ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ

ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEBINTELLIGENCE

ΤΕΧΝΙΚΕΣ ΜΕΙΩΣΗΣ ΔΕΔΟΜΕΝΩΝ ΕΚΠΑΙΔΕΥΣΗΣ: ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΚΑΙ ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

της

ΑΛΕΞΑΝΔΡΑΣ ΤΣΑΒΔΑΡΙΔΟΥ

Επιβλέπων : Στέφανος Ουγιάρογλου
Επίκουρος Καθηγητής, ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Ιούνιος 2024

ΔΙΕΘΝΕΣ ΠΑΝΕΠΙΣΤΗΜΙΟ ΤΗΣ ΕΛΛΑΔΟΣ

ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ – WEB INTELLIGENCE

ΤΕΧΝΙΚΕΣ ΜΕΙΩΣΗΣ ΔΕΔΟΜΕΝΩΝ ΕΚΠΑΙΔΕΥΣΗΣ ΑΝΑΣΚΟΠΗΣΗ ΤΗΣ ΒΙΒΛΙΟΓΡΑΦΙΑΣ ΚΑΙ ΠΕΙΡΑΜΑΤΙΚΗ ΜΕΛΕΤΗ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

της

ΑΛΕΞΑΝΔΡΑΣ ΤΣΑΒΔΑΡΙΔΟΥ

Επιβλέπων : Στέφανος Ουγιάρογλου
Επίκουρος Καθηγητής ΔΙ.ΠΑ.Ε.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή στις Choose a date.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Όνομα Επώνυμο	Όνομα Επώνυμο	'Ονομα Επώνυμο
Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.	Choose an item. ΔΙ.ΠΑ.Ε.

Θεσσαλονίκη, Ιούνιος 2024

(Υπογραφή)

.....

Click here to enter text.

Click here to enter text.

Περίληψη

Η εκπόνηση της παρούσας διπλωματικής εργασίας επικεντρώνεται στη μελέτη και την ανάλυση διαφόρων αλγορίθμων μείωσης δεδομένων, με έμφαση στους αλγορίθμους επιλογής προτύπων. Σκοπός της εργασίας είναι η ανάλυση των αλγορίθμων αυτών, η διενέργεια πειραμάτων μέσω του KEEL σε διαφορετικά σύνολα δεδομένων καθώς και η σύγκριση των μετρήσεων ακρίβειας μέσω των στατιστικών τεστ Friedman και Wilcoxon. Στα πλαίσια της διπλωματικής εργασίας, τα πειράματα πραγματοποιήθηκαν σε διάφορα σύνολα δεδομένων από διαφορετικούς τομείς. Με βάση τη στατιστική μελέτη και τη σύγκριση των αλγορίθμων δεν υπήρξαν στατιστικές διαφορές κάτι το οποίο επιβεβαιώνει την ανάγκη προσαρμογής των παραμέτρων για τον κάθε αλγόριθμο. Γενικά, προτείνεται η δοκιμή των αλγορίθμων σε διάφορα σύνολα δεδομένων και η δημιουργία διεπαφών η οποία θα διευκολύνει τη χρησιμότητα του KEEL. Η παρούσα διπλωματική συμβάλλει στη βιβλιογραφική ανασκόπηση κάποιων βασικών αλγορίθμων μείωσης δεδομένων καθώς και στην πειραματική μελέτη μέσω του λογισμικού KEEL.

Λέξεις Κλειδιά: KEEL, Κατηγοριοποίηση, k-NN, Τεχνικές Μείωσης Δεδομένων, Αλγόριθμοι Παραγωγής Προτύπων, Συμπύκνωση

Abstract

This thesis focuses on the study and analysis of various data reduction algorithms, with emphasis on prototype selection algorithms. Its aim is to analyze these algorithms, conduct experiments through KEEL, on different datasets and compare the accuracy measurements through Friedman and Wilcoxon statistical tests. After comparing the algorithms no statistical differences occurred, which confirms the need to adjust the parameters for each algorithm. In general, it is suggested to test the algorithms on different datasets and create interfaces that will facilitate the utility of KEEL. Moreover, it contributes in the algorithms' understanding, as well as, of data reduction techniques and how KEEL works.

Keywords: KEEL, Classification, k-NN, Data reduction, Prototype Selection Algorithms, Condensing

Πίνακας περιεχομένων

Κατάλογος εικόνων

Κατάλογος πινάκων

1 *Εισαγωγή*

Η αποδοτικότητα και η αποτελεσματικότητα των αλγορίθμων Εξόρυξης Δεδομένων (Data Mining) είναι ένα σημαντικό ερευνητικό πρόβλημα που έχει προσελκύσει την προσοχή τόσο της ακαδημαϊκής κοινότητας όσο και της βιομηχανίας. Η ταξινόμηση θεωρείται κρίσιμη για την εξόρυξη δεδομένων. Η εξόρυξη δεδομένων λοιπόν, είναι η διαδικασία ταξινόμησης μεγάλων συνόλων δεδομένων για τον εντοπισμό μοτίβων και σχέσεων που μπορούν να βοηθήσουν στην επίλυση επιχειρηματικών προβλημάτων μέσω της ανάλυσης δεδομένων. Οι τεχνικές και τα εργαλεία εξόρυξης δεδομένων βοηθούν τις επιχειρήσεις να προβλέπουν τις μελλοντικές τάσεις και να λαμβάνουν πιο τεκμηριωμένες επιχειρηματικές αποφάσεις. Χρησιμοποιεί τεχνικές από διάφορους τομείς όπως η στατιστική, η μηχανική μάθηση, η αναγνώριση προτύπων και οι βάσεις δεδομένων. Αποτελεί βασικό μέρος της ανάλυσης δεδομένων και έναν από τους βασικούς κλάδους της επιστήμης των δεδομένων, η οποία χρησιμοποιεί προηγμένες τεχνικές ανάλυσης για την εύρεση χρήσιμων πληροφοριών σε σύνολα δεδομένων. Οι διαδικασίες εξόρυξης δεδομένων περιλαμβάνουν την επιλογή δεδομένων, την προεπεξεργασία, την εξόρυξη δεδομένων, την ερμηνεία και την αξιολόγηση. Οι δύο πρώτες διαδικασίες (επιλογή δεδομένων και προεπεξεργασία) παίζουν καθοριστικό ρόλο στην επιτυχή εξόρυξη δεδομένων. Η προεπεξεργασία εκτελείται μόνο μία φορά. Ωστόσο, το κόστος προεπεξεργασίας είναι ένα κριτήριο σύγκρισης και οι μετρήσεις πρέπει να αξιολογούνται λαμβάνοντας υπόψη τις επιδόσεις που επιτυγχάνουν οι αντίστοιχοι ταξινομητές όσον αφορά την ακρίβεια και το κόστος ταξινόμησης.

Στην παρούσα διπλωματική εργασία παρουσιάζεται ένα μη εμπορικό εργαλείο λογισμικού Java, με την ονομασία KEEL (Knowledge Extraction based on Evolutionary Learning). Το εργαλείο αυτό δίνει τη δυνατότητα στο χρήστη να αναλύσει τη συμπεριφορά των αλγορίθμων Εξόρυξης Δεδομένων, καθώς της Εξελικτικής Μάθησης, για διάφορα είδη προβλημάτων Εξόρυξης Δεδομένων όπως είναι η παλινδρόμηση, η ταξινόμηση, η μάθηση χωρίς επίβλεψη.

Το KEEL λοιπόν μπορεί να προσφέρει πολλά πλεονεκτήματα. Πρώτα απ' όλα, όπως αναφέρθηκε παραπάνω, μειώνει τις εργασίες προγραμματισμού. Περιλαμβάνει μια βιβλιοθήκη με αλγορίθμους εξελικτικής μάθησης που βασίζονται σε διαφορετικά στιγμιότυπα, απλοποιώντας την ενσωμάτωση αλγορίθμων εξελικτικής μάθησης με διαφορετικές τεχνικές προ-επεξεργασίας. Μπορεί να απαλλάξει τους ερευνητές από την

«τεχνική εργασία» του προγραμματισμού και να τους επιτρέψει να επικεντρωθούν περισσότερο στην ανάλυση νέων μοντέλων μάθησης σε σύγκριση με τα υπάρχοντα. Δεύτερον, διευρύνει το φάσμα των πιθανών χρηστών που εφαρμόζουν εξελικτικούς αλγορίθμους μάθησης. Μια εκτεταμένη βιβλιοθήκη ΕΑ σε συνδυασμό με το εύχρηστο λογισμικό, μειώνουν σημαντικά το επίπεδο γνώσεων και της εμπειρίας που απαιτείται από τους ερευνητές στον εξελικτικό υπολογισμό. Ως αποτέλεσμα, ερευνητές με λιγότερες γνώσεις, όταν χρησιμοποιούν αυτό το εργαλείο, θα μπορούν να εφαρμόζουν με επιτυχία αυτούς τους αλγορίθμους στα προβλήματά τους. Τρίτον, λόγω της χρήσης μιας αυστηρά αντικειμενοστραφούς προσέγγισης για τη βιβλιοθήκη και το εργαλείο λογισμικού, μπορεί να χρησιμοποιηθεί σε οποιαδήποτε μηχανή η οποία έχει ενσωματωμένη τη Java. Κατά συνέπεια, κάθε ερευνητής μπορεί να χρησιμοποιήσει το KEEL στο μηχάνημά του, ανεξάρτητα από το λειτουργικό σύστημα.

1.1 Κατηγοριοποίηση (Classification)

Η Κατηγοριοποίηση ή αλλιώς Ταξινόμηση είναι μια λειτουργία εξόρυξης δεδομένων η οποία αναθέτει δεδομένα μιας συλλογής σε κατηγορίες ή κλάσεις - στόχους. Ο στόχος της ταξινόμησης είναι η ακριβής πρόβλεψη της κατηγορίας - στόχου και οι τεχνικές ταξινόμησης είναι χρήσιμες στον χειρισμό μεγάλου όγκου δεδομένων. Οι αλγόριθμοι ταξινόμησης προσπαθούν να κατατάξουν νέα, μη ταξινομημένα στοιχεία δεδομένων σε ένα σύνολο προκαθορισμένων κλάσεων, με βάση τα διαθέσιμα δεδομένα εκπαίδευσης, δηλαδή ένα σύνολο ήδη ταξινομημένων στοιχείων. Η κατηγοριοποίηση μπορεί να εφαρμοστεί σε δυαδικά προβλήματα. Ένα τυπικό παράδειγμα ταξινόμησης είναι η ανάθεση ενός ηλεκτρονικού ταχυδρομείου είτε στην κλάση «spam» είτε στην κλάση «μη spam» ή κατηγοριοποίηση συναλλαγών με πιστωτικές κάρτες ως νόμιμες ή όχι.

Η κατηγοριοποίηση χωρίζεται σε δύο μέρη:

1. Μάθηση με επίβλεψη (Supervised Learning), όπου το σύστημα πρέπει να κατατάξει σε κατηγορίες ένα σύνολο δεδομένων.
2. Μάθηση χωρίς επίβλεψη (Unsupervised Learning). Εδώ τα δεδομένα δεν έχουν προκαθορισμένες ετικέτες. Το μοντέλο προσπαθεί να βρει συσχετίσεις, μοτίβα ή ομάδες μέσα στα δεδομένα.

Το σύνολο δεδομένων εισόδου διακρίνεται στο σύνολο εκπαίδευσης (training set) και στο σύνολο ελέγχου (test set). Το σύνολο εκπαίδευσης είναι υπεύθυνο για την κατασκευή του μοντέλου, ενώ το σύνολο ελέγχου για να την επικύρωσή του. Κάποιες από τις πιο γνωστές

τεχνικές κατηγοριοποίησης είναι τα Δέντρα Απόφασης (decision trees), η Κατηγοριοποίηση εγγύτερων γειτόνων (k-NN), τα Νευρωνικά Δίκτυα, ο Naïve Bayes και το Support Vector Machines (SVM). Στη συγκεκριμένη εργασία θα αναλύσουμε μόνο την k-NN τεχνική.

Οι ταξινομητές μπορούν να χωριστούν σε δύο κατηγορίες αλγορίθμων: τους eager ταξινομητές και τους instance based ταξινομητές οι οποίοι έχουν ως κύριο στόχο την ακριβή πρόβλεψη ταξινόμησης. Ωστόσο, διαφέρουν ως προς τον τρόπο λειτουργίας τους. Βασικό ρόλο για την αποτελεσματικότητα των αλγορίθμων και των δύο κατηγοριών παίζει το διαθέσιμο σύνολο εκπαίδευσης. Ένας eager ταξινομητής προ-επεξεργάζεται τα διαθέσιμα δεδομένα εκπαίδευσης και δημιουργεί ένα μοντέλο ταξινόμησης το οποίο στη συνέχεια χρησιμοποιείται για την ταξινόμηση νέων, μη ταξινομημένων στοιχείων. Από την άλλη πλευρά, οι instance based ταξινομητές δεν κατασκευάζουν κανένα μοντέλο ταξινόμησης. Στην πραγματικότητα, θεωρούν το σύνολο δεδομένων εκπαίδευσης ως μοντέλο ταξινόμησης. Ένας instance based αλγόριθμος ταξινομεί ένα νέο στοιχείο σαρώνοντας το σύνολο εκπαίδευσης τη στιγμή που αυτό φτάνει[10].

Δεδομένου ότι οι eager ταξινομητές δημιουργούν ένα μοντέλο ταξινόμησης πριν από την άφιξη οποιουδήποτε νέου στοιχείου, η διαδικασία ταξινόμησης είναι πολύ γρήγορη. Παρόλο που οι instance based ταξινομητές δεν ξοδεύουν χρόνο για να χτίσουν το μοντέλο, η διαδικασία ταξινόμησής τους είναι πιο χρονοβόρα από εκείνη των eager ταξινομητών. Ένα μειονέκτημα των eager ταξινομητών είναι ότι πρέπει να δημιουργήσουν μία μόνο υπόθεση που να καλύπτει ολόκληρο το σύνολο εκπαίδευσης. Κάτι το οποίο επηρεάζει την ακρίβεια ταξινόμησης καθιστώντας την κατασκευή του μοντέλου εξαιρετικά χρονοβόρα και περίπλοκη εργασία προ-επεξεργασίας. Από την άλλη πλευρά, οι instance based ταξινομητές χρησιμοποιούν ολόκληρο το σύνολο εκπαίδευσης και, ως εκ τούτου, μπορούν να υιοθετήσουν πιο σύνθετες υποθέσεις για τα δεδομένα, βελτιώνοντας την ακρίβεια ταξινόμησης[10]. Ένα μειονέκτημα των instance based ταξινομητών είναι ότι απαιτούν όλα τα δεδομένα εκπαίδευσης να είναι πάντα διαθέσιμα, γεγονός που οδηγεί σε υψηλές απαιτήσεις αποθήκευσης. Αντίθετα, στην eager ταξινόμηση, μετά την κατασκευή του μοντέλου ταξινόμησης, τα δεδομένα εκπαίδευσης μπορούν να αφαιρεθούν προκειμένου να εξοικονομηθεί χώρος.

1.2 Κατηγοριοποίηση Εγγύτερων Γειτόνων (k-NN)

Ο k-NN είναι ένας ευρέως γνωστός αλγόριθμος ταξινόμησης και παλινδρόμησης, επιβλεπόμενης μάθησης (Supervised Learning). Είναι απλός όσο αφορά στην εφαρμογή και χρησιμοποιείται σε διάφορους τομείς. Προκειμένου να κατηγοριοποιηθεί ένα νέο δείγμα, ο αλγόριθμος επικεντρώνεται στην απόσταση των k – εγγύτερων δεδομένων εκπαίδευσης. Ο βασικός τρόπος λειτουργίας του είναι να ταξινομεί ένα νέο στοιχείο x αναζητώντας στο σύνολο εκπαίδευσης (Training Set) τα k κοντινότερα στοιχεία (γείτονες) στο x, σύμφωνα με μια μέτρηση απόστασης (π.χ. Ευκλείδεια απόσταση). Άρα, για κάθε νέο δείγμα που προκύπτει, υπολογίζεται η απόσταση του δείγματος αυτού και όλων των υπολοίπων που υπάρχουν στο σύνολο εκπαίδευσης. Έπειτα, γίνεται ταξινόμηση των αποστάσεων και επιλογή των k πλησιέστερων γειτόνων. Στο νέο δείγμα που θα προκύψει αποδίδεται η κατηγορία εκείνη που εμφανίζεται συχνότερα μεταξύ των k πλησιέστερων γειτόνων. Προκειμένου να επιτευχθεί η μέγιστη αποτελεσματικότητα, ο καθορισμός των k εγγύτερων γειτόνων παίζει πολύ σημαντικό ρόλο.

Αν και ο ταξινομητής k-NN θεωρείται αποτελεσματική μέθοδος, έχει δύο σημαντικά μειονεκτήματα που καθιστούν δύσκολη τη χρήση του για μεγάλα σύνολα δεδομένων. Πρώτον, περιλαμβάνει υψηλό υπολογιστικό κόστος, αφού όλες οι αποστάσεις μεταξύ του νέου, μη ταξινομημένου είδους και των στοιχείων δεδομένων εκπαίδευσης πρέπει να εκτιμηθούν. Με άλλα λόγια, πρέπει να υπολογίσει όλες τις αποστάσεις μεταξύ κάθε μη ταξινομημένου στοιχείου και όλων των στοιχείων που είναι αποθηκευμένα στο σύνολο εκπαίδευσης. Όταν υπάρχουν μεγάλα σύνολα δεδομένων, το μειονέκτημα αυτό καθιστά τη χρήση του k-NN χρονοβόρα. Για παράδειγμα, ας υποθέσουμε ότι ένα σύστημα ταξινόμησης αποθηκεύει 100.000 στοιχεία εκπαίδευσης. Επιπλέον, ας υποθέσουμε ότι το σύστημα πρέπει να ταξινομήσει περίπου 50.000 μη ταξινομημένα στοιχεία εκτελώντας τον ταξινομητή k-NN πάνω στα δεδομένα εκπαίδευσης. Αυτό σημαίνει ότι το σύστημα πρέπει να υπολογίσει πέντε δισεκατομμύρια αποστάσεις[10]. Αν και σήμερα τα συστήματα είναι εξοπλισμένα με ισχυρούς επεξεργαστές, αυτοί οι υπολογισμοί είναι εξαιρετικά χρονοβόροι. Πρέπει να αναφερθεί ότι, εκτός από το μέγεθος του συνόλου εκπαίδευσης, το υπολογιστικό κόστος της εργασίας ταξινόμησης εξαρτάται επίσης από τη διάσταση των δεδομένων. Όσο μεγαλύτερη είναι, τόσο περισσότεροι είναι οι υπολογισμοί που εκτελούνται για τον υπολογισμό μιας απόστασης.

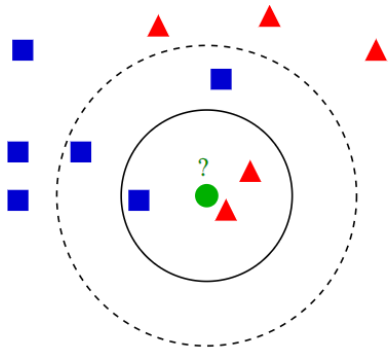
Ένα ακόμη μειονέκτημα είναι ότι οι απαιτήσεις αποθήκευσης είναι υψηλές αφού πρέπει να διατηρηθεί το Training Set. Άρα αποθηκεύει αν όχι όλα, τα περισσότερα δεδομένα του TS. Σε αντίθεση με τους άλλους ταξινομητές που μπορούν να απορρίψουν τα δεδομένα εκπαίδευσης αφού έχει δημιουργηθεί το μοντέλο ταξινόμησης, ο ταξινομητής k-NN χρειάζεται τα

δεδομένα εκπαίδευσης να είναι πάντα διαθέσιμα. Κατά συνέπεια, ο ταξινομητής k-NN πρέπει να εκτελείται σε συστήματα υπολογιστών με αρκετή κύρια μνήμη για την αποθήκευση των δεδομένων εκπαίδευσης.

Τέλος, ένα μειονέκτημα ακόμη είναι ότι ο ταξινομητής k-NN, όπως και πολλές άλλες μέθοδοι ταξινόμησης, είναι μια μέθοδος ευαίσθητη στο θόρυβο. Ειδικότερα, η ακρίβεια ταξινόμησης εξαρτάται σε μεγάλο βαθμό από την ποιότητα των δεδομένων εκπαίδευσης. Ο θόρυβος και τα λανθασμένα δεδομένα συνήθως οδηγούν σε λιγότερο ακριβή ταξινόμηση.

Αντίθετα, από τα πλεονεκτήματα του k-NN είναι ότι μπορεί να χρησιμοποιηθεί για ταξινόμηση και παλινδρόμηση καθώς επίσης είναι εύχρηστος και εξαιρετικά ακριβής. Ακολουθεί παράδειγμα εφαρμογής του k-NN αλγορίθμου (Εικόνα 1):

Ο πράσινος κύκλος με το ερωτηματικό σημαίνει ότι πρέπει να ταξινομηθεί, αποδίδοντας του μια ετικέτα. Οι δύο κατηγορίες ταξινόμησης είναι είτε το «κόκκινο τρίγωνο» είτε το «μπλε τετράγωνο». Ο k-NN λειτουργεί υποθέτωντας ότι τα στιγμιότυπα που βρίσκονται κοντά σε απόσταση είναι παρόμοια, ενώ όταν βρίσκονται σε μεγαλύτερη απόσταση ισχύει το αντίθετο. Σε περίπτωση που επιλεχθεί k = 3 τότε ο κύκλος θα ταξινομηθεί στην κατηγορία «κόκκινο τρίγωνο». Αν όμως το k=5, τότε ο κύκλος θα ταξινομηθεί στην κατηγορία «μπλε τετράγωνο».



Εικόνα 1. Εφαρμογή k-NN αλγορίθμου.

Η απόδοση της ταξινόμησης λοιπόν, εξαρτάται από την τιμή της παραμέτρου k. Η τιμή του k που επιτυγχάνει την υψηλότερη ακρίβεια ταξινόμησης εξαρτάται από το σύνολο δεδομένων που χρησιμοποιείται. Αν και ο προσδιορισμός του k δεν είναι εφικτό να ακολουθεί κάποιον γενικό κανόνα και το «καλύτερο» k μπορεί να είναι εντελώς διαφορετικό για διαφορετικά σύνολα δεδομένων, οι μεγαλύτερες τιμές k είναι κατάλληλες για σύνολα δεδομένων με θόρυβο. Ωστόσο, δεν ορίζουν ξεκάθαρα τα όρια μεταξύ διαφορετικών κλάσεων. Αντίθετα, μικρές τιμές παραμέτρων καθιστούν τον ταξινομητή περισσότερο ευαίσθητο στο θόρυβο. Επομένως, σε δεδομένα εκπαίδευσης που περιέχουν θόρυβο, η ταξινόμηση είναι πιθανώς λιγότερο ακριβής. Αξίζει να αναφέρουμε ότι ακόμη και η καλύτερη τιμή k μπορεί να μην

είναι η βέλτιστη δυνατή. Αυτό οφείλεται στο ότι ο ταξινομητής k-NN χρησιμοποιεί μια μοναδική τιμή k. Διαφορετικές τιμές k μπορεί να είναι καλύτερες για διαφορετικές περιοχές του χώρου δεδομένων. Κατά συνέπεια, μπορούν να υιοθετηθούν ευρετικές μέθοδοι για τον δυναμικό προσδιορισμό του k οι οποίες να μπορούν να επιτύχουν μεγαλύτερη ακρίβεια από τον ταξινομητή k-NN με τον «καλύτερο» προσδιορισμό της τιμής k.

Σε περιπτώσεις προβλημάτων δυαδικής ταξινόμησης (σύνολα δεδομένων με δύο κλάσεις), το k θα έπρεπε να έχει μονή τιμή ώστε να αποφεύγεται και οι δύο κλάσεις είναι οι πιο κοινές κατά την επιλογή των πλησιέστερων γειτόνων. Σε περιπτώσεις μη δυαδικών προβλημάτων, το k μπορεί να έχει οποιαδήποτε τιμή. Σε περίπτωση που υπάρξει κοινή επιλογή κλάσεων τότε το πρόβλημα αυτό επιλύεται επιλέγοντας μια τυχαία κλάση η οποία θεωρείται η πιο κοινή[11].

Ένα άλλο σημαντικό ζήτημα που προκύπτει είναι η επιλογή της μετρικής που χρησιμοποιείται για τον υπολογισμό της απόστασης μεταξύ των στοιχείων. Η επιλογή αυτή λαμβάνει υπόψη τους τύπους δεδομένων των χαρακτηριστικών (μεταβλητών) του συνόλου δεδομένων. Σε περιπτώσεις πραγματικών ή/και ακέραιων χαρακτηριστικών, η Ευκλείδεια απόσταση είναι η πιο ευρέως χρησιμοποιούμενη μετρική απόστασης. Ωστόσο, άλλες μετρικές αποστάσεων μπορούν να υιοθετηθούν π.χ. Manhatan, Minkowski, Chebyshev.

```
Είσοδος: T                                  // Σύνολο δεδομένων εκπαίδευσης
    K                                       // Αριθμός κοντινότερων γειτόνων
    t                                       // Πλειάδα προς κατηγοριοποίηση
    Έξοδος: c                               // Κλάση όπου θα κατηγοριοποιηθεί η t
    N = Ø
    Για κάθε d ∈ T επανέλαβε
    Αν |N| ≤ K τότε
     N = N ∪ {d};
     Αν ∃ u ∈ N τέτοιο ώστε dist(t,u) ≤ dist (t, d), τότε
     N = N – {u};
     N = N ∪ {d};
     Τέλος _ αν
     Τέλος _ επανάληψης
     c = κλάση όπου τα περισσότερα u ∈ N κατηγοριοποιούνται
 Τέλος αλγορίθμου
```

Πίνακας 1. Εκτέλεση k-NN αλγορίθμου

1.3 Μείωση Δεδομένων (Data Reduction)

Η Μείωση Δεδομένων (Data Reduction) είναι μια διαδικασία η οποία αποσκοπεί στη μείωση του όγκου δεδομένων, διατηρώντας τις σημαντικότερες πληροφορίες οι οποίες χρησιμοποιούνται προκειμένου να είναι όσο ακριβέστερη γίνεται η ανάλυση και κατόπιν η πρόβλεψη. Η διαδικασία αυτή είναι σημαντική για την αύξηση της αποδοτικότητας των αλγορίθμων και τη μείωση των απαιτήσεων αποθήκευσης αλλά και των υπολογιστικών πόρων. Η μείωση δεδομένων μπορεί να επιτευχθεί μέσω διαφόρων τεχνικών όπως η επιλογή χαρακτηριστικών, η εξαγωγή χαρακτηριστικών και η συμπίεση δεδομένων, τα οποία θα αναλυθούν στο 2ο κεφάλαιο της παρούσας εργασίας.

1.4 Κίνητρο και Συνεισφορά

Αφορμή για την υλοποίηση της διπλωματικής αυτής εργασίας είναι η ανάγκη που προκύπτει για μείωση του συνόλου δεδομένων, μία τεχνική που στόχο έχει την αποθήκευση των σημαντικότερων συνόλων δεδομένων ώστε να πραγματοποιείται ακριβέστερη πρόβλεψη με αποτέλεσμα να μην χρειάζεται μεγάλο υπολογιστικό κόστος.

Η παρούσα διπλωματική εργασία έχει ως στόχο τη βιβλιογραφική ανασκόπηση κάποιων αλγορίθμων. Στη συνέχεια πραγματοποιείται η πειραματική μελέτη, χρησιμοποιώντας το KEEL, με βάση τους αλγόριθμους αυτούς. Το λογισμικό KEEL προσφέρει μια ευρεία γκάμα αλγορίθμων, δίνοντας έτσι τη δυνατότητα στους χρήστες να επικεντρωθούν στην ανάλυση των δεδομένων. Οι δυνατότητες και τα εργαλεία που προσφέρει το λογισμικό αυτό συμβάλλουν στη διευκόλυνση της έρευνας.

1.5 Οργάνωση εργασίας

Η διπλωματική εργασία είναι δομημένη ως εξής:

Στο 1ο Κεφάλαιο γίνεται αναφορά στις έννοιες της Κατηγοριοποίησης, της Κατηγοριοποίησης εγγύτερων γειτόνων, και τη Μείωση δεδομένων.

Στο 2ο Κεφάλαιο αναλύονται οι Τεχνικές Μείωσης Δεδομένων και οι κατηγορίες αυτών, οι Τεχνικές επιλογής Προτύπων καθώς και οι έννοιες της Συμπύκνωσης και της Απομάκρυνσης Θορύβου.

Το 3ο Κεφάλαιο γίνεται η βιβλιογραφική ανασκόπηση των αλγορίθμων που χρησιμοποιήθηκαν προκειμένου να πραγματοποιηθεί η πειραματική μελέτη.

Το 4ο Κεφάλαιο γίνεται η παρουσίαση του λογισμικού KEEL.

Στο 5ο Κεφάλαιο υλοποιείται η πειραματική μελέτη με βάση τους αλγορίθμους που αναφέρθηκαν στο 3ο Κεφάλαιο. Επίσης, πραγματοποιείται η στατιστική ανάλυση των μετρήσεων που διενεργήθηκαν στα πλαίσια της πειραματικής μελέτης.

Τέλος, στο 6ο Κεφάλαιο παρουσιάζονται τα συμπεράσματα και οι μελλοντικές επεκτάσεις της έρευνας που πραγματοποιήθηκε.

2 *Τεχνικές Μείωσης Δεδομένων*

Οι Τεχνικές Μείωσης Δεδομένων (Data Reduction Techniques - DRT) είναι υπεύθυνες για την μείωση της ποσότητας των πληροφοριών προκειμένου να μειωθεί τόσο η μνήμη όσο και ο χρόνος εκτέλεσης. Χωρίζονται σε δύο κατηγορίες: **την μείωση των στοιχείων** (Item Reduction) και **τη μείωση των διαστάσεων** (Dimensionality Reduction).

Οι τεχνικές αυτές ομαδοποιούνται σε δύο κατηγορίες αλγορίθμων:

- Επιλογής Προτύπων (Prototype Selection)
- Αφαίρεσης Προτύπων (Prototype Abstraction)

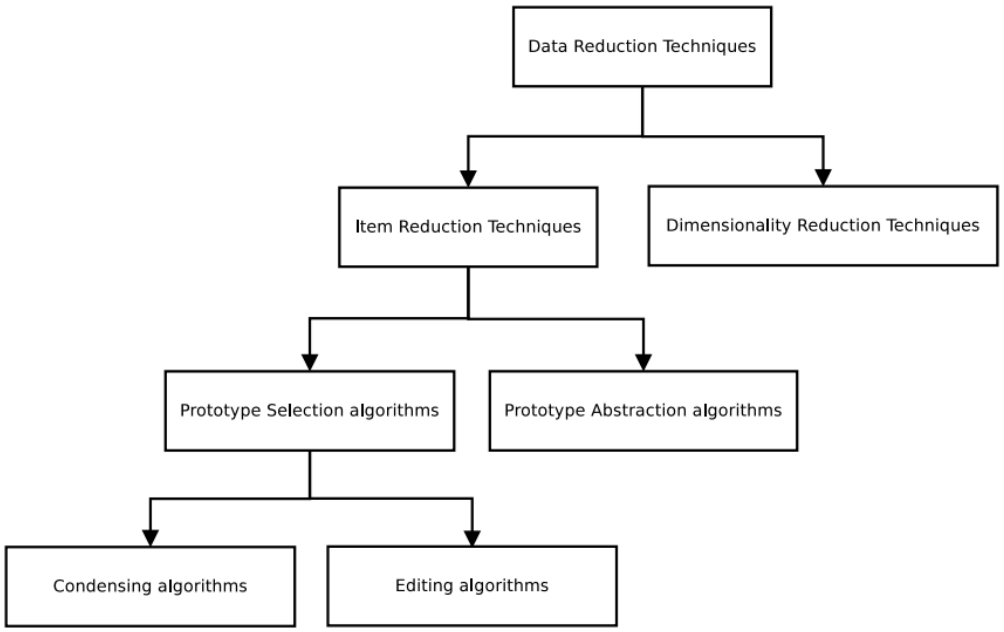
Οι αλγόριθμοι Επιλογής Προτύπων (PS) έχουν ως στόχο τη μείωση του όγκου δεδομένων, επιλέγοντας πρότυπα (ή ένα υποσύνολο) από το αρχικό σύνολο εκπαίδευσης, ενώ οι αλγόριθμοι Αφαίρεσης Προτύπων (PA) δημιουργούν πρότυπα τα οποία αντιπροσωπεύουν ομάδες παρόμοιων δεδομένων. Στην πραγματικότητα, κάθε πρότυπο αντιπροσωπεύει μια συγκεκριμένη περιοχή δεδομένων του πολυδιάστατου χώρου και καθένα από αυτά μπορεί να χρησιμοποιηθεί σε διαδικασίες όπως, για παράδειγμα, η ομαδοποίηση και η ταξινόμηση[10].

Οι Τεχνικές Μείωσης Δεδομένων αξιολογούνται μέσω κάποιων κριτηρίων. Ένα από αυτά είναι το ποσοστό μείωσης που δείχνει πόσο μικρό είναι το μέγεθος του συμπυκνωμένου συνόλου σε σχέση με το μέγεθος του αρχικού συνόλου εκπαίδευσης. Πρακτικά, είναι ο λόγος του αριθμού των στοιχείων που απορρίπτονται προς τον αριθμό των αρχικών στοιχείων του συνόλου εκπαίδευσης. Όσο πιο υψηλό είναι το ποσοστό μείωσης, τόσο ταχύτερη είναι η k-NN ταξινόμηση. Ένα άλλο σημαντικό κριτήριο είναι η ακρίβεια ταξινόμησης που επιτυγχάνει ο k-NN όταν εκτελείται πάνω στο σύνολο συμπύκνωσης. Το τελευταίο κριτήριο είναι το υπολογιστικό κόστος προ-επεξεργασίας, δηλαδή το κόστος που απαιτείται για τη δημιουργία του συμπυκνωμένου συνόλου.

Στο λογισμικό KEEL, το οποίο και είναι το κύριο θέμα της παρούσας εργασίας και θα αναλυθεί περαιτέρω στο 5^{ο} κεφάλαιο, χρησιμοποιούνται πολλές τεχνικές με σκοπό τη μείωση δεδομένων.

2.1 Κατηγορίες τεχνικών μείωσης δεδομένων

Υπάρχουν δύο είδη τεχνικών μείωσης δεδομένων: οι **Τεχνικές Μείωσης Αντικειμένων** (Item Reduction Techniques) και οι **Τεχνικές Μείωσης Διαστάσεων** (Dimensionality Reduction Techniques). Η εργασία αυτή εστιάζει στις Τεχνικές Μείωσης Αντικειμένων (Item Reduction). Όπως φαίνεται και στην Εικόνα 2, οι τεχνικές αυτές κατηγοριοποιούνται στους αλγόριθμους Επιλογής Προτύπων (Prototype Selection) και στους αλγόριθμους Παραγωγής Προτύπων (Prototype Abstraction), όπως ήδη αναφέρθηκε παραπάνω. Οι αλγόριθμοι Επιλογής Προτύπων επιλέγουν τα πιο αντιπροσωπευτικά στοιχεία από το αρχικό σύνολο εκπαίδευσης, ενώ οι αλγόριθμοι Αφαίρεσης Προτύπων δημιουργούν στοιχεία συνοψίζοντας σε παρόμοια στοιχεία εκπαίδευσης και τα χρησιμοποιούν ως πρότυπα. Στην πραγματικότητα, κάθε πρότυπο αντιπροσωπεύει μια συγκεκριμένη περιοχή δεδομένων του πολυδιάστατου χώρου[10].



Εικόνα 2. Κατηγορίες τεχνικών μείωσης δεδομένων[10]

Ωστόσο, υπάρχουν και οι Τεχνικές Μείωσης Διαστάσεων (Dimensionality Reduction Techniques) όπως, η Ανάλυση Κύριων Συνιστωσών (Principal Component Analysis - PCA) η οποία είναι μία γραμμική τεχνική μείωσης των διαστάσεων των δεδομένων με στόχο τον μετασχηματισμό των αρχικών δεδομένων σε ένα νέο σύνολο συνιστωσών. Μία ακόμη τεχνική που αξίζει να αναφερθεί είναι η Ανάλυση Διασποράς (Linear Discriminant Analysis -

LDA). Η συγκεκριμένη μέθοδος είναι και αυτή γραμμική και διαχωρίζει δύο ή και παραπάνω ομάδες δεδομένων και επικεντρώνεται κυρίως στη μεγιστοποίηση της διασποράς των δεδομένων. Στόχος είναι να βρεθούν οι άξονες που μεγιστοποιούν τη διαχωριστική ικανότητα των δεδομένων.

2.2 Τεχνικές επιλογής προτύπων (Prototype Selection Techniques)

Οι Τεχνικές Επιλογής Προτύπων (Prototype Selection Techniques) στοχεύουν στην επιλογή ενός υποσυνόλου δεδομένων το οποίο πρέπει να είναι επαρκές ώστε να διατηρήσει την πληροφορία και την απόδοση του αρχικού συνόλου δεδομένων. Αυτές οι τεχνικές παίζουν σημαντικό ρόλο στη μείωση του υπολογιστικού κόστους και την αποδοτική λειτουργία των αλγορίθμων, ειδικά σε εφαρμογές που χρησιμοποιούν τον αλγόριθμο k-NN.

Οι αλγόριθμοι επιλογής προτύπων διακρίνονται σε αλγόριθμους ***Συμπύκνωσης (Condensing)*** και αλγόριθμους ***Επεξεργασίας (Editing)***. Οι αλγόριθμοι αφαίρεσης προτύπων και συμπύκνωσης μπορούν να αντιμετωπίσουν τα μειονεκτήματα του k-NN ταξινομητή, όπως είναι το υψηλό υπολογιστικό κόστος ταξινόμησης αλλά και οι απαιτήσεις αποθήκευσης. Αυτό επιτυγχάνεται με τη δημιουργία ενός μικρότερου αντιπροσωπευτικού συνόλου των αρχικών δεδομένων εκπαίδευσης. Αυτό το σύνολο ονομάζεται σύνολο συμπύκνωσης και περιέχει μόνο τα πιο σημαντικά στοιχεία[10]. Εφαρμόζοντας τον k-NN, χρησιμοποιώντας το σύνολο συμπύκνωσης, τα πλεονεκτήματα που προκύπτουν είναι χαμηλότερο υπολογιστικό κόστος καθώς και των απαιτήσεων αποθήκευσης, ενώ η ακρίβεια παραμένει υψηλή ή δεν μειώνεται σε σημαντικό βαθμό. Από την άλλη πλευρά, οι αλγόριθμοι επεξεργασίας προσπαθούν να βελτιώσουν την ακρίβεια αφαιρώντας τα δεδομένα εκπαίδευσης που περιέχουν πληροφορίες θορύβου, εξομαλύνοντας τα όρια απόφασης μεταξύ των κλάσεων.

Η Επεξεργασία (Editing) αποσκοπεί στην βελτίωση της απόδοσης του ταξινομητή k-NN αφαιρώντας περιπτώσεις οι οποίες έχουν μεγάλη πιθανότητα να εμφανίσουν θόρυβο, αυξάνοντας τα ποσοστά μείωσης και τα επίπεδα ακρίβειας. Πιο συγκεκριμένα, τα ποσοστά μείωσης πολλών προτύπων αλγορίθμων αφαίρεσης και συμπύκνωσης εξαρτώνται από το επίπεδο θορύβου στα δεδομένα εκπαίδευσης. Τα υψηλά επίπεδα θορύβου στο σύνολο εκπαίδευσης εμποδίζουν πολλούς αλγόριθμους συμπύκνωσης ή αφαίρεσης προτύπων να επιτύχουν υψηλά ποσοστά μείωσης. Στην πραγματικότητα, όσο υψηλότερο είναι το επίπεδο θορύβου, τόσο χαμηλότερα είναι τα ποσοστά μείωσης που επιτυγχάνονται. Επομένως, η αποτελεσματική εφαρμογή αυτών των αλγορίθμων προϋποθέτει την αφαίρεση θορύβου από τα δεδομένα, δηλαδή την εκ των προτέρων εφαρμογή ενός αλγορίθμου επεξεργασίας [17]. Ως

εκ τούτου, ένας αλγόριθμος επεξεργασίας θα πρέπει να χρησιμοποιείται σε ένα σύνολο εκπαίδευσης με θόρυβο, προκειμένου είτε να βελτιωθεί η ακρίβεια είτε να καταστεί αποτελεσματικότερη η εφαρμογή των αλγορίθμων συμπύκνωσης και αφαίρεσης προτύπων. Η διαδικασία επεξεργασίας πραγματοποιείται σε περιοχές του χώρου χαρακτηριστικών με υψηλό βαθμό επικάλυψης μεταξύ των κλάσεων, παράγοντας ομαλότερα όρια κλάσεων.

Τέλος, υπάρχουν οι τεχνικές οι οποίες περιλαμβάνουν και την Επεξεργασία των ορίων των κλάσεων αλλά και τη Συμπύκνωση δεδομένων (Hybrid).

2.2.1 Συμπύκνωση (Condensation)

Η Συμπύκνωση περιλαμβάνει τεχνικές που αποσκοπούν στη διατήρηση των σημείων που βρίσκονται πιο κοντά στα όρια απόφασης, τα οποία ονομάζονται σημεία ορίων. Η λογική πίσω από τη διατήρηση των σημείων ορίων είναι ότι τα εσωτερικά σημεία δεν επηρεάζουν τα όρια απόφασης τόσο πολύ όσο τα σημεία που βρίσκονται πιο κοντά στα σύνορα και, επομένως, μπορούν να αφαιρεθούν με σχετικά μικρή επίδραση στην ταξινόμηση. Ο λόγος χρήσης λοιπόν αυτών των μεθόδων είναι η διατήρηση της ακρίβειας στο σύνολο εκπαίδευσης, αλλά η ακρίβεια γενίκευσης στο σύνολο δοκιμής (Test Set) μπορεί να επηρεαστεί αρνητικά. Παρ' όλα αυτά, η ικανότητα μείωσης των μεθόδων συμπύκνωσης είναι συνήθως υψηλή λόγω του γεγονότος ότι υπάρχουν λιγότερα σημεία συνόρων από ό,τι εσωτερικά σημεία στα περισσότερα δεδομένα[13].

Κάποια από τα πλεονεκτήματα της συμπύκνωσης είναι η μείωση υπολογιστικού φόρτου, άρα αυτό σημαίνει πως επιτυγχάνεται ταχύτερη εκπαίδευση και πρόβλεψη, η βελτίωση της απόδοσης και η ευκολία αποθήκευσης εξαιτίας της συμπίεσης των δεδομένων σε μικρότερα σύνολα. Έκτος όμως από τα πλεονεκτήματα που προσφέρει αυτή η τεχνική, η χρήση της μπορεί να επιφέρει και κάποια μειονεκτήματα όπως είναι η απώλεια πληροφορίας λόγω υπερβολικής συμπύκνωσης.

Ορισμένοι βασικοί αλγόριθμοι συμπύκνωσης είναι οι : Condensed Nearest Neighbor (CNN), Reduced Nearest Neighbor (RNN), Fast Condensed Nearest Neighbor (FCNN), Generalized Condensed Nearest Neighbor (GCNN) οι οποίοι θα αναλυθούν εκτενέστερα στο επόμενο κεφάλαιο.

2.2.2 Απομάκρυνση Θορύβου

Η απομάκρυνση θορύβου είναι μια διαδικασία που ως στόχο έχει τη βελτίωση της ποιότητας των δεδομένων και την ακρίβεια των αλγορίθμων. Ο θόρυβος μπορεί να επηρεάσει αρνητικά την απόδοση ενός μοντέλου και να οδηγήσει σε λανθασμένα αποτελέσματα. Το πρόβλημα του θορύβου έχει αντιμετωπιστεί με την ανάπτυξη αλγορίθμων που είναι ανεκτικοί στο θόρυβο χωρίς αυτό να σημαίνει ότι μπορεί να επιτευχθεί ρητά η αφαίρεση του.

Τα ποσοστά μείωσης πολλών αλγορίθμων συμπύκνωσης και αφαίρεσης προτύπων εξαρτώνται από το επίπεδο θορύβου στα δεδομένα εκπαίδευσης. Τα υψηλά επίπεδα θορύβου στο σύνολο εκπαίδευσης εμποδίζουν πολλούς αλγορίθμους αφαίρεσης προτύπων αλλά και συμπύκνωσης να επιτύχουν υψηλά ποσοστά μείωσης[10]. Στην πραγματικότητα, όπως ήδη έχει σημειωθεί, όσο υψηλότερο είναι το επίπεδο θορύβου, τόσο χαμηλότερα είναι τα ποσοστά μείωσης που επιτυγχάνονται. Επομένως, η αποτελεσματική εφαρμογή αυτών των αλγορίθμων προϋποθέτει την αφαίρεση του θορύβου από τα δεδομένα, δηλαδή την εκ των προτέρων εφαρμογή ενός αλγορίθμου επεξεργασίας. Ως εκ τούτου, ένας αλγόριθμος επεξεργασίας θα πρέπει να χρησιμοποιείται σε ένα σύνολο εκπαίδευσης με θόρυβο, προκειμένου είτε να βελτιωθεί η ακρίβεια είτε να καταστεί αποτελεσματικότερη η εφαρμογή των αλγορίθμων συμπύκνωσης και αφαίρεσης προτύπων.

2.3 Τεχνικές παραγωγής προτύπων (Prototype Generation Techniques)

Οι Τεχνικές Παραγωγής Προτύπων (Prototype Generation Methods) χρησιμοποιούνται για τη δημιουργία ενός νέου συνόλου προτύπων τα οποία αντικαθιστούν το αρχικό σύνολο εκπαίδευσης. Αυτό το νέο σύνολο αναμένεται να είναι μικρότερο από το αρχικό, καθώς τα όρια απόφασης μπορούν να οριστούν πιο αποτελεσματικά[19]. Τα εν λόγω πρότυπα χωρίζονται στις παρακάτω κατηγορίες:

- Υποσύνολα προτύπων του αρχικού συνόλου εκπαίδευσης με βάση το κεντροειδές, τα οποία ομαδοποιούνται λαμβάνοντας υπόψη κριτήρια όπως είναι η εγγύτητα, η επισήμανση και η αναπαράσταση. Στη συνέχεια, το κεντροειδές αυτού του υποσυνόλου δημιουργείται ως νέο πρωτότυπο για το τελικό σύνολο.
- Ρύθμιση θέσης από ένα αρχικό υποσύνολο του συνόλου εκπαίδευσης. Τα πρότυπα μετακινούνται γύρω από την γειτονιά τους ακολουθώντας μια συγκεκριμένη

ευρετική μέθοδο. Ο στόχος είναι να βρεθεί η θέση μέσω της οποίας θα βελτιώνεται η απόδοση των αλγορίθμων ταξινόμησης.

- Διαχωρισμός χώρου, όπου ο χώρος χωρίζεται σε υποπεριοχές και στη συνέχεια, δημιουργούνται αντιπρόσωποι κάθε χώρου.

Όπως αναφέρθηκε παραπάνω, οι τεχνικές παραγωγής προτύπων στοχεύουν στη μείωση του μεγέθους του συνόλου δεδομένων ενώ διατηρούν την πληροφορία που είναι απαραίτητη ώστε να γίνει σωστά η ταξινόμηση. Παρακάτω ακοκλουθεί η καταγραφή ορισμένων τεχνικών παραγωγής προτύπων.

I. Learning Vector Quantization (LVQ)

Η Learning Vector Quantization (LVQ) είναι μια τεχνική μάθησης με επίβλεψη που χρησιμοποιείται για την ταξινόμηση και την προσαρμογή προτύπων. Κύριο στόχο έχει τη βελτίωση της απόδοσης των ταξινομητών πλησιέστερων γειτόνων (Nearest Neighbor) με τη χρήση προτύπων τα οποία είναι αντιπροσωπευτικά των κατηγοριών των δεδομένων.

Δεδομένου ότι η LVQ προορίζεται αυστηρά ως στατιστική μέθοδος ταξινόμησης ή αναγνώρισης, ο μόνος σκοπός της είναι να καθορίσει περιοχές κλάσεων στο χώρο των δεδομένων εισόδου. Για το σκοπό αυτό, ένα υποσύνολο διανυσμάτων με παρόμοιες ετικέτες τοποθετείται σε κάθε περιοχή κλάσης, ακόμη και αν οι κατανομές κλάσεων των δειγμάτων εισόδου επικαλύπτονται στα όρια των κλάσεων[12].

Βασικά χαρακτηριστικά της LVQ είναι τα **Πρότυπα** (Prototypes) όπου κάθε κατηγορία αντιπροσωπεύεται από ένα ή περισσότερα πρότυπα τα οποία προσδιορίζονται ως σημεία στο χαρακτηριστικό χώρο. Ένα ακόμη χαρακτηριστικό είναι η **Εκπαίδευση με Εποπτεία** μέσω της οποίας η διαδικασία εκπαίδευσης της LVQ περιλαμβάνει την προσαρμογή των προτύπων ώστε να μετακινηθούν προς τα σημεία της σωστής κατηγορίας και μακριά από τα σημεία της λανθασμένης. Ένα τελευταίο χαρακτηριστικό είναι η **Ενημέρωση Προτύπων**. Αν το πρότυπο αντιπροσωπεύει σωστά ένα σημείο δεδομένων τότε αυτό μετακινείται προς αυτό το σημείο, ειδάλλως, σε περίπτωση που το πρότυπο αντιπροσωπεύει λανθασμένα ένα σημείο δεδομένων, τότε μετακινείται μακριά από αυτό.

Από την LVQ τεχνική προκύπτουν οι αλγόριθμοι LVQ1, LVQ2, LVQ3 και OLVQ1. Η τεχνική αυτή χρησιμοποιείται με σκοπό να βελτιώσει την απόδοση των ταξινομητών πλησιέστερων γειτόνων, καθιστώντας τους πιο αποδοτικούς και ακριβείς.

II. Self-generating prototypes (SGP)

Η βασική ιδέα αυτής της μεθόδου είναι να σχηματιστεί ένας αριθμός ομάδων, καθεμία από τις οποίες περιέχει ορισμένα πρότυπα της ίδιας κατηγορίας, και ο μέσος όρος κάθε ομάδας να χρησιμοποιείται ως πρότυπο για την ομάδα. Αρχικά, τα μοτίβα κάθε κλάσης σχηματίζουν μια ομάδα και ο μέσος όρος τους υπολογίζεται ως πρότυπο της αρχικής ομάδας. Στη συνέχεια διαχωρίζονται ορισμένες ομάδες διαδοχικά, μετατοπίζονται ορισμένα μοτίβα από τη μία ομάδα στην άλλη και ενδεχομένως χρειάζεται να συγχωνευτούν ορισμένες από αυτές τις ομάδες ως βήμα κλαδέματος[18]. Όλες οι λειτουργίες που εκτελούνται είναι πολύ απλές και μπορούν να ταξινομηθούν σύμφωνα με τις τέσσερις πιθανές καταστάσεις που μπορεί να προκύψουν. Αυτές περιγράφονται λεπτομερώς παρακάτω:

- Εάν για όλα τα μοτίβα μιας ομάδας το πλησιέστερο πρότυπο είναι το πρότυπο της ομάδας, τότε δεν πραγματοποιείται καμία τροποποίηση.
- Εάν για όλα τα μοτίβα μιας ομάδας το πλησιέστερο πρότυπο είναι κάποιο από λανθασμένη κλάση. Αυτό συμβαίνει συχνά όταν τα μοτίβα της ομάδας συγκεντρώνονται σε υποομάδες που χωρίζονται από μοτίβα άλλων κλάσεων, η ομάδα χωρίζεται σε δύο υποομάδες. Αυτό επιτυγχάνεται με το διαχωρισμό των σημείων με ένα υπερεπίπεδο το οποίο διέρχεται από το μέσο της αρχικής ομάδας και το οποίο είναι κάθετο στην πρώτη κύρια συνιστώσα. Εάν για ορισμένα μοτίβα μιας ομάδας το πλησιέστερο πρότυπο είναι ένα πρότυπο μιας διαφορετικής ομάδας αλλά της ίδιας κατηγορίας, τα μοτίβα αυτά μετατοπίζονται από την αρχική ομάδα στην ομάδα του πλησιέστερου αυτού προτύπου.
- Εάν για ορισμένα μοτίβα μιας ομάδας το πλησιέστερο πρότυπο είναι ένα πρότυπο μιας διαφορετικής ομάδας και μιας λανθασμένης κλάσης, τα μοτίβα αυτά αφαιρούνται από την αρχική ομάδα και σχηματίζουν μια νέα ομάδα και ο μέσος όρος της υπολογίζεται ως νέο πρότυπο.

III. Self-Generating Neural Tree (SGNT)

Ο SGNT αλγόριθμος συνδυάζει δομές δεδομένων δέντρων και νευρωνικών δικτύων προκειμένου να δημιουργήσει ένα ευέλικτο μοντέλο. Πιο συγκεκριμένα, επεκτείνεται δυναμικά προσθέτοντας ή αφαιρώντας κόμβους (νευρώνες) και κλαδιά (συνδέσεις) με βάση την πολυπλοκότητα του προβλήματος αλλά και τις απαιτήσεις του μοντέλου. Χρησιμοποιεί μια δομή δέντρου όπου κάθε κόμβος περιέχει έναν νευρώνα ή ένα μικρό νευρωνικό δίκτυο το οποίο επιτρέπει την κατασκευή μοντέλων τα οποία παρουσιάζουν πολυπλοκότητα. Ακόμα, ο τρόπος λειτουργίας του τον καθιστά κατάλληλο για εφαρμογές όπου η δομή των δεδομένων μπορεί να αλλάζει ή να απαιτεί διαφορετικά επίπεδα ανάλυσης.

Ο SGNT λειτουργεί αρχικά χρησιμοποιώντας έναν μικρό αριθμό κόμβων και στη συνέχεια, καθώς προκύπτει η ανάγκη βελτίωσης της απόδοσης του μοντέλου, προστίθενται νέοι κόμβοι με βάση τα νέα δεδομένα που έχουν προκύψει. Κάθε κόμβος περιέχει ένα μικρό νευρωνικό δίκτυο που εκπαιδεύεται με τα δεδομένα που φτάνουν σε αυτόν τον κόμβο. Οι συνδέσεις μεταξύ των κόμβων εκπαιδεύονται με στόχο την βελτιστοποίηση της συνολικής απόδοσης του μοντέλου. Η διακλάδωση νέων κόμβων προκύπτει όταν ένας κόμβος δεν μπορεί να διαχειριστεί την πολυπλοκότητα των δεδομένων. Το κλάδεμα μπορεί να συμβεί όταν ένας κόμβος δεν προσφέρει βελτίωση στην απόδοση ή όταν μειώνεται η πολυπλοκότητα των δεδομένων.

IV. Hybrid Vote-Based Reduction (HYB)

Αποτελεί μια υβριδική μέθοδο διαφόρων τεχνικών μείωσης προτύπων. Συγκεκριμένα, το HYB συνδυάζει μηχανές διανυσμάτων υποστήριξης με την LVQ3 μέθοδο και εκτελεί μια αναζήτηση για την εύρεση των καταλληλότερων παραμέτρων του LVQ3.

V. Reduction by Space Partitioning 3 (RSP3)

Ο RSP3 είναι ένας αλγόριθμος επιλογής προτύπων που χρησιμοποιείται για τη μείωση του μεγέθους του συνόλου δεδομένων. Καθορίζει αυτόματα το μέγεθος του συμπυκνούμενου συνόλου και βασίζεται κυρίως στην έννοια της ομοιογένειας των συστάδων. Αρχικά, ο RSP3 καθορίζει τις δύο πιο απομακρυσμένες περιπτώσεις στο σύνολο εκπαίδευσης και σχηματίζει δύο συστάδες αναθέτοντας κάθε περίπτωση εκπαίδευσης στην πλησιέστερη πιο απομακρυσμένη περίπτωση. Ο αλγόριθμος εφαρμόζεται αναδρομικά σε κάθε μη ομοιογενή συστάδα που δημιουργείται. Στο τέλος, κάθε ομοιογενής συστάδα αντικαθίσταται από τον κατάλληλα επισημασμένο αντιπρόσωπό της για να σχηματιστεί το σύνολο συμπύκνωσης. Το σύνολο συμπύκνωσης που δημιουργείται από το RSP3 δεν εξαρτάται από τη σειρά των παραδειγμάτων στο σύνολο εκπαίδευσης[27]. Ο RSP3 δημιουργεί ένα μικρό και ακριβές σύνολο πρωτοτύπων εκπαίδευσης. Όταν ο ταξινομητής k-NN χρησιμοποιεί την έξοδο του RSP3, αντί των αρχικών δεδομένων εκπαίδευσης, επιτυγχάνει συγκρίσιμη ακρίβεια αλλά με πολύ χαμηλότερο υπολογιστικό κόστος.

2.4 Συνδυασμός τεχνικών

Δεδομένου ότι η Επεξεργασία (Editing) χρησιμοποιείται για τον καθαρισμό των λανθασμένα επισημασμένων δειγμάτων από το σύνολο εκπαίδευσης, ο κύριος στόχος είναι η βελτίωση της ακρίβειας αναγνώρισης. Η Συμπύκνωση, ωστόσο, χρησιμοποιείται κυρίως με σκοπό τη μείωση του αριθμού των δειγμάτων. Παρόλο που προσπαθεί να ελαχιστοποιήσει τη μεταβολή της ακρίβειας αναγνώρισης, ένα ατυχές χαρακτηριστικό πολλών διαδικασιών Συμπύκνωσης είναι ότι συχνά μπορεί να οδηγήσουν σε οριακά χειρότερη απόδοση αναγνώρισης. Ως αποτέλεσμα, το ποσοστό της μείωσης του συνόλου δεδομένων, λόγω της επεξεργασίας, είναι συχνά μικρό σε σύγκριση με τις μεθόδους Συμπύκνωσης, αλλά η ακρίβεια αναγνώρισης για τα επεξεργασμένα σύνολα εκπαίδευσης είναι καλύτερη. Εξαιτίας αυτών των διακρίσεων, ενώ η επεξεργασία είναι συχνά προτιμότερη, η συμπύκνωση έχει τη μεγαλύτερη πρακτική σημασία για την ανάπτυξη ενός συστήματος αναγνώρισης προτύπων.

Με τη συνδυαστική χρήση των τεχνικών αυτών, η βελτιωμένη αναγνώριση που προκαλείται από την Επεξεργασία μπορεί να συνδυαστεί με τη μεγαλύτερη μείωση που παρέχεται από τα εργαλεία Συμπύκνωσης για να παραχθεί ένα σύνολο εκπαίδευσης που είναι σημαντικά μικρότερο από το αρχικό με παρόμοιες ή καλύτερες δυνατότητες αναγνώρισης. Αυτή η σύνθετη προσέγγιση βελτιώνει την απόδοση τόσο από υπολογιστική άποψη όσο και από άποψη αναγνώρισης.

3 *Παρουσίαση Αλγορίθμων*

Στο παρόν κεφάλαιο έχουν επιλεγεί για ανάλυση και μελέτη κάποιοι αλγόριθμοι επιλογής προτύπων. Γίνεται αναφορά τόσο στον τρόπο λειτουργίας τους όσο και στα προτερήματα και τα μειονεκτήματα που μπορεί να έχει ο καθένας από αυτούς. Συγκεκριμένα, οι αλγόριθμοι που θα παρουσιαστούν, με την σειρά που αναφέρονται παρακάτω, είναι οι εξής:

1. Condensed Nearest Neighbor [3]
2. Reduced Nearest Neighbor [7]
3. Fast Condensed Nearest Neighbor [9]
4. Generalized Condensed Nearest Neighbor [6]
5. Instanced Based Learning 3 [16]
6. Selective Nearest Neighbor [22]
7. Modified Condensed Nearest Neighbor [15]
8. Iterative Case Filtering [16]
9. Decremental Reduction Optimization Procedure 3 [26]

3.1 Condensed Nearest Neighbor (CNN) [3]

Ο Condensed Nearest Neighbor[3] (CNN) συνδέεται με τον k-NN αλγόριθμο και είναι μια μέθοδος υπο-δειγματοληψίας (under-sampling) η οποία αποθηκεύει τα στοιχεία ένα προς ένα και στη συνέχεια εξαλείφει όσα εμφανίζονται δύο ή περισσότερες φορές. Ως εκ τούτου, ο CNN αφαιρεί τα στοιχεία εκείνα που δεν προσθέτουν περισσότερες πληροφορίες και δείχνουν ομοιότητα με άλλα σετ δεδομένων εκπαίδευσης. Τα στοιχεία που παραμένουν χρησιμοποιούνται για την εκπαίδευση του συστήματος. Στην ουσία ο CNN εστιάζει στην εύρεση ενός συνεπούς υποσυνόλου προτύπων εκπαίδευσης.

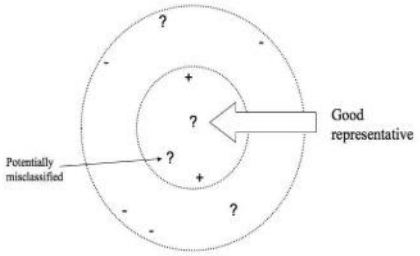
Η εύρεση ενός υποσυνόλου επισημασμένων σημείων δεδομένων μπορεί να οδηγήσει σε ταχύτερη και ακριβέστερη ταξινόμηση. Η εύρεση όμως του καλύτερου υποσυνόλου, δεν παύει να αποτελεί ένα δυσεπίλυτο πρόβλημα[11]. Ο CNN μπορεί να θεωρηθεί ως μια απλή τεχνική για την προσέγγιση ενός τέτοιου υποσυνόλου επισημασμένων σημείων δεδομένων.

Ο αλγόριθμος CNN, ορίζεται στον Πίνακα 2, με το T να είναι το σύνολο των επισημασμένων σημείων δεδομένων και το T(t) προβλέπεται για t από έναν ταξινομητή πλησιέστερου γείτονα «εκπαιδευμένο» στο T[4].

$T = \{<x_1, y_1>, \ldots, <x_n, y_n>\}$, $C = \emptyset$
for $<x_i, y_i> \in T$ do
    if $C(xi) 6= y_i$ then
        $C = C \cup \{<x_1, y_1>\}$
  end if
end for
return C

Πίνακας 2. Αλγόριθμος Condensed Nearest Neighbor[4]

Ο CNN προσπαθεί να βρει τους καλύτερους δυνατούς αντιπροσώπους για κάθε συστάδα στην διανομή των δεδομένων. Με άλλα λόγια, προσπαθεί να βρει τα σημεία που βρίσκονται πιο κοντά στο κέντρο κάθε συστάδας. Ιδανικά, ο CNN επιστρέφει ένα σημείο για κάθε συστάδα, το κέντρο δηλαδή κάθε συστάδας. Ωστόσο, ένα δείγμα δεδομένων με ετικέτα ενδέχεται να μην περιλαμβάνει σημεία δεδομένων που βρίσκονται κοντά στο κέντρο μιας συστάδας. Κατά συνέπεια, ο CNN ορισμένες φορές χρειάζεται πολλά σημεία προκειμένου να σταθεροποιήσει την αναπαράσταση μιας συστάδας π.χ. τα δύο θετικά όπως φαίνονται στην Εικόνα 3.



Εικόνα 3. Αναπαράσταση συστάδας. Τα δεδομένα χωρίς ετικέτα βοηθούν στην εύρεση καλύτερων εκπροσώπων σε condensed training sets[11]

Πιο συγκεκριμένα, ο CNN χρησιμοποιεί δύο περιοχές αποθήκευσης, το Condensing set (CS) και το Training set (TS). Για λόγους συντομίας θα γίνεται αναφορά σε αυτά τα δύο με τα αρχικά τους. Αρχικά, το TS περιέχει ολόκληρο το σετ εκπαίδευσης και το CS είναι άδειο. Επιλέγεται με τυχαίο τρόπο ένα στιγμιότυπο από το TS και μεταφέρεται στο CS για να ξεκινήσει η διαδικασία. Στη συνέχεια, κάθε στιγμιότυπο $x \in$ TS συγκρίνεται με τα στιγμιότυπα που είναι αποθηκευμένα στο CS. Για κάθε στιγμιότυπο $x \in$ TS, ο CNN βρίσκει τον πλησιέστερο γείτονά του στο τρέχον CS, χρησιμοποιώντας την Ευκλείδεια απόσταση. Εάν το x έχει ταξινομηθεί σωστά από τον πλησιέστερο γείτονά του στο CS, διατηρείται στο TS, διαφορετικά, το x αφαιρείται από το TS και προστίθεται στο CS. Αφού ληφθούν υπόψη όλα τα $x \in$ TS, η διαδικασία συνεχίζεται μέχρι μια επόμενη σάρωση του TS. Ο αλγόριθμος σταματά όταν όλα τα στιγμιότυπα του TS ταξινομούνται σωστά από το περιεχόμενο του CS. Με άλλα λόγια, ο CNN τερματίζεται όταν κανένα στιγμιότυπο δεν μεταφέρεται από το TS στο CS κατά τη διάρκεια μιας πλήρους σάρωσης του TS[5].

Ένα από τα πιο σημαντικά πλεονεκτήματα του αλγόριθμου αυτού είναι η επιτάχυνση της διαδικασίας πρόβλεψης αλλά και της εξοικονόμησης χώρου αποθήκευσης λόγω της μείωσης του μεγέθους του εκπαιδευτικού συνόλου. Η μείωση αυτή επιτυγχάνεται χωρίς να υπάρχει κάποια σημαντική απώλεια στην ακρίβεια της ταξινόμησης. Μερικά από τα μειονεκτήματά του όμως, είναι πως ο αλγόριθμος εξετάζει πολλές φορές το TS το οποίο έχει ως αποτέλεσμα το υψηλό υπολογιστικό κόστος, καθώς επίσης, όλα τα στιγμιότυπα πρέπει να βρίσκονται στη μνήμη καθώς ο αλγόριθμος βασίζεται σε αυτή.

3.2 Reduced Nearest Neighbor (RNN) [7]

Ο Reduced Nearest Neighbor[7] (RNN) αποτελεί επέκταση του Condensed Nearest Neighbor ο οποίος όμως χρησιμοποιεί μόνο ένα υποσύνολο δειγμάτων από το TS. Αυτό το υποσύνολο, όταν χρησιμοποιείται ως αποθηκευμένο σύνολο αναφοράς, ταξινομεί σωστά όλα τα δείγματα που ανήκουν στο αρχικό TS. Εν ολίγοις, ο αλγόριθμος μειώνει το μέγεθος του TS αφαιρώντας τα δείγματα εκείνα που είναι περιττά και δεν επηρεάζουν την ακρίβεια της ταξινόμησης[7].

Πιο συγκεκριμένα, ο RNN λειτουργεί ως εξής:

1. Ο αλγόριθμος ξεκινά με το πλήρες εκπαιδευτικό σύνολο δεδομένων.
2. Στη συνέχεια, αφαιρείται προσωρινά ένα δείγμα (π.χ. το 1ο) από το σύνολο δεδομένων.
3. Αν το σύνολο δεδομένων που έχει απομείνει, χωρίς το δείγμα που αφαιρέθηκε στο προηγούμενο βήμα, μπορεί να ταξινομήσει σωστά όλα τα υπολειπόμενα δείγματα, με

τη βοήθεια του k-NN αλγορίθμου, τότε το συγκεκριμένο δείγμα αφαιρείται μόνιμα από το σύνολο δεδομένων.

4. Στην περίπτωση όμως που η συγκεκριμένη αφαίρεση προκαλέσει προβλήματα στην ταξινόμηση, τότε το δείγμα πρέπει να επανατοποθετηθεί στο σύνολο δεδομένων.
5. Η διαδικασία αυτή επαναλαμβάνεται ωσότου να μην είναι απαραίτητο να αφαιρεθούν άλλα δείγματα και κατά συνέπεια να μην μειώνεται η ακρίβεια της ταξινόμησης.

Ένα από τα πλεονεκτήματα που διαθέτει ο RNN είναι η μείωση του μεγέθους του εκπαιδευτικού συνόλου με αποτέλεσμα να επιτυγχάνεται ταχύτερα η πρόβλεψη[25]. Επίσης, επιτυγχάνεται η μείωση του συνόλου δεδομένων χωρίς να υπάρχουν σημαντικές απώλειες στην ακρίβεια της ταξινόμησης.

Ο RNN όμως διαθέτει και κάποια μειονεκτήματα. Ένα από αυτά είναι η υπολογιστική απαίτηση. Η επαναλαμβανόμενη διαδικασία αφαίρεσης είναι υπολογιστικά ακριβή. Ακόμα, εξαιτίας της ευαισθησίας που παρουσιάζει ο αλγόριθμος στον θόρυβο, δεν αποδίδει κατάλληλα σε δεδομένα που παρουσιάζουν θόρυβο.

3.3 Fast Condensed Nearest Neighbor (FCNN) [9]

Ο Fast Condensed Nearest Neighbor[9] (FCNN) είναι ένας αλγόριθμος με μεγάλα πολυδιάστατα σύνολα δεδομένων τα οποία χρησιμοποιούνται για τη δημιουργία υποσυνόλων που εξυπηρετούν ως σετ εκπαίδευσης, με βάση τον κανόνα k-NN.

Ο FCNN ακολουθεί την εξής προσέγγιση: Αρχικοποιεί ένα υποσύνολο δεδομένων με ένα αρχικό στοιχείο από κάθε ετικέτα κλάσης του συνόλου εκπαίδευσης. Στη συνέχεια, για κάθε επανάληψη, εξετάζει το κάθε δείγμα από το αρχικό σύνολο δεδομένων. Σε κάθε επανάληψη το σύνολο που προκύπτει, αυξάνεται μέχρι να επιτευχθεί η κατάσταση διακοπής. Οπότε, η διαδικασία επαναλαμβάνεται έως ότου κανένα δείγμα του συνόλου δεδομένων να μην μπορεί να ταξινομηθεί λανθασμένα από το υποσύνολο. Η παραπάνω περιγραφή φαίνεται στον πίνακα που ακολουθεί (Πίνακας 3)[9].

```
Algorithm FCNN rule
Input: A training set consistent subset S of T;
Method:
    S = ∅;
    ΔS = Centroids (T);
    while (ΔS ≠ 0) {
    S = S ∪ ΔS;
    ΔS = ∅;
    for each (p ∈ S)
    ΔS = ΔS ∪ {rep(p, Voren(p,S,T))};
 }
 return (S);
```

Πίνακας 3. Ο FCNN αλγόριθμος[9]

Ο αλγόριθμος είναι σταδιακά αυξανόμενος: κατά τη διάρκεια κάθε αλληλεπίδρασης το σύνολο S αυξάνεται μέχρι να επιτευχθεί η συνθήκη διακοπής. Για κάθε στοιχείο του S, ένα αντιπροσωπευτικό στοιχείο του Voren(p,S,T), που συμβολίζεται ως rep(p, Voren(p,S,T)) σύμφωνα με τον Πίνακα 3, εισάγεται και επιλέγεται στο S.

Κάποιες από τις ιδιότητες που προκύπτουν από τον αλγόριθμο, όπως περιγράφεται στον Πίνακα 2 είναι:

1. Το S είναι ένα υποσύνολο εκπαίδευσης του T, για τον NN, εάν για κάθε στοιχείο p του S, το Voren(p,S,T) είναι κενό.
2. Ο FCNN ολοκληρώνεται σε πεπερασμένο χρόνο, υπολογίζει το υποσύνολο ενός συνόλου εκπαίδευσης και δεν εξαρτάται από τη σειρά επιλογής των δειγμάτων.

Ορισμένα από τα χαρακτηριστικά που ξεχωρίζουν τον αλγόριθμο αυτό είναι η απλότητά του, απαιτεί λίγες επαναλήψεις για να συγκλίνει και είναι πιθανό να επιλέξει τα σημεία που βρίσκονται πολύ κοντά στο όριο απόφασης.

3.4 Generalized Condensed Nearest Neighbor (GCNN) [6]

O Generalized Condensed Nearest Neighbor[6] (GCNN) λειτουργεί επιλέγοντας ένα αντιπροσωπευτικό υποσύνολο δειγμάτων από το αρχικό σύνολο δεδομένων, το οποίο διατηρεί τις ιδιότητες ταξινόμησης του πλήρους συνόλου. Ένα χαρακτηριστικό του συγκεκριμένου αλγορίθμου είναι ότι επιτρέπει την επιλογή μιας ποικιλίας παραδειγμάτων, ακόμα και εκείνων που δεν μπορούν να ταξινομηθούν σωστά από το τρέχον σύνολο, αλλά και άλλων που συμβάλλουν στη διατήρηση της γενίκευσης του μοντέλου.

Σύμφωνα με το CNN, ένα δείγμα x απορροφάται εάν

$$\| x - q \| - \| x - p \| > 0 \text{ , (1)}$$

όπου p και q είναι πρότυπα, p είναι το πλησιέστερο ομοιογενές πρότυπο στο x και q είναι το πλησιέστερο ετερογενές πρότυπο στο x. Για τον GCNN, ωστόσο, υιοθετείται το ακόλουθο κριτήριο[6]:

$$\rho\delta \text{ n } \| x - q \| - \| x - p \| > \text{ , }) \ \rho \in [0,1], \text{ (2)}$$

Ένα δείγμα είναι ασθενώς απορροφημένο αν ικανοποιεί την (1) και ισχυρά απορροφημένο αν ικανοποιεί την (2). Να σημειωθεί ότι η (1) αντιστοιχεί στην περίπτωση όπου ρ = 0 στο (2). Η υιοθέτηση της (2) καθιστά δυνατή τη βελτίωση του ταξινομητή βελτιστοποιώντας το ρ.

Τα βήματα που ακολουθούνται κατά τη διάρκεια εκτέλεσης του GCNN είναι:

i. G1 Έναρξη: Για κάθε ετικέτα y, επίλεξε ένα δείγμα y ως αρχικό y-πρότυπο.

ii. G2 Έλεγχος απορρόφησης: Έλεγξε αν κάθε δείγμα είναι απορροφημένο. Εάν όλα τα δείγματα έχουν απορροφηθεί, τερμάτισε τη διαδικασία διαφορετικά, προχώρα στο επόμενο βήμα.

iii. G3 Ενίσχυση πρωτοτύπου: Για κάθε y, εάν υπάρχουν μη απορροφημένα δείγματα y, επίλεξε ένα ως νέο y-πρότυπο- διαφορετικά, δεν προστίθεται νέο πρότυπο στην ετικέτα y. Επέστρεψε στο G2 για να συνεχίσεις.

Στο G1, ένα δείγμα y επιλέγεται ως εξής. Αφήνουμε κάθε y-δείγμα να ρίξει μια ψήφο στο πλησιέστερο y-δείγμα και επιλέγουμε αυτό που λαμβάνει τον μεγαλύτερο αριθμό ψήφων. Στο G3, ένα μη απορροφημένο δείγμα y επιλέγεται ως εξής: Έστω Ψy = {xi: l(xi)=y & xi είναι μη απορροφημένο. Αφήνουμε κάθε μέλος του Ψy να δώσει μια ψήφο για το πλησιέστερο μέλος σε αυτό το σύνολο. Το επιλεγμένο δείγμα y είναι το μέλος του Ψy που λαμβάνει τον μεγαλύτερο αριθμό ψήφων[24].

Ένα από τα πλεονεκτήματα του GCNN είναι, προφανώς, η μείωση της ποσότητας δεδομένων με αποτέλεσμα να μειώνεται η ανάγκη για αποθηκευτικό χώρο. Ακόμα, βελτιώνεται η ακρίβεια της ταξινόμησης καθώς και η αφαίρεση δεδομένων που προκαλούν θόρυβο.

3.5 Instanced Based Learning 3 (IBL3) [16]

Στην παρούσα υποενότητα, περιγράφεται μια μεθοδολογία, που ονομάζεται Μάθηση Βασισμένη σε Περιπτώσεις (Instance Based Learning – IBL3)[16], η οποία παράγει προβλέψεις ταξινόμησης χρησιμοποιώντας μόνο συγκεκριμένες περιπτώσεις. Οι αλγόριθμοι μάθησης που βασίζονται σε περιπτώσεις, δεν διατηρούν ένα σύνολο από αφαιρέσεις που προέρχονται από συγκεκριμένες περιπτώσεις. Η προσέγγιση αυτή επεκτείνει τον αλγόριθμο του πλησιέστερου γείτονα, ο οποίος έχει μεγάλες απαιτήσεις αποθήκευσης.

Η μάθηση με βάση τις περιπτώσεις, λοιπόν, βασίζεται στην άμεση εφαρμογή της παραδοχής της ομοιότητας. Στην απλούστερη περίπτωση, η μάθηση πραγματοποιείται με την αποθήκευση όλων των παρατηρούμενων παραδειγμάτων. Ένα νέο παράδειγμα ταξινομείται με την εύρεση του πλησιέστερου αποθηκευμένου παραδείγματος, σύμφωνα με κάποια συνάρτηση ομοιότητας και την ανάθεση της κλάσης του τελευταίου στο πρώτο. Τα αποθηκευμένα παραδείγματα που χρησιμοποιούνται για την ταξινόμηση νέων περιπτώσεων αναφέρονται ως παραδείγματα ή υποδείγματα. Η απόδοση του IBL εξαρτάται καθοριστικά από τη μετρική ομοιότητας (ή, αντίστροφα, απόστασης) που χρησιμοποιείται[21].

Οι αλγόριθμοι IBL υποθέτουν ότι παρόμοιες περιπτώσεις έχουν παρόμοιες ταξινομήσεις. Επίσης, υποθέτουν ότι, χωρίς προηγούμενη γνώση, τα χαρακτηριστικά θα έχουν την ίδια σημασία για τις αποφάσεις ταξινόμησης.

Οι αλγόριθμοι IBL διαφέρουν από τις περισσότερες άλλες μεθόδους μάθησης με επίβλεψη: δεν κατασκευάζουν δέντρα αποφάσεων ή κανόνες, αντλούν γενικεύσεις από τις περιπτώσεις όταν αυτές παρουσιάζονται και χρησιμοποιούν απλές διαδικασίες αντιστοίχισης για να ταξινομήσουν τις περιπτώσεις[21]. Ο φόρτος εργασίας τους είναι υψηλότερος όταν παρουσιάζονται επόμενες περιπτώσεις για ταξινόμηση, οπότε υπολογίζουν τις ομοιότητες των αποθηκευμένων παραδειγμάτων τους σύμφωνα με το νέο παράδειγμα.

Υπάρχουν τρεις διαφορετικές εκδοχές αυτού του αλγορίθμου: ο IBL1, ο IBL2 και ο IBL3, στον οποίο επικεντρώνεται η συγκεκριμένη υποενότητα.

Ο IBL3, που περιγράφεται στον Πίνακα 4, είναι μια επέκταση του IBL2 ο οποίος χρησιμοποιεί μια μέθοδο συλλογής αποδείξεων «αναμονής» για να καθορίσει ποιες από τις αποθηκευμένες περιπτώσεις αναμένεται να αποδώσουν καλά κατά την ταξινόμηση. Η συνάρτηση ομοιότητας του IBL3 είναι πανομοιότυπη με εκείνη του IBL2. Η συνάρτηση ταξινόμησης και ο αλγόριθμος ενημέρωσης διαφέρουν ως εξής[16]:

1. Ο IBL3 διατηρεί ένα αρχείο ταξινόμησης (δηλ. τον αριθμό των σωστών και λανθασμένων προσπαθειών ταξινόμησης) με κάθε αποθηκευμένο παράδειγμα. Ένα αρχείο ταξινόμησης συνοψίζει την απόδοση ταξινόμησης μιας περίπτωσης σε μεταγενέστερα παρουσιαζόμενα παραδείγματα εκπαίδευσης και υποδεικνύει πως θα αποδώσει στο μέλλον.

2. Ο IBL3 χρησιμοποιεί ένα τεστ σημαντικότητας για να καθορίσει ποιες περιπτώσεις είναι καλοί ταξινομητές και ποιες θεωρεί ότι είναι θορυβώδεις.

Σύμφωνα με τον Πίνακα 4, για κάθε παράδειγμα εκπαίδευσης t, ενημερώνονται τα αρχεία ταξινόμησης για όλα τα αποθηκευμένα παραδείγματα που είναι τουλάχιστον τόσο παρόμοια όσο ο πιο παρόμοιος αποδεκτός γείτονας του t. Εάν κανένα από τα αποθηκευμένα παραδείγματα δεν είναι αποδεκτό, χρησιμοποιείται μια πολιτική που προσομοιώνει τη συμπεριφορά του αλγορίθμου όταν τουλάχιστον ένα παράδειγμα είναι αποδεκτό. Εάν κανένα δεν είναι αποδεκτό, παράγεται ένας τυχαίος αριθμός r από το εύρος [1, n], όπου n είναι ο αριθμός των αποθηκευμένων περιπτώσεων, και ενημερώνονται τα αρχεία ταξινόμησης των r πιο παρόμοιων αποθηκευμένων περιπτώσεων[21].

Ο IBL3 αποδέχεται μια περίπτωση εάν η ακρίβεια ταξινόμησής της είναι σημαντικά μεγαλύτερη από την παρατηρούμενη συχνότητα της κλάσης της και αφαιρεί την περίπτωση από την περιγραφή της έννοιας εάν η ακρίβειά της είναι σημαντικά μικρότερη. Οι αλγόριθμοι IBL που χρησιμοποιούν αυτό το τεστ σημαντικότητας έχουν καλή απόδοση σε πολλές εφαρμογές.

CD,←Ø

for each x in Training Set **do**

1. **for each** y E CD **do**

   Sim [y] ← Similarity (x, y)

2. **if** ∃ {y E CD | acceptable(y)}

   then Y_{max} ← some acceptable y E Cd with maximal Sim[y]

   else

   2.1 i ← a randomly-selected value in [1, |CD|]

   2.2 Y_{max} ← some y E CD that is the i-th most similar instance to x

3. **if** class(x) ≠ class (Y_{max})

   then classification ← **correct**

   else

   3.1 classification ← **incorrect**

   3.2 CD ← CD ∪ {x}

4. **for each** y in CD **do**

   if Sim[y] ≥ Sim [Y_{max}]

then

4.1 Update y's classification record

4.2 if y's record is significantly poor

then CD ← C - {y}

Πίνακας 4. Αλγόριθμος IBL3[21]

Ο IBL3 λοιπόν, είναι ένας δυναμικός αλγόριθμος μάθησης με βάση τα στιγμιότυπα που αποθηκεύει και στη συνέχεια ενημερώνει τα στιγμιότυπα αυτά κατά τη διάρκεια της εκπαίδευσης, αφαιρώντας παράλληλα θορυβώδη δεδομένα. Αυτό τον καθιστά ιδανικό για εφαρμογές όπου τα δεδομένα μπορούν να αλλάζουν με την πάροδο του χρόνου και απαιτείται συνεχής προσαρμογή.

3.6 Selective Nearest Neighbor (SNN) [22]

Ο αλγόριθμος Selective Nearest Neighbor[22] (SNN) είναι μια παραλλαγή του αλγορίθμου k-Nearest Neighbor (k-NN) και έχει ως κύριο στόχο την αναζήτηση παραδείγματος με τη μεγαλύτερη χρησιμότητα, με απώτερο σκοπό τη βελτίωση της απόδοσης και την ακρίβεια της ταξινόμησης, επιλέγοντας τους πιο αξιόπιστους γείτονες. Αυτό επιτυγχάνεται μέσω της αναγνώρισης και απομάκρυνσης θορύβου καθώς και των μη αντιπροσωπευτικών δεδομένων κατά τη διάρκεια της διαδικασίας της εκπαίδευσης[22]. Ο υπολογισμός της αναμενόμενης χρησιμότητας ενός παραδείγματος, απαιτεί την εκτίμηση της πιθανότητας των πιθανών ετικετών του.

Τρία κριτήρια που χρησιμεύουν ως βάση για το σύνολο των Selective Nearest Neighbors:

i. το υποσύνολο πρέπει να είναι ένα συνεπές υποσύνολο,
ii. όλα τα δείγματα πρέπει να είναι πιο κοντά (και πιο παρόμοια) σε έναν επιλεκτικό γείτονα της ίδιας κλάσης παρά σε οποιοδήποτε δείγμα της άλλης κλάσης,
iii. δεν πρέπει να υπάρχει υποσύνολο που να ικανοποιεί τα κριτήρια 1 και 2 και να περιέχει λιγότερα μέλη από το επιλεκτικό υποσύνολο.

Σύμφωνα με τον Πίνακα 5, έστω X ένα σύνολο αντικειμένων που περιγράφονται από μια πεπερασμένη συλλογή χαρακτηριστικών. Ένας αλγόριθμος μάθησης λαμβάνει ένα σύνολο παραδειγμάτων, {x1, f (x1),...,xn, f(xn)}, και επιστρέφει μια υπόθεση h: $X \rightarrow \{0, 1\}$. Έστω X ένα μη επισημασμένο σύνολο εκπαίδευσης, ένα σύνολο αντικειμένων που αντλούνται τυχαία από το X σύμφωνα με την κατανομή p. Έστω $D = \{xi, f (xi) : xi \in X, i = 1,..., n\}$ τα δεδομένα εκπαίδευσης - ένα σύνολο επισημασμένων παραδειγμάτων από το X. Ένας αλγόριθμος επιλεκτικής δειγματοληψίας, που καθορίζεται σε σχέση με έναν αλγόριθμο μάθησης L, λαμβάνει το X και το D ως είσοδο και επιστρέφει ένα μη επισημασμένο στοιχείο του X. Αυτό έρχεται σε αντίθεση με το κοινό, μη επαυξητικό, πλαίσιο μάθησης, όπου οι αλγόριθμοι λαμβάνουν μόνο ως είσοδο το D.

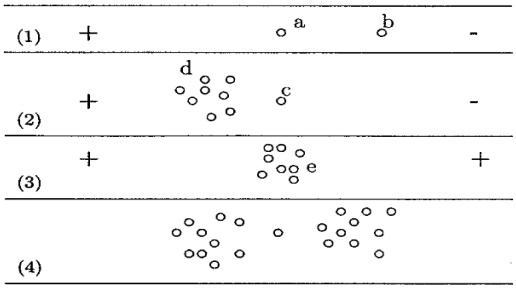
Η διαδικασία μάθησης με επιλεκτική δειγματοληψία, λοιπόν, μπορεί να περιγραφεί ως μια επαναληπτική διαδικασία όπου σε κάθε επανάληψη καλείται η διαδικασία επιλεκτικής δειγματοληψίας για να ληφθεί ένα μη επισημασμένο παράδειγμα και ο Active learner

καλείται να επισημάνει το παράδειγμα αυτό. Το επισημασμένο παράδειγμα προστίθεται στο σύνολο των διαθέσιμων επισημασμένων παραδειγμάτων και το ενημερωμένο σύνολο δίνεται στη διαδικασία μάθησης, η οποία δημιουργεί έναν νέο ταξινομητή. Αυτή η ακολουθία επαναλαμβάνεται έως ότου ικανοποιηθεί κάποιο κριτήριο διακοπής. Υιοθετώντας το πρώτο κριτήριο διακοπής, ο στόχος του αλγορίθμου επιλεκτικής δειγματοληψίας είναι να παράγει μια ακολουθία μήκους M η οποία θα οδηγεί σε έναν καλύτερο ταξινομητή σύμφωνα με κάποιο δεδομένο[23].

Active Learner (X, f ()):
1. $D \leftarrow \emptyset$.
2. $h \leftarrow L(\emptyset)$.
3. While stopping-criterion is not satisfied do:
   a) $x \leftarrow SL(X, D)$; Apply SL and get the next example.
   b) $\omega \leftarrow f(x)$; Ask the teacher to label x.
   c) $D \leftarrow D \cup \{x, \omega\}$; Update the labeled examples set.
   d) $h \leftarrow L(D)$
4. Return classifier *h*.

Πίνακας 5. Αλγόριθμος Selective Nearest Neighbor[23]

Σύμφωνα με την Εικόνα 4 που ακολουθεί παρακάτω, η οριακή δειγματοληψία θα επιλέξει τη δειγματοληψία του σημείου a έναντι του σημείου b στη διαμόρφωση (1), αφού το σημείο b φαίνεται να είναι λιγότερο σημαντικό λόγω της εγγύτητάς του στο επισημασμένο σημείο. Στη διαμόρφωση (2) η μέθοδος των ορίων θα κάνει δειγματοληψία στο σύνορο (σημείο c). Εάν το σημείο c επισημανθεί ως «-», η αβεβαιότητα που σχετίζεται με την ταξινόμηση της συστάδας d θα παραμείνει υψηλή. Από την άλλη πλευρά, η δειγματοληψία σε οποιοδήποτε σημείο της συστάδας d θα αποδώσει μια ταξινόμηση με υψηλή βεβαιότητα για όλα τα σημεία της. Συνεπώς, στη διαμόρφωση (2), η δειγματοληψία στη συστάδα d είναι προτιμότερη από τη δειγματοληψία στο σημείο c. Αυτά τα δύο παραδείγματα φανερώνουν ότι οι αλγόριθμοι επιλεκτικής δειγματοληψίας θα πρέπει να λαμβάνουν υπόψη όχι μόνο την αβεβαιότητα του υποψήφιου σημείου δείγματος, αλλά και την επίδραση της ταξινόμησής του στα υπόλοιπα μη επισημασμένα σημεία. Έτσι, η δειγματοληψία σε πυκνές περιοχές μπορεί να προτιμάται από τη δειγματοληψία σε ένα απομονωμένο σημείο. Ακόμη και όταν μια συμπαγής ομάδα μη επισημασμένων σημείων περιβάλλεται από περιπτώσεις της ίδιας ταξινόμησης, όπως απεικονίζεται στη διαμόρφωση (3), είναι λογικό να πραγματοποιείται η δειγματοληψία σε αυτήν, επειδή η ετικέτα που προκύπτει επηρεάζει πολλά σημεία.



Εικόνα 4. Δειγματοληψία σημείων. (1) Ένα σημείο κοντά στα σύνορα. (2) Ομάδα σημείων κοντά στα σύνορα. (3) Συμπαγής ομάδα σημείων μακριά από επισημασμένα παραδείγματα. (4) Δύο γειτονικές ομάδες σημείων, πολύ μακριά από επισημασμένα παραδείγματα[22].

3.7 Modified Condensed Nearest Neighbor (MCNN) [15]

Σε οποιοδήποτε πρόβλημα, τα όρια κάθε κλάσης είναι πολύ δύσκολο να προσδιοριστούν. Στη μέθοδο η οποία θα αναλυθεί παρακάτω, γίνεται μια προσπάθεια να χωριστεί η περιοχή μιας κλάσης σε απλούστερες περιοχές. Αυτό γίνεται σταδιακά, προσθέτοντας πρότυπα σε ένα αντιπροσωπευτικό σύνολο προτύπων, έως ότου όλα τα πρότυπα εκπαίδευσης ταξινομηθούν σωστά χρησιμοποιώντας αυτό το σύνολο αντιπροσωπευτικών προτύπων. Σε αυτό το στάδιο, η περιοχή που αφορά κάθε κλάση έχει αναλυθεί σε προσεγγιστικές περιοχές Voronoi.

$$\mathbf{R_j = \cup V_{ji},\ j = 1\ldots., c}$$

όπου n είναι ο αριθμός των περιοχών της κατηγορίας j και c είναι ο συνολικός αριθμός των κλάσεων.

Στον MCNN[15] αλγόριθμο, λαμβάνεται υπόψιν ένα σύνολο προτύπων με σταδιακό τρόπο. Ξεκινάει με ένα βασικό σύνολο προτύπων το οποίο περιλαμβάνει ένα πρότυπο από κάθε κλάση. Το σύνολο εκπαίδευσης ταξινομείται χρησιμοποιώντας αυτά τα πρότυπα. Καθορίζεται ένα αντιπροσωπευτικό πρότυπο για κάθε κλάση και προστίθεται στο σύνολο των βασικών προτύπων, με βάση τα λανθασμένα ταξινομημένα δείγματα. Στη συνέχεια, το σύνολο εκπαίδευσης ταξινομείται ξανά με το επαυξημένο σύνολο προτύπων. Τα αντιπροσωπευτικά πρότυπα για κάθε κλάση προσδιορίζονται και πάλι με βάση τα λανθασμένα δείγματα. Αυτή η διαδικασία επαναλαμβάνεται μέχρι να ταξινομηθούν σωστά όλα τα πρότυπα στο σύνολο εκπαίδευσης. Ο προσδιορισμός του αντιπροσωπευτικού προτύπου για κάθε κλάση για τα λανθασμένα ταξινομημένα δείγματα γίνεται, επίσης, με επαναληπτικό τρόπο.

Η μέθοδος που χρησιμοποιείται για την εύρεση ενός μόνο αντιπροσωπευτικού δείγματος μιας ομάδας προτύπων εξαρτάται από το σύνολο δεδομένων που χρησιμοποιείται. Μια απλή

μέθοδος είναι η χρήση του κεντροειδούς ως αντιπροσωπευτικού δείγματος της ομάδας μοτίβων.

Σε κάθε επανάληψη του αλγορίθμου, πρέπει να βρούμε ένα σύνολο αντιπροσωπευτικών δειγμάτων για μια ομάδα προτύπων. Αυτό πραγματοποιείται με την εύρεση ενός δείγματος που αντιπροσωπεύει όλα τα δείγματα σε κάθε κλάση. Μια μέθοδος εύρεσης ενός μόνο αντιπροσωπευτικού δείγματος μιας ομάδας προτύπων είναι η χρήση του μέσου όρου του δείγματος ή κεντροειδές όλων των προτύπων. Ο μέσος όρος δείγματος ή το κεντροειδές M μιας συλλογής μοτίβων X_1, X_2, X_r δίνεται από τη σχέση,

$$M = \sum_{i=1}^{r} \frac{Xi}{r},$$

όπου M είναι ένα διάνυσμα που περιέχει f στοιχεία και όπου f είναι ο αριθμός των χαρακτηριστικών. Ο μέσος όρος του δείγματος δίνει το διάνυσμα κατεύθυνσης και το μοτίβο σε αυτή την κλάση, που είναι πλησιέστερο στο M και επιλέγεται ως αντιπροσωπευτικό δείγμα[15].

Ενώ αυτό μπορεί να λειτουργήσει καλά στην περίπτωση των σημείων που έχουν κατανομή Gauss, σε περιπτώσεις όπου το σχήμα της συστάδας είναι ομόκεντρο, οι διάφορες κατηγορίες μπορεί να έχουν πανομοιότυπους δειγματικούς μέσους όρους.

1. T = {$T_{ij}/j = 1, ..., c, i=1, ..., n_j$}, όπου n_j = το νούμερο των δειγμάτων στην κλάση j.

2. t = 0.

3. Let $\mathbf{S_1}$ =**T.**

4. t = t + 1.

5. $\mathbf{S_t}$ = {$X_{ij}/j = 1, ..., c, i=1, ..., n1_j$}, όπου $n1_j$ = το νούμερο των δειγμάτων της κλάσης j στο $\mathbf{S_t}$.

6. $\forall_j$, j=1,…..c, C_{tj} = .

/* Αυτό το βήμα βρίσκει το κεντροειδές των λανθασμένα ταξινομημένων δειγμάτων*/

7. $\forall_j$, j=1, …..,c, $P_{tj} = X_{kj}$

8. Let $S_r = \emptyset$.

Let $S_m = \emptyset$.

9. $\forall X_{ij}$, j ∈ $\mathbf{S_t}$

if d (X_{ij}, P_{tj}) < d (X_{ij}, P_{tk}) k ≠ 0

$S_r = S_r \cup \{X_{ij}\}$

else if ∃ k d (X_{ij}, P_{tj}) < d (X_{ij}, P_{tk}) k ≠ 0

$S_m = S_m \cup \{X_{ij}\}$ /* Εάν το πρότυπο ταξινομείται εσφαλμένα, προσθέστε το στο S_m.

Εάν ταξινομείται σωστά, προσθέστε το στο S_r./*

10. Set $S_m = S_r$

11. if $S_m \neq \emptyset$ Πήγαινε στο βήμα 4

/* Επανέλαβε τα βήματα 4-10 μέχρι να μην υπάρχουν λανθασμένα δείγματα/*

12. $Q = Q \cup \{P_{ti,}\ i=1, ……., c\}$

/* Προσθήκη αντιπροσωπευτικών προτύπων του P στο συνολικό σύνολο προτύπων Q */

13. Let Q = {Q_{ij}/i = 1,……$n2_j$, j=1,……,c}.

14. Let $S_r = \emptyset$.

Let $S_m = \emptyset$.

15. $\forall X_{ij}$, ∈ T

If ∃ Q_{kj}

d (X_{ij}, Q_{tj}) ≤ d (X_{ij}, Q_{pq}),

∀q = 1,.....c, q ≠ j

and ∀p = 1, ..., $n2_q$,

then $S_r = S_r \cup \{X_{ij}\}$

else $S_m = S_m \cup \{X_{ij}\}$ /* Εάν το πρότυπο ταξινομείται εσφαλμένα, προσθέστε το στο S_m.

Εάν ταξινομείται σωστά, προσθέστε το στο S_r./*

16. Set $S_t = S_m$

17. If $S_m = \emptyset$ Σταμάτα αλλιώς πήγαινε στο βήμα 4.

/* εάν κανένα δείγμα δεν έχει ταξινομηθεί εσφαλμένα, σταματά και επέστρεψε το Q ως το επιλεγμένο σύνολο προτύπων /*

Πίνακας 6. Περιγραφή αλγορίθμου MCNN[15]

Σε αυτόν τον αλγόριθμο τα σύμβολα που χρησιμοποιούνται είναι:

Q = σύνολο που περιέχει τα πρότυπα σε κάθε στάδιο. Τα πρότυπα συνεχίζουν να προστίθενται στο Q σταδιακά. Μετά τον τερματισμό του αλγορίθμου, το Q περιέχει το πλήρες σύνολο των προτύπων που έχουν επιλεγεί.

T = σύνολο εκπαίδευσης.

S_r = σύνολο σωστά ταξινομημένων δειγμάτων.

S_m = σύνολο λανθασμένα ταξινομημένων δειγμάτων.

c = αριθμός κλάσεων.

Σύμφωνα με τον Πίνακα 6, τα βήματα 4-10 επαναλαμβάνονται μέχρι να υπάρξουν αντιπροσωπευτικά δείγματα για μια ομάδα δειγμάτων που είναι υποσύνολο των λανθασμένα ταξινομημένων δειγμάτων. Καθώς λαμβάνονται τα αντιπροσωπευτικά δείγματα σε κάθε στάδιο, προστίθενται στο υπάρχον σύνολο προτύπων και ολόκληρο το σύνολο των προτύπων εκπαίδευσης ταξινομείται χρησιμοποιώντας αυτά τα πρότυπα. Η διαδικασία διακόπτεται όταν όλα τα πρότυπα του συνόλου εκπαίδευσης ταξινομηθούν σωστά. Καθώς η παραπάνω διαδικασία εκτελείται στο σύνολο εκπαίδευσης, ο αριθμός των λανθασμένα ταξινομημένων δειγμάτων μειώνεται συνεχώς, έως ότου τελικά όλα τα δείγματα ταξινομηθούν σωστά. Έπειτα από αυτή τη διαδικασία ο αλγόριθμος δίνει το σύνολο των προτύπων που ταξινομεί σωστά όλα τα πρότυπα εκπαίδευσης. Ο αριθμός των προτύπων στο σύνολο κάθε κατηγορίας μπορεί να μην είναι ίσος. Σε οποιαδήποτε επανάληψη, εάν όλα τα πρότυπα μιας συγκεκριμένης κλάσης ταξινομηθούν σωστά, τότε στην επόμενη επανάληψη δεν θα προστεθεί κανένα αντιπροσωπευτικό πρότυπο για την εν λόγω κλάση. Καθώς προχωρούν οι επαναλήψεις, ο αριθμός των κλάσεων στις οποίες προστίθεται αντιπροσωπευτικό δείγμα συνεχίζει να μειώνεται.

Παρακάτω ακολουθεί μια καταγραφή των βασικών ιδιοτήτων του αλγορίθμου MCNN οι οποίες τον καθιστούν πολύ χρήσιμο.

Ιδιότητα 1. *Ο αλγόριθμος MCNN συγκλίνει σε ένα πεπερασμένο χρόνο.*

Απόδειξη. Στα βήματα 4-12 του αλγορίθμου MCNN, επιλέγεται τουλάχιστον ένα πρότυπο για να προστεθεί στο σύνολο Q. Ακόμη και αν επιλέγεται ένα πρότυπο σε κάθε επανάληψη, θα χρειαστούν το πολύ n επαναλήψεις για να συμπεριληφθούν όλα τα δείγματα του συνόλου εκπαίδευσης στο σύνολο Q. Στην πραγματικότητα, δεδομένου ότι c πρότυπα προστίθενται στο πρώτο πέρασμα μέσω του αλγορίθμου, ακόμη και αν ένα δείγμα συμπεριληφθεί από το σύνολο εκπαίδευσης στις υπόλοιπες επαναλήψεις, απαιτούνται το πολύ n – c + 1 επαναλήψεις. Αν ληφθεί υπόψη ότι υπάρχουν πεπερασμένα δείγματα στο σύνολο

εκπαίδευσης, το n είναι πεπερασμένο. Συνεπώς, ο αλγόριθμος MCNN συγκλίνει σε άπειρο χρόνο.

Ιδιότητα 2. *Το σύνολο Q των προτύπων που παράγονται από τον αλγόριθμο MCNN παρέχει 100% ακρίβεια στο σύνολο εκπαίδευσης.*

Απόδειξη. Τα πρότυπα P_{ij} που προστίθενται στο σύνολο Q, στο βήμα 12, είναι όλα τα δείγματα του συνόλου εκπαίδευσης, δηλαδή $P_{ij} \in \mathbf{T}$. Στο βήμα 17, το κριτήριο τερματισμού είναι ότι όλα τα $X \in T$, ταξινομούνται σωστά. Στη χειρότερη περίπτωση, εάν όλα τα $X_{ij} \in \mathbf{T}$ προστεθούν στο σύνολο Q, όλα τα δείγματα στο σύνολο εκπαίδευσης θα ταξινομηθούν σωστά. Συνεπώς, ο αλγόριθμος MCNN θα σταματήσει όταν επιτευχθεί 100% ακρίβεια στο σύνολο εκπαίδευσης.

Ιδιότητα 3. *Ο αλγόριθμος MCNN είναι ανεξάρτητος από τη σειρά.*

Απόδειξη. Σε αυτόν τον αλγόριθμο, εκτελούνται επανειλημμένα τρεις λειτουργίες:

1. Εύρεση του κεντροειδούς ή του μέσου όρου ενός συνόλου σημείων (βήμα 6).
2. Εύρεση του πλησιέστερου δείγματος στο κεντροειδές (βήμα 7).
3. Χρήση ταξινομητή πλησιέστερων γειτόνων για την εύρεση των δειγμάτων σε ένα σύνολο δεδομένων που ταξινομούνται σωστά και εκείνων που ταξινομούνται εσφαλμένα (βήματα 9 και 15).

3.8 Iterative Case Filtering (ICF) [16]

Ο αλγόριθμος Iterative Case Filtering (ICF)[16] αποσκοπεί στη μείωση του συνόλου δεδομένων μέσω μιας επαναληπτικής διαδικασίας, διατηρώντας μόνο τα πιο σημαντικά δείγματα που είναι απαραίτητα για την ταξινόμηση. Στην ουσία, ο συγκεκριμένος αλγόριθμος προσδιορίζει τις περιπτώσεις (instances) που πρέπει να διαγραφούν και στη συνέχεια, αφού αφαιρεθούν, ο αλγόριθμος ‘’τρέχει’’ ξανά μέχρις ότου καμία άλλη περίπτωση δεν πληροί τις προϋποθέσεις του κανόνα. Πιο συγκεκριμένα, ο κανόνας αφαίρεσης λειτουργεί ως εξής: αφαιρούμε τις περιπτώσεις που έχουν μέγεθος προσβάσιμου συνόλου μεγαλύτερο από το μέγεθος του συνόλου κάλυψης. Μια πιο διευκρινιστική ανάγνωση αυτού του κανόνα είναι ότι μια περίπτωση c αφαιρείται όταν περισσότερες περιπτώσεις μπορούν να λύσουν την c από ό,τι η c μπορεί να λύσει τον εαυτό της. Αυτές οι περιπτώσεις θα είναι εκείνες που απέχουν περισσότερο από τα όρια της κλάσης, καθώς τα προσιτά σύνολα τους θα είναι μεγάλα.

Ο αλγόριθμος προχωρά υπολογίζοντας επανειλημμένα αυτές τις ιδιότητες μετά το φιλτράρισμα. Συνήθως, επιπλέον περιπτώσεις αρχίζουν να πληρούν τα κριτήρια καθώς προχωρά η αραίωση και οι ζώνες που περιβάλλουν τα όρια των κλάσεων στενεύουν. Μετά από μερικές επαναλήψεις αφαίρεσης περιπτώσεων και εκ νέου υπολογισμού, το κριτήριο δεν ισχύει πλέον. Αυτό το σημείο αποδεικνύεται ένα πολύ καλό σημείο για να σταματήσει η αφαίρεση περιπτώσεων. Παρακάτω περιγράφεται ο τρόπος εκτέλεσης του αλγορίθμου βήμα προς βήμα[16]:

1. S = X, Ξεκινάει με το σύνολο εκπαίδευσης S να είναι ίδιο με το αρχικό σύνολο X.
2. R = {} (Κενό σύνολο), Για κάθε δείγμα στο S ελέγχει αν μπορεί να ταξινομηθεί σωστά από τους k πλησιέστερους γείτονες στο υπόλοιπο S (δηλαδή, S χωρίς το δείγμα). Αν το δείγμα ταξινομηθεί σωστά, θεωρείται περιττό και προστίθεται στο σύνολο R.
3. Αν υπάρχουν δείγματα στο R, αφαιρούνται από το S. Η διαδικασία επαναλαμβάνεται μέχρι να μην υπάρχουν δείγματα που να μπορούν να αφαιρεθούν από το S.

Ο ICF λοιπόν, στοχεύει στη μείωση του όγκου δεδομένων που είναι απαραίτητα για την ταξινόμηση, βελτιώνοντας παράλληλα την αποδοτικότητα, χωρίς να μειώνεται σε μεγάλο βαθμό η ακρίβεια της ταξινόμησης. Ο αλγόριθμος αυτός είναι ιδιαίτερα χρήσιμος σε περιπτώσεις όπου το αρχικό σύνολο δεδομένων είναι μεγάλο και η ταχύτητα ταξινόμησης είναι κρίσιμη.

3.9 Decremental Reduction Optimization Procedure 3 (DROP3) [26]

Σε αυτήν την υποενότητα παρουσιάζεται ένας αλγόριθμος, μείωσης στιγμιοτύπων (instances) ο οποίος παρέχει ανοχή στο θόρυβο, υψηλή ακρίβεια γενίκευσης και σημαντική μείωση του αποθηκευτικού χώρου, η οποία με τη σειρά της βελτιώνει την ταχύτητα γενίκευσης. Ο DROP3[26] χρησιμοποιείται προκειμένου να αποφασιστεί ποιες περιπτώσεις πρέπει να διατηρηθούν και ποιες να αφαιρεθούν από ένα σύνολο εκπαίδευσης. Να σημειωθεί ότι υπάρχουν περισσότερες από μία εκδοχές του συγκεκριμένου αλγορίθμου (DROP1, DROP2, DROP4). Παρόλα αυτά η παρούσα υποενότητα εστιάζεται μόνο στον DROP3.

Ένα σημείο, λοιπόν, το οποίο εμφανίζει θόρυβο στο κέντρο μιας συστάδας, προκαλεί πολλά σημεία σε αυτή τη συστάδα να θεωρούνται σημεία συνόρων, ακόμη και μετά την αφαίρεση του θορυβώδους σημείου[26]. Συνεπώς, ο DROP3 χρησιμοποιεί ένα πέρασμα φιλτραρίσματος θορύβου πριν από την ταξινόμηση των περιπτώσεων. Αυτό αφαιρεί θορυβώδη στιγμιότυπα, καθώς και κοντινά σημεία συνόρων. Μετά την αφαίρεση των θορυβωδών περιπτώσεων, οι περιπτώσεις αυτές ταξινομούνται με βάση την απόσταση από τον πλησιέστερο ‘’εχθρό’’ τους που παραμένει στο μειωμένο σύνολο δεδομένων, και έτσι αφαιρούνται πρώτα τα σημεία που βρίσκονται μακριά από το πραγματικό όριο απόφασης[25]. Αυτό επιτρέπει στα εσωτερικά σημεία των συστάδων να αφαιρούνται νωρίς κατά τη διάρκεια της διαδικασίας, ακόμη και αν υπήρχαν θορυβώδη σημεία κοντά.

Και εδώ ο αλγόριθμος αυτός, όπως και ο προηγούμενος, ξεκινά με το σύνολο εκπαίδευσης να είναι ίσο με το αρχικό. Στη συνέχεια, ελέγχει αν κάθε δείγμα ταξινομείται σωστά από τους k πλησιέστερους γείτονές του στο αρχικό σύνολο. Αν ταξινομείται λανθασμένα, αφαιρείται από το μειωμένο σύνολο. Αν κατά την αφαίρεση ενός δείγματος δεν μειωθεί η ακρίβεια, τότε το δείγμα αυτό δεν επανατοποθετείται στο σύνολο εκπαίδευσης. Αν όμως επηρεαστεί η ακρίβεια, τότε επανεισάγεται στο σύνολο.

Ένα μειονέκτημα του DROP3 είναι ότι σε σπάνιες περιπτώσεις μπορεί να αφαιρέσει πάρα πολλά δείγματα κατά τη διάρκεια μείωσης θορύβου.

Σε γενικές γραμμές λοιπόν, ο DROP3 διατηρεί την ακρίβεια αφαιρώντας παράλληλα τα περιττά δείγματα τα οποία όχι μόνο δε συμβάλλουν σε καλύτερο ποσοστό ακρίβειας αλλά την επιβαρύνουν.

4 *Το λογισμικό KEEL*

Το λογισμικό KEEL (Knowledge Extraction based on Evolutionary Learning) αρχικά αναπτύχθηκε ως ένα εργαλείο το οποίο επικεντρώνεται κυρίως στην εφαρμογή εξελικτικών αλγορίθμων αλλά και σε ευφυείς τεχνικές των προβλημάτων εξόρυξης δεδομένων όπως είναι η παλινδρόμηση (regression), η ταξινόμηση (classification) και η προ-επεξεργασία δεδομένων (data pre-processing).

Το KEEL κυκλοφόρησε πρώτη φορά το 2009 και το 2011 αναβαθμίστηκε σε Java. Η τελευταία έκδοση κυκλοφόρησε το 2018. Παρέχει ένα απλό GUI για τη σχεδίαση πειραμάτων με διαφορετικά σύνολα δεδομένων και αλγόριθμους υπολογιστικής νοημοσύνης με σκοπό την αξιολόγηση της συμπεριφοράς των αλγορίθμων. Επιπλέον, έχει εφαρμογή σε δύο πεδία: το ερευνητικό και το εκπαιδευτικό.

Στην Εικόνα 5 παρατίθεται η αρχική οθόνη του KEEL η οποία περιλαμβάνει τις παρακάτω λειτουργίες[2]:

- **Διαχείριση Δεδομένων (Data Management)**: Αυτή η λειτουργία αποτελείται από ένα σύνολο εργαλείων που μπορούν να χρησιμοποιηθούν για τη δημιουργία νέων δεδομένων, για την εξαγωγή και την εισαγωγή δεδομένων σε διαφορετικές μορφές, και για την ανάλυση και την οπτικοποίηση δεδομένων.
- **Σχεδιασμός πειραμάτων (Experiments, off-line λειτουργία):** Η λειτουργία αυτή επιτρέπει στους χρήστες να εκτελούν πειράματα για την αξιολόγηση και τη σύγκριση διαφορετικών αλγορίθμων μηχανικής μάθησης. Η βιβλιοθήκη του KEEL περιλαμβάνει αλγόριθμους για ταξινόμηση, παλινδρόμηση, συσταδοποίηση και εξόρυξη κανόνων.
- **Εκπαιδευτικά Πειράματα (Educational, on-line λειτουργία):** Έχοντας παρόμοιο τρόπο λειτουργίας με την προηγούμενη, η επιλογή αυτή επιτρέπει το σχεδιασμό ενός πειράματος που μπορεί να εκτελεστεί βήμα - βήμα προκειμένου να εφαρμοστεί η διαδικασία μάθησης ενός συγκεκριμένου μοντέλου, χρησιμοποιώντας το εργαλείο λογισμικού για εκπαιδευτικούς σκοπούς.

- **Ενότητες (Modules):** Αυτό το μέρος περιλαμβάνει κάποιες υποενότητες οι οποίες επεκτείνουν τις λειτουργίες του KEEL:
    - ✓ Imbalanced learning: Μια ενότητα ειδικά σχεδιασμένη για τη δημιουργία πειραμάτων σε Imbalanced δεδομένα.
    - ✓ Non - Parametric Statistical Analysis: Αυτή η ενότητα επιτρέπει την εκτέλεση πολλών μη παραμετρικών στατιστικών δοκιμών σε ένα σύνολο αποτελεσμάτων.
    - ✓ Semi – Supervised Learning.
    - ✓ Multiple Instance Learning: Ο πίνακας επιλογής συνόλων δεδομένων εμφανίζει τα διαθέσιμα σύνολα δεδομένων για το τρέχον πείραμα.

Κάποια από τα πλεονεκτήματα του KEEL είναι ότι μειώνει το φόρτο στο προγραμματιστικό κομμάτι με αποτέλεσμα οι χρήστες να επικεντρώνονται στην ανάλυση των αλγορίθμων και των αποτελεσμάτων τους. Επίσης, λόγω των βιβλιοθηκών και των διάφορων εργαλείων που προσφέρει, το εύρος των χρηστών που μπορούν να το αξιοποιήσουν είναι μεγάλο. Δηλαδή, ακόμα και χρήστες με λιγότερες γνώσεις μπορούν να πειραματιστούν στους αλγόριθμους που αυτοί επιθυμούν. Εξαιτίας αυτού του χαρακτηριστικού, οποιοσδήποτε μπορεί να χρησιμοποιήσει το συγκεκριμένο λογισμικό για προσωπική χρήση εγκαθιστώντας απλά την Java.

Το KEEL λοιπόν, μπορεί να είναι χρήσιμο από διαφορετικούς τύπους χρηστών, καθένας από τους οποίους μπορεί να βρει συγκεκριμένα χαρακτηριστικά σε ένα λογισμικό Εξόρυξης Δεδομένων. Στις επόμενες υποενότητες περιγράφεται αναλυτικά το προφίλ των χρηστών για τους οποίους προορίζεται το KEEL, τα κύρια χαρακτηριστικά του καθώς και οι διαφορετικές λειτουργίες που έχει ενσωματωμένες.

Εικόνα 5. Αρχική οθόνη του KEEL [2]

4.1 Προφίλ χρηστών

Το KEEL, όπως αναφέρθηκε παραπάνω, χρησιμοποιείται κυρίως από δύο κατηγορίες χρηστών:

- Τους ερευνητές και
- Τους φοιτητές

Η πιο κοινή χρήση του KEEL για έναν ερευνητή είναι η αυτοματοποίηση των πειραμάτων και η στατιστική των αποτελεσμάτων. Συνήθως, ένας πειραματικός σχεδιασμός περιλαμβάνει έναν συνδυασμό εξελικτικών αλγορίθμων, στατιστικών και τεχνικών που σχετίζονται με την τεχνητή νοημοσύνη. Δεδομένου ότι τα τρέχοντα πρότυπα μηχανικής μάθησης απαιτούν ισχυρή υπολογιστική ισχύ, το KEEL, ως εργαλείο έρευνας, δεν έχει σχεδιαστεί με σκοπό να παρέχει την προβολή προόδου των αλγορίθμων σε πραγματικό χρόνο. Στόχο έχει τη δημιουργία ενός script το οποίο θα εκτελείται σε μια ομάδα υπολογιστών. Ακόμα, το KEEL επιτρέπει στον ερευνητή να εφαρμόσει την ίδια σειρά προ-επεξεργασίας, πειραμάτων και ανάλυσης σε ένα πλήθος προβλημάτων και να εστιάσει την προσοχή του στη σύνοψη των αποτελεσμάτων.

Από την άλλη πλευρά, το KEEL, ως εκπαιδευτικό εργαλείο, χρησιμοποιείται με εντελώς διαφορετικό τρόπο από έναν μαθητή. Στην περίπτωση αυτή, δεν υπάρχει η ανάγκη επανάληψης ενός πειράματος διότι ο χρόνος εκτέλεσης του λογισμικού κατά τη διάρκεια του μαθήματος πρέπει να είναι πολύ πιο σύντομος. Άρα, ο χρόνος εκτέλεσης είναι μικρότερος και η εξέλιξη των εξελικτικών αλγορίθμων γίνεται σε πραγματικό χρόνο με αποτέλεσμα ο μαθητής με αυτόν τον τρόπο να εκπαιδεύεται στην προσαρμογή των παραμέτρων για τον εκάστοτε αλγόριθμο. Η εκπαιδευτική έκδοση του KEEL, λοιπόν, είναι μια απλοποιημένη έκδοσή του και μόνο κάποιοι συγκεκριμένοι αλγόριθμοι είναι διαθέσιμοι. Η εκτέλεση πραγματοποιείται σε πραγματικό χρόνο και ο χρήστης έχει οπτική ανατροφοδότηση της προόδου των αλγορίθμων, έχοντας παράλληλα πρόσβαση στα τελικά αποτελέσματα από το ίδιο interface που χρησιμοποιήθηκε εξ αρχής για το σχεδιασμό του πειράματος.

4.2 Ανάλυση κύριων χαρακτηριστικών KEEL

Όπως έχει ήδη αναφερθεί, το KEEL είναι ένα εργαλείο λογισμικού που αναπτύχθηκε με σκοπό να συνδυάζει και να χρησιμοποιεί διαφορετικά μοντέλα Εξόρυξης Δεδομένων (Data Mining). Αξίζει να σημειωθεί ότι είναι η πρώτη εργαλειοθήκη λογισμικού αυτού του τύπου που περιέχει μια βιβλιοθήκη αλγορίθμων εξελικτικής μάθησης με ανοιχτό κώδικα σε Java. Τα κύρια χαρακτηριστικά του KEEL είναι:

- Οι Εξελικτικοί Αλγόριθμοι (Evolutionary Algorithms) παρουσιάζονται σε μοντέλα πρόβλεψης, προ-επεξεργασίας (εξελικτικό χαρακτηριστικό και επιλογή στιγμιότυπου) και μετα-επεξεργασίας (εξελικτικός συντονισμός ασαφών κανόνων).
- Αλγόριθμοι προ-επεξεργασίας δεδομένων: μετασχηματισμός δεδομένων, διακριτοποίηση, επιλογή στιγμιότυπου και επιλογή χαρακτηριστικών.
- Περιέχει μια στατιστική βιβλιοθήκη για την ανάλυση των αποτελεσμάτων των αλγορίθμων η οποία περιλαμβάνει ένα σύνολο στατιστικών δοκιμών για την ανάλυση των αποτελεσμάτων, καθώς επίσης πραγματοποιεί παραμετρικές και μη παραμετρικές συγκρίσεις των αλγορίθμων.
- Περιέχει, επίσης, ένα φιλικό ως προς το χρήστη interface, βασισμένο στην ανάλυση αλγορίθμων.
- Το λογισμικό έχει σχεδιαστεί για πειράματα που περιέχουν πολλαπλά σύνολα δεδομένων και αλγόριθμους που συνδέονται μεταξύ τους προκειμένου να επιτύχουν το επιθυμητό αποτέλεσμα.
- Περιέχει μια βιβλιοθήκη αλγορίθμων εξαγωγής γνώσης η οποία ενσωματώνει πολλαπλούς αλγόριθμους εξελικτικής μάθησης σε συνδυασμό με τις κλασικές προσεγγίσεις μάθησης. Οι κύριες τεχνικές που περιλαμβάνονται είναι τα Μοντέλα μάθησης εξελικτικών κανόνων, τα Ασαφή συστήματα, τα Εξελικτικά νευρωνικά δίκτυα, ο Γενετικός προγραμματισμός (Εξελικτικοί αλγόριθμοι που χρησιμοποιούν αναπαραστάσεις δέντρων για εξαγωγή γνώσης) και η Μείωση δεδομένων.

4.3 Διαχείριση Δεδομένων (Data Management)

Η υποενότητα Διαχείρισης Δεδομένων (Data Management) δίνει τη δυνατότητα στον χρήστη να προετοιμάσει τα δεδομένα μέσω ένος συνόλου επιλογών οι οποίες τον διευκολύνουν στην εισαγωγή, την προ-επεξεργασία, την ανάλυση και διαχείριση δεδομένων. Η συγκεκριμένη λειτουργία, λοιπόν, είναι σημαντική για την προετοιμασία των δεδομένων, πριν την εφαρμογή αλγορίθμων μηχανικής μάθησης, γιατί έτσι διασφαλίζεται η ακρίβεια των αποτελεσμάτων.

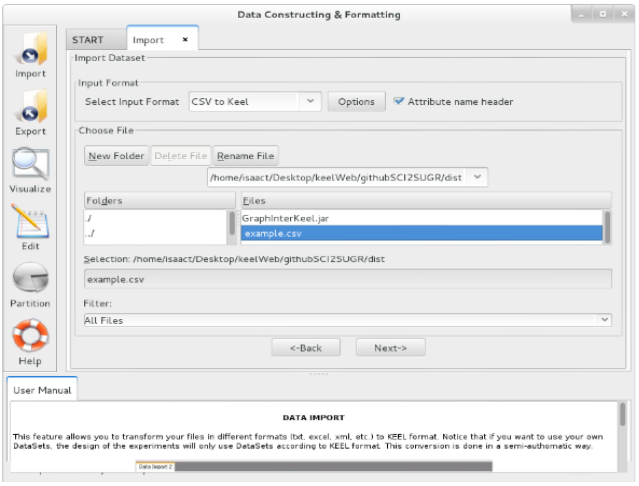
Κάποιες από τις κύριες λειτουργίες της υποενότητας αυτής είναι η Εισαγωγή Δεδομένων, η Προ-επεξεργασία Δεδομένων, η Διαχείριση Συνόλου Δεδομένων, η Ανάλυση Δεδομένων, η Οπτικοποίηση των Δεδομένων και των Αποτελεσμάτων καθώς και η Εξαγωγή Δεδομένων.

4.3.1 Παράδειγμα Διαχείρισης Δεδομένων

Παρακάτω παρατίθεται συνοπτικά η διαδικασία που ακολουθεί ο χρήστης χρησιμοποιώντας τα εργαλεία της συγκεκριμένης υποενότητας.

Βήμα 1: Εισαγωγή Αρχείου CSV

Ο χρήστης επιλέγει την εισαγωγή δεδομένων από το μενού και φορτώνει το CSV αρχείο. Έπειτα, μπορεί να δει τα δεδομένα ώστε να βεβαιωθεί ότι προβάλλονται σωστά.



Εικόνα 6. Εισαγωγή αρχείου CSV

Βήμα 2: Προ-επεξεργασία Δεδομένων

Εφαρμογή εργαλείων καθαρισμού για την αντιμετώπιση ελλιπών τιμών.

Βήμα 3: Διαχωρισμός σε Εκπαιδευτικά και Δοκιμαστικά Σύνολα

Βήμα 4: Ανάλυση και Οπτικοποίηση Αποτελεσμάτων

Αφού ο χρήστης πρώτα αναλύσει τα αποτελέσματα, συνεχίζει με την οπτικοποίηση των επιδόσεων του μοντέλου μέσω γραφημάτων.

Βήμα 5: Αποθήκευση και Εξαγωγή Αποτελεσμάτων

Τέλος, ο χρήστης προχωρά με την εξαγωγή των αποτελεσμάτων του πειράματός του σε γραφήματα.

4.4 Σχεδιασμός πειραμάτων (Experiments, off-line λειτουργία)

Η λειτουργία αυτή επιτρέπει το σχεδιασμό πειραμάτων για την επίλυση διάφορων προβλημάτων παλινδρόμησης, ταξινόμησης και μάθησης χωρίς επίβλεψη. Τα πειράματα μοντελοποιούνται γραφικά και απεικονίζονται από γραφήματα με συνδέσεις κόμβου - ακμής.

Για να ξεκινήσει ο σχεδιασμός του πειράματος πρέπει αρχικά ο χρήστης να εισάγει τα δεδομένα που θα χρησιμοποιηθούν στο πείραμα φορτώνοντας αρχεία ή σύνολα δεδομένων από την υπάρχουσα βιβλιοθήκη του KEEL. Στη συνέχεια, αφού επιλέξει τους αλγορίθμους που θα χρησιμοποιηθούν στο πείραμα, πρέπει να τους σύρει στον χώρο εργασίας και να συνδέσει τα δεδομένα με τους αλγορίθμους αλλά και άλλες τεχνικές (όπως η προ-επεξεργασία δεδομένων) δημιουργώντας μια ροή εργασιών. Οι συνδέσεις που θα δημιουργηθούν, καθορίζουν τη σειρά με την οποία θα εκτελεστούν οι διαδικασίες. Τέλος, αφού ολοκληρωθεί το πείραμα, τα αποτελέσματα παρουσιάζονται στο παράθυρο εξόδου. Και σε αυτήν την περίπτωση, τα αποτελέσματα οπτικοποιούνται μέσω διαφόρων γραφημάτων και διαγραμμάτων και τα δεδομένα των αποτελεσμάτων μπορούν να εξαχθούν για περαιτέρω ανάλυση.

4.5 Εκπαιδευτικά Πειράματα (Educational, on-line λειτουργία)

Αυτή η λειτουργία είναι παρόμοια με την προηγούμενη με τη διαφορά ότι χρησιμοποιούνται συγκεκριμένοι αλγόριθμοι, αποκλειστικά για ακαδημαϊκό σκοπό. Όταν σχεδιάζεται ένα πείραμα, ο χρήστης μπορεί να επιλέξει είτε να αποθηκεύσει το πείραμα σε ένα αρχείο XML είτε απλά να το εκτελέσει. Αν επιλέξει να το εκτελέσει, τότε το σύστημα θα εμφανίσει ένα βοηθητικό παράθυρο για τη διαχείριση και την οπτικοποίηση της εκτέλεσης κάθε αλγορίθμου. Όταν ολοκληρωθεί η εκτέλεση, το παράθυρο αυτό θα εμφανίσει τα αποτελέσματα που λαμβάνονται για κάθε αλγόριθμο σε ξεχωριστές ετικέτες οι οποίες εμφανίζουν, για παράδειγμα, τους πίνακες σύγχυσης για ταξινόμηση ή το μέσο τετραγωνικό σφάλμα για προβλήματα παλινδρόμησης.

Εν ολίγοις λοιπόν, το KEEL είναι ένα μη εμπορικό εργαλείο λογισμικού Java το οποίο παρέχει τη δυνατότητα ανάλυσης των μεθόδων εξελικτικής μάθησης που εφαρμόζονται σε προβλήματα εξόρυξης δεδομένων. Το εργαλείο αυτό απαλλάσσει τους ερευνητές από πολλές τεχνικές εργασίες και τους επιτρέπει να επικεντρωθούν στην ανάλυση των νέων μοντέλων μάθησης σε σύγκριση με τα υπάρχοντα.

5 *Πειραματική Μελέτη*

Στο παρόν κεφάλαιο θα περιγραφεί η πειραματική μελέτη η οποία πραγματοποιήθηκε χρησιμοποιώντας το KEEL και τους αλγόριθμους που περιγράφηκαν στο 3ο κεφάλαιο. Στη συνέχεια, ακολουθεί ο τρόπος υλοποίησης των πειραμάτων καθώς και η στατιστική μελέτη των μετρήσεων που πραγματοποιήθηκαν προκειμένου να βρεθεί η ακρίβεια (Accuracy).

5.1 Σύνολα Δεδομένων

Στην υποενότητα αυτή αναλύονται τα σύνολα δεδομένων που χρησιμοποιήθηκαν προκειμένου να υλοποιηθεί μια σειρά από πειράματα μέσω του KEEL. Συγκεκριμένα, χρησιμοποιήθηκαν 12 διαφορετικά σύνολα δεδομένων.

Το σύνολο δεδομένων «**balance»** περιλαμβάνει πληροφορίες σχετικά με την ισορροπία μιας ζυγαριάς και χρησιμοποιείται για την πρόβλεψη της κατάστασης ισορροπίας της ζυγαριάς. Κάθε παράδειγμα ταξινομείται ως προς το αν η ζυγαριά έχει κλίση προς τα δεξιά, προς τα αριστερά ή αν είναι ισορροπημένη. Τα χαρακτηριστικά είναι τα εξής:

- Left-Weight: Βάρος στο αριστερό δίσκο της ζυγαριάς
- Left-Distance: Απόσταση του αριστερού δίσκου από το κέντρο της ζυγαριάς
- Right-Weight: Βάρος στο δεξί δίσκο της ζυγαριάς
- Right-Distance: Απόσταση του δεξιού δίσκου από το κέντρο της ζυγαριάς

Η ετικέτα υποδεικνύει την ισορροπία της ζυγαριάς με τις μεταβλητές

- B: Η ζυγαριά γέρνει προς τα αριστερά
- R: Η ζυγαριά γέρνει προς τα δεξιά
- L: Η ζυγαριά είναι ισορροπημένη

Το σύνολο δεδομένων **«diabetes»** χρησιμοποιείται κυρίως για την ανάπτυξη και την αξιολόγηση αλγορίθμων ταξινόμησης που προβλέπουν την εμφάνιση διαβήτη με βάση ορισμένα ιατρικά χαρακτηριστικά. Το σύνολο δεδομένων περιλαμβάνει 8 ανεξάρτητες μεταβλητές (χαρακτηριστικά):

1. **Pregnancies**: Αριθμός κυήσεων,
2. **Glucose**: Συγκέντρωση γλυκόζης 2 ώρες μετά από εξέταση ανοχής στη γλυκόζη,
3. **BloodPressure**: Διαστολική αρτηριακή πίεση (mm Hg),
4. **SkinThickness**: Πάχος δέρματος στον τρικέφαλο (mm),
5. **Insulin**: Επίπεδα ινσουλίνης 2 ώρες μετά από εξέταση (μIU/ml),
6. **BMI**: Δείκτης μάζας σώματος (weight in kg/ (height in m) ^2),
7. **DiabetesPedigreeFunction**: Λειτουργία γενετικής διαβήτη,
8. **Age**: Ηλικία (Ετη)

Η ετικέτα είναι δυαδική υποδεικνύοντας αν ο ασθενής έχει διαγνωστεί με διαβήτη (1) ή όχι (0). Οπότε, ο αριθμός των στιγμιοτύπων είναι 769, ο αριθμός των στηλών είναι 9 (χαρακτηριστικά και ετικέτα) και ο αριθμός των κλάσεων 2 (διαβητικός ή μη διαβητικός).

Το **«ecoli»** περιλαμβάνει πληροφορίες σχετικά με τις πρωτεΐνες του βακτηρίου E. coli και χρησιμοποιείται για την πρόβλεψη της τοποθέτησης των πρωτεϊνών εντός του κυττάρου. Περιλαμβάνει τα εξής χαρακτηριστικά:

- **Sequence Name**
- **mcg**: McGeoch για την αναγνώριση ακολουθίας σημάτων,
- **gvh**: Von Heijne για την αναγνώριση ακολουθίας σημάτων,
- **lip**: Παρουσία λιποπρωτεϊνικής αλληλουχίας σήματος,
- **chg**: Βαθμολογία της ανάλυσης διάκρισης των πρωτεϊνών της εξωτερικής μεμβράνης,
- **aac**: Βαθμολογία της συνάρτησης διάκρισης της σύνθεσης των αμινοξέων,
- **alm1**: Βαθμολογία του προγράμματος πρόβλεψης της περιοχής που εκτείνεται στη μεμβράνη ALOM,
- **alm2**: Βαθμολογία του προγράμματος ALOM μετά τον αποκλεισμό των υποθετικών περιοχών σήματος που μπορούν να διασπαστούν

Το πλήθος των εγγραφών είναι 336, το πλήθος των στηλών είναι είναι 8 και ο αριθμός κλάσεων είναι επίσης 8.

Το σύνολο δεδομένων **«iris»** είναι από τα πρώτα σύνολα δεδομένων που χρησιμοποιούνται στη βιβλιογραφία σχετικά με τις μεθόδους ταξινόμησης και χρησιμοποιούνται ευρέως στη στατιστική και τη μηχανική μάθηση. Το σύνολο δεδομένων περιέχει 3 κλάσεις των 50 περιπτώσεων η καθεμία, όπου κάθε κλάση αναφέρεται σε έναν τύπο φυτού ίριδας. Η μία κλάση είναι γραμμικά διαχωρίσιμη από τις άλλες 2- οι τελευταίες δεν είναι γραμμικά διαχωρίσιμες μεταξύ τους. Άρα περιέχει συνολικά 150 εγγραφές και οι τρεις κλάσεις από τις οποίες αποτελείται είναι η Iris-setosa, η Iris-versicolor και η Iris-virginica. Το σύνολο δεδομένων περιλαμβάνει τις ακόλουθες στήλες (χαρακτηριστικά): **Sepal Length (cm)**:

Μήκος κάλυκα σε εκατοστά, **Sepal Width (cm)**: Πλάτος κάλυκα σε εκατοστά, **Petal Length (cm)**: Μήκος πετάλου σε εκατοστά και **Petal Width (cm)**: Πλάτος πετάλου σε εκατοστά.

Το σύνολο δεδομένων **«Monks»** αποτελείται από τρία υποσύνολα, το **Monks 1** με αριθμό εγγραφών 556 και αριθμό χαρακτηριστικών (στηλών) 7, το **Monks 2** με 601 εγγραφές και 7 στήλες και τον **Monks 3** με 554 εγγραφές και 7 στήλες. Κάθε υποσύνολο περιλαμβάνει πληροφορίες για το χαρακτηριστικό και την κατηγορία των δεδομένων. Το «Monks» χρησιμοποιείται για τη δοκιμή και τη σύγκριση αλγορίθμων ταξινόμησης και μηχανικής μάθησης. Αποτελούνται από δύο κλάσεις, 0 και 1, και περιέχουν συμβολικά δεδομένα. Κάθε εγγραφή αποτελείται από χαρακτηριστικά που είναι κατηγορικά και χρησιμοποιούνται για ταξινόμηση σε μία από αυτές τις δύο κλάσεις.

Το **«pbc»** είναι ένα σύνολο δεδομένων το οποίο περιέχει ιατρικά δεδομένα ασθενών με μια χρόνια ασθένεια του ήπατος, την πρωτοπαθή χολική κίρρωση (Primary Biliary Cirrhosis). Περιλαμβάνει 19 στήλες με δημογραφικές, κλινικές και εργαστηριακές μετρήσεις και 418 εγγραφές με πληροφορίες σχετικά με τους ασθενείς. Οι στήλες είναι: ηλικία, φύλο, χολερυθρίνη, αλκαλική φωσφατάση, λευκωματίνη, χοληστερόλη, και άλλες βιοχημικές μετρήσεις καθώς επίσης η κατάσταση επιβίωσης και ο χρόνος παρακολούθησης.

Το **«penbased»** χρησιμοποιείται για την αναγνώριση χειρόγραφων ψηφίων. Το σύνολο δεδομένων περιέχει χαρακτηριστικά που αντιστοιχούν σε χειρόγραφα ψηφία από 0 έως 9 (αυτές είναι οι κλάσεις). Αποτελείται από 10,992 εγγραφές, 16 στήλες και 10 κλάσεις (ψηφία 0 – 9).

Το **«satellite»** περιέχει δεδομένα που αφορούν την κατάταξη των εδαφικών κατηγοριών από δορυφορικές εικόνες. Περιέχει 6435 εγγραφές, 36 στήλες και αποτελείται από 6 κλάσεις. Οι κλάσεις είναι οι εξής: Κλάση 1: Κόκκινο καλλιεργημένο έδαφος (Red Soil), Κλάση 2: Ελαφρώς καλλιεργημένο έδαφος (Cotton Crop), Κλάση 3: Περιοχές με δέντρα (Grey Soil), Κλάση 4: Εδάφη με αμμώδη καλλιέργεια (Damp Grey Soil), Κλάση 5: Μαύρο χώμα (Soil with Vegetation Stubble) και Κλάση 6: Χώμα με πέτρες και βράχους (Very Damp Grey Soil).

Το **«segmentation»** είναι ένα σύνολο δεδομένων που ως στόχο έχει την ταξινόμηση των pixels σε μια εικόνα, σε διαφορετικές κατηγορίες τμημάτων. Αποτελείται από 2,310 εγγραφές, 19 στήλες και 7 κλάσεις (Brickface, Sky, Foliage, Cement, Window, Path, Grass). Κάθε εγγραφή στο σύνολο δεδομένων αντιπροσωπεύει τα χαρακτηριστικά ενός pixel και περιέχει 19 χαρακτηριστικά που περιγράφουν τις ιδιότητες του pixel. Αυτά τα χαρακτηριστικά περιλαμβάνουν το Χρώμα (Color), την Ένταση (Intensity): φωτεινότητα του pixel, τα Στατιστικά Χρώματος (Color Statistics): μέση τιμή, διασπορά και άλλες στατιστικές ιδιότητες των καναλιών χρώματος στην περιοχή του pixel και Θέση (Position): θέση του pixel στην εικόνα (συντεταγμένες x και y).

Το **«spambase»** είναι ένα σύνολο δεδομένων που χρησιμοποιείται για την ανίχνευση ανεπιθύμητων μηνυμάτων (spam) και ταξινομεί αν ένα email είναι spam ή όχι. Αποτελείται από 4,601 εγγραφές, 57 στήλες και 2 κλάσεις (spam, non-spam). Τα χαρακτηριστικά του συνόλου δεδομένων περιλαμβάνουν διάφορα στατιστικά για τις λέξεις και τους χαρακτήρες στα email. Αυτό το σύνολο δεδομένων χρησιμοποιείται ευρέως για την ανάλυση και την επεξεργασία εικόνας αλλά και για τον πειραματισμό σε ταξινόμηση εικόνας και αναγνώριση προτύπων.

Το σύνολο δεδομένων **«wine»** περιέχει αποτελέσματα χημικής ανάλυσης κρασιών που καλλιεργούνται στην ίδια περιοχή της Ιταλίας αλλά προέρχονται από τρεις διαφορετικές ποικιλίες. Η ανάλυση προσδιορίζει τις ποσότητες 13 συστατικών που βρέθηκαν σε κάθε έναν από τους τρεις τύπους κρασιών.

Το **«New – Thyroid»** είναι ένα σύνολο δεδομένων το οποίο περιέχει πληροφορίες για την ανίχνευση και διάγνωση ασθενειών του θυρεοειδούς. Έχει 215 εγγραφές, 5 στήλες (TSH, T3, TT4, T4U, FTI) και 3 κλάσεις (Υπερθυρεοειδισμός, Υποθυρεοειδισμός, Υγιής θυρεοειδής) και αποτελείται από αριθμητικά δεδομένα.

Τέλος, το **«bupa»** είναι ένα σύνολο δεδομένων με πληροφορίες σχετικές με ηπατικές διαταραχές και κυρίως χρησιμοποιείται για την πρόβλεψη της παρουσίας ηπατικών ασθενειών. Αποτελείται από 345 εγγραφές, 6 στήλες (Mcv, Alkphos, Sgpt, Sgot, Gamma-GT, Drinks) και 2 κλάσεις. Η κλάση είναι δυαδική και καθορίζει εάν ο ασθενής έχει ηπατική διαταραχή ή όχι (1 για ηπατική διαταραχή, 2 για μη ηπατική διαταραχή).

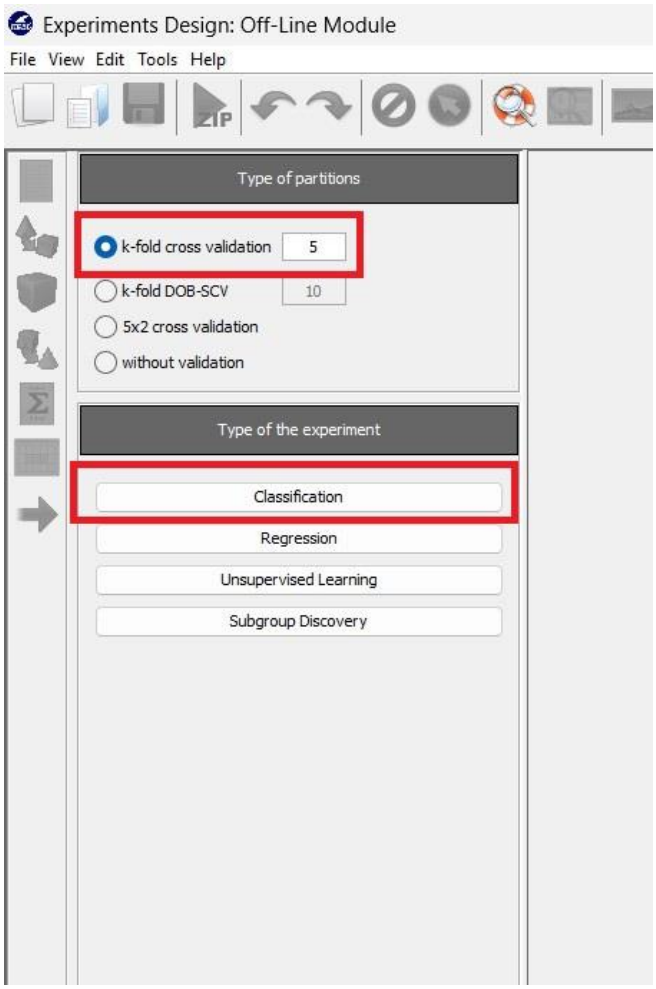
Όνομα Συνόλου Δεδομένων (Datasets)	Πλήθος Εγγραφών	Πλήθος Γραμμών	Πλήθος Στηλών
balance	625	625	5
diabetes	768	768	9
ecoli	336	336	8
iris	150	150	4
monks	556	556	7
pbc	418	418	19
penbased	10,992	10,992	16
satellite	6,435	6,435	36
segmentation	2,310	2,310	19
spambase	4,601	4,601	57
wine	178	178	13
New - Thyroid	215	215	5
bupa	345	345	6

Πίνακας 7. Πληροφορίες συνόλων δεδομένων

5.2 Εγκαθίδρυση Πειραμάτων

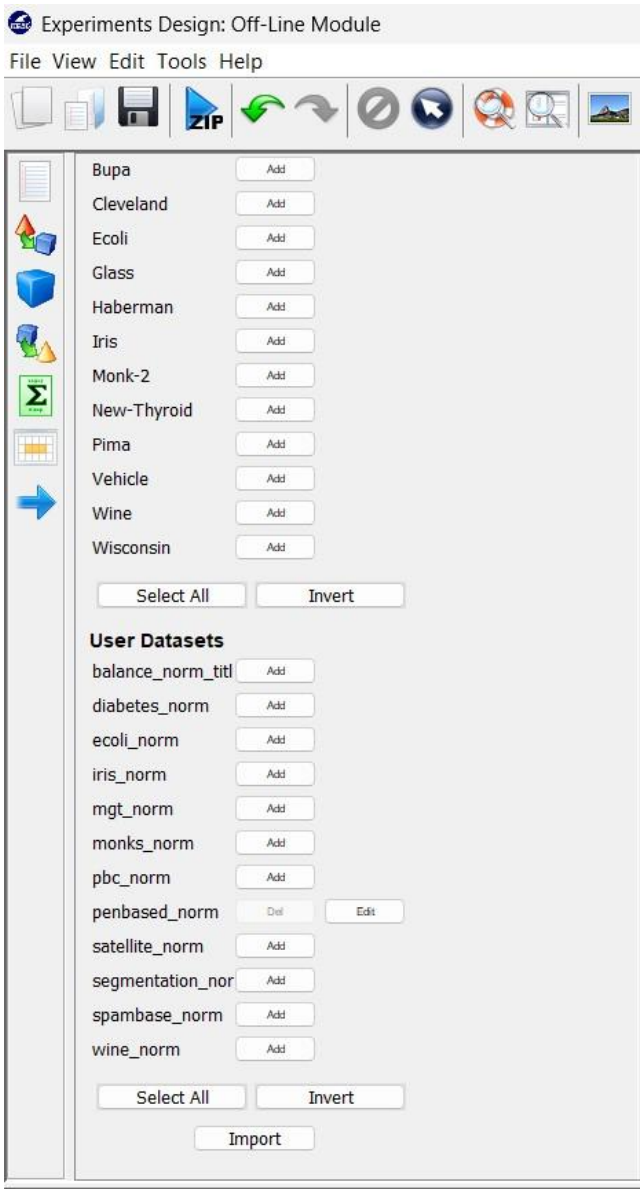
Στο παρόν κεφάλαιο θα παρουσιαστεί ο τρόπος εκτέλεσης των πειραμάτων που διενεργήθηκαν, στα πλαίσια υλοποίησης της παρούσας διπλωματικής εργασίας, χρησιμοποιώντας το KEEL.

Από την αρχική οθόνη του KEEL, ο χρήστης επιλέγει «Experiments», όπου στη συνέχεια, όπως φαίνεται στην Εικόνα 7, εμφανίζεται το κύριο παράθυρο της συγκεκριμένης λειτουργίας. Το k-fold έχει επιλεχθεί να είναι 5 και ο τύπος του πειράματος είναι η Κατηγοριοποίηση (Classification). Αφού ολοκληρωθεί η επιλογή του πειράματος, θα εμφανιστεί ο πίνακας επιλογής συνόλων δεδομένων.



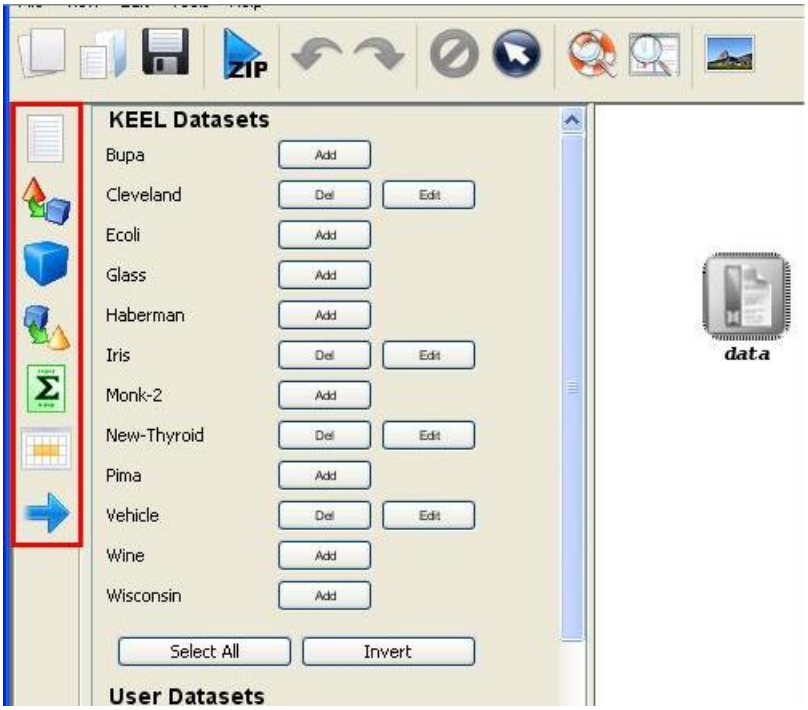
Εικόνα 7. Οθόνη πειραμάτων

Στη συνέχεια, ο πίνακας επιλογής συνόλων δεδομένων εμφανίζει τα διαθέσιμα σύνολα δεδομένων για το τρέχον πείραμα. Το περιεχόμενό του εξαρτάται από τον τύπο του πειράματος που έχει ήδη επιλεγεί. Το επόμενο βήμα είναι η επιλογή των επιθυμητών συνόλων δεδομένων από τον πίνακα. Σε περίπτωση που εξυπηρετεί τον χρήστη, μπορεί να εισάγει όλα τα διαθέσιμα σύνολα.



Εικόνα 8. Σύνολα δεδομένων

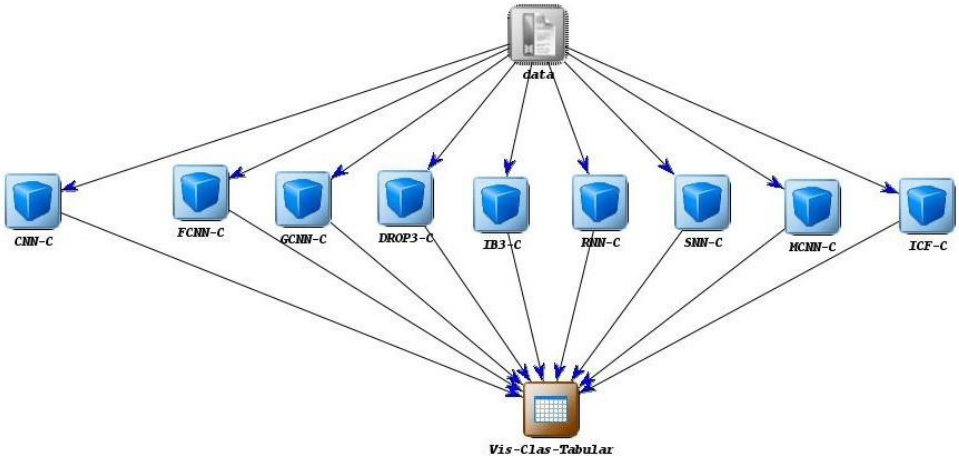
Όταν επιλεγεί το απαραίτητο σύνολο δεδομένων, η διαδικασία σχεδιασμού του πειράματος μπορεί να συνεχιστεί. Για να γίνει αυτό, ο χρήστης πρέπει να κάνει κλικ στον λευκό πίνακα γραφημάτων για να ορίσει τον κόμβο των συνόλων δεδομένων του πειράματος.



Εικόνα 9. Σχεδιασμός πειράματος – Φάση 1

Το γράφημα του πειράματος δείχνει τα στοιχεία του τρέχοντος πειράματος και περιγράφει τις σχέσεις μεταξύ τους. Ο χρήστης μπορεί να προσθέσει νέα στοιχεία χρησιμοποιώντας το μενού που βρίσκεται αριστερά. Όπως φαίνεται και στην Εικόνα 9, η οθόνη γραφημάτων περιλαμβάνει τις εξής κατηγορίες: τα Σύνολα δεδομένων, όπου γίνεται εισαγωγή των συνόλων δεδομένων των πειραμάτων, οι Μέθοδοι προ-επεξεργασίας. Μέσω της κατηγορίας αυτής γίνεται η προ-επεξεργασία των αρχικών συνόλων δεδομένων. Ακολουθεί η κατηγορία Τυπικές μέθοδοι εξόρυξης δεδομένων και οι Μέθοδοι μετα-επεξεργασίας όπου πραγματοποιείται η μετα-επεξεργασία των αποτελεσμάτων των τυποποιημένων μεθόδων. Οι Στατιστικές δοκιμές είναι χρήσιμες για την αντιπαραβολή των αποτελεσμάτων που επιτεύχθηκαν στο πείραμα. Η Οπτικοποίηση βοηθά στην εμφάνιση των αποτελεσμάτων των πειραμάτων και τέλος, οι Συνδέσεις μεταξύ των στοιχείων του πειράματος.

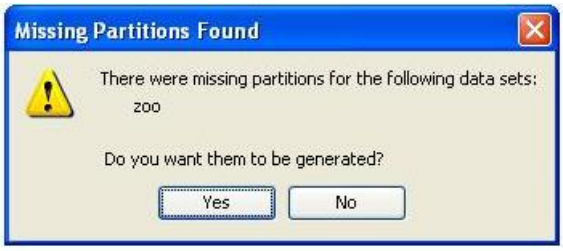
Αφού ολοκληρωθεί το «χτίσιμο» του πειράματος, έχει τη μορφή της παρακάτω εικόνας (Εικόνα 10).



Εικόνα 10. Σχεδιασμός πειράματος – Φάση 2

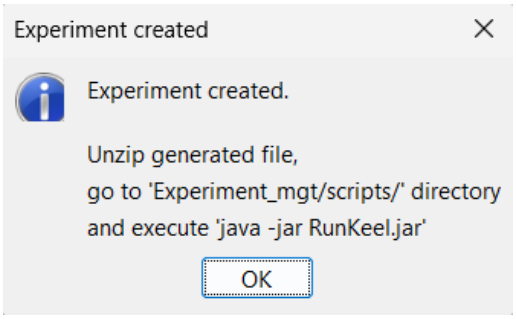
Μόλις σχεδιαστεί ένα πείραμα, ο χρήστης μπορεί να το δημιουργήσει μέσω της επιλογής Run Experiment (Εκτέλεση πειράματος) του μενού 'Tools' (Εργαλεία). Επιπλέον, είναι δυνατή η χρήση του κουμπιού **ZIP** της γραμμής εργαλείων.

Σε αυτό το σημείο, το KEEL θα εκτελέσει διάφορους ελέγχους σχετικά με την πληρότητα του πειράματος. Πρώτον, εάν διαπιστώσει ότι λείπουν partitions για ορισμένα από τα σύνολα δεδομένων που χρησιμοποιήθηκαν, θα εμφανιστεί το ακόλουθο παράθυρο διαλόγου που επιτρέπει την αναδημιουργία τους.



Εικόνα 11. Μήνυμα για ελλιπή partitions

Η διαδικασία δημιουργίας παράγει ένα ZIP αρχείο που περιέχει όλα τα στοιχεία που απαιτούνται για την εκτέλεση του πειράματος. Εφόσον η δημιουργία του πειράματος ολοκληρωθεί με επιτυχία, θα εμφανιστεί το ακόλουθο μήνυμα (Εικόνα 12).



Εικόνα 12. Επιτυχής δημιουργία πειράματος

Μετά από αυτό το βήμα, ο χρήστης κάνει δεξί κλικ μέσα στον φάκελο scripts, επιλέγει Open in Terminal και προσθέτει την εντολή **java -jar RunKeel.jar** για να τρέξει το πείραμα.

Εικόνα 13. Εκτέλεση Java εντολής

Μόλις ολοκληρωθεί η εκτέλεση του πειράματος τα αποτελέσματα εμφανίζονται στον φάκελο Results.

5.3 Πειραματικά αποτελέσματα

Στον πίνακα που ακολουθεί (Πίνακας 8), περιγράφονται οι μετρήσεις ακρίβειας ανά σύνολο δεδομένων και ανά αλγόριθμο. Σύμφωνα με τις μετρήσεις που προκύπτουν, τα σύνολα δεδομένων που έχουν υψηλότερα ποσοστά μετρήσεων είναι κυρίως τα: iris, satellite, segmentation και wine ενώ αυτά που εμφανίζουν χαμηλότερα ποσοστά ακρίβειας είναι τα: bupa, diabetes και penbased.

Σε ότι αφορά τους αλγόριθμους, σε γενικές γραμμές, αυτοί που ανταποκρίνονται καλύτερα είναι οι RNN, FCNN και CNN. Αναλυτικότερα, θα αναφερθεί παρακάτω για ποιον αλγόριθμο προκύπτει η καλύτερη μέτρηση της ακρίβειας και για ποιο σύνολο δεδομένων: για το **«balance»** καλύτερος αλγόριθμος είναι ο DROP3 με 0.82, για το **«diabetes»** ο RNN με 0.69, για το **«ecoli»** ο RNN με 0.82, για το «iris» είναι ο RNN με 0.96, για το **«monks»** είναι ο RNN με 0.87, για το **«penbased»** είναι ο GCNN με 0.78, για το **«satellite»** είναι ο IBL3 με 0.93, για το «segmentation» είναι οι RNN και FCNN με 0.95 και **«spambase»** είναι ο FCNN με 0.95, για το **«wine»** είναι οι RNN, GCNN και ICF με 0.95, για το **«New-Thyroid»** είναι ο ICF με 1 και για το **«bupa»** είναι οι αλγόριθμοι RNN, SNN και MCNN. Θα ακολουθήσει στατιστική μελέτη όπου οι συγκεκριμένες μετρήσεις θα σχολιαστούν εκτενέστερα.

	Accuracy								
	CNN	**RNN**	**FCNN**	**GCNN**	**IBL3**	**SNN**	**MCNN**	**ICF**	**DROP3**
Dataset									
balance	0,72	0,79	0,71	0,79	0,73	0,6	0,68	0,72	**0,82**
diabetes	0,68	**0,69**	0,64	0,67	0,66	0,54	0,64	0,66	0,68
ecoli	0,78	**0,82**	0,76	0,77	0,74	0,47	0,79	0,79	0,79
iris	0,92	**0,96**	0,94	0,92	0,94	0,86	0,92	0,89	0,94
monks	0,83	**0,87**	0,93	0,68	0,79	0,59	0,86	0,68	0,73
penbased	0,69	0,72	0,71	**0,78**	0,65	0,67	0,63	0,67	0,6
satellite	0,83	0,7	0,73	0,9	**0,93**	0,87	0,82	0,79	0,8
segmentation	0,94	**0,95**	**0,95**	0,67	0,92	0,76	0,91	0,81	0,9
spambase	0,89	0,92	**0,95**	0,71	0,87	0,82	0,85	0,79	0,86
wine	0,92	**0,95**	0,92	**0,95**	0,92	0,9	0,93	**0,95**	0,9
New-Thyroid	0,9	0,9	0,95	0,9	0,86	0,9	0,9	**1**	0,95
bupa	0,57	**0,65**	0,6	0,62	0,6	**0,65**	**0,65**	0,45	0,57

Πίνακας 8. Πίνακας μετρήσεων ακρίβειας

5.4 Στατιστική μελέτη

Σε αυτό το σημείο θα πραγματοποιηθεί η στατιστική μελέτη των μετρήσεων που προέκυψαν κατά τη διάρκεια εκτέλεσης της πειραματικής διαδικασίας. Πιο συγκεκριμένα, θα υλοποιηθεί το Friedman Test και το Wilcoxon Test.

Το Friedman Test είναι μια μη παραμετρική στατιστική δοκιμή που ως στόχο έχει τη σύγκριση πολλών ομάδων δεδομένων και εφαρμόζεται όταν κάποιος θέλει να εξετάσει αν υπάρχουν στατιστικά σημαντικές διαφορές μεταξύ των ποσοστών που προκύπτουν σε περισσότερες από δύο ομάδες. Για την υλοποίηση της συγκεκριμένης στατιστικής μελέτης, χρησιμοποιήθηκε με σκοπό να υλοποιηθεί η σειρά κατάταξης των αλγορίθμων, από το υψηλότερο προς το χαμηλότερο ποσοστό. Ο αλγόριθμος που εμφανίζει το υψηλότερο ποσοστό είναι αυτός ο οποίος έχει καλύτερη ακρίβεια σε σχέση με τους υπόλοιπους.

Από την άλλη το Wilcoxon Test είναι και αυτό μια μη παραμετρική στατιστική δοκιμή που όμως αποσκοπεί στη σύγκριση δύο δειγμάτων μεταξύ τους. Χρησιμοποιείται για να συγκριθούν οι μετρήσεις δύο δειγμάτων ώστε να συμπεράνει ο χρήστης αν προέρχονται από την ίδια κατανομή ή αν υπάρχουν στατιστικά σημαντικές διαφορές μεταξύ τους.

RNN	6,25
FCNN	5,71
CNN	5,33
GCNN	5,13
DROP3	5,08
IBL3	5
ICF	4,92
MCNN	4,71
SNN	2,88

Πίνακας 9. Friedman Test

Όπως φαίνεται και στον παραπάνω πίνακα, ο RNN βρίσκεται υψηλότερα στην λίστα κατάταξης με 6.25, ενώ ο SNN βρίσκεται χαμηλότερα με 2.88.

Στη συνέχεια ακολουθεί η ανάλυση του Wilcoxon Test.

CNN x RNN	0,514
CNN x FCNN	0,373
CNN x GCNN	0,799
CNN x IBL3	0,265
CNN x SNN	0,033
CNN x MCNN	0,385
CNN x ICF	0,385
CNN x DROP3	0,646
DROP3 x ICF	0,875
DROP3 x MCNN	1
DROP3 x MCNN	0,075
DROP3 x IBL3	0,789
DROP3 x GCNN	0,665
DROP3 x FCNN	0,61
DROP3 x RNN	0,638
DROP3 x CNN	0,646
IBL3 X SNN	0,019
IBL3 x ICF	0,593
IBL3 x MCNN	0,969
IBL3 x SNN	0,019
IBL3 x GCNN	0,969
IBL3 x FCNN	0,259
IBL3 x RNN	0,271
RNN x ICF	0,721
RNN x FCNN	0,875
RNN x GCNN	0,314
RNN x IBL3	0,271
RNN x CNN	0,037
RNN x MCNN	0,674

Πίνακας 10. Wilcoxon Test

Στο Wilcoxon test εάν η τιμή αυτή είναι μικρότερη από 0.05, τότε η διαφορά είναι στατιστικά σημαντική. Τα αποτελέσματα δείχνουν ότι στην πλεοψηφία των μετρήσεων που διεξήχθησαν δεν υπάρχει στατιστική διαφορά στην ακρίβεια μεταξύ των συγκρινόμενων αλγορίθμων, παρά μόνο στα ζεύγη CNN x SNN με 0.033, IBL3 x SNN με 0.019 και ο RNN x CNN με 0.037. Αυτό σημαίνει ότι δεν μπορούμε να απορρίψουμε την μηδενική υπόθεση ότι οι δύο αλγόριθμοι έχουν παρόμοια απόδοση.

Σε ό,τι αφορά το ποσοστό μείωσης δεδομένων, το KEEL δε δίνει τη δυνατότητα μέτρησής του. Παρ'όλα αυτά, σύμφωνα με τις πηγές [9], [13], [25] οι αλγόριθμοι που μελετήθηκαν εμφανίζουν τα παρακάτω ποσοστά με σειρά κατάταξης από το υψηλότερο στο χαμηλότερο ποσοστό: ο RNN 0.92, ο MCNN 0.91, ο DROP3 0.82, ο SNN 0.75, ο ICF 0.71, ο IBL3 0.71, ο FCNN 0.63, ο CNN 0.57 και ο GCNN 0.45.

Σύμφωνα λοιπόν με τις παραπάνω μετρήσεις, ο αλγόριθμος RNN είναι αυτός που εμφανίζει, και σε αυτήν την περίπτωση, το καλύτερο ποσοστό ένω την χαμήλοτερη θέση στη σειρά κατάξης έχει ο GCNN.

RNN	+++++
MCNN	+++++
DROP3	++++
SNN	+++
ICF	+++
IBL3	+++
FCNN	++
CNN	++
GCNN	+

Πίνακας 11. Ποσοστό μείωσης δεδομένων (Reduction Rate)

6 *Συμπεράσματα και μελλοντικές επεκτάσεις*

6.1 Συμπεράσματα

Ο κύριος στόχος της παρούσας διπλωματικής εργασίας ήταν η βιβλιογραφική ανασκόπηση ορισμένων αλγορίθμων επιλογής προτύπων καθώς και η υλοποίηση πειραματικής μελέτης χρησιμοποιώντας το KEEL. Συνολικά χρησιμοποιήθηκαν εννέα αλγόριθμοι και δώδεκα σύνολα δεδομένων.

Αρχικά, πραγματοποιήθηκε η βιβλιογραφική ανασκόπηση των αλγορίθμων που χρησιμοποιήθηκαν ώστε να γίνει σαφής και κατανοητός ο τρόπος λειτουργίας τους, στη συνέχεια πραγματοποιήθηκε η πειραματική μελέτη και τέλος η στατιστική μελέτη συγκρίνοντας όλους τους αλγόριθμους μεταξύ τους και έπειτα σε ζεύγη.

Με βάση τη στατιστική μελέτη που πραγματοποιήθηκε, οι αλγόριθμοι που δοκιμάστηκαν παρουσίασαν διάφορα επίπεδα απόδοσης, με ορισμένους να υπερέχουν σε συγκεκριμένα σύνολα δεδομένων. Η ανάλυση έδειξε ότι κανένας αλγόριθμος δεν εμφανίζει στατιστική διαφορά.

Το Friedman Test έδειξε ότι ο καλύτερος αλγόριθμος με βάση την απόδοση είναι ο RNN με ποσοστό 6.25, ενώ χαμηλότερο ποσοστό, με 2.88, εμφάνισε ο SNN. Από την άλλη πλευρά, το Wilcoxon Test έδειξε ότι δεν υπάρχουν στατιστικές διαφορές ανάμεσα στα συγκρινόμενα ζεύγη αλγορίθμων που εξετάστηκαν.

Γενικά, οι αλγόριθμοι που εξετάστηκαν, όπως οι CNN, RNN και FCNN, παρουσίασαν διαφορετικά επίπεδα απόδοσης σε διαφορετικά σύνολα δεδομένων. Οι διαφορές στην απόδοση των αλγορίθμων οφείλονται σε μεγάλο βαθμό στα χαρακτηριστικά των δεδομένων, όπως η κατανομή των κλάσεων, η πολυπλοκότητα των μοτίβων και η παρουσία θορύβου.

6.2 Μελλοντικές επεκτάσεις

Η διπλωματική εργασία θα μπορούσε να επεκταθεί με την προσθήκη νέων και μεγαλύτερων συνόλων δεδομένων, δοκιμάζοντας ποικιλία αλγορίθμων με βάση τα σύνολα αυτά, με στόχο να εξεταστεί η απόδοσή τους. Επίσης, θα μπορούσαν να προστεθούν περαιτέρω παράμετροι για τον εκάστοτε αλγόριθμο, που θα έχουν ως αποτέλεσμα τη βέλτιστη απόδοσή τους καθώς και η δημιουργία περισσότερων και ακόμα πιο εύχρηστων διεπαφών. Το KEEL, παρόλο που έχει σταματήσει να ενημερώνεται, είναι ένα πολύ χρήσιμο εργαλείο δοκιμής αλγορίθμων το οποίο θα μπορούσε να εξελιχθεί τυπώνοντας ακόμα περισσότερες μετρήσεις, όπως παραδείγματος χάριν το ποσοστό μείωσης δεδομένων. Η ενημέρωσή του λοιπόν, μπορεί να φανεί ιδιαίτερα χρήσιμη καθώς μπορεί να χρησιμοποιηθεί από διάφορους χρήστες, χωρίς απαραίτητα να έχουν ιδιαίτερες γνώσεις στον προγραμματισμό, ενώ ταυτόχρονα, η χρήση του επιταχύνει τη διαδικασία εκτέλεσης πειραμάτων και δοκιμών.

7 *Βιβλιογραφία*

[1] S. Ben Meskina, “On the effect of data reduction on classification accuracy”, in *2013 3rd International Conference on Information Technology and e-Services (ICITeS)*, 2013.

[2] J. Alcalá-Fdez *et al.*, “KEEL: a software tool to assess evolutionary algorithms for data mining problems”, *Soft Comput.*, vol. 13, no. 3, pp. 307–318, 2009.

[3] N. Bhatia and Vandana, “Survey of nearest neighbor techniques,” *arXiv [cs.CV]*, 2010.

[4] M.-A. Amal and B.-A. Ahmed, “Survey of nearest neighbor condensing techniques”, *Int. J. Adv. Comput. Sci. Appl.*, vol. 2, no. 11, 2011.

[5] P. Filippakis, S. Ougiaroglou, and G. Evangelidis, “Condensed nearest neighbour rules for multi-label datasets”, in *International Database Engineered Applications Symposium Conference*, 2023.

[6] C.-H. Chou, B.-H. Kuo, and F. Chang, “The generalized condensed nearest neighbor rule as A data reduction method”, in *18th International Conference on Pattern Recognition (ICPR’06)*, 2006.

[7] G. Gates, “The reduced nearest neighbor rule (Corresp.)”, *IEEE Trans. Inf. Theory*, vol. 18, no. 3, pp. 431–433, 1972.

[8] K. Gowda and G. Krishna, “The condensed nearest neighbor rule using the concept of mutual nearest neighborhood (Corresp.),” *IEEE Trans. Inf. Theory*, vol. 25, no. 4, pp. 488–490, 1979.

[9] F. Angiulli, “Fast condensed nearest neighbor rule”, in *Proceedings of the 22nd international conference on Machine learning - ICML ’05*, 2005.

[10] S. Ougiaroglou, “Algorithms and techniqes for efficient and effective nearest neighbours classification”, *Ihu.gr*. [Online]. Available: https://people.iee.ihu.gr/~stoug/papers/phd.pdf.

[11] A. Søgaard, “Semisupervised condensed nearest neighbor for part-of-speech tagging”,

Acm.org. [Online]. Available: https://dl.acm.org/doi/pdf/10.5555/2002736.2002748.

[12] T. Kohonen, "Learning Vector Quantization", in *Self-Organizing Maps*, Berlin, Heidelberg: Springer Berlin Heidelberg, 2001, pp. 245–261.

[13] S. García, J. Derrac, J. R. Cano, and F. Herrera, "Prototype selection for nearest neighbor classification: taxonomy and empirical study", *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 417–435, 2012.

[14] I. Triguero, J. Derrac, S. Garcia, and F. Herrera, "A taxonomy and experimental study on prototype generation for nearest neighbor classification", *IEEE Trans. Syst. Man Cybern. C Appl. Rev.*, vol. 42, no. 1, pp. 86–100, 2012.

[15] V. Susheela Devi and M. N. Murty, "An incremental prototype set building technique", *Pattern Recognit.*, vol. 35, no. 2, pp. 505–513, 2002.

[16] H. Brighton and C. Mellish, "Advances in instance selection for instance-based learning algorithms", *Data Min. Knowl. Discov.*, vol. 6, no. 2, pp. 153–172, 2002.

[17] R. M. O. Cruz, R. Sabourin, and G. D. C. Cavalcanti, "Analyzing different prototype selection techniques for dynamic classifier and ensemble selection", *Arxiv.org*. [Online]. Available: http://arxiv.org/abs/1811.00677.

[18] H. A. Fayed, S. R. Hashem, and A. F. Atiya, "Self-generating prototypes for pattern classification", *Pattern Recognit.*, vol. 40, no. 5, pp. 1498–1509, 2007.

[19] J. Calvo-Zaragoza, J. J. Valero-Mas, and J. R. Rico-Juan, "Prototype generation on structural data using dissimilarity space representation", *Neural Comput. Appl.*, vol. 28, no. 9, pp. 2415–2424, 2017.

[20] P. Domingos, "Unifying instance-based and rule-based induction", *Mach. Learn.*, vol. 24, no. 2, pp. 141–168, 1996.

[21] D. W. Aha, D. Kibler, and M. K. Albert, "Instance-based learning algorithms", *Mach. Learn.*, vol. 6, no. 1, pp. 37–66, 1991.

[22] M. Lindenbaum, S. Markovitch, and D. Rusakov, "Selective sampling for nearest neighbor classifiers", *Mach. Learn.*, vol. 54, no. 2, pp. 125–152, 2004.

[23] C. E. Brodley, "Addressing the selective superiority problem: Automatic algorithm/model class selection", in *Machine Learning Proceedings 1993*, Elsevier, 1993, pp. 17–24.

[24] F. Chang and F. @iis S. E. Tw, "Adaptive prototype learning algorithms: Theoretical and experimental studies", *Jmlr.org*, 2006. [Online]. Available: https://jmlr.org/papers/volume7/chang06a/chang06a.pdf.

[25] D. R. Wilson and T. R. Martinez, “Reduction techniques for instance-based learning algorithms”, *Mach. Learn.*, vol. 38, no. 3, pp. 257–286, 2000.

[26] P. D. Rosero-Montalvo *et al.*, “Sign language recognition based on intelligent glove using machine learning techniques”, in *2018 IEEE Third Ecuador Technical Chapters Meeting (ETCM)*, 2018.

[27] S. Ougiaroglou, P. Filippakis, G. Fotiadou, and G. Evangelidis, “Data reduction via multi-label prototype generation”, *Neurocomputing*, vol. 526, pp. 1–8, 2023.