοειδής μη γραμμικότητα έχει πρόσφατα χάσει την εύνοια της και χρησιμοποιείται σπάνια. Έχει δύο σημαντικά μειονεκτήματα:

- Φτάνει σε κορεσμό και "σκοτώνει" τις κλίσεις. Μια πολύ ανεπιθύμητη ιδιότητα του σιγμοειδούς νευρώνα είναι ότι όταν η ενεργοποίηση του νευρώνα κορεστεί σε οποιαδήποτε "ουρά" του 0 ή του 1, η κλίση σε αυτές τις περιοχές είναι σχεδόν μηδενική. Κατά την οπισθοδιάδοση, αυτή η (τοπική) κλίση θα πολλαπλασιαστεί με την κλίση της εξόδου αυτής της πύλης για ολόκληρο τον στόχο. Επομένως, εάν η τοπική κλίση είναι πολύ μικρή, θα "σκοτώσει" ουσιαστικά την κλίση και σχεδόν κανένα σήμα δεν θα ρέει μέσω του νευρώνα στα

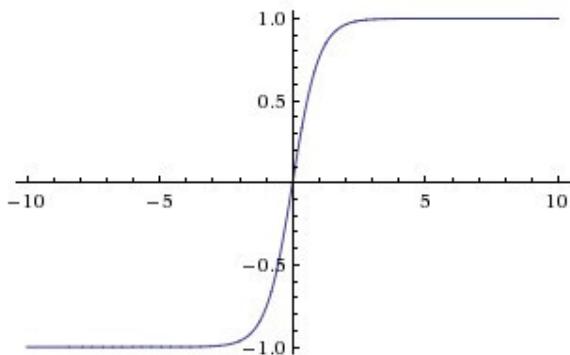
βάρη του και αναδρομικά στα δεδομένα του. Επιπλέον, πρέπει να δίνεται ιδιαίτερη προσοχή κατά την αρχικοποίηση των βαρών των σιγμοειδών νευρώνων για να αποφευχθεί ο κορεσμός. Για παράδειγμα, αν τα αρχικά βάρη είναι πολύ μεγάλα, τότε οι περισσότεροι νευρώνες θα κορεστούν και το δίκτυο δύσκολα θα μάθει.

- Οι σιγμοειδείς έξοδοι δεν είναι μηδενικοκεντρικές. Αυτό είναι ανεπιθύμητο, δεδομένου ότι οι νευρώνες σε μεταγενέστερα επίπεδα επεξεργασίας σε ένα Νευρωνικό Δίκτυο θα λαμβάνουν δεδομένα που δεν είναι μηδενικά κεντροειδέα. Αυτό έχει επιπτώσεις στη δυναμική κατά την κάθιση αλίσης, διότι αν τα δεδομένα που εισέρχονται σε έναν νευρώνα είναι πάντα θετικά (π.χ.  $x > 0$  στοιχειωδώς στο  $f = w^T x + b$ ), τότε οι αλίσεις των βαρών  $w$  κατά την οπισθοδιάδοση θα γίνουν είτε όλες θετικές, είτε όλες αρνητικές (ανάλογα με την αλίση ολόκληρης της έκφρασης  $f$ ). Αυτό θα μπορούσε να εισάγει ανεπιθύμητα μπρος-πίσω στις ενημερώσεις της αλίσης για τα βάρη. Ωστόσο, παρατηρείται ότι μόλις αυτές οι αλίσεις αθροιστούν σε μια δέσμη δεδομένων, η τελική ενημέρωση για τα βάρη μπορεί να έχει μεταβλητά πρόσημα, μετριάζοντας κάπως αυτό το ζήτημα. Επομένως, πρόκειται για μια ενόχληση, η οποία έχει λιγότερο σοβαρές συνέπειες σε σύγκριση με το παραπάνω πρόβλημα κορεσμένης ενεργοποίησης.

- **Υπερβολική Εφαπτομένη - Tanh**

Η συνάρτηση  $\tanh$  φαίνεται στο σχήμα 3.28. Συρρικνώνει έναν πραγματικό αριθμό στην περιοχή  $[-1, 1]$ . Όπως και ο σιγμοειδής νευρώνας, οι ενεργοποιήσεις της προκαλούν κορεσμό, αλλά σε αντίθεση με τον σιγμοειδή νευρώνα η έξοδός της είναι μηδενική. Επομένως, στην πράξη η μη γραμμικότητα  $\tanh$  προτιμάται πάντα από τη σιγμοειδή μη γραμμικότητα. Να σημειωθεί επίσης ότι ο νευρώνας  $\tanh$  είναι απλώς ένας κλιμακωτός σιγμοειδής νευρώνας, ειδικότερα ισχύει το εξής:

$$\tanh x = 2\sigma(2x) - 1$$



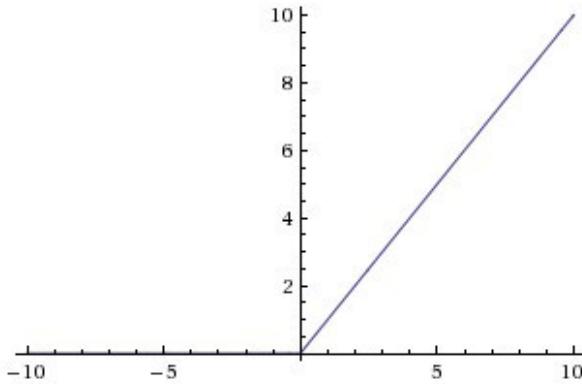
Σχήμα 3.28: Συνάρτηση Υπερβολικής Εφαπτομένης [63]

- **ReLU**

Η ανορθωμένη γραμμική μονάδα έχει γίνει πολύ δημοφιλής τα τελευταία χρόνια. Η ενεργοποίηση απλά κατωφλιώνεται στο μηδέν (βλ. σχήμα 3.29). Υπολογίζει τη

συνάρτηση

$$f(x) = \max(0, x) \equiv f(x) = \begin{cases} x, & \text{Αν } x > 0 \\ 0, & \text{Διαφορετικά} \end{cases}$$



Σχήμα 3.29: Συνάρτηση Rectified Linear Unit - ReLU [63]

Υπάρχουν διάφορα πλεονεκτήματα και μειονεκτήματα στη χρήση των ReLUs:

- (+) Διαπιστώθηκε ότι επιταχύνει σημαντικά (π.χ. κατά 6 φορές στην εργασία για το ImageNet των Krizhevsky, Sutskever, Hinton) τη σύγκλιση της στοχαστικής καθόδου κλίσης σε σύγκριση με τις συναρτήσεις sigmoid/tanh. Υποστηρίζεται ότι αυτό οφείλεται στη γραμμική, μη κορεσμένη μορφή της.
- (+) Σε σύγκριση με τους νευρώνες tanh/sigmoid που περιλαμβάνουν δαπανηρές πράξεις (εκθετικά κλπ.), ο ReLU μπορεί να υλοποιηθεί με απλή κατωφλίωση ενός πίνακα ενεργοποιήσεων στο μηδέν.
- (-) Δυστυχώς, οι μονάδες ReLU μπορεί να είναι εύθραυστες κατά τη διάρκεια της εκπαίδευσης και να "πεθάνουν". Για παράδειγμα, μια μεγάλη κλίση που διαρρέει έναν νευρώνα ReLU μπορεί να προκαλέσει την ενημέρωση των βαρών με τέτοιο τρόπο ώστε ο νευρώνας να μην ενεργοποιηθεί ποτέ ξανά σε κανένα σημείο δεδομένων. Αν συμβεί αυτό, τότε η κλίση που ρέει μέσω της μονάδας θα είναι για πάντα μηδέν από εκείνο το σημείο και μετά. Δηλαδή, οι μονάδες ReLU μπορούν να πεθάνουν μη αναστρέψιμα κατά τη διάρκεια της εκπαίδευσης, αφού μπορούν να χτυπηθούν από την πολλαπλότητα δεδομένων. Για παράδειγμα, μπορεί να διαπιστωθεί ότι έως και το 40% του δικτύου μπορεί να είναι "νεκρό" (δηλαδή νευρώνες που δεν ενεργοποιούνται ποτέ σε ολόκληρο το σύνολο δεδομένων εκπαίδευσης), εάν ο ρυθμός μάθησης έχει ρυθμιστεί πολύ υψηλά. Με την κατάλληλη ρύθμιση του ρυθμού μάθησης αυτό είναι λιγότερο συχνό ζήτημα.

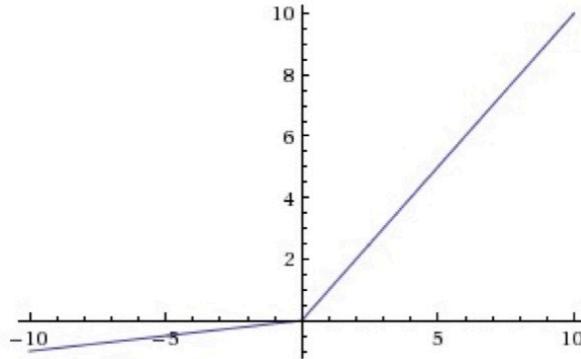
#### • Leaky ReLU

Η Leaky ReLU είναι μια προσπάθεια να διορθωθεί το προαναφερθέν πρόβλημα που υπάρχει με την συνάρτηση ReLU. Αντί η συνάρτηση να είναι μηδενική όταν

$x < 0$ , μια διαφορέουσα ReLU θα έχει μια μικρή θετική κλίση (0.01, περίπου). Δηλαδή, η συνάρτηση υπολογίζει την

$$f(x) = 1(x < 0)(\alpha x) + 1(x \geq 0)(x) = \begin{cases} x, & \text{Αν } x > 0 \\ \alpha x, & \text{Διαφορετικά} \end{cases}$$

όπου  $\alpha$  είναι μια μικρή σταθερά.



Σχήμα 3.30: Συνάρτηση Leaky ReLU [63]

Μερικοί άνθρωποι αναφέρουν επιτυχία με αυτή τη μορφή συνάρτησης ενεργοποίησης, αλλά τα αποτελέσματα δεν είναι πάντα συνεπή. Η κλίση στην αρνητική περιοχή μπορεί επίσης να γίνει παράμετρος κάθε νευρώνα, όπως φαίνεται στους νευρώνες PReLU, που παρουσιάζονται στο Delving Deep into Rectifiers, του Kaiming He [64]. Ωστόσο, η συνέπεια του οφέλους σε όλες τις εργασίες είναι προς το παρόν ασαφής.

- **Maxout**

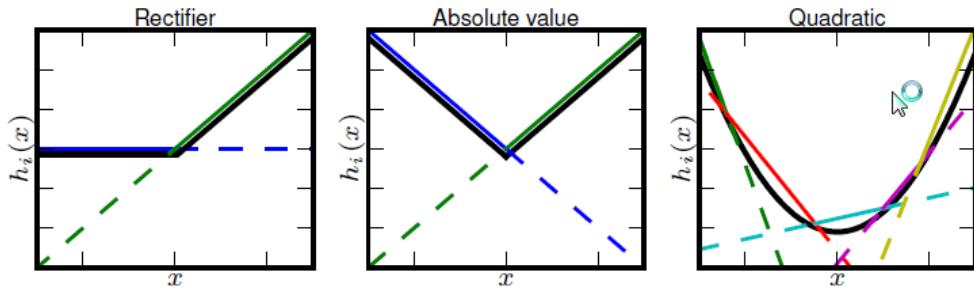
Έχουν προταθεί άλλοι τύποι μονάδων που δεν έχουν τη λειτουργική μορφή  $f(w^T x + b)$ , όπου εφαρμόζεται μια μη γραμμικότητα στο γινόμενο τελείας μεταξύ των βαρών και των δεδομένων. Μια σχετικά δημοφιλής επιλογή είναι ο νευρώνας Maxout (που εισήχθη πρόσφατα [65]) ο οποίος γενικεύει τον ReLU και τη διαφορέουσα εκδοχή του. Ο νευρώνας Maxout υπολογίζει τη συνάρτηση

$$f(x) = \max(w_1^T x + b_1, w_2^T x + b_2).$$

Να σημειωθεί ότι τόσο η ReLU όσο και η Leaky ReLU αποτελούν ειδική περίπτωση αυτής της μορφής (για παράδειγμα, για  $w1, b1 = 0$  παίρνει την μορφή της ReLU). Συνεπώς, ο νευρώνας Maxout απολαμβάνει όλα τα πλεονεκτήματα μιας μονάδας ReLU (γραμμικό καθεστώς λειτουργίας, μη κορεσμός) και δεν έχει τα μειονεκτήματά της. Ωστόσο, σε αντίθεση με τους νευρώνες ReLU διπλασιάζει τον αριθμό των παραμέτρων για κάθε νευρώνα, οδηγώντας σε υψηλό συνολικό αριθμό παραμέτρων.

Πρακτικά, όσα μοντέλα NN χρησιμοποιούν τη συνάρτηση Maxout μπορούν να μάθουν τη συνάρτηση ενεργοποίησης όλων των νευρώνων ενός ή περισσότερων επιπέδων. Στο σχήμα 3.31 φαίνονται μερικά παραδείγματα της συνάρτησης Maxout, μετά από την εκπαίδευση δικτύων με αυτήν.

Ακολουθούν ορισμένες σκέψεις για την επιλογή μιας συνάρτησης ενεργοποίησης για μια εργασία ταξινόμησης σε ένα MLP:



Σχήμα 3.31: Συνάρτηση Maxout [65]

- Ταξινόμηση έναντι παλινδρόμησης: Εάν η εργασία είναι εργασία ταξινόμησης, τότε η συνάρτηση ενεργοποίησης θα πρέπει να επιλεγεί ανάλογα. Συναρτήσεις ενεργοποίησης όπως η sigmoid και η softmax χρησιμοποιούνται συνήθως για εργασίες ταξινόμησης, καθώς αντιστοιχίζουν την έξοδο του δικτύου σε μια κατανομή πιθανότητας πάνω στις κλάσεις. Εάν η εργασία είναι εργασία παλινδρόμησης, όπου η έξοδος είναι μια συνεχής μεταβλητή, τότε μια συνάρτηση ενεργοποίησης όπως η ReLU ή η tanh μπορεί να είναι πιο κατάλληλη.
- Εύρος εξόδου: Θα πρέπει να ληφθεί υπόψη το εύρος εξόδου της συνάρτησης ενεργοποίησης. Για παράδειγμα, οι συναρτήσεις ενεργοποίησης sigmoid και softmax δίνουν τιμές εξόδου μεταξύ 0 και 1, πράγμα χρήσιμο για εργασίες ταξινόμησης όπου η έξοδος είναι μια πιθανότητα. Οι συναρτήσεις ενεργοποίησης ReLU και tanh δίνουν τιμές μεταξύ -1 και 1, οι οποίες μπορεί να είναι πιο κατάλληλες για εργασίες όπου η έξοδος είναι μια συνεχής μεταβλητή.
- Ύπολογιστική πολυπλοκότητα: Θα πρέπει επίσης να ληφθεί υπόψη η υπολογιστική πολυπλοκότητα της συνάρτησης ενεργοποίησης. Ορισμένες συναρτήσεις ενεργοποίησης, όπως η sigmoid και η tanh, είναι υπολογιστικά πιο δαπανηρές για να αξιολογηθούν από άλλες, όπως η ReLU.
- Εξαφανιζόμενες κλίσεις: Οι συναρτήσεις ενεργοποίησης που έχουν μη μηδενική παράγωγο για ένα μεγάλο εύρος τιμών εισόδου προτιμώνται γενικά, καθώς αποφεύγουν το πρόβλημα των εξαφανιζόμενων κλίσεων. Οι συναρτήσεις ενεργοποίησης όπως η ReLU και η leaky ReLU έχουν παράγωγο 1 για ένα μεγάλο εύρος τιμών εισόδου και επομένως είναι λιγότερο επιρρεπείς στο πρόβλημα των εξαφανιζόμενων κλίσεων.
- Αραιότητα: Οι συναρτήσεις ενεργοποίησης που ενθαρρύνουν τη σπανιότητα (δηλ. τιμές κοντά στο 0) στις ενεργοποιήσεις των νευρώνων μπορεί να είναι χρήσιμες για εργασίες όπου η σπανιότητα είναι επιθυμητή, όπως η επιλογή χαρακτηριστικών. Συναρτήσεις ενεργοποίησης όπως η ReLU και η leaky ReLU έχουν αυτή την ιδιότητα.

Δεν υπάρχει μία "καλύτερη" συνάρτηση ενεργοποίησης για όλες τις εργασίες ταξινόμησης και η επιλογή εξαρτάται από τα συγκεκριμένα χαρακτηριστικά των δεδομένων και της εκάστοτε εργασίας. Ο πειραματισμός με διαφορετικές συναρτήσεις ενεργοποίησης και η επιλογή αυτής που λειτουργεί καλύτερα για μια συγκεκριμένη εργασία είναι συχνά μια καλή προσέγγιση.

### 3.3.3 Συναρτήσεις Σφάλματος/Κόστους

Μια συνάρτηση κόστους είναι ένα μέτρο του πόσο καλά ένα νευρωνικό δίκτυο είναι σε θέση να προβλέψει την επιθυμητή έξοδο για μια δεδομένη είσοδο. Με άλλα λόγια, μετράει το σφάλμα μεταξύ της προβλεπόμενης εξόδου και της πραγματικής εξόδου.

Υπάρχουν διάφοροι τύποι συναρτήσεων κόστους που μπορούν να χρησιμοποιηθούν στα νευρωνικά δίκτυα, όπως το μέσο τετραγωνικό σφάλμα, το μέσο απόλυτο σφάλμα, η δυαδική διασταυρούμενη εντροπία, η κατηγορική διασταυρούμενη εντροπία, η απώλεια άρθρωσης και η τετραγωνική απώλεια άρθρωσης. Η επιλογή της συνάρτησης κόστους εξαρτάται από τη συγκεκριμένη εργασία και τα χαρακτηριστικά των δεδομένων.

Η συνάρτηση κόστους αποτελεί βασικό παράγοντα για τον καθορισμό της απόδοσης ενός νευρωνικού δικτύου. Μια καλά σχεδιασμένη συνάρτηση κόστους μπορεί να βοηθήσει το δίκτυο να μάθει αποτελεσματικά και να κάνει ακριβείς προβλέψεις, ενώ μια κακώς σχεδιασμένη συνάρτηση κόστους μπορεί να οδηγήσει σε κακή απόδοση.

Συνολικά, η συνάρτηση κόστους είναι ένα κρίσιμο στοιχείο των νευρωνικών δικτύων και η κατανόηση του τρόπου λειτουργίας της είναι απαραίτητη για το σχεδιασμό και την εκπαίδευση αποτελεσματικών δικτύων.

Ακολουθεί ένας κατάλογος συναρτήσεων κόστους που χρησιμοποιούνται συνήθως για εργασίες ταξινόμησης στα νευρωνικά δίκτυα, με τους μαθηματικούς ορισμούς τους και μια σύντομη επεξήγηση της καθεμιάς [66].

- **Binary Cross-Entropy:** Αυτή η συνάρτηση κόστους χρησιμοποιείται όταν η έξοδος του δικτύου είναι μια δυαδική κλάση (δηλαδή 0 ή 1). Η δυαδική συνάρτηση κόστους cross-entropy μετρά τη μέση διαφορά μεταξύ της προβλεπόμενης πιθανότητας της θετικής κλάσης και της αληθινής ετικέτας για όλα τα παραδείγματα. Ελαχιστοποιείται όταν η μέση προβλεπόμενη πιθανότητα είναι κοντά στη μέση πραγματική ετικέτα. Η δυαδική συνάρτηση κόστους cross-entropy ορίζεται ως εξής:

$$BCE = -\frac{1}{N} \sum (y_{true} * \log(y_{pred}) + (1 - y_{true}) * \log(1 - y_{pred}))$$

όπου  $N$  είναι ο αριθμός των παραδειγμάτων,  $y_{true}$  είναι η πραγματική ετικέτα (είτε 0 είτε 1) και  $y_{pred}$  είναι η προβλεπόμενη πιθανότητα της θετικής κλάσης.

- **Categorical Cross-Entropy:** Αυτή η συνάρτηση κόστους χρησιμοποιείται όταν η έξοδος του δικτύου είναι μια κατηγορική κλάση (π.χ. μία από πολλές πιθανές κλάσεις). Μετρά τη διαφορά μεταξύ της προβλεπόμενης πιθανότητας κάθε κλάσης και της πραγματικής ετικέτας. Η συνάρτηση κόστους κατηγορικής διασταυρούμενης εντροπίας ορίζεται ως εξής:

$$CCE = - \sum (y_{true_i} * \log(y_{pred_i}))$$

όπου  $i$  είναι ο δείκτης της κατηγορίας.

Η κατηγορική συνάρτηση κόστους διασταυρούμενης εντροπίας είναι μια γενίκευση της δυαδικής συνάρτησης κόστους διασταυρούμενης εντροπίας σε πολλαπλές κλάσεις. Είναι ένα μέτρο της αβεβαιότητας που σχετίζεται με την πρόβλεψη ενός κατηγορικού αποτελέσματος. Η συνάρτηση κόστους ελαχιστοποιείται όταν οι προβλεπόμενες πιθανότητες κάθε κλάσης ( $y_{pred\_i}$ ) είναι κοντά στις πραγματικές ετικέτες ( $y_{true\_i}$ ).

- **Hinge Loss:** Αυτή η συνάρτηση κόστους χρησιμοποιείται για την εκπαίδευση μηχανών διανυσμάτων υποστήριξης (SVM). Μετρά τη διαφορά μεταξύ της προβλεπόμενης εξόδου και της πραγματικής εξόδου, με ποινή για τις προβλέψεις που δεν είναι σωστές. Η συνάρτηση κόστους hinge loss ορίζεται ως εξής:

$$HingeLoss = \max(0, 1 - y_{true} * y_{pred})$$

Η συνάρτηση κόστους απώλειας άρθρωσης χρησιμοποιείται για την εκπαίδευση γραμμικών ταξινομητών που είναι σε θέση να κάνουν σκληρά περιθώρια. Είναι ένα μέτρο του περιθωρίου μεταξύ της προβλεπόμενης εξόδου ( $y_{pred}$ ) και της πραγματικής εξόδου ( $y_{true}$ ). Η συνάρτηση κόστους ελαχιστοποιείται όταν η προβλεπόμενη έξοδος βρίσκεται στη σωστή πλευρά του περιθωρίου (δηλαδή, η προβλεπόμενη έξοδος είναι είτε 1 για τη θετική κλάση είτε -1 για την αρνητική κλάση).

- **Squared Hinge Loss:** Πρόκειται για μια παραλλαγή της συνάρτησης κόστους hinge loss που χρησιμοποιεί την τετραγωνική διαφορά μεταξύ της προβλεπόμενης εξόδου και της πραγματικής εξόδου ως ποινή για εσφαλμένες προβλέψεις. Η τετραγωνική συνάρτηση κόστους απώλειας άρθρωσης ορίζεται ως εξής:

$$SquaredHingeLoss = \frac{1}{2} \max(0, 1 - y_{true} * y_{pred})^2$$

Η τετραγωνική συνάρτηση κόστους απώλειας άρθρωσης είναι παρόμοια με τη συνάρτηση κόστους απώλειας άρθρωσης, αλλά τιμωρεί τις εσφαλμένες προβλέψεις πιο έντονα. Χρησιμοποιείται επίσης για την εκπαίδευση γραμμικών ταξινομητών που είναι σε θέση να κάνουν σκληρά περιθώρια. Η συνάρτηση κόστους ελαχιστοποιείται όταν η προβλεπόμενη έξοδος βρίσκεται στη σωστή πλευρά του περιθωρίου και το περιθώριο είναι όσο το δυνατόν μεγαλύτερο.

### 3.3.4 Αλγόριθμος Backpropagation

Ο αλγόριθμος *backpropagation* (αλγόριθμος 3.8 ή αλγόριθμος 3.9) πρωτοεμφανίστηκε το 1970 και υποτιμήθηκε μέχρι το 1986, όταν σε εργασία τους οι David Rumelhart, Geoffrey Hinton, και Ronald Williams έδειξαν την αποδοτικότητα του, κυρίως όσον αφορά στην ταχύτητα, στην εκπαίδευση των νευρωνικών δικτύων [67].

Η οπισθοδιάδοση είναι ένας αλγόριθμος που χρησιμοποιείται για την εκπαίδευση νευρωνικών δικτύων, συγκεκριμένα δικτύων πολλαπλών επιπέδων, για μάθηση με επίβλεψη. Είναι μια μέθοδος για τον αποτελεσματικό υπολογισμό των κλίσεων των βαρών και των προκαταλήψεων του δικτύου σε σχέση με τη συνάρτηση απώλειας, η οποία μπορεί στη συνέχεια να χρησιμοποιηθεί για την ενημέρωση των βαρών και των προκαταλήψεων με σκοπό την ελαχιστοποίηση της απώλειας.

Η βασική ιδέα πίσω από την οπισθοδιάδοση είναι η διάδοση του σφάλματος στην έξοδο του δικτύου προς τα πίσω μέσω των στρωμάτων του δικτύου, χρησιμοποιώντας τον κανόνα της αλυσιδωτής παραγώγισης για τον υπολογισμό της κλίσης της συνάρτησης απώλειας σε σχέση με κάθε βάρος και προκατάληψη. Οι κλίσεις χρησιμοποιούνται στη συνέχεια για την ενημέρωση των βαρών και των προκαταλήψεων σε μια διαδικασία που ονομάζεται κάθοδος κλίσης (gradient descent).

Η μαθηματική διατύπωση της οπισθοδιάδοσης περιλαμβάνει τα ακόλουθα βήματα [68]:

- Εμπρόσθια διάδοση:** Υπολογισμός της εξόδου του δικτύου με δεδομένο ένα παράδειγμα εισόδου και τα τρέχοντα βάροι και τις προκατάληψεις του δικτύου. Αυτό γίνεται περνώντας την είσοδο από κάθε επίπεδο του δικτύου, χρησιμοποιώντας την ακόλουθη εξίσωση για τον υπολογισμό της εξόδου κάθε νευρώνα:

$$a_j^{(l)} = \sigma \left( \sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \right)$$

όπου  $a_j^{(l)}$  είναι η έξοδος του  $j$ -οστού νευρώνα στο  $l$ -οστό στρώμα,  $n^{(l-1)}$  είναι ο αριθμός των νευρώνων στο  $(l-1)$ -οστό στρώμα,  $w_{ij}^{(l)}$  είναι το βάρος που συνδέει τον  $i$ -οστό νευρώνα στο  $(l-1)$ -οστό στρώμα με τον  $j$ -οστό νευρώνα στο  $l$ -οστό στρώμα,  $a_i^{(l-1)}$  είναι η έξοδος του  $i$ -οστού νευρώνα στο  $(l-1)$ -οστό στρώμα, και  $b_j^{(l)}$  είναι η προκατάληψη του  $j$ -οστού νευρώνα στο  $l$ -οστό στρώμα. Η συνάρτηση  $\sigma(\cdot)$  είναι η συνάρτηση ενεργοποίησης των νευρώνων.

- Υπολογισμός του σφάλματος εξόδου:** Υπολογισμός του σφάλματος της εξόδου του δικτύου σε σχέση με την πραγματική έξοδο χρησιμοποιώντας την ακόλουθη εξίσωση:

$$\delta_j^{(L)} = \frac{\partial C}{\partial a_j^{(L)}} \sigma' \left( \sum_{i=1}^{n^{(L-1)}} w_{ij}^{(L)} a_i^{(L-1)} + b_j^{(L)} \right)$$

όπου  $C$  είναι η συνάρτηση απωλειών,  $\delta_j^{(L)}$  είναι το σφάλμα του  $j$ -οστού νευρώνα στο στρώμα εξόδου και  $\sigma'(\cdot)$  είναι η παράγωγος της συνάρτησης ενεργοποίησης.

- Αναδρομική διάδοση του σφάλματος:** Το σφάλμα διαδίδεται προς τα πίσω μέσω των επιπέδων του δικτύου χρησιμοποιώντας την ακόλουθη εξίσωση:

$$\delta_j^{(l)} = \sum_{i=1}^{n^{(l+1)}} w_{ji}^{(l+1)} \delta_i^{(l+1)} \sigma' \left( \sum_{k=1}^{n^{(l-1)}} w_{jk}^{(l)} a_k^{(l-1)} + b_j^{(l)} \right)$$

όπου  $\delta_j^{(l)}$  είναι το σφάλμα του  $j$ -οστού νευρώνα στο  $l$ -οστό στρώμα,  $n^{(l+1)}$  είναι ο αριθμός των νευρώνων στο  $(l+1)$ -οστό στρώμα,  $w_{ji}^{(l+1)}$  είναι το βάρος που συνδέει τον  $j$ -οστό νευρώνα στο  $l$ -οστό στρώμα με τον  $i$ -οστό νευρώνα στο  $(l+1)$ -οστό στρώμα,  $\delta_i^{(l+1)}$  είναι το σφάλμα του  $i$ -οστού νευρώνα στο  $(l+1)$ -οστό στρώμα, και  $\sigma'(\cdot)$  είναι η παράγωγος της συνάρτησης ενεργοποίησης.

4. **Υπολογισμός της κλίσης των βαρών και των προκαταλήψεων:** Χρησιμοποιούνται τα σφάλματα που υπολογίστηκαν στο προηγούμενο βήμα για να υπολογίσετε την κλίση των βαρών και των προκαταλήψεων ως προς τη συνάρτηση απώλειας χρησιμοποιώντας τις ακόλουθες εξισώσεις:

$$\frac{\partial C}{\partial w_{ij}^{(l)}} = a_i^{(l-1)} \delta_j^{(l)}$$

$$\frac{\partial C}{\partial b_j^{(l)}} = \delta_j^{(l)}$$

5. **Ενημέρωση των βαρών και των προκαταλήψεων:** Χρησιμοποιούνται οι κλίσεις που υπολογίστηκαν στο προηγούμενο βήμα για να ενημερωθούν τα βάρη και οι προκαταλήψεις χρησιμοποιώντας την κάθοδο κλίσης:

$$w_{ij}^{(l)} = w_{ij}^{(l)} - \alpha \frac{\partial C}{\partial w_{ij}^{(l)}}$$

$$b_j^{(l)} = b_j^{(l)} - \alpha \frac{\partial C}{\partial b_j^{(l)}}$$

όπου  $\alpha$  είναι ο ρυθμός μάθησης, ο οποίος καθορίζει το μέγεθος της ενημέρωσης των βαρών και των προκαταλήψεων.

Αυτή η διαδικασία επαναλαμβάνεται για κάθε παράδειγμα εκπαίδευσης στο σύνολο δεδομένων και για πολλές εποχές, έως ότου η συνάρτηση απώλειας ελαχιστοποιηθεί σε ένα ικανοποιητικό επίπεδο.

---

### Αλγόριθμος 3.8 Οπισθοδιάδοση - μαθηματικός συμβολισμός

---

- 1: **procedure** BACKPROPAGATION( $X, y, \alpha, L$ )
  - 2:     Αρχικοποίηση βαρών  $w_{ij}^{(l)}$  και των προκαταλήψεων  $b_j^{(l)}$  για κάθε επίπεδο  $l$
  - 3:     **for** κάθε εποχή **do**
  - 4:         **for** κάθε δείγμα  $(x, y) \in (X, y)$  **do**
  - 5:             Εμπρόσθια διάδοση:  $a_j^{(l)} \leftarrow \sigma \left( \sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \right)$
  - 6:             Σφάλμα εξόδου:  $\delta_j^{(L)} \leftarrow \frac{\partial C}{\partial a_j^{(L)}} \sigma' \left( \sum_{i=1}^{n^{(L-1)}} w_{ij}^{(L)} a_i^{(L-1)} + b_j^{(L)} \right)$
  - 7:             Οπισθοδιάδοση σφάλματος:  $\delta_j^{(l)} \leftarrow \sum_{i=1}^{n^{(l+1)}} w_{ji}^{(l+1)} \delta_i^{(l+1)} \sigma' \left( \sum_{k=1}^{n^{(l-1)}} w_{jk}^{(l)} a_k^{(l-1)} + b_j^{(l)} \right)$
  - 8:             Υπολογισμός των κλίσεων:  $\frac{\partial C}{\partial w_{ij}^{(l)}} \leftarrow a_i^{(l-1)} \delta_j^{(l)}$   $\frac{\partial C}{\partial b_j^{(l)}} \leftarrow \delta_j^{(l)}$
  - 9:             Ενημέρωση βαρών, προκαταλήψεων:  $w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \alpha \frac{\partial C}{\partial w_{ij}^{(l)}}$   $b_j^{(l)} \leftarrow b_j^{(l)} - \alpha \frac{\partial C}{\partial b_j^{(l)}}$
  - 10:         **end for**
  - 11:     **end for**
  - 12: **end procedure**
-

**Άλγοριθμος 3.9** Οπισθοδιάδοση - ψευδοκώδικας

---

```

1: procedure Οπισθοδιάδοση( $X, y, \alpha, L$ )
2:   Αρχικοποίηση βαρών  $w_{ij}^{(l)}$  και των προκαταλήψεων  $b_j^{(l)}$  για κάθε επίπεδο  $l$ 
3:   for κάθε εποχή do
4:     for κάθε δείγμα  $(x, y) \in (X, y)$  do
5:       Εμπρόσθια διάδοση:
6:       for κάθε επίπεδο  $l$  do
7:          $a_j^{(l)} \leftarrow \sigma \left( \sum_{i=1}^{n^{(l-1)}} w_{ij}^{(l)} a_i^{(l-1)} + b_j^{(l)} \right)$ 
8:       end for
9:       Ύπολογίστε το σφάλμα εξόδου:
10:      for κάθε νευρώνα  $j$  στο επίπεδο εξόδου do
11:         $\delta_j^{(L)} \leftarrow \frac{\partial C}{\partial a_j^{(L)}} \sigma' \left( \sum_{i=1}^{n^{(L-1)}} w_{ij}^{(L)} a_i^{(L-1)} + b_j^{(L)} \right)$ 
12:      end for
13:      Σφάλμα οπισθοδιάδοσης:
14:      for κάθε επίπεδο  $l$  do
15:         $\delta_j^{(l)} \leftarrow \sum_{i=1}^{n^{(l+1)}} w_{ji}^{(l+1)} \delta_i^{(l+1)} \sigma' \left( \sum_{k=1}^{n^{(l-1)}} w_{jk}^{(l)} a_k^{(l-1)} + b_j^{(l)} \right)$ 
16:      end for
17:      Ύπολογισμός των κλίσεων:
18:      for κάθε βάρος  $w_{ij}^{(l)}$  do
19:         $\kappa\lambda\sigma\eta_{w_{ij}^{(l)}} \leftarrow a_i^{(l-1)} \delta_j^{(l)}$ 
20:      end for
21:      for κάθε προκατάληψη  $b_j^{(l)}$  do
22:         $\kappa\lambda\sigma\eta_{b_j^{(l)}} \leftarrow \delta_j^{(l)}$ 
23:      end for
24:      Ενημέρωση βαρών και προκαταλήψεων:
25:      for κάθε βάρος  $w_{ij}^{(l)}$  do
26:         $w_{ij}^{(l)} \leftarrow w_{ij}^{(l)} - \alpha \cdot \kappa\lambda\sigma\eta_{w_{ij}^{(l)}}$ 
27:      end for
28:      for κάθε προκατάληψη  $b_j^{(l)}$  do
29:         $b_j^{(l)} \leftarrow b_j^{(l)} - \alpha \cdot \kappa\lambda\sigma\eta_{b_j^{(l)}}$ 
30:      end for
31:    end for
32:  end for
33: end procedure

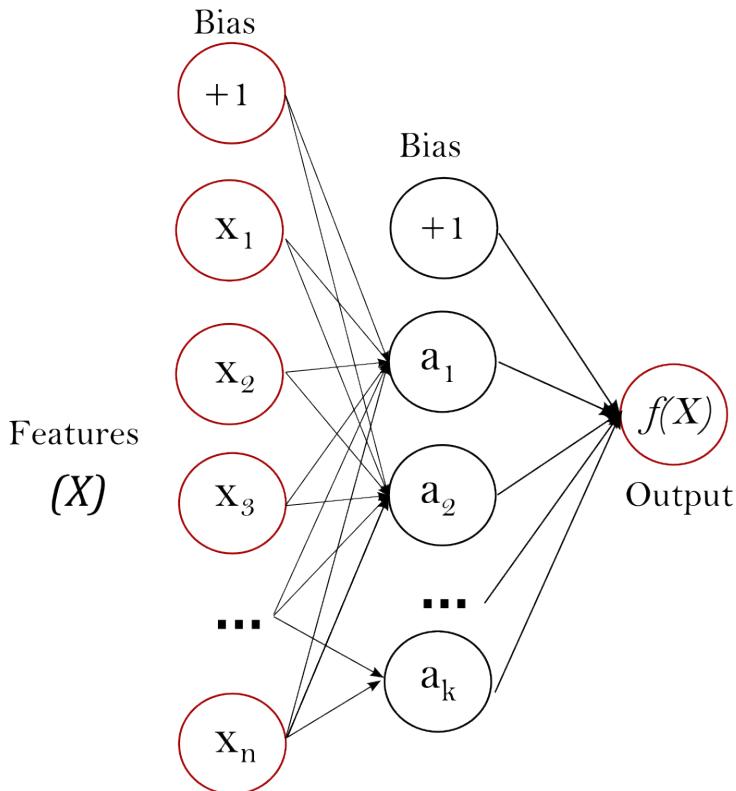
```

---

## 3.4 PERCEPTRON ΠΟΛΛΑΠΛΩΝ ΕΠΙΠΕΔΩΝ

### 3.4.1 Multi-layer Perceptron

Το Multi-layer Perceptron (MLP) είναι ένας αλγόριθμος μάθησης με επίβλεψη που μαθαίνει μια συνάρτηση  $f(\cdot) : R^m \rightarrow R^o$  με εκπαίδευση σε ένα σύνολο δεδομένων, όπου  $m$  είναι ο αριθμός των διαστάσεων για την είσοδο και  $o$  είναι ο αριθμός των διαστάσεων για την έξοδο. Δεδομένου ενός συνόλου χαρακτηριστικών  $X = x_1, x_2, \dots, x_m$  και έναν στόχο  $y$ , μπορεί να μάθει μια μη γραμμική συνάρτηση προσέγγισης είτε για ταξινόμηση είτε για παλινδρόμηση. Διαφέρει από τη λογιστική παλινδρόμηση, στο ότι μεταξύ του στρώματος εισόδου και του στρώματος έξοδου, μπορεί να υπάρχουν ένα ή περισσότερα μη γραμμικά στρώματα, που ονομάζονται ακρυφά στρώματα. Το σχήμα 3.32 δείχνει ένα MLP με ένα ακρυφό στρώμα και ακιμακωτή έξοδο.



Σχήμα 3.32: Perceptron πολλαπλών στρώμάτων με 1 ακρυφό στρώμα [69]

Το αριστερότερο στρώμα, γνωστό ως στρώμα εισόδου, αποτελείται από ένα σύνολο νευρώνων  $\{x_i | x_1, x_2, \dots, x_m\}$  που αντιπροσωπεύουν τα χαρακτηριστικά εισόδου. Κάθε νευρώνας στο ακρυφό στρώμα μετασχηματίζει τις τιμές από το προηγούμενο στρώμα με ένα σταθμισμένο γραμμικό άθροισμα  $w_1x_1 + w_2x_2 + \dots + w_mx_m$ , ακολουθούμενο από μια μη γραμμική συνάρτηση ενεργοποίησης  $g(\cdot) : R \rightarrow R$  - όπως η υπερβολική συνάρτηση tan. Το στρώμα έξοδου λαμβάνει τις τιμές από το τελευταίο ακρυφό στρώμα και τις μετατρέπει σε τιμές έξοδου.

Η ενότητα του scikit-learn περιέχει τα δημόσια χαρακτηριστικά `coefs_` και `intercepts_`.

### 3.4. PERCEPTRON ΠΟΛΛΑΠΛΩΝ ΕΠΙΠΕΔΩΝ

---

Το `coefs_` είναι ένας κατάλογος πινάκων βαρών, όπου ο πίνακας βαρών στον δείκτη  $i$  αντιπροσωπεύει τα βάρη μεταξύ του στρώματος  $i$  και του στρώματος  $i + 1$ . Το `intercepts_` είναι ένας κατάλογος διανυσμάτων μεροληφίας, όπου το διάνυσμα στον δείκτη  $i$  αντιπροσωπεύει τις τιμές μεροληφίας που προστίθενται στο στρώμα  $i + 1$ .

Τα πλεονεκτήματα του Perceptron πολλαπλών στρωμάτων είναι:

- Ικανότητα εκμάθησης μη γραμμικών μοντέλων.
- Δυνατότητα εκμάθησης μοντέλων σε πραγματικό χρόνο (on-line μάθηση) με χοήση.

Τα μειονεκτήματα του Perceptron πολλαπλών στρωμάτων περιλαμβάνουν:

- Τα MLP με κρυφά στρώματα έχουν μια μη κυρτή συνάρτηση απώλειών όπου υπάρχουν περισσότερα από ένα τοπικά ελάχιστα. Ως εκ τούτου, διαφορετικές τυχαίες αρχικοποιήσεις βαρών μπορούν να οδηγήσουν σε διαφορετική ακρίβεια επικύρωσης.
- Το MLP απαιτεί τη ρύθμιση ενός αριθμού υπερπαραμέτρων, όπως ο αριθμός των κρυφών νευρώνων, των στρωμάτων και των επαναλήψεων.
- Το MLP είναι ευαίσθητο στην κλιμάκωση των χαρακτηριστικών.

#### 3.4.2 Ταξινόμηση

Η κλάση `MLPClassifier` του `sklearn` εφαρμόζει έναν αλγόριθμο πολλαπλών επιπέδων `perceptron` (MLP) που εκπαιδεύεται χρησιμοποιώντας Backpropagation.

Το MLP εκπαιδεύεται σε δύο πίνακες: ένας πίνακας μεγέθους (`n_samples`, `n_features`), ο οποίος διατηρεί τα δείγματα εκπαίδευσης που αντιπροσωπεύονται ως διανύσματα χαρακτηριστικών κινητής υποδιαστολής, και ο πίνακας `y` μεγέθους (`n_samples`), ο οποίος διατηρεί τις τιμές-στόχους (επικέτες κλάσεων) για τα δείγματα εκπαίδευσης. Μετά την προσαρμογή (εκπαίδευση), το μοντέλο μπορεί να προβλέψει επικέτες για νέα δείγματα.

Επί του παρόντος, ο `MLPClassifier` υποστηρίζει μόνο τη συνάρτηση απώλειας Cross-Entropy, η οποία επιτρέπει εκτιμήσεις πιθανοτήτων εκτελώντας τη μέθοδο `predict_proba`.

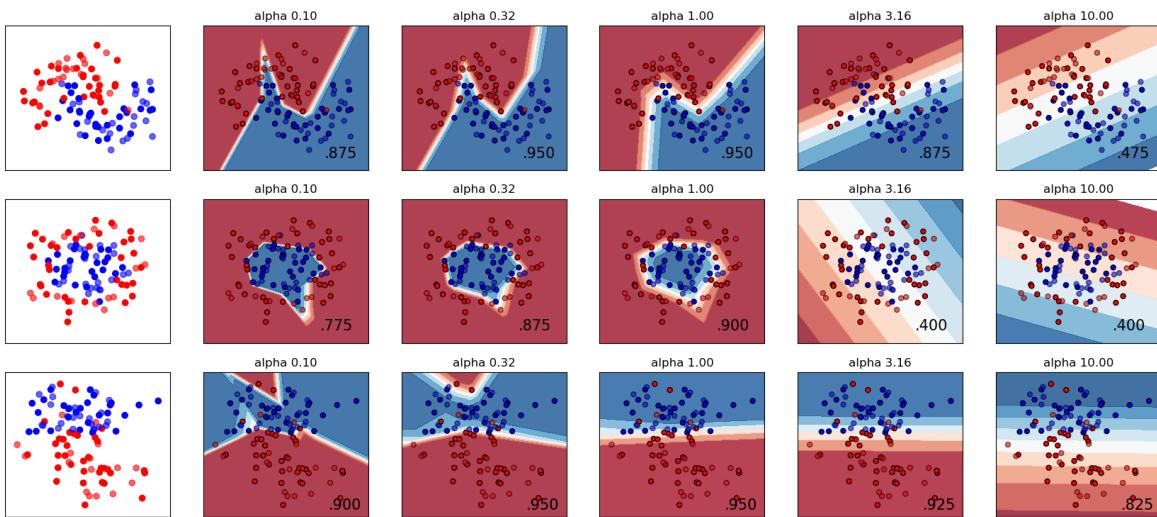
Το MLP εκπαιδεύεται χρησιμοποιώντας Backpropagation. Πιο συγκεκριμένα, προπονείται χρησιμοποιώντας κάποια μορφή gradient descent και οι κλίσεις υπολογίζονται χρησιμοποιώντας Backpropagation. Στην ταξινόμηση, ελαχιστοποιεί τη συνάρτηση απώλειας Cross-Entropy, δίνοντας ένα διάνυσμα εκτιμήσεων πιθανοτήτων  $P(y|x)$  ανά δείγμα  $x$ .

Το `MLPClassifier` υποστηρίζει ταξινόμηση πολλαπλών κλάσεων εφαρμόζοντας τη Softmax ως συνάρτηση εξόδου. Επιπλέον, το μοντέλο υποστηρίζει ταξινόμηση πολλαπλών επικετών στην οποία ένα δείγμα μπορεί να ανήκει σε περισσότερες από μία κλάσεις. Για κάθε κλάση, η ακατέργαστη έξοδος διέρχεται από την λογιστική

συνάρτηση. Τιμές μεγαλύτερες ή ίσες με 0.5 στρογγυλοποιούνται στο 1, διαφορετικά στο 0. Για μια προβλεπόμενη έξοδο ενός δείγματος, οι δείκτες όπου η τιμή είναι 1 αντιπροσωπεύουν τις εκχωρημένες κατηγορίες αυτού του δείγματος.

### 3.4.3 Όρος Κανονικοποίησης

Τόσο ο MLPRegressor όσο και ο MLPClassifier χρησιμοποιούν την παράμετρο alpha για τον όρο κανονικοποίησης (L2 regularization), ο οποίος βοηθά στην αποφυγή της υπερπροσαρμογής, τιμωρώντας τα βάρη με μεγάλα μεγέθη. Το σχήμα 3.33 παρουσιάζει τη μεταβαλλόμενη συνάρτηση απόφασης με την τιμή του alpha.



Σχήμα 3.33: Μεταβαλλόμενη συνάρτηση απόφασης με την τιμή του alpha [69]

### 3.4.4 Αλγόριθμοι

Το MLP εκπαιδεύεται με τη χρήση του Stochastic Gradient Descent, Adam ή L-BFGS. Η στοχαστική κάθιση (SGD) ενημερώνει τις παραμέτρους χρησιμοποιώντας την κλίση της συνάρτησης απώλειας σε σχέση με μια παράμετρο που χρειάζεται προσαρμογή, δηλαδή

$$w \leftarrow w - \eta(\alpha \frac{\partial R(w)}{\partial w} + \frac{\partial Loss}{\partial w})$$

όπου η είναι ο ρυθμός μάθησης που ελέγχει το μέγεθος του βήματος κατά την αναζήτηση στο χώρο των παραμέτρων. Το Loss είναι η συνάρτηση απωλειών που χρησιμοποιείται για το δίκτυο. Ο αλγόριθμος SGD έχει περιγραφεί αναλυτικά στην ενότητα 3.2.9.

Το Adam είναι παρόμοιο με το SGD υπό την έννοια ότι είναι ένας στοχαστικός βελτιστοποιητής, αλλά μπορεί να προσαρμόσει αυτόματα το ποσό ενημέρωσης των παραμέτρων με βάση προσαρμοστικές εκτιμήσεις των ροπών χαμηλότερης τάξης.

Με το SGD ή το Adam, η εκπαίδευση υποστηρίζει τη μάθηση σε απευθείας σύνδεση (on-line) και τη μάθηση σε μίνι παρτίδες (mini-batch).

Ο L-BFGS είναι ένας επιλύτης που προσεγγίζει τον πίνακα Hessian, ο οποίος αντιπροσωπεύει τη μερική παράγωγο δεύτερης τάξης μιας συνάρτησης. Περαιτέρω προσεγγίζει τον αντίστροφο του πίνακα Hessian για να εκτελεί ενημερώσεις παραμέτρων. Η υλοποίηση χρησιμοποιεί την έκδοση Scipy του L-BFGS.

Εάν ο επιλεγμένος επιλύτης είναι ο 'L-BFGS', η εκπαίδευση δεν υποστηρίζει on-line μάθηση ούτε μάθηση σε μίνι παρτίδες.

### 3.4.5 Πολυπλοκότητα

Ας υποτεθεί ότι υπάρχουν  $n$  δείγματα εκπαίδευσης,  $m$  χαρακτηριστικά,  $k$  κρυφά στρώματα, καθένα από τα οποία περιέχει  $h$  νευρώνες (για λόγους απλότητας), και ο νευρώνες εξόδου. Η χρονική πολυπλοκότητα της οπισθοδιάδοσης είναι  $O(n \cdot m \cdot h^k \cdot o \cdot i)$ , όπου  $i$  είναι ο αριθμός των επαναλήψεων. Δεδομένου ότι η οπισθοδιάδοση έχει υψηλή χρονική πολυπλοκότητα, είναι σκόπιμο να ξεκινήσει η εκπαίδευση με μικρό αριθμό κρυφών νευρώνων και λίγα κρυφά στρώματα και σταδιακά να αυξάνονται αν τα αποτελέσματα δεν είναι ικανοποιητικά.

### 3.4.6 Μαθηματική διατύπωση

Δεδομένου ενός συνόλου δειγμάτων εκπαίδευσης  $(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)$  όπου  $x_i \in \mathbf{R}^n$  και  $y_i \in \{0, 1\}$ , ένα MLP με ένα κρυφό στρώμα και έναν κρυφό νευρώνα μαθαίνει τη συνάρτηση

$$f(x) = W_2g(W_1^T x + b_1) + b_2$$

όπου  $W_1 \in \mathbf{R}^m$  και  $W_2, b_1, b_2 \in \mathbf{R}$  είναι παράμετροι του μοντέλου. Τα  $W_1, W_2$  αντιπροσωπεύουν τα βάρη του στρώματος εισόδου και του κρυφού στρώματος, αντίστοιχα, και τα  $b_1, b_2$  αντιπροσωπεύουν την προκατάληψη που προστίθεται στο κρυφό στρώμα και στο στρώμα εξόδου, αντίστοιχα. Η  $g(\cdot) : R \rightarrow R$  είναι η συνάρτηση ενεργοποίησης, η οποία έχει οριστεί από προεπιλογή ως το υπερβολικό tan. Δίνεται ως εξής,

$$g(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

Για δυαδική ταξινόμηση, η  $f(x)$  περνάει μέσα από τη λογιστική συνάρτηση  $g(z) = 1/(1 + e^{-z})$  για να λάβει τιμές εξόδου μεταξύ μηδέν και ένα. Ένα κατώφλι, που ορίζεται σε 0.5, θα αναθέσει τα δείγματα εξόδων μεγαλύτερα ή ίσα με 0.5 στη θετική κλάση και τα υπόλοιπα στην αρνητική κλάση.

Εάν υπάρχουν περισσότερες από δύο κλάσεις, η ίδια η  $f(x)$  θα είναι ένα διάνυσμα μεγέθους ( $n$  classes). Αντί να περάσει από τη λογιστική συνάρτηση, περνάει από τη συνάρτηση softmax, η οποία γράφεται ως εξής,

$$\text{softmax}(z)_i = \frac{\exp(z_i)}{\sum_{l=1}^k \exp(z_l)}$$

όπου το  $z_i$  αντιπροσωπεύει το  $i$ -οστό στοιχείο της εισόδου στο softmax, το οποίο αντιστοιχεί στην κλάση  $i$ , και είναι ο αριθμός των κλάσεων. Το αποτέλεσμα είναι

ένα διάγυσμα που περιέχει τις πιθανότητες ότι το δείγμα  $x$  ανήκει σε κάθε κλάση. Η έξοδος είναι η κλάση με την υψηλότερη πιθανότητα.

Το MLP χρησιμοποιεί διαφορετικές συναρτήσεις απωλειών ανάλογα με τον τύπο του προβλήματος. Η συνάρτηση απωλειών για ταξινόμηση είναι η Average Cross-Entropy, η οποία στη δυαδική περίπτωση δίνεται ως εξής,

$$Loss(\hat{y}, y, W) = -\frac{1}{n} \sum_{i=0}^n (y_i \ln \hat{y}_i + (1 - y_i) \ln (1 - \hat{y}_i)) + \frac{\alpha}{2n} \|W\|_2^2$$

όπου το  $\alpha \|W\|_2^2$  είναι ένας όρος κανονικοποίησης L2 (ή αλλιώς ποινή) που τιμωρεί πολύπλοκα μοντέλα και το  $\alpha > 0$  είναι μια μη αρνητική υπερπαράμετρος που ελέγχει το μέγεθος της ποινής.

Για την παλινδρόμηση, το MLP χρησιμοποιεί τη συνάρτηση απώλειας Μέσου Τετραγωνικού Σφάλματος, η οποία γράφεται ως εξής,

$$Loss(\hat{y}, y, W) = \frac{1}{2n} \sum_{i=0}^n \|\hat{y}_i - y_i\|_2^2 + \frac{\alpha}{2n} \|W\|_2^2$$

Ξεκινώντας από τα αρχικά τυχαία βάρη, το πολυεπίπεδο perceptron (MLP) ελαχιστοποιεί τη συνάρτηση απώλειας ενημερώνοντας επανειλημμένα αυτά τα βάρη. Μετά τον υπολογισμό της απώλειας, ένα αντίστροφο πέρασμα τη μεταδίδει από το στρώμα εξόδου στα προηγούμενα στρώματα, παρέχοντας σε κάθε παράμετρο βάρους μια τιμή ενημέρωσης που αποσκοπεί στη μείωση της απώλειας.

Στην κάθιδο κλίσης, η κλίση  $\nabla Loss_W$  της απώλειας ως προς τα βάρη υπολογίζεται και αφαιρείται από το  $W$ . Πιο τυπικά, αυτό εκφράζεται ως

$$W^{i+1} = W^i - \epsilon \nabla Loss_W^i$$

όπου  $i$  είναι το βήμα επανάληψης και  $\epsilon$  είναι ο ρυθμός μάθησης με τιμή μεγαλύτερη από 0.

Ο αλγόριθμος σταματά όταν φτάσει σε έναν προκαθορισμένο μέγιστο αριθμό επαναλήψεων ή όταν η βελτίωση της απώλειας είναι κάτω από έναν ορισμένο, μικρό αριθμό.

# 4

## Εργαλεία και Τεχνικές

Για λόγους πληρότητας, στο κεφάλαιο αυτό παρουσιάζονται τα βασικά εργαλεία που χρησιμοποιήθηκαν τόσο για την εκπαίδευση των αλγορίθμων (hardware) όσο και για τις υλοποιήσεις αυτών (software). Επίσης θα αναλυθούν οι μετρικές που θα χρησιμοποιηθούν, οι τεχνικές επικύρωσης καθώς και οι τρόποι με τους οποίους θα βρεθούν οι καλύτερες παράμετροι για τις μεθόδους επιλογής χαρακτηριστικών αλλά και για τις υπερπαραμέτρους των αλγορίθμων μηχανικής μάθησης.

### 4.1 HARDWARE

Τα βασικά μέρη του συστήματος που χρησιμοποιήθηκε για την εκπαίδευση των αλγορίθμων είναι τα εξής :

- Ο επεργαστής Intel® Core™ i5-8400<sup>8</sup>, ο οποίος είναι χτισμένος στην αρχιτεκτονική Coffee Lake και διαθέτει 9MB μνήμης cache. Είναι συμβατός με μητρικές πλακέτες που διαθέτουν υποδοχή LGA 1151 και chipset της σειράς Intel 300, επομένως η CPU μπορεί να παρέχει έως και δεκαέξι γραμμές PCIe 3.0 για μονάδες αποθήκευσης NVMe, κάρτες γραφικών και άλλα εξαρτήματα υψηλής απόδοσης. Σε αυτό το σημείο να αναφερθεί ότι μια κάρτα γραφικών (GPU) θα έκανε την διαδικασία εκπαίδευσης πολύ πιο γρήγορη.

<sup>8</sup>Όλες οι προδιαγραφές του Intel® Core™ i5-8400 : <https://www.intel.com/content/www/us/en/products/sku/126687/intel-core-i58400-processor-9m-cache-up-to-4-00-ghz/specifications.html>

## ΚΕΦΑΛΑΙΟ 4. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΙΚΕΣ

---

Πίνακας 4.1: Βασικές προδιαγραφές του επεξεργαστή Intel® Core™ i5-8400

Μονάδα	Intel® Core™ i5-8400
Αριθμός Πυρήνων	6
Αριθμός Νημάτων	6
Μέγιστη συχνότητα επεξεργαστή	4.00 GHz
Μέγιστη συχνότητα πυρήνα	4.00 GHz
Βασική συχνότητα επεξεργαστή	2.80 GHz
Προσωρινή μνήμη Cache	9 MB
Ταχύτητα Διαύλου	8 GT/s
Ισχύς (TDP)	65 W

- Δύο μνήμες RAM HyperX HX426C16FB3/4<sup>9</sup> σε διπλό κανάλι. Αυτή είναι μια μονάδα μνήμης 512M x 64-bit (4GB) DDR4-2666 CL16 SDRAM (Σύγχρονη DRAM) 1Rx8, που βασίζεται σε οκτώ στοιχεία FBGA 512M x 8-bit ανά μονάδα. Κάθε μονάδα υποστηρίζει Intel® Extreme Memory Profiles (Intel® XMP) 2.0. Κάθε μονάδα έχει δοκιμαστεί για να λειτουργεί σε DDR4-2666 σε χαμηλό χρόνο καθυστέρησης 16-18-18 στα 1.2 V. Οι ηλεκτρικές και μηχανολογικές προδιαγραφές του προτύπου JEDEC είναι οι εξής:

Πίνακας 4.2: Βασικές προδιαγραφές της μνήμης HyperX HX426C16FB3/4

Μονάδα	HyperX HX426C16FB3/4
CL(IDD)	16 κύκλους
Χρόνος κύκλου σειράς (tRCmin)	45.75 ns(min.)
Ανανέωση σε Ενεργή/ Χρόνος Εντολής Ανανέωσης (tRFCmin)	260 ns(min.)
Ενεργός χρόνος σειράς (tRASmin)	29.25 ns(min.)
Βαθμολογία UL	94 V - 0
Θερμοκρασία λειτουργίας	0 °C to +85 °C
Θερμοκρασία αποθήκευσης	-55 °C to +100 °C

<sup>9</sup>Όλες οι προδιαγραφές της HyperX HX426C16FB3/4 : [https://www.kingston.com/dataSheets/HX426C16FB3\\_4.pdf](https://www.kingston.com/dataSheets/HX426C16FB3_4.pdf)

## 4.2 ΕΡΓΑΛΕΙΑ ΛΟΓΙΣΜΙΚΟΥ

Εξαιτίας της ραγδαίας εξέλιξης της επιστήμης της μηχανικής μάθησης, τα τελευταία χρόνια έχουν αναπτυχθεί πολλά εργαλεία λογισμικού (βιβλιοθήκες, SDKs, frameworks) για γρήγορη ή/και αποτελεσματική σχεδίαση και υλοποίηση πολυεπίπεδων νευρωνικών δικτύων καθώς και για αλγορίθμους μηχανικής μάθησης όπως για παράδειγμα αυτοί που αναφέρθηκαν στο υποκεφάλαιο 3.2.

Μερικά από τα πιο γνωστά εργαλεία για αυτούς τους σκοπούς, τα οποία χρησιμοποιούνται σε αυτήν την εργασία είναι:

- Visual Studio Code : Το Visual Studio Code, που αναφέρεται συνήθως ως VS Code, είναι ένας επεξεργαστής πηγαίου κώδικα που κατασκευάζεται από τη Microsoft με το Electron Framework, για Windows, Linux και macOS. Τα χαρακτηριστικά του περιλαμβάνουν υποστήριξη για αποσφαλμάτωση, επισήμανση συντακτικού, έξυπνη συμπλήρωση κώδικα, αποσπάσματα, αναδιαμόρφωση κώδικα και ενσωματωμένο Git. Οι χρήστες μπορούν να αλλάξουν το θέμα, τις συντομεύσεις πληκτρολογίου, τις προτιμήσεις και να εγκαταστήσουν επεκτάσεις που προσθέτουν πρόσθετη λειτουργικότητα.
- Python <sup>10</sup> : Διερμηνευόμενη (interpreted), γενικού σκοπού (general-purpose) και υψηλού επιπέδου, γλώσσα προγραμματισμού. Ανήκει στις γλώσσες προστακτικού προγραμματισμού (Imperative programming) και υποστηρίζει τόσο το διαδικαστικό (procedural programming) όσο και το αντικειμενοστρεφές (object-oriented programming) προγραμματιστικό υπόδειγμα (programming paradigm). Είναι δυναμική γλώσσα προγραμματισμού (dynamically typed) και υποστηρίζει συλλογή απορριμμάτων (garbage collection ή GC).
- SciKit Learn <sup>11</sup> : Βιβλιοθήκη μηχανικής μάθησης ανοιχτού κώδικα που υποστηρίζει την εποπτευόμενη και χωρίς επίβλεψη μάθηση. Παρέχει επίσης διάφορα εργαλεία για προσαρμογή μοντέλου, προεπεξεργασία δεδομένων, επιλογή μοντέλου, αξιολόγηση μοντέλου και πολλά άλλα βοηθητικά προγράμματα.
- Pandas <sup>12</sup> : Ένα γρήγορο, ισχυρό, ευέλικτο και εύκολο στη χρήση εργαλείο ανάλυσης και χειρισμού δεδομένων ανοιχτού κώδικα, χτισμένο πάνω στη γλώσσα προγραμματισμού Python.
- NumPy <sup>13</sup> : Βιβλιοθήκη Python που παρέχει ένα πολυδιάστατο αντικείμενο πίνακα, διάφορα παράγωγα αντικείμενα και μια ποικιλία από ρουτίνες για γρήγορες λειτουργίες σε πίνακες, συμπεριλαμβανομένων μαθηματικών, λογικών, χειρισμού σχήματος, ταξινόμησης, επιλογής, εισόδου/εξόδου, διαχριτούς μετασχηματισμούς Fourier, βασική γραμμική άλγεβρα, βασικές στατιστικές πράξεις, τυχαία προσομοίωση και πολλά άλλα.

<sup>10</sup><https://www.python.org/>

<sup>11</sup><https://scikit-learn.org/stable/>

<sup>12</sup><https://pandas.pydata.org/>

<sup>13</sup><https://numpy.org/>

- SciPy <sup>14</sup> : Μια συλλογή μαθηματικών αλγορίθμων και συναρτήσεων που έχουν δημιουργηθεί στην επέκταση NumPy της Python. Προσθέτει σημαντική ισχύ στη διαδραστική συνεδρία Python παρέχοντας στον χρήστη εντολές και κλάσεις υψηλού επιπέδου για χειρισμό και οπτικοποίηση δεδομένων.
- Matplotlib <sup>15</sup> : Ολοκληρωμένη βιβλιοθήκη για τη δημιουργία στατικών, κινούμενων και διαδραστικών απεικονίσεων στην Python.
- Seaborn <sup>16</sup> : Βιβλιοθήκη οπτικοποίησης δεδομένων Python που βασίζεται στο matplotlib. Παρέχει μια διεπαφή υψηλού επιπέδου για τη σχεδίαση ελκυστικών και ενημερωτικών στατιστικών γραφικών.

## 4.3 ΜΕΤΡΙΚΕΣ ΑΠΟΔΟΣΗΣ

---

Τα μοντέλα ταξινόμησης έχουν διακριτή έξοδο, επομένως χρειάζεται μια μετρική που να συγκρίνει διακριτές κλάσεις με κάποια μορφή. Οι μετρικές ταξινόμησης [70] αξιολογούν την απόδοση ενός μοντέλου και λένε πόσο καλή ή κακή είναι η ταξινόμηση, αλλά κάθε μία από αυτές την αξιολογεί με διαφορετικό τρόπο.

Για να γίνουν κατανοητές οι μετρικές που θα χρησιμοποιηθούν, θα χρειαστούν τα παρακάτω μεγέθη:

- TP, True Positive, Αληθώς Θετικά : Δηλώνει πόσα θετικά δείγματα κλάσεων προέβλεψε σωστά το μοντέλο
- TN, True Negative, Αληθώς Αρνητικά : Δηλώνει πόσα αρνητικά δείγματα κλάσεων προέβλεψε σωστά το μοντέλο.
- FP, False Positive, Ψευδώς Θετικά : Δηλώνει πόσα αρνητικά δείγματα κλάσεων προέβλεψε λανθασμένα το μοντέλο ως θετικά. Αυτός ο παράγοντας αντιπροσωπεύει το σφάλμα τύπου I στη στατιστική ονοματολογία.
- FN, False Negative, Ψευδώς Αρνητικά : Δηλώνει πόσα θετικά δείγματα κλάσεων προέβλεψε λανθασμένα το μοντέλο ως αρνητικά. Αυτός ο παράγοντας αντιπροσωπεύει σφάλμα τύπου II στη στατιστική ονοματολογία.
- P, Positive, όλα τα θετικά στον πληθυσμό
- N, Negative, όλα τα αρνητικά τον πληθυσμό

### 4.3.1 Ορθότητα (Accuracy)

Η ορθότητα ταξινόμησης είναι ίσως η απλούστερη μετρική που μπορεί να χρησιμοποιηθεί και να εφαρμοστεί και ορίζεται ως ο αριθμός των σωστών προβλέψεων διαιρεμένος με τον συνολικό αριθμό των προβλέψεων.

$$\frac{TP + TN}{\text{Sample Size}}$$

<sup>14</sup><https://scipy.org/>

<sup>15</sup><https://matplotlib.org/>

<sup>16</sup><https://seaborn.pydata.org/>

### 4.3.2 Ακρίβεια (Precision)

Η ακρίβεια είναι ο λόγος των αληθώς θετικών και του συνόλου των θετικών που έχουν προβλεφθεί :

$$\frac{TP}{TP + FP} = \frac{TP}{\text{All Positive Predictions}} =$$

Cancer patients correctly identified

---

Cancer patients correctly identified + Non cancer patients labelled as cancerous

Η μετρική ακρίβειας επικεντρώνεται στα σφάλματα τύπου I (FP). Ένα σφάλμα τύπου I συμβαίνει όταν απορρίπτεται μια αληθής μηδενική υπόθεση ( $H^0$ ). Έτσι, σε αυτή την περίπτωση, το σφάλμα τύπου I είναι η εσφαλμένη επισήμανση των μη καρκινοπαθών ως καρκινοπαθών.

Μια βαθμολογία ακρίβειας προς το 1 θα σημαίνει ότι το μοντέλο δεν χαρακτήρισε θετικά (καρκινικά) περισσότερα από όσα πράγματι υπάρχουν και είναι σε θέση να ταξινομήσει καλά μεταξύ της σωστής και της λανθασμένης επισήμανσης των καρκινοπαθών. Αυτό που δεν μπορεί να μετρήσει είναι η ύπαρξη σφάλματος τύπου II, το οποίο είναι τα φευδώς αρνητικά - περιπτώσεις όπου ένας καρκινικός ασθενής αναγνωρίζεται ως μη καρκινικός.

Ένα χαμηλό σκορ ακρίβειας (<0,5) σημαίνει ότι ο ταξινομητής έχει μεγάλο αριθμό φευδώς θετικών αποτελεσμάτων, τα οποία μπορεί να είναι αποτέλεσμα ανισόρροπης κλάσης ή μη συντονισμένων υπερπαραμέτρων του μοντέλου. Σε ένα πρόβλημα ανισόρροπης κλάσης, πρέπει τα δεδομένα να προετοιμαστούν εκ των προτέρων με υπερδειγματοληψία/υποδειγματοληψία (υποενότητα 5.1.3) ή εστιακή απώλεια, προκειμένου να περιοριστεί το FP/FN.

### 4.3.3 Ανάκληση (Recall)

Η ανάκληση είναι ουσιαστικά ο λόγος των αληθώς θετικών αποτελεσμάτων προς όλα τα θετικά αποτελέσματα που θα έπρεπε να προβλεφθούν :

$$\frac{TP}{TP + FN} = \frac{TP}{P} =$$

Cancer patients correctly identified

---

Cancer patients correctly identified + Cancer patients labelled as non-cancerous

Η μετρική ανάκλησης επικεντρώνεται στα σφάλματα τύπου II (FN). Ένα σφάλμα τύπου II συμβαίνει όταν γίνεται αποδεκτή μια φευδή μηδενική υπόθεση ( $H^0$ ). Έτσι, σε αυτή την περίπτωση, το σφάλμα τύπου-II είναι η εσφαλμένη επισήμανση καρκινικών ασθενών ως μη καρκινικών.

Η ανάκληση προς το 1 θα σημαίνει ότι το μοντέλο δεν έχασε κανένα αληθώς θετικό αποτέλεσμα και είναι σε θέση να ταξινομήσει καλά μεταξύ της σωστής και λανθασμένης επισήμανσης των καρκινικών ασθενών.

## ΚΕΦΑΛΑΙΟ 4. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΙΚΕΣ

---

Αυτό που δεν μπορεί να μετρήσει είναι η ύπαρξη σφάλματος τύπου I, το οποίο είναι τα φευδώς θετικά αποτελέσματα, δηλαδή οι περιπτώσεις κατά τις οποίες ένας μη καρκινικός ασθενής αναγνωρίζεται ως καρκινικός.

Ένα χαμηλό σκορ ανάκλησης ( $<0,5$ ) σημαίνει ότι ο ταξινομητής έχει μεγάλο αριθμό φευδώς αρνητικών αποτελεσμάτων, τα οποία μπορεί να είναι αποτέλεσμα ανισόρροπης κλάσης ή μη συντονισμένων υπερπαραμέτρων του μοντέλου.

Για να βελτιωθεί το μοντέλο, μπορεί να βελτιωθεί είτε η ακρίβεια είτε η ανάκληση - άλλα όχι και τα δύο! Δηλαδή η μείωση των Ψευδώς Αρνητικών δεν θα μεταβάλει άμεσα το ποσοστό των Ψευδώς Θετικών.

### 4.3.4 F1-score

Η μετρική F1-score χρησιμοποιεί έναν συνδυασμό ακρίβειας και ανάκλησης. Στην πραγματικότητα, η βαθμολογία F1 είναι ο αρμονικός μέσος όρος των δύο. Ο αρμονικός μέσος  $n$  αριθμών  $x_1, x_2, \dots, x_n$  ορίζεται ως:

$$H = \frac{n}{\frac{1}{x_1} + \frac{1}{x_2} + \dots + \frac{1}{x_n}}$$

Έτσι, η μετρική  $F_1$  είναι:

$$F_1 = 2 \cdot \frac{1}{\frac{1}{\text{precision}} + \frac{1}{\text{recall}}} = 2 \cdot \frac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$$

Τώρα, ένα υψηλό σκορ F1 συμβολίζει υψηλή ακρίβεια καθώς και υψηλή ανάκληση. Παρουσιάζει μια καλή ισορροπία μεταξύ ακρίβειας και ανάκλησης και δίνει καλά αποτελέσματα σε ανισόρροπα προβλήματα ταξινόμησης.

Μια χαμηλή βαθμολογία F1 δεν δίνει (σχεδόν) καμία πληροφορία - δίνει μόνο για την απόδοση σε ένα κατώφλι. Χαμηλή ανάκληση σημαίνει ότι το μοντέλο δεν τα πήγε καλά σε πολύ μεγάλο μέρος ολόκληρου του συνόλου δοκιμών. Χαμηλή ακρίβεια σημαίνει ότι, μεταξύ των περιπτώσεων που αναγνωρίστηκαν ως θετικές περιπτώσεις, τελικά δεν ήταν πολλές από αυτές σωστές.

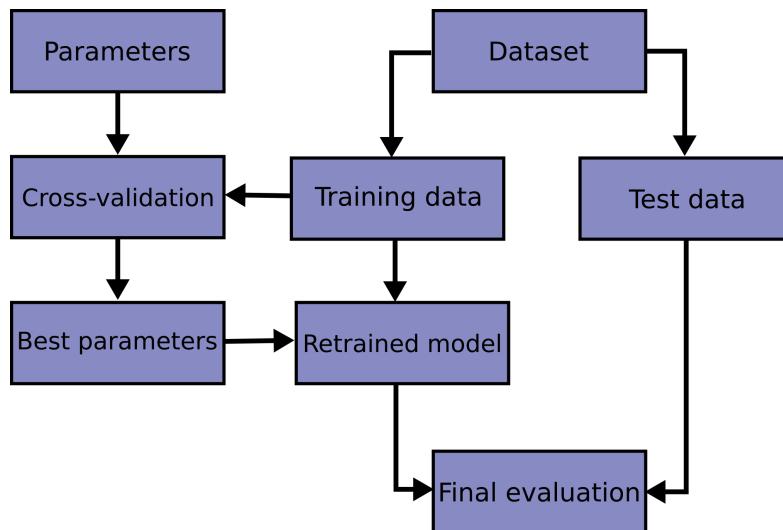
Άλλα το χαμηλό F1 δεν λέει ποιες περιπτώσεις. Το υψηλό F1 σημαίνει ότι πιθανότατα έχουμε υψηλή ακρίβεια και ανάκληση σε ένα μεγάλο μέρος της απόφασης (το οποίο είναι πληροφοριακό). Με χαμηλό F1, δεν είναι σαφές ποιο είναι το πρόβλημα (χαμηλή ακρίβεια ή χαμηλή ανάκληση;) και αν το μοντέλο πάσχει από σφάλμα τύπου I ή τύπου II.

Επομένως, η F1 χρησιμοποιείται ευρέως και θεωρείται μια καλή μετρική για τη σύγκλιση σε μια απόφαση, άλλα όχι χωρίς κάποιες βελτιώσεις. Η χρήση του FPR (ποσοστά φευδώς θετικών αποτελεσμάτων) μαζί με το F1 θα βοηθήσει στον περιορισμό των σφαλμάτων τύπου I.

Για το συγκεκριμένο πρόβλημα, κρίθηκε ορθότερο να χρησιμοποιηθεί η μετρική F1-score.

## 4.4 ΓΕΝΙΚΕΥΜΕΝΗ ΑΞΙΟΛΟΓΗΣΗ

Η εκμάθηση των παραμέτρων μιας συνάρτησης πρόβλεψης και η δοκιμή της στα ίδια δεδομένα είναι μεθοδολογικό λάθος. Θα ήταν ένα μοντέλο που απλώς θα επαναλάμβανε τις επικέτες των δειγμάτων που μόλις είδε θα είχε τέλεια βαθμολογία, αλλά δεν θα μπορούσε να προβλέψει τίποτα χρήσιμο σε δεδομένα που δεν θα είχε ακόμη δει. Αυτή η κατάσταση ονομάζεται υπερπροσαρμογή. Για να αποφευχθεί, είναι κοινή πρακτική κατά την εκτέλεση ενός πειράματος μηχανικής μάθησης (με επίβλεψη) να χρωτιέται μέρος των διαθέσιμων δεδομένων ως σύνολο δοκιμής. Να σημειωθεί ότι η λέξη "πείραμα" δεν έχει σκοπό να υποδηλώσει μόνο ακαδημαϊκή χρήση, διότι ακόμη και σε εμπορικά περιβάλλοντα η μηχανική μάθηση συνήθως ξεκινά πειραματικά. Στο σχήμα 4.1 φαίνεται ένα διάγραμμα ροής της τυπικής ροής εργασιών διασταυρούμενης επικύρωσης στην εκπαίδευση μοντέλων. Οι καλύτερες παράμετροι μπορούν να προσδιοριστούν με τεχνικές αναζήτησης πλέγματος που θα αναφερθούν στο υπόκεφαλο 4.5.



Σχήμα 4.1: Τυπική ροή εργασιών διασταυρούμενης επικύρωσης [71]

### 4.4.1 Διασταυρούμενη Επικύρωση

Κατά την αξιολόγηση διαφορετικών ρυθμίσεων ("υπερπαραμέτρων") για εκτιμήσεις, όπως για παράδειγμα η ρύθμιση C που πρέπει να ρυθμιστεί χειροκίνητα για ένα SVM, εξακολουθεί να υπάρχει κίνδυνος υπερπροσαρμογής στο σύνολο δοκιμών, επειδή οι παράμετροι μπορούν να ρυθμιστούν έως ότου ο εκτιμητής αποδώσει βέλτιστα. Με αυτόν τον τρόπο, η γνώση σχετικά με το σύνολο δοκιμών μπορεί να "διαρρεύσει" στο μοντέλο και οι μετρικές αξιολόγησης δεν αναφέρουν πλέον την απόδοση γενίκευσης. Για την επίλυση αυτού του προβλήματος, ένα ακόμη μέρος του συνόλου δεδομένων μπορεί να χρησιμεύει ως το λεγόμενο "σύνολο επικύρωσης". Η εκπαίδευση συνεχίζεται στο σύνολο εκπαίδευσης, μετά από αυτό η αξιολόγηση γίνεται στο σύνολο επικύρωσης, και όταν το πείραμα φαίνεται να είναι επιτυχές, η τελική αξιολόγηση μπορεί να γίνει στο σύνολο δοκιμής.

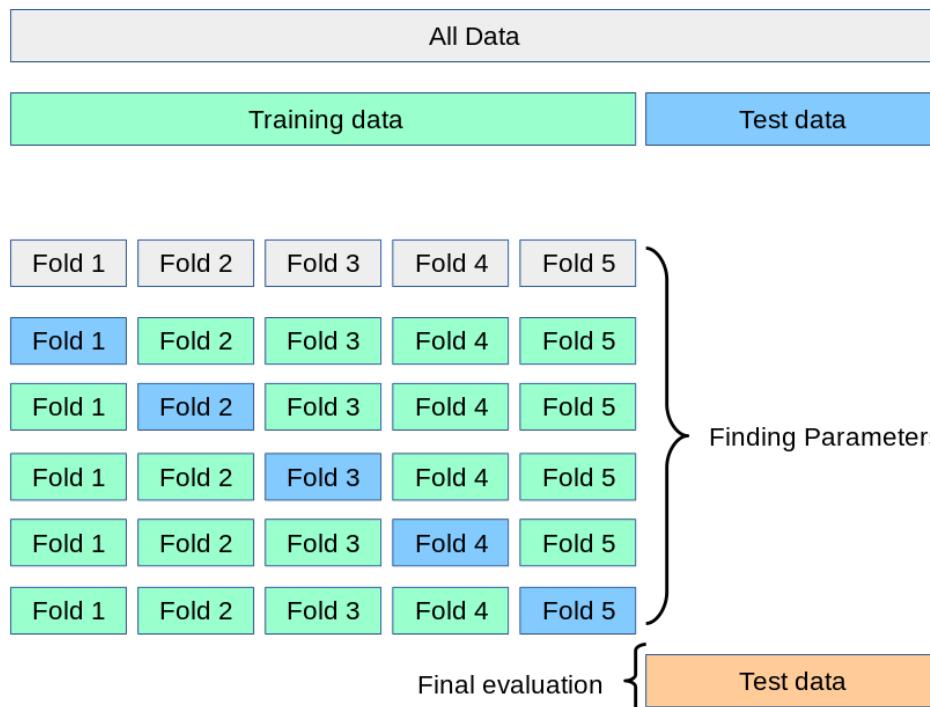
## ΚΕΦΑΛΑΙΟ 4. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΙΚΕΣ

Ωστόσο, χωρίζοντας τα διαθέσιμα δεδομένα σε τρία σύνολα, μειώνουμε δραστικά τον αριθμό των δειγμάτων που μπορούν να χρησιμοποιηθούν για την εκμάθηση του μοντέλου, και τα αποτελέσματα μπορεί να εξαρτώνται από μια συγκεκριμένη τυχαία επιλογή για το ζεύγος των συνόλων (εκπαίδευσης, επικύρωσης).

Μια λύση σε αυτό το πρόβλημα είναι μια διαδικασία που ονομάζεται διασταυρούμενη επικύρωση (cross-validation, έν συντομία CV). Ένα σύνολο δοκιμής θα πρέπει να εξακολουθεί να διατηρείται για την τελική αξιολόγηση, αλλά το σύνολο επικύρωσης δεν είναι πλέον απαραίτητο όταν γίνεται CV. Στη βασική προσέγγιση, που ονομάζεται k-fold CV, το σύνολο εκπαίδευσης χωρίζεται σε k μικρότερα σύνολα (υπάρχουν και άλλες προσεγγίσεις, αλλά γενικά ακολουθούν τις ίδιες αρχές). Για κάθε ένα από τα k "διπλώματα" ακολουθείται η ακόλουθη διαδικασία (σχήμα 4.2):

- Εκπαιδεύεται ένα μοντέλο χρησιμοποιώντας τις  $k - 1$  αναδιπλώσεις ως δεδομένα εκπαίδευσης,
- το μοντέλο που προκύπτει επικυρώνεται στο υπόλοιπο μέρος των δεδομένων (δηλαδή χρησιμοποιείται ως σύνολο δοκιμής για την υπολογισμό ενός μέτρου απόδοσης, όπως η ακρίβεια).

Το μέτρο απόδοσης που αναφέρεται από τη διασταυρούμενη επικύρωση k-πτυχών είναι στη συνέχεια ο μέσος όρος των τιμών που υπολογίζονται στον βρόχο. Αυτή η προσέγγιση μπορεί να είναι υπολογιστικά δαπανηρή, αλλά δεν σπαταλάει πάρα πολλά δεδομένα (όπως συμβαίνει όταν καθορίζεται ένα αυθαίρετο σύνολο επικύρωσης), γεγονός που αποτελεί σημαντικό πλεονέκτημα σε προβλήματα όπως η αντίστροφη συμπερασματολογία, όπου ο αριθμός των δειγμάτων είναι πολύ μικρός.

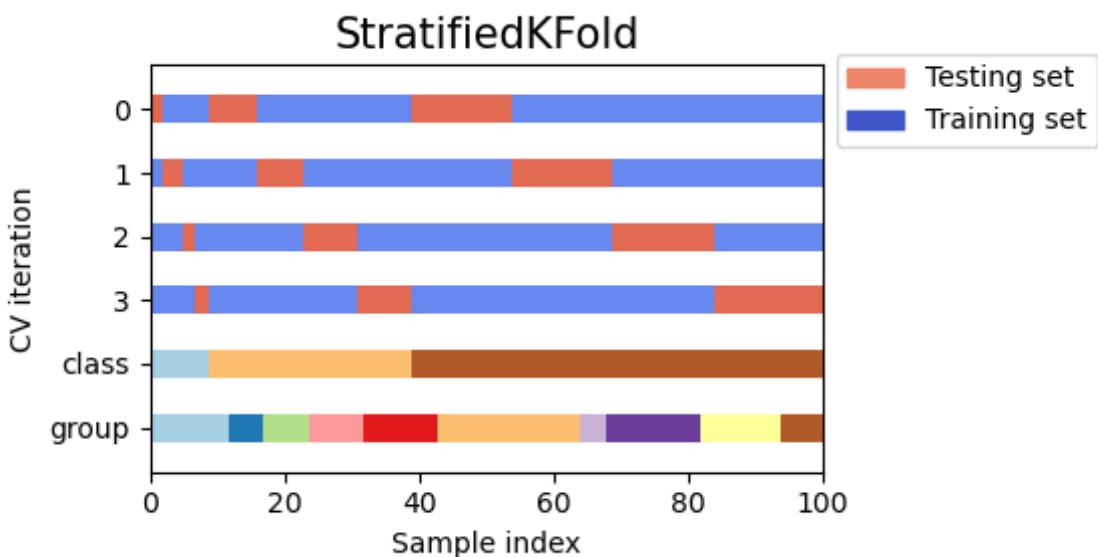


Σχήμα 4.2: Διασταυρούμενη Επικύρωση [71]

### Stratified k-fold

Το Stratified K-fold είναι μια παραλλαγή του k-fold που επιστρέφει στρωματοποιημένες αναδιπλώσεις, δηλαδή κάθε σύνολο περιέχει περίπου το ίδιο ποσοστό δειγμάτων κάθε κλάσης-στόχου με το πλήρες σύνολο. Αυτή είναι και η μέθοδος που θα χρησιμοποιηθεί στην υλοποίηση των αλγορίθμων.

Για παράδειγμα, το σχήμα 4.3 απεικονίζει 100 τυχαία παραγόμενα σημεία εισόδου, 3 άνισες κλάσεις και 10 ομάδες (δείγματα που συλλέγονται από διαφορετικά αντικείμενα μελέτης, πειράματα, συσκευές μέτρησης). Πραγματοποιούνται 4 διαχωρισμοί των δεδομένων όπου σε κάθε διάσπαση, οπτικοποιούνται οι δείκτες που επιλέγονται για το σύνολο εκπαίδευσης (με μπλε χρώμα) και το σύνολο ελέγχου (με κόκκινο χρώμα). Ο στόχος, όπως αναφέρθηκε, είναι να διατηρηθεί το ποσοστό των δειγμάτων για κάθε κλάση.



Σχήμα 4.3: Στρωματοποιημένες k αναδιπλώσεις [71]

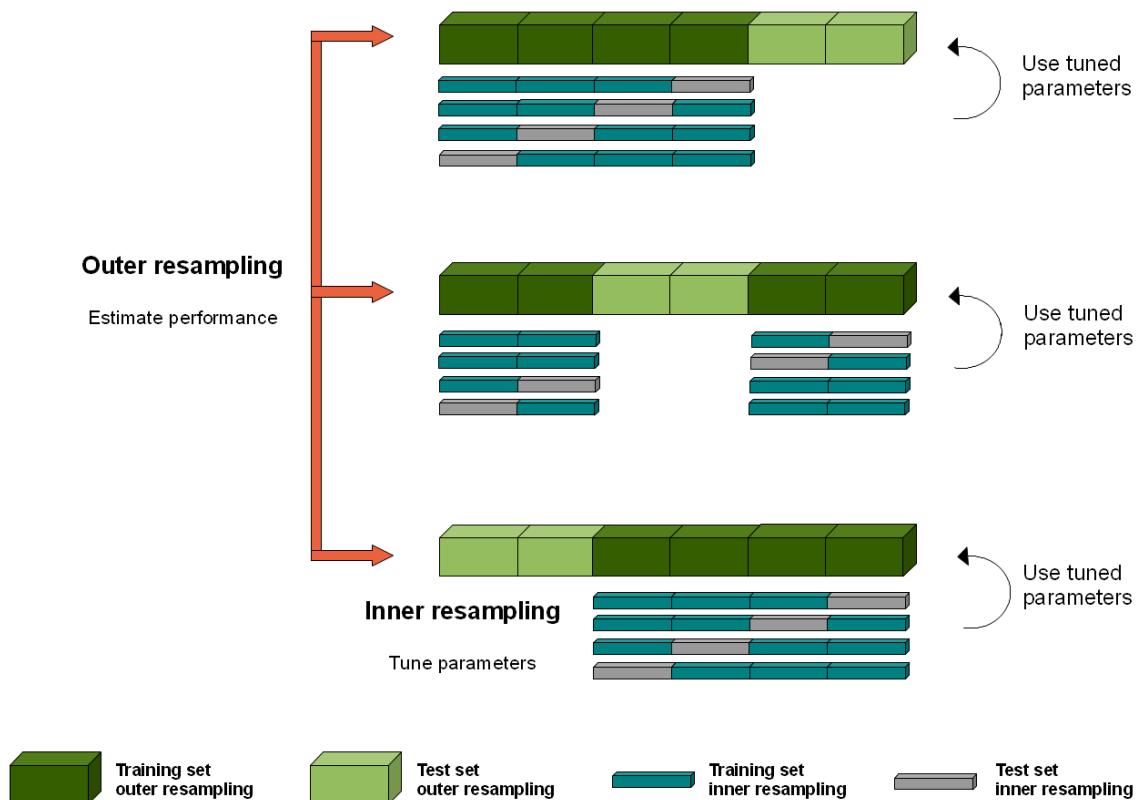
### 4.4.2 Εμφωλευμένη Διασταυρούμενη Επικύρωση

Η Εμφωλευμένη Διασταυρούμενη Επικύρωση χρησιμοποιείται συχνά για την εκπαίδευση ενός μοντέλου στο οποίο πρέπει να βελτιστοποιηθούν και οι υπερπαράμετροι [72] [73] [74] [75]. Η φωλιασμένη CV εκτιμά το σφάλμα γενίκευσης του υποκείμενου μοντέλου και την αναζήτηση των (υπερ)παραμέτρων του. Η επιλογή των παραμέτρων που μεγιστοποιούν το μη ένθετο CV στρεβλώνει το μοντέλο προς το σύνολο δεδομένων, αποδίδοντας μια υπερβολικά αισιόδοξη βαθμολογία.

Η επιλογή μοντέλου χωρίς εμφωλευμένο CV χρησιμοποιεί τα ίδια δεδομένα για τη ρύθμιση των παραμέτρων του μοντέλου και την αξιολόγηση της απόδοσης του μοντέλου. Οι πληροφορίες μπορεί έτσι να "διαρρεύσουν" στο μοντέλο και να προσαρμόσουν υπερβολικά τα δεδομένα. Το μέγεθος αυτού του αποτελέσματος εξαρτάται κυρίως από το μέγεθος του συνόλου δεδομένων και τη σταθερότητα του μοντέλου.

## ΚΕΦΑΛΑΙΟ 4. ΕΡΓΑΛΕΙΑ ΚΑΙ ΤΕΧΝΙΚΕΣ

Για την αποφυγή αυτού του προβλήματος, το εμφωλευμένο CV χρησιμοποιεί ουσιαστικά μια σειρά από διαχωρισμούς συνόλων εκπαίδευσης/επικύρωσης/δοκιμής. Στον εσωτερικό βρόχο, η βαθμολογία μεγιστοποιείται κατά προσέγγιση με την προσαρμογή ενός μοντέλου σε κάθε σύνολο εκπαίδευσης και στη συνέχεια μεγιστοποιείται άμεσα κατά την επιλογή (υπερ)παραμέτρων στο σύνολο επικύρωσης. Στον εξωτερικό βρόχο, το σφάλμα γενίκευσης εκτιμάται με τον μέσο όρο των βαθμολογιών του συνόλου δοκιμής σε διάφορα διαχωρίσματα συνόλων δεδομένων.



Σχήμα 4.4: Εμφωλευμένη Διασταυρούμενη Επικύρωση [76]

Πρέπει να σημειωθεί ότι αυτή η τεχνική είναι υπολογιστικά δαπανηρή επειδή εκπαιδεύονται και αξιολογούνται πολλά μοντέλα. Δυστυχώς, δεν υπάρχει ενσωματωμένη μέθοδος στο sklearn που να εκτελεί Nested k-Fold CV, οπότε φτιάχτηκε μια τέτοια μέθοδος για τις ανάγκες της εργασίας αυτής.

Τέλος, στις μεθόδους επιλογής χαρακτηριστικών και στους αλγορίθμους μηχανικής μάθησης θα χρησιμοποιηθεί διασταυρούμενη επικύρωση 10 επαναλήψεων, ενώ στον εσωτερικό βρόγχο της εμφωλευμένης διασταυρούμενης επικύρωσης θα χρησιμοποιηθούν 5 επαναλήψεις.

## 4.5 ΜΕΘΟΔΟΙ ΕΠΙΛΟΓΗΣ ΥΠΕΡ-ΠΑΡΑΜΕΤΡΩΝ

Σε αντίθεση με τις παραμέτρους του μοντέλου που μαθαίνονται κατά τη διάρκεια της εκπαίδευσης, οι υπερπαράμετροι του μοντέλου καθορίζονται από τον επιστήμονα δεδομένων πριν από την εκπαίδευση και ελέγχουν τις πτυχές εφαρμογής του μοντέλου. Οι υπερπαράμετροι μπορούν να θεωρηθούν ως ρυθμίσεις του μοντέλου. Αυτές οι ρυθμίσεις πρέπει να συντονίζονται, επειδή οι ιδανικές ρυθμίσεις για ένα σύνολο δεδομένων δεν θα είναι ίδιες σε όλα τα σύνολα δεδομένων. Κατά τον συντονισμό των υπερπαραμέτρων ενός εκτιμητή, η αναζήτηση πλέγματος και η τυχαία αναζήτηση είναι οι δύο πιο δημοφιλείς μέθοδοι.

### 4.5.1 Αναζήτηση Πλέγματος

Η **Αναζήτηση Πλέγματος** μπορεί να θεωρηθεί ως μια εξαντλητική αναζήτηση για την επιλογή ενός μοντέλου. Στην αναζήτηση πλέγματος, ο επιστήμονας δεδομένων δημιουργεί ένα πλέγμα τιμών υπερπαραμέτρων και για κάθε συνδυασμό, εκπαιδεύει ένα μοντέλο και βαθμολογεί τα δεδομένα δοκιμής. Σε αυτή την προσέγγιση, δοκιμάζεται κάθε συνδυασμός τιμών υπερπαραμέτρων, γεγονός που μπορεί να είναι πολύ αναποτελεσματικό. Για παράδειγμα, η αναζήτηση 20 διαφορετικών τιμών παραμέτρων για κάθε μία από τις 4 παραμέτρους θα απαιτήσει 160.000 δοκιμές διασταυρούμενης επικύρωσης. Αυτό ισοδυναμεί με 1.600.000 προσαρμογές μοντέλων και 1.600.000 προβλέψεις εάν χρησιμοποιείται 10πλή διασταυρούμενη επικύρωση. Ενώ το Scikit Learn προσφέρει τη λειτουργία GridSearchCV για την απλοποίηση της διαδικασίας, εξακολουθεί να είναι μια εξαιρετικά δαπανηρή διαδικασία τόσο σε υπολογιστική ισχύ όσο και σε χρόνο.

### 4.5.2 Τυχαιοποιημένη Αναζήτηση

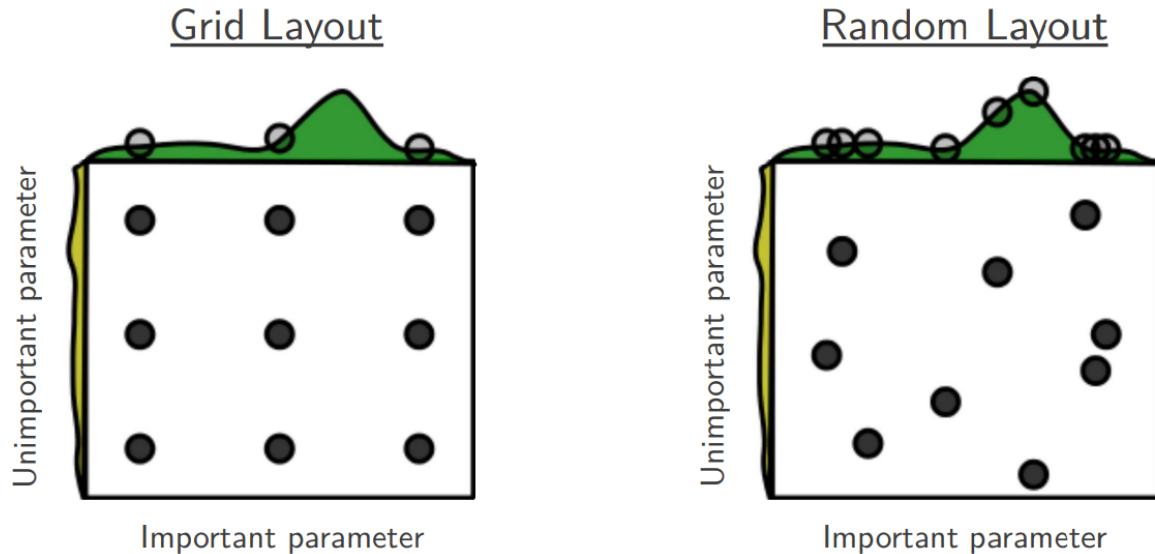
Αντίθετα, η **Τυχαία Αναζήτηση** δημιουργεί ένα πλέγμα τιμών υπερπαραμέτρων και επιλέγει τυχαίους συνδυασμούς για να εκπαιδεύσει το μοντέλο και να το βαθμολογήσει. Αυτό επιτρέπει ρητά τον έλεγχο του αριθμού των συνδυασμών των παραμέτρων που επιχειρούνται. Ο αριθμός των επαναλήψεων αναζήτησης ορίζεται με βάση το χρόνο ή τους πόρους που έχει κανείς διαθέσιμα. Το Scikit Learn προσφέρει τη συνάρτηση RandomizedSearchCV για αυτή τη διαδικασία.

### 4.5.3 Σύγκριση

Ενώ είναι πιθανό το RandomizedSearchCV να μην βρίσκει τόσο ακριβή αποτελέσματα όσο το GridSearchCV, παραδόξως επιλέγει το καλύτερο αποτέλεσμα τις περισσότερες φορές και σε ένα μέρος του χρόνου που θα χρειαζόταν το GridSearchCV. Με τους ίδιους πόρους, η τυχαία αναζήτηση μπορεί ακόμη και να ξεπεράσει την αναζήτηση πλέγματος.

Στο σχήμα 4.5 απεικονίζεται η αναζήτηση πλέγματος και η τυχαία αναζήτηση εννέα δοκιμών για τη βελτιστοποίηση μιας συνάρτησης  $f(x, y) = g(x) + h(y) \approx g(x)$  με χαμηλή πραγματική διαστατικότητα. Πάνω από κάθε τετράγωνο το  $g(x)$  εμφανίζεται με πράσινο χρώμα και αριστερά από κάθε τετράγωνο το  $h(y)$  εμφανίζεται

με κίτρινο χρώμα. Με την αναζήτηση πλέγματος, εννέα δοκιμές δοκιμάζουν μόνο το  $g(x)$  σε τρία διαφορετικά σημεία. Με τυχαία αναζήτηση, και οι εννέα δοκιμές διερευνούν διαφορετικές τιμές του  $g$ . Αυτή η αποτυχία της αναζήτησης πλέγματος είναι ο κανόνας και όχι η εξαίρεση στη βελτιστοποίηση υπερπαραμέτρων υψηλής διάστασης [77].



Σχήμα 4.5: Διάταξη Πλέγματος και Τυχαία Διάταξη [77]

#### 4.5.4 Συμπέρασμα

Συμπερασματικά, με μικρά σύνολα δεδομένων και πολλούς πόρους, η Αναζήτηση Πλέγματος θα παράγει ακριβή αποτελέσματα. Ωστόσο, με μεγάλα σύνολα δεδομένων, οι υψηλές διαστάσεις θα επιβραδύνουν σημαντικά το χρόνο υπολογισμού και θα είναι πολύ δαπανηρές. Σε αυτή την περίπτωση, συνιστάται η χρήση Τυχαιοποιημένης Αναζήτησης, καθώς ο αριθμός των επαναλήψεων ορίζεται ρητά από τον επιστήμονα δεδομένων.

Σε αυτήν την εργασία, με το συγκεκριμένο σύνολο δεδομένων το οποίο θεωρείται μικρό, θα χρησιμοποιηθούν και οι δύο αυτές μέθοδοι. Αυτό θα γίνει διότι με τη χρήση της διασταυρούμενης επικύρωσης θα αυξηθεί περαιτέρω ο χρόνος εκπαίδευσης και ελέγχου, αλλά και γιατί κάποιοι αλγόριθμοι μηχανικής μάθησης μπορούν να δεχτούν πολύ μεγάλο αριθμό παραμέτρων, για τις οποίες είναι αρκετά δύσκολο να γίνει μια αρχική πρόβλεψη για το ποιες είναι χρήσιμες και ποιες όχι έτσι ώστε να μειωθούν.

## 4.6 ΑΝΑΛΥΣΗ ΚΥΡΙΩΝ ΣΥΝΙΣΤΩΣΩΝ ΜΕ ΑΠΟΣΥΝΘΕΣΗ ΙΔΙΑΖΟΥΣΩΝ ΤΙΜΩΝ

---

### 4.6.1 Εισαγωγή

Η Ανάλυση Κύριων Συνιστωσών, ή PCA, είναι μια γνωστή και ευρέως χρησιμοποιούμενη τεχνική που εφαρμόζεται σε μια ευρεία ποικιλία εφαρμογών, όπως η μείωση διαστάσεων, η συμπίεση δεδομένων, η εξαγωγή χαρακτηριστικών και η οπτικοποίηση. Η βασική ιδέα είναι η προβολή ενός συνόλου δεδομένων από πολλές συσχετισμένες συντεταγμένες σε λιγότερες ασυσχέτιστες συντεταγμένες που ονομάζονται κύριες συνιστώσες, διατηρώντας παράλληλα το μεγαλύτερο μέρος της μεταβλητότητας που υπάρχει στα δεδομένα [78].

Η Αποσύνθεση Ιδιαζουσών Τιμών, ή SVD, είναι μια υπολογιστική μέθοδος που χρησιμοποιείται συχνά για τον υπολογισμό των κύριων συνιστωσών για ένα σύνολο δεδομένων. Η χρήση της SVD για την εκτέλεση της PCA είναι αποτελεσματική και αριθμητικά ισχυρή.

### 4.6.2 Δεδομένα σε πίνακα

Πριν περιγραφεί η εκτέλεση της PCA, θα είναι χρήσιμο να τυποποιηθεί ο τρόπος με τον οποίο τα δεδομένα κωδικοποιούνται σε μορφή πίνακα. Δεδομένου ότι η PCA και η SVD είναι γραμμικές τεχνικές, αυτό επιτρέπει τον χειρισμό των δεδομένων χρησιμοποιώντας γραμμικούς μετασχηματισμούς πιο ένυκλα.

Είναι απλούστερο να εξηγηθεί με ένα παράδειγμα, ας υποτεθεί λοιπόν ότι δίνεται το πλάτος  $w$ , το ύψος  $h$  και το μήκος  $l$  από  $n = 1000$  μετρήσεις ορθογώνιων κουτιών. Οι μετρήσεις του κουτιού  $i$  μπορούν να κωδικοποιηθούν ως μια πλειάδα  $(w_i, h_i, l_i)$  που ονομάζεται δείγμα. Κάθε δείγμα είναι ένα διάνυσμα σε  $d = 3$  διαστάσεις, αφού υπάρχουν 3 αριθμοί που το περιγράφουν. Επειδή τα διανύσματα συνήθως γράφονται οριζόντια, τα διανύσματα μεταθέτονται ώστε να γραφτούν κάθετα:

$$x_i = \begin{pmatrix} w_i \\ h_i \\ l_i \end{pmatrix} \quad \text{και} \quad x_i^T = (w_i, h_i, l_i).$$

Για να συσκευαστούν τα δεδομένα σε ένα ενιαίο αντικείμενο, τα δείγματα απλά στοιβάζονται ως γραμμές ενός πίνακα δεδομένων,

$$X = \begin{pmatrix} w_1 & h_1 & l_1 \\ \hline w_2 & h_2 & l_2 \\ \hline \vdots & & \\ \hline w_{1000} & h_{1000} & l_{1000} \end{pmatrix}.$$

Οι στήλες αυτού του πίνακα αντιστοιχούν σε μία μόνο συντεταγμένη, δηλαδή σε όλες τις μετρήσεις ενός συγκεκριμένου τύπου.

Στη γενική περίπτωση, εργαζόμαστε με ένα  $d$ -διάστατο σύνολο δεδομένων που αποτελείται από  $n$  δείγματα. Αντί να χρησιμοποιηθούν γράμματα όπως  $h, w$  και  $l$

για να δηλωθούν οι διάφορες συντεταγμένες, απλά απαριθμούνται οι καταχωρήσεις κάθε διανύσματος δεδομένων έτσι ώστε  $x_i^T = (x_{i1}, \dots, x_{id})$ . Πριν τοποθετηθούν αυτά τα διανύσματα σε έναν πίνακα δεδομένων, θα βρεθεί ο μέσος όρος των δεδομένων

$$\mu = \frac{1}{n} \sum_{i=1}^n x_i = \left( \frac{1}{n} \sum_{i=1}^n x_{i1}, \dots, \frac{1}{n} \sum_{i=1}^n x_{id} \right)^T$$

από κάθε δείγμα για μετέπειτα ευκολία. Στη συνέχεια χρησιμοποιούνται αυτά τα μηδενικά διανύσματα ως γραμμές του πίνακα

$$X = \begin{pmatrix} \frac{x_1^T - \mu^T}{x_1^T - \mu^T} \\ \frac{x_2^T - \mu^T}{x_2^T - \mu^T} \\ \vdots \\ \frac{x_n^T - \mu^T}{x_n^T - \mu^T} \end{pmatrix}.$$

Η τοποθέτηση των δεδομένων σε έναν πίνακα είναι ιδιαίτερα βολική επειδή επιτρέπει να γραφτεί η δειγματική συνδιακύμανση γύρω από το μέσο όρο χρησιμοποιώντας πίνακες, όπως

$$S = \frac{1}{n-1} \sum_{i=1}^n (x_i - \mu)(x_i - \mu)^T = \frac{1}{n-1} X^T X.$$

Διαιρώντας με  $n-1$  είναι ένας τυπικός τρόπος διόρθωσης της μεροληψίας που εισάγεται από τη χρήση του μέσου όρου του δείγματος αντί του πραγματικού μέσου όρου του πληθυσμού. Ο πίνακας συνδιακύμανσης θα βρεθεί στο επίκεντρο της προσοχής καθώς θα αναλύονται οι έννοιες των κύριων συνιστωσών και της αποσύνθεσης μοναδιαίων τιμών.

### Πίνακας Συνδιακύμανσης

Η  $j$ -οστή στήλη του  $X$  δεν είναι τίποτα άλλο παρά η  $j$ -οστή συντεταγμένη στην οποία είναι κωδικοποιημένο το μηδενο-κεντρικό σύνολο δεδομένων. Η  $jk$ -οστή εγγραφή του γινομένου  $\frac{1}{n-1} X^T X$  δίνεται επομένως ως το (κλιμακωτό) τετραγωνικό γινόμενο της  $j$ -οστής στήλης του  $X$  που συμβολίζεται με  $x_{\bullet,j}$  και της  $k$ -οστής στήλης που συμβολίζεται με  $x_{\bullet,k}$ . Αυτό είναι

$$\frac{1}{n-1} x_{\bullet,j} \cdot x_{\bullet,k} = \frac{1}{n-1} x_{\bullet,j}^T x_{\bullet,k} = \frac{1}{n-1} \sum_{i=1}^n x_{ij} x_{ik}.$$

Όταν  $k = j$  αυτό δίνει τη διακύμανση των δεδομένων κατά μήκος του  $k$ -οστού ήτοντα συντεταγμένων, διαφορετικά λαμβάνεται ένα μέτρο του πόσο μεταβάλλονται οι δύο συντεταγμένες μαζί.

### 4.6.3 Ανάλυση Κύριων Συνιστωσών

Μια από τις πρώτες εργασίες που εισήγαγε την PCA όπως είναι γνωστή σήμερα δημοσιεύτηκε το 1933 από τον Hotelling. Το κίνητρο του συγγραφέα ήταν να

#### 4.6. ΑΝΑΛΥΣΗ ΚΥΡΙΩΝ ΣΥΝΙΣΤΩΣΩΝ ΜΕ ΑΠΟΣΥΝΘΕΣΗ ΙΔΙΑΖΟΥΣΩΝ ΤΙΜΩΝ

---

μετατρέψει ένα σύνολο πιθανώς συσχετιζόμενων μεταβλητών σε "κάποιο πιο θεμελιώδες σύνολο ανεξάρτητων μεταβλητών που καθορίζουν τις τιμές που θα λάβουν [οι αρχικές μεταβλητές]". Προερχόμενος από την φυχολογία, η πρώτη του επιλογή για την ονομασία τους ήταν "παράγοντες", αλλά δεδομένου ότι ο όρος αυτός είχε ήδη μια σημασία στα μαθηματικά, ονόμασε το μειωμένο σύνολο μεταβλητών "συνιστώσες" και την τεχνική για την εύρεσή τους "μέθοδο των κύριων συνιστώσων". Οι συνιστώσες αυτές επιλέγονται διαδοχικά με τρόπο που να επιτρέπει στις "συνεισφορές τους στις αποκλίσεις [των αρχικών μεταβλητών] να έχουν όσο το δυνατόν μεγαλύτερο σύνολο".

Μαθηματικά, ο στόχος της Ανάλυσης Κύριων Συνιστώσων, ή PCA, είναι να βρεθεί μια συλλογή από  $k \leq d$  μοναδιαία διανύσματα  $v_i \in \mathbb{R}^d$  (για  $i \in 1, \dots, k$ ) που ονομάζονται κύριες συνιστώσες (Principal Components ή PCs), έτσι ώστε

1. η διακύμανση του συνόλου δεδομένων που προβάλλεται στην κατεύθυνση που καθορίζεται από το  $v_i$  να μεγιστοποιείται και
2. το  $v_i$  επιλέγεται να είναι κάθετο ως προς τα  $v_1, \dots, v_{i-1}$ .

Τώρα, η προβολή ενός διανύσματος  $x \in \mathbb{R}^d$  στη γραμμή που καθορίζεται από οποιοδήποτε  $v_i$  δίνεται απλά ως το γινόμενο τελείας  $v_i^T x$ . Αυτό σημαίνει ότι η διακύμανση του συνόλου δεδομένων που προβάλλεται στην πρώτη κύρια συνιστώσα  $v_1$  μπορεί να γραφεί ως εξής

$$\frac{1}{n-1} \sum_{i=1}^n (v_1^T x_i - v_1^T \mu)^2 = v_1^T S v_1.$$

Για να βρεθεί το  $v_1$  πρέπει να μεγιστοποιηθεί η παραπάνω ποσότητα, με τον πρόσθετο περιορισμό ότι  $\|v_1\| = 1$ . Στην επίλυση αυτού του προβλήματος βελτιστοποίησης με τη μέθοδο των πολλαπλασιαστών Lagrange, συνεπάγεται ότι

$$S v_1 = \lambda_1 v_1,$$

που σημαίνει ότι το  $v_1$  είναι ιδιοδιάνυσμα του πίνακα συνδιακύμανσης  $S$ . Στην πραγματικότητα, δεδομένου ότι  $\|v_1\| = v_1^T v_1 = 1$  βγαίνει το συμπέρασμα ότι η αντίστοιχη ιδιοτιμή είναι ακριβώς ίση με τη διακύμανση του συνόλου δεδομένων κατά μήκος του  $v_1$ , δηλαδή

$$v_1^T S v_1 = \lambda_1.$$

Αυτή η διαδικασία μπορεί να συνεχιστεί προβάλλοντας τα δεδομένα σε μια νέα κατεύθυνση  $v_2$  επιβάλλοντας παράλληλα τον πρόσθετο περιορισμό ότι  $v_1 \perp v_2$  και γενικά σε  $v_n$  ενώ επιβάλλεται το  $v_n \perp v_1, \dots, v_{n-1}$ .

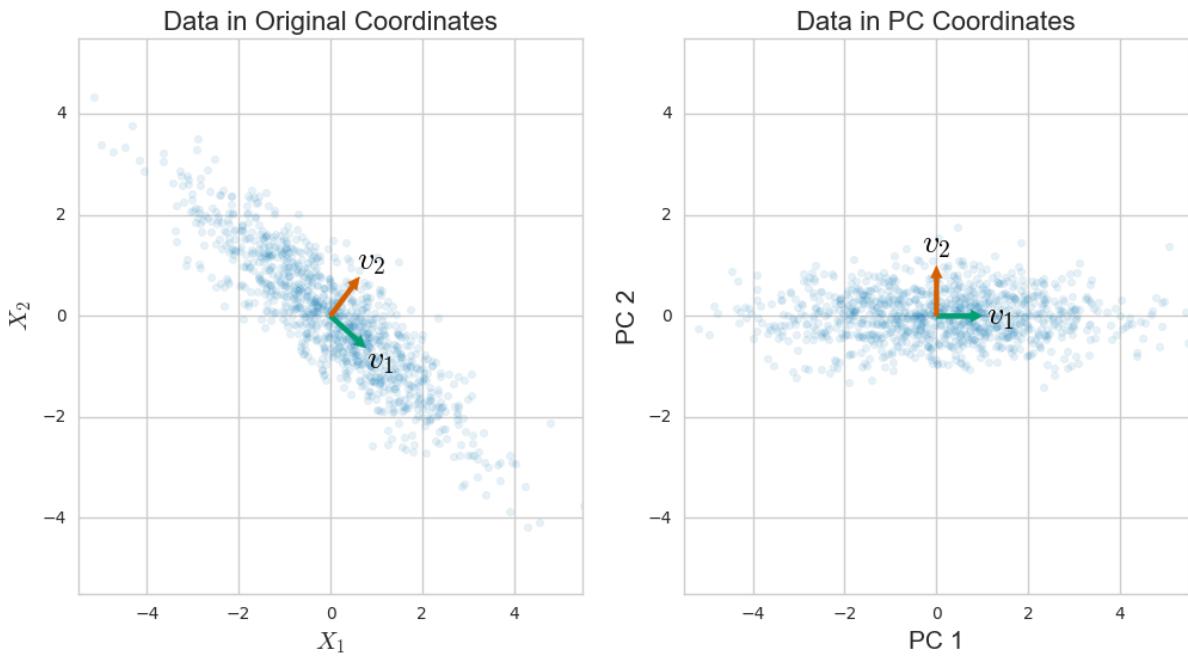
Το τελικό αποτέλεσμα είναι ότι οι πρώτες  $k$  κύριες συνιστώσες του  $X$  αντιστοιχούν ακριβώς στα ιδιοδιανύσματα του πίνακα συνδιακύμανσης  $S$  ταξινομημένες με βάση τις ιδιοτιμές τους. Επιπλέον, οι ιδιοτιμές είναι ακριβώς ίσες με τη διακύμανση του συνόλου δεδομένων κατά μήκος των αντίστοιχων ιδιοδιανυσμάτων.

Σύμφωνα με τα παραπάνω, υπάρχει ένα πλήρες σύνολο ορθοκανονικών ιδιοδιανυσμάτων για το  $S$  στο  $\mathbb{R}$ . Το  $S$  είναι ένας πραγματικός συμμετρικός πίνακας,

που σημαίνει ότι  $S = S^T$ , από το Πραγματικό Φασματικό Θεώρημα συνεπάγεται ακριβώς αυτό. Αυτό είναι ένα μη τετριμένο αποτέλεσμα το οποίο θα χρησιμοποιηθεί αργότερα, οπότε ας αναλυθεί λίγο σε αυτό το σημείο. Θεωρείται η περίπτωση  $k = d < n$ . Παίρνοντας  $k = d$  κύριες συνιστώσες μπορεί να φαίνεται σαν μια παράξενη επιλογή αν ο σκοπός μας είναι να κατανοηθεί το  $X$  μέσω ενός υποδιαστήματος χαμηλότερης διάστασης, αλλά αυτό επιτρέπει την κατασκευή ενός  $d \times d$  πίνακα  $V$  του οποίου οι στήλες είναι τα ιδιοδιανύσματα του  $S$  και το οποίο επομένως διαγωνιοποιεί τον  $S$ , δηλαδή

$$S = V\Lambda V^T = \sum_{i=1}^r \lambda_i v_i v_i^T,$$

όπου  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_d)$  και  $r = \text{rank}(X)$ . Με άλλα λόγια, οι κύριες συνιστώσες είναι οι στήλες ενός πίνακα περιστροφής και αποτελούν τους άξονες μιας νέας βάσης η οποία μπορεί να θεωρηθεί ότι "ευθυγραμμίζεται" με το σύνολο δεδομένων  $X$ .



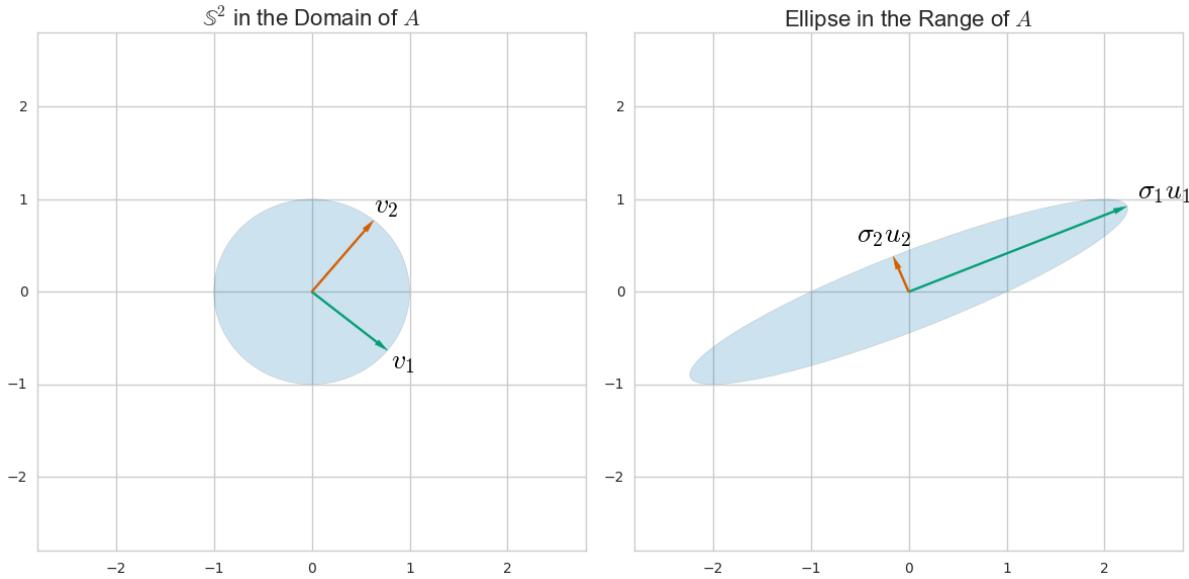
Σχήμα 4.6: Η αρχική και ασυσχέτιστη προβολή 1000 δειγμάτων που προέρχονται από μια πολυμεταβλητή Γκαουσιανή [78]

#### 4.6.4 Αποσύνθεση Ιδιαζουσών Τιμών

Η Αποσύνθεση Ιδιαζουσών Τιμών είναι μια μέθοδος παραγοντοποίησης πινάκων που χρησιμοποιείται σε πολλές αριθμητικές εφαρμογές της γραμμικής άλγεβρας, όπως η PCA. Αυτή η τεχνική βελτιώνει την κατανόηση του τι είναι οι κύριες συνιστώσες και παρέχει ένα ισχυρό υπολογιστικό πλαίσιο που επιτρέπει τον υπολογισμό τους με ακρίβεια για περισσότερα σύνολα δεδομένων.

#### 4.6. ΑΝΑΛΥΣΗ ΚΥΡΙΩΝ ΣΥΝΙΣΤΩΣΩΝ ΜΕ ΑΠΟΣΥΝΘΕΣΗ ΙΔΙΑΖΟΥΣΩΝ ΤΙΜΩΝ

Ας υποτεθεί ένας αυθαίρετος πίνακας  $A$ , διαστάσεων  $n \times d$ . Η SVD έχει ως κύνητρο το γεγονός ότι όταν θεωρείται ως γραμμικός μετασχηματισμός, ο  $A$  χαρτογραφεί τη μοναδιαία σφαίρα  $\mathbb{S}^d \subset \mathbb{R}^d$  σε μια (υπερ)έλλειψη στο  $\mathbb{R}^n$ . Θα εξεταστεί ένα παράδειγμα με  $n = d = 2$  για να γίνει πιο εύκολα αντιληπτό αυτό το γεγονός.



Σχήμα 4.7: Χαρτογράφηση μοναδιαίας σφαίρας σε (υπερ)έλλειψη

$$\text{Δράση του } A = \begin{pmatrix} 1 & 2 \\ 0 & 1 \end{pmatrix} \text{ στη μοναδιαία σφαίρα στο } \mathbb{R}^2 \text{ [78]}$$

Από αυτό το σχήμα 4.7 μπορούν να εξαχθούν μερικοί χρήσιμοι ορισμοί που ισχύουν για αυθαίρετες διαστάσεις με μερικούς επιπλέον περιορισμούς:

- Τα μήκη  $\sigma_i$  των ημιαξόνων της έλλειψης  $A\mathbb{S}^d \in \mathbb{R}^n$  είναι οι μοναδικές τιμές του  $A$ .
- Τα ιδιάζοντα διανύσματα  $u_i$  κατά μήκος των ημιαξόνων της έλλειψης ονομάζονται τα "αριστερά" ιδιάζοντα διανύσματα του  $A$ .
- Τα ιδιάζοντα διανύσματα  $u_i$  έτσι ώστε  $Au_i = \sigma_i u_i$ , ονομάζονται τα "δεξιά" ιδιάζοντα διανύσματα του  $A$ .

Αυτό που μπορεί να μην είναι άμεσα εμφανές από την εικόνα είναι ότι ανάλογα με τα  $n, d$  και  $r = \text{rank}(A)$ , ορισμένα από τα αριστερά ιδιάζοντα διανύσματα μπορεί να "καταρρεύσουν" στο μηδέν. Αυτό συμβαίνει όταν ο πίνακας  $A$  δεν έχει πλήρη τάξη, για παράδειγμα, αφού τότε το εύρος του πρέπει να είναι ένας υποχώρος του  $\mathbb{R}^n$  με διάσταση  $r < d$ . Γενικά, υπάρχουν ακριβώς  $r = \text{rank}(A)$  ιδιάζουσες τιμές και ο ίδιος αριθμός αριστερών ιδιαζουσών διανυσμάτων.

Στοιβάζοντας τα διανύσματα  $v_i$  και  $u_i$  σε στήλες πινάκων  $\widehat{V}$  και  $\widehat{U}$ , αντίστοιχα, οι σχέσεις  $Au_i = \sigma_i u_i$ , μπορούν να γραφούν ως

$$A\widehat{V} = \widehat{U}\widehat{\Sigma},$$

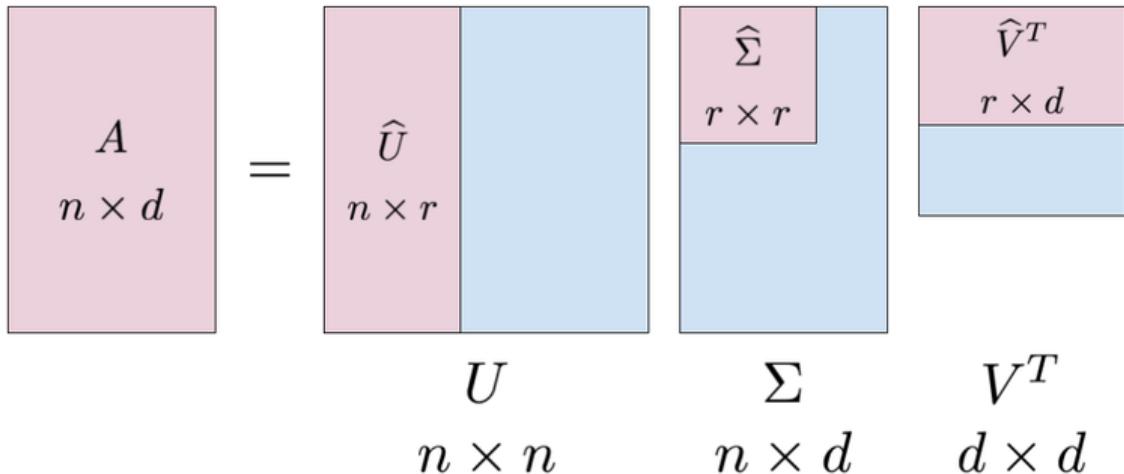
όπου  $\widehat{\Sigma} = \text{diag}(\sigma_1, \dots, \sigma_r)$ . Γεμίζοντας το  $\widehat{\Sigma}$  με μηδενικά και προσθέτοντας αυθαίρετες ορθοκανονικές στήλες στο  $\widehat{V}$  και στο  $\widehat{U}$ , λαμβάνεται μια πιο βολική παραγοντοποίηση

$$AV = U\Sigma.$$

Από τη στιγμή που το  $V$  είναι μοναδιαίο, δηλαδή έχει μοναδιαίου μήκους ορθοκανονικές στήλες, προκύπτει ότι  $V^{-1} = V^T$ , οπότε πολλαπλασιάζοντας με  $V^T$  δίνει την ανάλυση ιδιάζουσας τιμής του  $A$ :

$$A = U\Sigma V^T.$$

Μια σύνοψη αυτού του αποτελέσματος φαίνεται στο σχήμα 4.8.



Σχήμα 4.8: Συνιστώσες της αποσύνθεσης της ιδιάζουσας τιμής ενός πίνακα  $A$  [78]

Οι εσωτερικές περιοχές αντιστοιχούν στα στοιβαγμένα διανύσματα και τις μοναδικές τιμές που παρακινούνται από το σχήμα 4.7

#### 4.6.5 Σχέση μεταξύ SVD και PCA

Δεδομένου ότι κάθε πίνακας έχει μια αποσύνθεση μοναδιαίων τιμών, με  $A = X$ , η αποσύνθεση γράφεται

$$X = U\Sigma V^T.$$

Μέχρι στιγμής ο πίνακας  $A$  έχει εκφραστεί ως γραμμικός μετασχηματισμός, αλλά τίποτα δεν εμποδίζει την χρήση της SVD σε έναν πίνακα δεδομένων. Από την αποσύνθεση βγαίνει ότι

$$X^T X = (U\Sigma V^T)^T (U\Sigma V^T) = V\Sigma^T U^T U\Sigma V^T = V(\Sigma^T \Sigma)V^T,$$

που σημαίνει ότι τα  $X^T X$  και  $\Sigma^T \Sigma$  είναι παρόμοια. Παρόμοιοι πίνακες έχουν τις ίδιες ιδιοτιμές, οπότε οι ιδιοτιμές  $\lambda_i$  του πίνακα συνδιακύμανσης  $S = \frac{1}{n-1} X^T X$  σχετίζονται με τις ιδιάζουσες τιμές  $\sigma_i$  του πίνακα  $X$  μέσω της

#### 4.6. ΑΝΑΛΥΣΗ ΚΥΡΙΩΝ ΣΥΝΙΣΤΩΣΩΝ ΜΕ ΑΠΟΣΥΝΘΕΣΗ ΙΔΙΑΖΟΥΣΩΝ ΤΙΜΩΝ

---

$$\sigma_i^2 = (n - 1)\lambda_i,$$

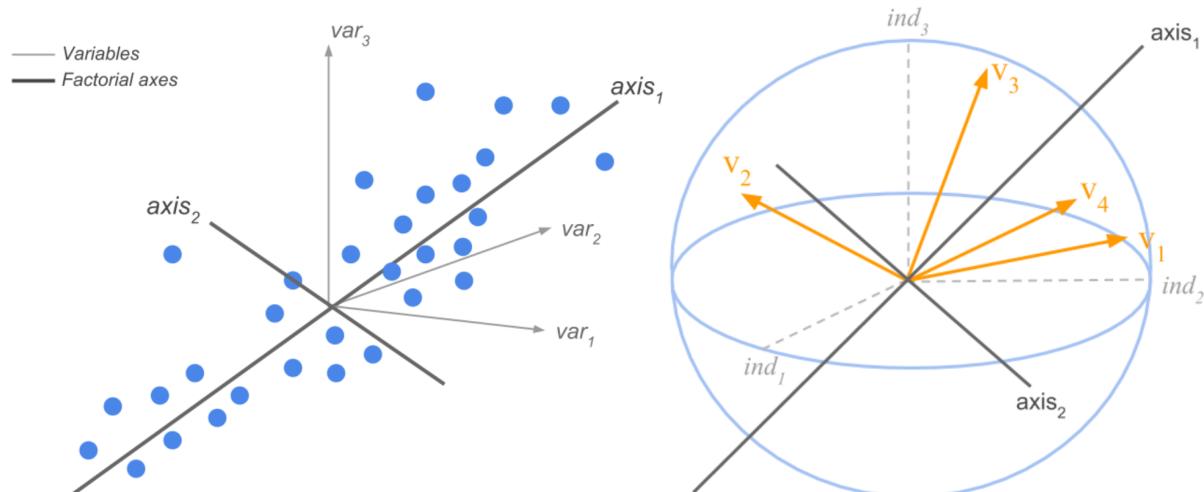
για  $i \in \{1, \dots, r\}$ , όπου ως συνήθως  $r = \text{rank}(A)$ .

Για να συσχετιστεί πλήρως η SVD και η PCA πρέπει επίσης να περιγραφεί η αντιστοιχία μεταξύ των κύριων συνιστώσων και των ιδιαζουσών διανυσμάτων. Για τα δεξιά ιδιάζοντα διανύσματα ισχύει ότι

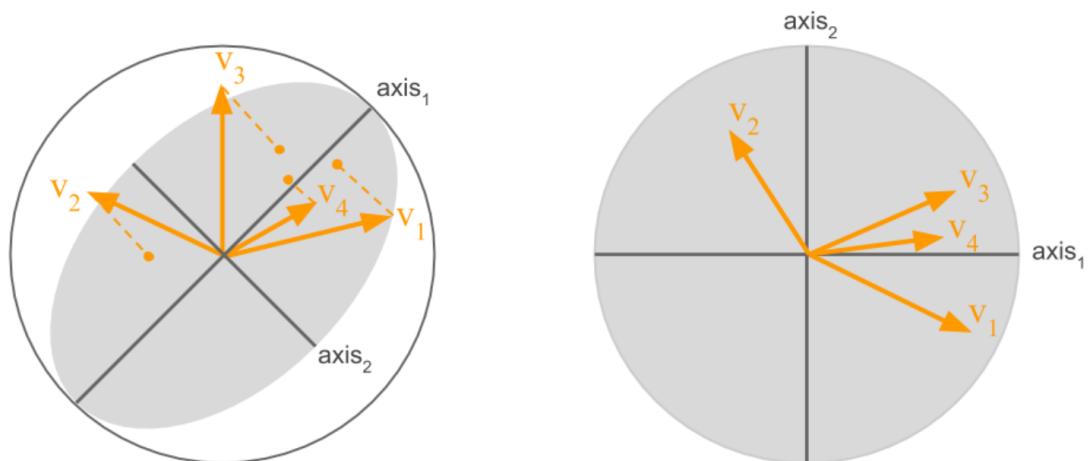
$$\widehat{V}^T = \begin{pmatrix} v_1^T \\ \vdots \\ v_r^T \end{pmatrix}$$

όπου  $v_i$  είναι οι κύριες συνιστώσες του  $X$ . Για τα αριστερά ιδιάζοντα διανύσματα ισχύει

$$u_i = \frac{1}{\sqrt{(n - 1)\lambda_i}} X v_i.$$



Σχήμα 4.9: Αλλαγή βάσης και μείωση διαστάσεων [79]



Σχήμα 4.10: Προβολή των μεταβλητών στο πρώτο παραγοντικό επίπεδο [79]

# 5

## Τλοποίηση

Η παρούσα διπλωματική εργασία καταπιάνεται με το πρόβλημα της ταξινόμησης του καρκίνου του μαστού, με εφαρμογή αλγορίθμων μηχανικής μάθησης σε αριθμητικά δεδομένα που εξήχθησαν από βιοφίες με τη μέθοδο FNA (όπως αναφέρεται αναλυτικά στο υποκεφάλαιο 2.2). Στόχος είναι η εύρεση των βέλτιστων αλγορίθμων για την ταξινόμηση αυτού του τύπου καρκίνου και πιθανή χρήση τους στην ιατρική, μετά από την εκπαίδευσή τους σε μεγαλύτερους όγκους δεδομένων.

Η εκπαίδευση αλγορίθμων μηχανικής μάθησης καθώς και νευρωνικών δικτύων είναι μια διαδικασία που καταναλώνει πολλούς υπολογιστικούς πόρους του συστήματος. Συνεπώς μπορεί να γίνει πολύ χρονοβόρα όσο αυξάνεται η πολυπλοκότητά τους και ο όγκος των δεδομένων στα οποία εκπαιδεύονται. Έτσι, έχουν ληφθεί υπόψη τα παραπάνω και η εκπαίδευσή τους γίνεται σε μικρότερη κλίμακα σε σχέση με τα μοντέλα που βγαίνουν στην παραγωγή.

Η υλοποίηση χωρίζεται σε 5 μέρη:

- Προεπεξεργασία δεδομένων
- Οπτικοποίηση
- Επιλογή χαρακτηριστικών
- Μείωση διαστατικότητας
- Εφαρμογή των αλγορίθμων μηχανικής μάθησης

Αρχικά χρησιμοποιούνται τα διάφορα εργαλεία και οι τεχνικές που αναφέρθηκαν στο κεφάλαιο 4 για την επεξεργασία και την οπτικοποίηση των δεδομένων, που έχουν στόχο την καλύτερη αξιοποίησή τους στους αλγορίθμους μηχανικής μάθησης που θα εφαρμοστούν στη συνέχεια. Στη συνέχεια, παρουσιάζονται τα αποτελέσματα των αλγορίθμων που χρησιμοποιήθηκαν. Τέλος, να αναφερθεί ότι όλες οι προαναφερθείσες διαδικασίες υπάρχουν στον ιστοχώρο: <https://github.com/LazarosPan/Breast-Cancer-Classification-with-Machine-Learning-Methods>.

## 5.1 ΠΡΟ-ΕΠΕΞΕΡΓΑΣΙΑ ΔΕΔΟΜΕΝΩΝ

### 5.1.1 Διάβασμα των δεδομένων

Το πρώτο βήμα είναι η εισαγωγή των δεδομένων που θα χρησιμοποιηθούν στην ταξινόμηση. Αυτό, συνήθως γίνεται μέσω αρχείων CSV (Comma-Separated Values) αλλά μπορεί να γίνει και με άλλες μορφές αρχείων όπως για παράδειγμα τα αρχεία XLS, XLSX (Excel) και JSON (JavaScript Object Notation).

### 5.1.2 ”Καθαρισμός” των δεδομένων

Σε αυτό το στάδιο υλοποιήθηκαν τα παρακάτω:

#### 1. Αφαίρεση ανεπιθύμητων παρατηρήσεων

Αυτό περιλαμβάνει τη διαγραφή διπλότυπων/περιττών ή άσχετων τιμών από το σύνολο δεδομένων. Οι διπλές παρατηρήσεις προκύπτουν συχνότερα κατά τη συλλογή δεδομένων και οι άσχετες παρατηρήσεις είναι εκείνες που στην πραγματικότητα δεν ταιριάζουν στο συγκεκριμένο πρόβλημα.

#### 2. Διόρθωση δομικών σφαλμάτων

Τα σφάλματα που προκύπτουν κατά τη μέτρηση, τη μεταφορά δεδομένων ή άλλες παρόμοιες καταστάσεις ονομάζονται δομικά σφάλματα. Τα δομικά σφάλματα περιλαμβάνουν τυπογραφικά λάθη στο όνομα των χαρακτηριστικών, το ίδιο χαρακτηριστικό με διαφορετικό όνομα, κλάσεις με εσφαλμένη επισήμανση ή ασυνεπής χρήση κεφαλαίων.

#### 3. Διαχείριση ανεπιθύμητων έκτοπων τιμών

Τα έκτοπα σημεία μπορεί να προκαλέσουν προβλήματα με ορισμένους τύπους μοντέλων. Για παράδειγμα, τα μοντέλα γραμμικής παλινδρόμησης είναι λιγότερο ανθεκτικά σε έκτοπες τιμές από τα μοντέλα δέντρων αποφάσεων. Γενικότερα, η αφαίρεση των έκτοπων τιμών δεν είναι πάντα σωστή καθώς μερικές φορές επηρεάζει αρνητικά την απόδοση. Επομένως, η αφαίρεση έκτοπων τιμών πρέπει να γίνεται με προσοχή και μόνο σε ύποπτες μετρήσεις που είναι απίθανο να αποτελούν μέρος των πραγματικών δεδομένων.

#### 4. Χειρισμός δεδομένων που λείπουν

Η έλλειψη δεδομένων είναι ένα παραπλανητικά δύσκολο ζήτημα στη μηχανική μάθηση. Δεν αποτελεί σωστή πρακτική η αγνόηση ή η αφαίρεση δεδομένων που λείπουν. Πρέπει να αντιμετωπίζονται προσεκτικά καθώς μπορεί να αποτελούν ένδειξη για κάτι σημαντικό. Οι δύο πιο συνηθισμένοι τρόποι αντιμετώπισης δεδομένων που λείπουν είναι:

(α') Αφαίρεση παρατηρήσεων που περιλαμβάνουν τιμές που λείπουν.

- Το γεγονός ότι η τιμή έλειπε μπορεί από μόνο του να είναι κατατοπιστικό.

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---

- Επιπλέον, στον πραγματικό κόσμο, συχνά χρειάζεται να γίνουν προβλέψεις για νέα δεδομένα, ακόμα κι αν λείπουν κάποια από τα χαρακτηριστικά!

(β') Καταλογισμός των τιμών που λείπουν από προηγούμενες παρατηρήσεις.

- Η απουσία τιμών είναι σχεδόν πάντα κατατοπιστική από μόνη της και θα πρέπει να δηλωθεί στον αλγόριθμο.
- Ακόμα κι αν δημιουργηθεί ένα μοντέλο για να συμπληρωθούν οι τιμές που λείπουν, δεν προστίθεται καμία πραγματική πληροφορία. Απλώς ενισχύονται τα μοτίβα που παρέχονται ήδη από άλλα χαρακτηριστικά.

Στα συγκεκριμένα δεδομένα, παρατηρήθηκε ότι δεν υπήρχαν τιμές που λείπουν και δεν βρέθηκαν διπλότυπες ή άσχετες τιμές.

### 5.1.3 Υποδειγματοληψία

Η υποδειγματοληψία είναι μια τεχνική για την εξισορρόπηση ανομοιόμορφων συνόλων δεδομένων διατηρώντας όλα τα δεδομένα στην κατηγορία μειοφηφίας και μειώνοντας το μέγεθος της πλειοφηφικής τάξης. Είναι μία από τις πολλές τεχνικές που μπορούν να χρησιμοποιήσουν οι επιστήμονες δεδομένων για να εξάγουν πιο ακριβείς πληροφορίες από αρχικά μη ισορροπημένα σύνολα δεδομένων. Παρ' όλα τα μειονεκτήματά της, όπως η απώλεια δυνητικά σημαντικών πληροφοριών, παραμένει μια κοινή και σημαντική δεξιότητα για τους επιστήμονες δεδομένων.

- **Πλεονεκτήματα**

Το κύριο πλεονέκτημα της υποδειγματοληψίας είναι ότι οι επιστήμονες δεδομένων μπορούν να διορθώσουν τα μη ισορροπημένα δεδομένα, ώστε να μειωθεί ο κίνδυνος του να στραφούν οι αλγόριθμοι ανάλυσης ή μηχανικής μάθησης προς την πλειονότητα. Για παράδειγμα χωρίς επαναδειγματοληψία, οι επιστήμονες θα μπορούσαν να εκτελέσουν ένα μοντέλο ταξινόμησης με ακρίβεια 90%. Σε πιο προσεκτική εξέταση, ωστόσο, θα διαπιστώσουν ότι τα αποτελέσματα είναι σε μεγάλο βαθμό εντός της πλειοφηφικής τάξης. Αυτό είναι γνωστό ως το παράδοξο της ακρίβειας.

Με απλά λόγια, τα μειονοτικά γεγονότα είναι πιο δύσκολο να προβλεφθούν επειδή είναι λιγότερα. Ένας αλγόριθμος έχει λιγότερες πληροφορίες για να μάθει. Άλλα η επαναδειγματοληψία μέσω της υποδειγματοληψίας μπορεί να διορθώσει αυτό το ζήτημα και να καταστήσει την τάξη μειοφηφίας ίση με την τάξη της πλειοφηφίας για τους σκοπούς της ανάλυσης δεδομένων.

Άλλα πλεονεκτήματα της υποδειγματοληψίας περιλαμβάνουν τις λιγότερες απαιτήσεις αποθήκευσης των δεδομένων και μικρότερους χρόνους εκτέλεσης για αναλύσεις. Λιγότερα δεδομένα αποσκοπούν σε μικρότερο χώρο αποθήκευσης και λιγότερο χρόνο απόκτησης πολύτιμων πληροφοριών.

- **Μειονεκτήματα**

Κατά τη διάρκεια της υποδειγματοληψίας, οι επιστήμονες δεδομένων ή ο αλγόριθμος μηχανικής μάθησης αφαιρούν δεδομένα από την πλειοφηφική τάξη. Εξαιτίας αυτού, οι επιστήμονες μπορεί να χάσουν δυνητικά σημαντικές πληροφορίες. Σκεφτείτε τη διαφορά στον όγκο των δεδομένων στις τάξεις της πλειοφηφίας έναντι της

μειοψηφίας. Η αναλογία μπορεί να είναι 500:1, 1.000:1, 100.000:1 ή 1.000.000:1. Η κατάργηση αρκετών γεγονότων πλειοψηφίας ώστε η κλάση πλειοψηφίας να έχει ίδιο ή παρόμοιο μέγεθος με την κατηγορία μειοψηφίας έχει ως αποτέλεσμα σημαντική απώλεια δεδομένων.

Η απώλεια δυνητικά σημαντικών δεδομένων ισχύει ιδιαίτερα με την τυχαία υποδειγματοληψία όταν τα γεγονότα αφαιρούνται χωρίς να λαμβάνεται υπόψη το τι είναι και πόσο χρήσιμα μπορεί να είναι για την ανάλυση. Οι επιστήμονες δεδομένων μπορούν να αντιμετωπίσουν αυτό το μειονέκτημα χρησιμοποιώντας μια προσεκτική και καταποιητική τεχνική υποδειγματοληψίας. Μπορούν επίσης να καταπολεμήσουν την απώλεια δυνητικά σημαντικών δεδομένων συνδυάζοντας τεχνικές υποδειγματοληψίας και υπερδειγματοληψίας. Με αυτόν τον τρόπο, δεν μειώνουν απλώς την πλειοψηφική τάξη, αλλά αυξάνουν και την κατηγορία μειοψηφίας για να φτάσουν σε ένα ισορροπημένο σύνολο δεδομένων.

Ένα άλλο μειονέκτημα της υποδειγματοληψίας είναι ότι το δείγμα της πλειοψηφικής κατηγορίας που επιλέχθηκε θα μπορούσε να είναι προκατειλημμένο. Το δείγμα μπορεί να μην αντιπροσωπεύει με ακρίβεια τον πραγματικό κόσμο και το αποτέλεσμα της ανάλυσης μπορεί να είναι ανακριβές.

- **Συμπέρασμα**

Λόγω αυτών των μειονεκτημάτων, ορισμένοι επιστήμονες μπορεί να προτιμούν την υπερδειγματοληψία. Δεν οδηγεί σε απώλεια πληροφοριών και σε ορισμένες περιπτώσεις μπορεί να έχει καλύτερη απόδοση από την υποδειγματοληψία. Αλλά ούτε η υπερδειγματοληψία δεν είναι τέλεια. Επειδή η υπερδειγματοληψία συχνά περιλαμβάνει την αναπαραγωγή γεγονότων μειοψηφίας, μπορεί να οδηγήσει σε υπερπροσαρμογή. Για να εξισορροπηθούν αυτά τα ζητήματα, ορισμένα σενάρια ενδέχεται να απαιτούν συνδυασμό και των δύο για να αποκτηθεί το πιο "αληθινό" σύνολο δεδομένων και τα πιο ακριβή αποτελέσματα.

Στο συγκεκριμένο σετ δεδομένων, κρίθηκε σωστό να χρησιμοποιηθεί η υποδειγματοληψία (σχήμα 5.1) για τους παρακάτω λόγους:

1. Πρόκειται για δεδομένα από βιοψία, οπότε θεωρούνται μοναδικά και δεν θα ήταν σωστό να υπάρχει παραπάνω από μια φορά η ίδια εκχώρηση για τον ίδιο ασθενή.
2. Τα δείγματα που χάνονται είναι λίγα, οπότε δεν θεωρείται σοβαρή απώλεια πληροφορίας για την ταξινόμηση.
3. Στόχος είναι να γίνουν οι αλγόριθμοι όσο το δυνατόν πιο απλοί και πιο γρήγοροι, λόγω της απώλειας κάρτας γραφικών αλλά και του αδύναμων υπολογιστικών πόρων γενικότερα.

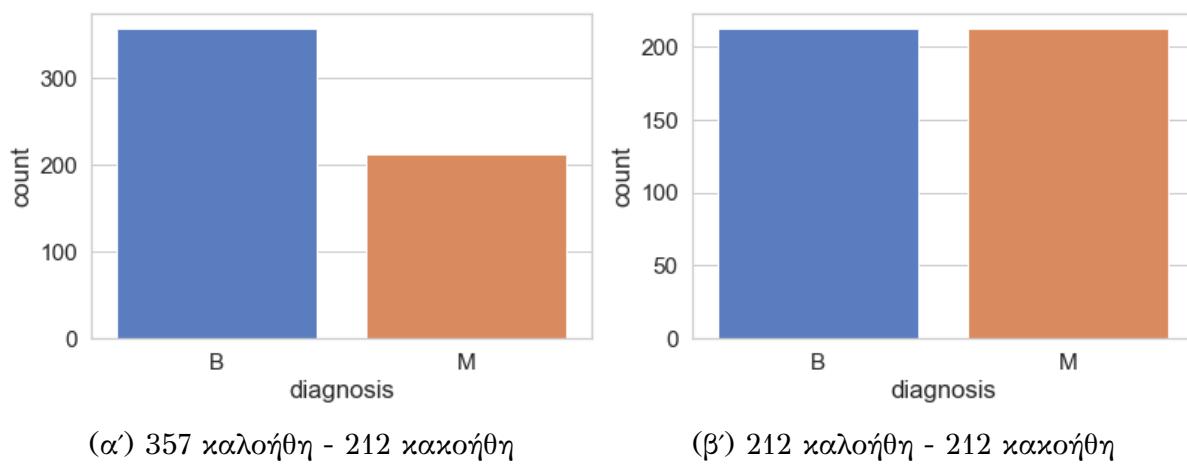
## 5.2 ΟΠΤΙΚΟΠΟΙΗΣΗ

---

Σε αυτή την ενότητα θα οπτικοποιηθούν τα δεδομένα προκειμένου να εξαχθούν κάποια συμπεράσματα γι' αυτά, τα οποία θα βοηθήσουν στην επιλογή των χαρακτηριστικών που θα μείνουν τελικά για να εισαχθούν στα μοντέλα μηχανικής μάθησης.

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---



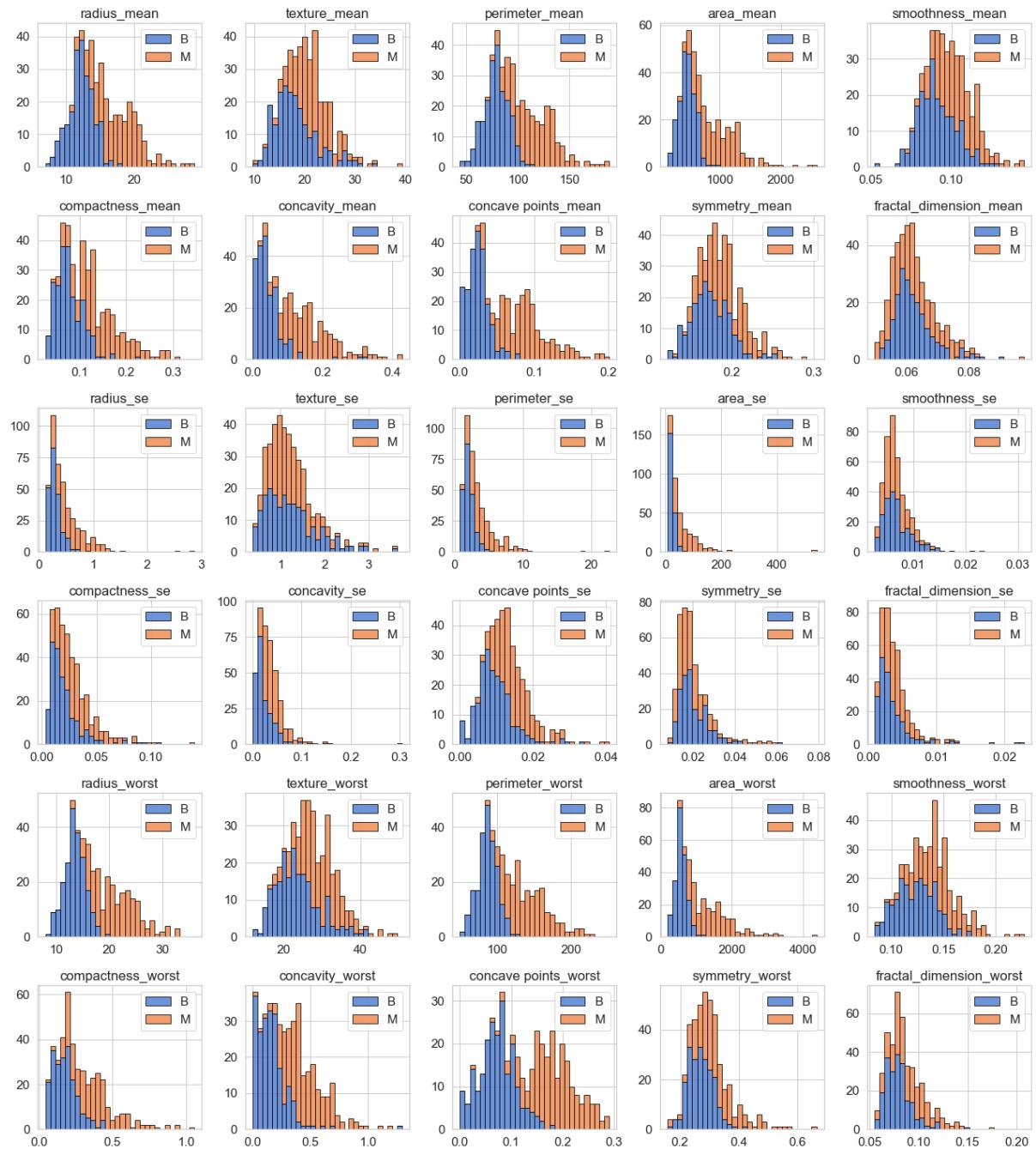
Σχήμα 5.1: Αριθμός δειγμάτων πριν και μετά την υποδειγματοληψία

Στο σχήμα 5.2 φαίνονται οι κατανομές όλων των χαρακτηριστικών που υπάρχουν στο σετ δεδομένων, τα οποία περιγράφονται αναλυτικά στο υποκεφάλαιο 2.2.  
Από το σχήμα 5.2 παρατηρούνται τα εξής:

1. Σε πολλά από τα χαρακτηριστικά, είναι εμφανής ο διαχωρισμός μεταξύ των δειγμάτων του καλοήθους και του κακοήθους καρκίνου (τα κακοήθη δείγματα έχουν συνήθως μεγαλύτερες τιμές). Αυτή η διαχωρισμότητα θα πρέπει να είναι όσο το δυνατόν μεγαλύτερη, ώστε τα χαρακτηριστικά να παρέχουν καλές πληροφορίες για την ταξινόμηση στα μοντέλα μηχανικής μάθησης που θα κατασκευαστούν στη συνέχεια. Εξετάζοντας αυτό, μπορεί να γίνει μια εκτίμηση για το ποια χαρακτηριστικά είναι πιθανό να είναι τα πιο ενδεικτικά για τους τύπους καρκίνου.
2. Οι τιμές των δειγμάτων έχουν μεγάλη απόκλιση μεταξύ των χαρακτηριστικών (π.χ. "area mean" →  $\{min = 143.5, max = 2501\}$  και "smoothness mean" →  $\{min = 0.05263, max = 0.1447\}$ ).

Χρειάζεται να γίνει κανονικοποίηση πριν από τα διαγράμματα βιολιού και σμήνους, επειδή οι διαφορές μεταξύ των τιμών των χαρακτηριστικών είναι πολύ μεγάλες για να φαίνονται στα διαγράμματα. Ένας ακόμη λόγος για να γίνει κανονικοποίηση, είναι ότι ορισμένες μέθοδοι μηχανικής μάθησης δίνουν καλύτερα αποτελέσματα εάν τα δεδομένα είναι κανονικοποιημένα.

## 5.2. ΟΠΤΙΚΟΠΟΙΗΣΗ



Σχήμα 5.2: Κατανομές των χαρακτηριστικών

Με μπλε χρώμα απεικονίζονται τα καλούχθη περιστατικά (Benign) ενώ με πορτοκαλί τα κακούχθη (Malignant).

Υπάρχουν διάφοροι τρόποι για την κανονικοποίηση των δειγμάτων. Μερικές από αυτές είναι ο μετασχηματισμός των δεδομένων έτσι ώστε:

- οι τιμές κάθε στήλης να κυμαίνονται στο διάστημα  $[0, 1]$ .
- οι τιμές κάθε στήλης να κυμαίνονται στο διάστημα  $[-1, 1]$ .
- οι τιμές κάθε στήλης να έχουν μέση τιμή 0 και διακύμανση 1.

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---

Όλα αυτά (και πολλά άλλα) μπορούν να γίνουν από το ‘scikit-learn’. Έτσι τα δεδομένα θα κανονικοποιηθούν ώστε να έχουν μέση τιμή 0 και διακύμανση 1, χρησιμοποιώντας τον ‘StandardScaler’. Αυτό σημαίνει ότι ο ‘StandardScaler’ θα μάθει, για κάθε στήλη, τη μέση τιμή  $\mu$  και τη διακύμανση  $\sigma^2$ . Στην πράξη ο ‘StandardScaler’ εκτελεί για κάθε στοιχείο  $x$  την πράξη:

$$x_{scaled} = \frac{x - \mu}{\sigma^2}$$

όπου  $\mu$  είναι ο μέσος όρος της στήλης και  $\sigma^2$  είναι η διακύμανση της στήλης.

To Seaborn θα χρησιμοποιηθεί προκειμένου να οπτικοποιηθούν τα δεδομένα και να αναδειχθεί η ποικιλομορφία των γραφικών παραστάσεων. Τα χαρακτηριστικά θα χωριστούν σε 3 ομάδες και κάθε ομάδα θα περιλαμβάνει 10 χαρακτηριστικά για να είναι ευκολότερη η παρατήρησή τους.

Το διάγραμμα βιολιού είναι παρόμοιο με το διάγραμμα κουτιού με μύστακες. Δείχνει την κατανομή των ποσοτικών δεδομένων σε διάφορα επίπεδα μιας (ή περισσότερων) κατηγορικών μεταβλητών, έτσι ώστε οι κατανομές αυτές να μπορούν να συγκριθούν. Σε αντίθεση με το διάγραμμα κουτιού, στο οποίο όλα τα στοιχεία του διαγράμματος αντιστοιχούν σε πραγματικά σημεία δεδομένων, το διάγραμμα βιολιού διαθέτει μια εκτίμηση πυκνότητας πυρήνα της υποκείμενης κατανομής.

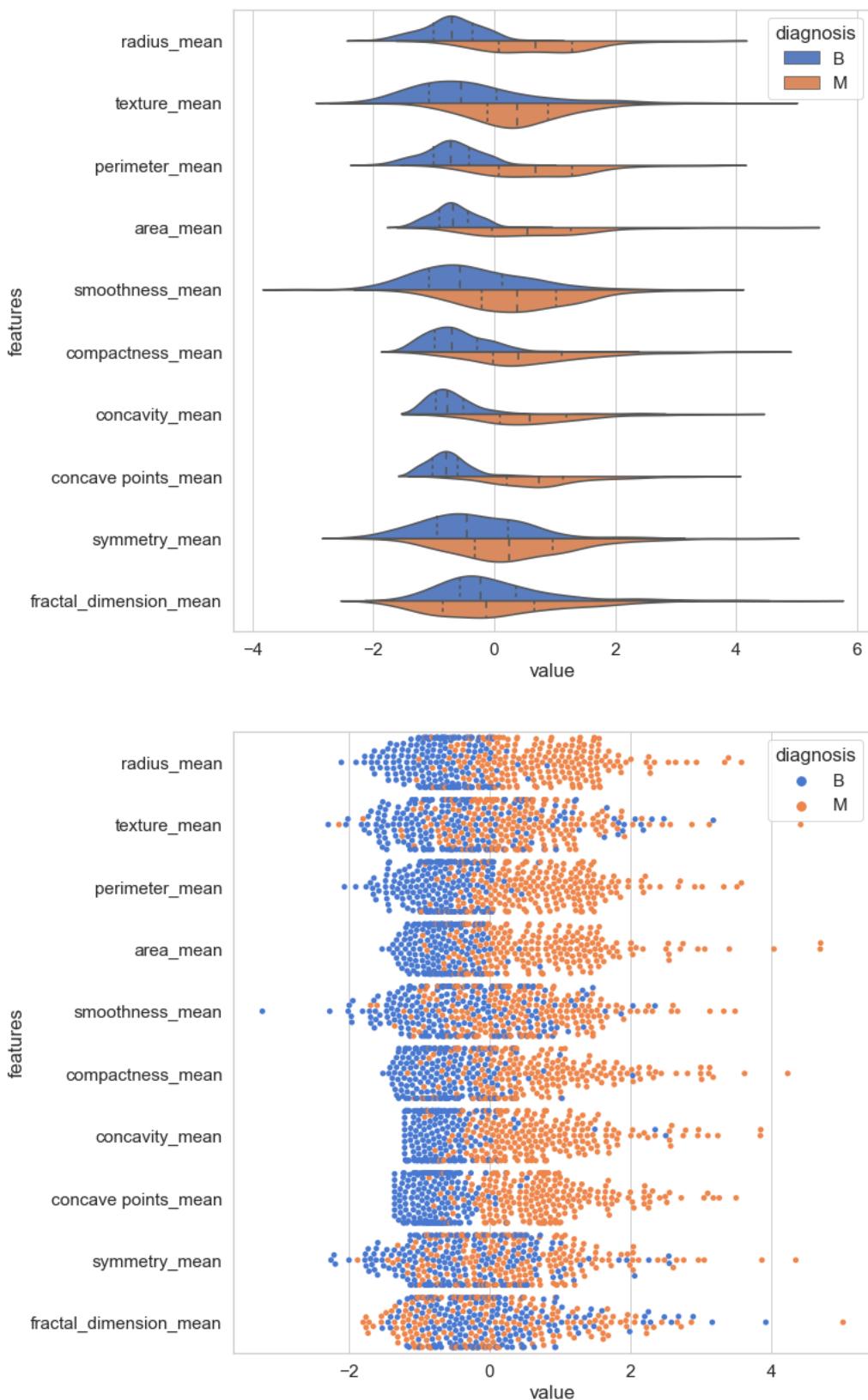
Στο διάγραμμα σμήνους, τα σημεία προσαρμόζονται (μόνο κατά μήκος του κατηγορικού άξονα) ώστε να μην επικαλύπτονται. Αυτό δίνει μια καλύτερη αναπαράσταση της κατανομής των τιμών, αλλά δεν κλιμακώνεται καλά σε μεγάλο αριθμό παρατηρήσεων. Αυτό το στυλ διαγράμματος ονομάζεται μερικές φορές ”μελισσοστοιχείο”. Ένα διάγραμμα σμήνους μπορεί να σχεδιαστεί από μόνο του, αλλά αποτελεί επίσης ένα καλό συμπλήρωμα ενός διαγράμματος κουτιού ή ενός διαγράμματος βιολιού σε περιπτώσεις όπου θέλει κανείς να παρουσιάσει όλες τις παρατηρήσεις μαζί με κάποια αναπαράσταση της υποκείμενης κατανομής.

Από το σχήμα 5.3 το οποίο απεικονίζει τις μέσες τιμές των χαρακτηριστικών, διακρίνεται ότι τα χαρακτηριστικά με την μεγαλύτερη διαχωρισμότητα στα δεδομένα μεταξύ καλοήθους και κακοήθους καρκίνου είναι τα: **radius mean, perimeter mean, area mean, compactness mean, concavity mean, concave points mean**.

Έπειτα από το σχήμα 5.4, στο οποίο φαίνονται τα τυπικά σφάλματα των χαρακτηριστικών, αυτά που θα δώσουν τις πιο χρήσιμες πληροφορίες για ταξινόμηση είναι τα **radius se, area se**. Ειδικά τα χαρακτηριστικά [texture se, smoothness se, compactness se, concavity se, symmetry se, fractal dimension se] φαίνεται να έχουν κάποιες υψηλές τιμές για την καλοήθη κλάση, πράγμα που θα ”μπερδέψει” τους αλγορίθμους. Όπως φάνηκε και από τις κατανομές, στα περισσότερα χαρακτηριστικά τα δείγματα που έχουν υψηλές τιμές ανήκουν στην κακοήθη κλάση.

Τέλος στο σχήμα 5.5 που δείχνει τις ακραίες τιμές των χαρακτηριστικών, αυτά που φαίνεται ότι θα είναι τα πιο καθοριστικά για την ταξινόμηση είναι τα : **radius worst, perimeter worst, area worst, concavity worst, concave points worst**.

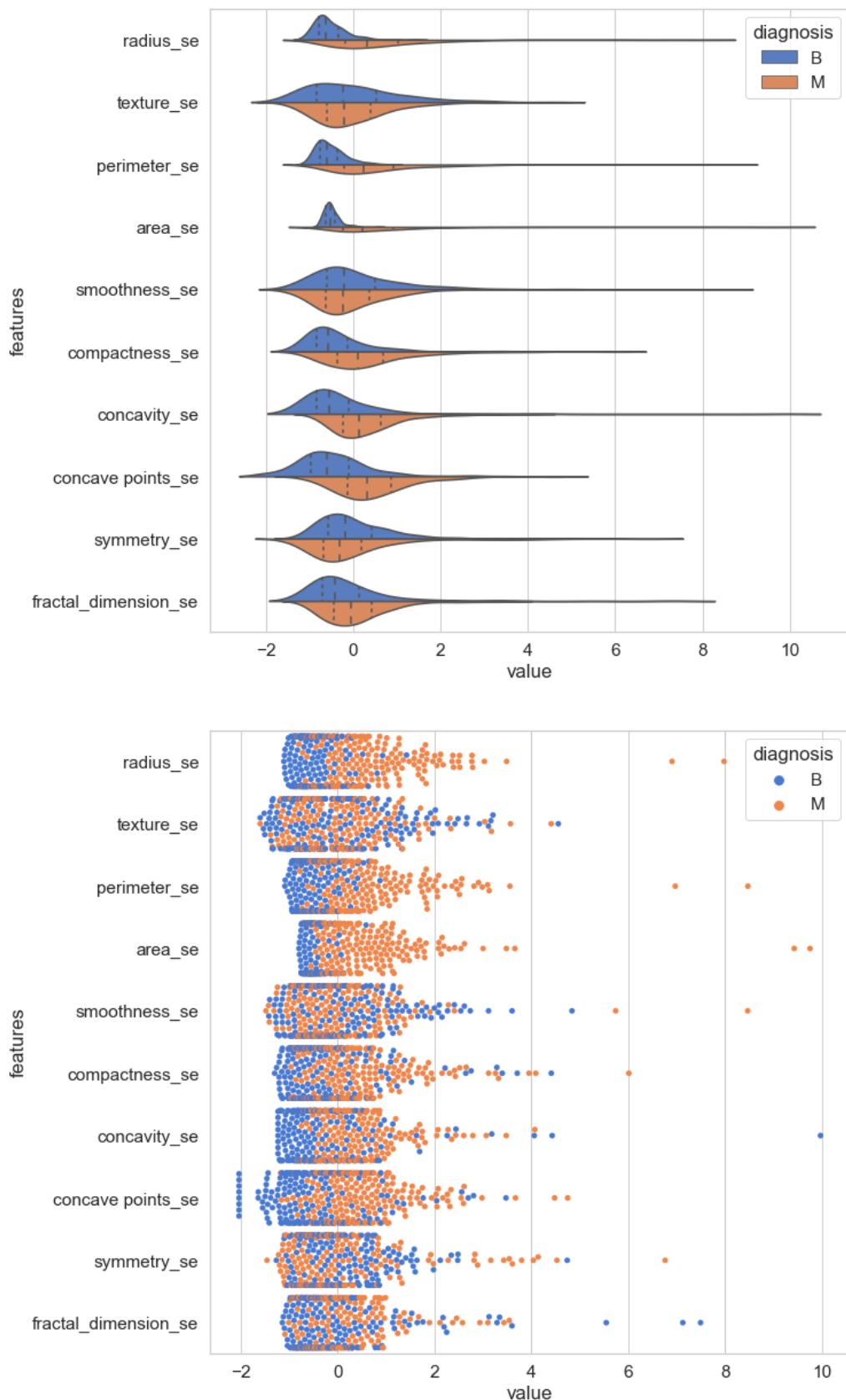
Παρόλο που τα δεδομένα είναι καλό να έχουν όσο το δυνατόν μεγαλύτερη διαχωρισμότητα, αυτό από μόνο του δεν αρκεί, καθώς κάποια από αυτά μπορεί να έχουν μεγάλη συσχέτιση μεταξύ τους, ή με άλλα λόγια να είναι σχεδόν τα ίδια χαρακτηριστικά. Παρακάτω θα γίνει ανάλυση της συσχέτισης αυτής και θα επιλεχθούν τα χαρακτηριστικά με την μικρότερη μεταξύ τους συσχέτιση.



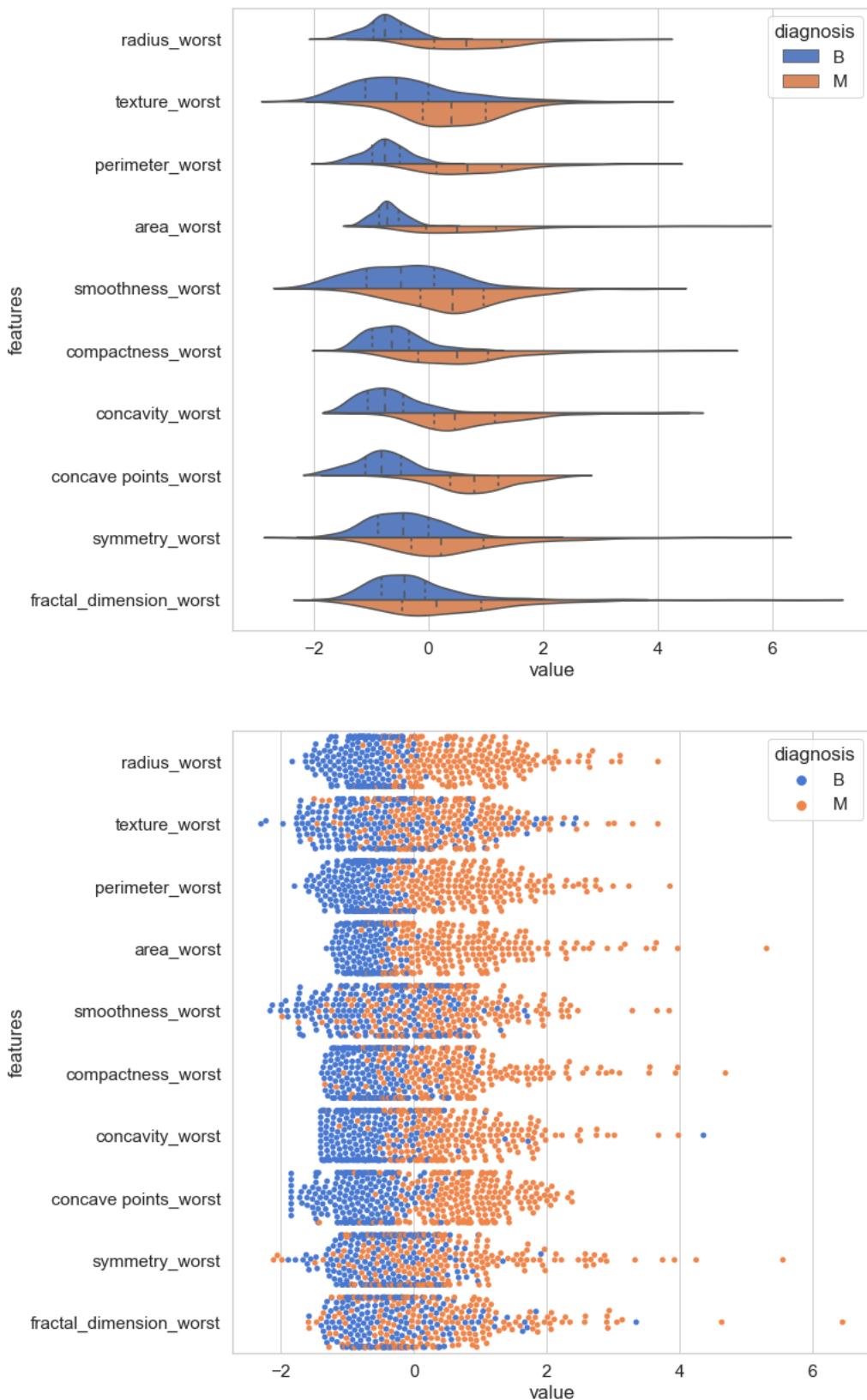
Σχήμα 5.3: Μέσες τιμές των χαρακτηριστικών

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---



Σχήμα 5.4: Τυπικό σφάλμα των χαρακτηριστικών



Σχήμα 5.5: Ακραίες τιμές των χαρακτηριστικών

## 5.3 ΕΠΙΛΟΓΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

### 5.3.1 Επιλογή χαρακτηριστικών βάση συσχέτισης

#### Συντελεστής Pearson r

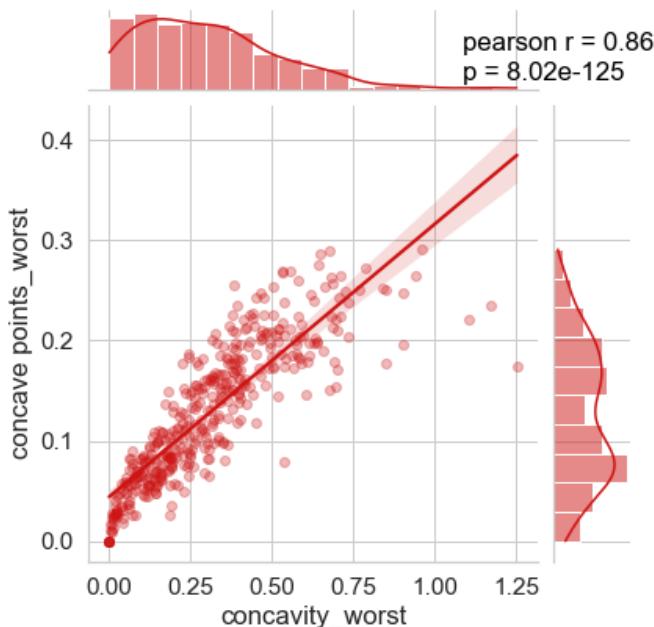
Ο συντελεστής συσχέτισης Pearson μετρά τη γραμμική σχέση μεταξύ δύο συνόλων δεδομένων. Όπως και άλλοι συντελεστές συσχέτισης, αυτός κυμαίνεται μεταξύ  $-1$  και  $+1$  με το  $0$  να σημαίνει ότι δεν υπάρχει συσχέτιση. Οι συσχετίσεις  $-1$  ή  $+1$  υποδηλώνουν ακριβή γραμμική σχέση. Οι θετικές συσχετίσεις υποδηλώνουν ότι καθώς αυξάνεται το  $x$ , αυξάνεται και το  $y$ . Οι αρνητικές συσχετίσεις υποδηλώνουν ότι καθώς αυξάνεται το  $x$ , μειώνεται το  $y$ .

Με δεδομένα  $n$  ζευγάρια δεδομένων  $\{(x_1, y_1), \dots, (x_n, y_n)\}$ , ο συντελεστής  $r_{xy}$  μπορεί να διατυπωθεί ως:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2} \sqrt{\sum_{i=1}^n (y_i - \bar{y})^2}}$$

Αυτή η συνάρτηση εκτελεί επίσης έναν έλεγχο της μηδενικής υπόθεσης ότι οι κατανομές που διέπουν τα δείγματα είναι ασυσχέτιστες και κανονικά κατανεμημένες. Η τιμή  $p$  δείχνει κατά προσέγγιση την πιθανότητα ένα ασυσχέτιστο σύστημα να παράγει σύνολα δεδομένων που έχουν συσχέτιση Pearson τουλάχιστον τόσο ακραία όσο και αυτή που υπολογίζεται από αυτά τα σύνολα δεδομένων.

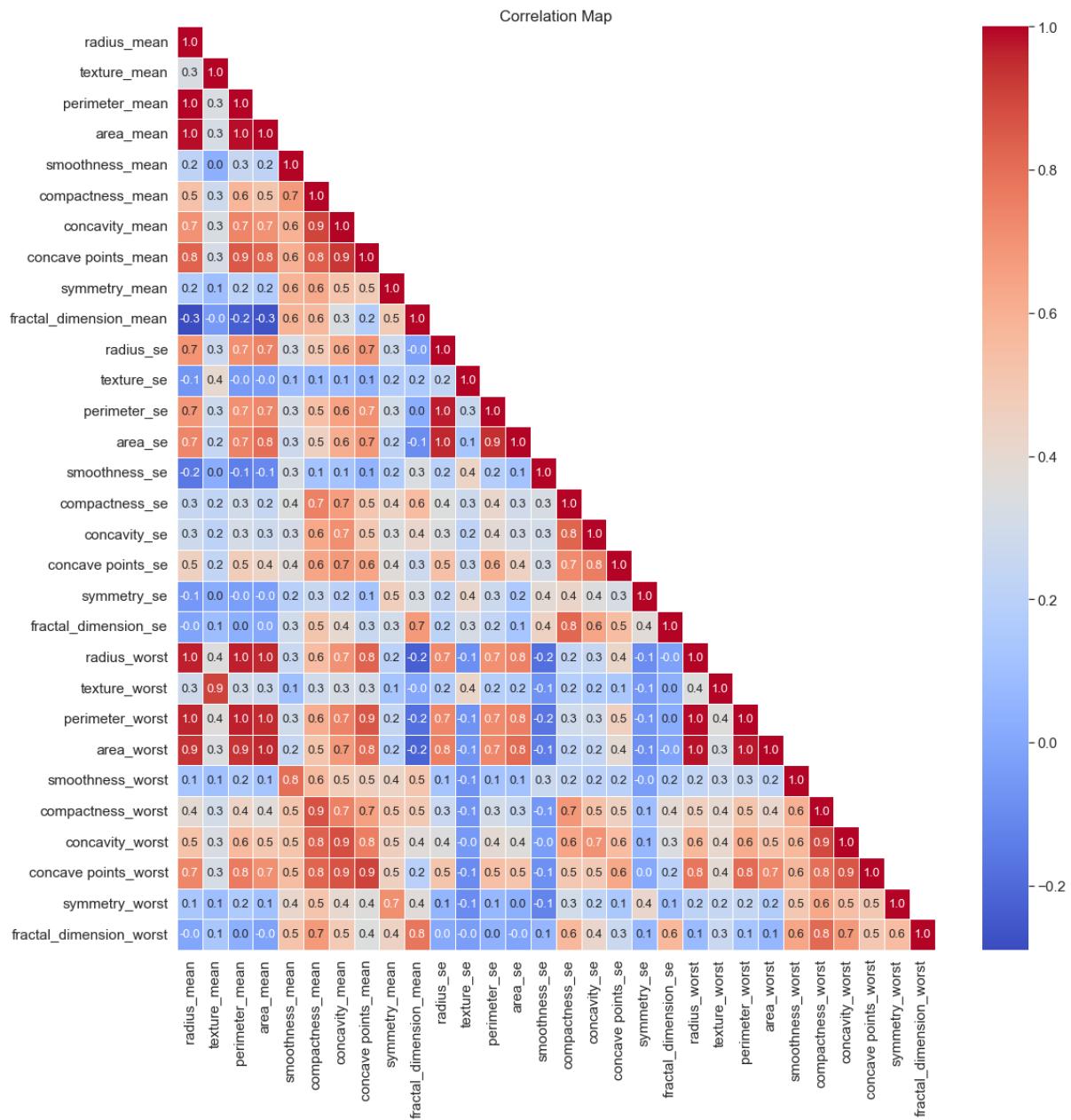
Στο σχήμα 5.6, για παράδειγμα, παρουσιάζεται η συσχέτιση του συντελεστή Pearson μεταξύ *concavity\_worst* και *concave points worst*.



Σχήμα 5.6: Γραμμική συσχέτιση μεταξύ 2 χαρακτηριστικών

### 5.3. ΕΠΙΛΟΓΗ ΧΑΡΑΚΤΗΡΙΣΤΙΚΩΝ

Για να παρατηρηθεί η συσχέτιση μεταξύ όλων των χαρακτηριστικών, θα χρησιμοποιηθεί ο χάρτης θερμότητας όπως φαίνεται στο σχήμα 5.7.



Σχήμα 5.7: Γραμμική συσχέτιση όλων των χαρακτηριστικών

Τα χαρακτηριστικά που συσχετίζονται σε μεγάλο βαθμό μεταξύ τους ( $r > 0.8$ ), μπορούν να αφαιρεθούν καθώς δεν είναι χρήσιμα για την ταξινόμηση. Επιπλέον, θα καταναλώσουν υπολογιστικούς πόρους από την εκπαίδευση των μοντέλων μηχανικής μάθησης.

### Αμοιβαία Πληροφορία (Mutual Information)

Η αμοιβαία πληροφορία (MI) μοιάζει πολύ με τη συσχέτιση, καθώς μετράει μια σχέση μεταξύ δύο ποσοτήτων. Το πλεονέκτημα της αμοιβαίας πληροφορίας είναι ότι μπορεί να ανιχνεύσει κάθε είδους σχέση, ενώ η συσχέτιση με τον συντελεστή Pearson ανιχνεύει μόνο γραμμικές σχέσεις.

Η αμοιβαία πληροφορία μεταξύ δύο τυχαίων μεταβλητών μετρά τις μη γραμμικές σχέσεις μεταξύ τους [80]. Εκτός αυτού, δείχνει πόση πληροφορία μπορεί να ληφθεί από μια τυχαία μεταβλητή παρατηρώντας μια άλλη τυχαία μεταβλητή.

Συνδέεται στενά με την έννοια της εντροπίας. Αυτό οφείλεται στο γεγονός ότι μπορεί επίσης να είναι γνωστή ως η μείωση της αβεβαιότητας μιας τυχαίας μεταβλητής εάν μια άλλη είναι γνωστή. Επομένως, μια υψηλή τιμή αμοιβαίας πληροφορίας υποδηλώνει μεγάλη μείωση της αβεβαιότητας, ενώ μια χαμηλή τιμή υποδηλώνει μικρή μείωση. Εάν η αμοιβαία πληροφορία είναι μηδέν, αυτό σημαίνει ότι οι δύο τυχαίες μεταβλητές είναι ανεξάρτητες.

Για δύο διακριτές μεταβλητές  $X$  και  $Y$  των οποίων η κοινή κατανομή πιθανότητας είναι  $P_{XY}(x, y)$ , η αμοιβαία πληροφορία μεταξύ τους, η οποία συμβολίζεται ως  $I(X; Y)$ , δίνεται από τη σχέση [81]:

$$I(X; Y) = \sum_{x,y} P_{XY}(x, y) \log \frac{P_{XY}(x, y)}{P_X(x)P_Y(y)} = E_{P_{XY}} \log \frac{P_{XY}}{P_X P_Y}$$

Εδώ τα  $P_X(x)$  και  $P_Y(y)$  είναι τα οι οριακές συναρτήσεις μάζας πιθανότητας

$$P_X(x) = \sum_y P_{XY}(x, y) \quad , \quad P_Y(y) = \sum_x P_{XY}(x, y)$$

και  $E_P$  είναι η αναμενόμενη τιμή στην κατανομή  $P$ .

Όπως εξηγήθηκε προηγουμένως, σχετίζεται με την εντροπία. Η σχέση αυτή φαίνεται στον ακόλουθο τύπο:

$$\begin{aligned} I(X; Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X) + H(Y) - H(X, Y) \\ &= H(X, Y) - H(X|Y) - H(Y|X) \end{aligned}$$

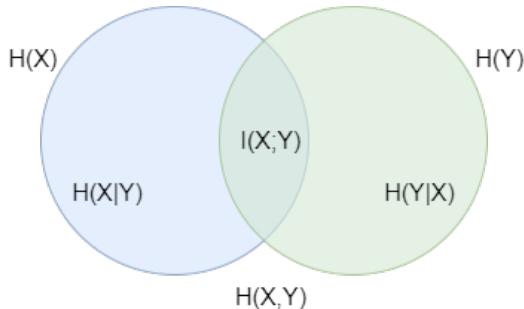
Η εντροπία ( $H$ ) μετρά το επίπεδο της αναμενόμενης αβεβαιότητας σε μια τυχαία μεταβλητή.

$$H(X) = - \sum_{x_i \in X} P(X = x_i) \cdot \log(P(X = x_i))$$

Η υπό συνθήκη εντροπία μετράει πόση αβεβαιότητα έχει η τυχαία μεταβλητή  $X$  όταν είναι γνωστή η τιμή της  $Y$ .

$$H(X|Y) = - \sum_{x,y} p(x, y) \cdot \log(p(x|y))$$

Για την καλύτερη κατανόηση, η σχέση μεταξύ εντροπίας και αμοιβαίας πληροφορίας απεικονίζεται στο σχήμα 5.8, όπου η περιοχή που μοιράζονται οι δύο κύκλοι είναι η αμοιβαία πληροφορία:



Σχήμα 5.8: Διάγραμμα Venn μεταξύ Εντροπίας και Αμοιβαίας Πληροφορίας [80]

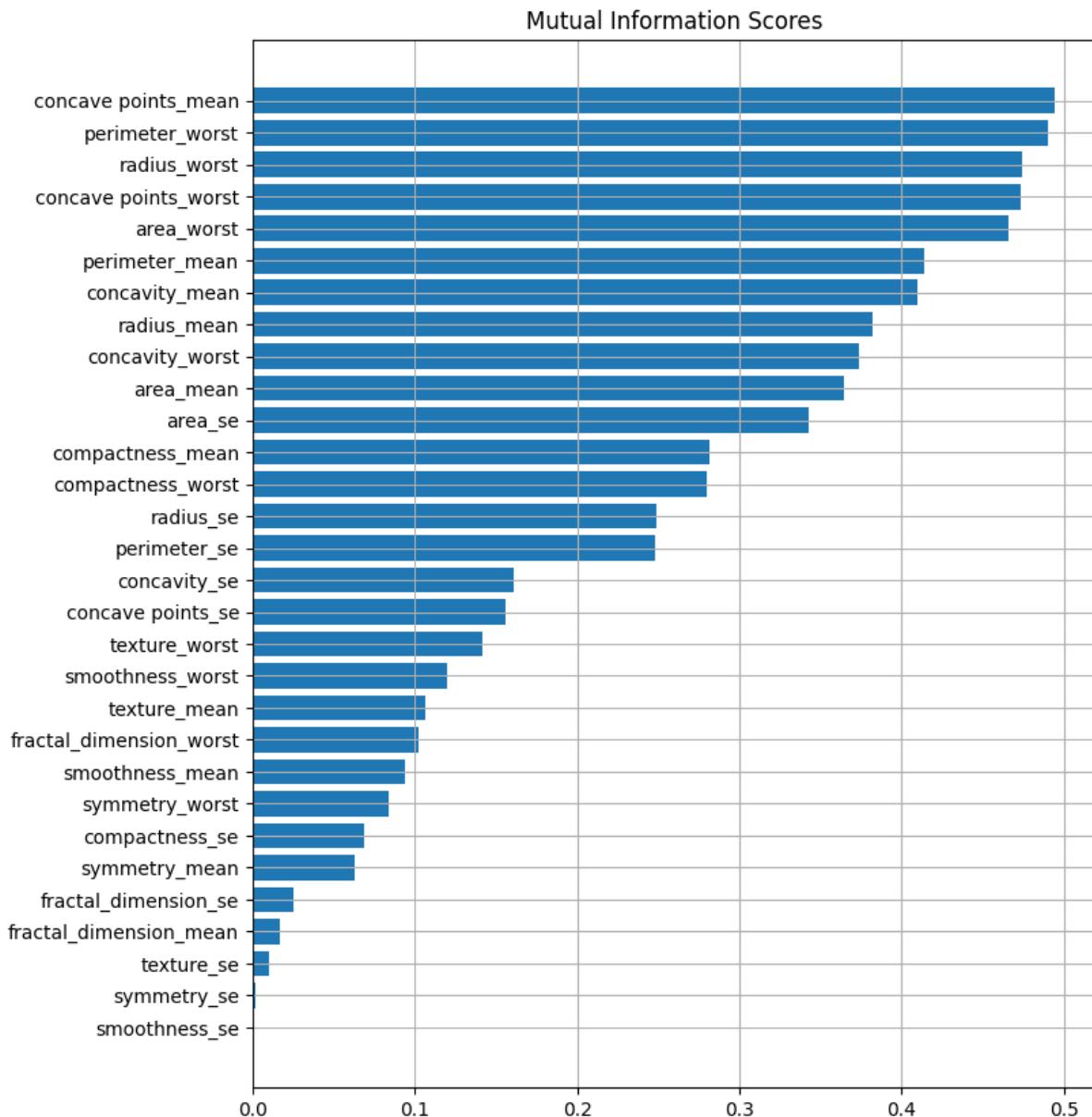
Ακολουθούν ορισμένα πράγματα που πρέπει να λάβει κανείς υπόψη κατά την εφαρμογή της αμοιβαίας πληροφόρησης:

1. Η MI μπορεί να βοηθήσει στην κατανόηση της σχετικής δυνατότητας ενός χαρακτηριστικού ως προβλεπτικού παράγοντα του στόχου, εξεταζόμενο από μόνο του.
2. Είναι δυνατόν ένα χαρακτηριστικό να είναι πολύ κατατοπιστικό όταν αλληλεπιδρά με άλλα χαρακτηριστικά, αλλά όχι τόσο κατατοπιστικό από μόνο του. Η MI δεν μπορεί να ανιχνεύσει αλληλεπιδράσεις μεταξύ χαρακτηριστικών. Πρόκειται για μια μονοπαραγοντική μετρική.
3. Η πραγματική χρησιμότητα ενός χαρακτηριστικού εξαρτάται από το μοντέλο με το οποίο θα χρησιμοποιηθεί. Ένα χαρακτηριστικό είναι χρήσιμο μόνο στον βαθμό που η σχέση του με τον στόχο είναι μια σχέση που το μοντέλο μπορεί να μάθει. Το γεγονός ότι ένα χαρακτηριστικό έχει υψηλή βαθμολογία MI δεν σημαίνει ότι το μοντέλο θα είναι σε θέση να κάνει οτιδήποτε με αυτές τις πληροφορίες. Το χαρακτηριστικό μπορεί να χρειαστεί να μετασχηματιστεί πρώτα για να αποκαλυφθεί η συσχέτιση.

Στο σχήμα 5.9 φαίνεται η αμοιβαία πληροφορία όλων των χαρακτηριστικών σε σχέση με τον στόχο, δηλαδή τη διάγνωση.

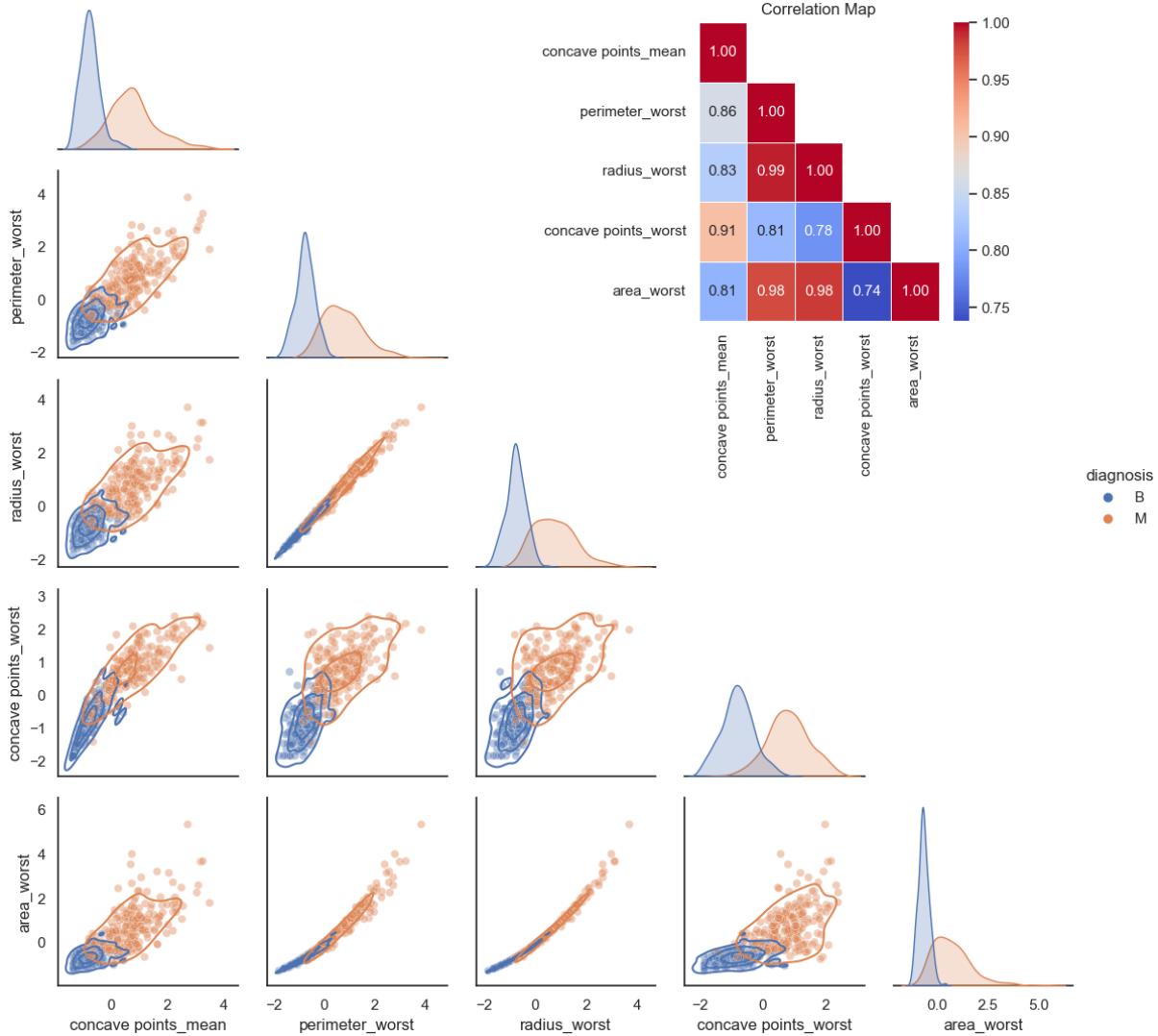
## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---



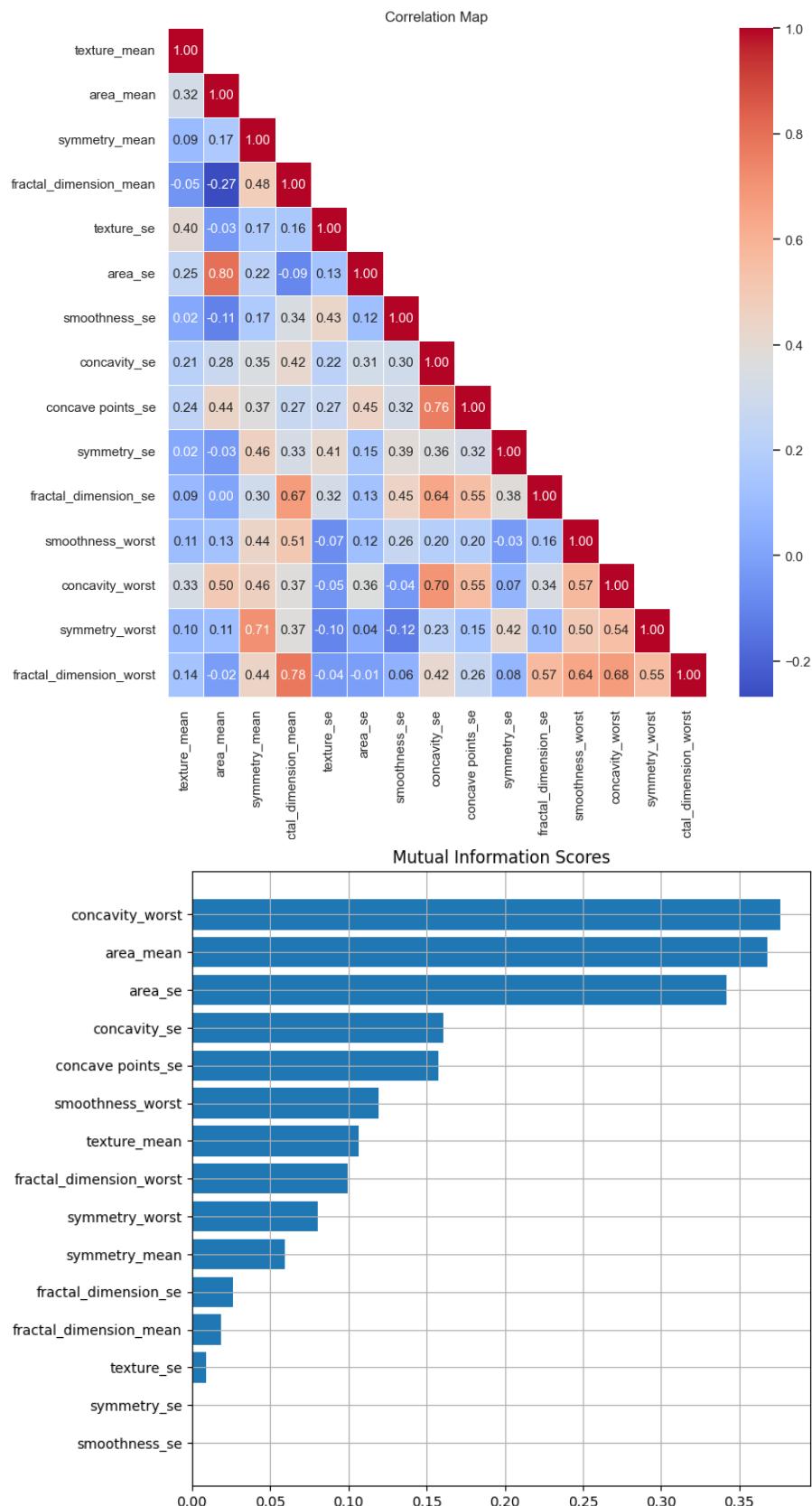
Σχήμα 5.9: Αμοιβαία πληροφορία όλων των χαρακτηριστικών σε σχέση με τη διάγνωση

Πιο πάνω, στα γραφήματα βιολιού και συμήνους, φάνηκαν τα χαρακτηριστικά με τη μεγαλύτερη διαχωρισιμότητα. Όλα αυτά είναι και τα χαρακτηριστικά με το μεγαλύτερο MI σκορ, που αποδεικνύει και την σχέση της αμοιβαίας πληροφορίας με την εντροπία των δεδομένων. Παρόλο που κάποια χαρακτηριστικά έχουν υψηλό MI σκορ, δεν θα προσφέρουν χρήσιμες πληροφορίες στα μοντέλα μηχανικής μάθησης, καθώς μπορεί να είναι συσχετισμένα μεταξύ τους. Για παράδειγμα, όπως φαίνεται στο σχήμα 5.9, τα 5 χαρακτηριστικά με το υψηλότερο MI σκορ είναι τα ['concave points mean', 'perimeter worst', 'radius worst', 'concave points worst', 'area worst'].



Σχήμα 5.10: Σχέση μεταξύ των χαρακτηριστικών με το μεγαλύτερο MI σκορ

Όπως φαίνεται στο σχήμα 5.10, αυτά τα 5 χαρακτηριστικά εκτός από υψηλό MI σκορ, εκτός από τη μεγάλη διαχωρισμότητα στα δεδομένα τους, έχουν και μεγάλη γραμμική συσχέτιση μεταξύ τους. Αυτό φαίνεται στο γράφημα αλλά και στο χάρτη θερμότητας. Την χαμηλότερη συσχέτιση την έχει το *area worst* με το *concave points worst* που είναι  $r = 0.74$  (ήδη αρκετά μεγάλη) και την μεγαλύτερη το *radius worst* με το *perimeter worst* που είναι  $r = 0.99$  (σχεδόν τέλεια γραμμική συσχέτιση). Οπότε, όλα αυτά τα χαρακτηριστικά με μεγάλη συσχέτιση ( $r > 0.8$  όπως αναφέρθηκε και παραπάνω) θα διωχθούν από αυτά που θα χρησιμοποιηθούν τελικά (σχήμα 5.11). Επίσης για να μειωθεί περαιτέρω ο αριθμός των χαρακτηριστικών, θα αφαιρεθούν αυτά που έχουν  $MI = 0$ , τα οποία είναι τα *symmetry se* και *smoothness se*.



Σχήμα 5.11: Χαρακτηριστικά με αποδεκτή τιμή γραμμικής συσχέτισης

### 5.3.2 Μονομεταβλητή επιλογή χαρακτηριστικών

Το API του Scikit-learn παρέχει την κλάση SelectKBest για την εξαγωγή των καλύτερων χαρακτηριστικών από ένα σύνολο δεδομένων. Λαμβάνει ως παράμετρο μια συνάρτηση βαθμολογίας, η οποία πρέπει να είναι εφαρμόσιμη σε ένα ζεύγος  $(X, y)$ . Η συνάρτηση βαθμολογίας πρέπει να επιστρέψει έναν πίνακα βαθμολογιών, μία για κάθε χαρακτηριστικό  $X[:, i]$  του  $X$  (επιπλέον, μπορεί επίσης να επιστρέψει  $p$ -τιμές, αλλά αυτές δεν χρειάζονται ούτε απαιτούνται). Στη συνέχεια, η SelectKBest απλώς διατηρεί τα πρώτα  $k$  χαρακτηριστικά του  $X$  με τις υψηλότερες βαθμολογίες.

Στην περίπτωση αυτή η SelectKBest χρησιμοποιεί τη συνάρτηση 'f\_classif' σκορ. Αυτή ερμηνεύει τις τιμές του  $y$  ως ετικέτες κλάσης και υπολογίζει, για κάθε χαρακτηριστικό  $X[:, i]$  του  $X$ , μια  $F$ -στατιστική. Ο τύπος που χρησιμοποιείται είναι αυτός:

$$F = \frac{\text{εξηγημένη διακύμανση}}{\text{ανεξήγητη διακύμανση}} \quad \text{ή} \quad F = \frac{\text{μεταβλητότητα μεταξύ των ομάδων}}{\text{μεταβλητότητα εντός των ομάδων}}$$

Η "εξηγημένη διακύμανση" ή "μεταβλητότητα μεταξύ ομάδων" είναι:

$$\sum_{i=1}^K n_i (\bar{Y}_{i\cdot} - \bar{Y})^2 / (K - 1)$$

όπου το  $\bar{Y}_{i\cdot}$  συμβολίζει τον μέσο όρο του δείγματος στην  $i$ -οστή ομάδα,  $n_i$  είναι ο αριθμός των παρατηρήσεων στην  $i$ -οστή ομάδα, το  $\bar{Y}$  συμβολίζει τον συνολικό μέσο όρο των δεδομένων και το  $K$  συμβολίζει τον αριθμό των ομάδων.

Η "ανεξήγητη διακύμανση" ή "μεταβλητότητα εντός των ομάδων" είναι

$$\sum_{i=1}^K \sum_{j=1}^{n_i} (Y_{ij} - \bar{Y}_{i\cdot})^2 / (N - K)$$

όπου  $Y_{ij}$  είναι η  $j$ -οστή παρατήρηση στην  $i$ -οστή από τις  $K$  ομάδες και  $N$  είναι το συνολικό μέγεθος του δείγματος. Αυτή η  $F$ -στατιστική ακολουθεί την  $F$ -κατανομή με βαθμούς ελευθερίας  $d_1 = K - 1$  και  $d_2 = N - K$  υπό την μηδενική υπόθεση. Το στατιστικό θα είναι μεγάλο αν η μεταβλητότητα μεταξύ των ομάδων είναι μεγάλη σε σχέση με τη μεταβλητότητα εντός των ομάδων, πρόγμα που είναι απίθανο να συμβεί αν οι πληθυσμιακοί μέσοι των ομάδων έχουν όλοι την ίδια τιμή.

Για να βρεθεί ο βέλτιστος αριθμός  $k$  χαρακτηριστικών που θα παραμείνουν, χρησιμοποιήθηκε η αναζήτηση πλέγματος δηλαδή το GridSearchCV.

Οπότε για  $k = 8$ , τα χαρακτηριστικά με το μεγαλύτερο σκορ ήταν τα : 'texture mean', 'area mean', 'symmetry mean', 'area se', 'concave points se', 'smoothness worst', 'concavity worst', 'symmetry worst'.

### 5.3.3 Αναδρομική εξάλειψη χαρακτηριστικών

#### RFE

Δεδομένου ενός εξωτερικού εκτιμητή που αποδίδει βάρη στα χαρακτηριστικά (π.χ. τους συντελεστές ενός γραμμικού μοντέλου), ο στόχος της αναδρομικής απαλοιφής χαρακτηριστικών (RFE) είναι η επιλογή χαρακτηριστικών με αναδρομική

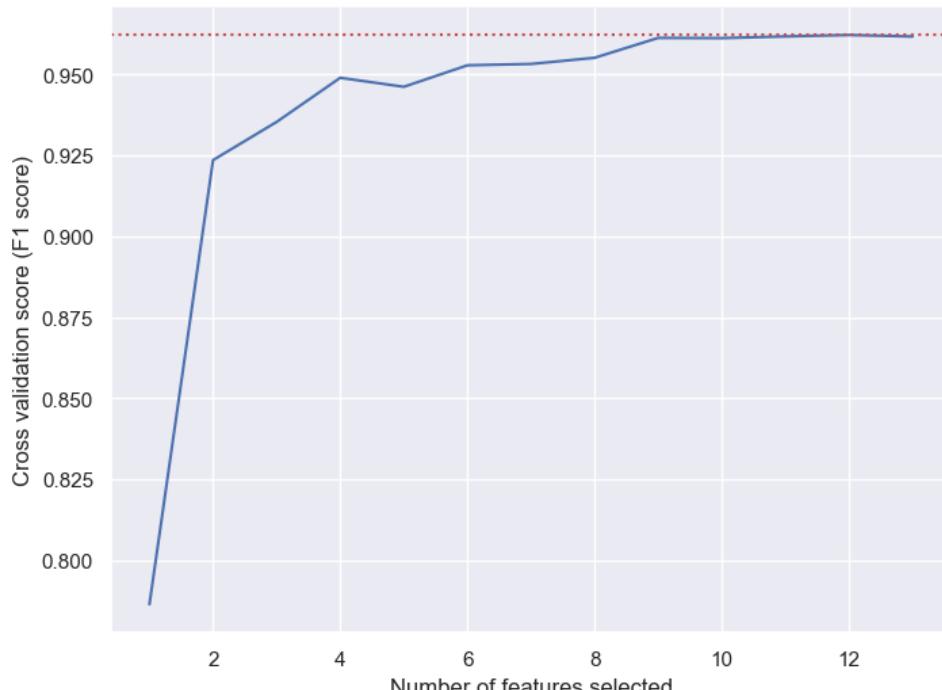
## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

εξέταση όλο και μικρότερων συνόλων χαρακτηριστικών. Αρχικά, ο εκτιμητής εκπαιδεύεται στο αρχικό σύνολο χαρακτηριστικών και η σπουδαιότητα κάθε χαρακτηριστικού λαμβάνεται είτε μέσω οποιουδήποτε συγκεκριμένου χαρακτηριστικού είτε μέσω κλήσης. Στη συνέχεια, τα λιγότερο σημαντικά χαρακτηριστικά περικόπτονται από το τρέχον σύνολο χαρακτηριστικών. Η διαδικασία αυτή επαναλαμβάνεται αναδρομικά στο σύνολο των περικοπών μέχρι να επιτευχθεί τελικά ο επιθυμητός αριθμός των προς επιλογή χαρακτηριστικών.

Σε αυτήν την εργασία, ο εξωτερικός εκτιμητής που χρησιμοποιήθηκε σε όλες τις μεθόδους επιλογής χαρακτηριστικών (που έχουν σαν όρισμα έναν εκτιμητή), είναι τα τυχαία δάση. Επίσης, βάση του  $k = 8$  που επιλέχθηκε στην μέθοδο SelectKBest, έτσι και σε αυτήν την μέθοδο επιλέχθηκε (χειροκίνητα αυτή τη φορά)  $n_{features} = 8$ . Αυτά τα 8 που έβγαλε ο αλγόριθμος είναι τα : **'texture mean', 'area mean', 'area se', 'concavity se', 'concave points se', 'smoothness worst', 'concavity worst', 'symmetry worst'**.

### RFECV

Το RFECV εκτελεί το RFE σε έναν βρόχο διασταυρούμενης επικύρωσης για την εύρεση του βέλτιστου αριθμού χαρακτηριστικών όπως φαίνεται στο σχήμα 5.12. Ο ιδανικός αριθμός χαρακτηριστικών σύμφωνα με τον αλγόριθμο RFECV είναι 12, τα οποία είναι τα εξής : **'texture mean', 'area mean', 'symmetry mean', 'fractal dimension mean', 'area se', 'concavity se', 'concave points se', 'fractal dimension se', 'smoothness worst', 'concavity worst', 'symmetry worst', 'fractal dimension worst'**.



Σχήμα 5.12: Μέσο όρος των F1 σκορ του RFECV

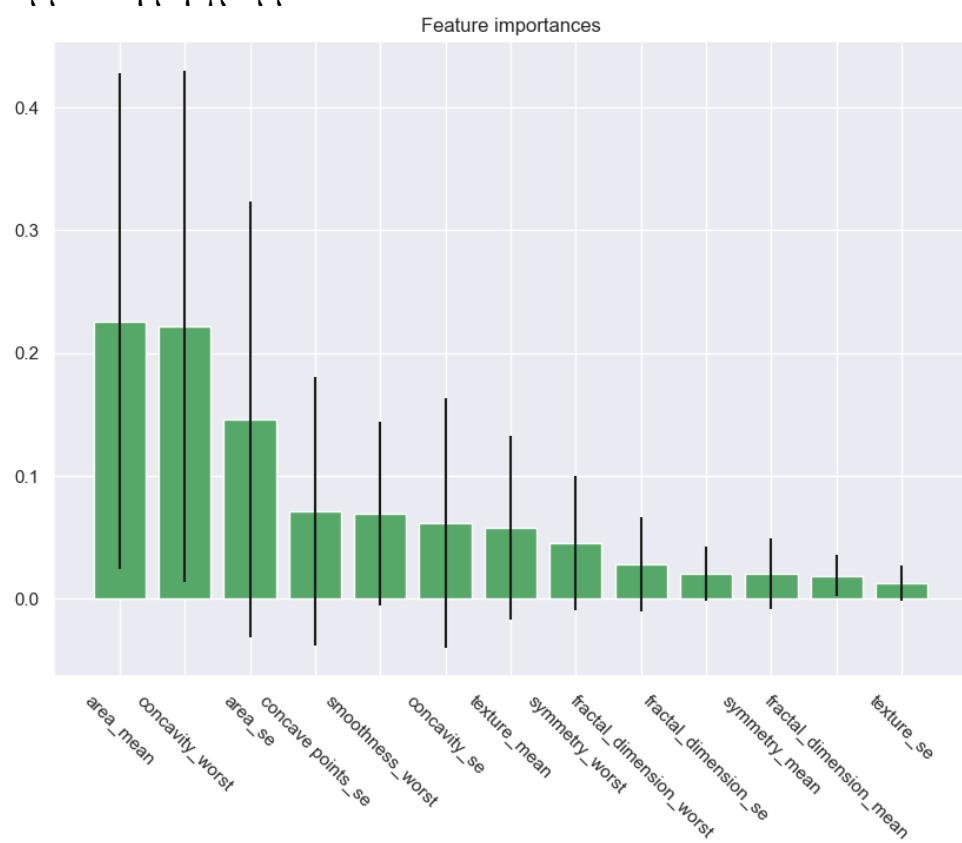
### 5.3.4 Σημαντικότητα των χαρακτηριστικών

Η σπουδαιότητα των χαρακτηριστικών είναι μια βασική έννοια στη μηχανική μάθηση που αναφέρεται στη σχετική σπουδαιότητα κάθε χαρακτηριστικού στα δεδομένα εκπαίδευσης. Με άλλα λόγια, λέει ποια χαρακτηριστικά είναι πιο προγνωστικά για τη μεταβλητή-στόχο. Η σημασία των χαρακτηριστικών μπορεί να υπολογιστεί με διάφορους τρόπους, αλλά όλες οι μέθοδοι βασίζονται συνήθως στον υπολογισμό κάποιου είδους βαθμολογίας που μετρά πόσο συχνά χρησιμοποιείται ένα χαρακτηριστικό στο μοντέλο και πόσο συμβάλλει στις συνολικές προβλέψεις.

Η σημασία των χαρακτηριστικών μπορεί να μετρηθεί σε μια κλίμακα από το 0 έως το 1, με το 0 να υποδεικνύει ότι το χαρακτηριστικό δεν έχει καμία σημασία και το 1 να υποδεικνύει ότι το χαρακτηριστικό είναι απολύτως απαραίτητο. Οι τιμές σημαντικότητας χαρακτηριστικών μπορούν επίσης να είναι αρνητικές, γεγονός που υποδηλώνει ότι το χαρακτηριστικό είναι πραγματικά επιβλαβές για την απόδοση του μοντέλου.

#### Σημαντικότητα βάση τυχαίων δασών

Η σημασία των χαρακτηριστικών μπορεί να μετρηθεί με διάφορες τεχνικές, αλλά μια από τις πιο δημοφιλείς είναι ο ταξινομητής τυχαίου δάσους. Χρησιμοποιώντας τον αλγόριθμο τυχαίου δάσους, η σημαντικότητα του χαρακτηριστικού μπορεί να μετρηθεί ως ο μέσος όρος και η τυπική απόκλιση της συσσώρευσης της μείωσης της ακαθαρσίας (Mean Decrease in Impurity - MDI) που υπολογίζεται από όλα τα δέντρα απόφασης στο δάσος. Αυτό γίνεται ανεξάρτητα από το αν τα δεδομένα είναι γραμμικά ή μη γραμμικά.



Σχήμα 5.13: Σημαντικότητα χαρακτηριστικών βάση τυχαίων δασών

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

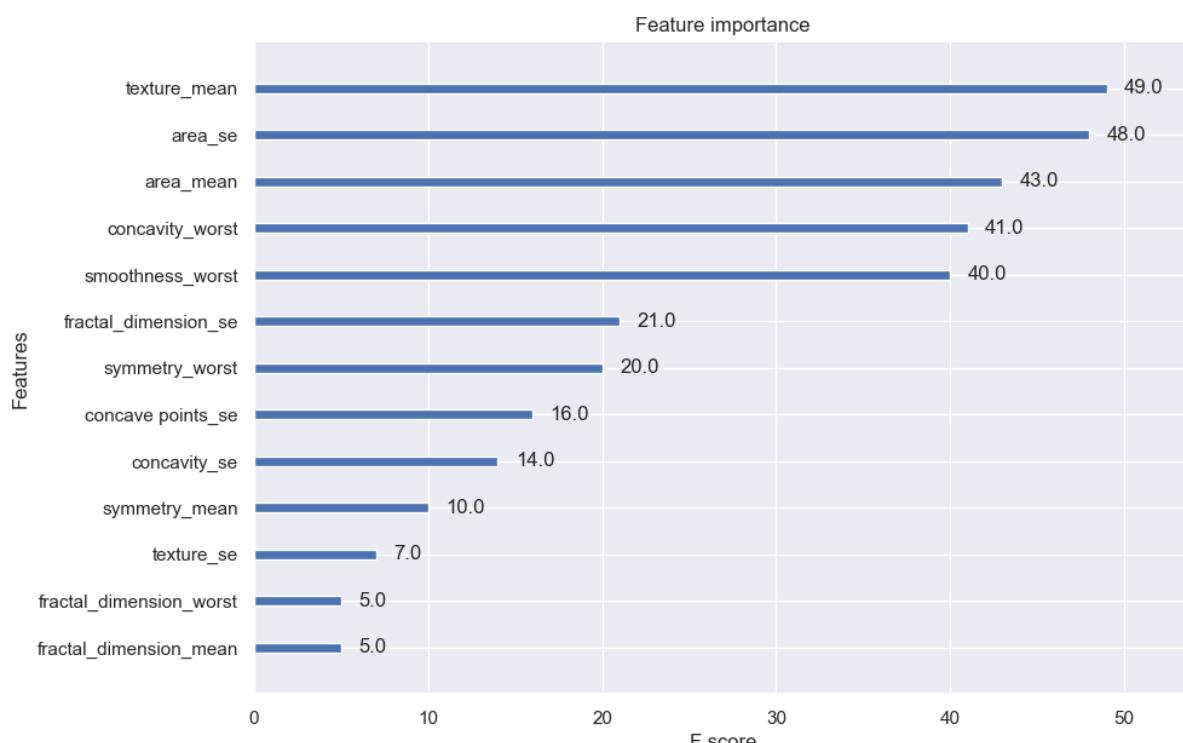
Τα 8 χαρακτηριστικά με το μεγαλύτερο σκορ, όπως φαίνεται και στο σχήμα 5.13, είναι : 'area mean', 'concavity worst', 'area se', 'concave points se', 'smoothness worst', 'concavity se', 'texture mean', 'symmetry worst'.

### Σημαντικότητα βάση του αλγορίθμου XGBoost

Όπως και προηγουμένως, η σημαντικότητα παρέχει μια βαθμολογία που υποδεικνύει πόσο χρησιμό ή πολύτιμο ήταν κάθε χαρακτηριστικό στην κατασκευή των ενισχυμένων δέντρων απόφασης εντός του μοντέλου. Όσο περισσότερο χρησιμοποιείται ένα χαρακτηριστικό για τη λήψη βασικών αποφάσεων με τα δέντρα αποφάσεων, τόσο υψηλότερη είναι η σχετική του σημασία.

Η σπουδαιότητα υπολογίζεται για ένα μεμονωμένο δέντρο αποφάσεων με το ποσό που κάθε σημείο διαχωρισμού χαρακτηριστικών βελτιώνει το μέτρο απόδοσης, σταθμισμένο με τον αριθμό των παρατηρήσεων για τις οποίες είναι υπεύθυνος ο κόμβος. Το μέτρο απόδοσης μπορεί να είναι η καθαρότητα (δείκτης Gini) που χρησιμοποιείται για την επιλογή των σημείων διαχωρισμού ή μια άλλη πιο συγκεκριμένη συνάρτηση σφάλματος. Στη συνέχεια, οι συντελεστές εισαγωγής χαρακτηριστικών υπολογίζονται κατά μέσο όρο σε όλα τα δέντρα απόφασης εντός του μοντέλου.

Η βιβλιοθήκη XGBoost παρέχει μια ενσωματωμένη συνάρτηση για την απεικόνιση των χαρακτηριστικών ταξινομημένων ανάλογα με τη σημασία τους, η οποία ονομάζεται `plot_importance()`. Σύμφωνα με το σχήμα 5.14, τα 8 χαρακτηριστικά με το μεγαλύτερο σκορ είναι : 'texture mean', 'area se', 'area mean', 'concavity worst', 'smoothness worst', 'fractal dimension se', 'symmetry worst', 'concave points se'



Σχήμα 5.14: Σημαντικότητα χαρακτηριστικών βάση του αλγορίθμου XGBoost

### 5.3.5 Ελάχιστος πλεονασμός και μέγιστη συνάφεια (mRMR)

Όταν χρησιμοποιείται το mRMR, ουσιαστικά απαιτείται να γίνει μόνο μία επιλογή: να αποφασιστεί ο αριθμός των χαρακτηριστικών που θα διατηρηθούν. Στην συγκεκριμένη περίπτωση επιλέχθηκε και πάλι  $K = 8$ .

Σε πραγματικές εφαρμογές, μπορεί κανείς να επιλέξει το  $K$  με βάση τη γνώση του τομέα ή άλλους περιορισμούς, όπως η χωρητικότητα του μοντέλου, η μνήμη της μηχανής ή ο διαθέσιμος χρόνος.

Το mRMR λειτουργεί επαναληπτικά. Σε κάθε επανάληψη, εντοπίζει το καλύτερο χαρακτηριστικό (σύμφωνα με έναν κανόνα) και το προσθέτει στο "καλάθι" των επιλεγμένων χαρακτηριστικών. Από τη στιγμή που ένα χαρακτηριστικό μπαίνει στο "καλάθι", δεν μπορεί ποτέ να βγει.

Υποθέτοντας ότι υπάρχουν συνολικά  $m$  χαρακτηριστικά, και για ένα δεδομένο χαρακτηριστικό  $X_i (i \in 1, 2, \dots, m)$ , η σπουδαιότητα του χαρακτηριστικού με βάση το κριτήριο mRMR μπορεί να εκφραστεί ως εξής [82] :

$$f_{mRMR}(X_i) = I(Y, X_i) - \frac{1}{|S|} \sum_{X_s \in S} I(X_s, X_i)$$

όπου  $Y$  είναι η μεταβλητή απόκρισης (επικέτα κλάσης),  $S$  είναι το σύνολο των επιλεγμένων χαρακτηριστικών,  $|S|$  είναι το μέγεθος του συνόλου χαρακτηριστικών (αριθμός χαρακτηριστικών),  $X_s \in S$  είναι ένα χαρακτηριστικό από το σύνολο χαρακτηριστικών  $S$ , το  $X_i$  δηλώνει ένα χαρακτηριστικό που δεν έχει επιλεγεί:  $X_i \notin S$ . Η συνάρτηση  $I(\cdot, \cdot)$  είναι η αμοιβαία πληροφορία όπως αναφέρθηκε πιο πάνω στην υποενότητα 5.3.1.

Κατά τη διαδικασία επιλογής χαρακτηριστικών mRMR, σε κάθε βήμα, το χαρακτηριστικό  $\max_{X_i \notin S} f_{mRMR}(X_i)$  με την υψηλότερη βαθμολογία σημαντικότητας χαρακτηριστικών θα προστίθεται στο επιλεγμένο σύνολο χαρακτηριστικών  $S$ .

Τα 8 χαρακτηριστικά που επιλέχθηκαν από τον αλγόριθμο είναι : 'area mean', 'fractal dimension worst', 'concavity worst', 'area se', 'smoothness worst', 'texture mean', 'concave points se', 'symmetry worst'.

### 5.3.6 Σύνοψη Επιλογής Χαρακτηριστικών

Στον πίνακα 5.1, φαίνονται τα χαρακτηριστικά που έχουν επιλεχθεί από την κάθε μέθοδο επιλογής χαρακτηριστικών. Τελικά, αυτά που θα χρησιμοποιηθούν, είναι μόνο όσα έχουν ψηφιστεί από όλες τις μεθόδους που χρησιμοποιήθηκαν. Αυτά είναι τα εξής 7: 'concavity worst', 'area mean', 'area se', 'concave points se', 'smoothness worst', 'texture mean', 'symmetry worst'.

Χαρακτηριστικά / Μέθοδοι επιλογής							
MI (φθίνουσα)	K-Best	RFE	RFECV	RF FI	XGB FI	MRMR	Σύνολο
concavity worst	✓	✓	✓	✓	✓	✓	6/6
area mean	✓	✓	✓	✓	✓	✓	6/6
area se	✓	✓	✓	✓	✓	✓	6/6
concavity se	-	✓	✓	✓	-	-	3/6
concave points se	✓	✓	✓	✓	✓	✓	6/6
smoothness worst	✓	✓	✓	✓	✓	✓	6/6
texture mean	✓	✓	✓	✓	✓	✓	6/6
fractal dimension worst	-	-	✓	-	-	✓	2/6
symmetry worst	✓	✓	✓	✓	✓	✓	6/6
symmetry mean	✓	-	✓	-	-	-	2/6
fractal dimension se	-	-	✓	-	✓	-	2/6
fractal dimension mean	-	-	✓	-	-	-	1/6
texture se	-	-	-	-	-	-	0/6

Πίνακας 5.1: Πίνακας επιλογής χαρακτηριστικών

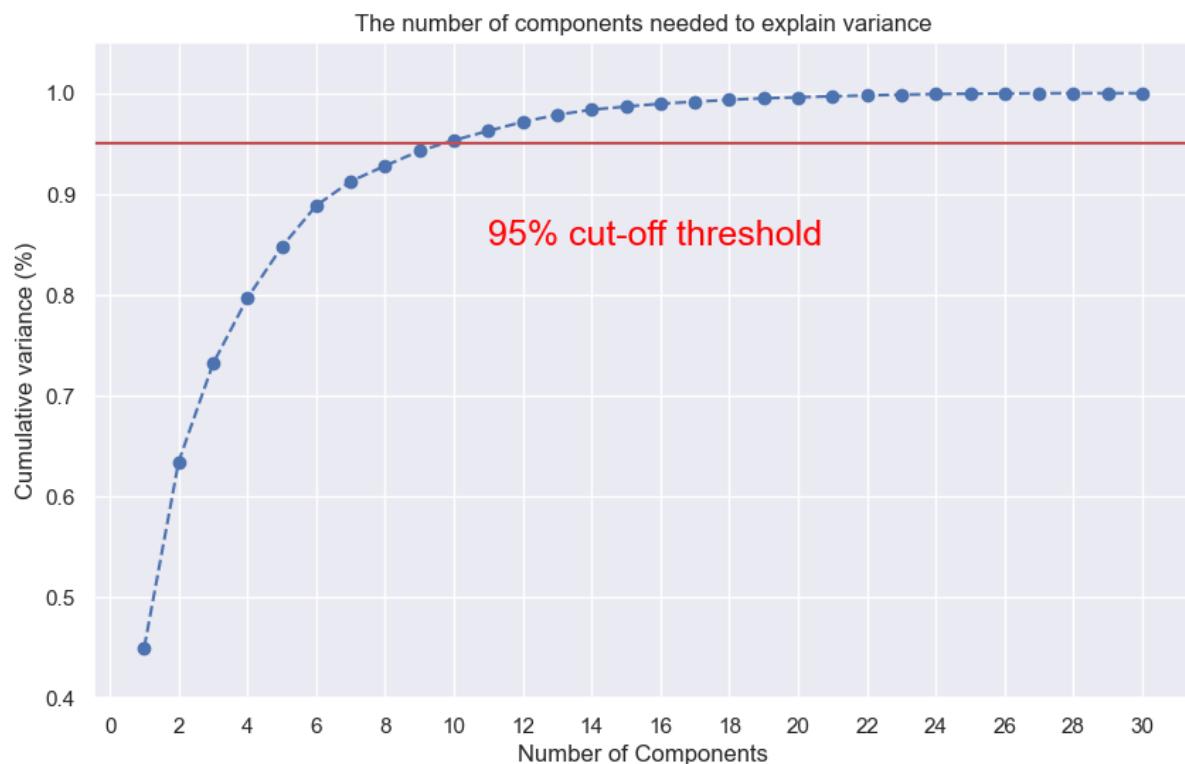
## 5.4 ΑΝΑΛΥΣΗ ΚΥΡΙΩΝ ΣΥΝΙΣΤΩΣΩΝ

Όπως αναφέρθηκε και στο υποκεφάλαιο 4.6, η ανάλυση κύριων συνιστωσών είναι μια τεχνική μείωσης των διαστάσεων και αποσυσχέτισης, που μετασχηματίζει μια συσχετισμένη πολυμεταβλητή κατανομή σε ορθογώνιους γραμμικούς συνδυασμούς των αρχικών μεταβλητών.

Ο σκοπός για τον οποίο χρησιμοποιείται στην παρούσα εργασία, είναι διότι όπως φάνηκε, τα χαρακτηριστικά που υπάρχουν στο συγκεκριμένο σύνολο δεδομένων έχουν πολύ υψηλή συσχέτιση μεταξύ τους. Εκτός αυτού, στόχος είναι να παρθεί όσο το δυνατόν περισσότερη πληροφορία με τα λιγότερα δυνατά χαρακτηριστικά.

### Επιλογή διαστάσεων βάσει της αθροιστικής διακύμανσης

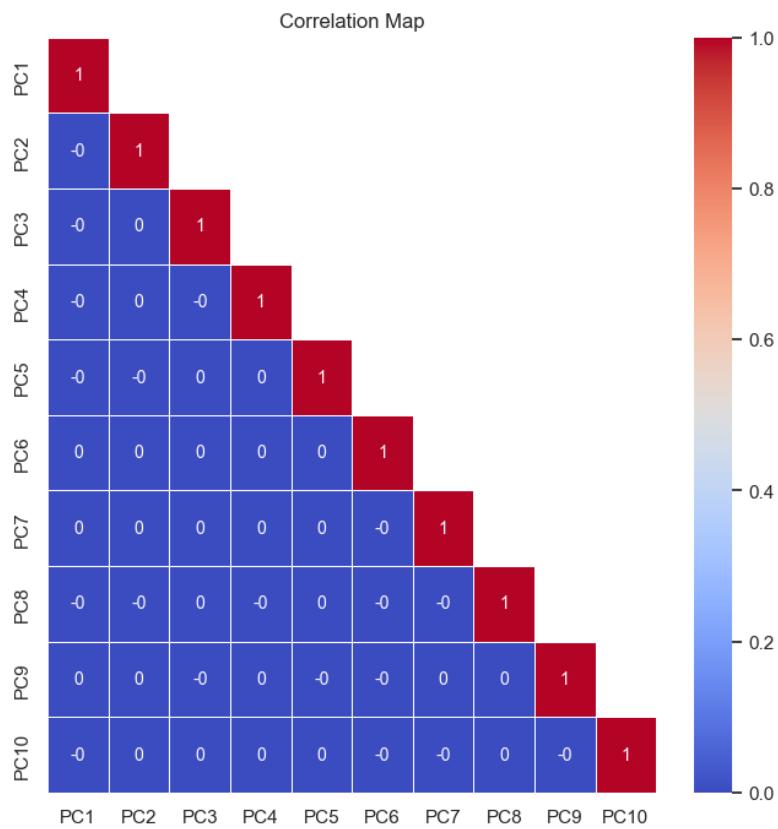
Προκειμένου να επιλεγεί ο αριθμός των κύριων συνιστωσών (διαστάσεων) που θα χρησιμοποιηθούν, τέθηκε ένα κατώτατο όριο ως προς την αθροιστική διακύμανση, δηλαδή το ποσοστό της πληροφορίας που θα εξαχθεί από τα 30 αρχικά χαρακτηριστικά. Αυτό το κατώφλι επιλέχθηκε να είναι το 95% της πληροφορίας, το οποίο όπως φαίνεται και στο σχήμα 5.15 αντιστοιχεί σε **10 κύριες συνιστώσες**.



Σχήμα 5.15: Επιλογή διαστάσεων βάσει της αθροιστικής διακύμανσης

### Αποσυσχέτιση

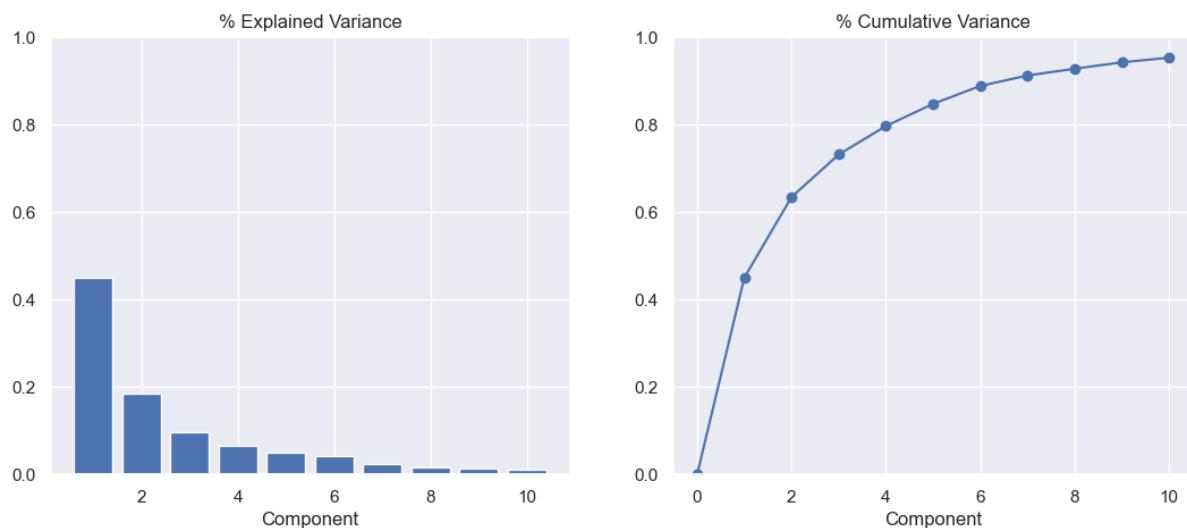
Στο σχήμα 5.16 αποδεικνύεται η αποσυσχέτιση που προκαλείται από την ανάλυση αυτή. Φαίνεται ότι οι κύριες συνιστώσες που επιλέχθηκαν, έχουν γραμμική συσχέτιση μεταξύ τους ίση με 0.



Σχήμα 5.16: Αποσυσχέτιση

### Εξηγούμενη Διακύμανση

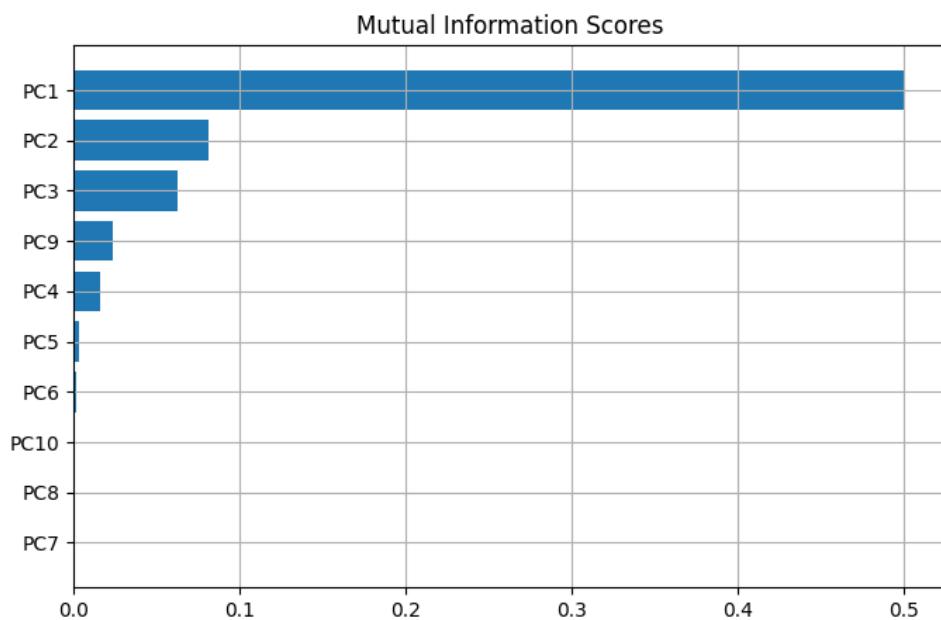
Στο σχήμα 5.17 φαίνεται ότι η πρώτη κύρια συνιστώσα περιέχει την περισσότερη πληροφορία και ακολουθούν οι υπόλοιπες κατά φθίνουσα σειρά. Επίσης, παρατηρείται ότι από ένα σημείο και μετά, η συνεισφορά πληροφορίας είναι ελάχιστη.



Σχήμα 5.17: Εξηγούμενη διακύμανση των κύριων συνιστωσών που επιλέχθηκαν

### Αμοιβαία Πληροφορία

Το σχήμα 5.18 εξηγεί ότι η πρώτη κύρια συνιστώσα, που έχει το μεγαλύτερο MI σκορ, είναι η πιο ενδεικτική στο να προβλέψει την διάγνωση. Αυτό είναι λογικό, καθώς όπως ειπώθηκε περιέχει την περισσότερη πληροφορία σε σχέση με τις υπόλοιπες.



Σχήμα 5.18: Αμοιβαία πληροφορία των κύριων συνιστωσών που επιλέχθηκαν

## 5.5 ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ

---

Σε αυτό το σημείο θα παρουσιαστούν τα αποτελέσματα των μοντέλων μηχανικής μάθησης για κάθε μία από αυτές τις μεθόδους επιλογής ή εξαγωγής χαρακτηριστικών.

- Όλα τα χαρακτηριστικά (ML All Features.ipynb)
- Τα 7 επιλεγμένα χαρακτηριστικά (ML Selected Features.ipynb)
- Χαρακτηριστικά που εξήχθησαν από την PCA (ML PCA.ipynb)

Προκειμένου να εξασφαλιστεί μια όσο το δυνατόν δικαιότερη σύγκριση, χρησιμοποιήθηκαν οι ίδιες υπερπαράμετροι στο πλέγμα αναζήτησης παραμέτρων και για τα τρία σύνολα χαρακτηριστικών. Επίσης, ο αλγόριθμος MLP βελτιστοποιήθηκε με δοκιμές (trial and error), επομένως δεν υπάρχει μέτρηση χρόνου στο σχήμα 5.20. Όμως να σημειωθεί ότι αυτή η διαδικασία διήρκεσε πολύ παραπάνω από οποιονδήποτε άλλο χρόνο φαίνεται στο σχήμα αυτό. Ο χρόνος εκτέλεσης των αλγορίθμων είναι ουσιαστικά ο χρόνος που χρειάστηκε ο κάθε αλγόριθμος στην διαδικασία της διασταυρούμενης επικύρωσης και συνεπώς, για τους αλγορίθμους με τις ρυθμισμένες παραμέτρους, ο χρόνος που χρειάστηκε και ο εσωτερικός βρόχος διασταυρούμενης επικύρωσης για την βελτιστοποίηση των παραμέτρων.

Τα αποτελέσματα των αλγορίθμων στις προεπιλεγμένες παραμέτρους που φαίνονται στο σχήμα 5.19 δείχνουν ότι:

- Οι βαθμολογίες F1 για όλους τους αλγορίθμους κυμαίνονται από 0,915 έως 0,981 με χρόνους υπολογισμού που κυμαίνονται από 0,062 έως 4,628 δευτερόλεπτα.
- Οι γραμμικοί ταξινομητές Ridge, LDA και GNB χρατάνε σταθερή την απόδοσή τους με τα μειωμένα χαρακτηριστικά, καθώς είναι πολύ κοντά σε F1 score και χρόνο εκτέλεσης. Παρατηρείται ότι οι Ridge και LDA πέτυχαν τα ίδια σκορ με τα 7 και 10 χαρακτηριστικά ενώ ήταν και αρκετά κοντά με όλα τα χαρακτηριστικά.
- Το F1 σκορ των 4 από τους 5 αλγόριθμους που σχετίζονται με τα δέντρα απόφασης (LGBM, Random Forest, XGBoost και Decision Tree) φαίνεται να μειώθηκε όταν εκπαιδεύτηκαν με τα 7 χαρακτηριστικά. Αυτό λογικά συμβαίνει γιατί χάθηκε αρκετή πληροφορία μειώνοντας τα χαρακτηριστικά, ενώ με τα χαρακτηριστικά από την PCA οι LGBM και Decision Tree είχαν βελτιωμένα αποτελέσματα.
- Ο ταξινομητής AdaBoost, ο οποίος παίρνει σαν βασικό εκτιμητή των ταξινομητή δέντρων που εκπαιδεύτηκε στο εκάστοτε σετ χαρακτηριστικών, φαίνεται να πετυχαίνει καλύτερα αποτελέσματα με τα αδύναμα δέντρα αποφάσεων όπως περιγράφηκε και στη θεωρία.

## 5.5. ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ

---

- Τα 7 επιλεγμένα χαρακτηριστικά και τα 10 χαρακτηριστικά από την PCA οδηγούν γενικά σε ταχύτερους χρόνους υπολογισμού.
- Η χρήση των 10 χαρακτηριστικών από την PCA τείνει να οδηγεί σε καλύτερες επιδόσεις όσον αφορά το σκορ F1, με τις καλύτερες επιδόσεις να επιτυγχάνονται από τους αλγορίθμους SVC (0.981 σε 0,093 sec) και MLP (0.981 σε 3,7 sec).

Τα αποτελέσματα των αλγορίθμων στις ρυθμισμένες παραμέτρους που φαίνονται στο σχήμα 5.20 δείχνουν ότι:

- Οι βαθμολογίες F1 για όλους τους αλγορίθμους κυμαίνονται από 0,927 έως 0,983 με χρόνους υπολογισμού που κυμαίνονται από 3,455 έως 352.877 δευτερόλεπτα.
- Οι γραμμικοί ταξινομητές Ridge και LDA κρατάνε σταθερή την απόδοσή τους με τα μειωμένα χαρακτηριστικά, καθώς είναι πολύ κοντά σε F1 score και χρόνο εκτέλεσης. Ο GNB φαίνεται να πήγε καλύτερα με τα χαρακτηριστικά από την PCA. Επίσης, παρατηρείται ότι οι Ridge και LDA έχουν ακριβώς τα ίδια σκορ για όλα τα σετ χαρακτηριστικών, αλλά ο LDA είναι αρκετά πιο γρήγορος.
- Το F1 σκορ των αλγορίθμων που σχετίζονται με τα δέντρα απόφασης (LGBM, Random Forest, XGBoost και Decision Tree) αυτήν την φορά φαίνεται να βελτιώθηκαν με την χρήση των 7 χαρακτηριστικών ενώ το αντίθετο συνέβη με τα χαρακτηριστικά από το PCA.
- Ο ταξινομητής AdaBoost, φαίνεται να τα πήγε και πάλι καλύτερα με το αδύναμο δέντρο απόφασης. Παρόλο που στα δέντρα απόφασης το σκορ ήταν ίδιο στην εκπαίδευση με όλα τα χαρακτηριστικά και με αυτά από την PCA, αυτό με όλα τα χαρακτηριστικά πήγε αρκετά καλύτερα, γεγονός που μπορεί να αποδοθεί στην τυχαία αναζήτηση παραμέτρων.
- Τα 7 επιλεγμένα χαρακτηριστικά και τα 10 χαρακτηριστικά από την PCA οδηγούν γενικά σε ταχύτερους χρόνους υπολογισμού.
- Η χρήση των 10 χαρακτηριστικών από την PCA τείνει να οδηγεί σε καλύτερες επιδόσεις όσον αφορά το σκορ F1, με τις καλύτερες επιδόσεις να επιτυγχάνονται από τους αλγορίθμους SVC (0.981 σε 9,945 sec) και MLP (0.981 και 0.983 με δοκιμές). Αξιοσημείωτη είναι και η βελτίωση των αλγορίθμων SGD (0.979 σε 4,961 sec) και KNN (0.979 σε 87,843).

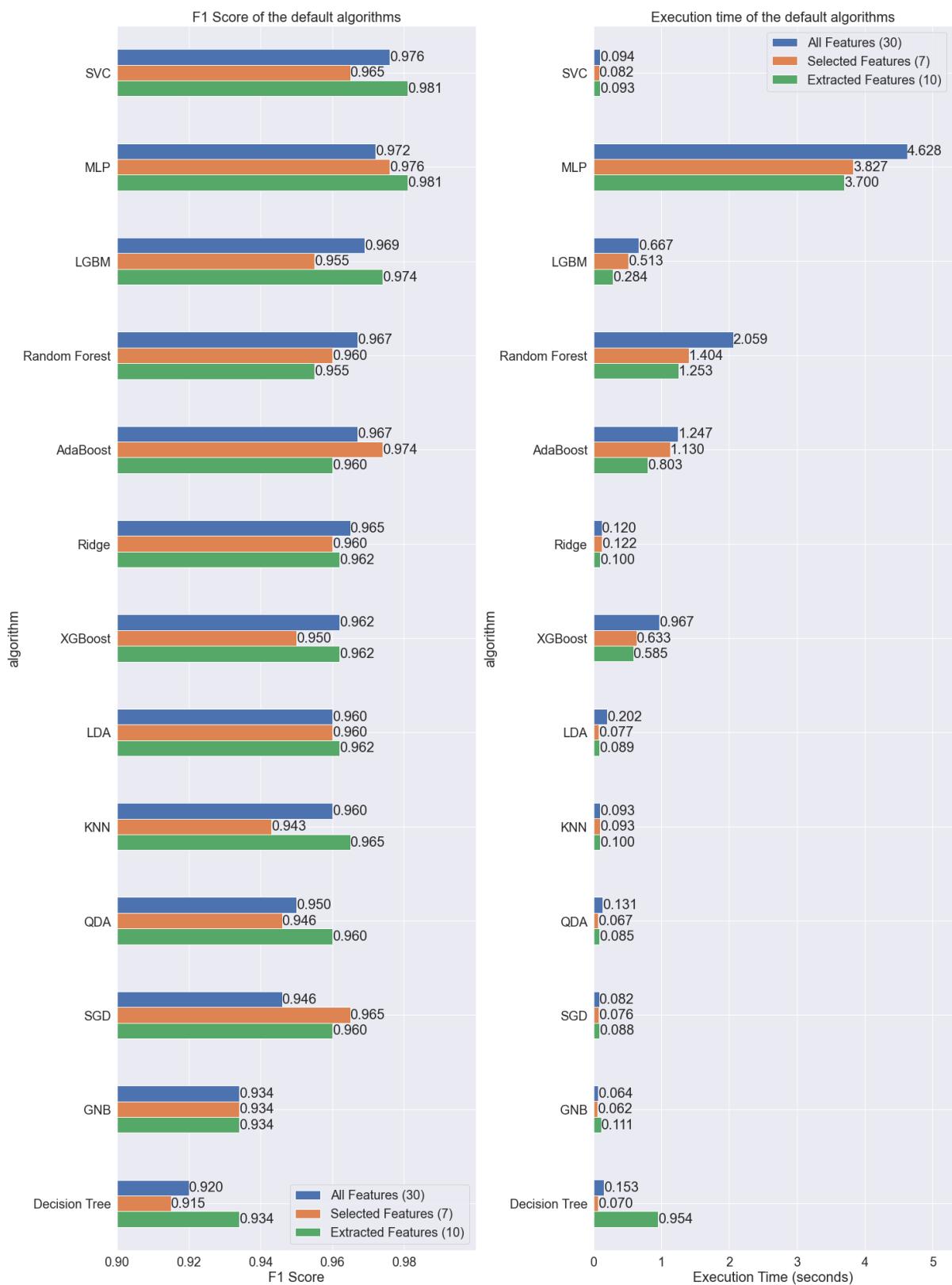
## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ

---

Προκειμένου να δειχθεί η βελτίωση (ή όχι) των αλγορίθμων με τις ρυθμισμένες παραμέτρους σε σχέση με αυτούς με τις προεπιλεγμένες, δημιουργήθηκε το σχήμα 5.21. Αυτό έγινε με τον υπολογισμό της διαφοράς των αποτελεσμάτων F1 και των χρόνων εκτέλεσης μεταξύ των δύο ομάδων αλγορίθμων. Από το σχήμα αυτό βγαίνουν τα εξής συμπεράσματα:

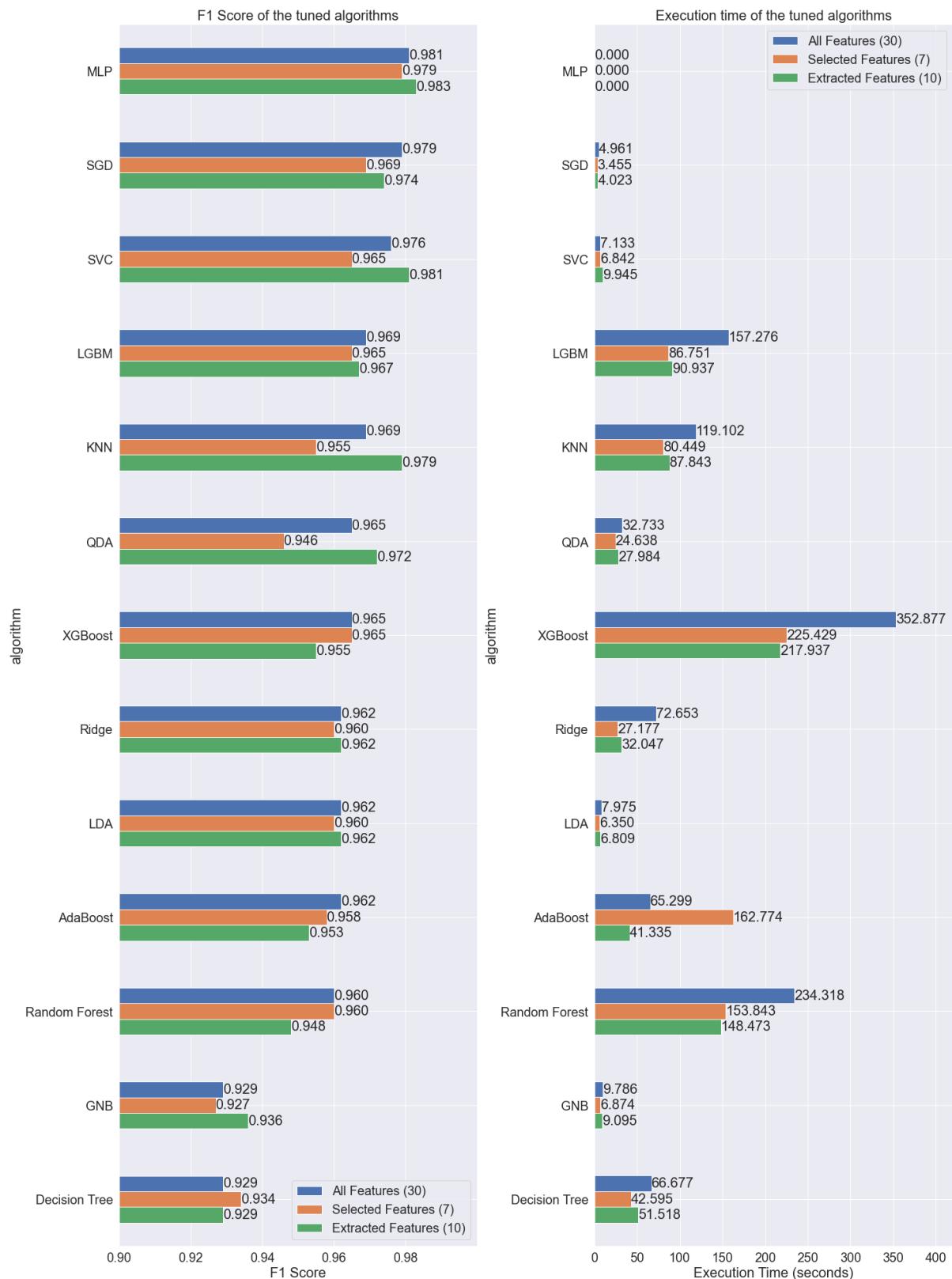
- Δεν βελτιώθηκαν όλοι οι αλγόριθμοι, αντίθετα κάποιοι έριξαν την απόδοσή τους. Συγκεκριμένα ο AdaBoost έριξε την απόδοσή του σε όλα τα σύνολα χαρακτηριστικών. Τα Τυχαία Δάση και ο GNB έριξαν την απόδοσή τους στα 2 από τα 3 σύνολα χαρακτηριστικών.
- Εδώ παρατηρείται ξεκάθαρα το πόσο επηρεάζει η αύξηση της απόδοσης του Δέντρου Αποφάσεων στην μείωση του σκορ του AdaBoost.
- Οι χρόνοι υπολογισμού αυξήθηκαν πάρα πολύ, ιδίως σε μερικές περιπτώσεις όπως στον AdaBoost, στα Δέντρα Απόφασης, στο KNN, στα Τυχαία Δάση και στον XGBoost.
- Οι αλγόριθμοι KNN, MLP και SGD αύξησαν την απόδοσή τους σε όλα τα σύνολα χαρακτηριστικών. Το Δέντρο Αποφάσεων, ο QDA και ο XGBoost βελτιώθηκαν στα 2 από τα 3 σύνολα χαρακτηριστικών.
- Μεγάλη βελτίωση πέτυχε ο αλγόριθμος SGD με όλα τα χαρακτηριστικά (+0.033 με +4,879 sec), αλλά και τα Δέντρα Απόφασης στα 7 επιλεγμένα χαρακτηριστικά (+0.019 με +42,525 sec).
- Ελάχιστα μεταβλήθηκε το σκορ των LDA (+0.002) και Ridge (-0.003) όταν εκπαιδεύτηκαν με όλα τα χαρακτηριστικά, ενώ καθόλου δεν μεταβλήθηκε ο αλγόριθμος SVC.

## 5.5. ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ



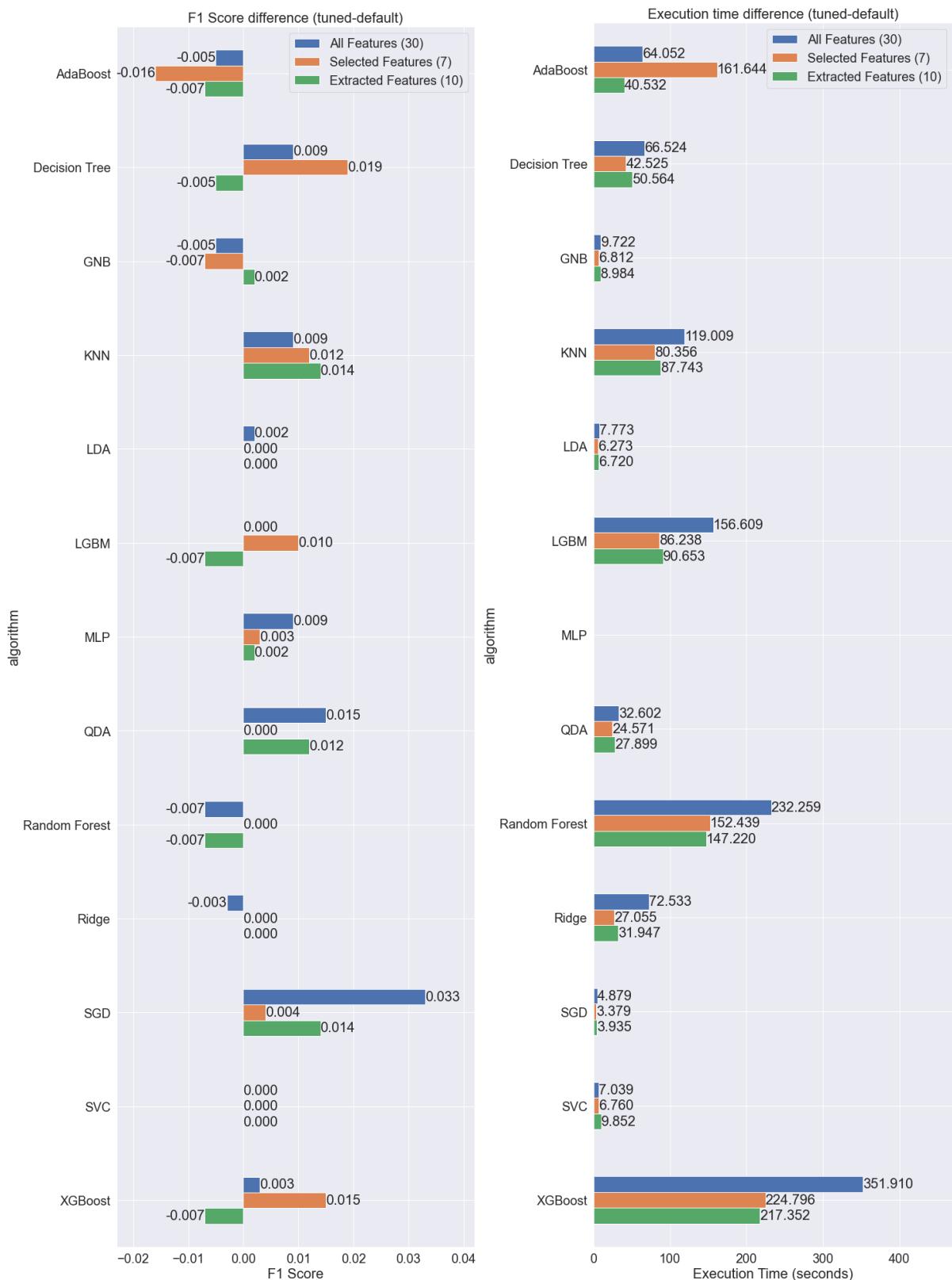
Σχήμα 5.19: F1-Score και ο χρόνος εκτέλεσης των αλγορίθμων στις προεπιλεγμένες παραμέτρους

## ΚΕΦΑΛΑΙΟ 5. ΥΛΟΠΟΙΗΣΗ



Σχήμα 5.20: F1-Score και ο χρόνος εκτέλεσης των αλγορίθμων στις ρυθμισμένες παραμέτρους

## 5.5. ΑΠΟΤΕΛΕΣΜΑΤΑ ΤΩΝ ΑΛΓΟΡΙΘΜΩΝ



Σχήμα 5.21: Η διαφορά του F1-Score και του χρόνου εκτέλεσης μεταξύ των αλγορίθμων στις ρυθμισμένες και στις προεπιλεγμένες παραμέτρους

# 6

## Συμπεράσματα, Προβλήματα & Μελλοντικές Επεκτάσεις

### 6.1 ΓΕΝΙΚΑ ΣΥΜΠΕΡΑΣΜΑΤΑ

---

Παρατηρείται ότι οι περισσότεροι αλγόριθμοι τα πήγαν καλύτερα στο F1 score με τα χαρακτηριστικά που εξήχθησαν από το PCA. Αυτό συμβαίνει διότι σε 10 μόλις χαρακτηριστικά από τα 30, εμπεριέχεται περίπου το 95.33% της συνολικής πληροφορίας και έτσι οι αλγόριθμοι μάθαιναν πιο εύκολα τα "μοτίβα" της ταξινόμησης.

Επίσης, σημαντικά αποτελέσματα, κυρίως όσον αφορά το χρόνο εκτέλεσης, πέτυχαν και οι αλγόριθμοι με τα 7 επιλεγμένα χαρακτηριστικά καθώς στις περισσότερες περιπτώσεις ήταν πιο γρήγοροι από τα άλλα 2 σύνολα χαρακτηριστικών.

Γενικότερα τις καλύτερες επιδόσεις φαίνεται να τις πέτυχαν οι μη γραμμικοί ταξινομητές και συγκεκριμένα οι MLP, SVC, KNN, SGD και QDA. Αντίθετα οι γραμμικοί ταξινομητές, δηλαδή οι Ridge, LDA και GNB, είχαν κακές επιδόσεις, όμως είχαν αρκετά μικρό χρόνο εκτέλεσης. Οι αλγόριθμοι που σχετίζονται με τα δέντρα απόφασης, δηλαδή οι LGBM, XGboost, AdaBoost, Random Forest και Decision Tree, είχαν σχετικά μέτρια αποτελέσματα αλλά πολύ μεγάλους χρόνους εκτέλεσης.

Συνοπτικά, τα μεγαλύτερα σκορ F1 τα πέτυχαν οι εξής αλγόριθμοι στα χαρακτηριστικά από την PCA:

#### 1. MLP

- 0.983 και βελτιστοποίηση με δοκιμές.
- 0.981 σε 3,7 δευτερόλεπτα με προεπιλεγμένες παραμέτρους.

#### 2. SVC

- 0.981 σε 0,093 δευτερόλεπτα με προεπιλεγμένες παραμέτρους.
- 0.981 σε 9,945 δευτερόλεπτα με ρυθμισμένες παραμέτρους.

Συμπερασματικά, τα αποτελέσματα των πειραμάτων έδειξαν ότι ο MLP είχε την υψηλότερη βαθμολογία F1, καθιστώντας τον, τον αλγόριθμο με την καλύτερη απόδοση από άποψη ακρίβειας. Ωστόσο, αυτό είχε ως κόστος τον μεγαλύτερο υπολογιστικό χρόνο λόγω της βελτιστοποίησης με δοκιμές. Από την άλλη πλευρά, ο SVC είχε ελαφρώς χαμηλότερη βαθμολογία F1, αλλά ήταν πολύ πιο αποδοτικός με υπολογιστικό χρόνο μόλις 0,093 δευτερόλεπτα.

Παρά ταύτα, όταν πρόκειται για πραγματικές εφαρμογές στο πλαίσιο της διάγνωσης του καρκίνου του μαστού, η ακρίβεια είναι υψηλή σημασίας. Έτσι, το MLP εξακολουθεί να είναι η προτιμώμενη επιλογή, καθώς κάθε μικρή βελτίωση στο σκορ F1 μπορεί ενδεχομένως να σώσει ζωές. Πρόκειται για έναν συμβιβασμό μεταξύ υπολογιστικής απόδοσης και ακρίβειας, αλλά σε αυτό το σενάριο, ένα υψηλότερο σκορ F1 αξίζει τον επιπλέον χρόνο που δαπανάται για τον συντονισμό.

## 6.2 ΠΡΟΒΛΗΜΑΤΑ

---

Μερικά από τα προβλήματα που υπήρξαν στην εργασία είναι :

- Έλλειψη υπολογιστικών πόρων:

Η έλλειψη ισχυρού προσωπικού υπολογιστή και ειδικής κάρτας γραφικών δυσχέραινε την εργασία με μεγαλύτερα σύνολα δεδομένων και την εκπαίδευση πιο σύνθετων μοντέλων μηχανικής μάθησης. Αυτός ο περιορισμός επηρέασε επίσης τη δυνατότητα εκτέλεσης πιο προηγμένων τεχνικών επιλογής χαρακτηριστικών και ενδεχομένως βελτίωσης των αποτελεσμάτων.

- Περιορισμένο μέγεθος και ποικιλομορφία δεδομένων:

Το μικρό μέγεθος του συνόλου δεδομένων εμπόδισε τη χρήση πιο προηγμένων μοντέλων βαθιάς μάθησης (δηλαδή πιο σύνθετων νευρωνικών δικτύων) που απαιτούν περισσότερα δεδομένα για να εκπαιδευτούν αποτελεσματικά. Επιπλέον, η έλλειψη ποικιλομορφίας στα δεδομένα, όσον αφορά τους διαφορετικούς υποτύπους καρκίνου του μαστού και τα δημογραφικά στοιχεία των ασθενών, περιόρισε επίσης τη δυνατότητα γενίκευσης των αποτελεσμάτων της μελέτης. Επιπλέον, λόγω του περιορισμένου όγκου δεδομένων, τα χαρακτηριστικά δεν διαχωρίστηκαν στην αρχή της μελέτης. Ωστόσο, η διαδικασία επιλογής χαρακτηριστικών στο πλαίσιο ενός βρόχου διασταυρούμενης επικύρωσης ήταν αρκετά πολύπλοκη και δεν υλοποιήθηκε, με αποτέλεσμα να μπορεί να θεωρηθεί σε ένα βαθμό μεροληπτική.

- Υψηλή διαστατικότητα των δεδομένων:

Η υψηλή διαστατικότητα των δεδομένων είχε ως αποτέλεσμα την ανάγκη για τεχνικές επιλογής χαρακτηριστικών για τη μείωση του αριθμού των χαρακτηριστικών που χρησιμοποιήθηκαν στην ανάλυση.

- Υψηλή συσχέτιση μεταξύ ορισμένων χαρακτηριστικών και η φύση τους:

Η υψηλή συσχέτιση μεταξύ ορισμένων χαρακτηριστικών καθώς και η φύση τους (δηλαδή τα χαρακτηριστικά ήταν κάποιες διαστάσεις του κυτταρικού

## ΚΕΦΑΛΑΙΟ 6. ΣΥΜΠΕΡΑΣΜΑΤΑ, ΠΡΟΒΛΗΜΑΤΑ & ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

---

πυρήνα και όχι χαρακτηριστικά που δεν απαιτούν κάποια ιατρική γνώση, όπως π.χ. η ηλικία του ασθενή) κατέστησε δύσκολη την κατανόηση της σχέσης μεταξύ των χαρακτηριστικών και της ταξινόμησης του καρκίνου του μαστού.

### 6.3 ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

---

Στην παρούσα εργασία, μελετήθηκε μια προσέγγιση μηχανικής μάθησης για την ταξινόμηση του καρκίνου του μαστού με τη χρήση δειγμάτων αναρρόφησης με λεπτή βελόνα (FNA). Η προσέγγιση αξιολογήθηκε χρησιμοποιώντας ένα σύνολο δεδομένων από δείγματα FNA και τα αποτελέσματα έδειξαν ότι τα μοντέλα μηχανικής μάθησης μπορούν να επιτύχουν καλές επιδόσεις για την ταξινόμηση του καρκίνου του μαστού. Ωστόσο, υπάρχουν διάφοροι τομείς για μελλοντική εργασία που μπορεί να γίνει για τη βελτίωση της απόδοσης των μοντέλων και την αντιμετώπιση των περιορισμών της παρούσας μελέτης. Μερικές ιδέες είναι οι παρακάτω:

- Η ανάπτυξη ενός φορητού συστήματος ανίχνευσης καρκίνου του μαστού. Το σύστημα αυτό θα περιλαμβάνει μια συσκευή, την οποία οι γυναίκες θα μπορούν να χρησιμοποιούν για τη λήψη εικόνων του στήθους τους, η οποία θα μπορούσε να περιγραφεί σαν αυτοελεγχόμενη μαστογραφία. Η συσκευή θα ενσωματώνει αλγορίθμους μηχανικής μάθησης για την ανάλυση των εικόνων και την παροχή της πιθανότητας να υπάρχει ή να υπάρχει καρκινικός όγκος. Αυτό θα επέτρεπε την έγκαιρη ανίχνευση και την αύξηση της προσβασιμότητας στον προσυμπτωματικό έλεγχο του καρκίνου του μαστού, ιδίως σε περιοχές με περιορισμένους ιατρικούς πόρους. Ωστόσο, θα απαιτηθεί σημαντική έρευνα, ανάπτυξη και επικύρωση προτού ένα τέτοιο σύστημα καταστεί διαθέσιμο για ευρεία χρήση.
- Η ενσωμάτωση πιο σύνθετων μοντέλων μηχανικής μάθησης, όπως αρχιτεκτονικές βαθιάς μάθησης (νευρωνικά δίκτυα συνελίξεων, αναδρομικά νευρωνικά δίκτυα κ.λπ.), τα οποία έχουν αποδειχθεί αποτελεσματικά για εργασίες ταξινόμησης εικόνων (μαστογραφίες). Αυτό θα μπορούσε να γίνει με τη χρήση περισσότερων δεδομένων και ισχυρών υπολογιστικών πόρων, όπως οι GPU, για την εκπαίδευση αυτών των μοντέλων.
- Η διερεύνηση της χρήσης πιο προηγμένων τεχνικών για την επιλογή μοντέλων, όπως η βελτιστοποίηση κατά Bayes. Αυτό θα μπορούσε να βοηθήσει στη βελτιστοποίηση των υπερ-παραμέτρων των μοντέλων και στη βελτίωση της απόδοσής τους. Επιπλέον, μια άλλη μέθοδος συνόλου που δεν χρησιμοποιείται, η στοίβαξη ταξινομητών, θα μπορούσε επίσης να φανεί ωφέλιμη στη βελτίωση της απόδοσης.
- Για να ξεπεραστεί ο περιορισμένος όγκος δεδομένων, μια πιθανή μελλοντική κατεύθυνση θα μπορούσε να είναι η διερεύνηση της χρήσης τεχνικών δημιουργίας συνθετικών δεδομένων, όπως τα GAN (Generative Adversarial Networks), για τη δημιουργία πρόσθετων σημείων δεδομένων που θα μπορούσαν να χρησιμοποιηθούν για την εκπαίδευση των μοντέλων.

### 6.3. ΜΕΛΛΟΝΤΙΚΕΣ ΕΠΕΚΤΑΣΕΙΣ

---

- Η διερεύνηση της χρήσης της μάθησης μεταφοράς (transfer learning) και της προσαρμογής στον τομέα (domain adaptation) για τη βελτίωση των επιδόσεων των μοντέλων μηχανικής μάθησης κατά την εργασία με διαφορετικά σύνολα δεδομένων ή τρόπους απεικόνισης. Αυτό θα μπορούσε να είναι ιδιαίτερα χρήσιμο για τη βελτίωση της γενίκευσης των μοντέλων και την ενίσχυση της ευρωστίας τους σε διαφορετικές κλινικές περιπτώσεις και πληθυσμούς.

Συμπερασματικά, ο τομέας της ανίχνευσης του καρκίνου του μαστού εξελίσσεται συνεχώς και υπάρχει ανάγκη για καινοτομία τόσο στο υλικό όσο και στο λογισμικό για τη βελτίωση της ακρίβειας και της προσβασιμότητας των μεθόδων ανίχνευσης ώστε να βελτιωθεί η ζωή των γυναικών και να σωθούν όσο το δυνατό περισσότερες ζωές.

# Βιβλιογραφία

- [1] John McCarthy. “*What is artificial intelligence*“. URL: <http://www-formal.stanford.edu/jmc/whatisai.html>, 2004.
- [2] Alan M Turing. “*Computing machinery and intelligence*“. In “*Parsing the turing test*“, pages 23–65. Springer, 2009.
- [3] Stuart J Russell. “*Artificial intelligence a modern approach*“. Pearson Education, Inc., 2010.
- [4] World Health Organization. “*Cancer*“. <https://www.who.int/en/news-room/fact-sheets/detail/cancer>. [Online; accessed 10-April-2022].
- [5] Christina Fitzmaurice, Christine Allen, Ryan M Barber, Lars Barregard, Zulfiqar A Bhutta, Hermann Brenner, Daniel J Dicker, Odgerel Chimed-Orchir, Rakhi Dandona, Lalit Dandona, et al. “*Global, regional, and national cancer incidence, mortality, years of life lost, years lived with disability, and disability-adjusted life-years for 32 cancer groups, 1990 to 2015: a systematic analysis for the global burden of disease study*“. *JAMA oncology*, 3(4):524–548, 2017.
- [6] UCI Machine Learning Repository. “*Breast Cancer Wisconsin (Diagnostic) Data Set*“. [https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+\(diagnostic\)](https://archive.ics.uci.edu/ml/datasets/breast+cancer+wisconsin+(diagnostic)). [Online; accessed 25-March-2022].
- [7] J Ross Quinlan. “*Improved use of continuous attributes in C4. 5*“. *Journal of artificial intelligence research*, 4:77–90, 1996.
- [8] Carlos Andrés Pena-Reyes and Moshe Sipper. “*A fuzzy-genetic approach to breast cancer diagnosis*“. *Artificial intelligence in medicine*, 17(2):131–155, 1999.
- [9] Detlef Nauck and Rudolf Kruse. “*Obtaining interpretable fuzzy classification rules from medical data*“. *Artificial intelligence in medicine*, 16(2):149–169, 1999.
- [10] Rudy Setiono. “*Generating concise and accurate classification rules for breast cancer diagnosis*“. *Artificial Intelligence in medicine*, 18(3):205–219, 2000.
- [11] Andreas A Albrecht, Georgios Lappas, Staal A Vinterbo, C Wong, and Lucila Ohno-Machado. “*Two applications of the LSA machine*“. In “*Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02.*“, volume 1, pages 184–189. IEEE, 2002.

- [12] Janos Abonyi and Ferenc Szeifert. “*Supervised fuzzy clustering for the identification of fuzzy classifiers*“. Pattern Recognition Letters, 24(14):2195–2207, 2003.
- [13] Tüba Kiyan and Tülay Yildirim. “*Breast cancer diagnosis using statistical neural networks*“. IU-Journal of Electrical & Electronics Engineering, 4(2):1149–1153, 2004.
- [14] Kemal Polat and Salih Güneş. “*Breast cancer diagnosis using least square support vector machine*“. Digital signal processing, 17(4):694–701, 2007.
- [15] Elif Derya Übeyli. “*Implementing automated diagnostic systems for breast cancer detection*“. Expert systems with Applications, 33(4):1054–1062, 2007.
- [16] Mehmet Fatih Akay. “*Support vector machines combined with feature selection for breast cancer diagnosis*“. Expert systems with applications, 36(2):3240–3247, 2009.
- [17] Yonghong Peng, Zhiqing Wu, and Jianmin Jiang. “*A novel feature selection approach for biomedical data classification*“. Journal of Biomedical Informatics, 43(1):15–23, 2010.
- [18] Gouda I Salama, M Abdelhalim, and Magdy Abd-elghany Zeid. “*Breast cancer diagnosis on three different datasets using multi-classifiers*“. Breast Cancer (WDBC), 32(569):2, 2012.
- [19] W Nick Street, William H Wolberg, and Olvi L Mangasarian. “*Nuclear feature extraction for breast tumor diagnosis*“. In “*Biomedical image processing and biomedical visualization*“, volume 1905, pages 861–870. SPIE, 1993.
- [20] i scoop. “*Artificial intelligence and cognitive computing: AI business guide*“. <https://www.i-scoop.eu/ai-artificial-intelligence-cognitive-computing/>.
- [21] Ahmad Hammoudeh. “*A concise introduction to reinforcement learning*“. Princess Suamaya University for Technology: Amman, Jordan, 2018.
- [22] Ryszard S Michalski, Jaime G Carbonell, and Tom M Mitchell. “*Machine learning: An artificial intelligence approach*“. Springer Science & Business Media, 2013.
- [23] Big Data tips. “*Machine Learning Methods*“. <http://www.big-data.tips/machine-learning-methods>.
- [24] Omar Y Al-Jarrah, Paul D Yoo, Sami Muhaidat, George K Karagiannidis, and Kamal Taha. “*Efficient machine learning for big data: A review*“. Big Data Research, 2(3):87–93, 2015.
- [25] programmersought.com. “*Machine Learning Algorithm (Introduction)*“. <https://www.programmersought.com/article/87971760827/>.
- [26] Aurélien Géron. “*Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow*“. O’Reilly Media, Inc., 2022.

## ΒΙΒΛΙΟΓΡΑΦΙΑ

---

- [27] Harry Zhang. “*The optimality of naive Bayes*“. *Aa*, 1(2):3, 2004.
- [28] Megha Rathi, Arun Kumar Singh, et al. “*Breast cancer prediction using Naïve Bayes classifier*“. *International Journal of Information Technology & Systems*, 1 (2):77–80, 2012.
- [29] Kilian Q. Weinberger. “*Bayes Classifier and Naive Bayes*“. <https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote05.html>, .
- [30] Scikit Learn. “*Linear and Quadratic Discriminant Analysis*“. [https://scikit-learn.org/stable/modules/lda\\_qda.html#lda-qda](https://scikit-learn.org/stable/modules/lda_qda.html#lda-qda), .
- [31] Trevor Hastie, Robert Tibshirani, Jerome H Friedman, and Jerome H Friedman. “*The elements of statistical learning: data mining, inference, and prediction*“, volume 2. Springer, 2009.
- [32] Richard O Duda, Peter E Hart, et al. “*Pattern classification*“. John Wiley & Sons, 2006.
- [33] Scikit Learn. “*Linear Models*“. [https://scikit-learn.org/stable/modules/linear\\_model.html#ridge-regression](https://scikit-learn.org/stable/modules/linear_model.html#ridge-regression), .
- [34] Johan AK Suykens and Joos Vandewalle. “*Least squares support vector machine classifiers*“. *Neural processing letters*, 9(3):293–300, 1999.
- [35] IBM. “*K-Nearest Neighbors Algorithm*“. <https://www.ibm.com/topics/knn>. [Online; accessed 10-November-2022].
- [36] Sebastian Raschka. “*STAT 479: Machine Learning Lecture Notes*“. [https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/02\\_knn\\_notes.pdf](https://sebastianraschka.com/pdf/lecture-notes/stat479fs18/02_knn_notes.pdf), . [Online; accessed 10-November-2022].
- [37] Kilian Q. Weinberger. “*k-Nearest Neighbors / Curse of Dimensionality*“. [https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02\\_kNN.html](https://www.cs.cornell.edu/courses/cs4780/2018fa/lectures/lecturenote02_kNN.html), . [Online; accessed 10-November-2022].
- [38] Christopher M Bishop and Nasser M Nasrabadi. “*Pattern recognition and machine learning*“, volume 4. Springer, 2006.
- [39] Scikit Learn. “*Support Vector Machines*“. <https://scikit-learn.org/stable/modules/svm.html#svm-classification>, .
- [40] Scikit Learn. “*Non-linear SVM*“. [https://scikit-learn.org/stable/auto\\_examples/svm/plot\\_svm\\_nonlinear.html](https://scikit-learn.org/stable/auto_examples/svm/plot_svm_nonlinear.html), .
- [41] Leo Breiman, Jerome H Friedman, Richard A Olshen, and Charles J Stone. “*Classification and regression trees. Belmont, CA: Wadsworth*“. International Group, 432(151-166):9, 1984.
- [42] Scikit Learn. “*Decision Trees*“. <https://scikit-learn.org/stable/modules/tree.html#tree>, .

- [43] Mohamed Hanafy and Ruixing Ming. “*Machine learning approaches for auto insurance big data*“. Risks, 9(2):42, 2021.
- [44] Leo Breiman. “*Pasting small votes for classification in large databases and on-line*“. Machine learning, 36(1):85–103, 1999.
- [45] Leo Breiman. “*Bagging predictors*“. Machine learning, 24(2):123–140, 1996.
- [46] Tin Kam Ho. “*The random subspace method for constructing decision forests*“. IEEE transactions on pattern analysis and machine intelligence, 20(8):832–844, 1998.
- [47] Gilles Louppe and Pierre Geurts. “*Ensembles on random patches*“. In “*Joint European Conference on Machine Learning and Knowledge Discovery in Databases*“, pages 346–361. Springer, 2012.
- [48] Gaurav Singhal. “*Ensemble Methods in Machine Learning: Bagging Versus Boosting*“. <https://www.pluralsight.com/guides/ensemble-methods:-bagging-versus-boosting>.
- [49] Rishi Kumar. “*Random Forest — a Sturdy algorithm*“. <https://medium.com/nerd-for-tech/random-forest-sturdy-algorithm-d60b9f9140d4>, .
- [50] Yoav Freund and Robert E Schapire. “*A decision-theoretic generalization of on-line learning and an application to boosting*“. Journal of computer and system sciences, 55(1):119–139, 1997.
- [51] Yoav Freund, Robert E Schapire, et al. “*Experiments with a new boosting algorithm*“. In “*icml*“, volume 96, pages 148–156. Citeseer, 1996.
- [52] Ryuk. “*Step-by-Step Guide to Implement Machine Learning VI - AdaBoost*“. <https://www.codeproject.com/Articles/4114375/Step-by-Step-Guide-to-Implement-Machine-Learning>.
- [53] Léon Bottou. “*Large-scale machine learning with stochastic gradient descent*“. In “*Proceedings of COMPSTAT’2010*“, pages 177–186. Springer, 2010.
- [54] Scikit Learn. “*Stochastic Gradient Descent*“. <https://scikit-learn.org/stable/modules/sgd.html#sgd>, .
- [55] Tiancheng Yuan Joshua Mathews Taiwo Olorunniwo Jonathon Price, Alfred Wong. “*Stochastic gradient descent*“. [https://optimization.cbe.cornell.edu/index.php?title=Stochastic\\_gradient\\_descent](https://optimization.cbe.cornell.edu/index.php?title=Stochastic_gradient_descent).
- [56] Divakar Kapil. “*Stochastic vs Batch Gradient Descent*“. [https://medium.com/@divakar\\_239/stochastic-vs-batch-gradient-descent-8820568eada1](https://medium.com/@divakar_239/stochastic-vs-batch-gradient-descent-8820568eada1).
- [57] Sebastian Raschka. “*How is stochastic gradient descent implemented in the context of machine learning and deep learning?*“. <https://sebastianraschka.com/faq/docs/sgd-methods.html#1-stochastic-gradient-descent-v1>, . [Online; accessed 4-November-2022].

## ΒΙΒΛΙΟΓΡΑΦΙΑ

---

- [58] Jerome H Friedman. “*Greedy function approximation: a gradient boosting machine*“. *Annals of statistics*, pages 1189–1232, 2001.
- [59] Tianqi Chen and Carlos Guestrin. “*Xgboost: A scalable tree boosting system*“. In “*Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*“, pages 785–794, 2016.
- [60] Guolin Ke, Qi Meng, Thomas Finley, Taifeng Wang, Wei Chen, Weidong Ma, Qiwei Ye, and Tie-Yan Liu. “*Lightgbm: A highly efficient gradient boosting decision tree*“. *Advances in neural information processing systems*, 30, 2017.
- [61] Ashik Kumar. “*XGBoost Vs LightGBM*“. <https://www.linkedin.com/pulse/xgboost-vs-lightgbm-ashik-kumar/>, .
- [62] Wikipedia. “*Νευρόνασ*“. <https://el.wikipedia.org/wiki/Νευρόνασ>.
- [63] Stanford CS. “*Neural Networks Part 1: Setting up the Architecture*“. <https://cs231n.github.io/neural-networks-1/>. [Online; accessed 15-December-2022].
- [64] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “*Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*“. CoRR, abs/1502.01852, 2015.
- [65] Ian J Goodfellow, David Warde-Farley, Mehdi Mirza, Aaron C Courville, and Yoshua Bengio. “*Maxout networks*“. ICML (3), 28:1319–1327, 2013.
- [66] Michael Nielsen. “*Improving the way neural networks learn*“. <http://neuralnetworksanddeeplearning.com/chap3.html>, . [Online; accessed 23-December-2022].
- [67] David E Rumelhart, Geoffrey E Hinton, and Ronald J Williams. “*Learning representations by back-propagating errors*“. Cognitive modeling, 5(3):1, 1988.
- [68] Michael Nielsen. “*How the backpropagation algorithm works*“. <http://neuralnetworksanddeeplearning.com/chap2.html>, . [Online; accessed 20-December-2022].
- [69] Scikit Learn. “*Neural network models (supervised)*“. [https://scikit-learn.org/stable/modules/neural\\_networks\\_supervised.html](https://scikit-learn.org/stable/modules/neural_networks_supervised.html), .
- [70] Aayush Bajaj. “*Performance Metrics in Machine Learning [Complete Guide]*“. <https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide#:~:text=Classification>. [Online; accessed 24-November-2022].
- [71] Scikit Learn. “*Cross-validation: evaluating estimator performance*“. [https://scikit-learn.org/stable/modules/cross\\_validation.html](https://scikit-learn.org/stable/modules/cross_validation.html), .
- [72] Sébastien Arlot and A Celisse. “*A survey of cross-validation procedures for model selection*“. Statistical methodology, 17:40–79, 2010.

- [73] Jacques Wainer and Gavin Cawley. “*Nested cross-validation when selecting classifiers is overzealous for most practical applications*“. Expert Systems with Applications, 182:115222, 2021.
- [74] Saeid Parvandeh, Hung-Wen Yeh, Martin P Paulus, and Brett A McKinney. “*Consensus features nested cross-validation*“. Bioinformatics, 36(10):3093–3098, 2020.
- [75] Ioannis Tsamardinos, Amin Rakhshani, and Vincenzo Lagani. “*Performance-estimation properties of cross-validation-based protocols with simultaneous hyper-parameter optimization*“. International Journal on Artificial Intelligence Tools, 24(05):1540023, 2015.
- [76] Stack Exchange. “*What’s the meaning of nested resampling*“. <https://stats.stackexchange.com/questions/292179/whats-the-meaning-of-nested-resampling>.
- [77] James Bergstra and Yoshua Bengio. “*Random search for hyper-parameter optimization*“. Journal of machine learning research, 13(2), 2012.
- [78] Andre Perunicic. “*HOW ARE PRINCIPAL COMPONENT ANALYSIS AND SINGULAR VALUE DECOMPOSITION RELATED?*“. <https://intoli.com/blog/pca-and-svd/>. [Online; accessed 4-December-2022].
- [79] Gaston Sanchez Tomas Aluja, Alain Morineau. “*Principal Component Analysis for Data Science*“. <https://stats.stackexchange.com/questions/292179/whats-the-meaning-of-nested-resampling>.
- [80] Pablo Aznar. “*What is Mutual Information?*“. <https://quantdare.com/what-is-mutual-information/>.
- [81] Yasser Roudi Peter Latham. “*Mutual Information*“. [http://www.scholarpedia.org/article/Mutual\\_information](http://www.scholarpedia.org/article/Mutual_information).
- [82] Zhenyu Zhao, Radhika Anand, and Mallory Wang. “*Maximum relevance and minimum redundancy feature selection methods for a marketing machine learning platform*“. In “*2019 IEEE international conference on data science and advanced analytics (DSAA)*“, pages 442–452. IEEE, 2019.