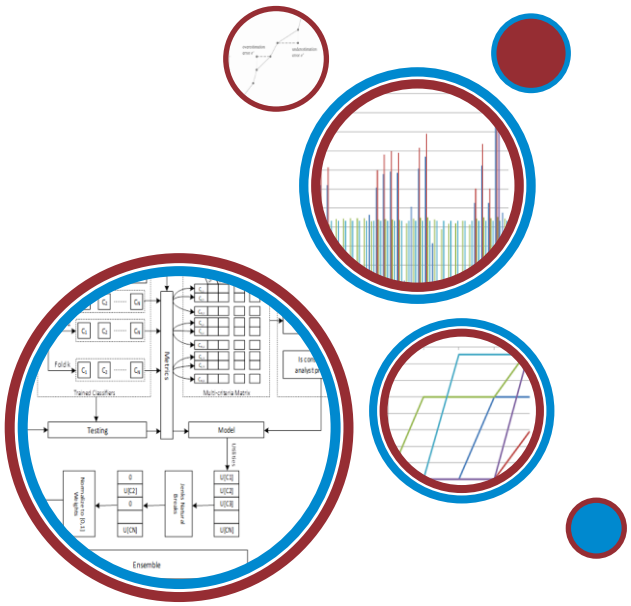
ΠΟΛΥΤΕΧΝΕΙΟ ΚΡΗΤΗΣ

ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ ΠΑΡΑΓΩΓΗΣ ΚΑΙ ΔΙΟΙΚΗΣΗΣ

Ανάπτυξη πολυκριτήριας μεθοδολογίας υπολογισμού βαρών σε ensemble τεχνικές μηχανικής μάθησης

ΔΙΠΛΩΜΑΤΙΚΉ ΕΡΓΑΣΙΑ

Φλώκος Θεόδωρος



Επιβλέπων : Ματσατσίνης Νικόλαος,
Καθηγητής

ΧΑΝΙΑ 2018

Πρόλογος

Με αφορμή την ολοκλήρωση των προπτυχιακών μου σπουδών καθώς και της διπλωματικής μου εργασίας θα ήθελα να ευχαριστήσω τους γονείς μου, για τη στήριξη που μου παρείχαν σε όλη τη διάρκεια των σπουδών μου. Επίσης θα ήθελα να ευχαριστήσω τους φίλους μου για όλες τις όμορφες στιγμές που περάσαμε μαζί. Ακόμα θα ήθελα να ευχαριστήσω το διδακτορικό φοιτητή Αλκαίο Σακελλάρη, για τη συνεισφορά του σε όλα τα στάδια εκπόνησης αυτής της εργασίας. Τέλος θα ήθελα να ευχαριστήσω τον καθηγητή και επιβλέπων της παρούσας διπλωματικής εργασίας, κύριο Ματσατσίνη Νικόλαο, για τη συνεχή καθοδήγηση την οποία μου παρείχε κατά τη διάρκεια εκπόνησης της εργασίας.

Φλώκος Θεόδωρος
Χανιά, Σεπτέμβριος 2018

Περιεχόμενα

Περίληψη 3
Abstract 4
ΚΕΦΑΛΑΙΟ 1 5
1.1 ΕΙΣΑΓΩΓΗ 5
1.2 ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ 6
1.2.1. Εισαγωγή στη μηχανική μάθηση 6
1.2.2. Εισαγωγή στην αναγνώριση προτύπων 7
1.2.3. Μάθηση 7
1.2.4. Χαρακτηριστικά 10
1.2.5. Κλάσεις 10
1.2.6. Σύνολο Δεδομένων 11
1.2.7. Η Διαδικασία της ταξινόμησης 12
1.2.8. Ο Διαχωρισμός των δεδομένων 13
1.2.9. Μέθοδοι διαχωρισμού του συνόλου δεδομένων 15
1.2.10. Μετρά απόδοσης των ταξινομητών 16
1.2.11. Bias vs Variance 27
ΚΕΦΑΛΑΙΟ 2 29
2.1 ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΑΞΙΝΟΜΗΤΩΝ 29
2.1.1. Εισαγωγή στη θεωρία των ταξινομητών 29
2.1.2. Logistic Regression 30
2.1.3. k-Nearest Neighbors 33
2.1.4. Naïve Bayes 36
2.1.5. Decision Trees 39
2.1.6. Support Vector Machine 42
2.1.7. Artificial Neural Networks 47
ΚΕΦΑΛΑΙΟ 3 53
3.1 ΣΥΝΔΥΑΣΜΟΣ ΤΑΞΙΝΟΜΗΤΩΝ 53
3.1.1. Εισαγωγή 53
3.1.2. Ensemble Learning 53
3.1.3. Προβλήματα στη δημιουργία ensemble μοντέλων 59
3.2 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΜΕ ΠΟΛΥΚΡΙΤΗΡΙΑ ΑΝΑΛΥΣΗ ΑΠΟΦΑΣΕΩΝ 60
3.2.1. Εισαγωγή 60
3.2.2. Πολυκριτηρία Ανάλυση Αποφάσεων: Η Αναλυτική-Συνθετική προσέγγιση 60

3.2.3. Η μέθοδος UTA 62
3.2.4. Ο αλγόριθμος UTASTAR 66
ΚΕΦΑΛΑΙΟ 4 69
4.1 ΠΡΟΤΕΙΝΟΜΕΝΗ ΜΕΘΟΔΟΛΟΓΙΑ 69
4.1.1. Εισαγωγή 69
4.1.2. Στάδιο 1^{ο} :Επιλογή συνόλων 70
4.1.3. Στάδιο 2^{ο} : Διαχωρισμός του συνόλου δεδομένων 70
4.1.4. Στάδιο 3^{ο} : Εκπαίδευση και έλεγχος των βασικών ταξινομητών 71
4.1.5. Στάδιο 4^{ο} : Μοντελοποίηση με χρήση της πολυκριτήριας ανάλυσης αποφάσεων 71
4.1.6. Στάδιο 5^{ο}: Εύρεση των βαρών και συνδυασμός των ταξινομητών 72
4.1.7. Στάδιο 6^{ο}: Επιλογή των καλύτερων ταξινομητών με τη χρήση συσταδοποίησης. 73
ΚΕΦΑΛΑΙΟ 5 76
5.1 Πειραματικό Στάδιο 76
5.2 Αποτελέσματα 78
5.3 Συμπεράσματα 93
5.4 Περαιτέρω Έρευνα 93
Βιβλιογραφία 94
Παράρτημα Α : Σύνολα Δεδομένων 97
A.1 Breast Cancer Wisconsin (Original) 97
A.2 Chess (King-Rook vs. King) 98
A.3 Pen-Based Recognition of Handwritten Digits 99
A.4 Statlog (Landsat Satellite) 100
A.5 Waveform Database Generator 101
Παράρτημα Β : Πολυκριτήριοι Πίνακες 103
B.1 Breast Cancer Wisconsin (Original) 103
B.2 Chess (King-Rook vs. King) 104
B.3 Pen-Based Recognition of Handwritten Digits 105
B.4 Statlog (Landsat Satellite) 106
B.5 Waveform Database Generator 107

Περίληψη

Η ανάγκη για το συνδυασμό της γνώσης των ειδικών και η χρησιμοποίηση αυτής για την υποβοήθηση της λήψης αποφάσεων, είναι ένα θέμα το οποίο απασχολεί σε μεγάλο βαθμό τη σύγχρονη επιστήμη των αποφάσεων. Ένα από τα προβλήματα που αντιμετωπίζεται συχνά από τους ειδικούς κατά τη διάρκεια της λήψης αποφάσεων είναι η ταξινόμηση αντικειμένων (πχ προϊόντων, υπηρεσιών, καταστάσεων κτλ) σε πολλαπλές κατηγορίες.

Από την άλλη μεριά, η «έκρηξη» των δεδομένων στη σημερινή εποχή, καθιστά τη διαδικασία της ταξινόμησης αντικειμένων ως ένα εξαιρετικά δύσκολο έργο για τους ειδικούς. Για αυτό το λόγο έχουν αναπτυχθεί υπολογιστικά μοντέλα στο τομέα της μηχανικής μάθησης, τα οποία προσομοιάζουν το τρόπο λήψης αποφάσεων των ειδικών με τη βοήθεια εμπειρικών δεδομένων. Αυτά τα μοντέλα στην περίπτωση της διαδικασίας της ταξινόμησης αντικειμένων είναι γνωστά ως «ταξινομητές». Με σκοπό την περαιτέρω βελτιστοποίηση της λήψης αποφάσεων, έχουν αναπτυχθεί τεχνικές για το συνδυασμό των «ταξινομητών», οι οποίες είναι γνωστές ως «ensemble» τεχνικές. Ένα από τα προβλήματα που παρουσιάζονται στην ανάπτυξη «ensemble» τεχνικών είναι η ποσοτικοποίηση της συνεισφοράς του κάθε «ταξινομητή» στην τελική απόφαση. Η πλειοψηφία των διαδεδομένων «ensemble» τεχνικών, είτε αποδίδει ένα «βάρος» σε κάθε «ταξινομητή» ανάλογα με την απόδοση του σε κάποιο συγκεκριμένο στατιστικό μέτρο ακρίβειας, είτε κάνει την υπόθεση ότι η συνεισφορά όλων των «ταξινομητών» είναι ισότιμη. Όμως είναι γνωστό ότι τα στατιστικά μέτρα ακρίβειας έχουν πλεονεκτήματα και μειονεκτήματα λόγο των υποθέσεων που κάνουν. Συνεπώς η επιλογή ενός μόνο στατιστικού μέτρου για την κατανομή των βαρών των «ταξινομητών» δημιουργεί ελλείψεις στο τρόπο επιλογής των βαρών. Εμείς θα διερευνήσουμε τη δυνατότητα το παραπάνω πρόβλημα να θεωρηθεί ως ένα πολυκριτήριο πρόβλημα, λαμβάνοντας ως κριτήρια ένα σύνολο από στατιστικά μέτρα ακρίβειας. Σκοπός της παρούσας διπλωματικής εργασίας είναι ανάπτυξη μιας «ensemble» τεχνικής η οποία θα λαμβάνει υπόψη της τα προαναφερθέντα στην κατανομή των βαρών των «ταξινομητών» χρησιμοποιώντας μεθόδους από τη πολυκριτήρια ανάλυση αποφάσεων.

Abstract

The need for expert knowledge fusion and it's use for decision support, is a troubling subject for the modern decision sciences. One of the problems the experts face during the process of decision making is the classification of objects (ex: products, services, conditions etc.) in multiple classes.

On the other hand, the «explosion» of data in the modern era , makes the classification of objects a difficult task for experts. To tackle this set of problems, numerous computational models in the field of machine learning have been developed, that simulate the way experts make decisions using empirical data. In the context of classification those models are called «classifiers» . For the post optimization of the decision making process a class of methods have been developed for the fusion of the classifiers which are called «ensemble» methods. One of the common problems in the process of creating an ensemble is defining the distribution of weights of the classifiers. Most of ensemble methods set the classifiers weights based on the classifiers performance or they set equal weights for all the classifiers. Though the classification performance measures show some advantages and disadvantages depending the hypothesis each of them is based on. So the selection of single classification performance measure to define the classifiers weights is not an effective technique. We are going to model such a problem as multicriteria problem, taking into account a set of classification performance measures. The goal of this dissertation is to develop an «ensemble» method that takes into account all of the things mentioned above in the definition of classifiers weights, with the help of multicriteria decision analysis.

ΚΕΦΑΛΑΙΟ 1

1.1 ΕΙΣΑΓΩΓΗ

Με την εισαγωγή του διαδικτύου στη καθημερινή ζωή των ανθρώπων τη τελευταία δεκαπενταετία, υπήρξε μια έκρηξη στο αριθμό των δεδομένων και των πληροφοριών που εκτίθεται ο μέσος άνθρωπος. Αντίστοιχη λοιπόν ήταν και η αύξηση των πληροφοριών που χρειάζεται να επεξεργαστεί ο άνθρωπος για να πάρει αποφάσεις κατά τη διάρκεια της καθημερινής ζωής του, από τις πιο «μικρές» (όπως το τι θα φάει) μέχρι και τις πιο «μεγάλες» (όπως το να διαγνώσει μια ασθένεια που έχει ένας άλλος άνθρωπος ή να επιλέξει τη καλύτερη για αυτόν επαγγελματική πορεία). Τα προβλήματα αυτά κατ'ουσίαν απαιτούν από τον άνθρωπο να εκτελέσει κάποιες ενέργειες για να πάρει αποφάσεις όπως πχ η ταξινόμηση αντικειμένων ή την κατάταξη αυτών. Τη λύση σε ορισμένα από αυτά τα προβλήματα έδωσε η τεχνητή νοημοσύνη και δη η μηχανική μάθηση με τη δημιουργία μοντέλων τα οποία με την κατάλληλη εκπαίδευση μπορούν να πάρουν τις αντίστοιχες αποφάσεις. Καθώς όμως αυτά τα μοντέλα λειτουργούν στα πλαίσια των υπολογιστικών συστημάτων μπορούν να επεξεργαστούν μεγαλύτερο όγκο δεδομένων και με μεγαλύτερη ταχύτητα από ότι ένας άνθρωπος. Επομένως καθίστανται κατάλληλα για την υποβοήθηση του ανθρώπου στη λήψη αποφάσεων. Επιπρόσθετα ο άνθρωπος στη καθημερινή του ζωή συμβουλεύεται πολλές φορές παραπάνω από ένα ειδικούς για τη λήψη μιας σημαντικής απόφασης έτσι ώστε να έχει σφαιρική άποψη επί των πραγμάτων. Αυτό το φαινόμενο παρατηρείται και από αντίστοιχα μοντέλα της μηχανικής μάθησης τις ensemble, οι οποίες συνδυάζουν τη γνώση που έχουν αποκομίσει πολλά «βασικά» μοντέλα για τη λήψη μιας πιο «σφαιρικής» απόφασης. Στη παρούσα εργασία θα παρουσιαστούν κάποιοι από τους τρόπους με τους οποίους μπορούν να δημιουργηθούν μοντέλα ensemble για προβλήματα ταξινόμησης. Γενικότερος σκοπός είναι η δημιουργία ενός συστήματος υποστήριξης αποφάσεων για την υποβοήθηση ενός αναλυτή στη δημιουργία ενός ensemble μοντέλου, με τη βοήθεια της πολυκριτήριας ανάλυσης αποφάσεων. Η δομή της εργασίας είναι η εξής. Αρχικά θα γίνει μια εισαγωγή στο τομέα της μηχανικής μάθησης και της αναγνώρισης προτύπων. Έπειτα θα παρουσιαστούν κάποιες εισαγωγικές έννοιες οι οποίες είναι αναγκαίες για την κατανόηση του περιεχόμενου αυτής της εργασίας. Στη συνέχεια θα παρουσιαστούν ορισμένοι από τους αλγόριθμους για τη δημιουργία βασικών μοντέλων ταξινόμησης. Επιπλέον θα παρουσιαστεί ένα σύνολο από μεθόδους με τους οποίους μπορούν να συνδυαστούν οι προβλέψεις πολλών «βασικών» μοντέλων ταξινόμησης για την λήψη «καλύτερων» αποφάσεων, καθώς και το προτεινόμενο σύστημα υποστήριξης αποφάσεων. Τέλος θα παρουσιαστούν τα αποτελέσματα και τα συμπεράσματα της εκτέλεσης μια σειράς πειραμάτων όσον αφορά τον τρόπο δημιουργίας και λειτουργίας ενός ensemble μοντέλου.

1.2 ΕΙΣΑΓΩΓΙΚΕΣ ΕΝΝΟΙΕΣ

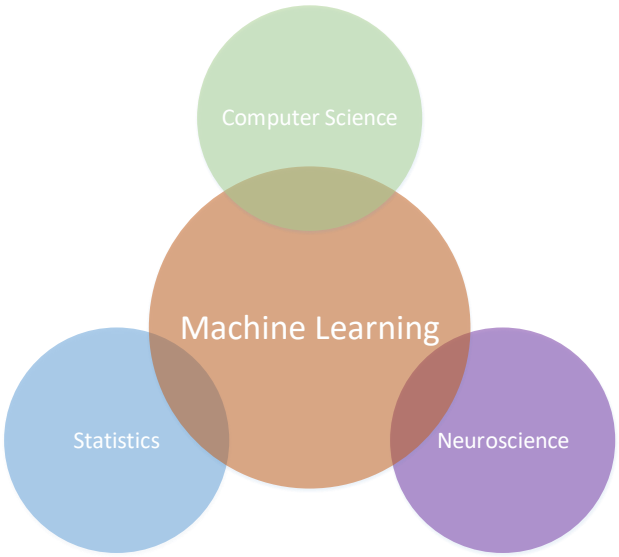
1.2.1. Εισαγωγή στη μηχανική μάθηση

Η μηχανική μάθηση είναι ο τομέας ο οποίος, προέρχεται από το συνδυασμό της επιστήμης υπολογιστών, της στατιστικής και της νευροεπιστήμης (Εικόνα 1.2.1-1). Ο τομέας αυτός προσπαθεί να απαντήσει στο ερώτημα «Πως μπορούμε να κατασκευάσουμε συστήματα τα οποία μπορούν να εξελίσσονται αυτόματα με την εμπειρία που αποκτούν, και ποιοί είναι οι θεμελιώδης νόμοι που διέπουν τις διαδικασίες μάθησης;»[1].

Η παραπάνω ερώτηση καλύπτει μια μεγάλη ποικιλία από προβλήματα μάθησης, όπως το πώς μπορούμε να σχεδιάσουμε μη επανδρωμένα αεροσκάφη τα οποία μαθαίνουν να κινούνται στο χώρο σαν μονάδες αλλά και ως σύνολο μιας ομάδας (reinforcement learning), πώς μπορούμε να κατασκευάσουμε μοντέλα τα οποία θα είναι ικανά να προβλέψουν τις καιρικές συνθήκες μια περιοχής από ιστορικά δεδομένα (time-series forecasting), πως μπορούμε να σχεδιάσουμε συστήματα τα οποία μαθαίνουν να εντοπίζουν ατέλειες στα προϊόντα που παράγει μια βιομηχανική μονάδα (classification of defects) και πως μπορούμε να εξορύξουμε δεδομένα από βάσεις δεδομένων στο διαδίκτυο για να μάθουμε τις καταναλωτικές προτιμήσεις των υποψήφιων πελατών μας, ώστε να τους προτείνουμε τα καταλληλότερα προϊόντα για αυτούς (recommendations systems, preference learning).

Πιο συγκεκριμένα λέμε ότι μια μηχανή μαθαίνει με βάση μια συγκεκριμένη εργασία T, μέτρο απόδοσης P, και τύπο της εμπειρίας που αποκτά E, αν το σύστημα βελτιώνει την απόδοση του P στην εργασία T με αξιοπιστία, με βάση την εμπειρία του E[2]. Ανάλογα με το πώς θα ορίσουμε τα T, P, E η διαδικασία μάθησης μπορεί να πάρει διάφορα ονόματα όπως εξόρυξη δεδομένων (data mining), αναγνώριση προτύπων (pattern recognition), κτλ[2]. Στη παρούσα εργασία θα ασχοληθούμε με το τομέα της αναγνώρισης προτύπων.

Ενδιαφέρον για τη μηχανική μάθηση έχουν δείξει διάφοροι επιστημονικοί τομείς τα τελευταία χρόνια όπως η βιολογία για τη χαρτογράφηση του ανθρώπινου γονιδιώματος, η νευροεπιστήμη ώστε να ρίξει φως στις διαδικασίες με τι οποίες μαθαίνει ο ανθρώπινος εγκέφαλος, καθώς και η φυσική επιστήμη για να ανακαλύψει μοτίβα στα δεδομένα που λαμβάνει από τον LHC στο ίδρυμα CERN.



Εικόνα 1.2.1-1: Αναπαράσταση σχέσης μηχανικής μάθησης, στατιστικής και νευροεπίστήμης

1.2.2. Εισαγωγή στην αναγνώριση προτύπων

Η αναγνώριση προτύπων είναι ο επιστημονικός τομέας ο οποίος ανήκει στο τομέα της μηχανικής μάθησης, του οποίου ο σκοπός είναι η ταξινόμηση (classification) αντικειμένων σε ένα αριθμό από κατηγορίες ή κλάσεις. Τα αντικείμενα αυτά αναφέρονται στη βιβλιογραφία και ως πρότυπα. Ανάλογα με τη φύση του προβλήματος προς επίλυση τα αντικείμενα αυτά μπορούν να πάρουν τη μορφή ήχου, εικόνας , κυματομορφής ή οποιασδήποτε μέτρησης η οποία πρέπει να ταξινομηθεί σε κάποια κατηγορία[3].

Η ταξινόμηση αντικειμένων αποτελεί μια διαδικασία την οποία οι άνθρωποι εκτελούν καθημερινά στη ζωή τους είτε με τη θέληση τους, είτε ασυναίσθητα. Από τα πρώτα κιόλας χρόνια της ζωής του, ο άνθρωπος μαθαίνει να ταξινομεί αντικείμενα, καταστάσεις και άλλους ανθρώπους στις κατηγορίες του «καλού» και του κακού», είτε μόνος του μέσα από κάποια υποτυπώδης λογική την οποία έχει αναπτύξει, είτε με τη βοήθεια των γονέων του μέσω υποδείξεων και παραδειγμάτων.

Από την άλλη μεριά μια μηχανή ενώ μπορεί να εκτελέσει πολλαπλούς υπολογισμούς και αναλύσεις σε κλάσματα του αντίστοιχου χρόνου που χρειάζεται ο άνθρωπος, δεν μπορεί να εκτελέσει καμία διαδικασία αν αυτή δεν έχει ορισθεί αυστηρά ή αν δεν εκπαιδευτεί στο να εκτελεί αυτή τη διαδικασία. Επομένως αναγκαίος είναι ο ακριβής ορισμός του προβλήματος της ταξινόμησης. Τα αντικείμενα λοιπόν περιγράφονται από χαρακτηριστικά τα οποία αποτελούνται συνήθως από μετρήσεις, έτσι ώστε να είναι δυνατή η εκπαίδευση του συστήματος.

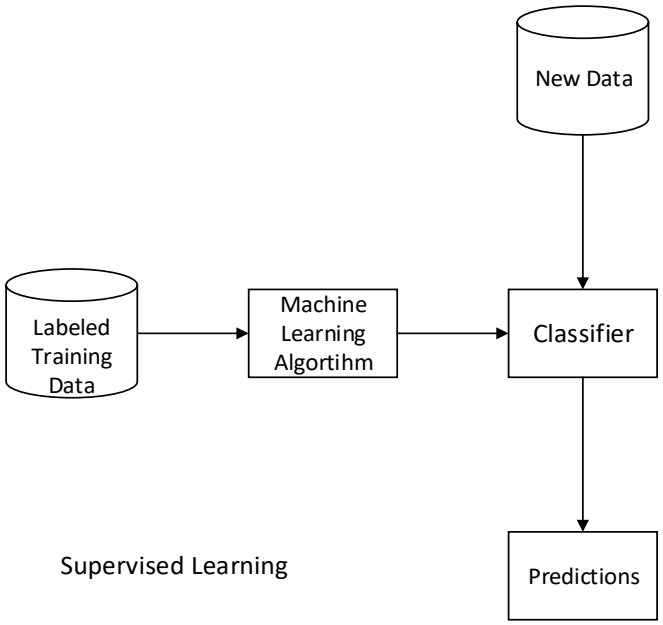
1.2.3. Μάθηση

Μέσω της διαδικασίας της εκπαίδευσης το σύστημα, αποσκοπεί στην απόκτηση γνώσης έτσι ώστε να είναι σε θέση να διακρίνει τα διαφορετικά αντικείμενα (πρότυπα) και να τα ταξινομεί στις κατηγορίες (κλάσεις) που πραγματικά ανήκουν. Απώτερος σκοπός της εκπαίδευσης του συστήματος είναι η εφαρμογή της ήδη αποκτημένης γνώσης σε νέα άγνωστα σε αυτό πρότυπα. Ανάλογα τον τρόπο και τη πληροφορία που είναι προσβάσιμη από το σύστημα σε κάθε στάδιο της εκπαίδευσης, τα προβλήματα αναγνώρισης προτύπων χωρίζονται σε 3 προβληματικές, τη μάθηση με επίβλεψη (supervised learning) , τη μάθηση χωρίς επίβλεψη (unsupervised learning) και τη μάθηση με μερική επίβλεψη (semi-supervised learning).

1.2.3.1. Μάθηση με επίβλεψη

Στη προβληματική της μάθησης με επίβλεψη το εκπαιδευόμενο σύστημα ταξινόμησης τροφοδοτείται με ένα σύνολο δεδομένων του οποίου κάθε πρότυπο έχει ήδη αντιστοιχηθεί σε μια κατηγορία (κλάση). Στο συγκεκριμένο πρόβλημα σκοπός του συστήματος ταξινόμησης είναι η ταξινόμηση των προτύπων σε κατηγορίες (κλάσεις) ενώ ήδη γνωρίζει τις πραγματικές κλάσεις στις οποίες ανήκουν (Εικόνα 1.2.3.1-1). Πιο συγκεκριμένα καθώς το σύστημα εκπαιδεύεται κάνει προβλέψεις για τα πρότυπα τα οποία δε γνωρίζει, οι οποίες συγκρίνονται με τις κλάσεις που πραγματικά ανήκουν αυτά τα πρότυπα και υπολογίζεται το ποσοστό λάθους των προβλέψεων. Μέσα από αυτή τη διαδικασία δοκιμής-λάθους το σύστημα τελικά απόκτα την εμπειρία έτσι να μπορεί να ταξινομήσει τα πρότυπα στις υπάρχουσες κλάσεις. Ένα από τα πιο κλασικά παραδείγ-

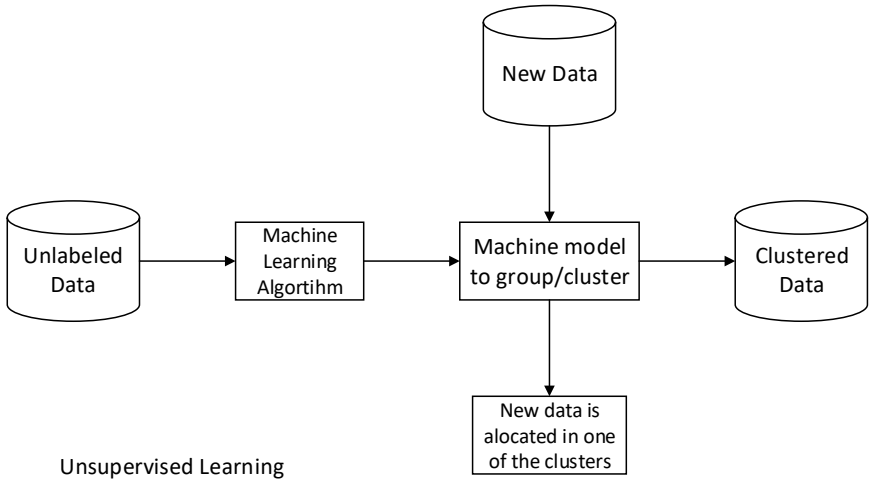
ματα της μάθησης με επίβλεψη είναι η αντιστοίχηση ενός συνόλου από e-mails σε «επιθυμητά» και «ανεπιθύμητα» e-mail.



Εικόνα 1.2.3.1-1: Διάγραμμα ροής για επιβλεπόμενη μάθηση.

1.2.3.2. Μάθηση χωρίς επίβλεψη

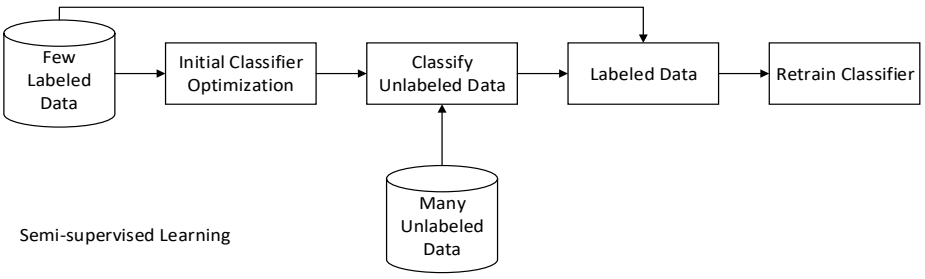
Στη προβληματική της μάθησης χωρίς επίβλεψη το εκπαιδευόμενο σύστημα ταξινόμησης τροφοδοτείται σκόπιμα με ένα σύνολο δεδομένων του οποίου κάθε πρότυπο δεν έχει αντιστοιχηθεί σε κάποια κατηγορία (κλάση). Στο συγκεκριμένο πρόβλημα σκοπός του συστήματος είναι η ανακάλυψη της δομής των δεδομένων (Εικόνα 1.2.3.2-1). Πιο συγκεκριμένα σκοπός είναι η διερεύνηση των δεδομένων για την ύπαρξη ή μη κλάσεων, καθώς και η ποσοτικοποίηση των τιμών των χαρακτηριστικών που κάνουν ένα υποσύνολο του συνόλου των προτύπων να ανήκει σε μια κλάση (ομάδα). Γενικά υπάρχουν πολλοί αλγόριθμοι για την ομαδοποίηση προτύπων σε κλάσεις και αυτός είναι ένας από τους λόγους, για τους οποίους ένα τέτοιο πρόβλημα μπορεί να έχει πολλές διαφορετικές προσεγγίσεις (λύσεις). Το γεγονός της μη ύπαρξης ήδη αντιστοιχισμένων προτύπων καθιστά τη μέτρηση της ακρίβειας αυτού το τύπου συστημάτων μια δύσκολη υπόθεση και συνήθως έγκειται στη κρίση του αναλυτή. Ένα από τα πιο χαρακτηριστικά παραδείγματα αυτή της κατηγορίας προβλημάτων είναι η τμηματοποίηση του καταναλωτικού κοινού.



Εικόνα 1.2.3.2-1: Διάγραμμα ροής για μάθηση χωρίς επίβλεψη

1.2.3.3. Μάθηση με μερική επίβλεψη

Σε αυτή τη προβληματική το σύστημα ταξινόμησης τροφοδοτείται με ένα σύνολο το οποίο αποτελείται από μη αντιστοιχισμένα δεδομένα αλλά και από ήδη αντιστοιχισμένα σε κλάσεις δεδομένα. Η λογική πίσω από αυτή τη προβληματική είναι ότι χρησιμοποιώντας ένα σύνολο δεδομένων, για το οποίο γνωρίζουμε τις κλάσεις που ανήκουν τα πρότυπα, για την εκπαίδευση του συστήματος , σε συνεργασία με ένα σύνολο μη αντιστοιχισμένων δεδομένων δίνουμε τη δυνατότητα στο σύστημα να εκπαιδευθεί σε ήδη αποκτημένη γνώση αλλά ταυτόχρονα να ψάξει για νέα (Εικόνα 1.2.3.3-1). Αυτού του τύπου τα συστήματα κάνουν την υπόθεση ότι τα μη αντιστοιχισμένα δεδομένα ακολουθούν κάποια στατιστική ή διακατέχονται από κάποια συγκεκριμένη δομή. Σκοπός αυτών των συστημάτων είναι είτε η αντιστοίχιση των προτύπων των μη αντιστοιχισμένων δεδομένων, είτε η μάθηση του τρόπου αντιστοίχισης των προτύπων σε κλάσεις. Το παραπάνω μπορεί να παρομοιασθεί με τη διαδικασία μάθησης των ανθρώπων. Αν παρομοιάσει κάποιος τα αντιστοιχισμένα δεδομένα με τι λυμένες ασκήσεις που δίνει ένας καθηγητής στο μαθητή του και περιμένει από αυτόν να τις μάθει αλλά ταυτόχρονα περιμένει από αυτόν να λύσεις κάποιες ασκήσεις μόνος του στο σπίτι (μη αντιστοιχισμένα δεδομένα). Και στις δύο περιπτώσεις ο καθηγητής περιμένει από το μαθητή να αποδώσει.



Εικόνα 1.2.3.3-1: Διάγραμμα ροής για μάθηση με μερική επίβλεψη.

1.2.4. Χαρακτηριστικά

Για την εκπαίδευση ενός ταξινομητή χρειάζονται κάποια χαρακτηριστικά τα οποία περιγράφουν το πρόβλημα ταξινόμησης, όπως προαναφέρθηκε και παραπάνω. Τα χαρακτηριστικά αυτά μπορούν να είναι είτε ποσοτικά (μετρήσιμα) ή ποιοτικά (μη μετρήσιμα).

Τα ποσοτικά χαρακτηριστικά μπορούν να είναι είτε διακριτά (πχ ο ετήσιος αριθμός γεννήσεων/θανάτων μιας χώρας) είτε συνεχής (πχ το μέσο ετήσιο εισόδημα των ανθρώπων μιας χώρας). Τα ποιοτικά χαρακτηριστικά συνήθως μπορούν να πάρουν πολύ πιο λίγες τιμές από ότι τα ποσοτικά χαρακτηριστικά. Επίσης τα ποιοτικά χαρακτηριστικά μπορεί είτε να είναι ονομαστικά (οι διαφορετικές κατηγορίες δανείων μιας τράπεζας) είτε να έχουν τη μορφή διαβάθμισης (η σειρά κατάταξης των χωρών της ευρωπαϊκής ένωσης ανάλογα με το επίπεδο ζωής των πολιτών τους).

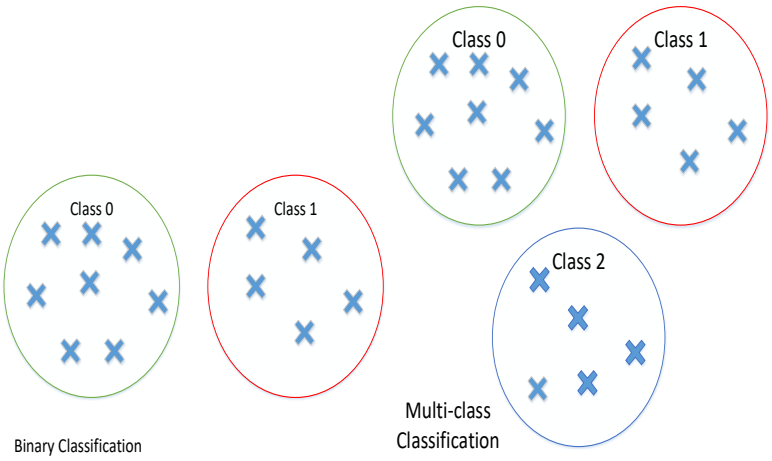
Πιο συγκεκριμένα ένα αντικείμενο/πρότυπο x ορίζεται από ένα σύνολο n χαρακτηριστικών οι τιμές $x_i, i = 1,2,3,...,n \in \mathbb{N}$ των οποίων αποτελούν ένα n-διάστατο διάνυσμα $x = [x_1, x_2, x_3, ..., x_n]^T \in \Re^n$ (feature vector). Το σύνολο $\Re^n$ αποτελεί το χώρο των χαρακτηριστικών του προβλήματος (feature space) και είναι ο χώρος στον οποίο μπορούν να πάρουν τιμές τα διαφορετικά χαρακτηριστικά. Κάθε διάσταση του χώρου αυτού είναι συσχετισμένη με ένα χαρακτηριστικό. Οι τιμές των χαρακτηριστικών συνήθως προέρχονται είτε από φυσικές μετρήσεις, είτε από το μετασχηματισμό και σύνθεση άλλων τύπων δεδομένων σε αριθμητικά (τεχνητά χαρακτηριστικά) από τον αναλυτή.

1.2.5. Κλάσεις

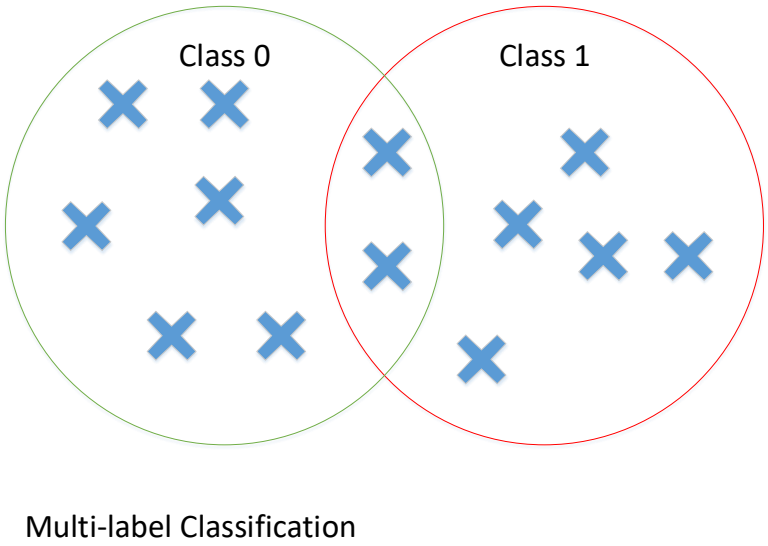
Οι κλάσεις στην ουσία αποτελούν κατηγορίες αντικειμένων τα οποία παρουσιάζουν όμοια χαρακτηριστικά. Ένα αντικείμενο μπορεί είτε να ανήκει μόνο σε μια κλάση (πχ «επιθυμητό» ή «ανεπιθύμητο» e-mail) είτε μπορεί να ανήκει σε παραπάνω από μια κλάσεις (πχ μια ταινία μπορεί να ανήκει στο είδος «περιπέτεια» αλλά και στο είδος «μυστήριο» ταυτόχρονα). Επίσης σε ένα πρόβλημα ταξινόμησης μπορεί να έχουμε παραπάνω από 2 κλάσεις (πχ «κακός», «μέτριος», «καλός»). Οι περιπτώσεις που αναφέρθηκαν παραπάνω αποτελούν η κάθε μια, μια κατηγορία προβλημάτων ταξινόμησης. Στη πρώτη περίπτωση το πρόβλημα ονομάζεται binary classification (Εικόνα 1.2.5-1), στη δεύτερη περίπτωση multi-label classification (Εικόνα 1.2.5-2) και στην τρίτη περίπτωση multi-class classification (Εικόνα 1.2.5-1). Στη παρούσα εργασία θα ασχοληθούμε μόνο με τη πρώτη και τη τρίτη κατηγορία προβλημάτων. Γενικά η διαδικασία του ορισμού των κλάσεων και των χαρακτηριστικών τους είναι μια δύσκολη υπόθεση. Σε ορισμένα προβλήματα είναι αρκετά απλός ο καθορισμός τους, όπως στο παράδειγμα με τα e-mail (γίνεται έλεγχος της εγκυρότητας του αποστολέα). Σε άλλα πάλι η διαδικασία είναι από δύσκολη έως και αδύνατη, όπως στο διαχωρισμό των επιπέδων ανάρρωσης ενός ασθενή μετά από ένα χειρουργείο, καθώς αυτό εξαρτάται από πάρα πολλούς παράγοντες, οι οποίοι πολλές φορές είναι και αστάθμητοι.

Ειδικότερα σε ένα πρόβλημα ταξινόμησης το οποίο αποτελείται από c διαφορετικές κλάσεις, αυτές ορίζονται ως $\omega_i, i = 1,2,3,...,c \in \mathbb{N}$ και όλες μαζί αποτελούν το σύνολο των κλάσεων του προβλήματος ταξινόμησης $\Omega = \{\omega_1, \omega_2, \omega_3, ..., \omega_c\}$. Τα ω_i ονομάζονται ετικέτες των κλάσεων (class labels) και αποτελούν την αναπαράσταση των κλάσεων σε μορφή κειμένου. Να σημειωθεί ότι ανάλογα τις σχέσεις που διέπουν τις δια-

φορετικές κλάσεις του προβλήματος της ταξινόμησης προκύπτουν και άλλου τύπου προβλήματα όπως η μονότονη ταξινόμηση, όπου η κλάσεις θεωρούνται μονότονες (η κλάση 1 θεωρείται καλύτερη/χειρότερη από τη 2 κτλ).



Εικόνα 1.2.5-1: Αναπαράσταση Binary και Multi-class Classification.



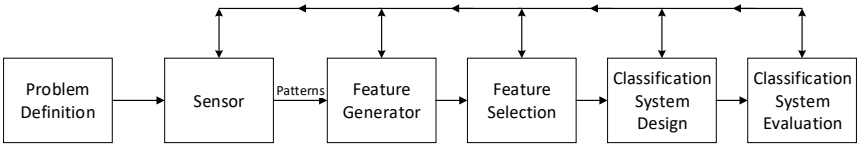
Εικόνα 1.2.5-2: Αναπαράσταση Multi-label Classification.

1.2.6. Σύνολο Δεδομένων

Τα πρότυπα με βάση τα οποία θα εκπαιδευθεί και θα κριθεί το σύστημα ταξινόμησης συγκροτούν ένα διακριτό σύνολο και ονομάζεται σύνολο δεδομένων (dataset). Αυτό το σύνολο δεδομένων αποτελείται από N πρότυπα $z_j, j = 1,2,3,...,N, N \in \mathbb{N}$ και ορίζεται ως $Z = \{z_1, z_2, z_3, ..., z_j, .., z_N\}, z_j \in \Re^n$. Οι ετικέτες των κλάσεων των προτύπων z_j δηλώνονται $l(z_j) \in \Omega, j = 1,2,3,...., N$.

1.2.7. Η Διαδικασία της ταξινόμησης

Η αναγνώριση προτύπων πάντα ακολουθεί μια συγκεκριμένη διαδικασία η οποία παραλλάσσεται ανάλογα το τύπο του προβλήματος (Εικόνα 1.2.7-1). Συνεπώς πρωταρχικός στόχος είναι ο καθορισμός των παραμέτρων του προβλήματος όπως ορίσθηκαν στη παράγραφο 1.1. Μόλις ορισθεί το πρόβλημα από τον αποφασίζοντα/αναλυτή, πρέπει να καθοριστούν τα χαρακτηριστικά που θα περιγράψουν τα πρότυπα. Η διαδικασία καθορισμού των χαρακτηριστικών είναι μια χρονοβόρα διαδικασία, καθώς μέσω αυτής πρέπει να αποτυπωθούν οι μεταβλητές του προβλήματος, κάτι το οποίο δεν είναι προφανές. Γι αυτό το λόγο πρέπει να ορισθούν όσο πιο πολλά χαρακτηριστικά γίνεται έτσι ώστε να καλυφθούν όλες οι οπτικές του προβλήματος. Πολλά από αυτά τα χαρακτηριστικά ίσως περιέχουν στοιχεία τα οποία περιέχονται και σε άλλα χαρακτηριστικά. Γενικά κάθε χαρακτηριστικό πρέπει να ορίζεται έτσι ώστε να υπάρχει η λιγότερη δυνατή σχέση με τα υπόλοιπα χαρακτηριστικά. Μόλις ορισθούν τα χαρακτηριστικά πρέπει να γίνει η λήψη του συνόλου δεδομένων του προβλήματος μέσα από κάποια πειραματική διαδικασία (συνήθως μετρήσεις με επιστημονικά όργανα). Άμεση συνέπεια του μεγάλου αριθμού χαρακτηριστικών είναι ο μεγάλος όγκος δεδομένων. Ακόμα πολλά από τα χαρακτηριστικά μπορεί να μην έχουν καμία σχέση με το πρόβλημα ή μόνο ένας συγκεκριμένος συνδυασμός αυτών να είναι σημαντικός. Για τη μείωση του μεγάλου όγκου δεδομένων και την αφαίρεση των μη σημαντικών χαρακτηριστικών ακολουθείται μια διαδικασία διαλογής των πιο σημαντικών για το πρόβλημα χαρακτηριστικών μέσα από στατιστικές μεθόδους (Correlation feature selection, Subset selection). Αφού έχει ολοκληρωθεί και ο καθορισμός του συνόλου δεδομένων επόμενο βήμα είναι ο σχεδιασμός του συστήματος ταξινόμησης, στο οποίο θα αναφερόμαστε από εδώ και ύστερα ως «ταξινομητή» (classifier). Ο σχεδιασμός του ταξινομητή είναι μια διαδικασία η οποία έχει διαφορές ανάλογα με τη προβληματική που ανήκει το πρόβλημα προς επίλυση, δηλαδή αν είναι πρόβλημα μάθησης με επίβλεψη, μάθησης χωρίς επίβλεψη ή μάθησης με μερική επίβλεψη (Εικόνες 1.2.7.1-1, **1.2.7.2-1**, **1.2.7.1-1**)[4]. Μετά το σχεδιασμό του ταξινομητή ακολουθεί η εκπαίδευση και ο έλεγχος του.



Εικόνα 1.2.7-2: Βασικά βήματα σχεδιασμού συστημάτων αναγνώρισης προτύπων.

Όπως φαίνεται και στην Εικόνα 1.2.7-2 η διαδικασία της επιλογής των χαρακτηριστικών (feature selection), της εκπαίδευσης (train) και του ελέγχου (test) είναι μια διαδικασία επαναληπτική. Κατά τη διάρκεια της εκπαίδευσης οι παράμετροι του αλγορίθμου μπορεί να αλλάξουν ανάλογα με τα αποτελέσματα που θα φέρουν στο στάδιο του ελέγχου και ο ταξινομητής να εκπαιδευτεί ξανά με τις νέες παραμέτρους με σκοπό τη βελτίωση της απόδοσης του. Πολύ συχνό φαινόμενο είναι η αλλαγή του αλγορίθμου μέσα από τον οποίο γίνεται η ταξινόμηση καθώς μπορεί να μη μπορεί να προσομοιάσει το πρόβλημα πλήρως λόγω των περιορισμών του. Η επιλογή του κατάλληλου αλγόριθμου για το κάθε πρόβλημα προς επίλυση αποτελεί μια δύσκολη διαδικασία η οποία επαφίεται σε μεγάλο βαθμό, στην εμπειρία του αναλυτή καθώς και στο τομέα εφαρμογής του.

1.2.8. Ο Διαχωρισμός των δεδομένων

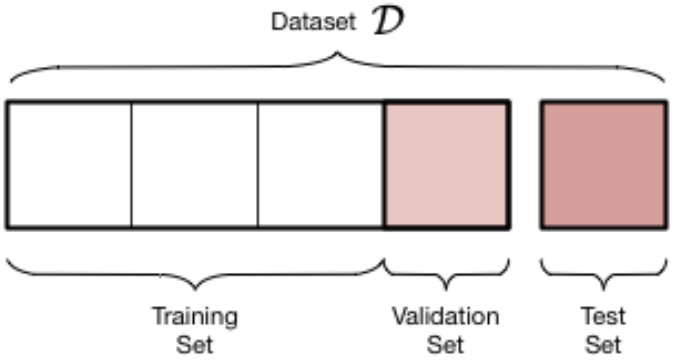
Κατά της διάρκεια του ελέγχου του ταξινομητή, η αξιολόγηση του μπορεί να γίνει χρησιμοποιώντας το ποσοστό του σφάλματος (error rate), που είναι ο λόγος των λανθασμένα ταξινομημένων προτύπων προς το συνολικό αριθμό των προτύπων.

$$error\_rate = \frac{N_{missclassified\_patterns}}{N_{patterns}} \qquad (1.1)$$

error_rate: Ποσοστό σφάλματος
$N_{missclassified\_patterns}$: Ο συνολικός αριθμός των προτύπων τα οποία έχουν ταξινομηθεί λανθασμένα.
$N_{pattems}$: Ο συνολικός αριθμός των προτύπων που περιέχονται στο σύνολο δεδομένων.

Αυτό το μέτρο απόδοσης όμως δεν είναι επιθυμητό, καθώς μη ξεχνάμε ότι αποφασίζων μπήκε στη διαδικασία για να εκπαιδεύσει ένα ταξινομητή με απώτερο σκοπό, την ταξινόμηση προτύπων τα οποία δεν έχει ξανά συναντήσει. Επιπρόσθετα με τη χρησιμοποίηση του ίδιου συνόλου δεδομένων για την εκπαίδευση και για τον έλεγχο του ταξινομητή, ελλοχεύει ο κίνδυνος το μοντέλο να κάνει over-fitting. Κάτι τέτοιο θα είχε ως συνέπεια το μοντέλο να αποστηθίσει όλο το σύνολο των δεδομένων, αλλά ταυτόχρονα να αποτύχει να γενικεύσει την γνώση που απόκτησε για ένα άλλο σύνολο δεδομένων, με αποτέλεσμα η απόδοση του σε ένα νέο σύνολο δεδομένων να είναι απαγορευτική. Επιτακτική είναι λοιπόν η ανάγκη για τη χρησιμοποίηση διαφορετικού συνόλου δεδομένων για την εκπαίδευση του ταξινομητή και διαφορετικού συνόλου δεδομένων για τον έλεγχο του. Με αυτό τον τρόπο η απόδοση η οποία θα υπολογίζεται θα αντικατοπτρίζει τη λειτουργία του σε πραγματικές συνθήκες.

Επομένως το σύνολο των δεδομένων χωρίζεται σε 3 υποσύνολα (Εικόνα 1.2.8-1), το σύνολο εκπαίδευσης (train set), το σύνολο επαλήθευσης (validation set) και το σύνολο ελέγχου (test set). Ο ταξινομητής εκπαιδεύεται χρησιμοποιώντας το train set. Το validation set χρησιμοποιείται για την βελτιστοποίηση των παραμέτρων του αλγορίθμου του ταξινομητή. Γενικά σε πολλά προβλήματα δεν μπορούν να βρεθούν οι βέλτιστοι παράμετροι του αλγορίθμου ταξινόμησης, γι' αυτό το λόγο ο αναλυτής συμβιβάζετε με ημιβέλτιστες λύσεις οι οποίες βέβαια ικανοποιούν τις προσδοκίες του. Η απόφαση αυτή για την λήξη της βελτιστοποίησης των παραμέτρων παίρνεται με βάση την αξιολόγηση της απόδοσης στο validation set. Στη συνέχεια ο ταξινομητής αξιολογείται υπολογίζοντας της απόδοση του στο test set. Για την αποφυγή του over-fitting του αλγορίθμου παρατηρούμε το train set και το validation set, όταν η αύξηση της απόδοσης στο train set δεν επιφέρει σίγουρη αύξηση της απόδοσης στο validation set, τότε η εκπαίδευση έχει φτάσει σε ικανοποιητικό επίπεδο και μπορεί να τερματιστεί. Αξιοσημείωτο είναι το γεγονός ότι τα πρώτα δύο σύνολα δεδομένων, δεν μπορούν να χρησιμοποιηθούν για τον έλεγχο (test) του ταξινομητή, καθώς τα δεδομένα που περιέχονται σε αυτά έχουν είδη χρησιμοποιηθεί από το ταξινομητή με αποτέλεσμα να τα έχει «μάθει». Η απόδοση σε αυτά τα σύνολα δεδομένων συνήθως είναι μεγαλύτερη από αυτή στο test set και δεν αντικατοπτρίζουν της πραγματικότητα.



Εικόνα 1.2.8-1: Διαχωρισμός του συνόλου δεδομένων σε train, validation και test set.

Συνέπεια των παραπάνω είναι ότι το training set και το validation set πρέπει να καταλαμβάνουν το μεγαλύτερο μέρος του συνόλου δεδομένων. Κάτι τέτοιο ισχύει καθώς τροφοδοτώντας το μοντέλο με όσα περισσότερα δεδομένα γίνεται, το βοηθάμε να καλύψει όσο πιο πολύ χώρο γίνεται από το χώρο των χαρακτηριστικών (feature space), με φυσικό επόμενο τη καλύτερη απόδοση του. Επιπλέον όσο πιο πολλά δεδομένα χρησιμοποιηθούν για την επαλήθευση του μοντέλου τόσο πιο αντικειμενική θα είναι η εκτίμηση της απόδοσης του. Παρόλα αυτά η χρησιμοποίηση των τριών προαναφερθέντων υποσυνόλων δεδομένων στην πραγματικότητα πολλές φορές αποδεικνύεται προβληματική και αυτό γιατί πολλές φορές τα σύνολα δεδομένων τα οποία έχουμε στη διάθεση μας είναι αρκετά μικρά σε μέγεθος (πχ λόγω της δυσκολίας απόκτησης αυτών των δεδομένων).

Το παραπάνω πρόβλημα μπορεί να επιλυθεί μερικώς με την παράληψη του validation set και συνεπώς το διαχωρισμό του συνόλου δεδομένων σε training set και test set. Με αυτό τον τρόπο ένα πολύτιμο κομμάτι δεδομένων, το οποίο σε αντίθετη περίπτωση θα είχε σπαταληθεί για την επαλήθευση και βελτιστοποίηση του μοντέλου, τροφοδοτείται στο ίδιο το μοντέλο με αποτέλεσμα να μπορεί να καλύψει ακόμα μεγαλύτερο κομμάτι του χώρου των χαρακτηριστικών.

Ακόμα με αυτό τον τρόπο το μοντέλο έχει στη διάθεση του παραπάνω δεδομένα για έλεγχο, με αποτέλεσμα την πιο σφαιρική γνώμη του αναλυτή για τη συμπεριφορά του μοντέλου υπό αβεβαιότητα. Στην πράξη συνήθως χρησιμοποιείται το 70% του συνόλου δεδομένων για την εκπαίδευση του μοντέλου και το 30% για τον έλεγχο αυτού. Τα παραπάνω ποσοστά βέβαια δεν είναι σταθερά αλλά επαφίενται στην κρίση του αναλυτή. Τα δεδομένα που αποτελούν τα δύο αυτά σύνολα δεδομένων θα πρέπει να είναι αντιπροσωπευτικά δείγματα του συνόλου δεδομένων στο οποίο θα εφαρμοσθεί η ταξινόμηση. Για το καλύτερο δυνατό διαχωρισμό του συνόλου δεδομένων σε training set και test set έχουν αναπτυχθεί διάφορες μέθοδοι, ορισμένες από τις οποίες θα αναφερθούν παρακάτω.

1.2.9. Μέθοδοι διαχωρισμού του συνόλου δεδομένων

1.2.9.1. Bootstrapping

Η συγκριμένη τεχνική βασίζεται στις αρχές του κεντρικού οριακού θεωρήματος για να υπολογίσει τη μέση τιμή καθώς και τη τυπική απόκλιση της εκτιμώμενης μεταβλητής, όπου στη περίπτωση της ταξινόμησης είναι η απόδοση του ταξινομητή. Πιο συγκεκριμένα έστω το σύνολο των δεδομένων T, το οποίο αποτελείται από $p \in \mathbb{N}$ πρότυπα. Μέσω αυτής της διαδικασίας πραγματοποιείται η λήψη k τυχαίων δειγμάτων με επανατοποθέτηση, μεγέθους $s \le p$. Δηλαδή κάθε δείγμα μπορεί να περιέχει ένα πρότυπο παραπάνω από μια φορές[5]. Το σύνολο των προτύπων όπου αποτελούν ένα δείγμα συμβολίζεται ως $t = \{z_1, z_2, z_3, ..z_i, ., z_s\}, z_i \in \Re^n$. Ο ταξινομητής εκπαιδεύεται στο t και υπολογίζεται η απόδοση του $\hat{\theta}_B, \mathrm{B} = 1,2,3...,\mathrm{k}$ στο T - t. Αν το k είναι αρκετά μεγάλο τότε $\hat{\theta} = \frac{1}{k}\sum_{i=1}^{k}\hat{\theta}_B$ λόγω του κεντρικού οριακού θεωρήματος, όπου $\hat{\theta}$ η απόδοση του ταξινομητή σε όλο το σύνολο δεδομένων.

1.2.9.2. Hold-out

.

Σε αυτή τη μέθοδο το σύνολο δεδομένων T χωρίζεται σε 2 υποσύνολα, το T_{tr} και το T_t, με μεγέθη n_{tr} και n_t αντίστοιχα. Ο ταξινομητής εκπαιδεύεται στο T_{tr} και έπειτα υπολογίζεται η απόδοση του στο T_t. Το T_t αναφέρεται στη βιβλιογραφία και ως hold-out set. Μια παραλλαγή της μεθόδου Hold-out είναι η επανάληψη του τυχαίου διαχωρισμού k φορές , όπου ο ταξινομητής εκπαιδεύεται και ελέγχεται σε όλα τα διαφορετικά T_{tr} και T_t αντίστοιχα. Στη συνέχεια υπολογίζεται η μέση τιμή όλων αποδόσεων που υπολογίστηκαν. Συνήθως το T_{tr} αποτελεί τα $\frac{2}{3}$ του συνόλου δεδομένων και το υπόλοιπο $\frac{1}{3}$ αποτελεί το T_t [6].

1.2.9.3. k-Fold Cross-validation

Σύμφωνα με τη συγκεκριμένη τεχνική το σύνολο των δεδομένων T, μεγέθους N χωρίζεται αρχικά σε k αμοιβαίως αποκλειόμενα υποσύνολα (folds). Όπου το μέγεθος των υποσυνόλων είναι $\frac{N}{k}$. Μετά το διαχωρισμό του συνόλου δεδομένων ο ταξινομητής χρησιμοποιεί $k-1$ από τα υποσύνολα που δημιουργήθηκαν για να εκπαιδευτεί και υπολογίζεται η απόδοση του στο εναπομείναν υποσύνολο[6]. Η διαδικασία αυτή επαναλαμβάνεται k φορές συνολικά έτσι ώστε κάθε υποσύνολο να έχει πάρει τη θέση του test set ακριβώς μια φορά. Στο τέλος υπολογίζεται η μέση τιμή των αποδόσεων που υπολογίστηκαν. Αξιοσημείωτο είναι το γεγονός ότι καθώς το k αυξάνεται , το μέγεθος του test set μειώνεται με αποτέλεσμα την αύξηση της αυστηρότητας του ελέγχου. Η τεχνική αυτή μπορεί να χρησιμοποιηθεί και για σύνολα δεδομένων με μικρό μέγε-

θος, κάτι το οποίο την κάνει να υπερτερεί σε σχέση με τον απλό διαχωρισμό σε train,validation και test set.

1.2.9.4. Leave-one-out

Αυτή η τεχνική αποτελεί μια παραλλαγή του k-Fold Cross Validation όπου το k γίνεται ίσο με το N. Πιο συγκεκριμένα ο ταξινομητής χρησιμοποιεί $N-1$ πρότυπα για να εκπαιδευτεί και το εναπομείναν πρότυπο για να ελεγχθεί . Η διαδικασία επαναλαμβάνεται N φορές μέχρι κάθε πρότυπο να έχει χρησιμοποιηθεί ακριβώς μια φορά για τον έλεγχο του ταξινομητή. Έπειτα υπολογίζεται ο μέσος όρος των αποδόσεων που υπολογίστηκαν[7]. Το γεγονός ότι για τον έλεγχο του ταξινομητή η τεχνική χρησιμοποιεί μόνο ένα πρότυπο, την καθιστά ως μια πολύ αυστηρή τεχνική, όπου η εκτίμηση της απόδοσης του ταξινομητή είναι αρκετά απαισιόδοξη.

1.2.9.5. Stratified k-Fold Cross Validation

Όλες οι παραπάνω μέθοδοι εισάγουν την έννοια της τυχαιότητας κατά τη διαδικασία του διαχωρισμού , όμως το γεγονός αυτό μπορεί να δημιουργήσει προβλήματα. Όπως αναφέρθηκε παραπάνω τα υποσύνολα της εκπαίδευσης και του ελέγχου θα πρέπει να αντιπροσωπεύουν το σύνολο των δεδομένων. Με την εισαγωγή της τυχαιότητας δημιουργούνται κάποια κενά σε αυτή τη προϋπόθεση, καθώς υπάρχει η περίπτωση το τυχαίο σύνολο εκπαίδευσης που θα προκύψει από το διαχωρισμό να μην περιέχει μια ή περισσότερες από τις κλάσεις του προβλήματος, με αποτέλεσμα ο ταξινομητής να μην εκπαιδευτεί ποτέ σε αυτές. Τη λύση του προβλήματος αυτού εξασφαλίζει η συγκεκριμένη μέθοδος. Αυτή η μέθοδος υπολογίζει τα ποσοστά εμφάνισης της κάθε κλάσης στο σύνολο των δεδομένων και δημιουργεί train sets και test sets, όπου ο πληθυσμός κάθε κλάσης μέσα σε αυτά είναι ανάλογος αυτών των ποσοστών που υπολογίσθηκαν για ολόκληρο το σύνολο των δεδομένων[6]. Τα υπόλοιπα βήματα της μεθόδου είναι ακριβώς τα ίδια όπως και στο k-Fold Cross Validation. Πιο συγκριμένα έστω ότι το σύνολο των δεδομένων T, μεγέθους N περιέχει c κλάσεις $\omega_i, i=1,2,3,...,\mathrm{c}\in\mathbb{N}$ με πληθυσμούς $\mathrm{N}(\omega_i)$, τότε ο πληθυσμός της κάθε κλάσης σε κάθε fold θα ισούται με $\frac{N}{k}\cdot\frac{N(\omega_i)}{N}=\frac{N(\omega_i)}{k}$.

1.2.10. Μέτρα απόδοσης των ταξινομητών

Παραπάνω αναπτύχθηκαν μερικές από τις πιο γνωστές μεθόδους για το διαχωρισμό των δεδομένων σε training και test set. Όπως προαναφέρθηκε αυτός ο διαχωρισμός πραγματοποιείται κατά τη διάρκεια της διαδικασίας της ταξινόμησης ώστε να είναι δυνατός ο υπολογισμός της απόδοσης του ταξινομητή σε δεδομένα τα οποία δεν έχει ξαναδεί (το test set) και να αντικατοπτρίζεται μέσα από αυτά η πραγματικότητα.

Η έννοια της απόδοσης ενός μοντέλου στην αναγνώριση προτύπων είναι στενά συνυφασμένη με την έννοια της ακρίβειας ενός μοντέλου, δηλαδή το ποσοστό των προτύπων τα οποία το μοντέλο ταξινόμησε στις πραγματικές κλάσεις τους (classification accuracy). Πιο συγκεκριμένα έστω ότι ο αριθμός των ορθά ταξινομημένων προτύπων

είναι $N_{correctly\_classified\_patterns}$ και ο συνολικός αριθμός των προτύπων τα οποία αποτελούν το σύνολο δεδομένων είναι $N_{patterns}$, τότε η ακρίβεια της ταξινόμησης θα ισούται με :

$$Accuracy = \frac{N_{correctly\_classified\_patterns}}{N_{patterns}} \quad (1.2)$$

Ο υπολογισμός της ακρίβειας της ταξινόμησης είναι μια εύκολη υπόθεση, όμως για να έχει νόημα η τιμή αυτή θα πρέπει όπως και προαναφέρθηκε το test set να είναι αντιπροσωπευτικό δείγμα του συνόλου δεδομένων. Εκτός την ακρίβεια υπάρχουν και άλλα μέτρα απόδοσης ενός ταξινομητή, όπως το Precision, το Recall, το F1, το Hamming Loss, το Cohen Cappa, ο συντελεστής συσχέτισης Mathews και ο συντελεστής Gini, τα οποία θα παρουσιαστούν παρακάτω.

Σε πολλά προβλήματα ταξινόμησης δεν έχει τόσο σημασία η αύξηση του πληθυσμού των ορθά ταξινομημένων προτύπων, αλλά η μείωση του πληθυσμού των λανθασμένα ταξινομημένων προτύπων (πχ στην ιατρική η λανθασμένη διάγνωση του προβλήματος ενός ασθενή είναι πιο άξια σημασίας από ότι η ορθή διάγνωση του, καθώς η λανθασμένη διάγνωση σημαίνει περισσότερα προβλήματα για τον ασθενή ή και το θάνατο του). Στη συνέχεια θα παρουσιαστεί και θα ορισθεί ο confusion matrix μέσα από τον οποίο προκύπτουν πολλά από τα μέτρα απόδοσης που αναφέρθηκαν παραπάνω.

1.2.10.1. Confusion Matrix

Κατά διάρκεια της διεξαγωγή ενός στατιστικού ελέγχου υποθέσεων υπάρχουν 4 ενδεχόμενα στα οποία μπορεί να καταλήξει ο έλεγχος, ανάλογα με το αποτέλεσμα της απόφασης της μηδενικής υπόθεσης (αποδοχή ή απόρριψη της) και το αν πραγματικά ισχύει ή δεν ισχύει η μηδενική υπόθεση . Πιο συγκεκριμένα έστω ότι H_0 είναι η μηδενική υπόθεση και H_1 η εναλλακτική υπόθεση τότε τα ενδεχόμενα που αναφέρθηκαν παραπάνω θα είναι :

1. Το ενδεχόμενο η H_0 να γίνει αποδεχτή και στη πραγματικότητα να ισχύει η H_0.
2. Το ενδεχόμενο να απορριφτεί η H_0 και στη πραγματικότητα να ισχύει η H_1.
3. Το ενδεχόμενο η H_0 να γίνει αποδεχτή ενώ στη πραγματικότητα ισχύει η H_1.
4. Το ενδεχόμενο να απορριφτεί η H_0 και στην πραγματικότητα να ισχύει η H_0 .

Από τη στατιστική γνωρίζουμε τις περιπτώσεις 1 και 2 ως στατιστικό σφάλμα τύπου I και τύπου II. Όλες οι παραπάνω περιπτώσεις μπορούν να οργανωθούν στο παρακάτω πίνακα τύπου σφαλμάτων (Πίνακας 1.2.10.1-1):

Πίνακας 1.2.10.1-1
Πίνακας τύπων στατιστικού σφάλματος .

Πίνακας τύπων σφάλματος		Η μηδενική υπόθεση (H_0) είναι	
		Αληθής	Ψευδής
Απόφαση της μηδενικής υπόθεσης(H_0)	Απόρριψη	Ψευδώς Θετικό (Τύπου Ι σφάλμα)	Αληθώς Θετικό
	Αποτυχία Απόριψης	Αληθώς Αρνητικό	Ψευδώς Αρνητικο (Τύπου ΙΙ σφάλμα)

Στη διαδικασία της ταξινόμησης και δη της δυαδικής ταξινόμησης, κάθε απόπειρα ενός ταξινομητή να αντιστοιχίσει ένα πρότυπο με μια κλάση αποτελεί μια απόφαση. Σε αυτή τη περίπτωση η μηδενική υπόθεση H_0 είναι ότι το πρότυπο ανήκει στην αρνητική κλάση και η εναλλακτική υπόθεση H_1 ότι ανήκει στη θετική κλάση.

Σκοπός μας όμως είναι να μετρήσουμε το πληθυσμό των προτύπων που ανήκουν σε κάθε περίπτωση από τις προαναφερθέντες και ταυτόχρονα ανήκουν στο test set. Για αυτό το λόγο ο παρακάτω πίνακας μετασχηματίζεται σε ένα πίνακα σύγχυσης (confusion matrix, Πίνακας 1.2.10.1-2) όπου κάθε κελί πια αναφέρεται στο πληθυσμό της κάθε περίπτωσης του πίνακα τύπου στατιστικού σφάλματος[8].

Πίνακας 1.2.10.1-2
Confusion Matrix(Πίνακας Σύγχυσης).

Actuall Class	Classifier Decision	
	Positive	Negative
True	TP	TN
False	FP	FN

Τα στοιχεία του παραπάνω πίνακα ερμηνεύονται ως εξής:

- True Positives (TP): Είναι το σύνολο των προτύπων όπου ορθά ταξινομήθηκαν στη θετική κλάση.
- True Negatives (TN): Είναι το σύνολο των προτύπων όπου ορθά ταξινομήθηκαν στην αρνητική κλάση.
- False Positives (FP): Είναι το σύνολο των προτύπων τα οποία λανθασμένα ταξινομήθηκαν στην θετική κλάση.
- False Negatives (FN): Είναι το σύνολο των προτύπων τα όποια λανθασμένα ταξινομήθηκαν στην αρνητική κλάση.

Ο παραπάνω πίνακας μπορεί να μεταφερθεί στο πρόβλημα του multi-class classification , όπως παρουσιάζεται παρακάτω. Έστω ένα πρόβλημα ταξινόμησης, το οποίο αποτελείται από $c \in \mathbb{N}$ κλάσεις $\omega_j, j = 1,2,3,....,c$ και ένα ταξινομητή D, τότε $a_{ij} \in \mathbb{N}$ είναι το σύνολο των προτύπων, τα οποία πραγματικά ανήκουν στην κλάση i και ο ταξινομητής D τα ταξινόμησε στη κλάση j. Επομένως ο confusion matrix διαμορφώνεται ως εξής (Πίνακας 1.2.10.1-3):

Πίνακας 1.2.10.1-3
Confusion Matrix στο πρόβλημα του multi-class classification.

Actuall Class	Classifier Decision					
	ω_1	ω_2	.	ω_j	.	ω_c
ω_1	a_{11}	a_{12}	.	a_{1j}	.	a_{1c}
ω_2	a_{21}	a_{22}	.	a_{2j}	.	a_{2c}
.	.	.	.	.	.	.
ω_i	a_{i1}	a_{i2}	.	a_{ij}	.	a_{ic}
.	.	.	.	.	.	.
ω_c	a_{c1}	a_{c2}	.	a_{cj}	.	a_{cc}

Με βάση των παραπάνω πίνακα μπορεί εύκολα να υπολογιστεί η ακρίβεια του ταξινομητή με το παρακάτω τύπο:

$$Accuracy = \frac{\sum_{i=j}^{c} \omega_{ij}}{\sum_{i=1}^{c}\sum_{j=1}^{c} \omega_{ij}}, c \in \mathbb{N} \qquad (1.3)$$

Παρακάτω θα ορισθούν κάποια μέτρα, τα οποία βασίζονται στα TP,TN,FP,FN που αναφέρθηκαν παραπάνω και είναι πολύ σημαντικά για το υπολογισμό της απόδοσης των ταξινομητών.

1.2.10.1.1. Precision

Το Precision ή αλλιώς Positive Predictive Power είναι ο λόγος των προτύπων τα οποία ορθά ταξινομήθηκαν στη θετική κλάση προς το σύνολο των προτύπων τα οποία ταξινομήθηκαν στη θετική κλάση[9].

$$\text{Precision} = \frac{TP}{TP + FP} \qquad (1.4)$$

1.2.10.1.2. Recall

Το Recall (Sensitivity,True Positive Rate, πιθανότητα ανίχνευσης) είναι ο λόγος των προτύπων τα οποία ορθά ταξινομήθηκαν στη θετική κλάση προς το σύνολο των προτύπων των οποίων η πραγματική κλάση είναι η θετική[9].

$$\text{Recall} = \frac{TP}{TP + FN} \qquad (1.5)$$

1.2.10.1.3. F1 Score

Το F1 Score είναι ο αρμονικός μέσος των Precision και Recall[9].

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} = \frac{2TP}{2TP + FP + FN} \qquad (1.6)$$

1.2.10.1.4. Micro-Macro Average

Οι παραπάνω τύποι των Precision, Recall, F1 Score ισχύουν στα πλαίσια της δυαδικής ταξινόμησης (binary classification). Ο υπολογισμός τους για το πρόβλημα του multi-class classification μπορεί να γίνει μόνο μέσω των εννοιών του micro και macro μέσου.

Η λογική πίσω από τους micro και macro μέσους είναι ότι υπολογίζονται τα TP,FP,FN για κάθε κλάση στα πλαίσια της δυαδικής ταξινόμησης, δηλαδή κάθε κλάση $\omega_i, i = 1,2,3,...,c \in \mathbb{N}$ θεωρείται ως η θετική κλάση και το σύνολο όλων των υπόλοιπων ως η αρνητική κλάση[9][10].

Έπειτα για το micro μέσο υπολογίζεται το άθροισμα των TP,FP,FN για όλες τις κλάσεις και με αυτά υπολογίζονται τα Precision,Recall και F1 Score. Πιο συγκεκριμένα θα ισχύουν τα παρακάτω:

$$TP_c = \sum_{i=1}^{c} TP_{\omega_i} \qquad (1.7)$$

$$FP_c = \sum_{i=1}^{c} FP_{\omega_i} \qquad (1.8)$$

$$FN_c = \sum_{i=1}^{c} FN_{\omega_i} \qquad (1.9)$$

$$\text{Precision}_c = \frac{TP_c}{TP_c + FP_c} \qquad (1.10)$$

$$\text{Recall}_c = \frac{TP_c}{TP_c + FN_c} \qquad (1.11)$$

$$F1_c = 2 \cdot \frac{\text{Precision}_c \cdot \text{Re}call_c}{\text{Precision}_c + \text{Re}call_c} = \frac{2TP_c}{2TP_c + FP_c + FN_c} \qquad (1.12)$$

Στη περίπτωση του macro μέσου θα ισχύουν τα εξής:

$$\text{Precision}_{\omega_i} = \frac{TP_{\omega_i}}{TP_{\omega_i} + FP_{\omega_i}} \qquad (1.13)$$

$$\text{Re}call_{\omega_i} = \frac{TP_{\omega_i}}{TP_{\omega_i} + FN_{\omega_i}} \qquad (1.14)$$

$$F1_{\omega_i} = 2 \cdot \frac{\text{Precision}_{\omega_i} \cdot \text{Re}call_{\omega_i}}{\text{Precision}_{\omega_i} + \text{Re}call_{\omega_i}} \qquad (1.15)$$

$$\text{Precision}_c = \frac{1}{c} \cdot \sum_{i=1}^{c} \text{Precision}_{\omega_i} \qquad (1.16)$$

$$\text{Recall}_c = \frac{1}{c} \cdot \frac{x-\mu}{\sigma} \sum_{i=1}^{c} \text{Recall}_{\omega_i} \qquad (1.17)$$

$$F1_c = \frac{1}{c} \cdot \sum_{i=1}^{c} F1_{\omega_i} \qquad (1.18)$$

1.2.10.1.5. Mathews Correlation Coefficient

Ο συντελεστής συσχέτισης Mathews είναι στην ουσία ένα μέτρο συσχέτισης των πραγματικών κλάσεων των προτύπων που συμμετέχουν στον έλεγχο και των κλάσεων που έχουν προκύψει από τη διαδικασία της ταξινόμησης. Επίσης όπως και τα παραπάνω μέτρα είναι βασισμένος στους αριθμούς TP,FP,FN και παίρνει τιμές στο $[-1,1]$, όπου το 1 σημαίνει ότι οι πραγματικές κλάσεις είναι πλήρως συσχετισμένες με τις κλάσεις που προέκυψαν από τη ταξινόμηση και το -1 ότι υπάρχει αρνητική συσχέτιση μεταξύ αυτών των δύο. Ο συντελεστής συσχέτισης Mathews στα πλαίσια του binary classification προκύπτει από το παρακάτω τύπο[11]:

$$MCC = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}} \qquad (1.19)$$

Παρακάτω ορίζεται ο MCC (Mathews Correlation Coeficient) στα πλαίσια του multi-class classification:

Έστω $X, Y \in M(S \mid N, F2)$ δύο πίνακες όπου $X_{sn} = 1$ αν το πρότυπο s ταξινομήθηκε στη κλάση n ($pc(s) = n$) αλλιώς $X_{sn} = 0$ και $Y_{sn} = 1$ αν η πραγματική κλάση του προτύπου s είναι η n αλλιώς $Y_{sn} = 0$. Χρησιμοποιώντας το δέλτα του Kronecker, το οποίο στη περίπτωση της άλγεβρας ισούται με το μοναδιαίο πίνακα, τα παραπάνω μπορούν να γραφούν ως εξής:

$$X = (\delta_{pc(s),n})_{sn}, Y = (\delta_{tc(s),n})_{sn} \qquad (1.20)$$

Επίσης $S = \sum_{k,l=1} C_{kl}$, όπου $C_{kk} = \left|\{s \in S : \mathrm{X}_{\mathrm{sk}} = Y_{sk} = 1\}\right| = \sum_{s=1}^{S} \mathrm{X}_{\mathrm{sk}} Y_{sk}$ (ο αριθμός των προτύπων που η πραγματική κλάση είναι η k αλλά και η προβλεπόμενη κλάση είναι η k) και για $k \neq l, \mathrm{C}_{\mathrm{kl}} = \left|\{s \in S : X_{sk} = 1 \,\&\, \mathrm{Y}_{sl} = 1\}\right|$ (ο αριθμός των προτύπων που η πραγματική κλάση είναι η l ενώ η προβλεπόμενη κλάση είναι η k).

Η συνδιασπορά των X και Y γράφεται ως εξής:

$$\mathrm{cov}(\mathrm{X}, \mathrm{Y}) = \sum_{k=1}^{N} w_k \,\mathrm{cov}(X_k, Y_k) = \frac{1}{N} \sum_{s=1}^{S} \sum_{k=1}^{N} (X_{sk} - \bar{X}_k)(\mathrm{Y}_{sk} - \bar{Y}_k) \quad (1.21)$$

Όπου $w_k = \frac{1}{N}$ και $\bar{X}_k, \bar{Y}_k$ οι μέσοι όροι κάθε κλάσης στους πίνακες X, Y ,οι οποίοι ορίζονται ως $\bar{X}_k = \frac{1}{S} \sum_{s=1}^{S} X_{sk} = \frac{1}{S} \sum_{l=1}^{N} C_{kl}$ και $\bar{Y}_k = \frac{1}{S} \sum_{s=1}^{S} Y_{sk} = \frac{1}{S} \sum_{l=1}^{N} C_{lk}$ αντίστοιχα.

Τέλος ο MCC στα πλαίσια του mutli-class classification ορίζεται ως[10]:

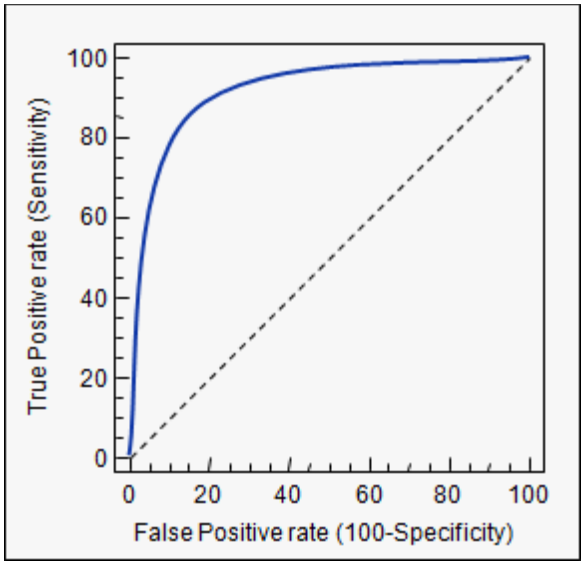
$$MCC = \frac{\text{cov}(X,Y)}{\sqrt{\text{cov}(X,X)\cdot\text{cov}(Y,Y)}} \qquad (1.22)$$

Ο συντελεστής MCC είναι ίσως το μόνο μέτρο το οποίο εκμεταλλεύεται την πληροφορία η οποία εξάγεται από το confusion matrix με το καλύτερο δυνατό τρόπο.

1.2.10.1.6. Ανάλυση Receiver Operating Characteristic (ROC)

Κατά τον υπολογισμό της απόδοσης ενός ταξινομητή όπως αναφέρθηκε παραπάνω δεν αρκεί ο υπολογισμός του ποσοστού των ορθά ταξινομημένων προτύπων (ακρίβεια). Το παραπάνω βασίζεται στο γεγονός ότι η ακρίβεια του ταξινομητή δεν εμπεριέχει την έννοια της κατανομής των κλάσεων. Για παράδειγμα σε ένα σύνολο δεδομένων το οποίο αποτελείται κατά 80% από τη θετική κλάση και 20% από την αρνητική κλάση, ο ταξινομητής που θα προκύψει μετά τη φάση της εκπαίδευσης θα είναι σε θέση να προβλέψει με μεγάλη ακρίβεια τα πρότυπα τα οποία ανήκουν στη πρώτη κλάση, αλλά δεν θα ισχύει το ίδιο για την αρνητική κλάση. Επομένως θα πρέπει εκτός από το ποσοστό των σωστά ταξινομημένων προτύπων τα οποία ανήκουν στη θετική κλάση (TPR), να συμπεριληφθούν στην απόδοση και αυτά τα οποία λανθασμένα ταξινομήθηκαν στη θετική κλάση (FPR). Κάτι τέτοιο όμως είναι αδύνατο να ανιχνευθεί από το μέτρο της ακρίβειας του ταξινομητή, γιατί η σύγκριση γίνεται με ολόκληρο το σύνολο δεδομένων. Για αυτό το λόγο τα παραπάνω ποσοστά θα πρέπει να μελετούνται σε ζεύγη.

Το παραπάνω πρόβλημα ήρθαν να λύσουν οι καμπύλες ROC, οι οποίες πρωτοεμφανίστηκαν το 1940 και χρησιμοποιούνταν για τον υπολογισμό της ακρίβεια με την οποία, μπορούν να ανιχνευθούν τα σήματα ενός sonar υπό την ύπαρξη θορύβου. Πιο συγκεκριμένα κάθε ζεύγος (TPR , FPR = $FP/N_{positive}$) αποτελεί ένα σημείο της καμπύλης ROC και το γράφημα ROC(Εικόνα 1.2.10.1.6-1) είναι η γραφική αναπαράσταση των TPR,FPR [12] .



Εικόνα 1.2.10.1-6: Αναπαράσταση καμπύλης ROC.

Όπως βλέπουμε στο παραπάνω γράφημα στον κατακόρυφο άξονα έχουμε το TPR και στον οριζόντιο το FPR. Το σύνολο των ζευγών (TPR,FPR) αποτελούν τη καμπύλη ROC (μπλε γραμμή) και η διαγώνιος ορίζει τη τυχαιότητα. Αν η καμπύλη ενός ταξινομητή βρίσκεται κάτω από τη διαγώνιο τότε η ακρίβεια του ταξινομητή είναι κάτω του μετρίου δηλαδή ο ταξινομητής παράγει τυχαία αποτελέσματα.

Αξιοσημείωτα στη παραπάνω καμπύλη είναι τα σημεία (0,0) , (1,1) και (1,0). Το σημείο (0,0) υποδεικνύει τη περίπτωση όπου όλα τα πρότυπα ταξινομούνται στην αρνητική κλάση, το σημείο (1,1) υποδεικνύει τη περίπτωση όπου όλα τα πρότυπα ταξινομούνται στη θετική κλάση και το σημείο (1,0) υποδεικνύει την ιδανική περίπτωση όπου όλα τα πρότυπα ταξινομούνται ορθά και υπάρχει τέλειος διαχωρισμός μεταξύ των κλάσεων.

Για το σχεδιασμό της καμπύλης ROC ο ταξινομητής ο οποίος ελέγχεται, πρέπει αρχικά να αποφανθεί για τη τιμή της πιθανότητας κάθε πρότυπου με την οποία αυτό ανήκει στη θετική κλάση. Οι πιθανότητες αυτές ταξινομούνται σε φθίνουσα σειρά και έπειτα ακολουθείται η παρακάτω επαναληπτική διαδικασία:

- Σε κάθε βήμα ορίζεται ένα κατώφλι P σύμφωνα με το οποίο διευκρινίζεται για κάθε πρότυπο η κλάση που ανήκει. Δηλαδή αν ισχύει $P \le P_i$ τότε το πρότυπο i ανήκει στη θετική κλάση, ενώ αν ισχύει το αντίθετο τότε ανήκει στην αρνητική κλάση.
- Το κατώφλι επιλέγεται σειριακά από τη λίστα με τις πιθανότητες που αναφέρθηκαν παραπάνω.
- Στη συνέχεια υπολογίζονται τα TPR και FPR για τα δεδομένα που προέκυψαν και σημειώνεται το σημείο (TPR,FPR) στο γράφημα.
- Τα σημεία (0,0) και (1,1) αποτελούν ειδικές περιπτώσεις όπου το κατώφλι ορίζεται ως 1 και 0 αντίστοιχα.

Από τα παραπάνω μπορεί κανείς εύκολα να συμπεράνει ότι η καμπύλη ενός ιδανικού ταξινομητή θα συγκεντρώνεται στο πάνω αριστερά μέρος του γραφήματος (TPR = 1, FPR = 0). Για τον πιο ακριβή υπολογισμό της απόδοσης ενός ταξινομητή και της σύγκρισης διαφορετικών ταξινομητών υπολογίζεται το AUC(Area Under the Curve) της καμπύλης ROC, δηλαδή το εμβαδό κάτω από τη καμπύλη ROC. Το AUCROC ενός ιδανικού ταξινομητή θα είναι πάντα 1 και το AUCROC ενός ταξινομητή ο οποίος παράγει τυχαίες προβλέψεις θα είναι 0.5 (το εμβαδό κάτω από τη διαγώνιο).

Τα παραπάνω ισχύουν στη περίπτωση του binary classification, παρακάτω ακολουθεί η γενίκευση του AUCROC σε προβλήματα multi-class classification:

Έστω ότι στη παραπάνω αναφερόμενη λίστα με τις πιθανότητες υπάρχουν n_1 πρότυπα τα οποία ανήκουν στη αρνητική κλάση και n_0 πρότυπα τα οποία ανήκουν στη θετική κλάση. Τότε η κατάταξη του i-οστού προτύπου το οποίο ανήκει στη θετική κλάση συμβολίζεται ως $r_i, i = 1, 2, ..., n_0$ (Πίνακας 1.2.10.1.6-1) και το AUCROC συμβολίζεται ως $\hat{A}$.

Πίνακας 1.2.10.1.6-1
Παράδειγμα κατάταξης προτύπων με βάση τη πιθανότητα να ανήκουν στην θετική κλάση.

	-	-	-	-	+	-	+	+	+	+
i					**1**		**2**	**3**	**4**	**5**
r_i					**5**		**7**	**8**	**9**	**10**

$$\hat{A} = \frac{S_0 - n_0(n_0+1)/2}{n_0 n_1} \qquad (1.23)$$

S_0 : Το άθροισμα των κατατάξεων των προτύπων που ανήκουν στη θετική κλάση

Επομένως στο πρόβλημα του multi-class classification το AUCROC συμβολίζεται ως M και ορίζεται ως[13]:

$$\left.\begin{aligned} M &= \frac{2}{c(c-1)} \sum_{i<j} \hat{A}(i,j) \\ \hat{A}(i,j) &= \left[\hat{A}(i \mid j) + \hat{A}(j \mid i)\right]/2 \end{aligned}\right\} \qquad (1.24)$$

$\hat{A}(i \mid j)$: Η πιθανότητα ένα τυχαίο δείγμα της κλάσης j να έχει μικρότερη πιθανότητα να ανήκει στη κλάση i από ότι ένα τυχαίο δείγμα της κλάσης i .

1.2.10.2. Hamming Loss

Το Hamming Loss είναι ένα μέτρο το οποίο σχεδιάστηκε για να χρησιμοποιηθεί σε προβλήματα multi-label classification[14]. Πιο συγκεκριμένα έστω D ένα υποσύνολο δεδομένων ελέγχου με πολλαπλές ετικέτες, το οποίο αποτελείται από $|D|$ multi-label πρότυπα $(x_i, Y_i), i = 1..|D|, Y_i \subseteq L$. Αν H είναι ένας mutli-label ταξινομητής και $dZ_i = H(x_i)$ το σύνολο των ετικετών στις οποίες ταξινομήθηκε το πρότυπο x_i ,τότε το Hamming Loss ορίζεται ως :

$$HM(H,D) = \frac{1}{|D|} \sum_{i=1}^{|D|} \frac{|Y_i \Delta Z_i|}{|L|} \qquad (1.25)$$

Όπου Δ είναι η συμμετρική διαφορά δύο συνόλων και αντιστοιχεί στο τελεστή *XOR* (επιστρέφει 1 αν τα σύνολα είναι ακριβώς τα ίδια σε αντικείμενα αλλά και σειρά, αλλιώς 0) από την άλγεβρα Μπουλ.

Το Hamming Loss στο πρόβλημα του multi-class classification μετατρέπεται στην απόσταση Hamming (Hamming Distance). Αν Y_{pred} είναι το διάνυσμα με τις κλάσεις, τις οποίες αντιστοίχισε ο ταξινομητής με τα πρότυπα του συνόλου ελέγχου (test set) και Y_{true} είναι το διάνυσμα με τις πραγματικές κλάσεις των προτύπων τότε η απόσταση Hamming ορίζεται ως ο αριθμός των θέσεων που τα δύο διανύσματα διαφέρουν και συμβολίζεται με $d(Y_{pred}, Y_{true})$.

1.2.10.3. Cohen's Kappa

Το Cohen's Kappa στην ουσία αποτελεί ένα μέτρο συμφωνίας/ασυμφωνίας με βάση την απόσταση των προβλεπόμενων κλάσεων από το ταξινομητή και των πραγματικών κλάσεων των προτύπων. Παρακάτω ακολουθεί ο ορισμός του:

Έστω ότι 2 παρατηρητές ταξινομούν ανεξάρτητα ο ένας με τον άλλο n παρατηρήσεις σε c κατηγορίες (κλάσεις), τότε οι ταξινομήσεις μπορούν να οργανωθούν σε ένα πίνακα διασταυρωμένης ταξινόμησης (Πίνακας 1.2.10.3-1).

Πίνακας 1.2.10.3-2
Διασταυρωμένη ταξινόμηση 3x3

Γραμμές	Στήλες			Άθροισμα Στηλών
	1	2	3	
1	P_{11}	P_{12}	P_{13}	$P_{1.}$
2	P_{21}	P_{22}	P_{23}	$P_{2.}$
3	P_{31}	P_{32}	P_{33}	$P_{3.}$
Άθροισμα Γραμμών	$P_{.1}$	$P_{.2}$	$P_{.3}$	$P_{..} = 1.0$

Σύμφωνα με το παραπάνω πίνακας το Cohen's Kappa ορίζεται ώς[15]:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \qquad (1.26)$$

Όπου $P_o = \sum_{i=1}^{c} P_{ii}$ και $P_e = \sum_{i=1}^{c} P_{\cdot i} \cdot P_{i\cdot}$

Το P_o είναι το παρατηρούμενο ποσοστό των παρατηρήσεων στο οποίο οι δύο παρατηρητές συμφωνούν, P_e είναι το ποσοστό των παρατηρήσεων για τα οποία οι παρατηρητές αναμένεται να συμφωνήσουν λόγω τύχης, $P_o - P_e$ είναι το ποσοστό των παρατηρήσεων για το οποίο οι παρατηρητές συμφωνούν πέρα από αυτό που αναμένεται λόγω τύχης, $1 - P_e$ είναι το μέγιστο πιθανό ποσοστό για το οποίο οι παρατηρητές συμφωνούν πέρα από το γεγονός ότι μπορεί να συμφωνήσουν λόγω τύχης και το κ είναι το ποσοστό συμφωνίας των παρατηρητών αφού αφαιρεθεί ο παράγοντας της τύχης. Επίσης το $\delta = 1 - P_o$ αντιπροσωπεύει το παρατηρούμενο ποσοστό διαφωνίας και το $\mu_\delta = 1 - P_e$ αντιπροσωπεύει το εκτιμώμενο ποσοστό διαφωνίας.

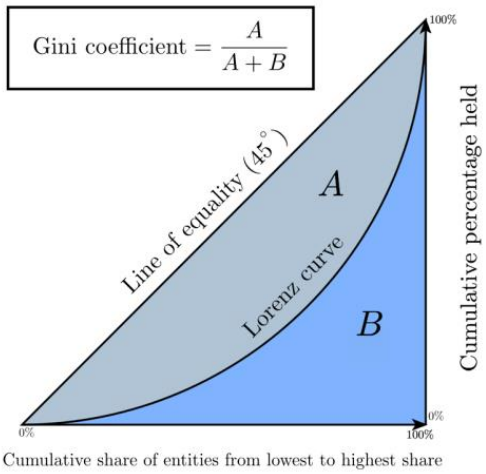
Με βάση αυτούς του ορισμούς ο τύπος του Cohen's Kappa γίνεται:

$$\kappa = 1 - \frac{\delta}{\mu_\delta} \qquad (1.27)$$

Επομένως το Cohen's kappa μπορεί να χρησιμοποιηθεί ως ένα μέτρο για τη μέτρηση της ασυμφωνίας δύο παρατηρητών όπου ο υπολογισμός των P_o, P_e γίνεται με τη καταμέτρηση των θέσεων συμφωνίας και ασυμφωνίας. Στη περίπτωση της ταξινόμησης η έννοια της ασυμφωνίας των παρατηρητών ερμηνεύεται ως η ασυμφωνία μεταξύ των κλάσεων των προτύπων που ταξινομήθηκαν από το ταξινομητή και των πραγματικών κλάσεων των προτύπων (η ευκλείδεια απόσταση μεταξύ τους).

1.2.10.4. Συντελεστής Gini – Καμπύλη Lorenz

Η καμπύλη Lorenz αποτελεί ένα μέσο για την αναπαράσταση της ανισότητας μεταξύ των τιμών μιας κατανομής συχνοτήτων (Εικόνα 1.2.10.4-1). Η καμπύλη Lorenz χρησιμοποιείται αρκετά στην οικονομική επιστήμη για την αναπαράσταση της ανισότητας του εισοδήματος μεταξύ των μελών ενός πληθυσμού. Κάθε σημείο της καμπύλης Lorenz αναπαριστά το ποσοστό του συνολικού εισοδήματος του πληθυσμού το οποίο λαμβάνει ένα ποσοστό του πληθυσμού (πχ το κατώτερο 20% όλων των νοικοκυριών λαμβάνει το 10% του συνολικού εισοδήματος).



Εικόνα 1.2.10.4-1: Αναπαράσταση της καμπύλης Lorenz και ο συντεκεστής Gini.

Όπως φαίνεται και στο παραπάνω γράφημα ο κατακόρυφος άξονας υποδεικνύει την αθροιστική σειρά της ποσοστιαίας κατανομής του εισοδήματος (από τα χαμηλότερα στα υψηλότερα εισοδήματα) και ο οριζόντιος άξονας τη αθροιστική σειρά της ποσοστιαίας κατανομής του πληθυσμού. Η διαγώνιος υποδεικνύει την ιδανική περίπτωση όπου το εισόδημα είναι ισότιμα κατανεμημένο σε όλο το πληθυσμό.

Ο συντελεστής Gini στην ουσία αποτελεί ένα μέτρο υπολογισμού αυτής της ανισότητας και ορίζεται ως εξής[16]:

$$Gini\ Coefficient = \frac{A}{A+B} \qquad (1.28)$$

A: Το εμβαδό μεταξύ της καμπύλης Lorenz και της διαγωνίου.

B:Το εμβαδό μεταξύ της καμπύλης Lorenz και των αξόνων.

Στη περίπτωση της ταξινόμησης ο όρος του εισοδήματος αντικαθίσταται από τη πιθανότητα ένα πρότυπο να ανήκει στη θετική κλάση.

Ο σχεδιασμός της καμπύλης γίνεται ως εξής:

1. Αρχικά υπολογίζονται από το ταξινομητή για κάθε πρότυπο οι πιθανότητες να ανήκουν στη θετική κλάση.

2. Έπειτα συγκεντρώνονται σε ένα πίνακα μαζί με τις πραγματικές τιμές και ο πίνακας ταξινομείται με βάση τη στήλη των προβλεπόμενων πιθανοτήτων (σε φθίνουσα σειρά).
3. Στη συνέχεια υπολογίζεται η αθροιστική σειρά των ταξινομημένων πια πραγματικών κλάσεων.
4. Ακόμα υπολογίζεται η αθροιστική σειρά του αριθμού των προβλεπόμενων πιθανοτήτων.
5. Χρησιμοποιώντας τα σημεία που δημιουργήθηκαν σχεδιάζεται η καμπύλη Lorenz αφού πρώτα κανονικοποιηθούν οι άξονες του γραφήματος.

Τέλος ολοκληρώνοντας τις κατάλληλες καμπύλες υπολογίζονται τα A,B καθώς και ο συντελεστής Gini.

Αξιοσημείωτο είναι το γεγονός της ύπαρξης σχέσεως μεταξύ του συντελεστή Gini και του AUCROC. Πιο συγκεκριμένα ισχύει[13]:

$$Gini\ Coefficient\ =\ 2\times AUCROC-1 \qquad (1.29)$$

Η παραπάνω σχέση ισχύει και στη περίπτωση του multi-class classification.

1.2.11. Bias vs Variance

Παραπάνω αναφέρθηκαν οι όροι overfitting και underfitting ως αποτελέσματα, αλλά δε δόθηκε βάση στα αίτια πίσω από αυτά. Επομένως θα θεωρούνταν παράλειψη αν δεν τα αναλύαμε.

Έστω ένας ταξινομητής C, το σύνολο δεδομένων T, το σύνολο εκπαιδεύσεως T_{tr} και το σύνολο ελέγχου T_t τότε μπορεί δειχθεί ότι το σφάλμα που κάνει ο C στη προσπάθεια του να ταξινομήσει το T_t εξαρτάται από τους παρακάτω όρους:

- Το Bias, $B(x)$: το οποίο είναι το σφάλμα το οποίο παράγεται από τη διαφορά της πραγματικής κλάσης ενός προτύπου και της πρόβλεψης του ταξινομητή C. Το Bias δίνεται από το τύπο:

$$B(\mathrm{x}) = \mathrm{E}\left[\hat{\theta}(x) - \theta(x)\right] \quad (1.30)$$

- Το Variance, $V(x)$: το οποίο είναι το σφάλμα, το οποίο παράγεται από τη διασπορά των προβλέψεων του ταξινομητή. Το Variance δίνεται από το τύπο:

$$V(x) = E\left[\hat{\theta}(x)^2\right] - E\left[\hat{\theta}(x)\right]^2 \qquad (1.31)$$

- Ο θόρυβος, $N(x)$: ο οποίος συμβολίζει το σφάλμα το οποίο εξαρτάται από τυχαίους παράγοντες και δε μπορεί να μοντελοποιηθεί.

Όπου $\hat{\theta}(x)$ είναι η εκτίμηση της κλάσης του προτύπου x από το ταξινομητή C και $\theta(x)$ η πραγματική κλάση του πρότυπου.

Η σχέση που συνδέει τους τρεις αυτούς όρους είναι η εξής[17]:

$$error = c_1 N(x) + B(x) + c_2 V(x) \qquad (1.32)$$

Τα c_1, c_2 είναι σταθερές και εξαρτώνται από το μέτρο απόδοσης που επιλέγεται για την εκτίμηση του σφάλματος, καθώς και τη φύση του προβλήματος ταξινόμησης.

Είναι γνωστό ότι αν ένας ταξινομητής έχει υψηλό Bias τότε αυτό σημαίνει ότι δεν καταφέρνει να συλλάβει τις σχέσεις μεταξύ των χαρακτηριστικών, διότι έχει κάνει πολλές λανθασμένες υποθέσεις για τα δεδομένα με αποτέλεσμα να κάνει underfitting. Αντίθετα το υψηλό Variance υποδεικνύει ότι μικρές αλλαγές στο T_{tr} θα επιφέρουν μεγάλες αλλαγές στις προβλέψεις του ταξινομητή, δηλαδή ο ταξινομητής μαθαίνει το θόρυβο που περιέχεται στα δεδομένα, με αποτέλεσμα να κάνει overfitting. Επομένως δημιουργείται ένα δίλλημα για το αν ένας ταξινομητής πρέπει να έχει υψηλό Bias ή υψηλό Variance και πως πρέπει να κατανεμηθεί το σφάλμα του ταξινομητή ώστε να μπορεί να πάρει όλη τη πληροφορία από το T_{tr} αλλά να μπορεί ταυτόχρονα να γενικεύσει τη γνώση του και σε ένα άγνωστο σε αυτό T_t. Το παραπάνω αποτελεί ένα από τα κυριότερα προβλήματα στο σχεδιασμό μοντέλων ταξινόμησης καθώς και στην επιλογή των παραμέτρων τους.

ΚΕΦΑΛΑΙΟ 2

2.1 ΘΕΩΡΗΤΙΚΟ ΥΠΟΒΑΘΡΟ ΤΑΞΙΝΟΜΗΤΩΝ

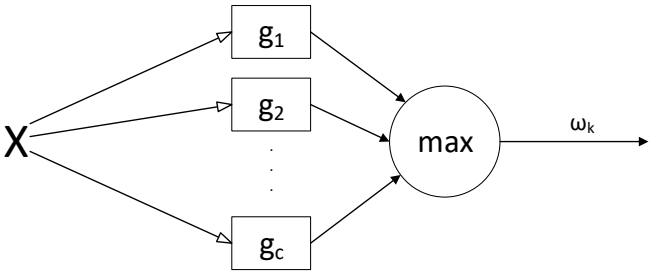
2.1.1. Εισαγωγή στη θεωρία των ταξινομητών

Στο παρών κεφάλαιο θα παρουσιαστούν εν συντομία οι αλγόριθμοι των ταξινομητών του πρώτου σταδίου ταξινόμησης (base learners, weak learners), καθώς και η βασική θεωρία που διέπει τη λειτουργία τους. Οι ταξινομητές του πρώτου σταδίου ταξινόμησης δρουν ανεξάρτητα, αδιαφορώντας για την ύπαρξη άλλων ταξινομητών. Η απόδοση των ταξινομητών του πρώτου σταδίου ταξινόμησης μπορεί να βελτιωθεί μέσα από την εφαρμογή διαφόρων μεθόδων και τεχνικών, οι οποίες αποσκοπούν στο συνδυασμό τους και θα αναλυθούν στο επόμενο κεφάλαιο. Παρακάτω ακολουθεί ο ορισμός της έννοιας του «ταξινομητή» και ο βασικός τρόπος λειτουργίας του.

Έστω ότι ένα πρότυπο z περιγράφεται από ένα διάνυσμα n χαρακτηριστικών x (feature vector) και ανήκει σε μια κλάση $l(z)$ του συνόλου των κλάσεων Ω. Τότε ένας «ταξινομητής» θα είναι μια οποιαδήποτε μαθηματική συνάρτηση μέσω της οποίας το διάνυσμα των χαρακτηριστικών ενός προτύπου $x=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$ αντιστοιχίζεται σε μια κλάση $l(z)\in\Omega$.

Πιο συγκεκριμένα για τον ορισμό ενός ταξινομητή πρέπει να οριστεί μια συνάρτηση υπόθεσης $h(\theta,\mathrm{x})$ (με παραμέτρους θ) παράλληλα μαζί με μια συνάρτηση κόστους $C(\mathrm{h}(.;\theta),\mathrm{X},\mathrm{Y})$ (όπου X,Y τα υποσύνολα των διανυσμάτων χαρακτηριστικών και των κλάσεων που ανήκουν στο σύνολο των δεδομένων). Ο αλγόριθμος του ταξινομητή θα δίνεται από το $\hat{\theta}=\arg\min_\theta C(\theta)$. Δηλαδή σκοπός του αλγόριθμου είναι η εύρεση των παραμέτρων θ που ελαχιστοποιούν τη συνάρτηση κόστους.

Κατά τη διάρκεια του σταδίου της εκπαίδευσης ένας ταξινομητής αναπτύσσει ένα σύνολο από συναρτήσεις διαχωρισμού $G=\{\mathrm{g}_1(\mathrm{x}),\mathrm{g}_2(\mathrm{x}),...\mathrm{g}_c(\mathrm{x})\}$ τόσες όσες είναι και οι κλάσεις c του προβλήματος. Κατά τη διάρκεια της λειτουργίας του ο ταξινομητής αποδίδει για κάθε πρότυπο μια πιθανότητα για κάθε κλάση, μέσα από τις συναρτήσεις διαχωρισμού και επιλέγεται η κλάση με τη μεγαλύτερη πιθανότητα (Εικόνα 2.1.1-1).



Εικόνα 2.1.1-1: Διάγραμμα λειτουργίας ενός ταξινομητή.

2.1.2. Logistic Regression

Το Logistic regression (Λογιστική παλινδρόμηση) είναι ένα μοντέλο το οποίο αναλύει τη σχέση πολλαπλών ανεξάρτητων μεταβλητών και μιας κατηγορικής μεταβλητής, και υπολογίζει τη πιθανότητα να συμβεί ένα ενδεχόμενο με τη προσαρμογή του συνόλου δεδομένων σε μια λογιστική καμπύλη .

Η λογιστική συνάρτηση είναι μια συνάρτηση (2.1) η οποία ορίζεται στο $(-\infty,+\infty)$ και παίρνει τιμές στο $[0,1]$. Επίσης έχει τη χαρακτηριστική μορφή του γράμματος S (Εικόνα 2.1.2-1) της αγγλικής γλώσσας και για αυτό το λόγο πολλές φορές αναφέρεται και ως σιγμοειδής συνάρτηση στη βιβλιογραφία.

$$S(\mathrm{x}) = \frac{1}{1+e^{-x}} = \frac{e^{x}}{e^{x}+1} \quad (2.1)$$

Πιο συγκεκριμένα η λογιστική παλινδρόμηση υπολογίζει τη πιθανότητα p ένα ενδεχόμενο να συμβεί προς τη πιθανότητα το ενδεχόμενο αυτό να μη συμβεί $(1-p)$. Λόγω των παραπάνω ο αντίκτυπος των ανεξάρτητων μεταβλητών (χαρακτηριστικών) περιγράφεται σε όρους των odds.

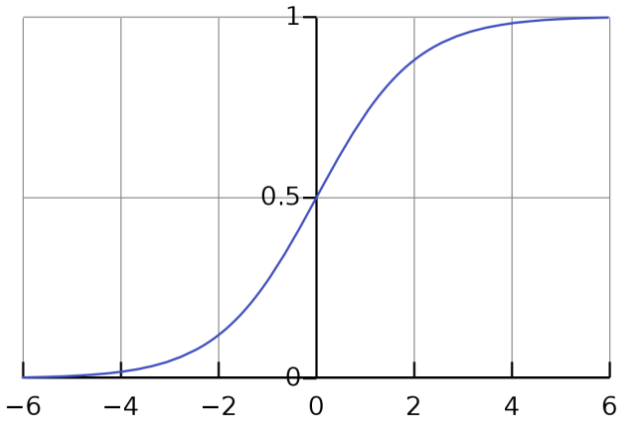
Τα odds ενός ενδεχόμενου είναι ο λόγος της πιθανότητας το ενδεχόμενο να συμβεί προς της πιθανότητα να μη συμβεί. Έστω ένα ενδεχόμενο A, τότε τα odds του ενδεχόμενου θα είναι:

$$odds\{\mathrm{A}\} = \frac{p}{1-p} \qquad (2.2)$$

p : η πιθανότητα να συμβεί το ενδεχόμενο A.

Η μέση τιμή του p στη λογιστική παλινδρόμηση αποτελεί ένα γραμμικό συνδυασμό των ανεξάρτητων μεταβλητών.

$$p = a + \beta x \qquad (2.3)$$



Εικόνα 2.1.2-1: Γραφική αναπαράσταση της λογιστικής συνάρτησης.

Η παραπάνω μοντελοποίηση του p όμως δημιουργεί προβλήματα, καθώς όταν το x πάρει ακραίες τιμές η εξίσωση (2.3) θα δώσει πιθανότητες οι οποίες δεν ανήκουν στο $[0,1]$, κάτι το οποίο δεν είναι δυνατό. Τη λύση στο παραπάνω πρόβλημα έρχεται να δώσει πάλι η λογιστική παλινδρόμηση αντικαθιστώντας τα odds με το φυσικό λογάριθμό τους (2.4).

$$\text{logit}(y) \;=\; \ln(odds = \frac{p}{1-p}) = \alpha + \beta x \qquad (2.4)$$

Τα παραπάνω ισχύουν όταν το πρόβλημα της ταξινόμησης αποτελείται μόνο από μια ανεξάρτητη μεταβλητή. Η εξίσωση 1.4 για k ανεξάρτητες μεταβλητές μετασχηματίζεται σε:

$$\text{logit}(y) \;=\; \ln(odds = \frac{p}{1-p}) = \alpha + \beta_1 x_i + ... + \beta_k x_k \qquad (2.5)$$

Παίρνοντας την αντίστροφη συνάρτηση του φυσικού λογαρίθμου και στα δύο μέλη της εξίσωσης (2.5), μπορούμε εύκολα να υπολογίσουμε τη πιθανότητα p. Η εξίσωση (2.6) αποτελεί τη συνάρτηση υποθέσεως της λογιστικής παλινδρόμησης $h(\theta, \text{x})$, όπου. $\theta = [\alpha, \beta_1, \beta_2, ..., \beta_n]$ [18].

$$p = P(Y = \text{έκβαση} / \text{X}, \alpha \text{ σταθ}) = \frac{e^{\alpha+\beta_1 x_1 + ... + \beta_k x_k}}{1 + e^{\alpha+\beta_1 x_1 + ... + \beta_k x_k}} = \\ = \frac{1}{1 + e^{-(\alpha+\beta_1 x_1 + ... + \beta_k x_k)}} \qquad (2.6)$$

Οι παράμετροι $\beta_1, \beta_2, ..., \beta_n$ περιγράφουν την αύξηση του λογάριθμου των odds, όταν οι αντίστοιχες ανεξάρτητες μεταβλητές αυξηθούν κατά μια μονάδα. Το παραπάνω μπορεί να εξηγηθεί καλύτερα σε όρους odds ratio. Το odds ratio στην ουσία είναι τα odds ενός ενδεχόμενου Α δεδομένου ότι έχει προηγηθεί ένα ενδεχόμενο Β.

$$odds\ ratio \;=\; \frac{odds\{A\}}{odds\{B\}} \qquad (2.7)$$

Επομένως το $e^{\beta_i}, i = 1, 2, ..., \text{k}$ είναι το odds ratio το οποίο αντιστοιχεί σε αύξηση μιας μονάδας της αντίστοιχης ανεξάρτητης μεταβλητής x_i.

Ακολουθούν μερικές από τις υποθέσεις που κάνει λογιστική παλινδρόμηση μέσα από τις οποίες κανείς μπορεί να συμπεράνει και τα μειονεκτήματα του αλγόριθμου:

1. Η εξαρτημένη μεταβλητή πρέπει να είναι διακριτή και σχεδόν διχοτομημένη (δυαδική).
2. Επειδή η λογιστική παλινδρόμηση υπολογίζει τη πιθανότητα ένα ενδεχόμενο να συμβεί $P(Y = 1)$, η εξαρτημένη μεταβλητή θα πρέπει να είναι κατάλληλα κωδικοποιημένη (επιθυμητή έκβαση = 1).
3. Η εκπαίδευση του μοντέλου θα πρέπει να γίνεται σωστά. Δε πρέπει το μοντέλο να γίνεται overfit παρουσία ασήμαντων χαρακτηριστικών και δε πρέπει να γίνεται underfit όταν σημαντικά χαρακτηριστικά απουσιάζουν.
4. Κάθε πρότυπο πρέπει να είναι ανεξάρτητο από τα άλλα και τα χαρακτηριστικά θα πρέπει να είναι όσο είναι δυνατό γραμμικώς ανεξάρτητα. Δηλαδή δε πρέπει να αποτελούν γραμμικό συνδυασμό άλλων χαρακτηριστικών.
5. Δε χρειάζεται να υπάρχει γραμμική σχέση μεταξύ των ανεξάρτητων μεταβλητών και της εξαρτημένη μεταβλητής. Επιθυμητή είναι όμως η γραμμική σχέση

μεταξύ των ανεξάρτητων μεταβλητών και του φυσικού λογάριθμού των odds ενός ενδεχόμενου.

6. Χρειάζεται μεγάλα δείγματα για τον υπολογισμό των παραμέτρων $\theta = [\alpha, \beta_1, \beta_2, ..., \beta_k]$.

Η εκτίμηση των παραμέτρων $\theta = [\alpha, \beta_1, \beta_2, ..., \beta_k]$ γίνεται με τη βοήθεια της μεθόδου της μέγιστης πιθανοφάνειας. Η μέθοδος της μέγιστης πιθανοφάνειας χρησιμοποιείται για να υπολογιστεί η πιθανότητα να παρατηρηθούν τα δεδομένα, όταν οι παράμετροι $\theta = [\alpha, \beta_1, \beta_2, ..., \beta_k]$ είναι άγνωστοι.

Έστω ότι για κάθε πρότυπο από το δείγμα που χρησιμοποιείται για την εκπαίδευση του ταξινομητή, η πιθανότητα να συμβεί ένα ενδεχόμενο είναι η ίδια. Τότε το $Y_i = 1$ θα υποδεικνύει ότι για το i-στο πρότυπο το ενδεχόμενο αυτό συμβαίνει και $Y_i = 0$ το αντίθετο.

Η δεσμευμένη πιθανότητα των δεδομένων (προτύπων) δίνεται από:

$$L = \prod_{i=1}^{n} p(y/x)^{Y_i} (1 - p(y/x))^{1-Y_i} = p(y/x)^{\sum_{i=1}^{n} Y_i} (1 - p(y/x))^{n - \sum_{i=1}^{n} Y_i} \qquad (2.8)$$

όπου: $p(y/x) = h(\theta = [\alpha, \beta_1, \beta_2, ..., \beta_k], x) = \dfrac{e^{\alpha + \beta_1 x_1 + ... + \beta_k x_k}}{1 + e^{\alpha + \beta_1 x_1 + ... + \beta_k x_k}}$

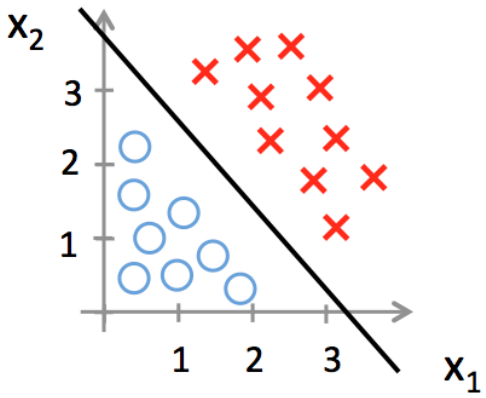
Για πρακτικούς λόγους συνήθως χρησιμοποιείται ο φυσικός λογάριθμός της συνάρτησης της πιθανοφάνειας.

$$l = \log(L) = \sum_{i=1}^{n} Y_i \log[p(y/x)] + \left(n - \sum_{i=1}^{n} Y_i \right) \log[1 - p(y/x)] \qquad (2.9)$$

Η μεγιστοποίηση της συνάρτησης της πιθανοφάνειας γίνεται με τη χρήση επαναληπτικών μεθόδων, όπου συνήθως χρησιμοποιείται ο αλγόριθμος σύγκλισης με ελάττωση της παραγώγου (gradient descent).

Τέλος μετά την εύρεση των παραμέτρων, μπορούν εύκολα να σχεδιαστούν τα όρια αποφάσεως που προκύπτουν από το μοντέλο, με τη χρήση της:

$$Y = 1 \text{ Αν } 0 \le \theta^{\mathrm{T}} x \xrightarrow{\theta = [\alpha, \beta_1, \beta_2, ..., \beta_k]} 0 \le \alpha + \beta_1 x_i + ... + \beta_k x_k \qquad (2.10)$$



Εικόνα 2.1.2-2: Αναπαράσταση ορίου αποφάσεως ταξινομητή. Η διαγώνιος υποδεικνύει το όριο αποφάσεως και τα «ο», «χ» τις κλάσεις των προτύπων.

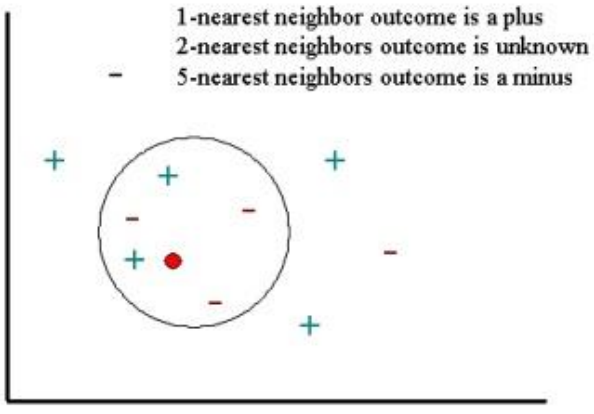
2.1.3. k-Nearest Neighbors

Ο αλγόριθμος k-Nearest Neighbors (k-Κοντινότερους Γείτονες, KNN) είναι ένας από τους πιο διαδεδομένους αλγόριθμους ταξινόμησης καθώς η λειτουργία του είναι πολύ απλή αλλά σε συγκεκριμένες περιπτώσεις παράγει πολύ καλά αποτελέσματα[19]. Η βασική λογική του αλγόριθμου KNN είναι ότι προσπαθεί να ταξινομήσει ένα πρότυπο, για το οποίο δεν γνωρίζει τη κλάση του, με βάση το σύνολο δεδομένων που του έχει δοθεί. Δηλαδή υπολογίζει την απόσταση από τα k κοντινότερα πρότυπα σε αυτό και δίνει στο άγνωστο πρότυπο τη κλάση που κυριαρχεί στους k γείτονες που τον περιβάλλουν.

Η διαδικασία μπορεί να περιγραφεί από τα παρακάτω βήματα:

1. Υπολογίζεται ο πίνακας των αποστάσεων κάθε πρότυπου από κάθε άλλο, το οποίο ανήκει στο σύνολο εκπαίδευσης. Ο πίνακας αυτός έχει διαστάσεις $m \times m$. Όπου m ο αριθμός των προτύπων που ανήκουν στο σύνολο εκπαίδευσης.
2. Έπειτα βρίσκονται οι k κοντινότεροι γείτονες του προτύπου με την άγνωστη κλάση. Ο υπολογισμός της απόστασης γίνεται με βάση τις τιμές των χαρακτηριστικών του προβλήματος ταξινόμησης $x = [x_1, x_2, x_3, ..., x_n]^T \in \Re^n$ και ένα μέτρο για τη μέτρηση της απόστασης.
3. Στη συνέχεια υπολογίζεται ο αριθμός των γειτόνων που ανήκουν σε κάθε κλάση ω_i .
4. Τέλος το άγνωστο πρότυπο ταξινομείται στη κλάση στην οποία ανήκουν οι περισσότεροι από του k γείτονες που το περιβάλλουν.

Στη περίπτωση που ο αριθμός των κλάσεων ισούται με 2 τότε ο αριθμός k που επιλέγεται θα πρέπει να είναι περιττός ώστε να αποφευχθεί κάποια πιθανή ισοπαλία μεταξύ των κλάσεων (Εικόνα 2.1.3-1). Γενικά ο αριθμός k δε πρέπει να είναι πολλαπλάσιο του αριθμού των κλάσεων M .



Εικόνα 2.1.3-1: Γραφική αναπαράσταση προβλήματος ισοπαλίας σε σχέση με τον αριθμό k.

Επίσης δε πρέπει να επιλέγεται πολύ μεγάλος αριθμός k , καθώς αν υπάρχουν κλάσεις με μεγάλο αριθμό ήδη ταξινομημένων προτύπων, θα υπερισχύσουν των κλάσεων που έχουν λίγα ταξινομημένα πρότυπα και τα αποτελέσματα θα είναι μεροληπτικά. Επι-

πρόσθετα ο αριθμός k δεν πρέπει να είναι ούτε πολύ μικρός, καθώς τότε το μοντέλο που θα προκύψει δε θα εκμεταλλεύεται το προτέρημα του αλγόριθμου να χρησιμοποιεί πολλά δεδομένα στο σύνολο εκπαίδευσης.

Άξια σημασίας είναι η επιλογή του μέτρου που θα χρησιμοποιηθεί για τον υπολογισμού του πίνακα αποστάσεων. Το μέτρο το οποίο είναι πιο επιθυμητό είναι αυτό στο οποίο, οι μικρές αποστάσεις μεταξύ των προτύπων υποδηλώνουν μεγαλύτερη πιθανότητα τα πρότυπα αυτά να ανήκουν στην ίδια κλάση. Μερικά από τα μέτρα που χρησιμοποιούνται για τον υπολογισμό του πίνακα αποστάσεων είναι η ευκλείδεια απόσταση, η απόσταση Manhattan καθώς και η απόσταση Minkowski.

$$\text{euclidean} = d(x,y) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + ... + (y_n - x_n)^2} = \\ = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2} \tag{2.11}$$

$$\begin{cases} \text{manhattan} = d_1(p,q) = \| p - q \|_1 = \sum_{i=1}^{n} |p_i - q_i| \\ Όπου\ p = (p_1, p_2, ..., p_n) \text{ και } q = (q_1, q_2, ..., q_n) \end{cases} \tag{2.12}$$

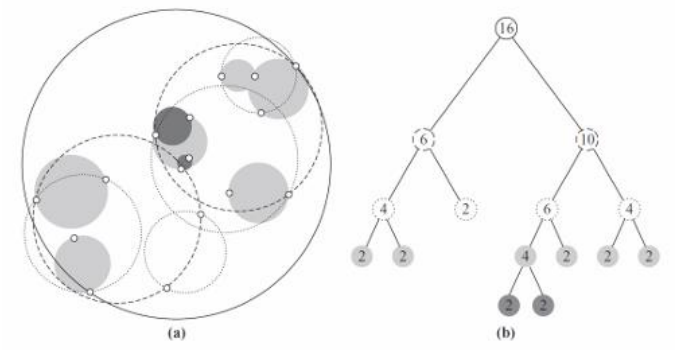
$$\text{minkowski} = D(X,Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{1/p} \tag{2.13}$$

Για $p = 1$ η απόσταση Minkowski ισούται με την απόσταση Manhattan και για $p = 2$ ισούται με την ευκλείδεια απόσταση.

Η εύρεση των k κοντινότερων γειτόνων πραγματοποιείται είτε σειριακά, είτε με τη χρήση ευρετικών αλγορίθμων. Η κατηγορία των ευρετικών αλγορίθμων που χρησιμοποιείται συνήθως είναι τα δυαδικά δέντρα αναζήτησης. Τα πιο διαδεδομένα δυαδικά δέντρα αναζήτησης που χρησιμοποιούνται είναι τα Ball trees και τα kd trees.

2.1.3.1. Balltrees

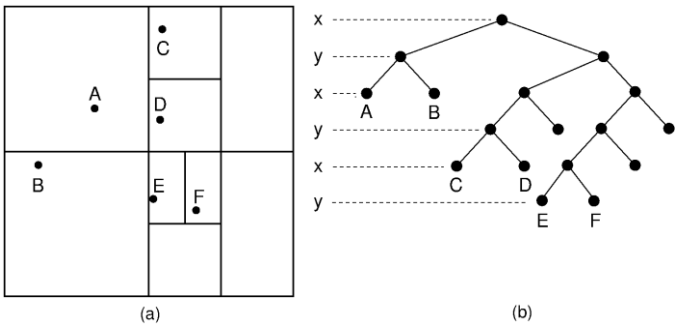
Έστω μια οριοθετημένη περιοχή από μια υπέρ-σφαίρα. Στο n-διάστατο ευκλείδειο χώρο $\Re^n$ αναφερόμαστε σε αυτή ως μια μπάλα[20]. Η μπάλα αναπαριστάτε από τις $n+1$ τιμές, οι οποίες καθορίζουν τις συντεταγμένες της και το μήκος της ακτίνα της. Ένα balltree είναι ένα πλήρες δυαδικό δέντρο, όπου κάθε κόμβος αντιστοιχίζεται με μια μπάλα έτσι ώστε κάθε εσωτερικός του κόμβος αποτελεί και αυτός μια μπάλα μικρότερη από τη δική του. Το σύνολο των εσωτερικών σε αυτό το κόμβο μπάλων αποτελούν τα παιδία του κόμβου αυτού. Τα φύλλα του δέντρου περιέχουν όλη τη πληροφορία, ενώ οι εσωτερικοί κόμβοι χρησιμοποιούνται μόνο για τη καθοδήγηση της αναζήτησης στη δομή των φύλλων. Δύο μπάλες οι οποίες βρίσκονται υπό τον ίδιο κόμβο μπορούν να τέμνονται και δε χρειάζεται να καταλαμβάνουν ολόκληρο το χώρο του γονέα τους.



Εικόνα 2.1.3.1-1:(a) Ένα σύνολο μπάλων στο χώρο. (b) Ένα δυαδικό δέντρο με βάση αυτές τις μπάλες.

2.1.3.2. Kd-trees

Ένα kd-tree είναι και αυτό ένα δυαδικό δέντρο το οποίο χωρίζει ένα σύνολο διανυσμάτων πάνω σε ένα k-διαστάτο υπέρ-επίπεδο, κάθετο σε ένα άξονα, το οποίο περνάει μέσα από το διάνυσμα που είναι αποθηκευμένο μέσα στο κόμβο[21]. Η διάσταση η οποία περιγράφεται από το παραπάνω αναφερόμενο άξονα, ονομάζεται διάσταση διαχωρισμού. Κάθε κόμβος αποτελείται από το σημείο δεδομένων το οποίο είναι κατειλημμένο από αυτόν και τον άξονα πάνω στον οποίο το υπέρ-επίπεδο χωρίζει τα παιδιά του δέντρου. Υπάρχουν 3 εναπομείναντα σημεία αναφοράς σε άλλους κόμβους του δέντρου, ένας από τους οποίους είναι ο γονέας του δέντρου. Τα άλλα δύο σημεία αναφοράς είναι οι περιοχές που περιέχουν όλα τα σημεία στα «αριστερά» του διαχωριστικού υπέρ-επίπεδου και οι περιοχές που περιέχουν όλα τα σημεία στα «δεξιά». Τέλος υπάρχει ένα ακόμα σημείο αναφοράς, το οποίο είναι το υπέρ-παραλληλεπίπεδο που περιέχει όλα τα σημεία της περιοχής.



Εικόνα 2.1.3.2-1:(a) Ένα σύνολο υπέρ-επίπεδων και κόμβων στο χώρο (b) Ένα δέντρο με βάση του κόμβους αυτών των υπέρ-επίπεδων.

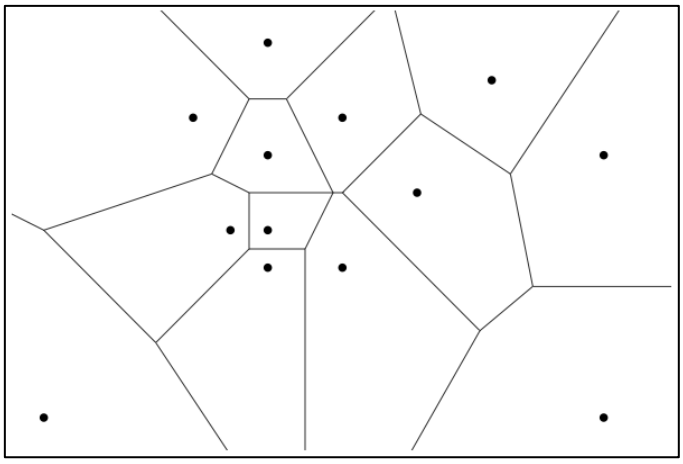
Μια ειδική περίπτωση (η απλούστερη περίπτωση) του αλγόριθμου KNN, είναι η περίπτωση όπου $k = 1$. Σε αυτή τη περίπτωση κάθε πρότυπο που ανήκει στο σύνολο εκπαιδεύσεως ορίζει μια περιοχή γύρω από αυτό, η οποία μπορεί να αποτελέσει ένα κύτταρο Voronoi. Ένα κύτταρο Voronoi περιέχει όλα τα γειτονικά σημεία που είναι κοντινότερα σε ένα συγκεκριμένο πρότυπο και ορίζεται ως:

$$R_i = \{x \in R_p : d(x, x_i) \le d(x, x_m), \forall i \ne m\} \qquad (2.14)$$

Όπου R_i είναι το κύτταρο Voronoi του πρότυπου x_i και το x συμβολίζει κάθε πρότυπο το οποίο ανήκει στο κύτταρο Voronoi R_i. Τα διαγράμματα Voronoi αντικατοπτρίζουν δύο χαρακτηριστικά του συστήματος συντεταγμένων:

- Όλα τα πιθανά πρότυπα μέσα σε ένα κύτταρο Voronoi αποτελούν υποψήφιους κοντινότερους γείτονες για το συγκεκριμένο πρότυπο.
- Το κοντινότερο πρότυπο σε αυτό που ορίζει ένα κύτταρο Voronoi βρίσκεται από τη κοντινότερη ακμή (όριο) του κύτταρου Voronoi.

Τέλος ο αλγόριθμος KNN είναι ένας «τεμπέλης» αλγόριθμος καθώς στη πραγματικότητα δεν δημιουργεί ένα εκπαιδευμένο μοντέλο, αλλά χρησιμοποιεί κάθε φορά όλα τα δεδομένα εκπαιδεύσεως για να υπολογίζει τις αποστάσεις από το πρότυπο με την άγνωστη κλάση. Αποτέλεσμα του προαναφερθέντος είναι αύξηση του υπολογιστικού κόστους και χρόνου, ειδικά σε μεγάλα σύνολα δεδομένων.



Εικόνα 2.1.3-2: Διάγραμμα Voronoi, κάθε σημείο αποτελεί ένα πρότυπο και οι ακμές συμβολίζουν τα όρια των κυττάρων Voronoi.

2.1.4. Naïve Bayes

Ο αλγόριθμος Naive Bayes είναι ένας απλός αλλά πρακτικός αλγόριθμός, ο οποίος βασίζεται στο κανόνα του Bayes (2.15) για να ταξινομήσει ένα σύνολο δεδομένων με βάση τα δεδομένα που του έχουν δοθεί. Πιο συγκεκριμένα υπολογίζει την εκ των υστέρων πιθανότητα ένα πρότυπο να ανήκει σε μια δεδομένη κλάση όταν το πρότυπο αυτό περιγράφεται από ένα διάνυσμα χαρακτηριστικών $x = [x_1, x_2, x_3, ..., x_n]^T \in \Re^n$.

Έστω δύο ενδεχόμενα A, B και $P(A)$, $P(\mathrm{B})$, $P(\mathrm{B} \mid \mathrm{A})$ γνωστές τότε η πιθανοφάνεια του A δεδομένου ότι το B ισχύει θα είναι:

$$P(A \mid B) = \frac{P(B \mid A) \cdot P(A)}{P(B)} \qquad (2.15)$$

Για να μεταφερθεί ο παραπάνω κανόνας στο πλαίσιο της ταξινόμησης πρέπει να γίνουν οι παρακάτω υποθέσεις:

- Έστω ένα σύνολο δεδομένων που αποτελείται από m πρότυπα τα οποία περιγράφονται από ένα διάνυσμα n χαρακτηριστικών

$x_i = [x_1, x_2, x_3, ..., x_n]^T \in \Re^n, i = 1,2,3,..,m$ και για τα οποία γνωρίζουμε σε πια κλάση $\omega_j \in \Omega$ ανήκουν.

- Έστω ένα άγνωστο πρότυπο x για το οποίο δεν γνωρίζουμε τη κλάση στην οποία ανήκει.
- $P(\omega_j)$: Η εκ των προτέρων πιθανότητα της κλάσης ω_j .
- $p(x \mid \omega_j)$: Η συνάρτηση πιθανοφάνειας του ω_j σε σχέση με το x .
- $p(x)$: Η συνάρτηση πυκνότητας πιθανότητας του x, όπου ισχύει $p(x) = \sum_{j=1}^{k} p(x \mid \omega_j) \mathrm{P}(\omega_j)$.

Αν υποθέσουμε ότι τα χαρακτηριστικά του x παίρνουν μόνο διακριτές τιμές τότε το $p(\omega_j \mid x)$ μετατρέπεται σε πιθανότητα.

Επομένως θα έχουμε[22]:

$$P(\omega_j \mid x) = \frac{p(x \mid \omega_j) \cdot P(\omega_j)}{p(x)} \qquad (2.16)$$

Μια επιπλέον υπόθεση που κάνει ο αλγόριθμος του Naïve Bayes είναι ότι οι τιμές των χαρακτηριστικών είναι ανεξάρτητες δεδομένου μιας μεταβλητής στόχου(της κλάσης του προτύπου) εξού και το «naïve» (αφελής) στο όνομα του.

Οι παραπάνω πιθανότητες οι οποίες χρειάζονται για τον υπολογισμό της εκ των υστέρων πιθανότητας, μπορούν να υπολογιστούν εύκολα από τα δεδομένα εκπαιδεύσεως του αλγορίθμου, μετρώντας τις αντίστοιχες συχνότητες κάθε φορά.

Έστω τώρα ότι έχουμε μόνο 2 κλάσεις, δηλαδή βρισκόμαστε στο πρόβλημα του binary classification. Η ταξινόμηση του άγνωστου προτύπου θα γίνεται με το παρακάτω κανόνα:

$$\begin{cases} \text{Αν } P(\omega_1 \mid x) > P(\omega_2 \mid x), \text{ το } x \text{ ανήκει στη κλάση } \omega_1 \\ \text{Αν } P(\omega_1 \mid x) < P(\omega_2 \mid x), \text{ το } x \text{ ανήκει στη κλάση } \omega_2 \end{cases} \qquad (2.17)$$

Ο παραπάνω κανόνας γενικεύεται στο πρόβλημα του multi-class classification ως εξής:

Έστω ένα πρόβλημα ταξινόμησης με M κλάσεις $\omega_1, \omega_2, ..., \omega_M$, τότε ένα άγνωστο πρότυπο x ταξινομείται στη κλάση ω_i αν:

$$P(\omega_i \mid x) > P(\omega_j \mid x), \forall i \neq j \qquad (2.18)$$

Εκτός το βασικό αλγόριθμο του Naïve Bayes έχουν κατασκευαστεί διάφορες παραλλαγές του αλγόριθμου ανάλογα με τη κατανομή που ακολουθούν οι τιμές των χαρακτηριστικών των προτύπων. Οι πιο δημοφιλείς παραλλαγές είναι ο Multinomial Naïve Bayes, ο Gaussian Naïve Bayes και ο Bernulli Naïve Bayes, οι οποίοι και θα παρουσιαστούν παρακάτω.

2.1.4.1. Multinomial Naïve Bayes

Ο Multinomial Naïve Bayes κάνει την υπόθεση ότι οι τιμές των χαρακτηριστικών είναι συχνότητες, οι οποίες σύμφωνα με κάποια ενδεχόμενα έχουν δημιουργηθεί από μια

πολυωνυμική κατανομή και p_i είναι η πιθανότητα με την οποία το i-στο ενδεχόμενο μπορεί να συμβεί. Σε αυτή τη περίπτωση το διάνυσμα χαρακτηριστικών ενός προτύπου $x=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$ αναπαριστά ένα ιστόγραμμα, όπου το x_i συμβολίζει τη συχνότητα που το ενδεχόμενο i παρατηρήθηκε .

Σε αυτή τη περίπτωση θα ισχύει ότι:

$$\mathrm{p(x} \mid \omega_j) = \frac{\left(\sum_i x_i\right)!}{\prod_i x_i\,!} \prod_i p_{ji}^{\,x_i} \qquad (2.19)$$

Ο παραπάνω τύπος όμως παρουσιάζει ένα πρόβλημα. Αν δεδομένης μια κλάσης, μια τιμή ενός χαρακτηριστικού δεν υπάρχει στο σύνολο δεδομένων, τότε η αντίστοιχη πιθανότητα θα είναι 0, με αποτέλεσμα επειδή έχουμε γινόμενο στο τύπο όλη η εξίσωση (2.19) να μηδενιστεί. Για αυτό το λόγο εφαρμόζουμε εξομάλυνση Laplace. Δηλαδή στον υπολογισμό των πιθανοτήτων προσθέτουμε μια μονάδα στη παρατηρούμενη συχνότητα και στο παρονομαστή τον αριθμό των διαφορετικών τιμών που μπορεί να πάρει ένα χαρακτηριστικό.

2.1.4.2. Gaussian Naïve Bayes

Ο Gaussian Naïve Bayes υποθέτει ότι τα χαρακτηριστικά ενός προβλήματος ταξινόμησης, είναι συνεχή και ακολουθούν κανονική κατανομή (κατανομή Gauss). Δηλαδή αν ένα χαρακτηριστικό x_i του διανύσματος των χαρακτηριστικών $x=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$, έχει μέση τιμή μ και διασπορά σ^2 για μια δεδομένη κλάση ω_j, τότε για μια δεδομένη τιμή u του x_i θα ισχύει[23]:

$$p(x_i = u \mid \omega_j) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(u-\mu)^2}{2\sigma^2}} \qquad (2.20)$$

2.1.4.3. Bernulli Naïve Bayes

Ο Bernulli Naïve Bayes θεωρεί ότι τα χαρακτηριστικά αποτελούν διακριτές δυαδικές μεταβλητές οι οποίες ακολουθούν μια κατανομή Bernulli. Πιο συγκεκριμένα αν ένα χαρακτηριστικό x_i του διανύσματος των χαρακτηριστικών $x=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$ παίρνει μόνο τις τιμές $\{0,1\}$ τότε για μια δεδομένη κλάση ω_j θα ισχύει:

$$p(\mathrm{x} \mid \omega_j) = \prod_{i=1}^{n} p_{ji}^{\,x_i} (1-p_{ji})^{(1-x_i)} \qquad (2.21)$$

Όπου p_{ji} είναι η πιθανότητα να υπάρχει το χαρακτηριστικό x_i δεδομένης μιας κλάσης ω_j.

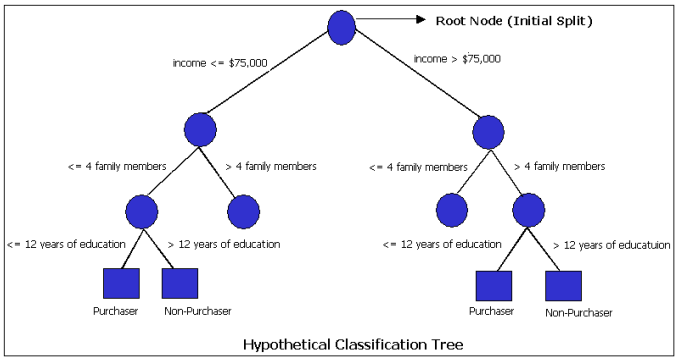
2.1.5. Decision Trees

Τα δέντρα αποφάσεων αποτελούν ένα πολυεπίπεδο σύστημα αποφάσεων, όπου οι κλάσεις απορρίπτονται σειριακά μέχρι να βρεθεί μια αποδεκτή κλάση[24][25]. Ο χώρος των χαρακτηριστικών χωρίζεται σε συγκεκριμένες περιοχές, σε σχέση με τις κλάσεις σειριακά. Η πρόβλεψη γίνεται λαμβάνοντας μια σειρά από αποφάσεις ανάλογα με τις τιμές των χαρακτηριστικών και καταλήγοντας σε μια κλάση.

Ειδικότερα ένα δέντρο αποφάσεων αποτελείται από τη ρίζα του δέντρου, ένα σύνολο από κόμβους, ένα σύνολο από κλαδιά και ένα σύνολο από φύλλα. Κάθε κόμβος έχει ένα γονέα και ένα σύνολο από παιδιά, και αναπαριστά μια απόφαση η οποία πρέπει να παρθεί. Σε κάθε τερματικό κόμβο (φύλλο του δέντρου) αντιστοιχεί μια κλάση. Ο διαχωρισμός σε κάθε κόμβο γίνεται με βάση ένα συγκεκριμένο χαρακτηριστικό. Όσο αναπτύσσετε το δέντρο, τόσο συρρικνώνονται τα υποσύνολα δεδομένων που αντιστοιχούν στους κόμβους, καθώς και η πληροφορία που περιέχεται σε αυτούς. Με άλλα λόγια τα παιδία κάθε κόμβου είναι πιο «αγνά» από τους γονείς τους. Η επιλογή του καλύτερου χαρακτηριστικού για ένα συγκεκριμένο κόμβο γίνεται με τον υπολογισμό ενός «μέτρου αγνότητας» (impurity measure).

Επιπρόσθετα τα δέντρα αποφάσεων ανάλογα με το είδος της απόφασης που παίρνεται σε κάθε κόμβο χωρίζονται σε:

- Δυαδικά (δύο επιλογές σε κάθε κόμβο)
- Μη Δυαδικά (παραπάνω από δύο επιλογές σε κάθε κόμβο)



Εικόνα 2.1.5-1: Παράδειγμα ενός υποθετικού δέντρου αποφάσεων. Οι κύκλοι αναπαριστούν τους κόμβους και τα παραλληλόγραμμα τα φύλλα του δέντρου.

Σκοπός είναι να κατασκευαστεί ένα δέντρο αποφάσεων με βάση τα δεδομένα του προβλήματος ταξινόμησης. Η διαδικασία της κατασκευής ενός δέντρου αποφάσεων στο πλαίσιο της ταξινόμησης διαμορφώνεται απαντώντας στις παρακάτω ερωτήσεις:

1. Πως θα γίνει η επιλογή των διαχωρισμών;
2. Ποιες θα είναι οι δυαδικές ερωτήσεις οι οποίες θα αντιστοιχούν στους κόμβους;
3. Πως θα γίνεται η αξιολόγηση ενός διαχωρισμού;
4. Πότε θα θεωρείται ένας κόμβος ως φύλλο ώστε να σταματήσει η διαδικασία του διαχωρισμού;
5. Πως θα οριστούν οι κλάσεις στις οποίες θα αντιστοιχούν τα φύλλα του δέντρου;

Έχουν αναπτυχθεί πολλοί αλγόριθμοι για τον τρόπο κατασκευής ενός δέντρου αποφάσεων, οι οποίοι διαφέρουν ο ένας με τον άλλο, ανάλογα με τις απαντήσεις που δίνουν στα παραπάνω ερωτήματα[25]. Οι πιο γνωστοί αλγόριθμοι κατασκευής δέντρων Trees αποφάσεων είναι ο ID3 (Iterative Dichotomiser 3), ο C4.5, ο C5.0 (μια βελτιωμένη έκδοση του C4.5) και ο CART (Classification And Regression). Παρακάτω θα παρουσιαστεί ο αλγόριθμος CART ο οποίος είναι ο πιο ολοκληρωμένος από τους τέσσερις καθώς μπορεί να χειριστεί και κατηγορικά αλλά και αριθμητικά δεδομένα. Επίσης μπορεί να χρησιμοποιηθεί και σε προβλήματα παλινδρόμησης (regression), αλλά εμείς θα εστιάσουμε στη ταξινόμηση που είναι και το αντικείμενο αυτής της εργασίας.

2.1.5.1. Ο αλγόριθμος CART

Αρχικά θα ορίσουμε κάποια σύνολα δεδομένων και χαρακτηριστικών τα οποία θα μας βοηθήσουν στη μετέπειτα επεξήγηση του αλγορίθμου[25]. Έστω ότι το A^i συμβολίζει όλες τις πιθανές τιμές του χαρακτηριστικού $a^i, i = 1, 2, ..., \mathrm{D}$ και $S, \{1, 2, ..., K\}$ το σύνολο των δεδομένων και των κλάσεων αντίστοιχα. Το S αποτελείται από n πρότυπα $s_m = (u_m, \mathrm{k_m}), \mathrm{m} = 1, 2, ...\mathrm{n}$, όπου:

- $u_m{}^i \in A^i$ είναι η τιμή του χαρακτηριστικού a^i του προτύπου s_m.
- $k_m \in \{1, 2, ..., \mathrm{K}\}$ είναι η κλάση του προτύπου s_m.

Το δέντρο απόφασης που θα προκύψει πρέπει να ταξινομήσει σε μια από τις κλάσεις $\{1, 2..., K\}$, ένα άγνωστο πρότυπο u. Η ρίζα του δέντρου (αρχικός κόμβος) συμβολίζεται ως L_0 και κάθε μη τερματικός κόμβος ως L_q. Σε κάθε κόμβο χρησιμοποιείται ένα υποσύνολο δεδομένων $S_q \in S$. Αν όλα τα πρότυπα που ανήκουν στο S_q ενός κόμβου, ανήκουν ταυτόχρονα και στην ίδια κλάση, τότε ο κόμβος θεωρείται ως φύλλο και δεν υπάρχει μετέπειτα διαχωρισμός.

Ο αλγόριθμός CART παράγει δέντρα αποφάσεων τα οποία είναι δυαδικά, δηλαδή κάθε κόμβος έχει δύο παιδιά. Με γνώμονα το παραπάνω το A^i χωρίζεται σε δύο ξένα υποσύνολα τα A^i_L και A^i_R, όπου αντιστοιχούν στο αριστερό και στο δεξιό παιδί του κόμβου αντίστοιχα (επειδή τα A^i_L, A^i_R είναι συμπληρωματικά θα αναφερόμαστε μόνο στο A^i_L από εδώ και πέρα). Το σύνολο όλων των πιθανών A^i_L του συνόλου A^i, συμβολίζεται ως V^i. Τα A^i_L, A^i_R χωρίζουν το S_q σε δύο ξένα υποσύνολα τα $L_q(A^i_L)$ και $R_q(A^i_L)$,όπου:

$$L_q\left(A^i_L\right) = \left\{s_j \in S_q \mid \mathrm{u}^{\,i}_{\,j} \in A^{\,i}_{\,L}\right\}, \qquad (2.22)$$

$$R_q\left(A^i_L\right) = \left\{s_j \in S_q \mid \mathrm{u}^{\,i}_{\,j} \in A^{\,i}_{\,R}\right\}. \qquad (2.23)$$

Τα $L_q(A^i_L)$ και $R_q(A^i_L)$ εξαρτώνται από το χαρακτηριστικό i και από τα κομμάτια που προκύπτουν από το διαχωρισμό των τιμών του. Έστω ότι τα $p_{L,q}\left(A^i_L\right)$ και $p_{R,q}\left(A^i_L\right)$ αποτελούν το κομμάτι των προτύπων του S_q τα οποία ανήκουν στα $L_q(A^i_L)$ και $R_q(A^i_L)$ αντίστοιχα. Επειδή το $p_{R,q}\left(A^i_L\right)$ είναι στην ουσία συμπληρωματικό του $p_{L,q}\left(A^i_L\right)$ θα ισχύει:

$$p_{R,q}\left(A^i_L\right)=1-p_{L,q}\left(A^i_L\right)\ . \qquad (2.24)$$

Τα πρότυπα που ανήκουν στα $L_q(A^i_L)$, $R_q(A^i_L)$ και ταυτόχρονα ανήκουν σε μια κλάση k συμβολίζονται ως $p_{kL,q}\left(A^i_L\right)$, $p_{kR,q}\left(A^i_L\right)$. Τέλος τα πρότυπα ενός μη τερματικού κόμβου L_q τα οποία απαρτίζουν το S_q και ανήκουν σε μια συγκεκριμένη κλάση k συμβολίζονται ως $p_{kq}, k=1,2,...,K$. **Να σημειωθεί ότι τα p_{kq} δεν εξαρτώνται ούτε από το χαρακτηριστικό a^i, ούτε από το A^i_L.**

Το «μέτρο αγνότητας» το οποίο χρησιμοποιεί ο αλγόριθμός CART είναι ο δείκτης Gini (Gini index), **ο οποίος δεν πρέπει να μπερδεύεται με το συντελεστή Gini**, καθώς αποτελούν διαφορετικές έννοιες. Ο δείκτης Gini, για κάθε υποσύνολο S_q του συνόλου εκπαιδεύσεως, ορίζεται ως (βάση των συμβολισμών που ορίστηκαν παραπάνω):

$$Gini\left(S_q\right)=1-\sum_{k=1}^{K}(p_{kq})^2. \qquad (2.25)$$

Ο δείκτης Gini ελαχιστοποιείται όταν κάθε πρότυπο του S_q ανήκει σε μια μοναδική κλάση και μεγιστοποιείται όταν τα πρότυπα του S_q είναι ισότιμα κατανεμημένα στο σύνολο των κλάσεων $\{1,2,...,K\}$.

Ο σταθμισμένος δείκτης Gini, ο οποίος προκύπτει από την επιλογή του A^i_L ορίζεται ως:

$$\begin{aligned} wGini\left(S_q,A^i_L\right)= & \ p_{L,q}\left(A^i_L\right)Gini\left(L_q\left(A^i_L\right)\right) \\ & +\left(1-p_{L,q}\left(A^i_L\right)\right)Gini\left(R_q\left(A^i_L\right)\right) \end{aligned}\ . \qquad (2.26)$$

Όπου:

$$L_q\left(A^i_L\right)=\left\{s_j\in S_q \mid \mathrm{u}^i{}_j\in A^i_L\right\} \qquad (2.27)$$

$$R_q\left(A^i_L\right)=\left\{s_j\in S_q \mid \mathrm{u}^i{}_j\in A^i_R\right\} \qquad (2.28)$$

Η ποιότητα του διαχωρισμού στον αλγόριθμο CART ορίζεται ως η διαφορά του δείκτη Gini και του σταθμισμένου δείκτη Gini και ονομάζεται «κέρδος Gini» .

$$g\left(S_q,A^i_L\right)=Gini\left(S_q\right)-wGini\left(S_q,A^i_L\right) \qquad (2.29)$$

Με βάση τα παραπάνω η κατασκευή ενός δέντρου αποφάσεων μέσω του αλγόριθμου CART συνοψίζεται στα παρακάτω βήματα:

1. Ο αλγόριθμος ξεκινάει από τη ρίζα του δέντρου L_0.
2. Βρίσκεται το βέλτιστό σύνολο $\tilde{A}^i_{L,\mathrm{q}}$ από όλα τα πιθανά $A^i_L\in V^i$ του συνόλου A^i, το οποίο μεγιστοποιεί το κέρδος Gini του S_q.
3. Υπολογίζεται το κέρδος Gini του S_q για το χαρακτηριστικού a^i, με βάση το $\tilde{A}^i_{L,\mathrm{q}}$ που βρέθηκε στο βήμα 2.
4. Τα βήματα 2, 3 επαναλαμβάνονται για κάθε χαρακτηριστικό $a^i, i=1,2,...,\mathrm{D}$, και επιλέγεται αυτό που αποφέρει το μεγαλύτερο κέρδος Gini στο S_q. Ταυτό-

χρονα το a^i που επιλέχτηκε αφαιρείται από τη λίστα με τα διαθέσιμα χαρακτηριστικά του κόμβου.

5. Ο κόμβος L_q χωρίζεται σε δύο κόμβους παιδιά με βάση το χαρακτηριστικό a^i που επιλέχτηκε στο βήμα 4. Τα παιδία του κόμβου L_q κληρονομούν τα διαθέσιμα χαρακτηριστικά του.
6. Τα βήματα 2, 3 , 4, 5 επαναλαμβάνονται για κάθε ένα από τα δύο παιδία του κόμβου L_q .
7. Η διαδικασία του διαχωρισμού σταματάει είτε όταν στη λίστα των διαθέσιμων χαρακτηριστικών ενός κόμβου L_q υπάρχει ένα μόνο χαρακτηριστικό, είτε όταν όλα τα πρότυπα στο S_q ανήκουν στην ίδια κλάση.

Η πρόβλεψη με ένα δυαδικό δέντρο αποφάσεων γίνεται ως εξής:

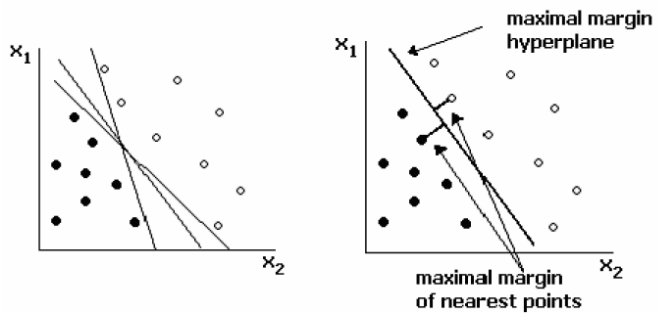
Έστω ένα άγνωστο πρότυπο $x=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$, το σύνολο των κλάσεων του προβλήματος Ω και ένα πλήρως αναπτυγμένο δέντρο αποφάσεων. Σε κάθε κόμβο συμπεριλαμβανομένου και της ρίζας του δέντρου παίρνεται μια απόφαση για το αν θα επιλεγεί το δεξί ή το αριστερό παιδί του κόμβου, ανάλογα με τιμή που έχει το πρότυπο στο χαρακτηριστικό που έχει ανατεθεί στο κόμβο. Η διαδικασία αυτή επαναλαμβάνεται μέχρι να φτάσουμε σε ένα φύλλο του δέντρου στο οποίο θα αντιστοιχεί μια συγκεκριμένη κλάση $\omega_i \in \Omega$. Τότε το δέντρο αποφάσεων ταξινομεί το πρότυπο σε αυτή τη κλάση. Αν το δέντρο δεν είναι δυαδικό τότε οι επιλογές στην απόφαση που παίρνεται σε κάθε κόμβο είναι παραπάνω από δύο.

Γενικότερα τα δέντρα αποφάσεων αποτελούν μια από τις μεθόδους ταξινόμησης, οι οποίες χρησιμοποιούνται στη βιομηχανία λόγω της ταχύτητας τους, καθώς και της ικανότητας τους να αναπαριστούν τη γνώση που απέκτησαν με απλό και κατανοητό, προς τους χρήστες τρόπο.

2.1.6. Support Vector Machine

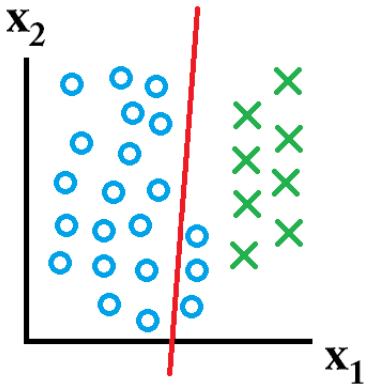
Τα Support Vector Machines (SVM) αποτελούν μια κλάση αλγορίθμων μάθησης με επίβλεψη, που αποσκοπούν στη αναζήτηση ενός υπέρ-επίπεδου, μεταξύ δύο κλάσεων έτσι ώστε να μεγιστοποιείται η απόσταση μεταξύ τους[26]. Τα SVM εκτός από προβλήματα ταξινόμησης μπορούν να επιλύσουν και προβλήματα παλινδρόμησης. Στη παρούσα εργασία θα εστιάσουμε στο κομμάτι της ταξινόμησης.

Πιο συγκεκριμένα αν ένα πρότυπο x_i αποτελείται από ένα διάνυσμα χαρακτηριστικών $x_i=[x_1,x_2,x_3,...,x_n]^T \in \Re^n$ και μια κλάση $\omega_j \in \{-1,1\}$, όπου το -1 συμβολίζει τη αρνητική κλάση και το 1 τη θετική, τότε στη περίπτωση των SVM αυτό το πρότυπο αποτελεί ένα σημείο (x_i,ω_i) στο χώρο. Το σύνολο των προτύπων του συνόλου εκπαιδεύσεως ενός SVM αποτελούν και αυτά ένα σύνολο από τέτοια σημεία $\{(x_1,\omega_1),(x_2,\omega_2),...,(x_N,\omega_N)\}$. Στη περίπτωση που τα δεδομένα είναι γραμμικά διαχωρίσιμα είναι προφανής η μη μοναδικότητα ενός υπέρ-επίπεδου, το οποίο διαχωρίζει τα δεδομένα.



Εικόνα 2.1.6-1: Αναπαράσταση της πολλαπλότητας της λύσης στο πρόβλημα της έυρεσης ενός υποερπίπεδού διαχωρισμού των δεδομένων, στη περίπτωση των γραμμικά διαχωρίσιμων δεδομένων.

Επομένως δημιουργείται το ερώτημα για το πιο είναι το βέλτιστο υπέρ-επίπεδο, το οποίο διαχωρίζει τα δεδομένα. Τα SVM υποθέτουν ότι αν ένα άγνωστο πρότυπο βρίσκεται κοντά στη περιοχή μιας κλάσης, είναι πολύ πιθανό αυτό να ανήκει σε αυτή τη κλάση. Αν όμως χρησιμοποιηθεί ένα οποιοδήποτε υπέρ-επίπεδο για το διαχωρισμό των κλάσεων, υπάρχει η περίπτωση το μοντέλο που θα προκύψει να μη ταξινομήσει σωστά νέα άγνωστα πρότυπα, τα οποία βρίσκονται πολύ κοντά στο υπερπίπεδο.



Εικόνα 2.1.6-2: Παράδειγμα όπου το επιλεγμένο υπερεπίπεδο αποτυγχάνει να διαχωρίσει άγνωστα πρότυπα τα οποία βρίσκονται πολύ κοντά σε αυτό.

Το πρόβλημα αυτό έλυσαν οι Vapnik και Chervonenkis αποδεικνύοντας ότι το βέλτιστο υπέρ-επίπεδο είναι αυτό το οποίο μεγιστοποιεί το περιθώριο μεταξύ αυτού και των δεδομένων και μάλιστα είναι και μοναδικό. Ταυτόχρονα αυτό το υπέρ-επίπεδο καλύπτει και τη περίπτωση εισαγωγής νέων δεδομένων, τα οποία είναι πολύ κοντά στο υπέρ-επίπεδο, δίνοντας τους το «περιθώριο» που χρειάζονται ώστε να ταξινομηθούν ορθά. Τα σημεία τα οποία βρίσκονται πάνω στα όρια του υπέρ-επίπεδου ορίζουν δύο παράλληλα ως προς το βέλτιστο υπέρ-επίπεδα, τα οποία ονομάζονται support vectors (διανύσματα υποστήριξης).

Το υπέρ-επίπεδο σύμφωνα με το οποίο γίνεται ο διαχωρισμός μπορεί να γραφεί σε μορφή εξίσωσης ως[27]:

$$w \cdot x - b = 0 \qquad (2.30)$$

Όπου το w είναι ένα διάνυσμα το οποίο δείχνει τη κατεύθυνση του υπέρ-επίπεδου και το b είναι μια σταθερά μέσω της οποία μπορεί να οριστεί το παραπάνω αναφερόμενο περιθώριο. Το σημείο στο οποίο πρέπει να εστιάσουμε είναι τα παράλληλα υπέρ-επίπεδα καθώς, η απόσταση μεταξύ αυτών είναι το κύριο ζήτημα.

Τα παράλληλα ως προς το βέλτιστο υπέρ-επίπεδα μπορούν εκφραστούν ως εξής:

$$w \cdot x - b = 1, \text{ για } \omega_i = 1 \qquad (2.31)$$

$$w \cdot x - b = -1 \text{ για } \omega_i = -1 \qquad (2.32)$$

Όπως είναι γνωστό η απόσταση ενός σημείου (x_0, y_0) από μια ευθεία $Ax + By + C = 0$ δίνεται από τον τύπο:

$$\frac{|Ax_0 + By_0 + C|}{\sqrt{A^2 + B^2}} \qquad (2.33)$$

Άρα στη περίπτωση των support vectors και ενός υπέρ-επίπεδου θα ισχύει:

$$\frac{|w \cdot x + b|}{\|w\|} = \frac{1}{\|w\|} \qquad (2.34)$$

Επομένως η συνολική απόσταση μεταξύ των δυο παράλληλων υπέρ-επίπεδων θα είναι $\frac{2}{\|w\|}$.

Δηλαδή για να μεγιστοποιηθεί αυτή η απόσταση πρέπει να ελαχιστοποιηθεί η ποσότητα $\|w\|$. Επιπρόσθετα για να αποφευχθεί η περίπτωση να εμφανιστούν σημεία στη περιοχή μεταξύ των δύο παράλληλων υπέρ-επίπεδων, ορίζονται οι παρακάτω περιορισμοί σε μορφή ανισότητας:

$$w \cdot x - b \le -1, \text{ για } \omega_i = 1 \qquad (2.35)$$

$$w \cdot x - b \ge 1, \text{ για } \omega_i = -1 \qquad (2.36)$$

Οι παραπάνω σχέσεις μπορούν να γραφούν ως μια ανισότητα:

$$\omega_i(\mathrm{w} \cdot \mathrm{x} - \mathrm{b}) \ge 1, i = 1, 2, ..., N \qquad (2.37)$$

Το πρόβλημα που πρέπει να επιλυθεί λοιπόν είναι το εξής:

$$\begin{cases} \min : \|w\| \\ \upsilon\pi\acute{o} : \\ \text{"να υπάρχει ένα όριο διαχωρισμού} \\ \text{μεταξύ των παράλληλων υπερεπίπεδων"} \end{cases} \qquad (2.38)$$

Το παραπάνω μπορεί να μετασχηματιστεί σε:

$$\begin{cases} \min : \frac{1}{2}\|w\|^2 \\ \upsilon\pi\acute{o} : \\ \omega_i(\mathrm{w} \cdot \mathrm{x} - \mathrm{b}) - 1 = 0 \end{cases} \qquad (2.39)$$

Το παραπάνω όμως αποτελεί ένα πρόβλημα «τετραγωνικού προγραμματισμού», άρα μπορεί να επιλυθεί με τη μέθοδο των πολλαπλασιαστών Lagrange, όπου $f(x) : \frac{1}{2}\|w\|^2$ και $g(x) : \omega_i(\mathrm{w} \cdot \mathrm{x} - \mathrm{b}) - 1$.

Είναι γνωστό ότι οι εξίσωση Lagrange δίνεται από το τύπο:

$$\mathcal{L}(x, a) = f(x) - \sum_i a_i g_i(x) \qquad (2.40)$$

Όπου το a_i συμβολίζει τους πολλαπλασιαστές Lagrange.

Στη περίπτωση του προβλήματος του SVM η λαγκραζιανή αναπαράσταση θα είναι[28]:

$$\begin{cases} \min \mathcal{L}_{\text{P}} = \frac{1}{2}\|w\|^2 - \sum_{i=1}^{N} a_i \omega_i (x_i \cdot w + b) + \sum_{i=1}^{N} a_i \\ \upsilon\pi o: \\ a_i \geq 0, \forall i \end{cases} \quad (2.41)$$

Το δυικό πρόβλημα θα είναι:

$$\begin{cases} \max \mathcal{L}_D(a_i) = \sum_{i=1}^{N} a_i - \frac{1}{2}\sum_{i=1}^{N} a_i a_j \omega_i \omega_j \left(x_i \cdot x_j\right) \\ \upsilon\pi o: \\ \sum_{i=1}^{N} a_i \omega_i = 0 \\ a_i \geq 0 \end{cases} \quad (2.42)$$

Από την ιδιότητα του πρωτεύοντος, ότι οι παράγωγοι θα είναι ίσοι με μηδέν στη βέλτιστη λύση παίρνουμε:

$$\begin{cases} w = \sum_{i=1}^{N} a_i \omega_i x_i \\ \sum_{i=1}^{N} a_i \omega_i = 0 \end{cases} \quad (2.43)$$

Με την αντικατάσταση των σχέσεων της (2.43) στο αρχικό πρόβλημα μπορούμε να αφαιρέσουμε τα w,b από τη σχέση, με αποτέλεσμα να περιέχονται σε αυτή μόνο τα διανύσματα χαρακτηριστικών εκπαίδευσης και οι κλάσεις που αντιστοιχούν.

Στη συνέχεια παίρνοντας τις μερικές παραγώγους του δυικού ως προς τα a_i και θέτοντας τις ίσες με μηδέν, μπορούν να υπολογιστούν οι πολλαπλασιαστές Lagrange a_i. Από το θεώρημα Kuhn-Tucker είναι γνωστό ότι η βέλτιστη λύση του πρωτεύοντος θα ισούται με τη βέλτιστη λύση του δυικού. Έπειτα εφόσον υπολογιστούν τα a_i, μπορεί επίσης να υπολογιστεί το w.

Ο λόγος ο οποίος επιλέχθηκε το $\frac{1}{2}\|w\|^2$ αντί του $\|w\|$, είναι για να διευκολύνουμε τη διαδικασία του υπολογισμού των απαραίτητων παραγώγων, καθώς μετά τη παραγώγηση του $\frac{1}{2}\|w\|^2$ θα παίρναμε τη ποσότητα $\|w\|$. Παραπάνω παρουσιάστηκε η διαδικασία εκπαίδευσης ενός SVM στο πρόβλημα της ταξινόμησης.

Η πρόβλεψη με ένα SVM γίνεται υπολογίζοντας το πρόσημο της:

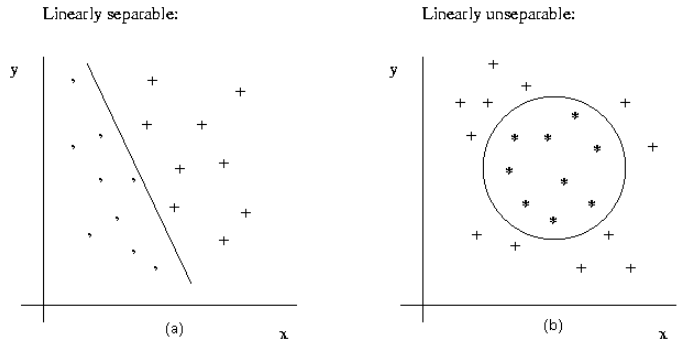
$$f(x) = w \bullet u + b = \left(\sum_{i=1}^{N} a_i \omega_i x_i \bullet u\right) + b \quad (2.44)$$

Αν αυτό είναι θετικό τότε το άγνωστο πρότυπο u ταξινομείται στη θετική κλάση +1 και αν είναι αρνητικό τότε ταξινομείται στη αρνητική κλάση -1.

Η συνάρτηση υποθέσεως του SVM δηλαδή θα είναι:

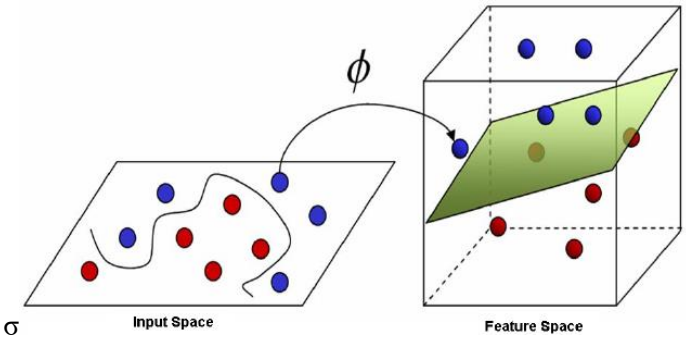
$$h(\theta = [\mathrm{b}, \mathrm{a}_1, \mathrm{a}_2, ..., \mathrm{a}_\mathrm{N}], \mathrm{x}) = sign(w \bullet x + b) \qquad (2.45)$$

Ότι αναφέρθηκε μέχρι τώρα όσον αφορά το SVM ισχύει **μόνο για γραμμικά διαχωρίσιμα δεδομένα .**



Εικόνα 2.1.6-1: (a) Γραμμικά Διαχωρίσιμα Δεδομένα, (b) Μη Διαχωρίσιμα Γραμμικά Δεδομένα.

Στη περίπτωση που τα δεδομένα δεν είναι γραμμικά διαχωρίσιμα, τότε και πάλι λύνεται το πρόβλημα μέσω ενός trick. Πιο συγκεκριμένα τα πρότυπα του συνόλου εκπαιδεύσεως x προβάλλονται σε μια επιπλέον διάσταση μέσω μιας συνάρτησης $\varphi(x)$ και πάνω σε αυτό το νέο χώρο χαρακτηριστικών που δημιουργείται εκτελείται η παραπάνω διαδικασία. Δηλαδή σε αυτή τη περίπτωση υπολογίζεται το $\varphi(x_i) \cdot \varphi(x_j)$ αντί του $x_i \cdot x_j$ κατά την επίλυση του προβλήματος και έπειτα τα υπέρ-επίπεδα αντιστοιχούνται με τον ίδιο τρόπο στον αρχικό χώρο.



Εικόνα 2.1.6-2: Αντιστοίχιση των προτύπων από το αρχικό χώρο, στο πολυδιάστατο χώρο των χαρακτηριστικών μέσω μιας μη γραμμικής συνάρτησης φ(x).

Όμως η διαδικασία αυτή της αντιστοίχισης των σημείων, μπορεί να αποδειχτεί υπολογιστικά ακριβή και χρονοβόρα, καθώς για κάθε πρότυπο θα πρέπει να υπολογίζεται η $\varphi(\mathrm{x})$ και ειδικά στις περιπτώσεις που ο νέος χώρος των χαρακτηριστικών είναι πολυδιάστατος.

Στη πραγματικότητα δε χρειάζεται να υπολογιστεί η $\varphi(x)$ αλλά το εσωτερικό γινόμενο $\varphi(x_i) \cdot \varphi(x_j)$. Το εσωτερικό αυτό γινόμενο ονομάζεται kernel (πυρήνας) και συμβολίζεται ως $k(x_i, x_j)$.

Άρα για τη μεταπήδηση στο μη γραμμικό χώρο των χαρακτηριστικών, πρέπει απλά να οριστεί η αναπαράσταση του εσωτερικού γινόμενου σε αυτό το χώρο, η οποία είναι συναρτήσει των x_i, x_j και όχι της $\varphi(x)$. Γενικά θα ισχύει ότι :

$$k(x_i, x_j) = \varphi(x_i) \cdot \varphi(x_j) \qquad (2.46)$$

Μερικοί από τους πιο γνωστούς πυρήνες παρουσιάζονται παρακάτω[29]:

- Ο πολυωνυμικός πυρήνας (polynomial kernel): $k(x_i, x_j) = (x_i \cdot x_j + 1)^d$, όπου d ο βαθμός του πολυωνυμικού πυρήνα.
- Ο πυρήνας συνάρτησης ακτινικών βάσεων (radial basis function kernel): $h_i(\theta_i, x)$, όπου γ είναι μια ελεύθερη παράμετρος.
- Ο σιγμοειδής πυρήνας(sigmoid kernel): $\tanh\left(ax_i \cdot x_j + r\right)$, όπου α, ρ δυο ελεύθεροι παράμετροι.

Τα SVM έγιναν δημοφιλή μέσα από αυτό το καινοτόμο kernel trick.

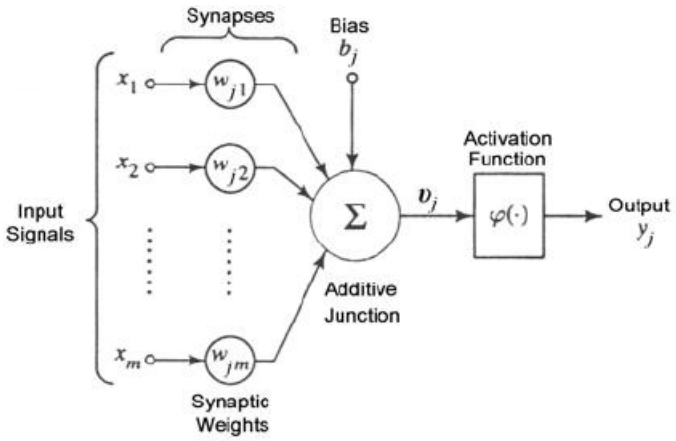
2.1.7. Artificial Neural Networks

Τα Artificial Neural Networks (Τεχνητά Νευρωνικά Δίκτυα) αποτελούν μια μεγάλη κατηγορία αλγορίθμων οι οποίοι βασίζονται στο τρόπο λειτουργίας του ανθρώπινου εγκεφάλου (**αλλά δε ταυτίζονται με αυτό**) για να ταξινομήσουν ένα σύνολο δεδομένων[30]. Σε αντίθεση με τους αλγορίθμους που έχουν παρουσιαστεί μέχρι τώρα τα τεχνητά νευρωνικά δίκτυα διαφέρουν στο γεγονός, ότι στη περίπτωση αυτών ο τρόπος με τον οποίο πραγματοποιείται η μάθηση δεν είναι ξεκάθαρος προς ένα εξωτερικό παρατηρητή. Για αυτό το λόγο πολλές φορές αναφέρονται στη βιβλιογραφία και ως «μαύρα κουτιά».

Η βασική μονάδα δόμησης ενός τεχνητού νευρωνικού δικτύου είναι ο «νευρώνας», του οποίου η λειτουργία μοιάζει πολύ με τη λειτουργία ενός νευρώνα του ανθρώπινου εγκεφάλου. Ένας νευρώνας δέχεται ως είσοδο πληροφορία και τη μετασχηματίζει μέσω μιας συνάρτησης ενεργοποίησης (activation function).

Οι πιο γνωστές συναρτήσεις ενεργοποίησης παρουσιάζονται παρακάτω:

- Η σιγμοειδής συνάρτηση (sigmoid function): $\varphi(\mathrm{x}) = \dfrac{1}{1+e^{-x}}$
- Η υπερβολική εφαπτομένη (tanh function): $\varphi(x) = \tanh(x) = \dfrac{2}{1+e^{-2x}} - 1$
- Η συνάρτηση Softmax: $\varphi(\mathrm{x}) = \sigma(\mathrm{x})_\mathrm{j} = \dfrac{e^{x_j}}{\sum_{k=1}^{K} e^{x_k}}$, όπου K ο αριθμός των νευρώνων του επίπεδου εξόδου.
- Η συνάρτηση ReLu: $\varphi(x) = \max(x, 0)$



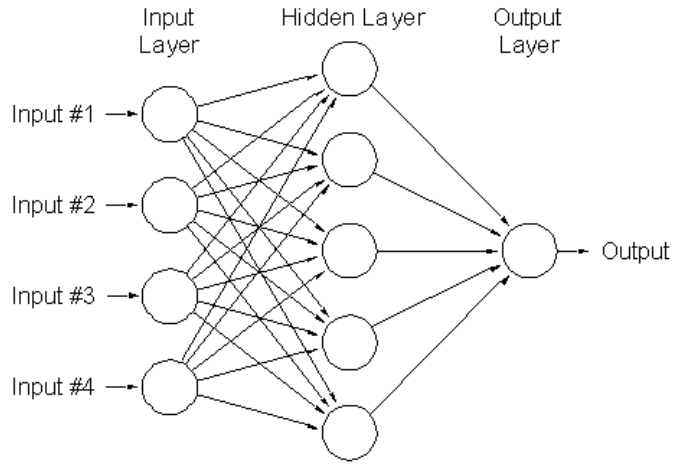
Εικόνα 2.1.7-1: Αναπαράσταση ενός νευρώνα, ενός τεχνητού νευρωνικού δικτύου.

Όπως βλέπουμε στην εικόνα 2.1.7-1 στις ακμές εισόδου ενός νευρώνα υπάρχει και η παράμετρος bias, μέσω της οποίας μπορεί δοθεί μια συγκεκριμένη κατεύθυνση στο νευρωνικό δίκτυο πριν καν αυτό εκπαιδευθεί.

Η έξοδος ενός νευρώνα υπολογίζεται με το τύπο:

$$y_j = \varphi\left(\sum_{k=1}^{m} w_{jk}\, \mathrm{x}_\mathrm{k} + b_j\right) \quad (2.47)$$

Επιπρόσθετα ένα τεχνητό νευρωνικό δίκτυο αποτελείται από ένα σύνολο νευρώνων (κόμβων), οι οποίοι συνδέονται μεταξύ του μέσω κάποιων ακμών και διατάσσονται σε επίπεδα. Το πρώτο επίπεδο ονομάζεται επίπεδο εισόδου και στην ουσία αναπαριστά ένα διάνυσμα χαρακτηριστικών ενός προτύπου αντιστοιχίζοντας ένα νευρώνα σε κάθε χαρακτηριστικό. Το τελευταίο επίπεδο ονομάζεται επίπεδο εξόδου, η έξοδος του οποίου αποτελεί και το αποτέλεσμα της πρόβλεψης. Κάθε ακμή του νευρωνικού δικτύου αντιστοιχίζεται με ένα βάρος, το σύνολο των οποίων αποτελούν τη μνήμη του τεχνητού νευρωνικού δικτύου. Στην ουσία τα βάρη αναπαριστούν τη ποιότητα τη σύνδεσης μεταξύ δύο νευρώνων. Εκτός των επίπεδων εισόδου και εξόδου υπάρχει και ένα επιπλέον επίπεδο μεταξύ των δύο το οποίο ονομάζεται κρυφό επίπεδο.

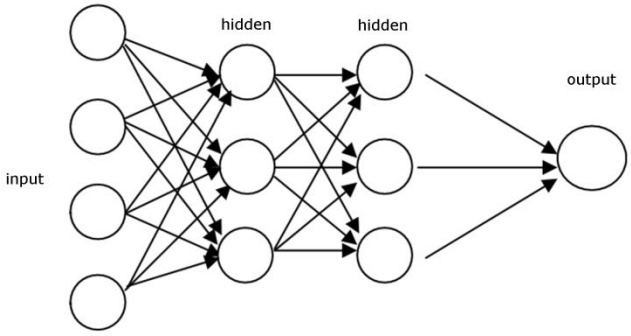


Εικόνα 2.1.7-2: Παράδειγμα αρχιτεκτονικής τεχνητού νευρωνικού δικτύου πρόσθιας τροφοδότησης.

Το bias ενός επιπέδου δεν αποτελεί νευρώνα για αυτό το λόγο δεν συνδέεται και με τους νευρώνες του επόμενου ή του προηγούμενου επίπεδου.

Αξιοσημείωτο είναι το γεγονός ότι στη περίπτωση της ταξινόμησης, η συνάρτηση Relu χρησιμοποιείται μόνο στους νευρώνες των κρυφών επιπέδων, καθώς η έξοδος της δεν επιστρέφει τιμές στο $[0,1]$. Σε αυτή τη περίπτωση στους νευρώνες εξόδου συνήθως χρησιμοποιείται η συνάρτηση softmax.

Ανάλογα με τον αριθμό των κρυφών επιπέδων, τον αριθμό των νευρώνων σε κάθε επίπεδο και τον τρόπο που συνδέονται οι νευρώνες μεταξύ τους, καθορίζεται η αρχιτεκτονική του νευρωνικού δικτύου. Στη παρούσα εργασία γίνεται χρήση τεχνητών νευρωνικών δικτύων πρόσθιας τροφοδότησης με πολλαπλά επίπεδα (feed forward neural net, multiple layer perceptron). Στα Multi-Layer Perceptron κάθε νευρώνας ενός επιπέδου i συνδέεται με κάθε νευρώνα ενός επιπέδου j με φορά από το i στο j, με $i < j$. Δηλαδή η έξοδος του επιπέδου i είναι και η είσοδος του επιπέδου j .



Εικόνα 2.1.7-3: Αναπαράσταση multi-layer perceptron με δύο κρυφά επίπεδα.

Τα νευρωνικά δίκτυα με ένα κρυφό επίπεδο δουλεύουν πολύ καλά στη περίπτωση των γραμμικά διαχωρίσιμων δεδομένων, αλλά δυσκολεύονται να ταξινομήσουν δεδομένα τα οποία είναι μη γραμμικά διαχωρίσιμα. Ισχύει ότι όσο πιο πολλά επίπεδα έχει ένα νευρωνικό δίκτυο τόσο αυξάνεται η ικανότητα του να ταξινομεί μη γραμμικά διαχωρίσιμα δεδομένα.

Η διαδικασία εκπαίδευσης ενός νευρωνικού δικτυού αποσκοπεί στην εύρεση των βαρών κάθε ακμής του νευρωνικού δικτύου, τα οποία ελαχιστοποιούν το σφάλμα μεταξύ των πραγματικών κλάσεων των προτύπων του συνόλου δεδομένων και των προβλεπόμενων κλάσεων.

Πιο συγκεκριμένα η διαδικασία εκπαίδευσης ενός Mutli-Layer Perceptron (MLP) αποτελείται από δύο βασικά βήματα, στο πρώτο βήμα γίνεται το **πρόσθιο πέρασμα** των δεδομένων με τυχαία αρχικά βάρη , γίνεται σύγκριση της πρόβλεψης με τη πραγματική κλάση, υπολογίζεται το σφάλμα και γίνεται η ανανέωση των βαρών **από το επίπεδο εξόδου προς το επίπεδο εισόδου (οπίσθιο πέρασμα)**. Τα βήματα αυτά επαναλαμβάνονται μέχρι να συγκλίνει το νευρωνικό δίκτυο ή μέχρι να πληρείται κάποιο κριτήριο τερματισμού. Η παραπάνω διαδικασία απαρτίζει τον αλγόριθμο backpropagation, ο οποίος θα παρουσιαστεί αναλυτικά παρακάτω[31]:

Έστω ο πίνακας των βαρών των νευρώνων W, όπου ο αριθμός της στήλης συμβολίζει το νευρώνα από τον οποίο ξεκινάει (πηγή) η σύνδεση και ο αριθμό γραμμής συμβολίζει τον αριθμό του νευρώνα που τερματίζει η σύνδεση (προορισμός) :

$$W = \begin{pmatrix} w_{11} & \cdots & w_{1R} \\ \vdots & \ddots & \vdots \\ w_{S1} & \cdots & w_{SR} \end{pmatrix}, \quad (2.48)$$

τότε για κάθε επίπεδο του νευρωνικού δικτύου θα υπάρχει ένα πίνακας W^k, όπου k ο αριθμός του επιπέδου στο οποίο αναφέρεται ο πίνακας.

Αν η είσοδος των νευρώνων συμβολίζεται με το διάνυσμα $p = [p_1, p_2, ..., p_R]$

και η έξοδος a τότε για κάθε νευρώνα θα ισχύει (όπως αναφέρθηκε και παραπάνω):

$$a = f(\mathrm{W}p + b), \quad (2.49)$$

όπου a η έξοδος του νευρώνα, b ο όρος του bias και f η συνάρτηση ενεργοποίησης.

Στη περίπτωση του Multi-Layer Percetron θα ισχύει ότι έξοδος του πρώτου επιπέδου a^0 θα ισούται με το διάνυσμα εισόδου p και η έξοδος του επιπέδου εξόδου a^M θα ισούται με την έξοδο του νευρωνικού δικτύου.

Επειδή η έξοδος ενός επιπέδου του νευρωνικού δικτύου γίνεται είσοδος για το επόμενο επίπεδο, θα ισχύει ότι:

$$a^{m+1} = f^{m+1}(\mathrm{W}^{m+1} a^m + b^{m+1}), m = 1, 2, ..., M \quad (2.50)$$

Επομένως στο πρόσθιο πέρασμα μέσω της σχέσης (2.50) και με τυχαία αρχικοποίηση των πινάκων των βαρών, γίνεται η πρόβλεψη των κλάσεων.

Στη συνέχεια υπολογίζεται το σφάλμα μεταξύ των προβλεπόμενων πιθανοτήτων και των πραγματικών κλάσεων μέσω μιας συνάρτησης κόστους.

Στη περίπτωση της ταξινόμησης συνήθως χρησιμοποιείται η συνάρτηση της λογιστικής απώλειας (log-loss function) :

$$\log - \mathrm{loss}(\mathrm{y}, \hat{\mathrm{y}}) = -\mathrm{y}\log\hat{\mathrm{y}} - (1 - \mathrm{y})\log(1 - \hat{\mathrm{y}}), \quad (2.51)$$

όπου y η πραγματική κλάση και $\hat{y}$ η προβλεπόμενη πιθανότητα για αυτή τη κλάση.

Η παραπάνω σχέση ισχύει για το πρόβλημα του binary classification. Στο multi-class classification για μια δεδομένη παρατήρηση (πρότυπο) o, η συνάρτηση κόστους θα ισούται με:

$$C(\mathrm{y}_o, \hat{\mathrm{y}}_o) = -\sum_{c=1}^{M} y_{o,c} \log(\hat{\mathrm{y}}_{o,c}), \quad (2.52)$$

όπου y ένα διάνυσμά που υποδεικνύει τη πραγματική κλάση (το στοιχείο του διανύσματος που είναι αντιστοιχισμένο με αυτή τη κλάση ισούται με 1 και τα υπόλοιπα με 0) και $\hat{y}$ το διάνυσμα των προβλεπόμενων πιθανοτήτων.

Το βάρη ανανεώνονται μέσω της μεθόδου gradient descent, έτσι ώστε να ελαχιστοποιηθεί η συνάρτηση του κόστους, όπου στη δική μας περίπτωση είναι η λογιστική απώλεια (log-loss).

$$w^m{}_{i,j}(k+1) = w^m{}_{i,j}(k) - a\frac{\partial \log - loss}{\partial w^m{}_{i,j}}, \quad (2.53)$$

με k να συμβολίζει τον αριθμό της επανάληψης και α το ρυθμό της μάθησης.

Η μερική παράγωγος του log-loss μπορεί να εκφραστεί μέσω του κανόνα της αλυσίδας ως:

$$\frac{\partial \log - loss}{\partial w^m{}_{i,j}} = \frac{\partial \log - loss}{\partial n_i{}^m} \times \frac{\partial n_i{}^m}{\partial w^m{}_{i,j}} \quad (2.54)$$

Ο πρώτο όρος του γινομένου ορίζεται ως «η ευαισθησία του log-loss σε αλλαγές της εισόδου του i-οστού νευρώνα στο επίπεδο m του δικτύου» και συμβολίζεται με $s_i{}^m$.

Δηλαδή θα ισχύει:

$$s_i{}^m = \frac{\partial \log - loss}{\partial n_i{}^m}, \qquad (2.55)$$

όπου $n_i{}^m$ η είσοδος του i-στου νευρώνα στο επίπεδο m του δικτύου.

Ο δεύτερος όρος είναι γνωστός καθώς $n_i{}^m = \sum_{j=1}^{S^{m-1}} w_{i,j}{}^m a_j{}^{m-1} + b_i{}^m$, επομένως θα ισχύει:

$$\frac{\partial n_i{}^m}{\partial w^m{}_{i,j}} = a_j{}^{m-1} \qquad (2.56)$$

Άρα η (2.54) μπορεί να γραφεί και ως:

$$\frac{\partial \log - loss}{\partial w^m{}_{i,j}} = s_i{}^m a_j{}^{m-1} \quad (2.57)$$

Ο gradient descent μέσω της (2.57) μπορεί να γραφεί ως:

$$w^m{}_{i,j}(k+1) = w^m{}_{i,j}(k) - a s_i{}^m a_j{}^{m-1} \quad (2.58)$$

και σε μορφή πίνακα:

$$W^m{}_{i,j}(k+1) = W^m(k) - a s^m (a^{m-1})^{\mathrm{T}} \quad (2.59)$$

Να σημειωθεί ότι τα a , a_j συμβολίζουν διαφορετικές έννοιες.

Οι ευαισθησίες $s_i{}^m$ υπολογίζονται με το παρακάτω τρόπο. Το $s_i{}^m$ μέσω του κανόνα της αλυσίδας μπορεί να γραφεί ως:

$$s_i{}^m = \frac{\partial \log - loss}{\partial n_i{}^{m+1}} \times \frac{\partial n_i{}^{m+1}}{\partial n_i{}^m} \quad (2.60)$$

Ο δεύτερος όρος στη μορφή πίνακα μπορεί να εκφραστεί από την ιακωβιανή:

$$\frac{\partial n^{m+1}}{\partial n^m} = \begin{pmatrix} \frac{\partial n_1^{m+1}}{\partial n_1^m} & \cdots & \frac{\partial n_1^{m+1}}{\partial n_{S^m}^m} \\ \vdots & \ddots & \vdots \\ \frac{\partial n_{S^{m+1}}^{m+1}}{\partial n_1^m} & \cdots & \frac{\partial n_{S^{m+1}}^{m+1}}{\partial n_{S^m}^m} \end{pmatrix}, \qquad (2.61)$$

όπου ένα στοιχείο στη θέση (i, j) του πίνακα μπορεί να γραφεί ως:

$$\frac{\partial n_i^{m+1}}{\partial n_j^m} = \frac{\partial \left(\sum_{l=1}^{S^m} w_{i,l}^{m+1} a_l^m + b_i^{m+1} \right)}{\partial n_j^m} = w_{i,j}^{m+1} \frac{\partial a_j^m}{\partial n_j^m} = w_{i,j}^{m+1} \frac{\partial f^m(n_j^m)}{\partial n_j^m} \qquad (2.62)$$

Αν θέσουμε όπου $n_j^m = x$ μπορούμε εύκολα να δούμε ότι, η μερική παράγωγος στην ουσία είναι ολική παράγωγος και η συνάρτηση ενεργοποίησης είναι γνωστή από την αρχή της εκπαίδευσης επομένως από τις (2.60), (2.61) και (2.62) θα έχουμε ότι:

$$s^m = F'(\mathrm{n}^m)(W^{m+1})^{\mathrm{T}} s^{m+1}, \quad (2.63)$$

όπου:

$$F'^m = \begin{pmatrix} f'^m(n_1^m) & & 0 \\ & \ddots & \\ 0 & & f'^m(n^m{}_{S^m}) \end{pmatrix} \qquad (2.64)$$

και:

$$\frac{\partial \log - loss}{\partial n_i^{m+1}} = s^{m+1} \ . \qquad (2.65)$$

Από τη (2.63) παρατηρούμε ότι οι ευαισθησίες μεταδίδονται προς τα πίσω από το επίπεδο m+1 στο επίπεδο m. Από αυτό το γεγονός πήρε το όνομα του και ο αλγόριθμος (οπίσθια διάδοση).

Για να ξεκινήσει ο αλγόριθμος όμως χρειάζεται να υπολογιστεί το s^M δηλαδή η ευαισθησία στο επίπεδο εξόδου του νευρωνικού δικτύου.

Άρα θα έχουμε:

$$s_i^M = \frac{\partial \log - loss}{\partial n_i^M}, \qquad (2.66)$$

όπου υπολογίζεται με αντικατάσταση του log-loss ή οποιασδήποτε συνάρτησης κόστους έχει επιλεχθεί.

ΚΕΦΑΛΑΙΟ 3

3.1 ΣΥΝΔΥΑΣΜΟΣ ΤΑΞΙΝΟΜΗΤΩΝ

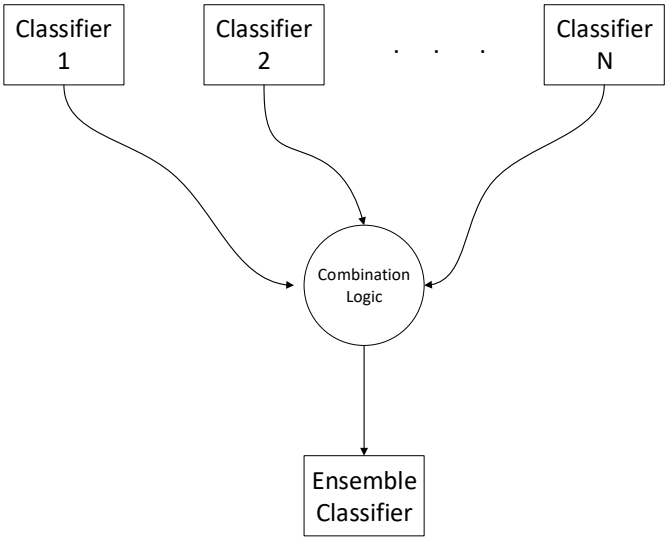
3.1.1. Εισαγωγή

Στο προηγούμενο κεφάλαιο παρουσιάσαμε εν συντομία τη θεωρία των ταξινομητών του πρώτου επιπέδου, καθώς και το τρόπο λειτουργίας τους. Σε αυτό το κεφάλαιο θα γίνει μια σύντομη εισαγωγή στη θεωρία του συνδυασμού των ταξινομητών (Ensemble Learning) και θα παρουσιαστούν οι βασικοί αλγόριθμοι συνδυασμού ταξινομητών πρώτου επιπέδου. Στη συνέχεια θα παρουσιαστούν δύο από τα προβλήματα τα οποία συναντάει κανείς κατά τη δημιουργία ensemble και θα παρουσιαστεί η προτεινόμενη μεθοδολογία.

3.1.2. Ensemble Learning

Το Ensemble Learning αποτελεί ένα τομέα του machine learning, ο οποίος ασχολείται με το συνδυασμό βασικών ταξινομητών (ή μοντέλων παλινδρόμησης) για τη δημιουργία ισχυρότερων ταξινομητών. Η φιλοσοφία πίσω από το Ensemble Learning είναι, ότι μια ομάδα από ειδικούς μπορεί να πάρει καλύτερες αποφάσεις από ότι αν οι ειδικοί αποφάσιζαν μόνοι τους[32]. Κάθε ειδικός αναπαριστά ένα μοντέλο ταξινόμησης του οποίου οι προβλέψεις συνδυάζονται με ένα συγκεκριμένο τρόπο και λαμβάνεται μια καλύτερη πρόβλεψη. Το παραπάνω σενάριο έχει μελετηθεί πειραματικά και θεωρητικά και έχει βρεθεί ότι οι συνδυασμένες προβλέψεις ενός συνόλου ταξινομητών δίνουν καλύτερα αποτελέσματα σε σχέση με τις μεμονωμένες προβλέψεις . Αυτό οφείλεται στην ύπαρξη της διαφορετικότητας μεταξύ των ταξινομητών. Όμως η διαφορετικότητα μεταξύ των ταξινομητών δε φτάνει ώστε να επιτευχτεί η επιθυμητή απόδοση από μια ensemble. Θα πρέπει και οι επιμέρους βασικοί ταξινομητές από μόνοι τους να μπορούν να πετύχουν υψηλές αποδόσεις. Έπειτα από πειραματικές μελέτες που έγιναν ανά τα χρόνια, αποδείχθηκε ότι είναι καλύτερο να χρησιμοποιούνται ισχυροί ταξινομητές, ακόμα και αν με τη χρησιμοποίηση ασθενών ταξινομητών επιτυγχάνεται μεγαλύτερη βελτίωση από την ensemble.

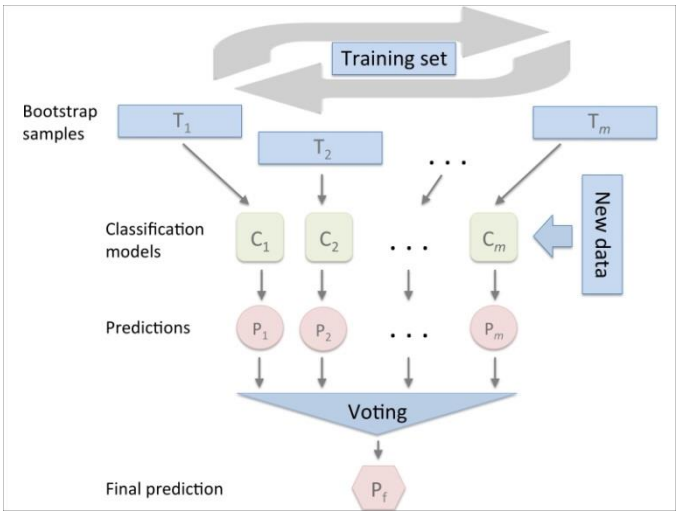
Ανάλογα με τον τρόπο εκπαίδευσης και επιλογής των ταξινομητών του πρώτου επιπέδου καθώς και τον τρόπο συνδυασμού των προβλέψεων των ταξινομητών έχουν αναπτυχθεί διάφορες μέθοδοι. Εκτός από βασικούς ταξινομητές μπορούν να χρησιμοποιηθούν ensembles στο πρώτο επίπεδο (ensembles of ensembles) ή μια μίξη από ensemble και βασικών ταξινομητών (hybrid ensembles). Παρακάτω παρουσιάζονται μερικές από τις βασικότερες ensemble μεθόδους .



Εικόνα 3.1.2-1: Αναπαράσταση ενός ensemble ταξινομητή.

3.1.2.1. Bagging

Το Bagging (Bootstrap Aggregating) αποτελεί μια μέθοδο ensemble, η οποία είναι βασισμένη στο bootstrapping, το οποίο παρουσιάστηκε στο 1ο Κεφάλαιο[33]. Πιο συγκεκριμένα, λαμβάνονται k τυχαία δείγματα με επανατοποθέτηση από το σύνολο των δεδομένων. Στη συνέχεια εκπαιδεύονται k ταξινομητές (ένας για κάθε δείγμα που πάρθηκε) με βάση τον αλγόριθμο ταξινόμησης που έχει επιλεγεί και παράγονται οι προβλέψεις για όλο το σύνολο δεδομένων. Τέλος οι προβλέψεις συνδυάζονται, με το κανόνα της πλειοψηφίας (majority voting) και κάθε πρότυπο ταξινομείται σε μια κλάση.



Εικόνα 3.1.2.1-1: Γραφική αναπαράσταση της διαδικασίας που ακολουθείται από το Bagging.

Ο κανόνας της πλειοψηφίας στη περίπτωση του Bagging ορίζεται ως εξής:

Έστω το σύνολο δεδομένων T το οποίο αποτελείται από p πρότυπα και το σύνολο των κλάσεων Ω. Αν $h(\theta, x)$ είναι η συνάρτηση υπόθεσης του επιλεγμένου αλγορίθ-

μου ταξινόμησης και $C_i(\theta_i, x), i = 1, 2..., k$ οι αντίστοιχες συναρτήσεις των ταξινομητών, που παράγονται από k τυχαία δείγματα με επανατοποθέτηση μεγέθους $s \le p$, τότε η κλάση ενός προτύπου $t \in T$ θα ισούται με :

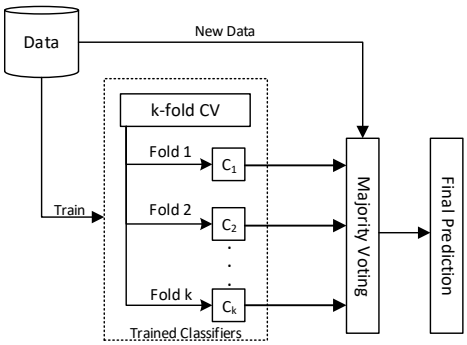
$$\hat{y} = \text{mode}\{C_1(\theta_i, t), C_2(\theta_i, t), ..., C_k(\theta_k, t)\}, \quad (3.1)$$

Αξιοσημείωτο είναι το γεγονός ότι η εκπαίδευση των ταξινομητών στη περίπτωση του Bagging γίνεται παράλληλα, δηλαδή ο χρόνος εκπαίδευσης ενός ταξινομητή δεν εξαρτάται από αυτό ενός άλλου. Επιπρόσθετα το Bagging αποτελεί μια μέθοδο ιδανική για τη βελτίωση της απόδοσης μοντέλων, τα οποία είναι ευαίσθητα σε πιθανές αλλαγές του συνόλου δεδομένων, δηλαδή μοντέλα που περιέχουν υψηλό Variance.

Γενικά υπάρχουν πολλές παραλλαγές του Bagging ανάλογα με το πώς πραγματοποιείται το στάδιο της δειγματοληψίας και το στάδιο του συνδυασμού των προβλέψεων, μια από τις οποίες είναι το Wagging[34]. Στη περίπτωση του Wagging ο κανόνας με τον οποίο συνδυάζονται οι προβλέψεις δεν είναι αυτός της πλειοψηφίας (majority voting) αλλά η σταθμισμένη πλειοψηφία. Δηλαδή ένα βάρος $w_i, i = 1, 2, ...k$ αντιστοιχίζεται με κάθε ταξινομητή και υπολογίζεται η σταθμισμένη πλειοψηφία των προβλέψεων με βάση αυτά τα βάρη.

3.1.2.2. Cross Validation Committees

Τα Cross Validation Commitees αποτελούν μια μέθοδο ensemble, που όπως και το Bagging βασίζεται στη χρησιμοποίηση υποσυνόλων του συνόλου δεδομένων για την εκπαίδευση των βασικών ταξινομητών. Αναλυτικότερα το σύνολο δεδομένων T χωρίζεται σε k ξένα υποσύνολα του T, μεγέθους $\frac{N}{k}$. Ύστερα εκπαιδεύονται παράλληλα (όπως και στο bagging) k ταξινομητές χρησιμοποιώντας κάθε φορά ως σύνολο εκπαίδευσης $k-1$ υποσύνολα από τα k υποσύνολα που δημιουργήθηκαν. Το εναπομείναν υποσύνολο χρησιμοποιείτε ως σύνολο επαλήθευσης (validation set) του ταξινομητή που δημιουργείται κάθε φορά. Μετά από αυτό το στάδιο οι ταξινομητές τροφοδοτούνται με νέα δεδομένα και παράγονται οι προβλέψεις. Τέλος οι προβλέψεις συνδυάζονται χρησιμοποιώντας το μέσο όρο αυτών ή το κανόνα της πλειοψηφίας (majority voting).



Εικόνα 3.1.2.2-1: Γραφική αναπαράστασης της διαδικασίας που ακολουθείτε από τις Cross Validation Committees

Τα Cross Validation Committees βελτιώνουν την απόδοση της ταξινόμησης μειώνοντας τη συσχέτιση των σφαλμάτων των επιμέρους ταξινομητών[35]. Θα αποτελούσε παράλειψη αν δεν αναφέραμε ότι το παραπάνω συμβαίνει, ακόμα και όταν οι αποδόσεις των ταξινομητών που είναι εκπαιδευμένοι σε κάθε fold, είναι χειρότερες από αυτές των ταξινομητών που είναι εκπαιδευμένοι σε ολόκληρο το σύνολο δεδομένων. Η απόδοση των Cross Validation Committees δεν παύει όμως να εξαρτάται από τη γενικότερη απόδοση των βασικών ταξινομητών.

3.1.2.3. Boosting

Το Boosting αναφέρεται σε μια κατηγορία μεθόδων ensemble οι οποίες δημιουργούν ιεραρχικά (σειριακά) νέους ταξινομητές, χρησιμοποιώντας τυχαία δείγματα του συνόλου δεδομένων για την εκπαίδευση των ταξινομητών, όπου σε κάθε βήμα ο ταξινομητής που εκπαιδεύεται, προσπαθεί να μάθει τα δεδομένα τα οποία απέτυχε να μάθει ο προηγούμενος[36].

Αν T το σύνολο των δεδομένων εκπαίδευσης και $h(\theta, x)$ η συνάρτηση υποθέσεως του επιλεγμένου αλγορίθμου ταξινόμησης, τότε τα βήματα με τα οποία παράγεται μια boosting ensemble συνοψίζονται ως εξής:

1. Αρχικά λαμβάνεται ένα τυχαίο δείγμα με επανατοποθέτηση από το T, το οποίο συμβολίζεται ως $S_0 \in T$.
2. Στη συνέχεια εκπαιδεύεται ένας ταξινομητής στο S_0 με βάση τον αλγόριθμο ταξινόμησης που επιλέχθηκε.
3. Έπειτα δημιουργείτε το σύνολο εκπαίδευσης του επόμενου ταξινομητή S_1 επιλέγοντας τυχαία «ένα ένα» τα πρότυπα που θα αποτελούν αυτό. Η επιλογή γίνεται με τη χρήση ενός «κέρματος», όπου ένας εξωτερικός παρατηρητής ρίχνει. Αν το κέρμα φέρει «κεφαλή» τότε το τυχαίο δείγμα ταξινομείται και αν ταξινομηθεί λανθασμένα από το προηγούμενο ταξινομητή, προστίθεται στο σύνολο δεδομένων εκπαιδεύσεως του επόμενου ταξινομητή, αλλιώς λαμβάνεται νέο δείγμα. Στη περίπτωση που έρθουν γράμματα ακολουθείτε η ίδια διαδικασία ακριβώς, όμως τώρα το πρώτο ορθά ταξινομημένο δείγμα προστίθεται επίσης στο προαναφερθέν σύνολο δεδομένων.
4. Μετά από αυτό δημιουργείται ένας νέος ταξινομητής πάνω στο σύνολο S_1 και το βήμα 3 επαναλαμβάνεται.
5. Η διαδικασία της δημιουργίας της ensemble τερματίζεται όταν δημιουργηθούν όσοι ταξινομητές επιθυμεί ο χρήστης.
6. Αφότου έχει τελειώσει η διαδικασία εκπαίδευσης των ταξινομητών παράγονται οι προβλέψεις σύμφωνα με την $h_i(\theta_i, x)$ κάθε ταξινομητή, σε ένα άγνωστο από τους ταξινομητές σύνολο δεδομένων και συνδυάζονται με το κανόνα της πλειοψηφίας.

Μια παραλλαγή του Boosting αποτελεί ο αλγόριθμος Ada-Boost. Ο αλγόριθμος Ada-Boost αντιστοιχεί ένα βάρος w_i με κάθε πρότυπο $t \in T$. Το βάρος w_i^m στην ουσία περιγράφει τη πιθανότητα να επιλεγεί το πρότυπο t_i στη δειγματοληψία της επανάληψης m του αλγορίθμου, σε αντίθεση με το bagging το οποίο θεωρεί ότι τα ενδεχόμενα της επιλογής κάποιου πρότυπου είναι ισοπίθανα. Αρχικά τα βάρη είναι ίδια και ίσα με $\frac{1}{N}$, όπου N ο αριθμός των προτύπων. Επιπρόσθετα σε κάθε βήμα το βάρος, κάθε πρότυπου του S_m ανανεώνεται με βάση τους τύπους:

$$w_i^{(m+1)} = w_i^m \sqrt{\frac{e_m}{1-e_m}},\ \text{αν ταξινομηθεί ορθά,} \quad (3.2)$$

$$w_i^{(m+1)} = w_i^m \sqrt{\frac{1-e_m}{e_m}},\ \text{αν ταξινομηθεί λανθασμένα,} \quad (3.3)$$

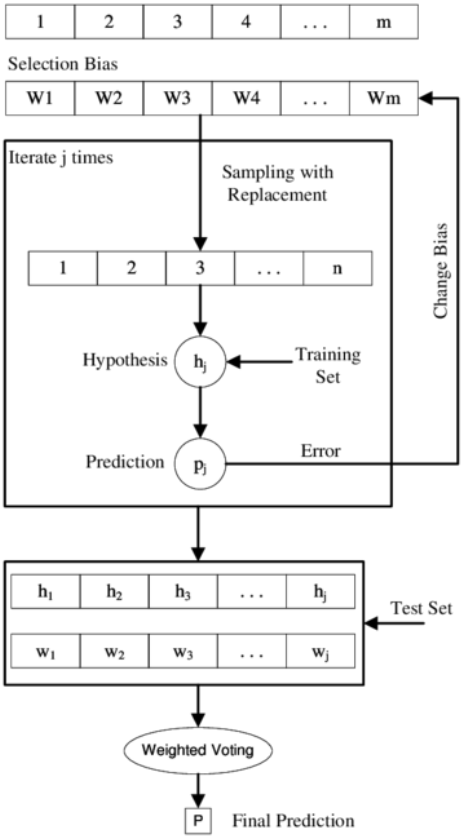
με $e_m = \frac{W_c}{W}$, $W = W_c + W_e$.

Το e_m συμβολίζει το ρυθμό του σφάλματος, με βάση τα βάρη των προτύπων και $W_c, \mathrm{W_e}$ είναι τα αθροίσματα των συνόλων των βαρών των προτύπων, που ταξινομήθηκαν σωστά και λανθασμένα στην επανάληψη m.

Όσον αφορά το συνδυασμό των ταξινομητών στο τέλος της δημιουργίας της ensemble, αυτή γίνεται με το κανόνα της σταθμισμένης πλειοψηφίας (weighted majority voting) . Τα βάρη που ορίζονται στους ταξινομητές κατά στη διαδικασία του συνδυασμού δίνονται από το τύπο:

$$a_m = \frac{1}{2}\ln\left(\frac{1-e_m}{e_m}\right) \quad (3.4)$$

Να σημειωθεί ότι οι boosting ensembles έχει αποδειχθεί ότι μειώνουν και το bias αλλά και το variance ενός ταξινομητή, δίνοντας μεγαλύτερη βαρύτητα στο bias.



Εικόνα 3.1.2.3-1: Γραφική αναπαράσταση της διαδικασίας που ακολουθείται από τον αλγόριθμο Adaboost.

3.1.2.4. Stacking

Το stacking αποτελεί μια μέθοδο με τελείως διαφορετική φιλοσοφία από αυτές που αναφέρθηκαν παραπάνω. Βασίζεται στην ιδέα ότι η διαδικασία του συνδυασμού των ταξινομητών είναι και αυτή μια εργασία την οποία κάποιος μπορεί να μάθει να εκτελεί. Επομένως εύκολα έρχεται κανείς στο συμπέρασμα, ότι μπορεί να χρησιμοποιηθεί κάποιος ταξινομητής σε δεύτερη φάση, ο οποίος θα μαθαίνει να ταξινομεί σωστά ένα σύνολο δεδομένων με βάση τις προβλέψεις των βασικών ταξινομητών σε αυτό[37].

Έστω το σύνολο δεδομένων T και το σύνολο των κλάσεων Ω. Αν T_{tr} και T_t είναι το σύνολο εκπαιδεύσεως και ελέγχου αντίστοιχα, τότε τα βήματα για τη δημιουργία μιας stacked ensemble παρουσιάζονται παρακάτω:

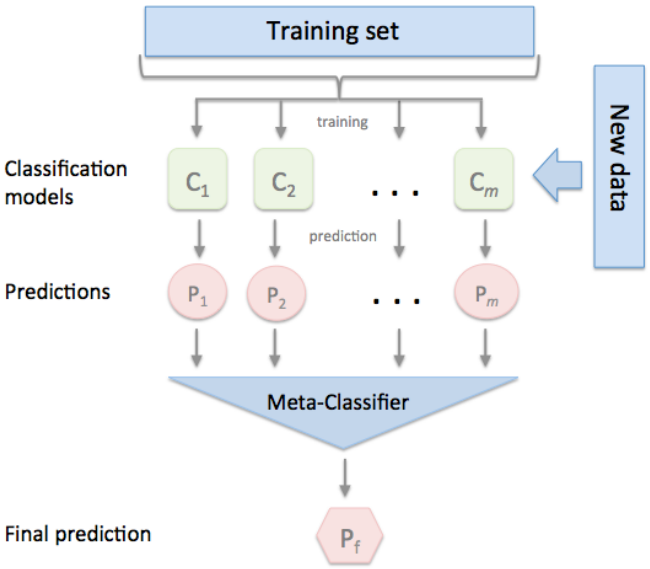
1. Αρχικά εκπαιδεύονται οι βασικοί ταξινομητές στο T_{tr} και διαμορφώνονται οι συναρτήσεις υποθέσεως τους $h_i(\theta_i, x), \mathrm{i} = 1, 2, ..., \mathrm{N}$. Το N συμβολίζει τον αριθμό των ταξινομητών.
2. Έπειτα παράγονται οι προβλέψεις των ταξινομητών στο T_{tr} και δημιουργείται ένα νέο σύνολο δεδομένων Y με βάση αυτές τις προβλέψεις. Δηλαδή κάθε χαρακτηριστικό (στήλη) του νέου συνόλου δεδομένων αποτελεί τις προβλέψεις ενός ταξινομητή στο T_{tr} , αλλά η στήλη των πραγματικών κλάσεων παραμένει ως έχει.
3. Στη συνέχεια εκπαιδεύεται ένας νέος ταξινομητής δευτέρου επιπέδου (meta classifier) πάνω στο νέο σύνολο δεδομένων Y , του οποίου στόχος είναι να ταξινομεί τα πρότυπα του T_{tr} με βάση τις προβλέψεις των ταξινομητών του πρώτου επιπέδου (βασικών ταξινομητών) σε αυτό.

Η τελική πρόβλεψη της ensemble παράγεται ως εξής:

1. Πρώτα παράγονται οι προβλέψεις των ταξινομητών του πρώτου επιπέδου με βάση τις $h_i(\theta_i, x), \mathrm{i} = 1, 2, ..., \mathrm{N}$ στο T_t.
2. Οι προβλέψεις αυτές περνάνε στο επόμενο επίπεδο ταξινόμησης.
3. Με βάση αυτές τις προβλέψεις ο ταξινομητής του δευτέρου επιπέδου παράγει τις τελικές προβλέψεις της ensemble.

Συνήθως στο δεύτερο επίπεδο ταξινόμησης χρησιμοποιείται ένας ταξινομητής βασισμένος στη λογιστική παλινδρόμηση (Logistic Regression). Σε αυτή τη περίπτωση οι παράμετροι $\beta_1, \beta_2, ..., \beta_N$ στην ουσία αποτελούν κατά κάποιο τρόπο τα βάρη, που αντιστοιχούνται στις προβλέψεις των βασικών ταξινομητών. Η τελική πρόβλεψη όμως δεν λαμβάνεται μέσω της σταθμισμένης πλειοψηφίας αλλά μέσω της λογιστικής συνάρτησης. Για να μπορεί να αποδοθεί στα $\beta_1, \beta_2, ..., \beta_N$ η έννοια του ποσοστού συμμετοχής κάθε ταξινομητή στην ensemble, θα πρέπει να εφαρμόζεται σε αυτά κατά τη διάρκεια της εκπαίδευσης του ταξινομητή δευτέρου επιπέδου, ένας περιορισμός μη αρνητικότητας.

Επιπρόσθετα έχει αποδειχθεί ότι η χρησιμοποίηση των πιθανοτήτων των κλάσεων αντί των ετικετών των κλάσεων αποδίδει καλύτερα αποτελέσματα. Το προαναφερθέν πραγματοποιείται εκπαιδεύοντας ένα μοντέλο λογιστικής παλινδρόμησης για κάθε κλάση, όπου το σύνολο εκπαίδευσης θα αποτελείται από τις πιθανότητες που προκύπτουν από τις προβλέψεις των βασικών ταξινομητών. Ως τελική πρόβλεψη θα επιλέγεται η κλάση με τη μεγαλύτερη πιθανότητα (από αυτές που παράχθηκαν από τα επιμέρους μοντέλα λογιστικής παλινδρόμησης).



Εικόνα 3.1.2.4-1: Γραφική αναπαράσταση της βασικής διαδικασίας που ακολουθείται από τις stacking ensemble.

3.1.3. Προβλήματα στη δημιουργία ensemble μοντέλων

Παραπάνω παρουσιάστηκαν οι βασικότερες μεθοδολογίες για τη δημιουργία ensemble μοντέλων. Όπως είναι φανερό κατά τη δημιουργία ensemble μοντέλων ταξινόμησης δημιουργούνται πολλαπλά προβλήματα όσον αφορά τη παραμετροποίηση αυτών. Οι βασικότεροι παράμετροι που επηρεάζουν την απόδοση των ensemble μοντέλων είναι:

- Οι αλγόριθμοι που είναι βασισμένοι οι ταξινομητές του πρώτου επιπέδου (βασικοί ταξινομητές).
- Ο αριθμός των βασικών ταξινομητών που θα περιέχονται στο τελικό ensemble μοντέλο.
- Ο βαθμός συμμετοχής κάθε βασικού ταξινομητή στην τελική απόφαση (πρόβλεψη).

Ο καθορισμός του αριθμού των βασικών ταξινομητών και των βαρών που αποδίδονται σε αυτούς, συνήθως πραγματοποιείται μέσω μιας λογικής δοκιμής-σφάλματος. Πιο συγκεκριμένα δοκιμάζονται πιθανοί συνδυασμοί των τιμών των παραμέτρων και κρατούνται τα μοντέλα που επιφέρουν την μεγαλύτερη απόδοση.

Η επιλογή των αλγορίθμων των βασικών ταξινομητών συνήθως πραγματοποιείται μέσα από τον έλεγχο αυτών πάνω σε ένα σύνολο από μέτρα απόδοσης και επίσης επαφίεται στην εμπειρία του αναλυτή πάνω στο πρόβλημα ταξινόμησης που εξετάζεται.

Επομένως η διαδικασία επιλογής των παραμέτρων αυτών, αποτελεί μια δύσκολη και χρονοβόρα διαδικασία για τον αναλυτή, καθώς οι συνδυασμοί που προκύπτουν μπορεί να είναι της τάξης των χιλιάδων (αν όχι παραπάνω).

3.2 ΜΟΝΤΕΛΟΠΟΙΗΣΗ ΜΕ ΠΟΛΥΚΡΙΤΗΡΙΑ ΑΝΑΛΥΣΗ ΑΠΟΦΑΣΕΩΝ

3.2.1. Εισαγωγή

Όπως αναφέρθηκε και παραπάνω κατά τη διάρκεια της δημιουργίας μιας ensemble είναι επιθυμητή η επιλογή των κατάλληλων αλγορίθμων για την εκπαίδευση των βασικών ταξινομητών.

Στο $1^{ο}$ κεφάλαιο μιλήσαμε για τα διάφορα μέτρα απόδοσης που υπάρχουν καθώς και για τις διαφορετικές υποθέσεις τα οποία αυτά βασίζονται. Επομένως η επιλογή μόνο ενός μέτρου απόδοσης θα αποτελούσε παράλειψη στο τρόπο ελέγχου ενός ταξινομητή. Σκοπός του αναλυτή είναι να επιλέξει ποιοι θα είναι οι αλγόριθμοι των βασικών ταξινομητών με βάση ένα σύνολο από μέτρα απόδοσης.

Αυτό το πρόβλημα μπορεί να εξεταστεί και ως ένα πολυκριτήριο πρόβλημα, καθώς η επιλογή των αλγορίθμων δεν εξαρτάται μόνο από ένα μέτρο απόδοσης αλλά από πολλά. Πιο συγκεκριμένα το παραπάνω πρόβλημα ανήκει στη προβληματική γ, όπου σκοπός είναι η κατάταξη ενός συνόλου εναλλακτικών.

Ο απώτερος στόχος είναι η δημιουργία ενός μοντέλου το οποίο θα μοντελοποιεί τις προτιμήσεις όσον αφορά ένα σύνολο από μέτρα απόδοσης και θα κατατάσσει τους εναλλακτικούς αλγόριθμους ταξινόμησης με βάση αυτές.

Παρακάτω θα γίνει μια σύντομη εισαγωγή στο τομέα της Πολυκρίτηριας Ανάλυσης Αποφάσεων και θα παρουσιαστεί μια από τις δημοφιλέστερες κλάσεις αλγορίθμων αυτού.

3.2.2. Πολυκριτήρια Ανάλυση Αποφάσεων: Η Αναλυτική-Συνθετική προσέγγιση

Η Πολυκριτηρία Ανάλυση Αποφάσεων (Multiple Criteria Decision Analysis) αποτελεί το τομέα ο οποίος ασχολείται με τον τρόπο επίλυσης προβλημάτων στα οποία περιέχονται πολλαπλά κριτήρια. Πιο συγκεκριμένα προσπαθεί να δώσει απάντηση στο ερώτημα «Πως πρέπει να παρθεί μια απόφαση δεδομένου ότι η έκβαση αυτής εξαρτάται από πολλαπλά κριτήρια;».

Το παραπάνω ερώτημα όμως μπορεί να διαμορφωθεί και με τον αντίθετο τρόπο, δηλαδή δεδομένης μιας ήδη υπάρχουσας απόφασης της οποίας η έκβαση εξαρτάται από πολλαπλά κριτήρια «Πως είναι δυνατό να βρεθεί η λογική με την οποία πάρθηκε αυτή η απόφαση;» ή «Πως είναι δυνατό να βρεθεί το προτιμησιακό μοντέλο που οδήγησε τον αποφασίζοντα σε αυτή την απόφαση;».

Στη πολυκριτηρία ανάλυση αποφάσεων υπάρχουν δύο προσεγγίσεις:

- Η χρησιμοποίηση ενός συνόλου μεθόδων/μοντέλων τα οποία μέσω της σύνθεσης πολλαπλών κριτηρίων έχουν τη δυνατότητα να επιλέξουν μια απόφαση από ένα σύνολο αποφάσεων A .
- Η χρησιμοποίηση μια διαδικασίας υποστήριξης της απόφασης σε συνεργασία με τον αποφασίζοντα.

Και στις δύο περιπτώσεις το σύνολο των αποφάσεων A αναλύεται σε πολλαπλά κριτήρια ώστε να μοντελοποιηθούν όλες οι εξαρτήσεις , οι συνέπειες και τα χαρακτηριστικά τα οποία σχετίζονται με τα το σύνολο των αποφάσεων A .

Τα προβλήματα αποφάσεων χωρίζονται σε 4 προβληματικές (κατηγορίες) ανάλογα με το στόχο που πρέπει να επιτευχθεί μέσα από την επίλυση τους:

- Προβληματική α: επιλογή (choice) μιας μοναδικής εναλλακτικής ενέργειας από το σύνολο A .
- Προβληματική β: ταξινόμηση (classification) των ενεργειών του συνόλου A σε ομογενής προκαθορισμένες κατηγορίες (κλάσεις)
- Προβληματική γ: κατάταξη (ranking) των ενεργειών του συνόλου A από τη καλύτερη μέχρι και τη χειρότερη.
- Προβληματική δ: περιγραφή (description) των ενεργειών του συνόλου A σε όρους κατανοητούς από τον αποφασίζοντα.

Παρακάτω παρουσιάζεται μια γενική μεθοδολογία για την επίλυση προβλημάτων απόφασης η οποία αποτελείται από 4 βήματα:

1. Αρχικά πρέπει να οριστεί το αντικείμενο της απόφασης, το σύνολο των πιθανών εναλλακτικών A , καθώς και η προβληματική στην οποία ανήκει το πρόβλημα.
2. Συνεχίζοντας, πρέπει να καθοριστεί μια συνεπής οικογένεια κριτηρίων, δηλαδή ένα σύνολο προτιμησιακών συναρτήσεων, όπου η κάθε συνάρτηση αποδίδει τις προτιμήσεις του αποφασίζοντα στο A όσον αφορά ένα συγκεκριμένο κριτήριο. Τα κριτήρια της συνεπής οικογένειας κριτηρίων πρέπει να έχουν της εξής ιδιότητες :
   - Μονοτονία: μια ενέργεια η οποία έχει μεγαλύτερη τιμή σε ένα συγκεκριμένο κριτήριο από μια άλλη προτιμάται έναντι της άλλης και αντίθετα.
   - Επάρκεια: τα κριτήρια περιγράφουν πλήρως το πρόβλημα και δεν λείπει από αυτά, κανένα σημαντικό ως προς το πρόβλημα κριτήριο.
   - Μη πλεονασμός: τα κριτήρια δεν πρέπει να είναι ίδια ή να περιέχουν μεγάλες ομοιότητες.
3. Έπειτα αναπτύσσεται ένα ολικό προτιμησιακό μοντέλο για τη σύνθεση των μερικών προτιμησιακών συναρτήσεων του αποφασίζοντα.
4. Τέλος υποβοηθιέται ή υποστηρίζεται η διαδικασία της απόφασης μέσω του ολικού προτιμησιακού μοντέλου που αναπτύχθηκε στο βήμα 3 και με βάση τον ορισμό της προβληματικής στο βήμα 1.

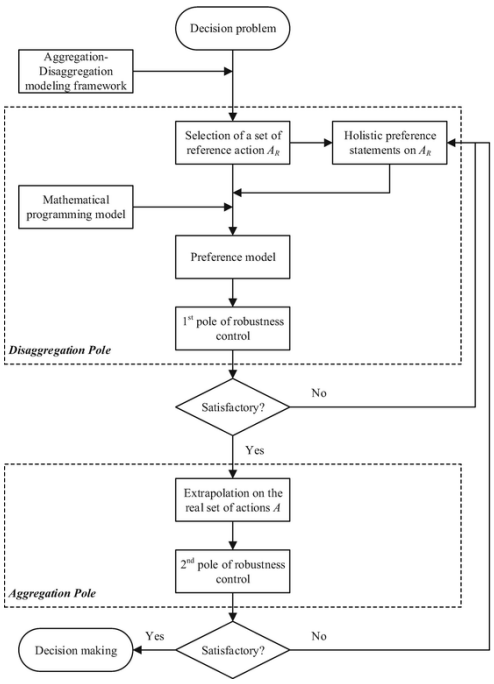
Στο παράδειγμα της σύνθεση των κριτηρίων που αναφέρεται και σε μορφή ερώτησης παραπάνω, το μοντέλο με το οποίο πραγματοποιείται η σύνθεση έτσι ώστε να παρθεί η τελική απόφαση είναι γνωστό εκ των προτέρων, ενώ οι τελικές προτιμήσεις είναι άγνωστες. Αντίθετα στη περίπτωση της αναλυτικής-συνθετικής προσέγγισης (aggregation – disaggregation approach) οι τελικές προτιμήσεις είναι γνωστές, μέσα από την ανάλυση των οποίων στις προτιμησιακές συναρτήσεις των χαρακτηριστικών προκύπτει το ολικό προτιμησιακό μοντέλο.

Αξιοσημείωτη είναι η αναγκαιότητα της ύπαρξης ενός συνόλου ενεργειών αναφοράς A_R για την διευκρίνιση της ολικής προτίμησης του αποφασίζοντα, το οποίο μπορεί να είναι:

a. ένα σύνολο από αποφάσεις που έχουν παρθεί σε προγενέστερες χρονικές στιγμές.
b. ένα υποσύνολο του συνόλου των ενεργειών A , ειδικά σε περιπτώσεις όπου το μέγεθος του A είναι αρκετά μεγάλο για να αναλυθεί ολόκληρο ($A_R \subseteq A$) .

c. Ένα σύνολο από τεχνητές/εικονικές ενέργειες , το οποίο ο αποφασίζοντας μπορεί με ευκολία να αξιολογήσει και να εκφράσει τις ολικές προτιμήσεις του πάνω σε αυτό.

Σε όλες τις παραπάνω περιπτώσεις, κατά τη διάρκεια της υποβοήθησης του αποφασίζοντα, ζητείται από αυτόν να εκφράσει τις προτιμήσεις του στο A_R λαμβάνοντας υπόψη τις τιμές που παίρνουν οι ενέργειες που περιέχονται σε αυτό, στα διάφορα κριτήρια του προβλήματος.



Εικόνα 3.2.2-1 Γραφική αναπαράσταση της διαδικασίας που ακολουθείται από την αναλυτική-συνθετική προσέγγιση.

Τέλος όπως φαίνεται και στην εικόνα 3.2.2-1 η διαδικασία της επιλογής ενός ολικού προτιμησιακού μοντέλου δεν αποτελεί μια στατική ή σειριακή διαδικασία, αλλά μια επαναλαμβανόμενη διαδικασία.

3.2.3. Η μέθοδος UTA

Η μέθοδος UTA [38] αποτελεί μια από τις δημοφιλέστερες μεθόδους της πολυκριτήριας ανάλυσης αποφάσεων η οποία αποτελεί τη βάση για μια ολόκληρη κλάση μεθόδων, οι οποίες είναι βασισμένες στην αναλυτική-συνθετική προσέγγιση. Σκοπός της μεθόδου είναι να καταλήξει σε μια η περισσότερες προσθετικές συναρτήσεις αξίας χρησιμοποιώντας την κατάταξη ενός δεδομένου συνόλου A_R .

Στη περίπτωση της μεθόδου UTA η σύνθεση των κριτηρίων γίνεται με τη βοήθεια μιας προσθετικής συνάρτησης αξιών η οποία έχει τη μορφή:

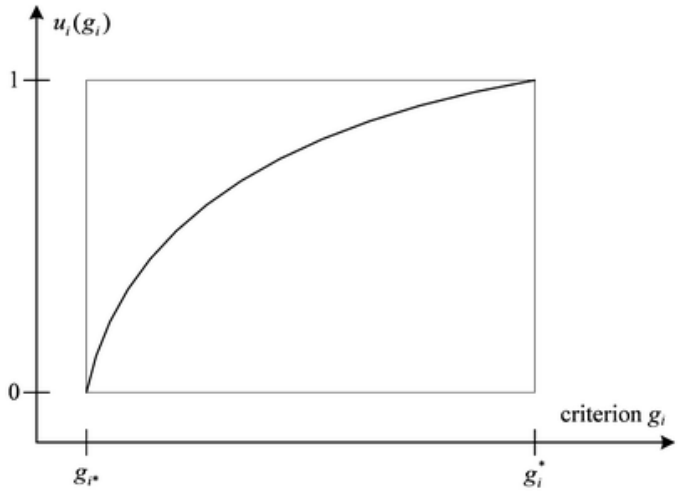
$$u(g) = \sum_{i=1}^{n} p_i u_i(g_i) \qquad (3.5)$$

υπό τους περιορισμούς:

$$\begin{cases} \sum_{i=1}^{n} p_i = 1 \\ u_i(g_i^*) = 0 \ \forall i = 1,2,...,n \end{cases} \quad (3.6)$$

Όπου $u_i(i = 1,2,...,n)$ είναι αύξουσες συναρτήσεις και κανονικοποιημένες στο $[0,1]$ που ονομάζονται μερικές συναρτήσεις αξίας και p_i είναι τα βάρη αυτών. Αυτές οι συναρτήσεις υπολογίζονται με τη χρήση τεχνικών γραμμικού προγραμματισμού. Να σημειωθεί ότι και στις δύο περιπτώσεις (μερικές και ολικές συναρτήσεις αξίας) ισχύει η ιδιότητα της μονοτονίας. Για παράδειγμα στην περίπτωση της ολικής συνάρτησης αξιών ισχύει ότι:

$$\begin{cases} u[g(a)] = u[g(b)] \Leftrightarrow a \sim b(\alpha\delta\iota\alpha\varphi o\rho\acute{\iota}\alpha) \\ u[g(a)] > u[g(b)] \Leftrightarrow a > b(\pi\rho o\tau\acute{\iota}\,\mu\eta\sigma\eta) \end{cases} \quad (3.7)$$



Εικόνα 3.2.3-1: Η κανονικοποιημένη ολική προσθετική συνάρτηση αξιών.

Στη περίπτωση της μεθόδου UTA εκτιμάται μια παραλλαγή της (3.5), όπου σε κάθε μερική συνάρτηση αξίας αποδίδεται βάρος ίσο με 1. Δηλαδή αποδίδεται η ίδια σημαντικότητα σε όλες τις μερικές συναρτήσεις αξιών.

$$u(g) = \sum_{i=1}^{n} u_i(g_i) \quad (3.8)$$

υπό τους περιορισμούς:

$$\begin{cases} \sum_{i=1}^{n} u_i(g_i^*) = 1 \\ u_i(g_i^*) = 0 \ \forall i = 1,2,...,n \end{cases} \quad (3.9)$$

Για να ισχύουν τα παραπάνω γίνεται η υπόθεση της προτιμησιακής ανεξαρτησίας των κριτηρίων από τον αποφασίζοντα.

Με βάση την (3.8) και τις σχέσεις προτιμήσεων στο (3.7) η αξία μια εναλλακτικής a, η οποία ανήκει στο A_R, μπορεί να γραφεί ως:

$$u'[g(a)] = \sum_{i=1}^{n} u_i[g_i(a)] + \sigma(a) \ \forall a \in A_R \qquad (3.10)$$

όπου $\sigma(a)$ είναι το πιθανό σφάλμα σε σχέση με το $u'[g(a)]$.

Για την εκτίμηση των μερικών συναρτήσεων αξίας σε μια γραμμικά κατά τμήματα μορφή, χρησιμοποιείται η γραμμική παρεμβολή. Επομένως για κάθε κριτήριο, το διάστημα των τιμών του $[g_{i*}, g_i^*]$ χωρίζεται σε $(\alpha_i - 1)$ ίσα υπό-διαστήματα, τα τελικά σημεία των οποίων δίνονται από τη σχέση:

$$g_i^{\ j} = g_{i*} + \frac{j-1}{\alpha_i - 1}(g_i^* - g_{i*}) \ \forall j = 1, 2, .., \alpha_i \qquad (3.11)$$

Η μερική αξία μιας εναλλακτικής ενέργειας a εκτιμάται μέσω του τύπου της γραμμικής παρεμβολής, δηλαδή για κάθε $g_i(a) \in [g_i^j - g_i^{j+1}]$:

$$u_i[g_i(a)] = u_i(g_i^j) + \frac{g_i(a) - g_i^j}{g_i^{j+1} - g_i^j}[u_i(g_i^{j+1}) - u_i(g_i^j)] \qquad (3.12)$$

Επιπρόσθετα οι εναλλακτικές ενέργειες $a_1, a_2, ..., a_m$ του συνόλου αναφοράς A_R «ανακατατάσσονται» με τέτοιο τρόπο ώστε να συνάδουν με τις προτιμήσεις του αποφασίζοντα από τη καλύτερη στη χειρότερη. Δηλαδή το a_1 να συμβολίζει τη καλύτερη εναλλακτική και το a_m τη χειρότερη εναλλακτική ενέργεια. Εφόσον η παραπάνω κατάταξη αποτελεί μια μορφή προδιάταξης R, τότε για κάθε $(a_k, a_k + 1)$ θα ισχύει είτε $a_k > a_{k+1}$ (προτίμηση), είτε $a_k \sim a_{k+1}$ (αδιαφορία).

Άρα αν θέσουμε

$$\Delta(a_k, a_{k+1}) = u'[g(a_k)] - u'[g(a_{k+1})] \qquad (3.13)$$

τότε θα ισχύει πάντα μια από τις παρακάτω περιπτώσεις:

$$\begin{cases} \Delta(a_k, a_{k+1}) \geq \delta, \ \text{αν } a_k > a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0, \ \text{αν } a_k \sim a_{k+1} \end{cases} \qquad (3.14)$$

Όπου δ είναι ένας πολύ μικρός θετικός αριθμός μέσω του οποίου είναι δυνατός ο διαχωρισμός δύο διαδοχικών κλάσεων της R.

Αν λάβουμε υπόψη μας το κριτήριο της μονοτονίας που πρέπει να ακολουθεί κάθε μερικής συνάρτηση αξίας, τότε οι μερικές αξίες πρέπει να ακολουθούν τους παρακάτω περιορισμούς:

$$u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \ \forall j = 1, 2, ..., \alpha - 1, \ i = 1, 2, .., n \qquad (3.15)$$

Όπου s_i είναι τα κατώφλια αδιαφορίας, τα οποία ορίζονται για κάθε κριτήριο g_i.

Η χρησιμοποίηση των κατωφλιών δεν είναι υποχρεωτική, όμως βοηθάνε για την αποφυγή φαινομένων όπου η μερική αξία μια εναλλακτικής είναι η ίδια με μια άλλη ενώ, η τιμή του κριτηρίου της μιας είναι μεγαλύτερη από αυτή της άλλης.

Οι μερικές συναρτήσεις αξίας εκτιμούνται με βάση το παρακάτω γραμμικό πρόγραμμα, το οποίο έχει μια αντικειμενική συνάρτηση η οποία εξαρτάται από το σφάλμα $\sigma(a)$ συμβολίζοντας το συνολικό ποσοστό της διασποράς.

$$\begin{cases} [\min] F = \sum_{a \in A_R} \sigma(a) \\ \upsilon\pi o: \\ \left.\begin{array}{l} \Delta(a_k, a_{k+1}) \geq \delta, \ \alpha\nu \ a_k > a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0, \ \alpha\nu \ a_k \sim a_{k+1} \end{array}\right\} \forall k \\ u_i(g_i^{j+1}) - u_i(g_i^j) \geq s_i \ \ \forall i, j \\ \sum_{i=1}^{n} u_i(g_i^*) = 1 \\ u_i(g_{i^*}) = 0, \ u_i(g_i^j) \geq 0, \ \sigma(a) \geq 0 \ \in A_R, \ \forall i \ \kappa\alpha\iota \ j \end{cases} \qquad (3.16)$$

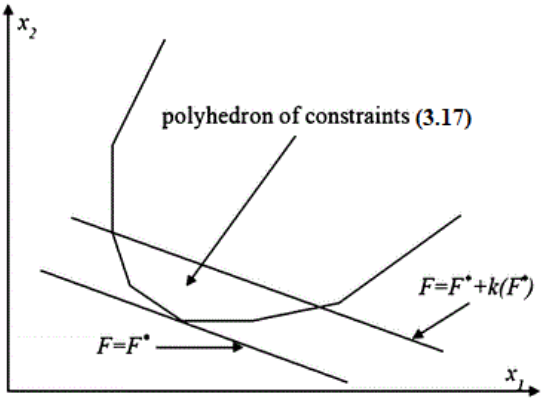
Η ανάλυση ευαισθησίας των αποτελεσμάτων που προκύπτουν από το παραπάνω γραμμικό πρόβλημα αποτελεί ένα πρόβλημα μετά-βελτιστοποίησης. Αν το βέλτιστο είναι $F^* = 0$, τότε το υπέρ-πολύεδρο των αποδεκτών λύσεων δεν είναι κενό και υπάρχουν πολλαπλές συναρτήσεις μερικών αξίων, οι οποίες μπορούν να οδηγήσουν στη προδιάταξη R.

Ακόμα και αν οι λύσεις του γραμμικού προβλήματος δεν είναι βέλτιστες αλλά ημιβέλτιστες, αυτές μπορούν να βελτιώσουν άλλα κριτήρια όπως το τ του kendall, δεδομένου ότι είναι αυστηρά θετικές.

Ο χώρος των μετά-βέλτιστων λύσεων ορίζεται από το παρακάτω υπέρ-πολύεδρο:

$$\begin{cases} F \leq F^* + k(\mathrm{F}^*) \\ \text{όλοι οι περιορισμοί του (3.16)} \end{cases} \qquad (3.17)$$

Όπου $k(\mathrm{F}^*)$ είναι ένα θετικό κατώφλι, το οποίο αποτελεί ένα μικρό ποσοστό της βέλτιστης λύσης F^*.



Εικόνα 3.2.3-2: Ανάλυση ευαισθησίας στη μέθοδο UTA.

Η επίλυση του προβλήματος της ανάλυσης ευαισθησίας του (3.16) πραγματοποιείται με την επίλυση των παρακάτω γραμμικών προβλημάτων:

$$\begin{cases} [\min] u_i(g_i^*) \ \kappa\alpha\iota \ [\max] u_i(g_i^*) \\ \sigma\tau o \qquad\qquad\qquad\qquad\qquad \forall i = 1,2,...,n \ (3.18) \\ \pi o\lambda\acute{\upsilon}\varepsilon\delta\rho o \ (3.17) \end{cases}$$

Τέλος ως λύση θεωρείται ο μέσος όρος των παραπάνω ΓΠ. Στη περίπτωση όπου υπάρχει μεγάλη αστάθεια, η λύση που περιέχει το μέσο όρο των παραπάνω ΓΠ είναι λιγότερο αντιπροσωπευτική, λόγω της μεγάλης απόκλισης των επιμέρους λύσεων. Αξιοσημείωτο είναι το γεγονός ότι με τη λύση των παραπάνω ΓΠ, μπορούμε να δούμε τη εσωτερική διασπορά των βαρών όλων των κριτηρίων g_i, καθώς και να πάρουμε μια ιδέα για τη σημαντικότητα αυτών των κριτηρίων στο προτιμησιακό μοντέλο του αποφασίζοντα.

3.2.4. Ο αλγόριθμος UTASTAR

Ο αλγόριθμος UTASTAR[38] αποτελεί μια βελτιωμένη έκδοση της μεθόδου UTA. Ειδικότερα στην αρχική έκδοση της μεθόδου UTA αντιστοιχείτε ένα σφάλμα $\sigma(a)$, το οποίο πρέπει να ελαχιστοποιηθεί, με κάθε εναλλακτική ενέργεια a του συνόλου A_R. Η συγκεκριμένη συνάρτηση σφάλματος δεν αρκεί για να ελαχιστοποιηθεί η διασπορά των σημείων γύρω από τη καμπύλη μονότονης παλινδρόμησης της εικόνας (3.9). Το πρόβλημα εντοπίζεται στα σημεία δεξιά της καμπύλης από τα οποία θα ήταν πιο επιθυμητό να αφαιρεθεί μια ποσότητα αξίας, αντί να αυξηθεί η αξία των υπολοίπων.

Ο αλγόριθμος UTASTAR έρχεται να επιλύσει το παραπάνω πρόβλημα με την εισαγωγή μια διπλής συνάρτησης σφάλματος:

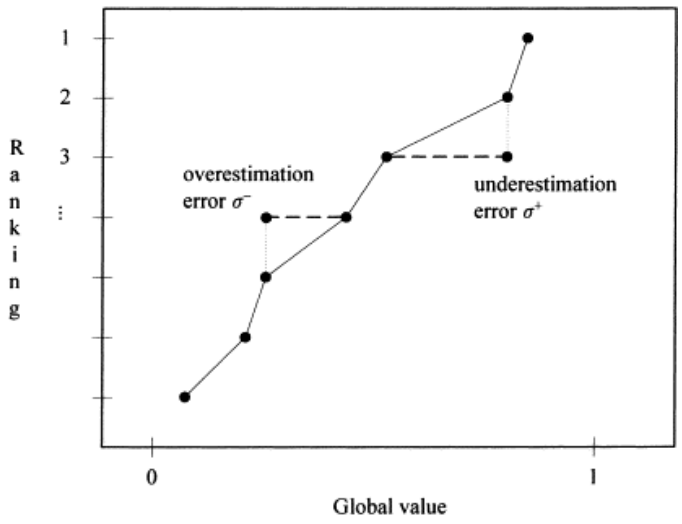
$$u'[g(a)] = \sum_{i=1}^{n} u_i[g_i(a)] - \sigma^+(a) + \sigma^-(a) \ \forall a \in A_R \ . \qquad (3.19)$$

Όπου $\sigma^+(a)$, $\sigma^-(a)$ είναι τα σφάλματα υπερεκτίμησης και υποεκτίμησης αντίστοιχα.

Επιπλέον μια άλλη προσθήκη του αλγόριθμου UTASTAR στην αρχική μέθοδο UTA, είναι η μετατροπή των περιορισμών που έχουν σχέση με τη μονοτονία των μερικών συναρτήσεων αξίας μέσω των μεταβλητών:

$$w_{ij} = u_i(g_i^{j+1}) - u_i(g_i^j) \geq 0 \ \forall i = 1,2,....,n \ \kappa\alpha\iota \ j = 1,2,...,\alpha_i - 1 . \ (3.20)$$

Άρα οι περιορισμοί αυτοί μπορούν να αντικατασταθούν με περιορισμούς μη αρνητικότητας των μεταβλητών w_{ij} (για $s_i = 0$).



Εικόνα 3.2.4-1: Καμπύλη μονότονης παλινδρόμησης.

Συμπεραίνοντας ο αλγόριθμός UTASTAR μπορεί να περιγραφεί από τα παρακάτω βήματα:

1. Έκφραση της ολικής αξίας των εναλλακτικών του συνόλου αναφοράς $u[g(\mathrm{a}_k)], k = 1,2,...,m$, αρχικά σε σχέση με τις μερικές αξίες $u_i(\mathrm{g}_i)$ και στη συνέχεια σε σχέση με τις τεχνητές μεταβλητές w_{ij} μέσω της σχέσης (3.20), με βάση τις παρακάτω σχέσεις:

$$\begin{cases} u_i(g_i^1) = 0 \qquad \forall i = 1,2,....,n \\ u_i(g_i^j) = \sum_{i=1}^{j-1} w_{ij} \forall i = 1,2,...,n \ \kappa\alpha\iota \ j = 1,2,...,\alpha_i - 1 \end{cases} \tag{3.21}$$

2. Εισαγωγή των δύο συναρτήσεων σφάλματος $\sigma^+(a)$, $\sigma^-(a)$ στο A_R, γράφοντας για κάθε ζεύγος εναλλακτικών στη κατάταξη R τις παρακάτω αναλυτικές σχέσεις:

$$\begin{aligned} \Delta(a_k, a_{k+1}) = u[g(a_k)] - \sigma^+(a_k) + \sigma^-(a_k) \\ - u[g(a_{k+1})] + \sigma^+(a_{k+1}) - \sigma^-(a_{k+1}) \end{aligned} \tag{3.22}$$

3. Επίλυση του ΓΠ:

$$\begin{cases} [\min] z = \sum_{k=1}^{m} [\sigma^+(a_k) + \sigma^-(a_k)] \\ \upsilon\pi\acute{o}: \\ \left.\begin{array}{l} \Delta(a_k, a_{k+1}) \geq \delta \ \alpha\nu \ a_k \succ a_{k+1} \\ \Delta(a_k, a_{k+1}) = 0 \ \ \alpha\nu \ a_k \sim a_{k+1} \end{array}\right\} \forall k \\ \sum_{i=1}^{n} \sum_{j=1}^{\alpha_i - 1} w_{ij} = 1 \\ w_{ij} \geq 0, \ \sigma^+(a_k) \geq 0, \ \sigma^-(a_k) \geq 0 \ \forall i, j \ \kappa\alpha\iota \ k \end{cases} \tag{3.23}$$

όπου το δ είναι ένα μικρός θετικός αριθμός.

4. Έλεγχος για ύπαρξη πολλαπλών ημιβέλτιστων λύσεων του ΓΠ (ανάλυση ευαισθησίας). Σε περίπτωση μη μοναδικότητας, βρίσκεται η μέση προσθετική αξία αυτών των ημιβέλτιστων λύσεων που μεγιστοποιούν τις αντικειμενικές συναρτήσεις:

$$u_i(g_i^*) = \sum_{j=1}^{\alpha_i - 1} w_{ij} \forall i = 1, 2, ..., n \,, \qquad (3.24)$$

στο υπέρ-πολύεδρο των περιορισμών του ΓΠ, το οποίο οριοθετείται από το νέο περιορισμό:

$$\sum_{k=1}^{m} [\sigma^+(a_k) + \sigma^-(a_k)] \leq z^* + \varepsilon \qquad (3.25)$$

Όπου z^* είναι η βέλτιστη λύση του ΓΠ και ε ένας πολύ μικρός θετικός αριθμός.

Οι μέθοδοι που παρουσιάστηκαν παραπάνω μπορούν να χρησιμοποιηθούν για τη μοντελοποίηση του προβλήματος που αναφέρθηκε στην εισαγωγή του παρόντος υποκεφαλαίου, μοντελοποιώντας τις προτιμήσεις του αναλυτή πάνω σε ένα σύνολο από ταξινομητές. Για την αναπαράσταση των προτιμήσεων του αναλυτή πάνω στα μέτρα απόδοσης μπορούν να χρησιμοποιηθούν οι μερικές συναρτήσεις αξίας ή τα βάρη των κριτηρίων που προκύπτουν από την εφαρμογή των παραπάνω μεθόδων πάνω σε ένα συγκεκριμένο σύνολο δεδομένων, ταξινομητών και μέτρων απόδοσης.

ΚΕΦΑΛΑΙΟ 4

4.1 ΠΡΟΤΕΙΝΟΜΕΝΗ ΜΕΘΟΔΟΛΟΓΙΑ

4.1.1. Εισαγωγή

Σε αυτό το κεφάλαιο θα παρουσιαστεί το μεθοδολογικό πλαίσιο στο οποίο βασίστηκε το προτεινόμενο σύστημα υποστήριξης αποφάσεων MuCE, για την υποβοήθηση ενός αναλυτή στη δημιουργία μιας classification ensemble. Η παρούσα μεθοδολογία βασίζεται στο συνδυασμό των βασικών ταξινομητών μέσω του κανόνα της σταθμισμένης πλειοψηφίας(weighted majority voting)[39], η επιλογή και βαροδότηση των οποίων πραγματοποιείται με τη χρησιμοποίηση της πολυκριτήριας ανάλυσης αποφάσεων. Η βασική υπόθεση που κάνει είναι ότι οι προτιμήσεις ενός αναλυτή πάνω σε ένα σύνολο από ταξινομητές μπορούν να αναλυθούν σε όρους ενός συνόλου μέτρων απόδοσης M_A.

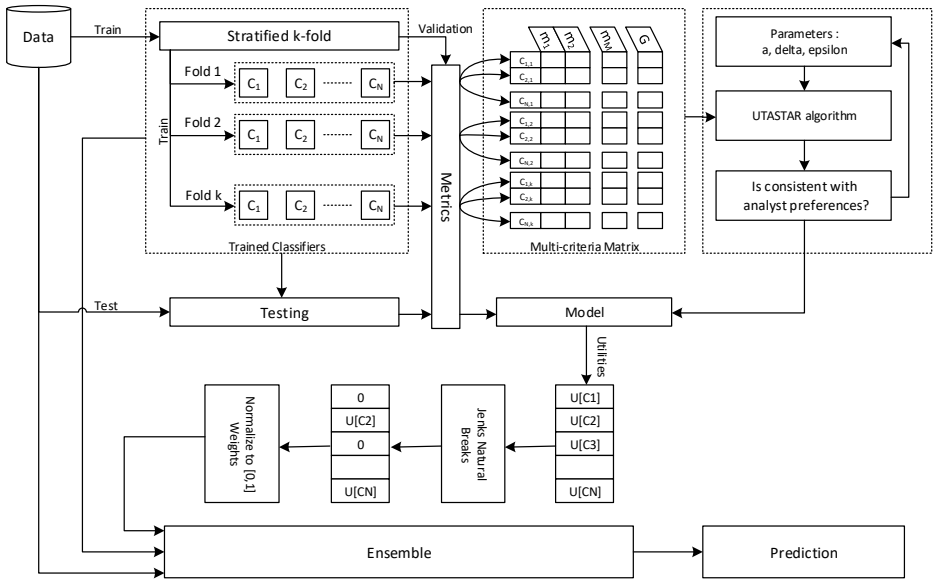
Οι στόχοι αυτής της προσέγγισης είναι:

- Να επιλεγούν οι καταλληλότεροι αλγόριθμοι ταξινόμησης για το εξεταζόμενο πρόβλημα ταξινόμησης από ένα σύνολο αλγορίθμων A_c .
- Να αποδοθούν βάρη σε κάθε ταξινομητή τα οποία συνδέονται άμεσα με τις προτιμήσεις του αναλυτή και μέσα από τα οποία μπορεί να αποδοθεί και η εμπειρία του αναλυτή πάνω στο εξεταζόμενο πρόβλημα ταξινόμησης.
- Να επιλεγούν οι καλύτεροι από τους παραγόμενους ταξινομητές για τη δημιουργία της ensemble.
- Η γνώση που αποκτάται από το σύστημα να είναι κατανοητή από τον αναλυτή.
- Η αποθήκευση της προϋπάρχουσας γνώσης, καθώς και της γνώσης που απέκτησε ο αναλυτής μέσα από τη διαδικασία της δημιουργίας της ensemble για χρήση σε μεταγενέστερες χρονικές στιγμές.
- Η βελτίωση της απόδοσης της δημιουργούμενης ensemble με την εισαγωγή μια μορφής νοημοσύνης στο τρόπο δημιουργίας της.

Η παρούσα μεθοδολογία συνοψίζεται στα παρακάτω κύρια στάδια:

1. Επιλογή του συνόλου των διαθέσιμων αλγορίθμων, του συνόλου δεδομένων και του συνόλου των μέτρων απόδοσης.
2. Διαχωρισμός του συνόλου δεδομένων
3. Εκπαίδευση και έλεγχος των βασικών ταξινομητών
4. Μοντελοποίηση και προσέγγιση των προτιμήσεων του αναλυτή με βάση τη πολυκριτήρια ανάλυση αποφάσεων.
5. Επιλογή των καλύτερων ταξινομητών με βάση το προτιμησιακό μοντέλο που αποκτήθηκε στο βήμα 4 και μεθόδων συσταδοποίησης.
6. Βαροδότηση των επιλεγμένων ταξινομητών με βάση το προτιμησιακό μοντέλο που αποκτήθηκε στο βήμα 4.
7. Δημιουργία της ensemble με βάση το κανόνα της σταθμισμένης πλειοψηφίας.

Παρακάτω θα αναλυθούν περαιτέρω αυτά τα βήματα έτσι ώστε να αποσαφηνιστεί ο τρόπος λειτουργίας του παρόντος συστήματος υποστήριξης αποφάσεων, καθώς και ο λόγος ύπαρξης αυτών. Μια επισκόπηση αυτού φαίνεται στην εικόνα 4.1.1-1 .



Εικόνα 4.1.1-1: Γραφική αναπαράσταση της διαδικασίας που ακολουθείται από το σύστημα υποστήριξης αποφάσεων MuCe.

4.1.2. Στάδιο 1^{o} :Επιλογή συνόλων

Σε αυτό το στάδιο ζητείται από τον αναλυτή να επιλέξει το σύνολο δεδομένων T το οποίο θα χρησιμοποιήσει για να εκπαιδεύσει την ensemble. Εκτός από αυτό το σύνολο πρέπει να επιλέξει και το σύνολο των αλγορίθμων που θα χρησιμοποιηθούν A_c, καθώς και το σύνολο των μέτρων απόδοσης M_A στο οποίο θα αξιολογηθούν οι παραγόμενοι ταξινομητές. Ο τρόπος επιλογής των μέτρων και των αλγορίθμων επαφίεται στη κρίση του αναλυτή και εξαρτάται από την εμπειρία του πάνω στο εξεταζόμενο πρόβλημα ταξινόμησης.

4.1.3. Στάδιο 2^{o} : Διαχωρισμός του συνόλου δεδομένων

Στο 2^{o} στάδιο πραγματοποιείται ο διαχωρισμός του συνόλου δεδομένων χρησιμοποιώντας τη μέθοδο stratified k-fold cross validation, η οποία αναφέρθηκε στο 1^{o} κεφάλαιο. Αρχικά το σύνολο δεδομένων T χωρίζεται σε δύο υποσύνολα, το σύνολο εκπαιδεύσεως T_{tr} και το σύνολο ελέγχου T_t και στη συνέχεια το T_{tr} χωρίζεται σε k folds με τη μέθοδο του stratified k-fold cross validation.

Μέσω αυτής της διαδικασίας σκοπός είναι η παραγωγή διαφορετικών ταξινομητών στο στάδιο 3, λόγω της διαφορετικότητας που υπάρχει μεταξύ των παραγόμενων υποσυνόλων δεδομένων. Επιπλέον με αυτό τον τρόπο δημιουργείται η βάση, για τη δημιουργία ενός επαρκούς συνόλου δεδομένων για τη κατασκευή του προτιμησιακού μοντέλου του αναλυτή στο στάδιο 4. Τέλος μέσω του stratification που περιέχεται στη διαδικασία καταπολεμάται το πρόβλημα της δυσαναλογίας των κλάσεων στο σύνολο των δεδομένων.

4.1.4. Στάδιο 3^{ο} : Εκπαίδευση και έλεγχος των βασικών ταξινομητών

Σε αυτό το βήμα παράγονται και ελέγχονται οι βασικοί ταξινομητές με βάση τα υποσύνολα δεδομένων που δημιουργήθηκαν στο στάδιο 2. Πιο συγκεκριμένα για κάθε αλγόριθμο $A \in A_c$ και για κάθε fold από αυτά που δημιουργήθηκαν στο στάδιο 2 εκπαιδεύεται ένας ταξινομητής. Μετά την εκπαίδευση των ταξινομητών, εκτιμάται η απόδοση του κάθε ταξινομητή στο εναπομείναν υποσύνολο δεδομένων του κάθε fold, το οποίο δεν χρησιμοποιείται στην εκπαίδευση του εκάστοτε ταξινομητή. Ο υπολογισμός της απόδοσης γίνεται με βάση κάθε μέτρο απόδοσης $m_A \in M_A$.

Στην ουσία τα στάδια 2, 3 αποτελούν τη βάση για τη δημιουργία μιας cross validation committee, η οποία έχει αποδειχθεί ότι παράγει διαφορετικούς ταξινομητές, των οποίων τα σφάλματα είναι ασυσχέτιστα. Η προσθήκη των διαφορετικών αλγορίθμων εκπαίδευσης αποσκοπεί στην περαιτέρω αύξηση της διαφορετικότητας μεταξύ των ταξινομητών.

4.1.5. Στάδιο 4^{ο} : Μοντελοποίηση με χρήση της πολυκριτήριας ανάλυσης αποφάσεων

Σε αυτό το στάδιο αρχικά ζητείται από τον αναλυτή να δώσει μια αρχική προδιάταξη R για όλο το σύνολο των παραγόμενων ταξινομητών. Στη συνέχεια δημιουργείται ένας πίνακας όπου στις στήλες του βρίσκονται όλα τα μέτρα απόδοσης που ανήκουν στο M_A και στις γραμμές όλοι διαφορετικοί ταξινομητές οι οποίοι παράχθηκαν στο στάδιο 3. Η συμπλήρωση του πίνακα γίνεται με τις εκτιμήσεις των μέτρων απόδοσης που πραγματοποιήθηκαν στο στάδιο 3. Στο πίνακα αυτό προστίθεται και η στήλη της προδιάταξης R, η οποία δόθηκε από τον αναλυτή. Στην ουσία ο πίνακας αυτός αποτελεί ένα «Πολυκριτήριο Πίνακα» ο οποίος περιγράφει το πρόβλημα προς επίλυση. Στη συγκεκριμένη περίπτωση οι διαφορετικοί παραγόμενοι ταξινομητές αποτελούν το σύνολο αναφοράς A_R των εναλλακτικών ενεργειών και τα μέτρα απόδοσης αποτελούν τα κριτήρια του πολυκριτήριου προβλήματος.

Πίνακας 3.1.2.4-1
Παράδειγμα παραγόμενου πολυκριτήριου πίνακα. Όπου a_{ij} η εκτίμηση της απόδοσης του ταξινομητή i με το μέτρο j και R_i η κατάταξη του ταξινομητή i, η οποία δόθηκε από τον αναλυτή .

Alt/Cri	MCC	F1	Accuracy	Ranking
SVM Fold 0	$a_{1,1}$	$a_{1,2}$	$a_{1,3}$	R_1
SVM Fold 0	$a_{2,1}$	$a_{2,2}$	$a_{2,3}$	R_2
SVM Fold 1	$a_{3,3}$	$a_{3,2}$	$a_{3,3}$	R_3
SVM Fold 2	$a_{4,4}$	$a_{4,2}$	$a_{4,3}$	R_4
MLP Fold 0	$a_{5,5}$	$a_{5,2}$	$a_{5,3}$	R_5
MLP Fold 1	$a_{6,6}$	$a_{6,2}$	$a_{6,3}$	R_6
MLP Fold 2	$a_{7,7}$	$a_{7,2}$	$a_{7,3}$	R_7

Όπως αναφέρθηκε παραπάνω σκοπός είναι η δημιουργία ενός μοντέλου το οποίο θα είναι σε θέση να κατατάσσει ένα οποιοδήποτε ταξινομητή με βάση τις προτιμήσεις του

αναλυτή πάνω στο σύνολο αναφοράς A_R. Για τη δημιουργία αυτού του μοντέλου η MuCE χρησιμοποιεί τον αλγόριθμο UTASTAR. Όπου οι μερικές συναρτήσεις αξίας οι οποίες παράγονται αναπαριστούν τις προτιμήσεις του αναλυτή για κάθε μέτρο απόδοσης και η ολική συνάρτηση αξίας συμβολίζει το προτιμησιακό μοντέλο του αναλυτή για το πρόβλημα της κατάταξης των ταξινομητών.

Για την εφαρμογή του αλγόριθμου UTASTAR ο αναλυτής οφείλει να ορίσει της παραμέτρους α_j,δ και ε. Το α_j συμβολίζει τον αριθμό των υπό-διαστημάτων στα οποία θα χωριστούν οι τιμές του κριτηρίου j και στην ουσία καθορίζει και την ακρίβεια της μερικής συνάρτησης η οποία θα βρεθεί. Το δ αποτελεί το κατώφλι αδιαφορίας μεταξύ δύο εναλλακτικών και το ε είναι ένας πολύ μικρός αριθμός ο οποίος χρησιμοποιείται στο βήμα της μεταβελτιστοποιήσης που περιέχεται στον αλγόριθμο UTASTAR.

Γενικά πρέπει να ισχύει η σχέση:

$$\textit{αριθμός εναλλακτικών} = (a_j - 1)(\textit{αριθμός κριτηρίων}) \quad (4.1)$$

και ως αρχική τιμή για το δ επιλέγεται το:

$$\delta = \frac{1}{\textit{αριθμός εναλλακτικών}} \qquad (4.2)$$

Η παραπάνω διαδικασία επιλογής των παραμέτρων επαναλαμβάνεται μέχρι το παραγόμενο μοντέλο να συνάδει όσο τον δυνατό γίνεται με τις προτιμήσεις του αναλυτή.

Μετά το πέρας της διαδικασίας εύρεσης του προτιμησιακού μοντέλου του αναλυτή υπολογίζονται οι ολικές αξίες των παραγόμενων ταξινομητών στο υποσύνολο ελέγχου κάθε fold, ανάλογα με το fold στο οποίο εκπαιδεύθηκε ο εκάστοτε ταξινομητής. Οι ολικές αυτές αξίες κατ' ουσία συμβολίζουν τη σημαντικότητα των βασικών ταξινομητών σύμφωνα με τις προτιμήσεις του αναλυτή.

4.1.6. Στάδιο 5^{o}: Εύρεση των βαρών και συνδυασμός των ταξινομητών

Όπως αναφέρθηκε στην αρχή του κεφαλαίου το MuCE χρησιμοποιεί το κανόνα της σταθμισμένης πλειοψηφίας(weighted majority voting) για το συνδυασμό των προβλέψεων των βασικών ταξινομητών. Ειδικότερα ανάλογα με τη μορφή των προβλέψεων υπάρχουν δύο παραλλαγές του weighted majority voting, το hard voting και το soft voting.

Το hard voting προϋποθέτει ότι οι προβλέψεις των βασικών ταξινομητών είναι σε μορφή ετικέτας και η τελική πρόβλεψη δίνεται από το τύπο:

$$\hat{y} = \arg\max_i \sum_{j=1}^{m} w_j \chi_\Omega\left(C_j(x) = i\right), \quad (4.3)$$

όπου χ_Ω είναι η χαρακτηριστική συνάρτηση $[C_j(x) = i \in \Omega]$, $C_j(\mathbf{x})$ η πρόβλεψη του ταξινομητή j για το πρότυπο x, w_j το βάρος του ταξινομητή j και Ω το σύνολο των κλάσεων του προβλήματος ταξινόμησης.

Από την άλλη μεριά το soft voting προϋποθέτει ότι οι προβλέψεις των βασικών ταξινομητών είναι σε μορφή πιθανοτήτων. Ειδικότερα η πρόβλεψη ενός ταξινομητή απο-

τελείται από ένα διάνυσμα $P_j(x) = [p_1, p_2, ..., p_i, ..., p_n]$ μεγέθους n, όπου το στοιχείο στη θέση i αποτελεί τη πιθανότητα το συγκεκριμένο πρότυπο να ανήκει στη κλάση i. Η τελική πρόβλεψη στη περίπτωση του soft voting δίνεται από το τύπο:

$$\hat{y} = \arg\max_i \sum_{j=1}^{m} w_j p_{ij}, \quad (4.4)$$

όπου p_{ij} είναι η πιθανότητα που δίδει ο ταξινομητής j στο ενδεχόμενο το συγκεκριμένο πρότυπο να ανήκει στη κλάση i και w_j το βάρος του ταξινομητή j.

Στη περίπτωση του MuCE χρησιμοποιείται soft voting καθώς οι πιθανότητες περιέχουν παραπάνω πληροφορία από ότι οι ετικέτες των κλάσεων. Κάτι τέτοιο ισχύει αφού μέσω των πιθανοτήτων εκτός από το πια κλάση ανήκει το πρότυπο, μπορούμε να δούμε και το βάρος το οποίο δίνει στην απόφαση του ο εκάστοτε ταξινομητής.

Επιπλέον ως βάρη στο weighted majority voting χρησιμοποιούνται οι κανονικοποιημένες ολικές αξίες των βασικών ταξινομητών, καθώς όπως αναφέρθηκε και στο στάδιο 4 οι ολικές αξίες συμβολίζουν τη σημαντικότητα των ταξινομητών σύμφωνα με τις προτιμήσεις του αναλυτή. Το βάρος ενός ταξινομητή θα δίνεται λοιπόν από τη σχέση:

$$w_j = \frac{u[g(a_j)]}{\sum_{i=1}^{m} u[g(a_i)]} = \frac{\sum_{i=1}^{z} u_i[g_i(a_j)]}{\sum_{k=1}^{m}\sum_{i=1}^{z} u_i[g_i(a_k)]} \forall j = 1, 2, ..., m, \quad (4.5)$$

όπου $u[g(a_j)]$ είναι η ολική αξία του ταξινομητή j, $u_i[g_i(a_j)]$ η μερική αξία του ταξινομητή j στο κριτήριο i και z ο αριθμός των μέτρων απόδοσης που ανήκουν στο M_A.

Αξιοσημείωτο είναι το γεγονός ότι στο στάδιο της πρόβλεψης, για τον έλεγχο της απόδοσης της ensemble, τα μέτρα απόδοσης των ταξινομητών δεν υπολογίζονται στο T_{tr} αλλά στο T_t. Αυτό συμβαίνει έτσι ώστε η εκτίμηση της απόδοσης της ensemble να αντικατοπτρίζει τη πραγματικότητα.

4.1.7. Στάδιο 6ο: Επιλογή των καλύτερων ταξινομητών με τη χρήση συσταδοποίησης.

Σε αυτό το στάδιο πραγματοποιείται ένα «κλάδεμα» της αρχικά δημιουργούμενης ensemble σύμφωνα με τις ολικές αξίες των βασικών ταξινομητών. Όπως αναφέρθηκε και στην αρχή του κεφαλαίου εκτός από την ύπαρξη διαφορετικότητας μεταξύ των βασικών ταξινομητών είναι επιθυμητή και η υψηλή απόδοση αυτών. Ο τρόπος δημιουργίας μιας ensemble που έχει παρουσιασθεί μέχρι τώρα δεν εξασφαλίζει την υψηλή απόδοση των βασικών ταξινομητών. Το πρόβλημα αυτό επιλύει η MuCE με τη χρησιμοποίηση συσταδοποίησης για την αναγνώριση πιθανών ομάδων υπεροχής, με βάση τις ολικές αξίες που πήραν οι ταξινομητές από τον αλγόριθμο UTASTAR. Κάτι τέτοιο είναι δυνατό καθώς στη συγκεκριμένη περίπτωση η συσταδοποίηση γίνεται σε μια διάσταση, αυτή των ολικών αξιών των ταξινομητών. Στη περίπτωση αυτή οι κατηγορίες που προκύπτουν αποτελούν στην ουσία διαστήματα τιμών. Επομένως μπορούν να γίνουν άμεσες συγκρίσεις μεταξύ των κατηγοριών. Ειδικότερα η μέθοδος που χρησιμοποιείται

από τη MuCE για τη συσταδοποίηση των ολικών αξιών των ταξινομητών είναι αυτή του Jenks natural breaks optimization.

4.1.7.1. Jenks Natural Breaks Optimization

Το Jenks natural breaks[40] optimization είναι μια μέθοδος συσταδοποίησης για μια διάσταση. Αρχικά κατασκευάστηκε με σκοπό το βέλτιστο διαχωρισμό ενός συνόλου τιμών Z σε κλάσεις έτσι ώστε να αποτυπωθούν σε ένα χωροπληθικό χάρτη. Ο βασικός στόχος της μεθόδου είναι να ελαχιστοποιήσει τη διασπορά των τιμών στο εσωτερικό των κλάσεων αλλά ταυτόχρονα να μεγιστοποιήσει τη διασπορά των τιμών μεταξύ των κλάσεων.

Τα κύρια βήματα του αλγορίθμου συνοψίζονται παρακάτω[41]:

1. Αρχικά επιλέγεται ο αριθμός των κλάσεων λ στις οποίες θα χωριστεί το σύνολο των τιμών και στη συνέχεια κάθε τιμή αντιστοιχίζεται με μια κλάση με αυθαίρετο ή τυχαίο τρόπο.
2. Έπειτα υπολογίζεται το άθροισμα των τυπικών αποκλίσεων μεταξύ των κλάσεων (Sum of squared deviations between classes) με το τύπο:

$$SBDC = \sum_{i=1}^{\lambda} \sum_{j=1}^{n} \left(z_{ij} - \overline{z_i} \right)^2 \quad (4.6)$$

   όπου z_{ij} η i-οστή τιμή της j-οστης κλάσης, η οποία ανήκει στο Z και $\overline{z_i}$ η μέση τιμή της i-οστης κλάσης.

3. Εν συνεχεία υπολογίζεται το άθροισμα των τυπικών αποκλίσεων από τη μέση τιμή του Z (Sum of squared deviations from the array mean), με το τύπο:

$$SDAM = \sum_{z \in Z} \left(z - \overline{z} \right)^2 \qquad (4.7)$$

   όπου $\overline{z}$ η μέση τιμή του Z.

4. Μετά από αυτό υπολογίζεται το άθροισμα των τυπικών αποκλίσεων από τη μέση τιμή των κλάσεων(Sum of squared deviations from the class means, με το τύπο:

$$SDCM = SDAM - SDBC \qquad (4.8)$$

5. Τέλος αφού ελεγχθούν όλα τα SBDC , μετακινείται μια τιμή από τη κλάση με το υψηλότερο SBDC στη κλάση με το χαμηλότερο SBDC.
6. Η διαδικασία επαναλαμβάνεται μέχρι να ελεγχθούν όλοι οι πιθανοί συνδυασμοί.

Τα διαστήματα των κλάσεων ορίζονται από τις μέγιστες και ελάχιστές τιμές αυτών. Ως μέτρο ελέγχου της διαδικασίας χρησιμοποιείται η ποιότητα της διασποράς των δεδομένων(goodness of variance fit) , η οποία δίνεται από το τύπο:

$$GVF = \frac{SDAM - SDCM}{SDAM} \quad ,(4.9)$$

όπου τιμές του GVF κοντά στο 1 υποδεικνύουν ένα καλό διαχωρισμό των τιμών και τιμές κοντά στο 0 το αντίθετο.

Μετά την εύρεση των διαστημάτων βέλτιστου διαχωρισμού σε λ κλάσεις επιλέγονται οι ταξινομητές που ανήκουν στις b_λ υψηλότερες κατηγορίες και η διαδικασία του σταδίου 4 επαναλαμβάνεται. Τέλος το b_λ αποτελεί ένα έμμεσο κατώφλι το οποίο εξασφαλίζει την υψηλή απόδοση των βασικών ταξινομητών, η επιλογή του οποίου επαφίεται στη κρίση του αναλυτή.

ΚΕΦΑΛΑΙΟ 5

5.1 Πειραματικό Στάδιο

Για την αξιολόγηση του προτεινόμενου συστήματος υποβοήθησης αποφάσεων σχεδιάστηκε και υλοποιήθηκε μια σειρά πειραμάτων. Ο σχεδιασμός των πειραμάτων βασίστηκε στη παραλλαγή των διαφόρων σταδίων της προτεινόμενης μεθοδολογίας έτσι ώστε να γίνει ευκολότερα αντιληπτό το κατά πόσο κάθε βήμα της επηρεάζει την απόδοση της παραγόμενης ensemble. Παρακάτω θα παρουσιαστούν τα πειράματα αυτά:

1. Cross-validation committee με Weighted Majority Voting(Soft) για το συνδυασμό των ταξινομητών και με βάρη τις κανονικοποιημένες ολικές αξίες των ταξινομητών, οι οποίες είναι βασισμένες στο συντελεστή gini, χωρίς το βήμα του κλαδέματος της ensemble.
2. Πείραμα 1 μαζί με το βήμα του κλαδέματος της ensemble (MuCE).
3. Cross-validation committee με Weighted Majority Voting(Soft) για το συνδυασμό των ταξινομητών και με βάρη τους κανονικοποιημένους συντελεστές gini των παραγόμενων βασικών ταξινομητών, χωρίς το βήμα του κλαδέματος.
4. Πείραμα 3 μαζί με το βήμα του κλαδέματος της ensemble.
5. Cross-validation committee με Weighted Majority Voting (Soft) για το συνδυασμό των ταξινομητών και με ίσα βάρη, τα οποία ισούνται με $\frac{1}{N}$. Όπου N ο αριθμός των παραγόμενων βασικών ταξινομητών.

Στα πειράματα 1, 2 όπου γίνεται χρήση του αλγόριθμου UTASTAR για το προσδιορισμό του προτιμησιακού μοντέλου, χρησιμοποιήθηκε ως προδιάταξη ο συντελεστής gini καθώς η απόκτηση της απαιτούμενης προδιάταξης από ένα πραγματικό αναλυτή ήταν αδύνατη. Ο υπολογισμός του συντελεστή gini για τα σύνολα δεδομένων τα οποία αναφέρονται σε multi-class classification προβλήματα υπολογίστηκε έμμεσα με το υπολογισμό του AUCROC μέσω των σχέσεων (1.23), (1.24) και με τη χρησιμοποίηση της σχέσης (1.28). Τη θέση του συντελεστή gini θα μπορούσε να πάρει ένα οποιοδήποτε μέτρο απόδοσης ή ακόμα και οποιαδήποτε πιθανή προδιάταξη την οποία θα έδινε κάποιο είδος τεχνητής νοημοσύνης. Τα μέτρα απόδοσης τα οποία χρησιμοποιήθηκαν ως κριτήρια κατά τη εφαρμογή του αλγόριθμου UTASTAR παρουσιάζονται παρακάτω:

- Precision
- Recall
- F1 Score
- Mathews Correlation Coefficient(MCC)
- Hamming Loss
- Cohen's Cappa
- Accuracy

Η παράμετρος k στο stratified k-fold validation που περιέχεται στις Cross-validation committees τέθηκε ίση με 5. Για k ίσο με 4 ικανοποιείται οριακά η συνθήκη (4.1), επομένως για περαιτέρω ασφάλεια επιλέγεται ο αμέσως επόμενος ακέραιος. Η ίδια τιμή δόθηκε και στην παράμετρο λ στα πειράματα που περιέχουν και το βήμα του κλαδέματος της ensemble. Επιπλέον στα πειράματα στα οποία χρησιμοποιήθηκε ο αλγόριθμος UTASTAR, οι παράμετροι a_j ορίστηκαν ίσοι με 5. Επιπρόσθετα στα πειράματα που περιέχουν το βήμα του κλαδέματος το κατώφλι b_λ τέθηκε αρχικά ίσο με 2 έτσι ώστε να αποφευχθεί το ενδεχόμενο η τελική ensemble να περιέχει μόνο ένα βασικό

ταξινομητή και στην συνέχεια δοκιμάστηκαν όλες οι τιμές που μπορεί να πάρει το b_λ στο $[0, \lambda]$.

Για τον έλεγχο των μεθόδων που προέκυψαν από τα παραπάνω πειράματα χρησιμοποιήθηκαν 5 σύνολα δεδομένων από το πλέον διαδεδομένο UC Irvine Repository [42](μια αποθήκη συνόλων δεδομένων για μηχανική μάθηση). Τα σύνολα δεδομένων που χρησιμοποιήθηκαν είναι τα εξής:

- Chess (King-Rook vs. King) Data Set
- Waveform Database Generator (Version 1) Data Set
- Statlog (Landsat Satellite) Data Set
- Breast Cancer Wisconsin (Original) Data Set
- Pen-Based Recognition of Handwritten Digits Data Set

Εκτός των παραπάνω συνόλων δεδομένων χρησιμοποιήθηκαν και τα:

- Abalone Data Set
- Adult Data Set
- Mushroom Data Set
- Iris Data Set
- Nursery Data Set
- Statlog (Shuttle) Data Set

Το δεύτερο σετ συνόλων δεδομένων δε χρησιμοποιήθηκε τελικά, επειδή κατά την εφαρμογή των προαναφερθέντων πειραμάτων, υπήρξαν διάφορα προβλήματα λόγω της περιορισμένης υπολογιστικής ισχύς του συστήματος στο οποίο έγιναν πάνω τα πειράματα καθώς και ενός προβλήματος στη βιβλιοθήκη για μηχανική μάθηση, που χρησιμοποιήθηκε κατά την υλοποίηση των πειραμάτων. Επίσης σε ορισμένα από τα σύνολα δεδομένων του δεύτερου σετ υπήρξε το πρόβλημα ότι οι βασικοί ταξινομητές που εκπαιδεύτηκαν σε αυτά, είχαν επιτύχει ήδη την μέγιστη απόδοση που μπορούν να επιτύχουν και για αυτό το λόγο θεωρήθηκαν μη κατάλληλα για τα συγκεκριμένα πειράματα. Τα σύνολα του πρώτου σετ παρουσιάζονται με λεπτομέρεια στο Παράρτημα Α.

Πριν τη διεξαγωγή των πειραμάτων κρίθηκε αναγκαία η προ-επεξεργασία των συνόλων δεδομένων έτσι ώστε να είναι δυνατή η εκτέλεση αυτών. Αρχικά αφαιρέθηκαν από κάθε σύνολο δεδομένων οι εγγραφές που είχαν ελλείπουσες τιμές σε ένα ή παραπάνω χαρακτηριστικά. Επιπλέον κωδικοποιήθηκαν όλες οι κατηγορικές μεταβλητές, είτε με απλή ακέραια αρίθμηση των κατηγοριών είτε με τη μέθοδο one-hot encoding. Το βήμα της επιλογής των καλύτερων χαρακτηριστικών δεν εκτελέστηκε έτσι ώστε να είναι δυνατή η απλοποίηση των πειραμάτων και της υλοποίησης τους.

Η εύρεση των βέλτιστων παραμέτρων κατά τη διαδικασία εκπαίδευσης των βασικών ταξινομητών έγινε με τη χρήση της μεθόδου k-fold Cross-validation με τη παράμετρο k να ισούται με 3. Το μέτρο που χρησιμοποιήθηκε για τον έλεγχο των βασικών ταξινομητών στο στάδιο της εύρεσης των βέλτιστων παραμέτρων ήταν αυτό της «ακρίβειας»(Accuracy).

Παρακάτω αναφέρονται οι αλγόριθμοι που χρησιμοποιήθηκαν για την εκπαίδευση των βασικών ταξινομητών:

- Logistic Regression
- k-Nearest Neighbors (με χρήση Ball Trees για την αναζήτηση των κοντινότερων γειτόνων)
- Gaussian Naïve Bayes
- Decision Trees(CART)

- Support Vector Machine(με χρήση γραμμικών. πολυωνυμικών και rbf πυρήνων)
- Artificial Neural Networks(MLP)

Τέλος η υλοποίηση των πειραμάτων έγινε με τη χρήση της γλώσσας προγραμματισμού Python σε περιβάλλον Jupyter Notebook[43] και με τη χρήση των βιβλιοθηκών «scikit-learn»[44] και «mlxtend»[45]. Ο αλγόριθμός UTASTAR, ο οποίος χρησιμοποιήθηκε κατά τη διεξαγωγή των πειραμάτων υλοποιήθηκε σε Python από το διδακτορικό φοιτητή Αλκαίο Σακελλάρη.

5.2 Αποτελέσματα

Σε αυτό το κεφάλαιο θα παρουσιαστούν τα αποτελέσματα των πειραμάτων που παρουσιάστηκαν παραπάνω. Αρχικά θα παρουσιαστούν κάποιοι συγκεντρωτικοί πίνακες με τις αποδόσεις των παραγόμενων ensemble που προκύπτουν από τα προαναφερθέντα πειράματα. Στη συνέχεια θα παρουσιαστούν τα αποτελέσματα τα οποία είναι σχετικά με την εφαρμογή του αλγόριθμου UTASTAR και τέλος θα παρουσιαστούν οι κατανομές των βαρών για κάθε ξεχωριστό πείραμα ή υπό-πείραμα.

Για την μέτρηση της απόδοσης των παραγόμενων ensemble χρησιμοποιείται ο κανονικοποιημένος συντελεστής GINI, ώστε να μπορεί να γίνει άμεση σύγκριση με τους συντελεστές GINI των βασικών ταξινομητών, οι οποίοι και χρησιμοποιούνται για το καθορισμό της αρχικής προδιάταξης R .

Πίνακας 4.1.7.1-1
Κανονικοποιημένος συντελεστής GINI ανά πείραμα και σύνολο δεδομένων

	Πείραμα/Μέθοδος				
Σύνολο Δεδομένων	**Πείραμα 1**	**Πείραμα 2 (MuCE, $b_\lambda = 2$)**	**Πείραμα 3**	**Πείραμα 4 ($b_\lambda = 2$)**	**Πείραμα 5**
Breast Cancer	**0.985388**	0.981839	0.971402	0.975994	0.9707755
Chess (King-Rook vs. King)	0.98001	**0.998595**	0.975766	0.998127	0.798224
Pen-Based Recognition of Handwritten Digits	**0.998024**	0.973966	0.997651	0.972429	0.95417
Statlog (Landsat Satellite)	**0.97632**	0.976241	0.971891	0.976313	0.971665
Waveform Database Generator	**0.954057**	0.952013	0.951714	0.95348	0.951313

Στο πίνακα 5.2-1 παρατηρούμε ότι τη καλύτερη απόδοση έχει η μέθοδος που προκύπτει από το πείραμα 1 σε όλα τα σύνολα δεδομένων, εκτός του Chess(King-Rook vs. King). Στο Chess(King-Rook vs. King) την καλύτερη απόδοση πετυχαίνει η μέθοδος του πειράματος 2, η οποία είναι και η προτεινόμενη μέθοδος MuCe για b_λ ίσο με 2. Επομένως μπορούμε να συμπεράνουμε ότι οι μέθοδοι που χρησιμοποιούν πολυκριτήρια ανάλυση αποφάσεων για το καθορισμό των βαρών των βασικών ταξινομητών επιτυγχάνουν καλύτερες αποδόσεις σε σχέση με τις μεθόδους που δεν χρησιμοποιούν.

Πίνακας 5.2-2
Κανονικοποιημένος συντελεστής GINI της μεθόδου MuCE(Πείραμα 2) $\forall b_\lambda \in [0,5]$.

Κατώφλι b_λ / Σύνολο Δεδομένων	0	1	2	3	4	5
Breast Cancer	0.985388	0.984135	0.981839	**0.986223**	0.985388	0.985388
Chess (King-Rook vs. King)	0.980096	0.761148	0.981947	**0.983737**	0.981739	0.981581
Pen-Based Recognition of Handwritten Digits	0.998023	0.998456	**0.998595**	0.9984	0.998109	0.998085
Statlog (Landsat Satellite)	0.97632	0.962435	0.976241	**0.976811**	0.976801	0.976414
Waveform Database Generator	0.954057	0.950622	0.952013	0.953955	**0.954318**	0.953985

Στο πίνακα 5.2-2 παρατηρούμε ότι η μέθοδος MuCe επιτυγχάνει τη καλύτερη δυνατή απόδοση για τιμές του b_λ που κυμαίνονται στο $[2,4]$. Αν εξαιρεθεί το 3ο και το 5ο σύνολο δεδομένων παρατηρούμε ότι, σε όλα τα υπόλοιπα σύνολα δεδομένων καλύτερη απόδοση πετυχαίνει η MuCE με b_λ ίσο με 3.

Πίνακας 5.2-3
Κανονικοποιημένος συντελεστής GINI της μεθόδου 4(Πείραμα 4) $\forall b_\lambda \in [0,5]$.

Κατώφλι b_λ / Σύνολο Δεδομένων	0	1	2	3	4	5
Breast Cancer	0.971402	**0.984135**	0.975994	0.968271	0.966392	0.97328
Chess (King-Rook vs. King)	0.975766	0.947592	0.963355	0.966537	0.968163	**0.978104**
Pen-Based Recognition of Handwritten Digits	0.997651	**0.99858**	0.998127	0.997775	0.997507	0.997608
Statlog (Landsat Satellite)	0.971891	0.972807	**0.976313**	0.97453	0.972114	0.97248
Waveform Database Generator	0.951714	0.952134	0.95348	**0.954098**	0.952546	0.952458

Στο πίνακα 5.2-3 φαίνεται ότι η μέθοδος 4 επιτυγχάνει τη καλύτερη απόδοση για τιμές του b_λ που κυμαίνονται στο $[1,5]$. Γενικά η συγκεκριμένη μέθοδος δεν παρουσιάζει κάποιο μοτίβο ως προς το πια τιμή του b_λ είναι η καταλληλότερη για να επιτευχθεί η μέγιστη απόδοση.

Σε σύγκριση με το πίνακα 5.2-3 παρατηρείται ότι η μέθοδος MuCE επιτυγχάνει καλύτερα αποτελέσματα, σε όλα τα εξεταζόμενα σύνολα δεδομένων, σε σχέση με τη μέθοδο

4, η οποία δεν χρησιμοποιεί τις ολικές αξίες που προκύπτον από τον αλγόριθμο UTASTAR, για το καθορισμό των βαρών των ταξινομητών. Πιο συγκεκριμένα η μέθοδος MuCE υπερείχε της μεθόδου 4 κατά **0.4%** στο σύνολο δεδομένων «Breast Cancer», **0.576%** στο «Chess (King-Rook vs. King)», **0.0015%** στο «Pen-Based Recognition of Handwritten Digits», **0.051%** στο «Statlog (Landsat Satellite)» και **0.023%** στο «Waveform Database Generator». Επιπρόσθετα παρατηρείτε στο πίνακα 4.1.7.1-1 ότι η μέθοδος MuCE, παράγει έως και **25.1%** αποδοτικότερες ensemble σε σχέση με τη περίπτωση του τυχαίου καθορισμού των βαρών των ταξινομητών.

Πίνακας 5.2-4
Κανονικοποιημένος συντελεστής GINI ανά βασικό (παραγόμενο) ταξινομητή και ανά σύνολο δεδομένων.

	Σύνολο Δεδομένων				
Ταξινομητής	Breast Cancer	Chess (King-Rook vs. King)	Pen-Based Recognition of Handwritten Digits	Statlog (Landsat Satellite)	Waveform Database Generator
Decision Tree FOLD0	0.872247	0.7458965	0.909343	0.89354	0.79509
Decision Tree FOLD1	0.872247	0.7523	0.872358	0.903286	0.810324
Decision Tree FOLD2	0.872247	0.761148	0.913002	0.88721	0.787323
Decision Tree FOLD3	0.872247	0.747357	0.913395	0.89721	0.793704
Decision Tree FOLD4	0.872247	0.75952	0.902069	0.889647	0.793054
GaussianNB FOLD0	0.95011	0.623306	0.930129	0.91012	0.920436
GaussianNB FOLD1	0.95752	0.630519	0.927357	0.909627	0.91905
GaussianNB FOLD2	0.963574	0.626364	0.927466	0.908868	0.922478
GaussianNB FOLD3	0.960756	0.611804	0.930204	0.909641	0.921769
GaussianNB FOLD4	0.963469	0.612068	0.928824	0.909196	0.921393
KNN FOLD0	0.918171	0.908302	0.973173	0.960871	0.948557
KNN FOLD1	0.94249	0.904605	0.971798	0.962435	0.948846
KNN FOLD2	0.953136	0.910914	0.966807	0.96183	0.949987
KNN FOLD3	0.945726	0.907694	0.975009	0.962585	0.94793
KNN FOLD4	0.941133	0.904103	0.97311	0.958779	0.947356
LR FOLD0	0.891869	0.795793	0.979066	0.875672	0.949986
LR FOLD1	0.959712	0.797973	0.977746	0.878289	0.949683
LR FOLD2	0.969732	0.794949	0.978821	0.879784	0.950622
LR FOLD3	0.975785	0.797776	0.980022	0.877279	0.949837
LR FOLD4	0.947396	0.796264	0.978942	0.881047	0.949711
MLP FOLD0	0.799186	0.939209	0.995672	0.935611	0.951707
MLP FOLD1	0.880388	0.945113	0.996351	0.938444	0.950786
MLP FOLD2	0.889782	0.947335	0.995155	0.933826	0.951856
MLP FOLD3	0.927147	**0.947592**	0.995754	0.935762	**0.952134**
MLP FOLD4	0.868907	0.946875	0.996434	0.934727	0.946107
SVM FOLD0	0.959294	0.928396	0.9985	**0.972807**	0.95046
SVM FOLD1	0.976412	0.928534	**0.99858**	0.972472	0.949646

SVM FOLD2	0.962217	0.929524	0.998144	0.971945	0.952093
SVM FOLD3	**0.984135**	0.931081	0.998297	0.97187	0.948092
SVM FOLD4	0.959294	0.930279	0.998456	0.972737	0.948862

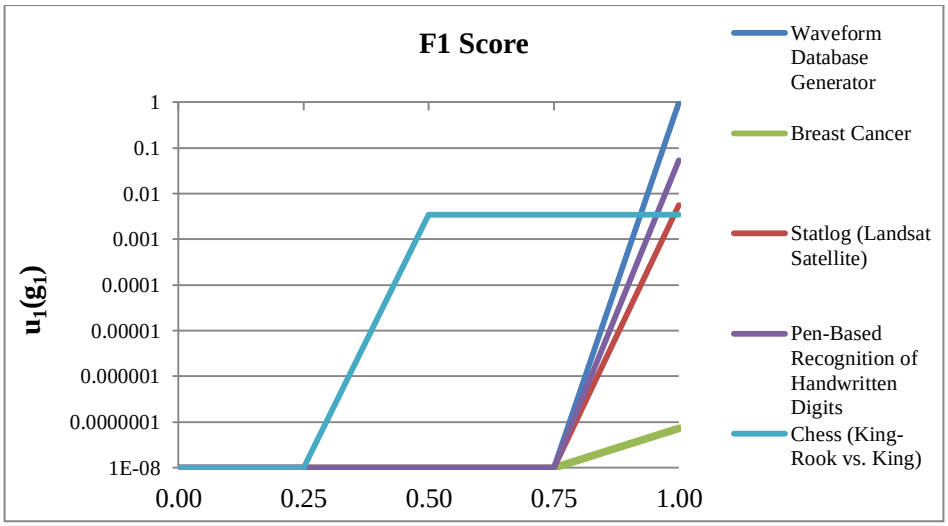
Στο πίνακα 5.2-4 παρουσιάζονται οι αποδόσεις των βασικών (παραγόμενων) ταξινομητών. Σε αυτόν παρατηρούμε ότι τη καλύτερη απόδοση συγκέντρωσαν ταξινομητές που είναι βασισμένοι στα SVM και στα MLP. Συγκρίνοντας με το πίνακα 5.2-1 και το πίνακα 5.2-2 εύκολα μπορεί κανείς να αντιληφθεί ότι η μέθοδος MuCE παράγει σε όλες τις περιπτώσεις, ensemble με μεγαλύτερη απόδοση από ότι επιτυγχάνουν οι βασικοί ταξινομητές.

Πίνακας 5.2-5
Κανονικοποιημένος συντελεστής GINI ανά βασικό (παραγόμενο) ταξινομητή και ανά σύνολο δεδομένων (ολόκληρο το υποσύνολο δεδομένων εκπαίδευσης).

	Σύνολο Δεδομένων				
Ταξινομητής	Breast Cancer	Chess (King-Rook vs. King)	Pen-Based Recognition of Handwritten Digits	Statlog (Landsat Satellite)	Waveform Database Generator
Decision Tree	0.872247	0.776943	0.899705	0.891559	0.819202
GaussianNB	0.958668	0.633815	0.928895	0.930495	0.921461
KNN	0.940090	0.906936	0.973363	0.964465	0.947519
LR	0.956372	0.799183	0.979198	**0.973008**	0.950671
MLP	0.923808	**0.948571**	0.995635	0.879265	**0.951210**
SVM	**0.963261**	0.939356	**0.998376**	0.909543	0.951100

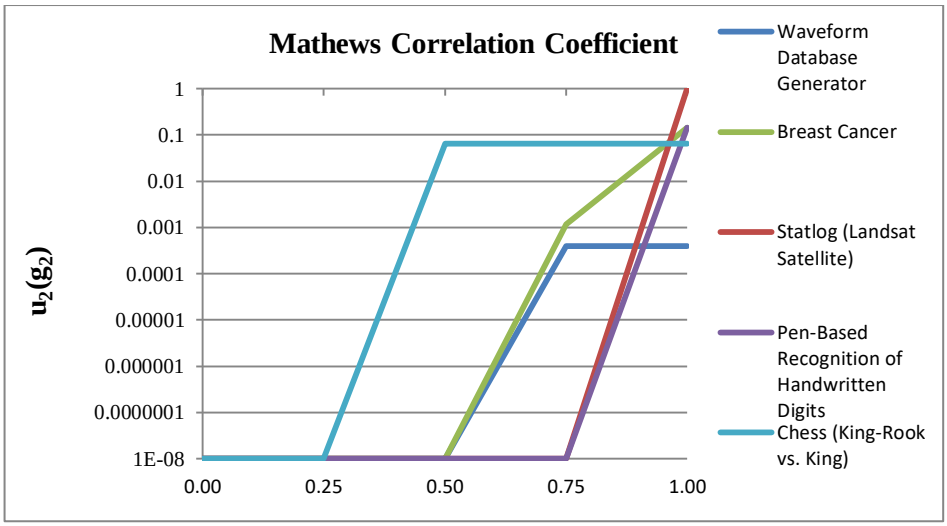
Ο παραπάνω πίνακας δείχνει τις αποδόσεις των βασικών ταξινομητών αν αυτοί εκπαιδεύονταν σε ολόκληρο το σύνολο δεδομένων (εκπαίδευσης) κάθε φορά αντίστοιχα. Παρατηρούμε ότι και σε αυτή τη περίπτωση τα SVM και τα MLP, παρουσιάζουν τις μεγαλύτερες αποδόσεις, με εξαίρεση το σύνολο δεδομένων «Statlog(Landsat Satellite)», στο οποίο την καλύτερη απόδοση πετυχαίνει το Logistic Regression. Από το πίνακα αυτό και σε σύγκριση με τους πίνακες 5.2-1, 5.2-2 μπορεί κανείς να συμπεράνει ότι ακόμα και στην περίπτωση που χρησιμοποιείται όλο το σύνολο δεδομένων (εκπαίδευσης) για την εκπαίδευση των βασικών ταξινομητών, η μέθοδος MuCE παράγει ensemble με μεγαλύτερες αποδόσεις.

Παρακάτω θα παρουσιαστούν τα γραφήματα των μερικών συναρτήσεων αξίας για κάθε εξεταζόμενο μέτρο απόδοσης (κριτήριο) καθώς και τα συνολικά βάρη αυτών, τα οποία προέκυψαν από την εκτέλεση των πειραμάτων 1, 2.



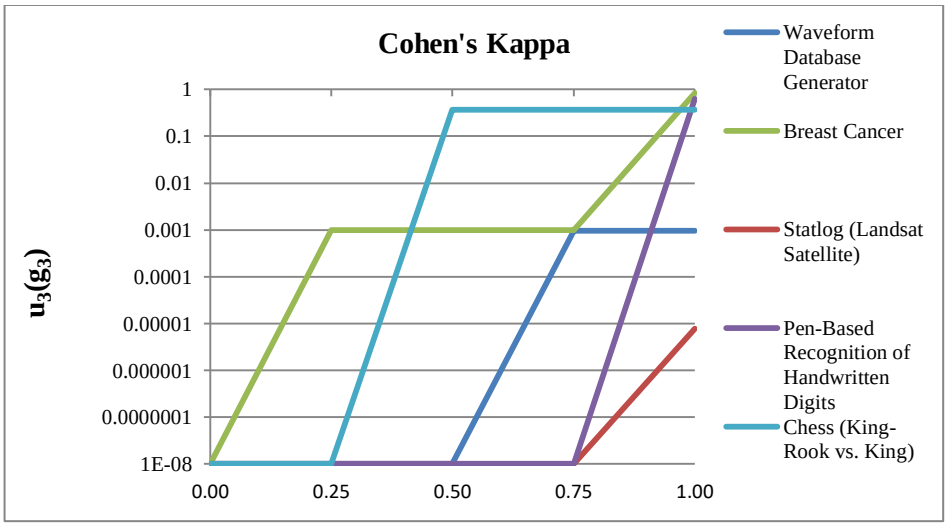
Γράφημα 5.2-1: Συναρτήσεις μερικών αξιών για το κριτήριο «F1 Score» ανά σύνολο δεδομένων.

Στο παραπάνω γράφημα παρατηρούμε ότι για το σύνολο δεδομένων «Chess (King-Rook vs. King)» η συνάρτηση μερικής αξίας είναι αύξουσα μεταξύ των τιμών 0.25 και 0.5 και σε όλο το υπόλοιπο διαθέσιμο διάστημα παραμένει σταθερή. Δηλαδή ο συντελεστής GINI προτιμάει ταξινομητές, οι οποίοι επιτυγχάνουν F1 Score, τα οποία βρίσκονται κοντά στο δεξί άκρο του $(0.25, 0.5)$ και παρουσιάζει αδιαφορία μεταξύ δύο ταξινομητών που ανήκουν στο $[0, 0.25]$ και $[0.5,1]$ αντίστοιχα. Βέβαια είναι προφανής η προτίμηση σε ταξινομητές που ανήκουν στο $[0.5,1]$ από άλλους που ανήκουν στο $[0, 0.25]$. Αντίθετα οι μερικές συναρτήσεις για το F1 Score στα υπόλοιπα σύνολα δεδομένων είναι σταθερές και ίσες με 0 μέχρι και τη τιμή του κριτηρίου 0.75 και στη συνέχεια αυξάνονται γραμμικά. Είναι προφανές δηλαδή ότι ο συντελεστής GINI σε αυτά τα σύνολα δεδομένων παρουσιάζει αδιαφορία μεταξύ δύο ταξινομητών που ανήκουν στο $[0, 0.75]$ και ότι προτιμάει ταξινομητές οι οποίοι βρίσκονται κοντά στο δεξί άκρο του $(0.75,1]$. Παρότι για όλα τα υπόλοιπα σύνολα δεδομένων οι μερικές συναρτήσεις είναι γραμμικά αύξουσες μετά τη τιμή 0.75, παρατηρείται ότι η κάθε μια μερική συνάρτηση διαφέρει όσον αφορά το ρυθμό που αυτή αυξάνεται. Η μερική συνάρτηση η οποία παίρνει τη μεγαλύτερη μέγιστη τιμή από όλες τις άλλες είναι αυτή του «Waveform Database Generator». Μπορεί δηλαδή κάποιος μέσα από τον υπολογισμό των μερικών συναρτήσεων να συμπεράνει όχι μόνο για τη σημαντικότητα ενός μέτρου απόδοσης αλλά και την αξία που αποδίδει ο αναλυτής για κάθε τιμή αυτού, του οποίου το ρόλο σε αυτή τη περίπτωση παίζει ο συντελεστής GINI. Με αντίστοιχη λογική μπορούν να ερμηνευτούν και τα γραφήματα των μερικών συναρτήσεων αξίας των υπολοίπων μέτρων απόδοσης, τα οποία παρουσιάζονται παρακάτω.



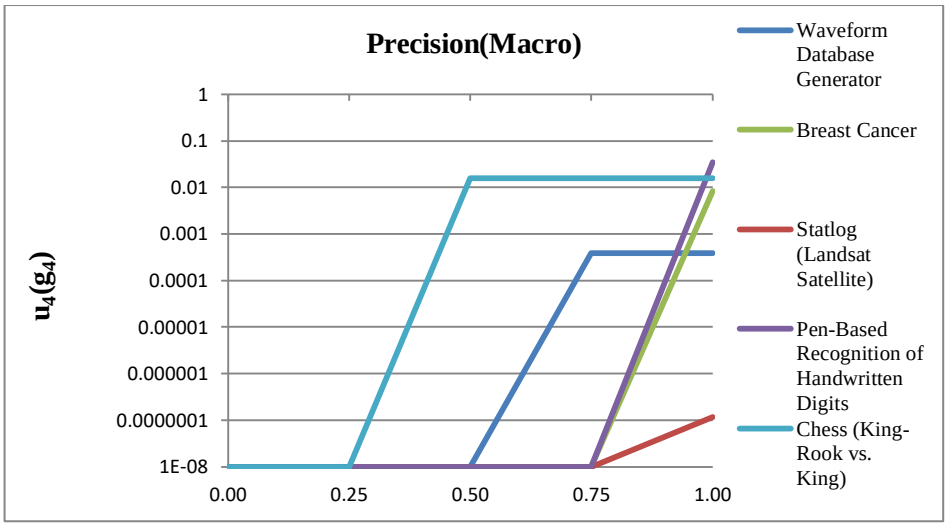
Γράφημα 5.2-2: Συναρτήσεις μερικών αξιών για το κριτήριο «Mathews Correlation Coefficient» ανά σύνολο δεδομένων.

Στο γράφημα 5.2-2 αξιοσημείωτη είναι η συμπεριφορά της μερικής συνάρτησης αξίας για το «Breast Cancer», η οποία σε αντίθεση με αυτές που έχουν παρουσιαστεί παραπάνω συνεχίζει να αυξάνεται μετά τη τιμή 0.75. Βλέπουμε ότι ενώ από τη τιμή 0.5 ο συντελεστής GINI ξεκινάει να δίνει αξία στο μέτρο απόδοσης MCC, ο ρυθμός με τον οποίο η αξία αυτή αυξάνεται, μειώνεται μετά τη τιμή 0.75.

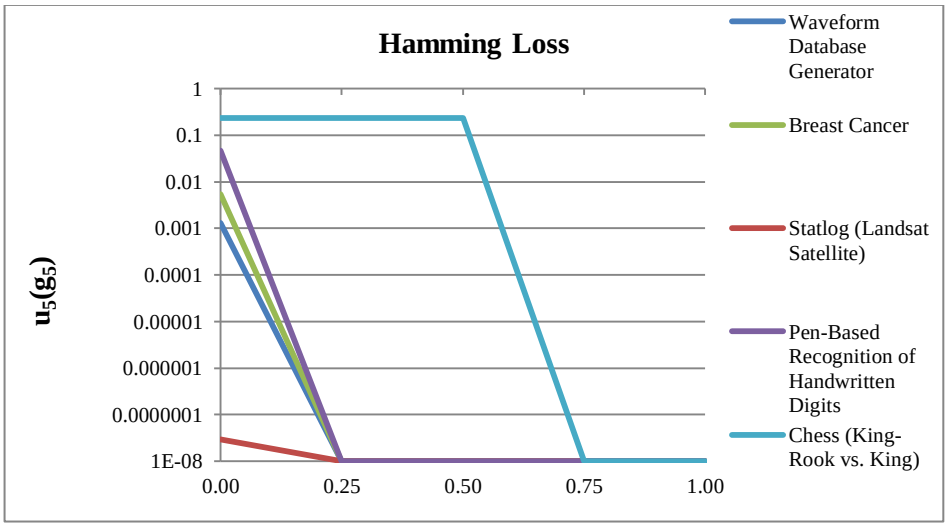


Γράφημα 5.2-3: Συναρτήσεις μερικών αξιών για το κριτήριο «Cohen's Kappa» ανά σύνολο δεδομένων.

Για το Cohen's Kappa και το σύνολο δεδομένων «Breast Cancer» βλέπουμε ότι ξεκινάει να παίρνει αξία από τη τιμή 0 κιόλας μέχρι και τη τιμή 0.25. Στη συνέχεια παραμένει σταθερή μέχρι και τη τιμή 0.75 και έπειτα αυξάνεται μέχρι να πάρει τη μέγιστη τιμή της. Δηλαδή ο συντελεστής GINI είναι αδιάφορος για ταξινομητές των οποίων η απόδοση στο μέτρο Cohen's Kappa είναι μέτρια.

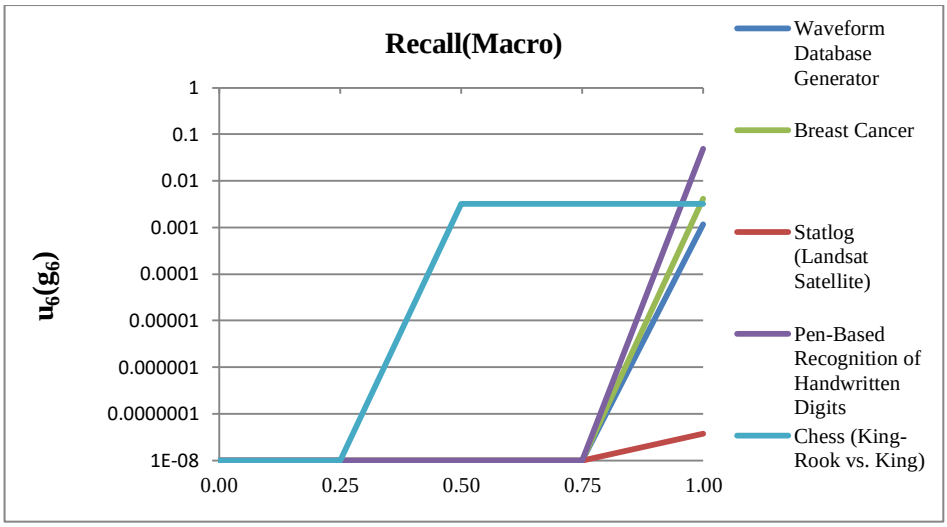


Γράφημα 5.2-4: Συναρτήσεις μερικών αξιών για το κριτήριο «Precision(Macro Average)» ανά σύνολο δεδομένων.

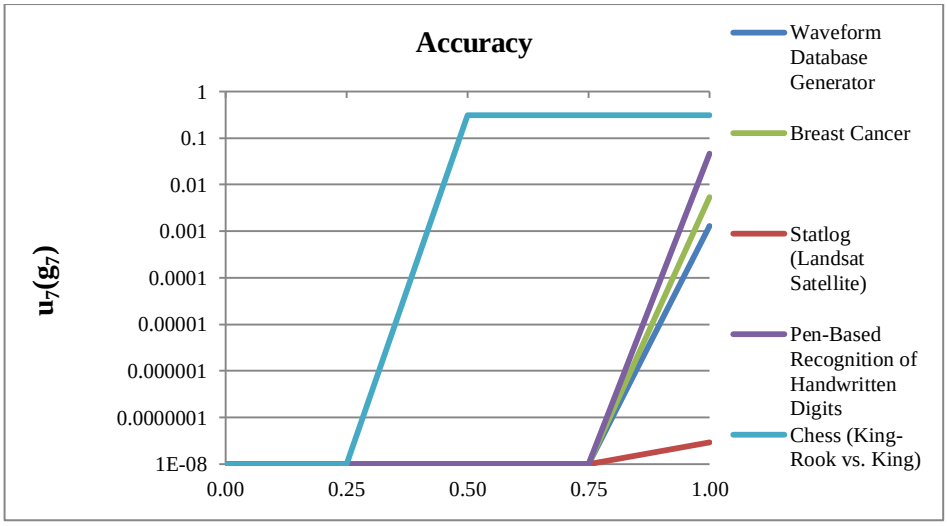


Γράφημα 5.2-5: Συναρτήσεις μερικών αξιών για το κριτήριο «Hamming Loss» ανά σύνολο δεδομένων.

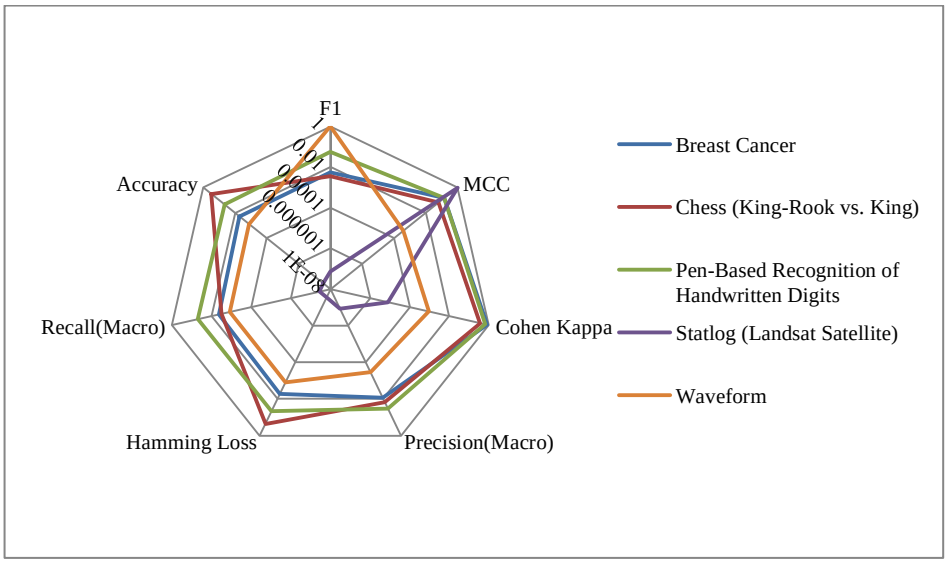
Στο γράφημα 4.1.7.1-5 παρατηρείται ότι όλες οι μερικές συναρτήσεις αξίας είναι φθίνουσες. Κάτι τέτοιο οφείλεται στο γεγονός ότι ένας ταξινομητής με χαμηλό μέτρο απόδοσης Hamming Loss είναι καλύτερο από έναν με υψηλό.



Γράφημα 5.2-6: Συναρτήσεις μερικών αξιών για το κριτήριο «Recall(Macro Average)» ανά σύνολο δεδομένων.



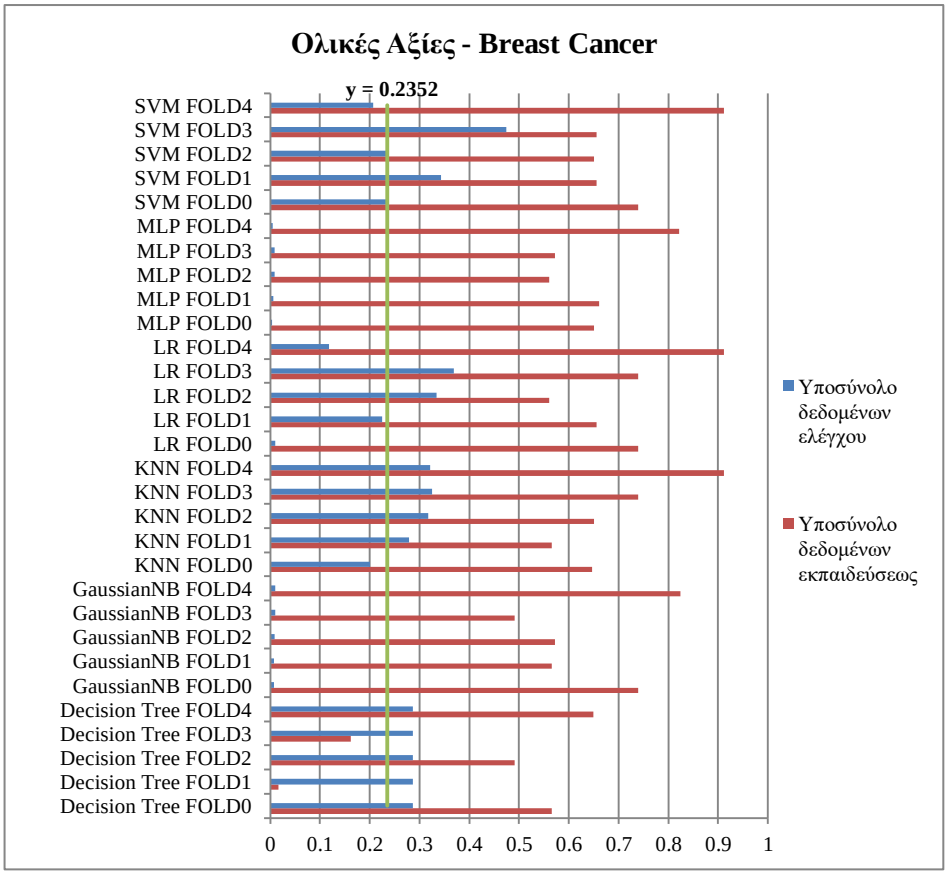
Γράφημα 5.2-7: Συναρτήσεις μερικών αξιών για το κριτήριο «Accuracy» ανά σύνολο δεδομένων.



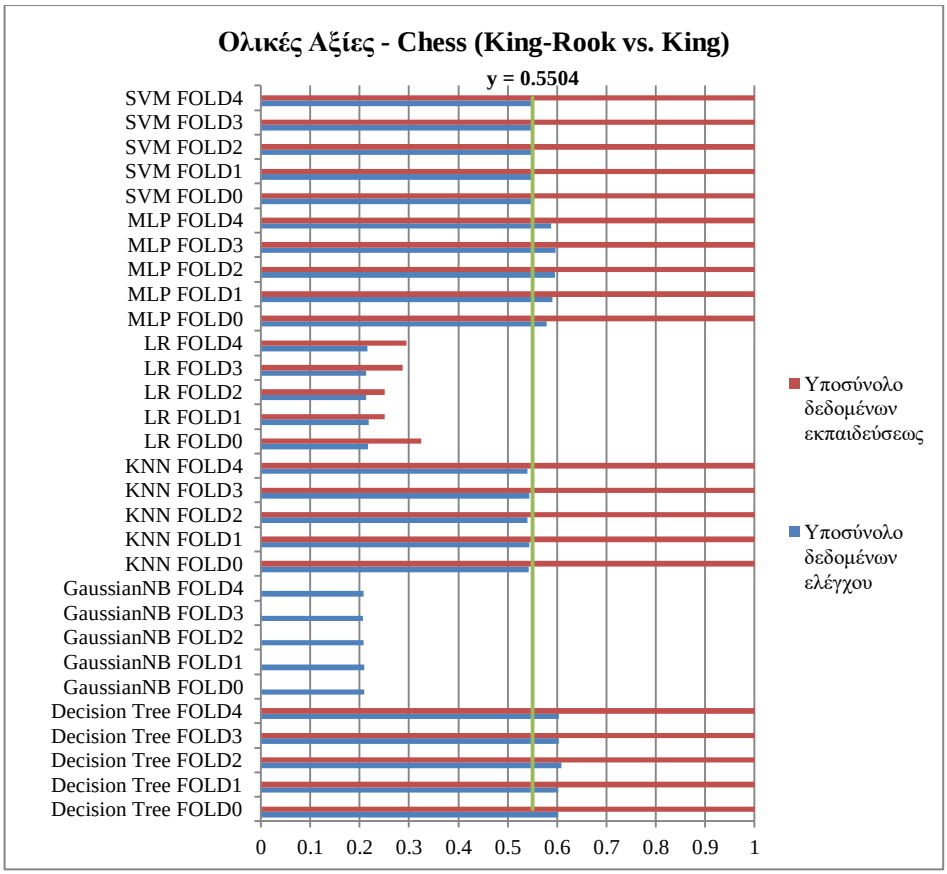
Γράφημα 5.2-8: Αναπαράσταση των συνολικών βαρών των κριτηρίων ανά σύνολο δεδομένων σε γράφημα ραντάρ.

Στο γράφημα 5.2-8 φαίνονται τα βάρη των κριτηρίων τα οποία προέκυψαν από την εκτέλεση του αλγόριθμου UTASTAR για κάθε εξεταζόμενο σύνολο δεδομένων. Πιο συγκεκριμένα για το σύνολο δεδομένων «Breast Cancer» τη μεγαλύτερη σημαντικότητα έχει το μέτρο απόδοσης «Cohen's Kappa» (82.8%) και ακολουθεί το «MCC» (14.2%), ενώ τα υπόλοιπα μέτρα πήραν περίπου ισότιμα βάρη. Παρόμοια συμπεριφορά παρατηρείται και στο «Pen-Based Recognition of Handwritten Digits», όπου το «Cohen's Kappa» πήρε τη τιμή 62.7% και το «MCC» 14.3%. Από την άλλη μεριά στο «Chess (King-Rook vs. King)» παρατηρείται μια πιο ομοιόμορφη κατανομή των βαρών, με το «Cohen's Kappa» να συγκεντρώνει τη μεγαλύτερη τιμή (36.4%), μαζί με το «Accuracy» (31.3%) καθώς και το «Hamming Loss» το οποίο πήρε μια εξίσου υψηλή τιμή (23.5%). Αξιοσημείωτο είναι το γεγονός ότι στο «MCC» για το σύνολο δεδομένων «Statlog (Landsat Satellite)», αποδίδεται βάρος ίσο με 99.9%. Κάτι τέτοιο, θα μπορούσε να διορθωθεί με την αφαίρεση αυτού του μέτρου απόδοσης από το σύνολο των εξεταζόμενων κριτηρίων. Τέλος παρόμοια συμπεριφορά παρατηρείται και για το «Waveform Database Generator», με το «F1 Score» να συγκεντρώνει 99.4%.

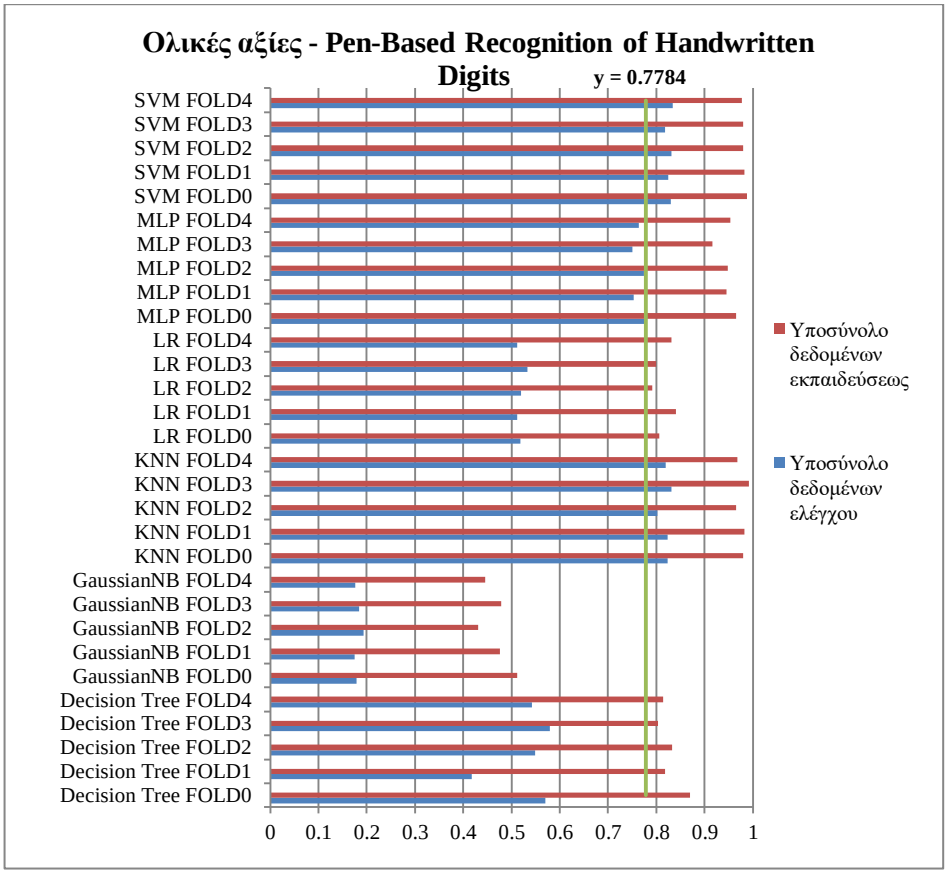
Παρακάτω θα παρουσιαστούν τα γραφήματα των ολικών αξιών των παραγόμενων ταξινομητών για κάθε σύνολο δεδομένων, καθώς και τα όρια που ορίζονται από το βέλτιστο b_λ . Όλοι οι ταξινομητές των οποίων οι ολικές αξίες στο υποσύνολο ελέγχου ξεπερνούν το όριο *y* συμμετέχουν στη τελική ensemble, η οποία δημιουργείται από τη μέθοδο MuCE.



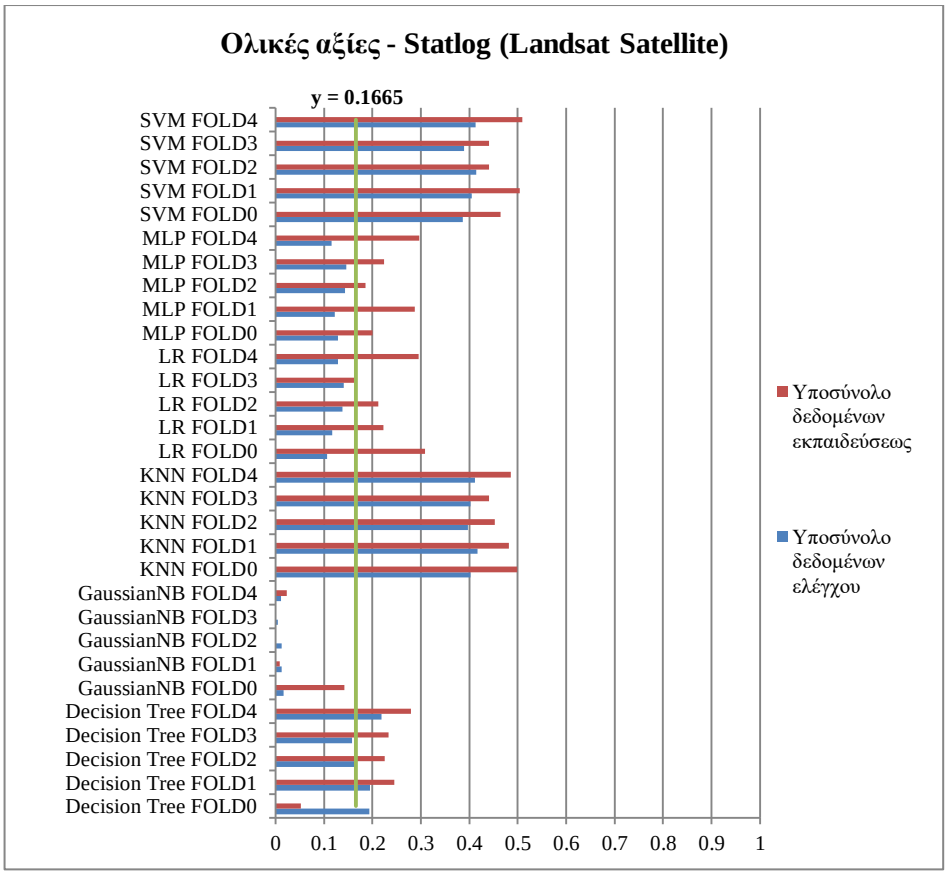
Γράφημα 5.2-9: Γράφημα ολικών αξιών για τα υποσύνολα εκπαίδευσης και ελέγχου του «Breast Cancer».



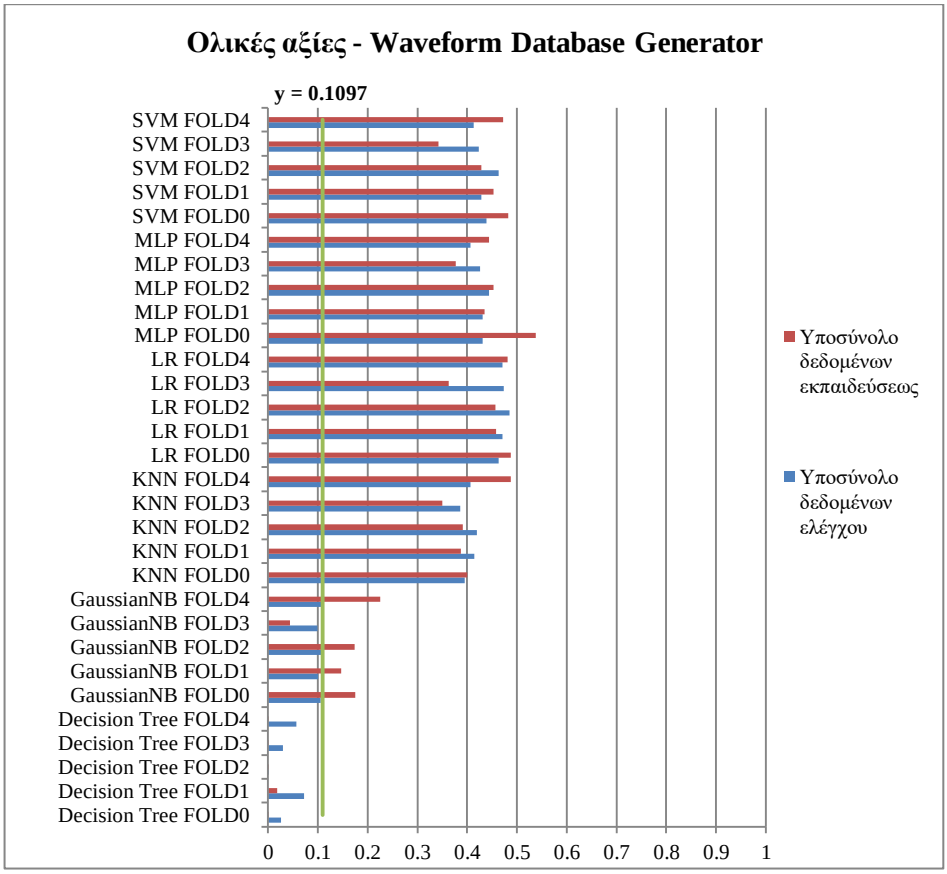
Γράφημα 5.2-10: Γράφημα ολικών αξιών για τα υποσύνολα εκπαίδευσης και ελέγχου του «Chess (King-Rook vs. King)».



Γράφημα 5.2-11: Γράφημα ολικών αξιών για τα υποσύνολα εκπαίδευσης και ελέγχου του «Pen-Based Recognition of Handwritten Digits».

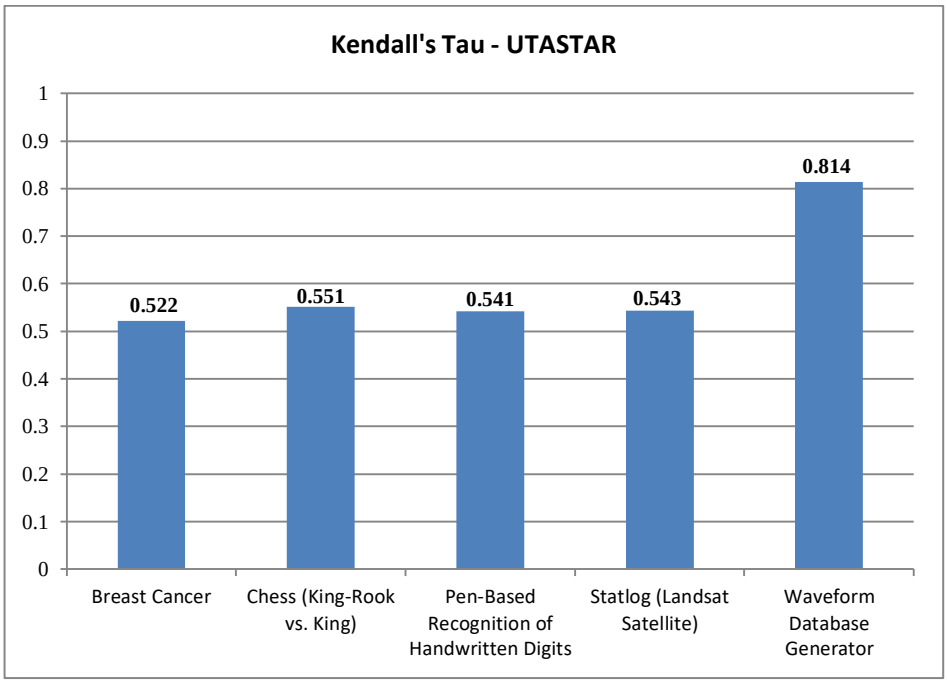


Γράφημα 5.2-12: Γράφημα ολικών αξιών για τα υποσύνολα εκπαίδευσης και ελέγχου του «Statlog (Landsat Satellite)».



Γράφημα 5.2-13: Γράφημα ολικών αξιών για τα υποσύνολα εκπαίδευσης και ελέγχου του «Waveform Database Generator».

Για τη μέτρηση της ακρίβειας του δημιουργούμενου προτιμησιακού μοντέλου χρησιμοποιήθηκε ως μέτρο το Kendall's Tau (συντελεστής συσχέτισης του Kendall). Οι συντελεστές συσχέτισης του Kendall για κάθε σύνολο δεδομένων παρουσιάζονται στο γράφημα .

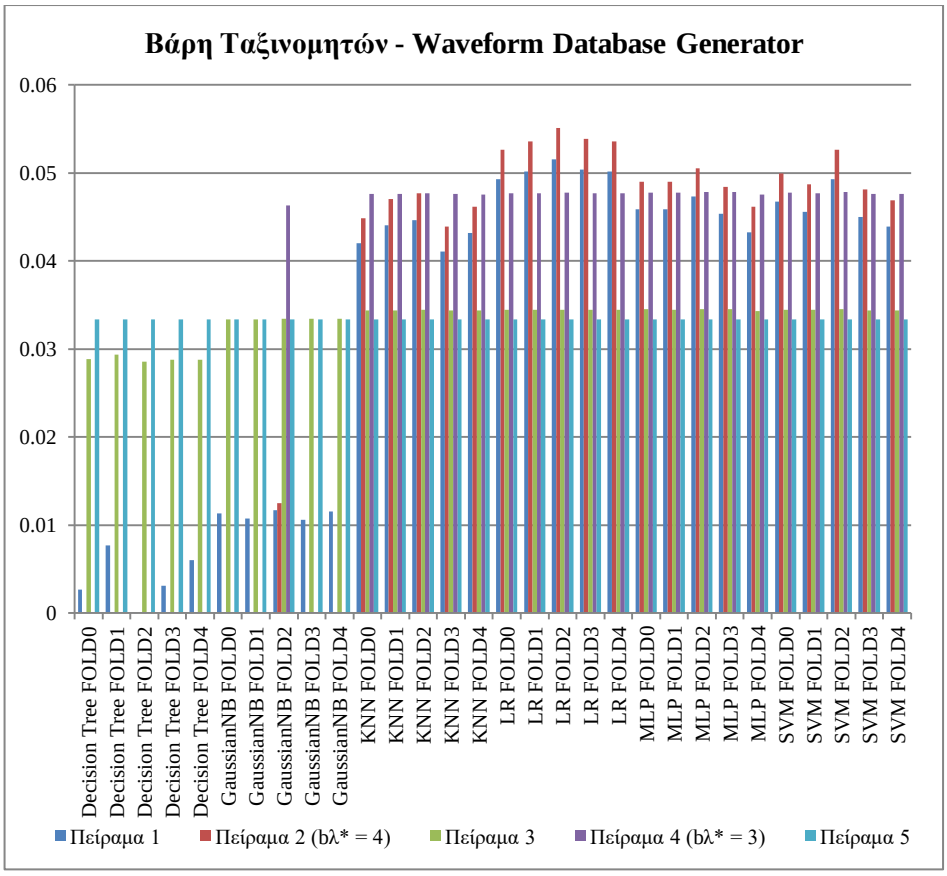


Γράφημα 5.2-14: Kendall's Tau των προκυπτουσών ολικών αξιών από τον αλγόριθμο UTASTAR, ανά σύνολο δεδομένων.

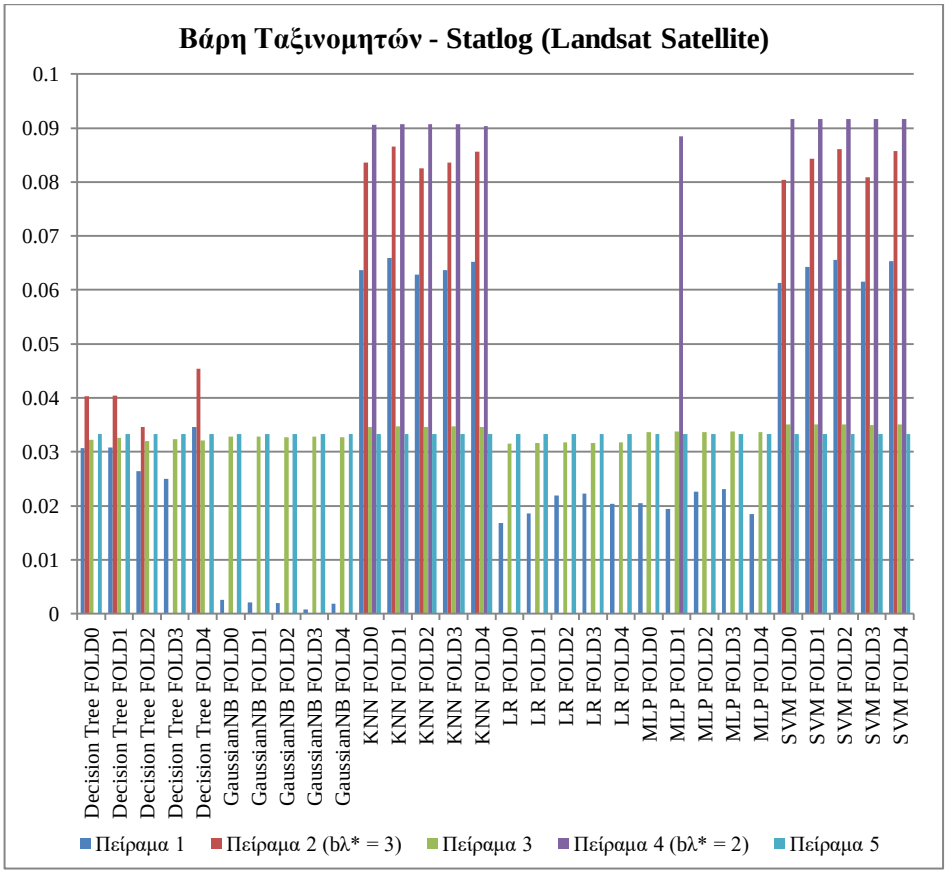
Στο γράφημα 5.2-14 παρατηρείται ότι για τη πλειοψηφία των συνόλων δεδομένων η εφαρμογή του αλγόριθμου UTASTAR παράγει προτιμησιακά μοντέλα, των οποίων ο συντελεστής συσχέτισης του Kendall κυμαίνεται στο $[0.52, 0.55]$. Αντίθετα εξαίρεση αποτελεί το προτιμησιακό μοντέλο το οποίο παράχθηκε για το σύνολο δεδομένων «Waveform Database Generator», το οποίο επιτυγχάνει Kendall's Tau ίσο με 81.44%. Από το παραπάνω συμπεραίνουμε ότι ο αλγόριθμος UTASTAR αποκομίζει περίπου το 50% της πληροφορίας που περιέχεται στο συντελεστή GINI των βασικών ταξινομητών. Παρόλα αυτά οι ensemble οι οποίες προκύπτουν από τη μέθοδο MuCE, η οποία χρησιμοποιεί τις ολικές αξίες που προκύπτουν από την εκτέλεση του αλγορίθμου UTASTAR, επιτυγχάνουν μεγαλύτερες αποδόσεις σε σχέση με τις υπόλοιπες εξεταζόμενες μεθόδους.

Επίσης επειδή το προκύπτων προτιμησιακό μοντέλο βασίζεται πάνω σε μέτρα απόδοσης και επειδή το Kendall's Tau δεν ισούται με 100% (ούτε και το πλησιάζει) , θα αποτελεί στην ουσία ένα νέο μέτρο απόδοσης. Το μέτρο αυτό είναι ένας σταθμισμένος μέσος των μέτρων απόδοσης που χρησιμοποιήθηκαν ως κριτήρια με τέτοιο τρόπο ώστε να συνάδει όσο πιο πολύ γίνεται με τις προτιμήσεις του αναλυτή.

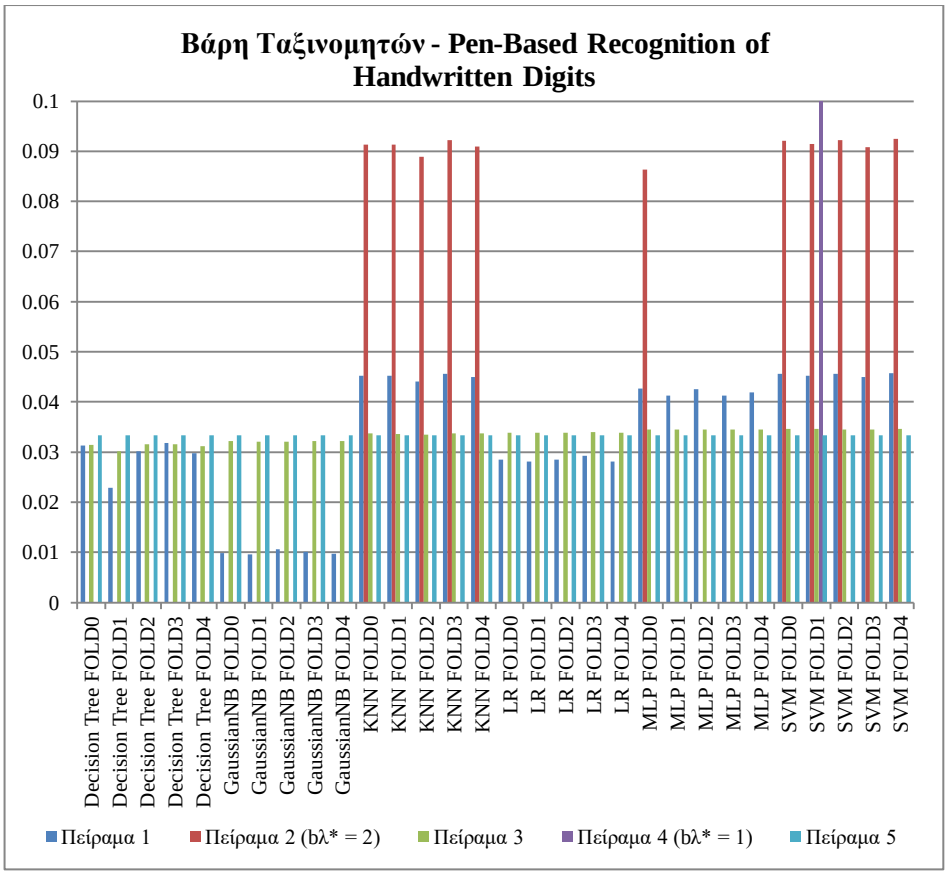
Στη συνέχεια θα παρουσιαστούν τα γραφήματα των κατανομών των βαρών των βασικών ταξινομητών, ανά πείραμα και σύνολο δεδομένων.



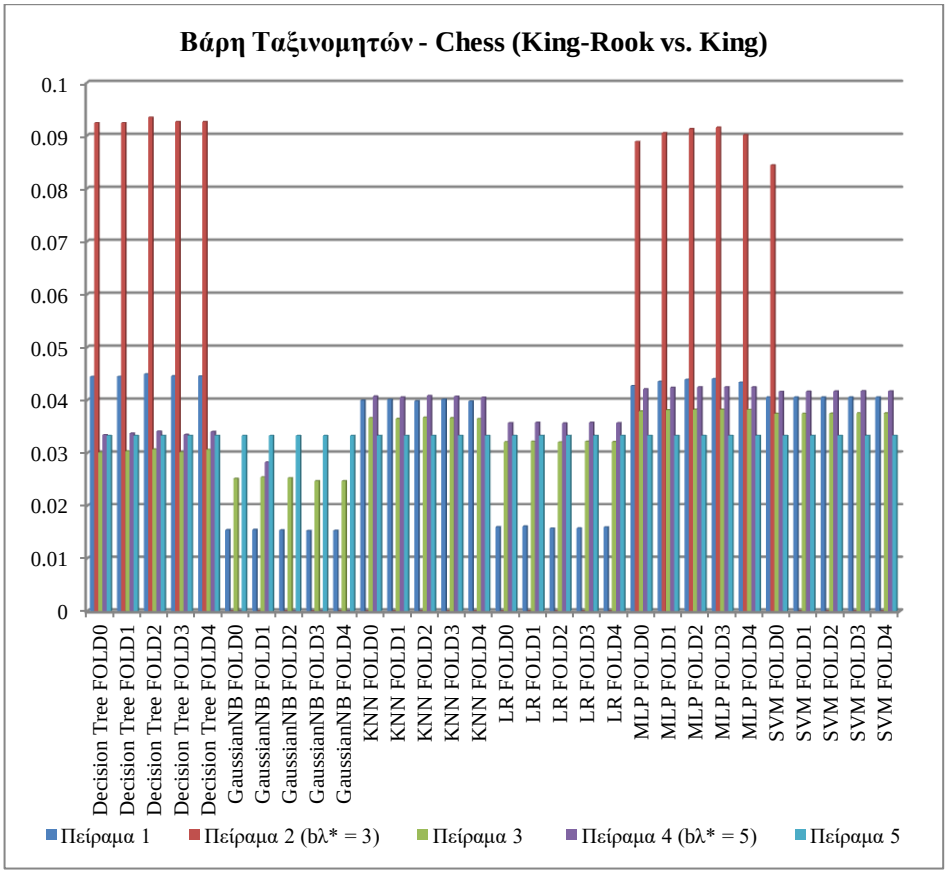
Γράφημα 5.2-15: Κατανομή των βαρών των βασικών ταξινομητών ανά πείραμα για το σύνολο δεδομένων «Waveform Database Generator».



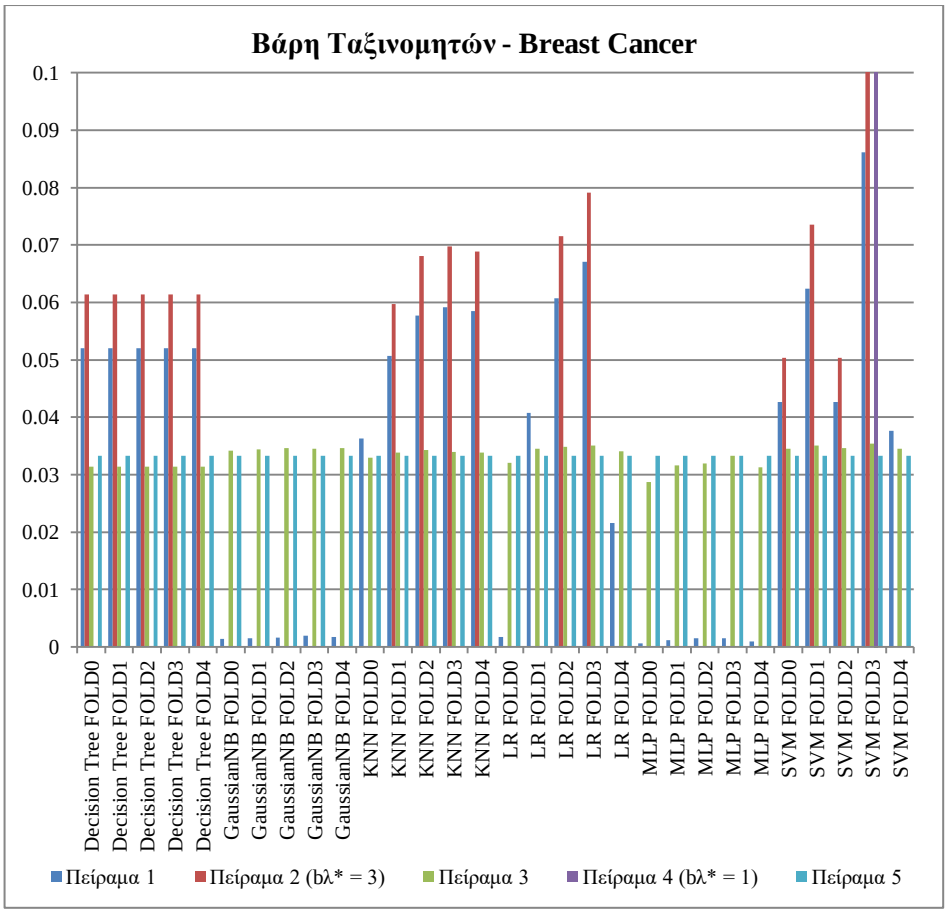
Γράφημα 5.2-16: Κατανομή των βαρών των βασικών ταξινομητών ανά πείραμα για το σύνολο δεδομένων «Statlog (Landsat Satellite)».



Γράφημα 5.2-17: Κατανομή των βαρών των βασικών ταξινομητών ανά πείραμα για το σύνολο δεδομένων «Pen-Based Recognition of Handwritten Digits».



Γράφημα 5.2-18: Κατανομή των βαρών των βασικών ταξινομητών ανά πείραμα για το σύνολο δεδομένων «Chess (King-Rook vs. King)».



Γράφημα 5.2-19: Κατανομή των βαρών των βασικών ταξινομητών ανά πείραμα για το σύνολο δεδομένων «Breast Cancer».

Παρατηρώντας τα γραφήματα των κατανομών των βαρών καθώς και τα γραφήματα των ολικών αξιών είναι φανερό ότι μέθοδος 3 αποτυγχάνει να ξεχωρίσει τα ασθενή μοντέλα από το σύνολο των μοντέλων. Κάτι τέτοιο συμβαίνει καθώς στις περισσότερες περιπτώσεις τα βάρη της μεθόδου 3 είναι πολύ κοντά σε αυτά της μεθόδου 5, η οποία κάνει την αφελή υπόθεση ότι όλα τα μοντέλα είναι ισάξια. Αντίθετα η μέθοδος MuCE (πείραμα 2) ακόμα και χωρίς το βήμα του κλαδέματος της ensemble (πείραμα 1) καταφέρνει να ανιχνεύσει τα ασθενή μοντέλα και να δώσει χαμηλά βάρη σε αυτά.

Επιπρόσθετα συγκρίνοντας τα βάρη της μεθόδου 2 και 4 καθώς και τις αντίστοιχες αποδόσεις , οι οποίες παρουσιάστηκαν παραπάνω, μπορούμε να συμπεράνουμε ότι το κλάδεμα της ensemble μέσω συσταδοποίησης ωφελεί περισσότερο τη μέθοδο 4. Κάτι τέτοιο είναι λογικό καθώς οι αποδόσεις της μεθόδου 3 είναι αρκετά χαμηλές σε σχέση με αυτές της μεθόδου 1. Όμως επειδή δυσκολεύεται να ανιχνεύσει αρχικά τα ασθενή μοντέλα και η ποιότητα της συσταδοποίησης είναι χαμηλή. Αποτέλεσμα του προαναφερθέντος είναι η τελική ensemble είτε να υπόκειται σε «επιθετικό» κλάδεμα είτε σε πολύ ασθενές κλάδεμα, κάτι το οποίο οδηγεί στη μείωση της πιθανής αύξησης της απόδοσης της ensemble. Από την άλλη μεριά μέσω της συσταδοποίησης η μέθοδος MuCE (πείραμα 2) καταφέρνει να αποκλείσει τα ασθενή μοντέλα, τα οποία εξαρχής είχε ανιχνεύσει πριν το βήμα του κλαδέματος. Με αυτό τον τρόπο βελτιστοποιείται η σύσταση της τελικής ensemble και επιτυγχάνει ακόμα υψηλότερες αποδόσεις.

5.3 Συμπεράσματα

Στη παρούσα εργασία μελετήθηκαν και υλοποιήθηκαν 5 διαφορετικοί τρόποι για τη δημιουργία μιας ensemble, της οποίας οι βασικοί ταξινομητές εκπαιδεύονται παράλληλα. Για να είναι πιο αντικειμενικά τα συμπεράσματα αυτής της μελέτης όλες οι μέθοδοι δοκιμάστηκαν σε 5 διαφορετικά σύνολα δεδομένων. Γενικά η βελτίωση που επέφερε ο συνδυασμός των βασικών ταξινομητών δεν ήταν αρκετά υψηλή. Παρόλα αυτά οι αποδόσεις που επιτεύχθηκαν (τουλάχιστον από τη προτεινόμενη μέθοδο MuCE) ήταν καλύτερες από αυτές των βασικών ταξινομητών. Μέσα από τα αποτελέσματα των πειραμάτων 1 και 2 δείχθηκε ότι η χρησιμοποίηση πολυκριτήριας ανάλυσης αποφάσεων για το καθορισμό των βαρών των βασικών ταξινομητών μιας ensemble αποτελεί μια αποδοτική τακτική για τη βελτίωση της συνολικής απόδοσης της ensemble. Ακόμα το βήμα της συσταδοποίησης το οποίο περιέχεται στις μεθόδους των πειραμάτων 2 και 4 ωφέλησε παραπάνω τη μέθοδο 4 και ο λόγος είναι οι χαμηλές αποδόσεις που ήδη είχε η παραγόμενη ensemble. Συνολικά όμως η μέθοδος MuCE παρήγαγε ensemble οι οποίες ήταν αποδοτικότερες σε σχέση με αυτές που παρήγαγε η μέθοδος 4. Επιπρόσθετα παρατηρήθηκε ότι μέσω της μορφής των μερικών συναρτήσεων αξίας μπορούν να βγουν συμπεράσματα για τον τρόπο με το οποίο επηρεάζονται η αποφάσεις του αναλυτή. Τέλος μέσα από τη διαδικασία εκπαίδευσης των βασικών ταξινομητών ο αναλυτής αποκτά πείρα η οποία και αποθηκεύεται με τη μορφή μερικών συναρτήσεων αξίας για μεταγενέστερη χρήση.

5.4 Περαιτέρω Έρευνα

Επιτακτική είναι ανάγκη για περαιτέρω μελέτη της προτεινόμενης μεθόδου MuCE, καθώς είναι πολλά τα κομμάτια της, των οποίων η παραμετροποίηση επηρεάζουν την απόδοση της τελικής παραγόμενης ensemble. Μερικά ενδεικτικά θέματα για μελλοντική έρευνα ακολουθούν:

- Εύρεση τρόπων απόκτησης της απαιτούμενης πληροφορίας για το καθορισμό της αρχικής προδιάταξης στο βήμα της εκτέλεσης του αλγορίθμου UTASTAR , της μεθόδου MuCE.
- Μελέτη για το πώς επηρεάζει η διασπορά των ολικών αξιών των ταξινομητών τη ποιότητα του βήματος της συσταδοποίησης της μεθόδου MuCE, και πως αυτή επηρεάζει την απόδοση της τελικής παραγόμενης ensemble.
- Μελέτη για το πώς επηρεάζει η επιλογή των λ και b_λ την απόδοση της τελικής παραγόμενης ensemble.
- Εφαρμογή της μεθόδου σε συγκεκριμένα προβλήματα ταξινόμησης, και περαιτέρω έλεγχος αυτής.
- Περαιτέρω ανάπτυξη της μεθόδου χρησιμοποιώντας τους ταξινομητές κάθε συστάδας για τη δημιουργία μιας ensemble η οποία θα χρησιμοποιείται ως βασικό ταξινομητής στην ensemble της αμέσως επόμενης (καλύτερης) συστάδας, από τη χειρότερη στη καλύτερη συστάδα.

Βιβλιογραφία

[1] T. M. Mitchell, “The Discipline of Machine Learning,” *Mach. Learn.*, vol. 17, no. July, pp. 1–7, 2006.

[2] T. M. Mitchell, *Machine Learning*, vol. 1, no. 3. 1997.

[3] S. Theodoridis and K. Koutroumbas, *Pattern Recognition*, vol. 8142. Academic Press, 2013.

[4] S. Kalyani and K. S. Swarup, “Design of pattern recognition system for static security assessment and classification,” *Pattern Anal. Appl.*, vol. 15, no. 3, pp. 299–311, Aug. 2012.

[5] K. Singh and M. Xie, “Bootstrap : A Statistical Method,” *Int. Encycl. Educ.*, pp. 46–51, 2010.

[6] R. Kohavi, “A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection,” in *Appears in the International Joint Conference on Articial Intelligence (IJCAI)*, 1995, vol. 118, no. 4, pp. 1–7.

[7] T.-T. Wong, “Performance evaluation of classification algorithms by k-fold and leave-one-out cross validation,” *Pattern Recognit.*, vol. 48, no. 9, pp. 2839–2846, Sep. 2015.

[8] K. M. Ting, “Confusion Matrix,” in *Encyclopedia of Machine Learning*, C. Sammut and G. I. Webb, Eds. Boston, MA: Springer US, 2010, p. 209.

[9] M. Sokolova and G. Lapalme, “A systematic analysis of performance measures for classification tasks,” *Inf. Process. Manag.*, vol. 45, no. 4, pp. 427–437, Jul. 2009.

[10] G. Jurman, S. Riccadonna, and C. Furlanello, “A Comparison of MCC and CEN Error Measures in Multi-Class Prediction,” *PLoS One*, vol. 7, no. 8, p. e41882, Aug. 2012.

[11] S. Boughorbel, F. Jarray, and M. El-Anbari, “Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric,” *PLoS One*, vol. 12, no. 6, p. e0177678, Jun. 2017.

[12] C. X. Ling, J. Huang, and H. Zhang, “AUC: A Statistically Consistent and More Discriminating Measure Than Accuracy,” in *Proceedings of the 18th International Joint Conference on Artificial Intelligence*, 2003, pp. 519–524.

[13] D. J. Hand and R. J. Till, “A Simple Generalisation of the Area Under the ROC Curve for Multiple Class Classification Problems,” *Mach. Learn.*, vol. 45, no. 2, pp. 171–186, 2001.

[14] G. Tsoumakas and I. Katakis, “Multi-label classification: An overview,” *Int J Data Warehous. Min.*, vol. 2007, pp. 1–13, 2007.

[15] J. Cohen, “A Coefficient of Agreement for Nominal Scales,” *Educ. Psychol. Meas.*, vol. 20, no. 1, pp. 37–46, Apr. 1960.

[16] J. L. Gastwirth, “The Estimation of the Lorenz Curve and Gini Index,” *Rev. Econ. Stat.*, vol. 54, no. 3, p. 306, Aug. 1972.

[17] P. Domingos, “A Unified Bias-Variance Decomposition.”

[18] H.-A. Park, “An Introduction to Logistic Regression: From Basic Concepts to Interpretation with Particular Attention to Nursing Domain,” *J. Korean Acad. Nurs.*, vol. 43, no. 2, p. 154, Apr. 2013.

[19] A. Mucherino, P. J. Papajorgji, and P. M. Pardalos, “k-Nearest Neighbor

Classification," 2009, pp. 83–106.

[20] S. M. Omohundro, "Five Balltree Construction Algorithms," 1989.

[21] H. Stern, "NEAREST NEIGHBOUR MATCHING USING K D-TREES," 2002.

[22] A. McCallum and K. Nigam, "A Comparison of Event Models for Naive Bayes Text Classification," *AAAI/ICML-98 Work. Learn. Text Categ.*, vol. 26, no. 9, pp. 41–48, May 1998.

[23] H. Zhang, "The optimality of naive Bayes," *A A*, vol. 1, no. 2, p. 3, 2004.

[24] L. Breiman, "Classification and regression trees," 1984.

[25] L. Rutkowski, M. Jaworski, L. Pietruczuk, and P. Duda, "The CART Decision Tree for Mining Data Streams."

[26] C. Cortes and V. Vapnik, "Support-Vector Networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Nov. 1995.

[27] C. J. C. J. C. J. C. Burges, "A tutorial on support vector machines for pattern recognition," *Data Min. Knowl. Discov.*, vol. 2, no. 2, pp. 121–167, Nov. 1998.

[28] R. Berwick and V. Idiot, "An Idiot ' s guide to Support vector machines (SVMs) SVMs : A New Generation of Learning Algorithms Key Ideas," pp. 1–28, 1990.

[29] K.-R. Muller, S. Mika, G. Ratsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 181–201, Mar. 2001.

[30] M. T. Hagan, H. B. Demuth, M. H. Beale, and O. De Jesús, *Neural network design*, vol. 20. Pws Pub. Boston, 1996.

[31] P. Sadowski, "Notes on backpropagation," *homepage https//www. ics. uci. edu/~ pjsadows/notes. pdf*, 2016.

[32] M. Re and G. Valentini, "Ensemble methods: A review," *Advances in Machine Learning and Data Mining for Astronomy*. pp. 563–594, 2012.

[33] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996.

[34] K. Hsu, "Weight-Adjusted Baggin gof classification Algorithms Sensitive to missing Values," *Int. J. Inf. Educ. Technol. (IJIET)*, vol. 3, no. 5, pp. 560–566, 2013.

[35] B. Parmanto, P. W. Munro, and H. R. Doyle, "Improving committee diagnosis with resampling techniques," *Adv. Neural Inf. Process. Syst.*, vol. 8, pp. 882–888, 1996.

[36] S. C. Bagui, "Combining Pattern Classifiers: Methods and Algorithms," *Technometrics*, vol. 47, no. 4, pp. 517–518, Nov. 2005.

[37] D. H. Wolpert, "Stacked generalization," *Neural Networks*, vol. 5, no. 2, pp. 241–259, Jan. 1992.

[38] Y. Siskos, E. Grigoroudis, and N. F. Matsatsinis, "UTA Methods," Springer, New York, NY, 2016, pp. 315–362.

[39] N. Littlestone and M. K. Warmuth, "The Weighted Majority Algorithm," *Inf. Comput.*, vol. 108, no. 2, pp. 212–261, Feb. 1994.

[40] JENKS and G. F., "The Data Model Concept in Statistical Mapping," *Int.*

Yearb. Cartogr., vol. 7, pp. 186–190, 1967.

[41] Expert Health Data Programming, “What is Jenks Natural Breaks?” [Online]. Available: https://www.ehdp.com/vitalnet/breaks-1.htm. [Accessed: 28-Aug-2018].

[42] D. Dheeru and E. Karra Taniskidou, “{UCI} Machine Learning Repository.” 2017.

[43] T. Kluyver *et al.*, “Jupyter Notebooks -- a publishing format for reproducible computational workflows,” in *Positioning and Power in Academic Publishing: Players, Agents and Agendas*, 2016, pp. 87–90.

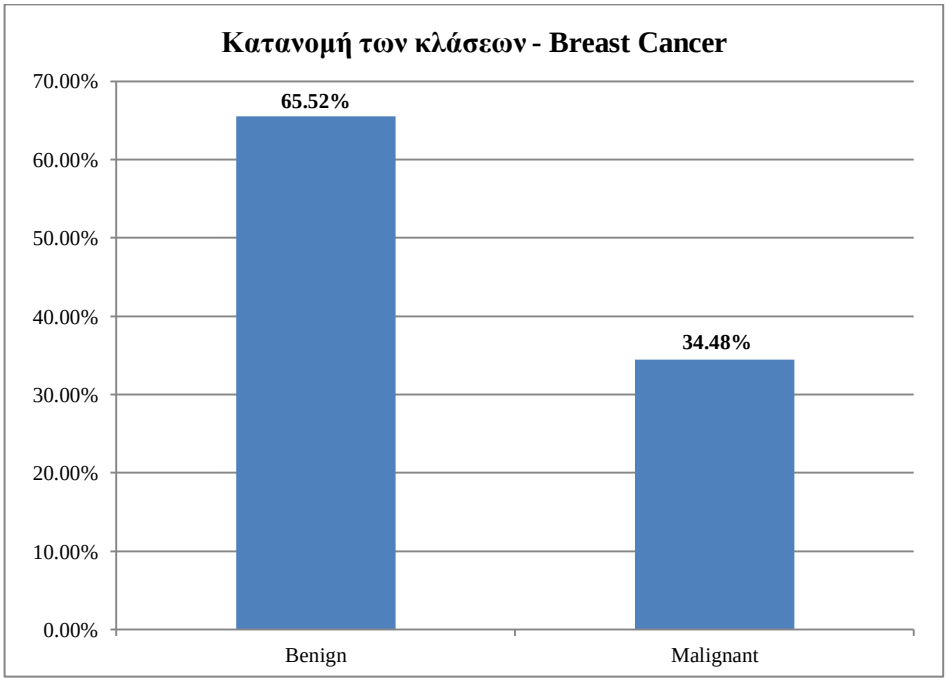
[44] F. Pedregosa *et al.*, “Scikit-learn: Machine Learning in Python ,” *J. Mach. Learn. Res.*, vol. 12, pp. 2825–2830, 2011.

[45] S. Raschka, “MLxtend: Providing machine learning and data science utilities and extensions to Python’s scientific computing stack Software • Review • Repository • Archive,” 2018.

Παράρτημα Α : Σύνολα Δεδομένων

A.1 Breast Cancer Wisconsin (Original)

Το «Breast Cancer Wisconsin (Original)» είναι ένα σύνολο δεδομένων το οποίο αποτελείται από 699 εγγραφές, 10 χαρακτηριστικά και 2 κλάσεις. Οι εγγραφές του συνόλου δεδομένων αποτελούν διάφορα περιστατικά ανθρώπων στους οποίους έχουν εμφανιστεί καρκινικοί όγκοι στη περιοχή του μαστού. Σκοπός είναι η ταξινόμηση αυτών των καρκινικών όγκων σε καλοήθη (benign) ή κακοήθη (malignant). Τα χαρακτηριστικά εξάχθηκαν από ψηφιοποιημένες εικόνες οι οποίες προέκυψαν από αναρροφητική βιοψία με λεπτή βελόνα. Τα δεδομένα είναι γραμμικά διαχωρίσιμα για όλα τα χαρακτηριστικά του συνόλου δεδομένων. Η κατανομή των κλάσεων φαίνεται στο γράφημα . Στο σύνολο δεδομένων περιέχονται 458 καλοήθης περιπτώσεις και 241 κακοήθης. Η κατανομή των κλάσεων φαίνεται στο γράφημα Α.1-1. Επιπλέον στο σύνολο δεδομένων περιέχονται 16 εγγραφές οι οποίες έχουν ελλείπουσες τιμές.



Γράφημα Α.1-1: Ποσοστιαία κατανομή των κλάσεων του συνόλου δεδομένων «Breast Cancer».

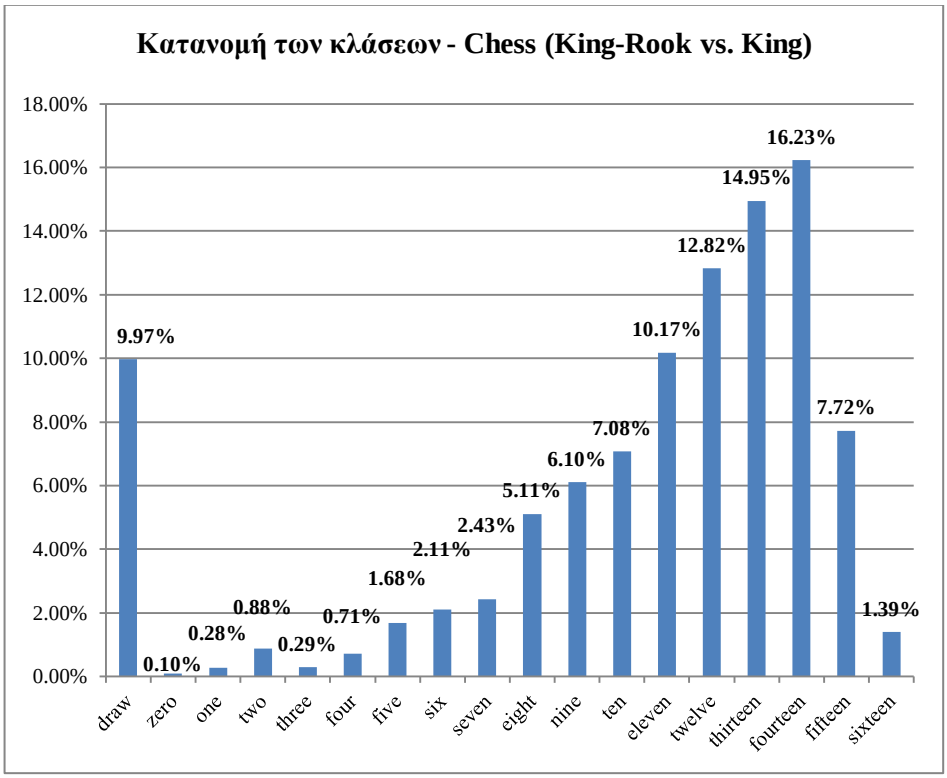
Παρακάτω ακολουθεί μια σύντομη παρουσίαση και επεξήγηση των χαρακτηριστικών του συνόλου δεδομένων «Breast Cancer»:

- **Clump thickness**: είναι ο αριθμός των στρωμάτων του κυτταρικού δείγματος.
- **Uniformity of cell size**: ο βαθμός της ομοιομορφίας του μεγέθους των κυττάρων που περιέχονται στο δείγμα.
- **Uniformity of cell shape:** ο βαθμός της ομοιομορφίας του σχήματος των κυττάρων που περιέχονται στο δείγμα.
- **Marginal adhesion**: ο οριακός βαθμός της προσκόλλησης των καρκινικών κυττάρων.
- **Single epithelial cell size**: το μέγεθος των μονών επιθηλιακών κυττάρων.

- **Bare nuclei**: η ύπαρξη γυμνών νουκλεοτιδίων.
- **Bland Chromatin**: η ύπαρξη μιας συγκεκριμένου τύπου υφής στην επιφάνεια των πυρήνων των κυττάρων.
- **Normal Nucleoli**: ο βαθμός της κανονικότητας μιας συγκεκριμένου τύπου δομής που βρίσκεται στο εσωτερικό των πυρήνων των κυττάρων.
- **Mitosis**: ο ρυθμός με τον οποίο πραγματοποιείται το φαινόμενο της μίτωσης.

A.2 Chess (King-Rook vs. King)

Το σύνολο δεδομένων «Chess (King-Rook vs. King)» αποτελείται από 28056 εγγραφές, 6 χαρακτηριστικά και 17 κλάσεις. Σκοπός της ταξινόμησης είναι να βρεθεί σε πόσα βήματα θα νικήσουν τα «άσπρα» σε ένα παιχνίδι σκακιού όταν τα «άσπρα» έχουν στη διάθεση τους ένα αξιωματικό και το βασιλιά και τα «μαύρα» μόνο το βασιλιά. Εκτός του ενδεχόμενο της νίκης των «άσπρων» υπάρχει και το ενδεχόμενο της ισοπαλίας, το οποίο αποτελεί και μια από τις κλάσεις του προβλήματος ταξινόμησης. Οι εγγραφές του συνόλου δεδομένων αποτελούνται από τις συντεταγμένες (γραμμή, στήλη) καθενός από τα προαναφερθέντα πιόνια. Η κατανομή των κλάσεων φαίνεται στο γράφημα Α.1-2 .



Γράφημα Α.2-1: Ποσοστιαία κατανομή των κλάσεων του συνόλου δεδομένων «Chess (King-Rook vs. King)».

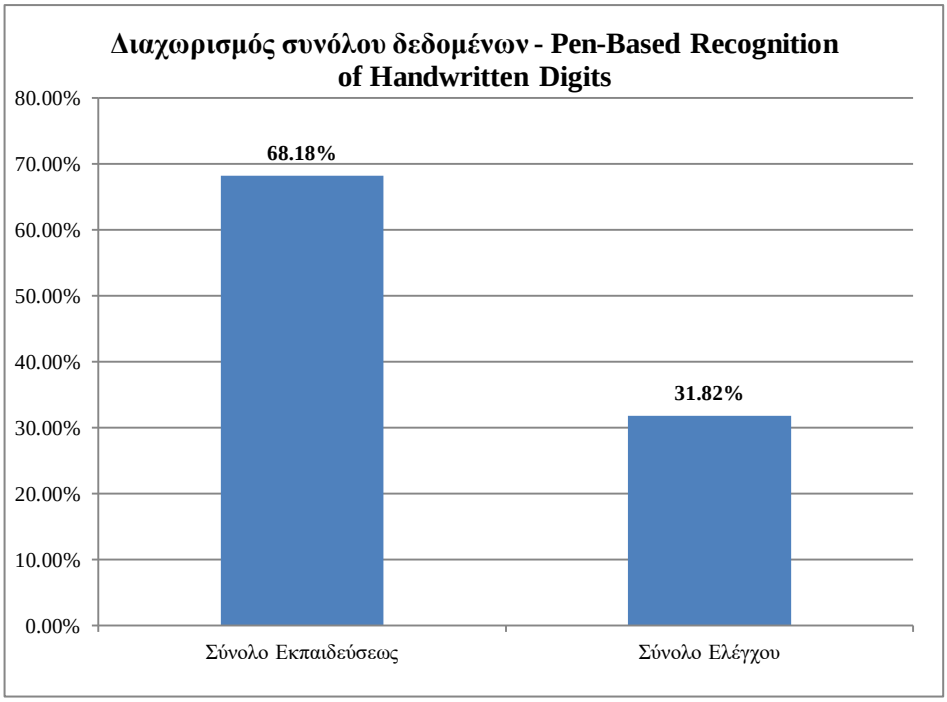
Παρακάτω ακολουθεί μια σύντομη παρουσίαση και επεξήγηση των χαρακτηριστικών του συνόλου δεδομένων «Chess (King-Rook vs. King)»:

- **White King file:** ο αριθμός στήλης της θέσης του «λευκού» βασιλιά.
- **White King rank:** ο αριθμός γραμμής της θέσης του «λευκού» βασιλιά.

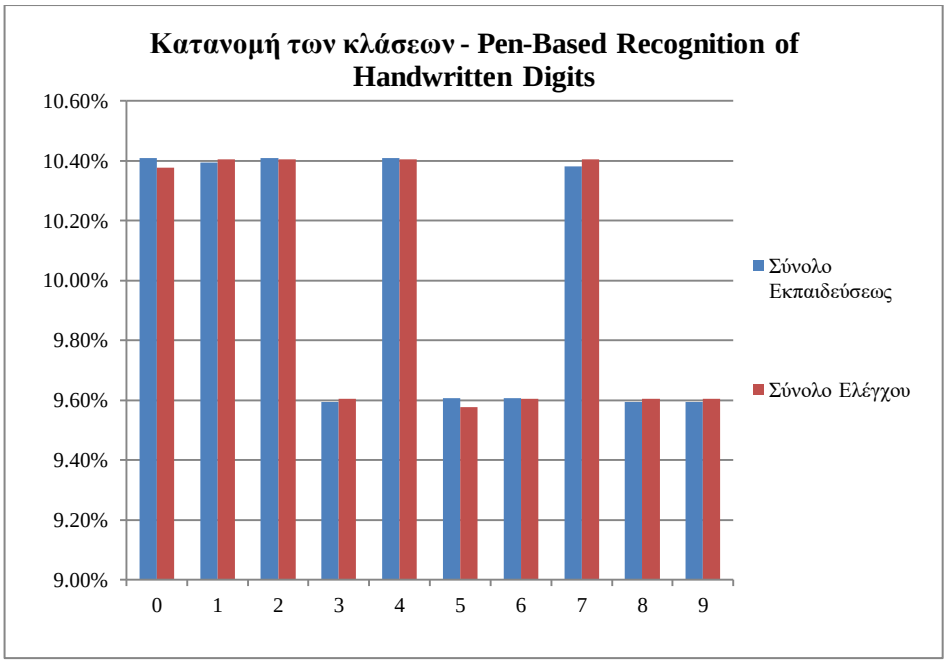
- **White Rook file:** ο αριθμός στήλης της θέσης του «λευκού» αξιωματικού.
- **White Rook rank:** ο αριθμός γραμμής της θέσης του «λευκού» αξιωματικού.
- **Black King file:** ο αριθμός στήλης της θέσης του «μαύρου» βασιλιά.
- **Black King rank:** ο αριθμός γραμμής της θέσης του «μαύρου» βασιλιά.

A.3 Pen-Based Recognition of Handwritten Digits

Το σύνολο δεδομένων «Pen-Based Recognition of Handwritten Digits» αποτελείται από 10992 εγγραφές. 16 χαρακτηριστικά και 10 κλάσεις. Η κάθε εγγραφή αποτελεί τις συντεταγμένες (x, y) 8 σημείων στο επίπεδο της επιφάνειας μιας οθόνης αφής, με τη ένωση των οποίων προκύπτει ένας χαρακτήρας από 0 μέχρι το 9. Σκοπός της ταξινόμησης είναι η χρησιμοποίηση των συντεταγμένων των σημείων για την αναγνώριση των πιθανών σχηματιζόμενων χαρακτήρων. Το συγκεκριμένο σύνολο δεδομένων είναι ήδη χωρισμένο σε υποσύνολα εκπαιδεύσεως και ελέγχου. Η κατανομή του συνόλου δεδομένων, καθώς και οι κατανομές των κλάσεων φαίνονται στα γραφήματα A.3-1, A.3-2 αντίστοιχα.



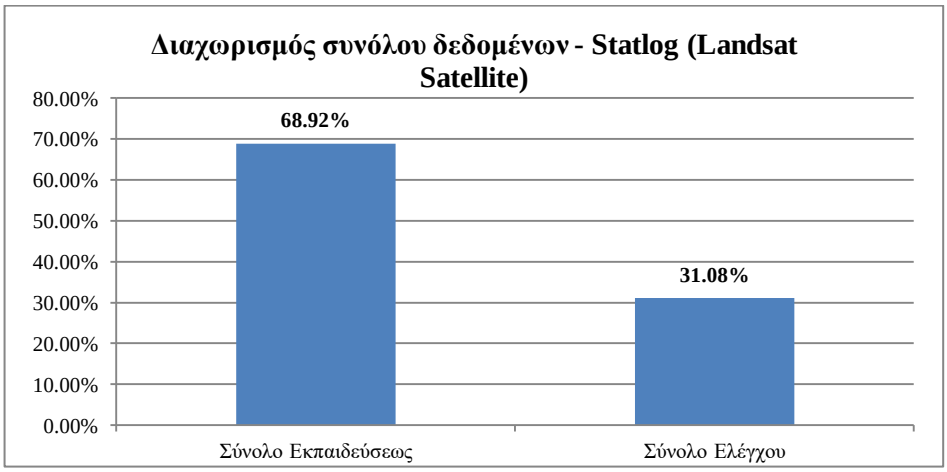
Γράφημα A.3-1: Διαχωρισμός συνόλου δεδομένων «Pen-Based Recognition of Handwritten Digits» σε υποσύνολα εκπαιδεύσεως και ελέγχου.



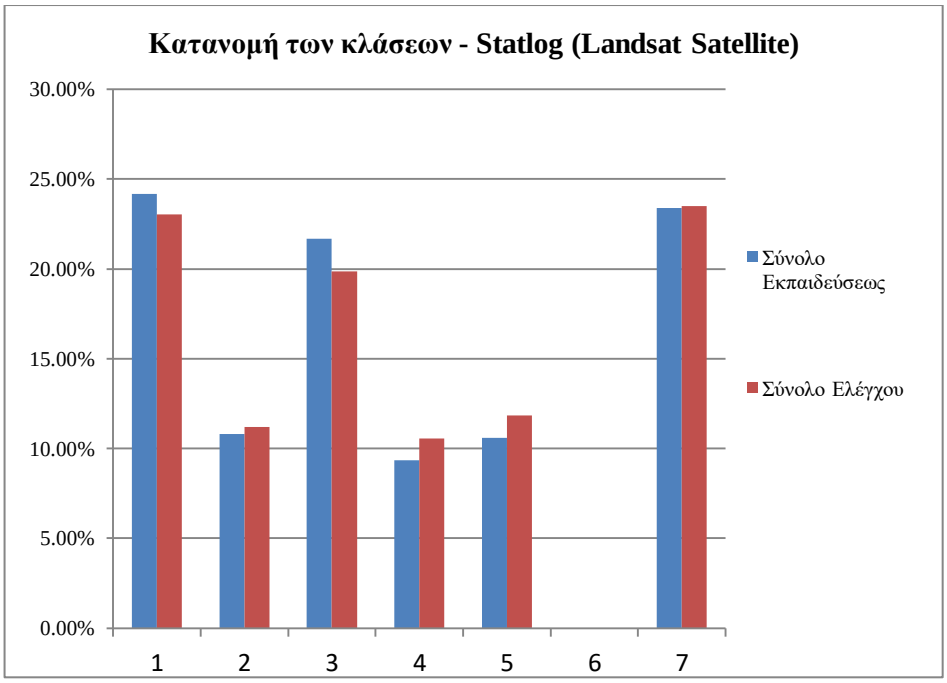
Γράφημα Α.3-2: Ποσοστιαία κατανομή των κλάσεων του συνόλου δεδομένων «Pen-Based Recognition of Handwritten Digits».

A.4 Statlog (Landsat Satellite)

Το σύνολο δεδομένων «Statlog (Landsat Satellite)» αποτελείται από 6345 εγγραφές, 36 χαρακτηριστικά και 7 κλάσεις . Τα δεδομένα αποτελούν ένα κομμάτι 82x100 μιας δορυφορικής φωτογραφίας 2340 x 3380 τραβηγμένης από τη NASA στις 4 φασματικές μπάντες. Οι εγγραφές αποτελούν γειτονιές 3x3 από pixel σε κάθε φασματική μπάντα. Οι τιμές των χαρακτηριστικών είναι ακέραιες και κυμαίνονται στο $[0, 255]$, με το 0 να συμβολίζει το άσπρο και 255 το μαύρο. Σκοπός της ταξινόμησης είναι η ταξινόμηση του αντικειμένου στο μεσαίο pixel. Κάθε pixel αντιστοιχεί σε εμβαδό 80x80 m^2 σε πραγματικές διαστάσεις. Το συγκεκριμένο σύνολο δεδομένων είναι ήδη χωρισμένο σε υποσύνολα εκπαιδεύσεως και ελέγχου. Η κατανομή του συνόλου δεδομένων, καθώς και οι κατανομές των κλάσεων φαίνονται στα γραφήματα Α.4-1, Α.4-2 αντίστοιχα.



Γράφημα Α.4-1: Διαχωρισμός συνόλου δεδομένων «Statlog (Landsat Satellite)» σε υποσύνολα εκπαιδεύσεως και ελέγχου.



Γράφημα Α.4-2: Ποσοστιαία κατανομή των κλάσεων του συνόλου δεδομένων «A.4 Statlog (Landsat Satellite)».

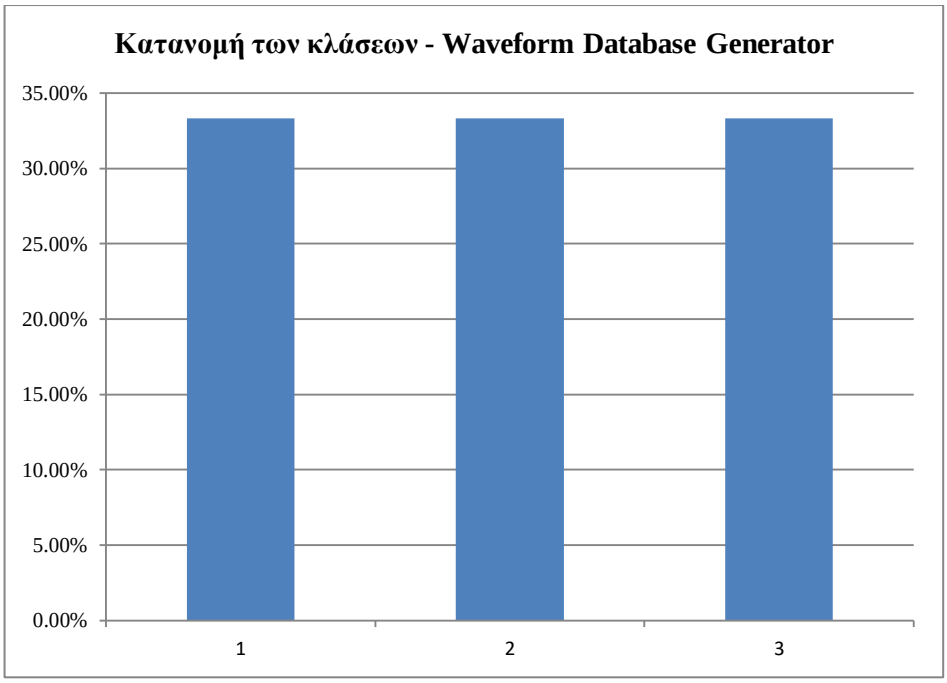
Πίνακας Α.4-1
Περιγραφή των κλάσεων

Κλάση	Περιγραφή
1	Κόκκινο χώμα
2	Καλλιέργεια βαμβακιού
3	Γκρι χώμα
4	Υγρό γκρι χώμα
5	Χώμα με ίχνη βλάστησης
6	Κλάση ανάμειξης (περιέχει όλους του τύπους)
7	Πολύ υγρό γκρι χώμα

A.5 Waveform Database Generator

Το σύνολο δεδομένων «Waveform Database Generator» αποτελείται από 5000 εγγραφές, 21 χαρακτηριστικά και 3 κλάσεις. Οι εγγραφές είναι κυματομορφές και αποτελούνται από 21 σημεία οι τιμές των οποίων είναι ακέραιες και κυμαίνονται στο $[0,6]$.

Σκοπός της ταξινόμησης είναι να ταξινομηθούν όλες οι κυματομορφές στις 3 προαναφερθέντες κλάσεις κυματομορφών. Η κατανομή των κλάσεων φαίνεται στο γράφημα Α.5-1.



Γράφημα Α.5-1: Ποσοστιαία κατανομή των κλάσεων του συνόλου δεδομένων «Chess (King-Rook vs. King)».

Παράρτημα Β : Πολυκριτήριοι Πίνακες

B.1 Breast Cancer Wisconsin (Original)

Εναλλακτικές/Κριτήρια	F1	MCC	Cohen's Kappa	Precision (Macro)	Hamming Loss	Recall (Macro)	Accuracy	Ranking
Decision Tree FOLD0	0.944789	0.889807	0.889590	0.942204	0.051020	0.947619	0.948980	26
Decision Tree FOLD1	0.869739	0.742734	0.739938	0.863538	0.122449	0.879365	0.877551	29
Decision Tree FOLD2	0.934869	0.873256	0.869969	0.927423	0.061224	0.946032	0.938776	27
Decision Tree FOLD3	0.892544	0.792497	0.785933	0.884895	0.102041	0.907937	0.897959	28
Decision Tree FOLD4	0.955300	0.911551	0.910640	0.950137	0.041237	0.961485	0.958763	25
GaussianNB FOLD0	0.966873	0.933982	0.933754	0.964158	0.030612	0.969841	0.969388	11
GaussianNB FOLD1	0.944789	0.889807	0.889590	0.942204	0.051020	0.947619	0.948980	21
GaussianNB FOLD2	0.945428	0.892878	0.890966	0.939035	0.051020	0.953968	0.948980	19
GaussianNB FOLD3	0.934869	0.873256	0.869969	0.927423	0.061224	0.946032	0.938776	23
GaussianNB FOLD4	0.977650	0.956275	0.955320	0.972222	0.020619	0.984127	0.979381	18
KNN FOLD0	0.954963	0.910891	0.909968	0.961772	0.040816	0.949206	0.959184	0
KNN FOLD1	0.944789	0.889807	0.889590	0.942204	0.051020	0.947619	0.948980	24
KNN FOLD2	0.955556	0.911111	0.911111	0.955556	0.040816	0.955556	0.959184	22
KNN FOLD3	0.966873	0.933982	0.933754	0.964158	0.030612	0.969841	0.969388	20
KNN FOLD4	0.988754	0.977757	0.977510	0.985714	0.010309	0.992063	0.989691	13
LR FOLD0	0.966873	0.933982	0.933754	0.964158	0.030612	0.969841	0.969388	5
LR FOLD1	0.956093	0.913105	0.912226	0.951263	0.040816	0.961905	0.959184	10
LR FOLD2	0.944083	0.888401	0.888179	0.947151	0.051020	0.941270	0.948980	17
LR FOLD3	0.966873	0.933982	0.933754	0.964158	0.030612	0.969841	0.969388	4
LR FOLD4	0.988754	0.977757	0.977510	0.985714	0.010309	0.992063	0.989691	3
MLP FOLD0	0.955556	0.911111	0.911111	0.955556	0.040816	0.955556	0.959184	7
MLP FOLD1	0.956580	0.916764	0.913313	0.948718	0.040816	0.968254	0.959184	7
MLP FOLD2	0.944083	0.888401	0.888179	0.947151	0.051020	0.941270	0.948980	15
MLP FOLD3	0.945428	0.892878	0.890966	0.939035	0.051020	0.953968	0.948980	11
MLP FOLD4	0.977358	0.954715	0.954715	0.977358	0.020619	0.977358	0.979381	14
SVM FOLD0	0.966873	0.933982	0.933754	0.964158	0.030612	0.969841	0.969388	1
SVM FOLD1	0.956093	0.913105	0.912226	0.951263	0.040816	0.961905	0.959184	6
SVM FOLD2	0.955556	0.911111	0.911111	0.955556	0.040816	0.955556	0.959184	15
SVM FOLD3	0.956093	0.913105	0.912226	0.951263	0.040816	0.961905	0.959184	9
SVM FOLD4	0.988754	0.977757	0.977510	0.985714	0.010309	0.992063	0.989691	2

B.2 Chess (King-Rook vs. King)

Εναλλακτικές/Κριτήρια	F1	MCC	Cohen's Kappa	Precision (Macro)	Hamming Loss	Recall (Macro)	Accuracy	Ranking
Decision FOLD0	0.746767	0.692747	0.692677	0.736107	0.275476	0.762758	0.724524	23
Decision FOLD1	0.761836	0.691692	0.691638	0.753311	0.276379	0.772508	0.723621	22
Decision FOLD2	0.750181	0.693801	0.693736	0.736431	0.274370	0.770333	0.725630	20
Decision FOLD3	0.748806	0.691972	0.691852	0.740489	0.276064	0.760611	0.723936	24
Decision FOLD4	0.764358	0.686596	0.686502	0.757279	0.280939	0.773571	0.719061	21
GaussianNB FOLD0	0.097166	0.068292	0.062175	0.152512	0.879034	0.249985	0.120966	25
GaussianNB FOLD1	0.076361	0.054710	0.050071	0.130106	0.891431	0.222377	0.108569	26
GaussianNB FOLD2	0.088461	0.068532	0.062779	0.167725	0.880377	0.246408	0.119623	28
GaussianNB FOLD3	0.094876	0.072878	0.066877	0.123868	0.871272	0.233143	0.128728	27
GaussianNB FOLD4	0.086266	0.075199	0.069004	0.152178	0.870885	0.192135	0.129115	29
KNN FOLD0	0.595503	0.571256	0.569956	0.663626	0.382465	0.559644	0.617535	10
KNN FOLD1	0.569121	0.554650	0.553537	0.645882	0.397407	0.535124	0.602593	11
KNN FOLD2	0.609670	0.579894	0.578585	0.718916	0.375159	0.570557	0.624841	13
KNN FOLD3	0.566512	0.567480	0.566498	0.691620	0.385674	0.522503	0.614326	14
KNN FOLD4	0.578219	0.564958	0.563806	0.652603	0.388364	0.552576	0.611636	12
LR FOLD0	0.324416	0.295190	0.293820	0.290613	0.638882	0.417192	0.361118	19
LR FOLD1	0.318078	0.274823	0.273415	0.285452	0.656751	0.407729	0.343249	17
LR FOLD2	0.316489	0.274692	0.273366	0.283122	0.656656	0.411547	0.343344	18
LR FOLD3	0.306995	0.284343	0.283099	0.279468	0.647464	0.389578	0.352536	16
LR FOLD4	0.322016	0.286370	0.285011	0.292407	0.645828	0.401009	0.354172	15
MLP FOLD0	0.615642	0.656836	0.656759	0.633277	0.307243	0.608599	0.692757	4
MLP FOLD1	0.654113	0.660599	0.660179	0.694299	0.304094	0.636023	0.695906	2
MLP FOLD2	0.647746	0.660489	0.660282	0.668747	0.304149	0.643232	0.695851	1
MLP FOLD3	0.644382	0.684503	0.684058	0.653982	0.282692	0.640754	0.717308	0
MLP FOLD4	0.647050	0.662512	0.662258	0.660977	0.302373	0.652110	0.697627	3
SVM FOLD0	0.600284	0.586844	0.586300	0.650115	0.368996	0.576995	0.631004	7
SVM FOLD1	0.563351	0.568849	0.568291	0.606846	0.385202	0.548013	0.614798	8
SVM FOLD2	0.618043	0.582698	0.582018	0.650035	0.372614	0.600984	0.627386	9
SVM FOLD3	0.592833	0.589312	0.588862	0.653847	0.366811	0.574512	0.633189	6
SVM FOLD4	0.602182	0.582394	0.581791	0.658860	0.373054	0.582832	0.626946	5

B.3 Pen-Based Recognition of Handwritten Digits

Εναλλακτικές/Κριτήρια	F1	MCC	Cohen's Kappa	Precision (Macro)	Hamming Loss	Recall (Macro)	Accuracy	Ranking
Decision Tree FOLD0	0.970101	0.966668	0.966661	0.970183	0.030000	0.970085	0.970000	21
Decision Tree FOLD1	0.958501	0.953373	0.953327	0.958788	0.042000	0.958600	0.958000	27
Decision Tree FOLD2	0.961395	0.957086	0.957030	0.961766	0.038667	0.961538	0.961333	26
Decision Tree FOLD3	0.954654	0.949591	0.949552	0.955073	0.045394	0.954584	0.954606	28
Decision Tree FOLD4	0.957128	0.952431	0.952395	0.957355	0.042838	0.957235	0.957162	29
GaussianNB FOLD0	0.885616	0.875594	0.874797	0.892389	0.112667	0.886592	0.887333	20
GaussianNB FOLD1	0.877667	0.866529	0.865909	0.882312	0.120667	0.878686	0.879333	24
GaussianNB FOLD2	0.865554	0.855215	0.854059	0.874336	0.131333	0.868002	0.868667	25
GaussianNB FOLD3	0.876815	0.867250	0.866455	0.883762	0.120160	0.878282	0.879840	22
GaussianNB FOLD4	0.869431	0.858891	0.857936	0.876896	0.127845	0.871296	0.872155	23
KNN FOLD0	0.995325	0.994817	0.994814	0.995435	0.004667	0.995246	0.995333	16
KNN FOLD1	0.996022	0.995556	0.995555	0.996060	0.004000	0.995994	0.996000	12
KNN FOLD2	0.992147	0.991115	0.991110	0.992199	0.008000	0.992147	0.992000	19
KNN FOLD3	0.997943	0.997775	0.997774	0.997974	0.002003	0.997917	0.997997	10
KNN FOLD4	0.992671	0.991825	0.991818	0.992694	0.007363	0.992709	0.992637	18
LR FOLD0	0.955039	0.950440	0.950367	0.955237	0.044667	0.955502	0.955333	13
LR FOLD1	0.963105	0.959298	0.959255	0.963090	0.036667	0.963515	0.963333	15
LR FOLD2	0.951593	0.946697	0.946660	0.951596	0.048000	0.951923	0.952000	11
LR FOLD3	0.953447	0.948850	0.948810	0.953812	0.046061	0.953461	0.953939	17
LR FOLD4	0.960718	0.956880	0.956857	0.960908	0.038822	0.960742	0.961178	14
MLP FOLD0	0.991939	0.991116	0.991110	0.991945	0.008000	0.991987	0.992000	6
MLP FOLD1	0.987390	0.985939	0.985924	0.987306	0.012667	0.987607	0.987333	5
MLP FOLD2	0.988092	0.986670	0.986664	0.988204	0.012000	0.988034	0.988000	7
MLP FOLD3	0.980483	0.978538	0.978487	0.980709	0.019359	0.980699	0.980641	9
MLP FOLD4	0.989263	0.988112	0.988099	0.989330	0.010710	0.989316	0.989290	4
SVM FOLD0	0.997276	0.997039	0.997037	0.997304	0.002667	0.997276	0.997333	0
SVM FOLD1	0.995967	0.995556	0.995555	0.995949	0.004000	0.995994	0.996000	3
SVM FOLD2	0.995350	0.994816	0.994814	0.995370	0.004667	0.995353	0.995333	1
SVM FOLD3	0.995236	0.994810	0.994807	0.995321	0.004673	0.995183	0.995327	8
SVM FOLD4	0.994611	0.994056	0.994049	0.994700	0.005355	0.994586	0.994645	2

B.4 Statlog (Landsat Satellite)

Εναλλακτικές/Κριτήρια	F1	MCC	Cohen's Kappa	Precision (Macro)	Hamming Loss	Recall (Macro)	Accuracy	Ranking
Decision Tree FOLD0	0.781721	0.762945	0.761967	0.781851	0.193476	0.787243	0.806524	29
Decision Tree FOLD1	0.821114	0.811478	0.811353	0.827719	0.152027	0.815406	0.847973	25
Decision Tree FOLD2	0.823545	0.806256	0.804839	0.822543	0.158963	0.833385	0.841037	23
Decision Tree FOLD3	0.823275	0.808387	0.806367	0.821734	0.158014	0.835913	0.841986	18
Decision Tree FOLD4	0.823407	0.819972	0.819720	0.824130	0.145763	0.823565	0.854237	28
GaussianNB FOLD0	0.807399	0.785592	0.783602	0.806029	0.176603	0.820757	0.823397	15
GaussianNB FOLD1	0.772307	0.752142	0.750850	0.776299	0.202703	0.777038	0.797297	17
GaussianNB FOLD2	0.764831	0.742632	0.741128	0.770490	0.210823	0.769113	0.789177	19
GaussianNB FOLD3	0.752808	0.720737	0.717737	0.756827	0.231377	0.765229	0.768623	20
GaussianNB FOLD4	0.779048	0.755932	0.753753	0.781578	0.201130	0.789242	0.798870	16
KNN FOLD0	0.878017	0.874675	0.874195	0.890121	0.101237	0.870379	0.898763	5
KNN FOLD1	0.872183	0.870446	0.870016	0.880553	0.104730	0.865985	0.895270	4
KNN FOLD2	0.860664	0.863118	0.862440	0.877658	0.110485	0.848722	0.889515	9
KNN FOLD3	0.863383	0.860177	0.859819	0.873701	0.112867	0.856492	0.887133	8
KNN FOLD4	0.870805	0.871385	0.870967	0.878911	0.103955	0.864725	0.896045	0
LR FOLD0	0.825709	0.827252	0.826600	0.837021	0.139483	0.820637	0.860517	22
LR FOLD1	0.800826	0.805859	0.805284	0.813526	0.156532	0.792842	0.843468	24
LR FOLD2	0.783597	0.802961	0.802259	0.791332	0.158963	0.782158	0.841037	27
LR FOLD3	0.788277	0.790455	0.790265	0.788854	0.169300	0.789737	0.830700	26
LR FOLD4	0.804356	0.823860	0.822714	0.815369	0.142373	0.802535	0.857627	21
MLP FOLD0	0.804663	0.800095	0.799215	0.808107	0.161980	0.804376	0.838020	12
MLP FOLD1	0.828857	0.821824	0.821198	0.833744	0.144144	0.826508	0.855856	10
MLP FOLD2	0.786748	0.796421	0.794869	0.798257	0.164600	0.786195	0.835400	13
MLP FOLD3	0.810243	0.806139	0.805151	0.818081	0.156885	0.807745	0.843115	14
MLP FOLD4	0.824793	0.824308	0.822681	0.842891	0.142373	0.821399	0.857627	11
SVM FOLD0	0.875051	0.866285	0.864683	0.868344	0.110236	0.890715	0.889764	1
SVM FOLD1	0.884600	0.876153	0.875216	0.881573	0.101351	0.892354	0.898649	3
SVM FOLD2	0.865236	0.860220	0.859680	0.861802	0.113867	0.871702	0.886133	7
SVM FOLD3	0.869221	0.860270	0.858658	0.865522	0.115124	0.881683	0.884876	6
SVM FOLD4	0.885365	0.877549	0.876306	0.881629	0.100565	0.895260	0.899435	2

B.5 Waveform Database Generator

Εναλλακτικές/Κριτήρια	F1	MCC	Cohen's Kappa	Precision (Macro)	Hamming Loss	Recall (Macro)	Accuracy	Ranking
Decision Tree FOLD0	0.747603	0.621840	0.621086	0.749920	0.252496	0.747244	0.747504	26
Decision Tree FOLD1	0.754474	0.632207	0.631950	0.754824	0.245364	0.754729	0.754636	25
Decision Tree FOLD2	0.733172	0.599686	0.599287	0.734349	0.267143	0.733016	0.732857	27
Decision Tree FOLD3	0.709956	0.564982	0.564954	0.709939	0.290000	0.710046	0.710000	29
Decision Tree FOLD4	0.738023	0.606732	0.606662	0.738348	0.262178	0.737874	0.737822	28
GaussianNB FOLD0	0.793524	0.728637	0.709312	0.834399	0.194009	0.807488	0.805991	21
GaussianNB FOLD1	0.786539	0.716571	0.696481	0.833810	0.202568	0.798694	0.797432	23
GaussianNB FOLD2	0.793108	0.729824	0.708990	0.838174	0.194286	0.807409	0.805714	22
GaussianNB FOLD3	0.760593	0.687516	0.664060	0.815786	0.224286	0.777627	0.775714	24
GaussianNB FOLD4	0.805948	0.744772	0.725212	0.849973	0.183381	0.817871	0.816619	20
KNN FOLD0	0.849847	0.781723	0.777571	0.857281	0.148359	0.852281	0.851641	2
KNN FOLD1	0.846624	0.773373	0.771132	0.850619	0.152639	0.847823	0.847361	7
KNN FOLD2	0.847612	0.774575	0.772946	0.849744	0.151429	0.849121	0.848571	11
KNN FOLD3	0.837233	0.761689	0.757974	0.844007	0.161429	0.839321	0.838571	16
KNN FOLD4	0.871570	0.810710	0.808769	0.874661	0.127507	0.872776	0.872493	5
LR FOLD0	0.871633	0.812172	0.809639	0.875252	0.126961	0.873719	0.873039	1
LR FOLD1	0.864249	0.797567	0.796756	0.865702	0.135521	0.864612	0.864479	13
LR FOLD2	0.863897	0.797522	0.796494	0.865258	0.135714	0.864744	0.864286	14
LR FOLD3	0.840423	0.765161	0.762263	0.845523	0.158571	0.842159	0.841429	18
LR FOLD4	0.870202	0.808364	0.806613	0.872883	0.128940	0.871351	0.871060	4
MLP FOLD0	0.884116	0.827182	0.826705	0.884402	0.115549	0.884859	0.884451	0
MLP FOLD1	0.858601	0.788390	0.788166	0.858933	0.141227	0.858760	0.858773	10
MLP FOLD2	0.863083	0.794439	0.794285	0.863459	0.137143	0.863050	0.862857	9
MLP FOLD3	0.844025	0.767226	0.766484	0.845024	0.155714	0.844702	0.844286	17
MLP FOLD4	0.860780	0.799166	0.793713	0.870681	0.137536	0.862710	0.862464	8
SVM FOLD0	0.870575	0.809455	0.807495	0.873285	0.128388	0.872221	0.871612	3
SVM FOLD1	0.862950	0.794906	0.794605	0.863379	0.136947	0.863176	0.863053	12
SVM FOLD2	0.857033	0.786515	0.785772	0.858123	0.142857	0.857523	0.857143	15
SVM FOLD3	0.835314	0.755074	0.753657	0.837670	0.164286	0.836240	0.835714	19
SVM FOLD4	0.867818	0.803295	0.802308	0.869382	0.131805	0.868410	0.868195	6